NASA-CR-179513-vol II

FURTHER DEVELOPMENT OF THE DYNAMIC GAS TEMPERATURE MEASUREMENT SYSTEM

VOLUME II - COMPUTER PROGRAM USER'S MANUAL

by
Dana R. Stocks

August 1986

Denny L. Elmore
Program Manager

Pratt & Whitney
Government Products Division
P.O. Box 109600
West Palm Beach, Florida 33410-9600

Prepared for
National Aeronautics and Space Administration
NASA-Lewis Research Center
Cleveland, Ohio 44135

(NASA-CR-179513-VOL-2) FURTHER DEVELOPMENT OF THE DYNAMIC GAS TEMPERATURE MEASUREMENT SYSTEM. VOLUME 2: COMPUTER PROGRAM USER'S MANUAL Final Report (Pratt and Whitney Aircraft) p N89-13771
CSCL 14B G3/35 Unclas 0181394

DATE FOR GENERAL RELEASE FEBRUARY 1989

UNITED TECHNOLOGIES PRATT&WHITNEY

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

INTRODUCTION

The Dynamic Gas Temperature Measurement System compensation software accepts digitized data from two different diameter, type B or K, thermocouples (T/C). The analysis method then determines the in situ value of an aerodynamic scaling parameter, Γ, by comparing ratios of calculated dynamic response with ratios of measured dynamic response. The value of Γ identified determines an in situ heat transfer coefficient h_g, and is used to compute a frequency response spectrum for one of the thermocouples.

Prior work in this area was performed under Contract NAS3-23154 and is reported in NASA Report NASA CR-168267, Volume II (Reference 1). The prior data analysis and compensation software was implemented on a digital computer based Hewlett Packard (HP) model 5451C Fourier Analyzer System. Under current efforts, this software has been rewritten and implemented for use on IBM mainframe computer systems which are more commonly available than the HP system.

Detailed discussions of the physical system, analytical model, and computer software are presented in this volume and in Volume I of this report under Task III activities. Computer program software restrictions and test cases are also presented. Compensated data are presented in either the time or frequency domain. Time domain data are presented as instantaneous temperature vs time (compensated or uncompensated) while frequency domain data may be presented in the forms shown in Table 1, below (compensated or uncompensated).

TABLE 1. — DATA PRESENTATION FORMS

Function	*Dimensions*	*Engineering Units*
Frequency Domain Presentations		
• Power Spectral Density (PSD)	Mean Square/Hz	K^2/Hz
• Log Power Spectral Density — 10 × log (PSD)	Mean Square/Hz	0 dB Ref 1 K^2/Hz
• Linear Power Spectral Density — Positive Square Root of PSD	rms/$\sqrt{Hz}$	K/$\sqrt{Hz}$
• Narrowband Frequency Spectrum — Positive square root of auto-spectral density (auto-power) function with narrowband signal correction for Fast Fourier Transform (FFT) windowing function applied and no normalization to per unit bandwidth	rms	K
Time Domain Presentation		
• Dynamic Temperature	Instantaneous	K

7222C

PHYSICAL MODEL

Probe Description

The dynamic temperature probe concept is shown in Figure 1. The probe employs two thermocouples of different wire diameters positioned in close proximity. The thermoelements are large enough in diameter that frequency response above a few Hz is limited by thermal inertia. When the thermocouples are exposed to the same instantaneous temperature and velocity in the gas stream, the difference in thermal responses will be governed by convective effects (proportional to wire diameter) and conductive effects (proportional to specific heat, thermal conductivity, and wire length). Many previous studies used thermoelements of sufficiently large ($\approx$ 100) length-to-diameter ratio that conduction effects may be neglected, and compensations were based on first order convective time constants. The present sensor, however, is designed for engine hot-section applications, and the smaller length-to-diameter ratios required for structural adequacy necessitates inclusion of transient conduction effects in the compensation method.

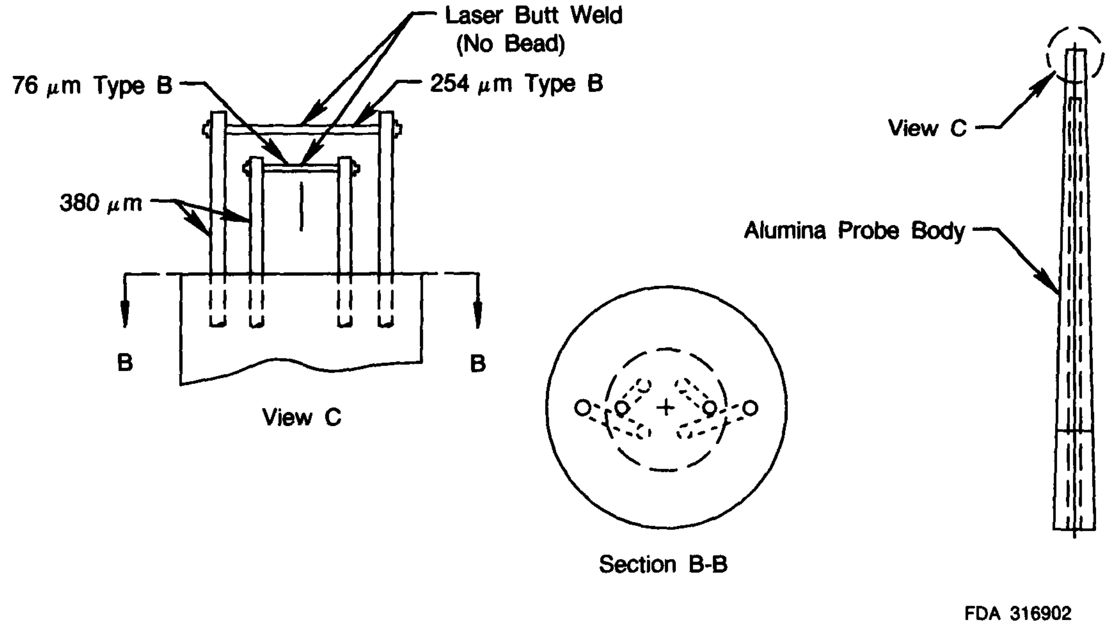


Figure 1. Dynamic Temperature Sensor Concept

For engine tests, ISA type B thermocouples were chosen based on electromotive force (EMF) output, known fabrication characteristics, high melting point, and ready availability. Detailed structural analysis revealed that allowable yield stresses for platinum-rhodium alloys constrained length-to-diameter ratios to less than 10 for the support wires and less than 15 for the thermoelements. This probe design has demonstrated more than five hour durability in a high temperature (greater than 3000°F), atmospheric pressure laboratory combustor exhaust, and more than one hour in a high temperature (greater than 2000°F), high pressure (several atmospheres) gas turbine engine combustor exhaust.

A unique feature of each thermocouple is the beadless, butt-welded thermoelement. The beadless construction allows the sensor to be modelled as a cylinder in crossflow, which simplifies the model considerably.

Data Acquisition and Processing Description

During rig tests, data were collected on a frequency modulated (FM) magnetic tape recorder. The data were reproduced post-test in an off-line data processing center to produce a digital tape for input to a mainframe IBM computer. The data analysis routines were executed in the IBM computer. A detailed description and discussion of equipment used for the data acquisition, playback, and digitization is presented in the Task 1d section of Volume 1.

ANALYTICAL MODEL

Approach

Historically the problem of compensating wire thermocouples for frequency response rests on accurate determination of the in situ film heat transfer coefficient, h_g. For wire thermocouples, three simplifications allow this to be done easily: (1) the thermocouple junction is fabricated without a bead, allowing the wire to be analyzed as a cylinder in crossflow, (2) two wires of different diameters may be co-located on the probe tip, and ratioing the different responses allows one to measure time constants τ, which are proportional to h_g for large L/D values; and (3) the thermocouple and support wires can be made long enough to eliminate conduction effects, facilitating the ratio analysis in (2). As shown in preceding reports such as Reference 1, however, probe durability requirements allowed only moderate L/D values, and initial calculations revealed that conduction effects should be included to meet accuracy goals.

The compensation approach used to include both conduction and convection effects involves ratioing signals of two different diameter beadless thermocouples. Heat transfer coefficient h_g is determined, however, by comparing finite-element conduction-convection calculations with experimental data. The calculations are done using h_g as a parameter, and require matching calculated thermocouple signal amplitude ratios at several discrete frequencies from ~6 to ~40 Hz. Agreement between calculated and experimentally observed signal amplitude ratios determines h_g. For convenience, h_g is combined with other system parameters into an aerodynamic parameter, Γ. The measured values of Γ obtained at specific frequencies are arithmetically averaged and used in the thermal model to compute the compensation frequency spectrum (gain and phase) for the smaller diameter thermocouple. The smaller diameter thermocouple should be selected for compensation since it has faster response and inherently better signal to noise characteristics at the higher frequency fluctuations. Compensation is performed digitally in the frequency domain by complex math division of the FFT spectrum of the thermocouple output by the compensation spectrum. The compensated time waveform is obtained by inverse Fourier transforming the compensated frequency spectrum.

Thermal Model

For the thermocouple probe modeled as a cylinder in crossflow, the basic thermal equation is:

$$\text{Rate of Energy Exchange} = \dot{Q}_{convection} + \dot{Q}_{conduction} + \dot{Q}_{radiation}$$

$$\frac{\partial T}{\partial t} = \frac{4h}{\rho_w C_{pw} D}(T_g - T) + \alpha \frac{\partial^2 T}{\partial X^2} + \frac{4\sigma\varepsilon}{\rho_w C_{pw} D}(T_e^4 - T^4) \tag{1}$$

Thermal analysis of the thermocouple model previously performed in Reference 1 showed that radiation effects could be neglected. It was also shown that neglecting conduction errors would typically introduce a 25 percent error (GAIN) in the compensated data. In the analysis that follows, only the radiation term from Equation 1 has been omitted.

The gas stream temperature can be expressed in terms of its mean and dynamic components as:

$$T_g = \overline{T}_g + \sum_{n=1}^{\infty} a_n \sin(\omega_n t - \varphi_n) \tag{2}$$

Substituting Equation 2 written for a single frequency for T_g in Equation 1 and normalizing, the transfer function between the temperature in the thermocouple wire and the gas stream temperature can be written as:

$$\frac{\partial \zeta}{\partial t} = \frac{4h}{\rho_\omega C_{pw} D}\left[a_n \sin(\omega_n t - \varphi_n) - \zeta \right] + \alpha \frac{\partial^2 \zeta}{\partial x^2} \tag{3}$$

$$\text{Where: } \zeta = \frac{T_n - \overline{T}_g}{T_g - \overline{T}_g} = \frac{\theta_{1n}}{a_n} \tag{4}$$

The finite difference solution for Equation 3 is of the form:

$$\frac{\partial \zeta}{\partial t} \simeq \frac{\zeta_{j+1} - \zeta_j}{\Delta t} = \frac{\zeta_x' - \zeta_x}{\Delta t} \tag{5}$$

$$\frac{\partial^2 \zeta}{\partial x^2} \simeq \frac{\zeta_{x+1} + \zeta_{x-1} - 2\zeta_x}{\Delta^2} \tag{6}$$

Application of the finite difference solution to Equation 3 for the 9 node model of the thermocouple in Figure 2 yields Equations 7 through 16.

$$\zeta_o = \text{O (Assumed)} \tag{7}$$

$$\zeta_1' = \frac{C}{2A}[\zeta_o + \zeta_2 - 2\zeta_1] + \zeta_1 \tag{8}$$

$$\zeta_2' = \frac{C}{2A}[\zeta_1 + \zeta_3 - 2\zeta_2] + \zeta_2 \tag{9}$$

$$\zeta_3' = \frac{1}{2A}[C(\zeta_2 + \zeta_4 - 2\zeta_3) + F(\sin C_n t - \zeta_3)] + \zeta_3 \tag{10}$$

$$\zeta_4' = \frac{1}{2A}[C(\zeta_3 + \zeta_5 - 2\zeta_4) + 2F(\sin C_n t - \zeta_4)] + \zeta_4 \tag{11}$$

$$\zeta_5' = \frac{1}{2A}[C(\zeta_4 + \zeta_6 - 2\zeta_5) + 2F(\sin C_n t - \zeta_5)] + \zeta_5 \tag{12}$$

$$\zeta_6' = \frac{1}{(A + B)}[C(\zeta_5 + \zeta_6) + E(\zeta_7 - \zeta_6) + (F + G)(\sin C_n t - \zeta_6)] + \zeta_6 \tag{13}$$

$$\zeta_7' = \frac{1}{2B}[E(\zeta_6 + \zeta_8 - 2\zeta_7) + 2G(\sin C_n t - \zeta_7)] + \zeta_7 \tag{14}$$

$$\zeta_8' = \frac{1}{2B}[E(\zeta_7 + \zeta_9 - 2\zeta_8) + 2G(\sin C_n t - \zeta_8)] + \zeta_8 \tag{15}$$

$$\zeta_9' = \frac{1}{B}[E(\zeta_8 - \zeta_9) + G(\sin C_n t - \zeta_9)] + \zeta_9 \tag{16}$$

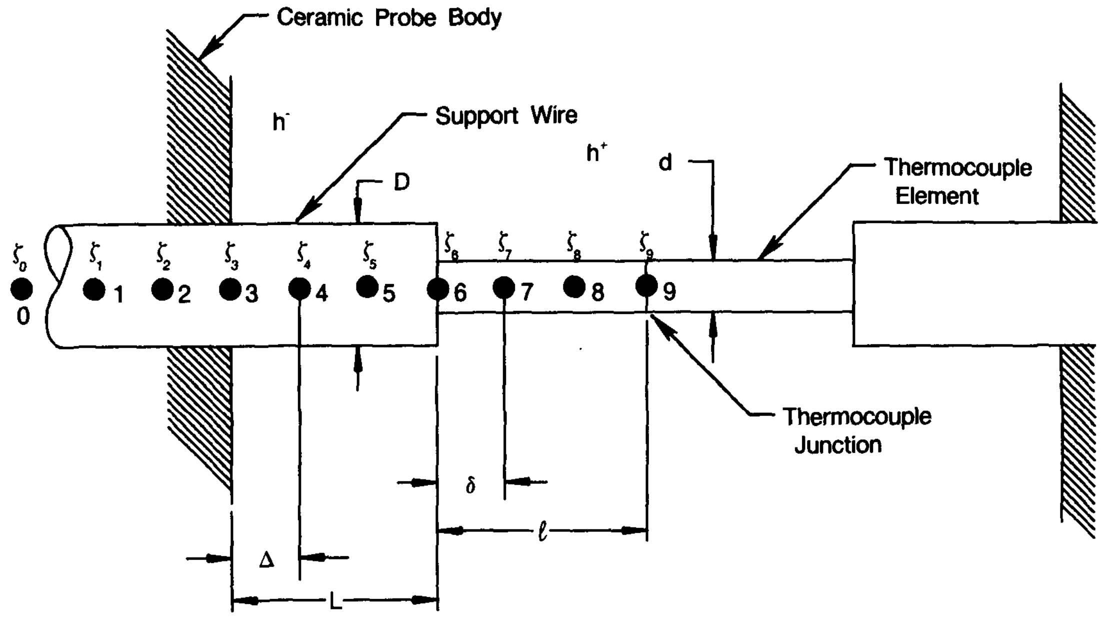


Figure 2. Finite Element Thermal Model — Used in Computer Compensation Program

- Definition of the dynamic temperature parameters is as follows:

θ_{1n} = Peak amplitude of smaller diameter thermocouple at frequency n

θ_1 (f) = θ_{1n} as a function of frequency

θ_{2n} = Peak amplitude of larger diameter thermocouple at frequency n

θ_2 (f) = θ_{2n} as a function of frequency

a_n = Peak amplitude of the dynamic component of the gas stream temperature at frequency n

a_n (f) = a_n as a function of frequency

φ_n = Phase shift of the gas temperature with respect to arbitrary time t_o at frequency f_n

η_{1n} = Phase shift of smaller diameter thermocouple with respect to gas temperature at frequency f_n

$\eta_1 (f)$ = η_{1n} as a function of frequency

η_{2n} = Phase shift of larger diameter thermocouple with respect to gas temperature at frequency f_n

$\eta_2 (f)$ = η_{2n} as a function of frequency

j = Time index

x = Spacial coordinate along length of thermocouple

T_n = Instantaneous temperature of thermocouple wire at frequency n

$T_{1 \ldots 9n}$ = Instantaneous temperature of thermocouple wire at spacial location at frequency n

$T_{x\ peak}$ = Maximum peak in instantaneous temperature of thermocouple wire at spacial location x

T_g = Instantaneous gas stream temperature

T_{gn} = Instantaneous gas stream temperature at frequency n

h^+ = Convective film coefficient of thermocouple element

h^- = Convective film coefficient of thermocouple support wire

σ = Boltzmann constant

α = Thermal diffusivity of the wire

$$\alpha = \frac{k_w}{\rho_w C_{pw}}$$

L = Length of larger diameter thermocouple support wire

ℓ = One half of length of smaller diameter thermocouple wire

D = Diameter of larger diameter support wire

d = Diameter of smaller diameter thermocouple wire

ρ_w = Density of thermocouple wire

k_w = Thermal conductivity of thermocouple wire

C_{pw} = Specific heat of thermocouple wire

ρ_g = Density of gas stream

k_g = Thermal conductivity of the gas stream

C_{pg} = Specific heat of the gas stream

Pr_g = Prandtl number of gas stream = $\frac{C_{pg}\,\mu_g}{k_g}$

U_g = Velocity of the gas stream

μ_g = Viscosity of the gas stream

γ_g = Ratio of specific heats of gas stream

P = Mean gas stream pressure

M_n = Mach number

$f_1 \rightarrow f_x$ = Frequencies of f_n at which transfer functions will be evaluated

F/A = Fuel air ratio

$A = \frac{D^2\,\Delta}{8\alpha(\Delta t)}$

$B = \frac{d^2\,\delta}{8\alpha(\Delta t)}$

$C = \frac{D^2}{4\Delta}$

$C_n = 2\pi f_n$

$\delta = \ell/3$

$E = \frac{d^2}{4\delta}$

$F = \frac{h^-\,D\Delta}{2K_w} = \frac{\Gamma D^{1/2}}{2\alpha}$

$G = \frac{h^+\,d\delta}{2K_w} = \frac{\Gamma d^{1/2}\delta}{2\alpha}$

Δ = L/8 = Space step

Δt = Time step

$$\Gamma = \frac{0.48\,k_g\,Pr_g^{1/3}U_g^{1/2}}{\left(\frac{\mu_g}{\rho_g}\right)^{1/2}\rho_w\,C_{pw}} = \text{Aerodynamic parameter} \quad (17)$$

H(f) = Measured transfer function (i.e., FFT frequency response function) of larger diameter thermocouple with respect to smaller diameter thermocouple

$G_{11}(f)$ = Measured FFT autospectral density function of smaller diameter thermocouple

$G_{12}(f)$ = Measured FFT cross-spectral density function between small thermocouple and large diameter thermocouple

$\gamma^2_{12}(f)$ = Measured FFT ordinary coherence function between larger diameter thermocouple and smaller diameter thermocouple

$S_1(f)$ = Measured FFT spectrum of smaller diameter thermocouple

Overview of Compensation Procedure

1. The theoretical transfer functions (frequency response function gain) between the 76 μm (3 mil) thermocouple and the gas stream (θ_{1n}/a_n) and the 254 μm (10 mil) thermocouple and the gas stream (θ_{2n}/a_n) are computed from the thermal finite difference solution using Equations 7 through 16, for a range of values of the aerodynamic parameter (Γ) at a number of discrete frequencies falling between the corner frequencies of the two thermocouples. These data are then used to compute the theoretical transfer function (θ_{2n}/θ_{1n}) between the 250 μm (10 mil) thermocouple and the 76 μm (3 mil) thermocouple for the corresponding values of Γ and frequency. These curves will be used to determine the in situ value of Γ from the measured transfer function of θ_{2n}/θ_{1n}. The process is described in the following paragraphs.

   a. The following parameters are input or already stored in the computer. For type B thermocouple wire — L, ℓ, D, d, ρ_w, k_w, C_{pw}, and α_w. For the gas stream — ρ_g, k_g, C_{pg}, γ_g, μ_g, and Pr_g.

   b. The average or mean conditions for the test data for the following variables are entered into the computer.

      T = Mean gas temperature

      P = Mean gas pressure

      F/A = Fuel air ratio

      $f_1 \rightarrow f_x$ = Frequencies of f_n at which transfer functions will be evaluated

      Mn = Mach number

   c. The program computes an estimated value of Γ based on the estimated run conditions using Equation 17.

   d. The program then computes ζ'_9, the transfer function between the wire thermocouple and the gas stream, for the 76 μm (0.003 in.) and the 254 μm (0.010 in.) thermocouple via Equations 7 through 16 from 0.2 Γ to 1.8 Γ in steps of 0.1 Γ at frequencies f_1, ..., f_x which are user selected to fall in between estimated values of the corner frequencies of the two thermocouples (Figure 3). The equations are evaluated until steady state conditions are reached. The criteria for steady state is that the positive maximum peak of θ_{xn}/a_n be within 0.1 percent of the absolute values of the negative maximum peak within the same period. The computer code determines the sampling interval for each frequency evaluated to ensure mathematical stability of the finite element model and minimize computation time. The normalized ratio of the magnitude of the temperature fluctuation in the wire to the temperature fluctuation of the gas stream ($\zeta'_9 = \theta_n/a_n$) at frequency f_n is determined by locating the maximum peak amplitude (Figure 4) after the model has iterated to steady-state conditions. The phase shift (η_x) of the temperature fluctuation in the wire is determined by locating the time at which ζ'_9 crossed zero going positive at the beginning of the period in which the model reached steady-state conditions (Figure 4).

e. The data from (d) are then used to compute the theoretical transfer function θ_{2n}/θ_{1n} from 0.2 Γ to 1.8 Γ (Figure 5) at frequencies of f_1 through f_x.

2. Thermocouple test data are digitized into the Fourier system computer, typically 32 to 120 records each of the 76 μm thermocouple dynamic signal and the 254 μm thermocouple dynamic and dc signals. Each record contains 2048 samples of the data. These data are then converted from millivolts to temperature using National Bureau of Standards (NBS) calibration curve coefficients for type B thermocouples. The 254 μm dc channel is used as the mean for both dynamic data channels in converting the nonlinear thermocouple mv signals to linearized temperature. These data records are then saved for recall for additional processing or plotting.

3. An ensemble averaged FFT transfer function (frequency response) analysis is then performed on *x* number of time records of the dynamic data to yield the measured value of θ_{2n}/θ_{1n} (i.e., H(f)) as a function of frequency. The transfer function is computed as the FFT cross spectral density function between the 76 μm thermocouple and the 250 μm thermocouple divided by the FFT autospectral density function of the 76 μm signal:

$$H(f) = \frac{G_{12}(f)}{G_{11}(f)} \tag{18}$$

In conjunction with the computation of the measured transfer function, the coherence function $\gamma^2_{12}(f)$ is computed and used to assess the quality of the measurement. For the 2048 time sample data record lengths used, 1024 line FFTs are produced. For the typical sampling rate of 4096 Hz (certain other sampling rates are permitted), the FFT analyses yield spectral information from dc to 2048 Hz in 2 Hz intervals. A standard Hewlett Packard windowing function (P301) is used prior to computation of the FFTs. This window is characterized by excellent spectral amplitude accuracy (less than ± 0.1 percent). Side lobe suppression is greater than −70 dB at ± 4 spectral lines and the effective noise bandwidth is 3.4 spectral lines.

4. Each measured value of H(f) at frequencies $f_n = f_1 \rightarrow f_x$ are used in conjunction with the theoretical values of θ_{2n}/θ_{1n} vs Γ at the corresponding frequencies to determine a measured value of Γ. (The program interpolates between the 0.1 Γ increments computed in (1) above.) The arithmetic average of Γ obtained for each frequency is taken as the in situ measured value.

5. Using the measured in situ average value of Γ obtained in (4), ζ'_9, the normalized transfer function (gain θ_{1n}/a_n and phase η_{1n}) of the 76 μm thermocouple with respect to the gas stream temperature is then computed at all frequencies from the first spectral line of the FFT spectrum to the Nyquist frequency of the FFT for each discrete frequency contained in the FFT (Figure 6). This is typically from 2 Hz to 2048 Hz in 2 Hz increments. This is the compensation spectrum $\theta_1(f)/a_n(f)/\eta_1(f)$ which is then used to compensate the 76 μm thermocouple data as follows:

6. To compute the compensated ensemble averaged PSD, the measured ensemble averaged autospectral density function of the 76 μm (3 mil) thermocouple obtained in (3) above is divided by the autospectral density function of its compensation spectrum:

$$\frac{G_{11}(f)}{\left(\frac{\theta_1(f)}{a_n(f)}\right) * \left(\frac{\theta_1(f)}{a_n(f)}\right)} \approx \frac{\theta_1(f) * \theta_1(f)}{\left(\frac{\theta_1(f)}{a_n(f)}\right) * \left(\frac{\theta_1(f)}{a_n(f)}\right)} \rightarrow a_n^2(f) \tag{19}$$

where: * = Complex conjugate multiplication.

Scaling factors for effective noise bandwidth and FFT symmetry are applied.

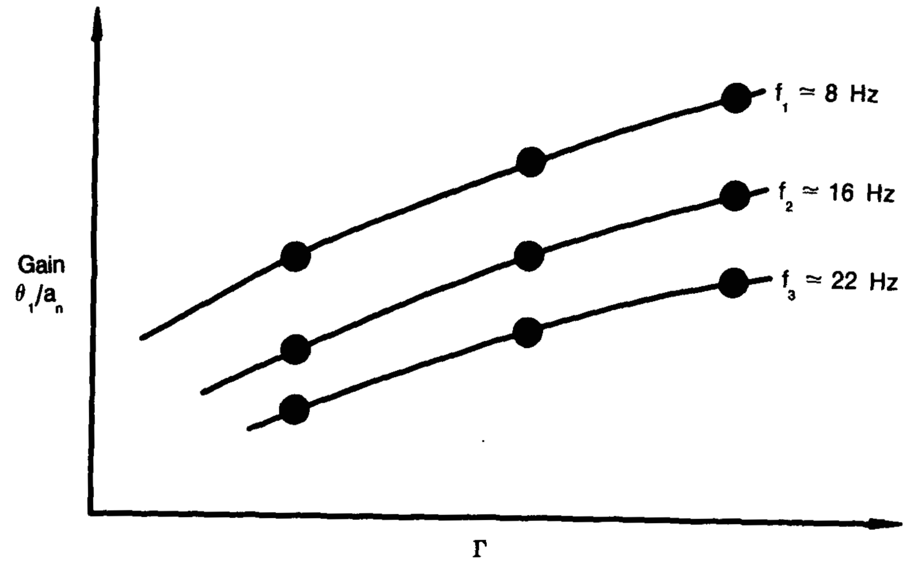
7. To compute the compensated instantaneous time waveform, an FFT spectrum ($S_1(f)$) is made on a specific user selected time record yielding amplitude and phase terms for each spectral component. This spectrum expressed in rectangular frequency coordinates is then divided (complex math) by the compensation spectrum. The compensated instantaneous spectrum is then inverse Fourier transformed to yield the compensated instantaneous time waveform. The software contains information on specific techniques employed to prevent time waveform distortions associated with the inverse Fourier transform. A threshold, in relative dB, is applied to the frequency spectrum of the data signal prior to division by the compensation spectrum to prevent errors where the signal to noise ratio is too low:

$$\frac{S_1(f)\angle\lambda_1(f)}{\left(\frac{\theta_1(f)}{a_n(f)}\right)\angle\eta_1(f)} \sim \frac{\theta_1(f)\angle\eta_1(f) + \varphi(f)}{\left(\frac{\theta_1(f)}{a_n(f)}\right)\angle\eta_1(f)} \rightarrow a_n(f)\angle\varphi(f) \tag{20}$$

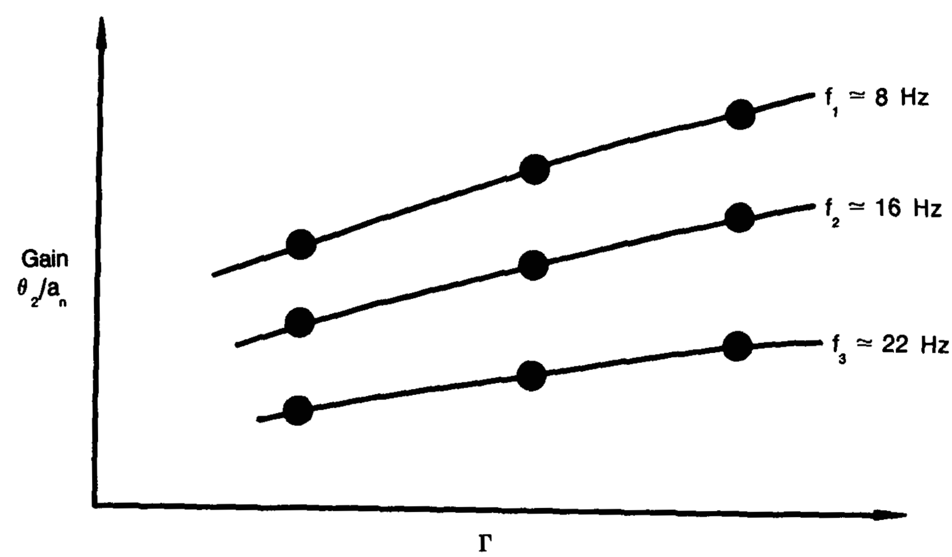
which is the compensated instantaneous frequency spectrum and

$$F^{-1}\, a_n(f)\angle\varphi(f) \tag{21}$$

which is the compensated time waveform where F^{-1} is the inverse Fourier Transform.



(a) Gain of 76 μm Thermocouple vs Aerodynamic Parameter



(b) Gain of 254 μm Thermocouple vs Aerodynamic Parameter

FDA 267182

Figure 3. Theoretical Curves of ζ_q for 76 μm and 250 μm Thermocouples

7195C

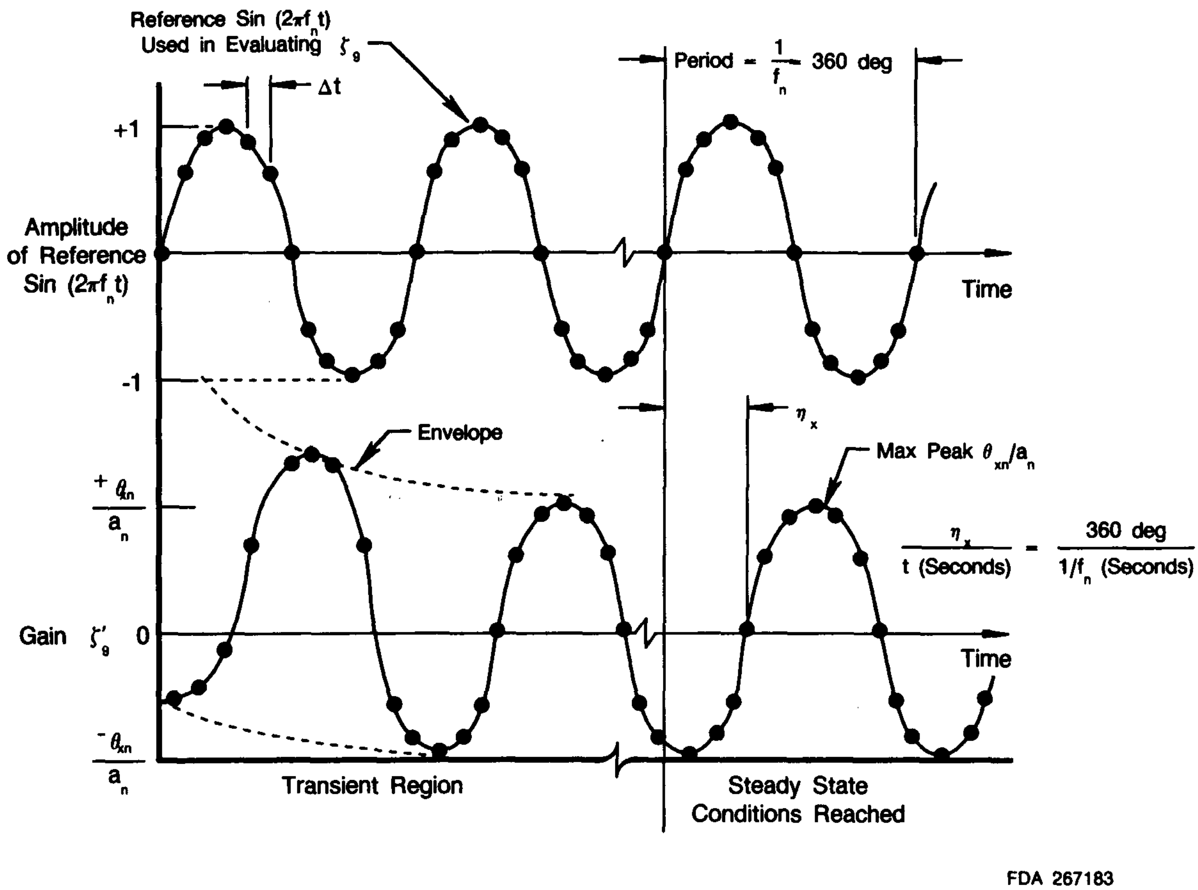
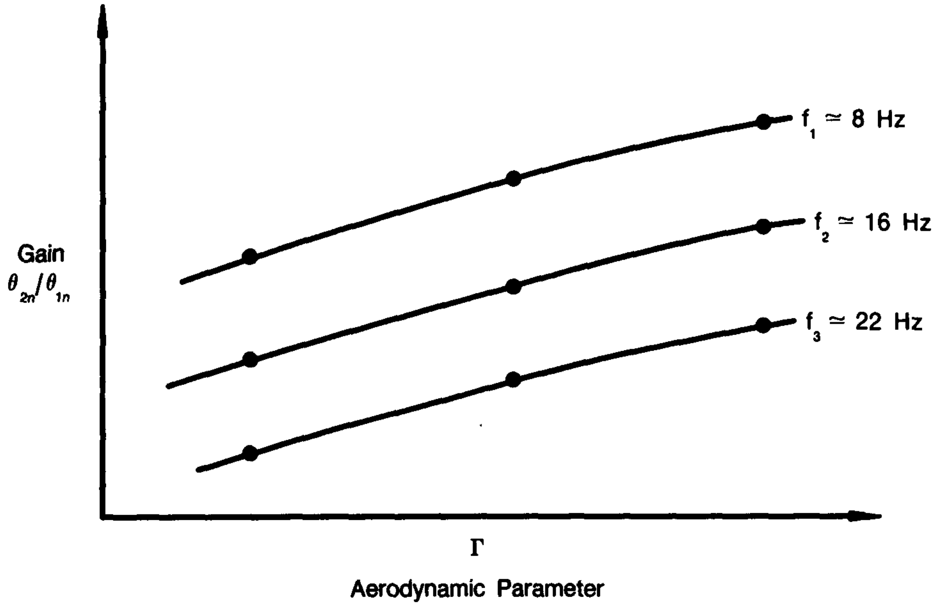
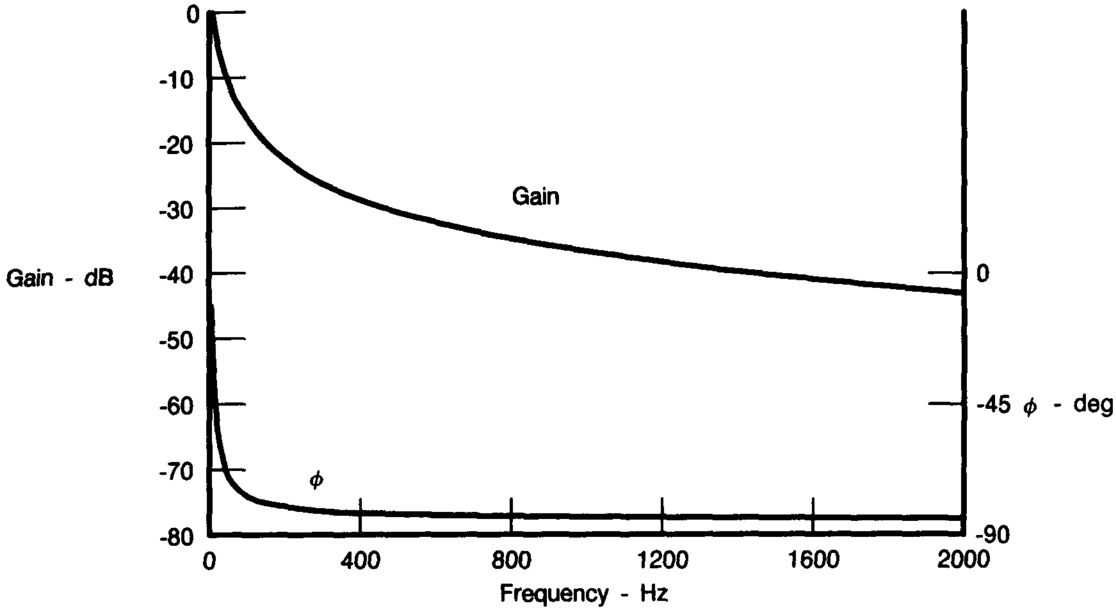


Figure 4. Transfer Function of Thermocouple With Respect to Gas Temperature



FDA 267184

Figure 5. Theoretical Curves of ζ_q for the 250 μm Thermocouple Divided by ζ_q of the 76 μm Thermocouple



FDA 267185

Figure 6. Typical Compensation Spectrum for 76 μm (3 mil) Thermocouple Output

7195C

DATA PREPROCESSING

Data Acquisition and Playback

Test data are normally recorded on an FM magnetic tape recorder and then reproduced and digitized post-test off-line. Typically, an IBM compatible digital tape is prepared, and data are transferred to a file of the proper format (for input to the FORTRAN program) through the use of user supplied routines. For a system providing on-line digitization and input to a mainframe IBM computer, data would not have to be recorded/reproduced from FM magnetic tape.

The Dynamic Gas Temperature Measurement System does not incorporate a calibration signal analysis routine. The user must provide the means of measuring the rms and 'dc offset' values required for determining the necessary inputs of gain, output/input, and dc offset into the main program. Referring to Figure 7, these necessary user inputs are described as follows:

- GAIN — The gain of the thermocouple signal amplifier at each specific test point. Used to optimize signal level for recording.

- OUTPUT/INPUT — The overall gain of the record/playback system which is used in conjuction with GAIN to scale data to voltage for conversion to scaled temperature. Compute output/input for entry using the ac calibration event as follows:

$$\frac{\text{OUTPUT}}{\text{INPUT}} = \frac{\text{rms volts (or counts) at conversion to data input file}}{\text{rms volts of input calibration signal}} \tag{22}$$

- DC OFFSET — Zero offset of record/playback system. Compute as average of measured volts (or counts) at conversion to data input file from the shorted input calibration event.

NOTE: GAIN assumes offset in thermocouple signal amplifier is zero volts at all gain settings. If this is not the case, the user must algebraically add the setting to OFFSET determined from the calibration event.

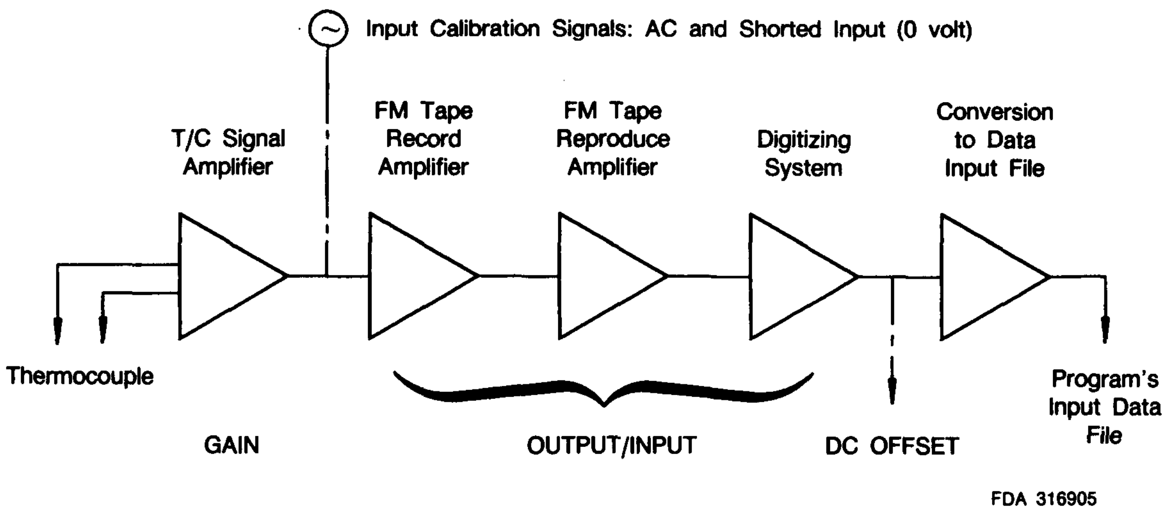


Figure 7. Elements of Data Acquisition/Playback

Conversion from Digitized Data to Test Data Input File

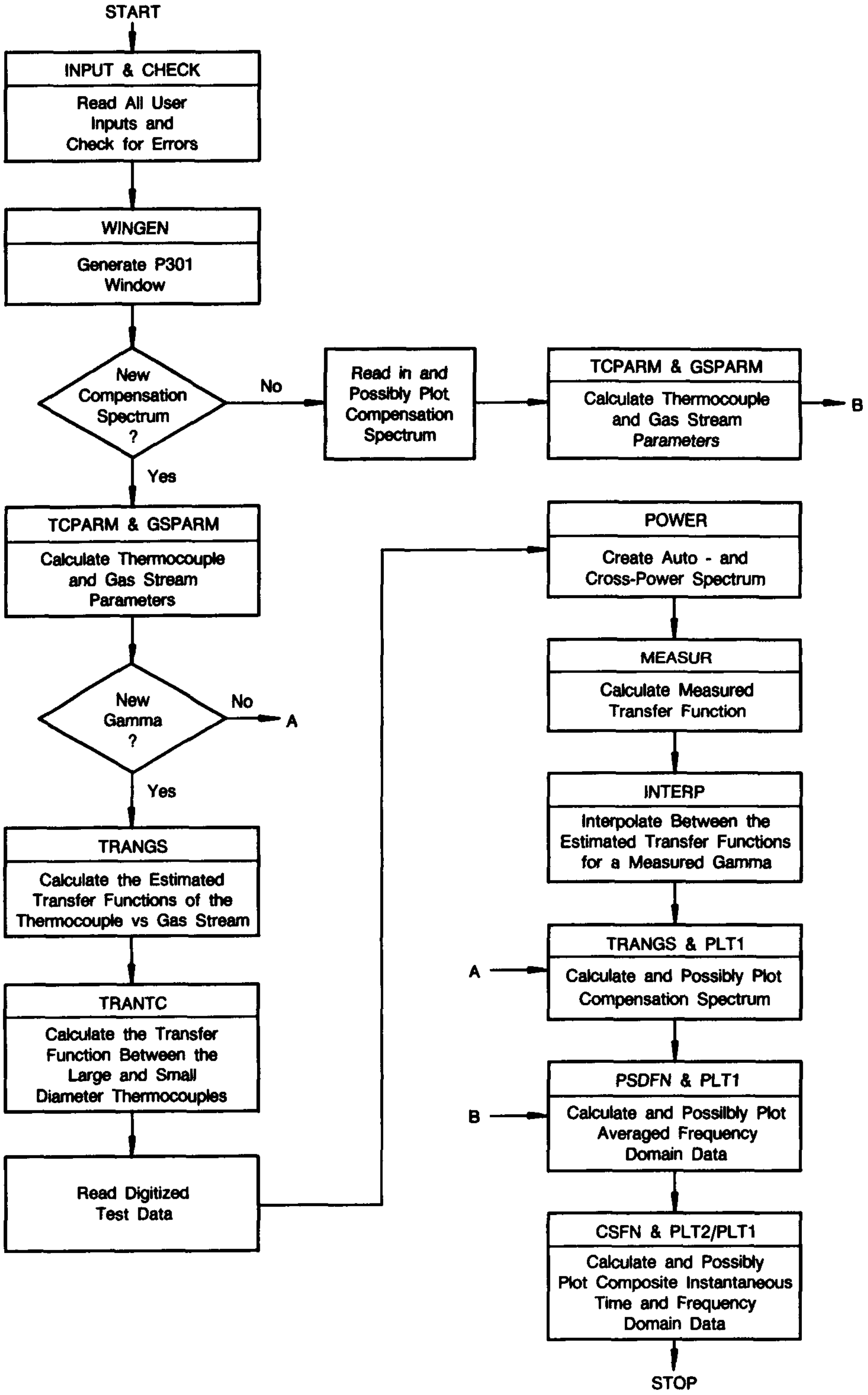
Once the data have been digitized, they must be put into a form accessible by the program. The program accesses the data input file as unit 4. This file must be in the exact format described below.

The data are arranged in blocks with a data block consisting of 2048 data points (or the blocksize designated in the user input file) of a specific type of data. The data file is set up as one block of small thermocouple ac (dynamic), one block of large thermocouple ac (dynamic), and one block of dc (steady-state) data. This set-up is repeated for each record of data desired. Each individual line of a data block is in the format (1X,6E13.0). For example, a data block with blocksize 2048 consists of 341 lines with six numbers (starting in column 2) in exponential format, and one line with only two numbers, completing the 2048 data points.

DESCRIPTION OF THE COMPUTER PROGRAM

General Description

The overall system logic is shown in the flow diagram of Figure 8. The names of the subroutines called to perform the various functions are enclosed in the flow diagram boxes. Detailed flow diagrams of the major subroutines are shown in the Appendix. The program operates in three major sections, (a) calculation of an estimated value of the aerodynamic parameter, gamma (Γ_e), and the estimated transfer functions, (b) evaluation of a measured value of gamma (Γ_m), and calculation of the compensation spectrum, and (c) calculation and plotting of averaged and/or instantaneous time and/or frequency domain data. On the TSS operating system, the program must be run twice, once for the calculations of the compensation spectrum, scaled data, and Fourier transforms, and again for the creation of plots. See "Program Execution on TSS Operating System Using the DISSPLA Graphics Package" section for details.



FDA 323713

Figure 8. Program Logic Design

7189C

Functions Performed

Referring to Figure 8, the sequence of functional operation is in the following order:

1. Input thermocouple dimensions, gas stream properties, digitized data recording information, and all user options.

2. Generate the P301 windowing function.

3. Calculate thermocouple and gas stream parameters. Note: Thermocouple parameters include the estimated value of gamma (Γ_e).

4. Check user options to see if calculation of a new compensation spectrum is desired. If not, go to 12.

5. Check user options to see if calculation of a new gamma is desired. If not, go to 11.

6. Calculate the estimated thermocouple vs gas stream transfer functions for $0.2\Gamma_e$ to $1.8\Gamma_e$ at steps of $0.1\Gamma_e$ for both small and large diameter thermocouples.

7. Find the estimated transfer functions of the large diameter thermocouple to the small diameter thermocouple for $0.2\Gamma_e$ to $1.8\Gamma_e$.

8. Form the auto-power and cross-power spectra as follows:

   A) Read in 1 block of digitized test data desired in the ensemble averaging.

   B) Scale the data to temperature (Kelvin) and multiply by the P301 windowing function.

   C) Perform an FFT to convert the data to frequency domain.

   D) Self-conjugate and cross-conjugate multiply to form the auto-power and cross-power spectra respectively.

   E) Have all records that were desired been used? If not, return to A and repeat.

   F) Ensemble average the auto-power and cross-power spectra for the desired records.

9. Calculate the measured transfer function and the coherence function as functions of the auto-power and cross-power spectra.

10. Using the measured transfer function, interpolate with estimated transfer functions in order to find a measured gamma (Γ_{m}).

11. Calculate compensation spectrum (the transfer function of the desired thermocouple vs gas stream using the measured gamma).

12. Find the averaged frequency domain data as follows:

    A) Access the frequency domain data calculated in 8C and stored on units 15 (small TC) and 16 (large TC).

    B) Self-conjugate multiply to form the auto-power spectrum.

    C) If data are to be uncompensated, skip to step F.

    D) Perform self-conjugate multiplication on the compensation spectrum to put in power form.

    E) Divide the auto-power spectrum by the power form of the compensation spectrum.

    F) Scale the data according to Table 1.

    G) Ensemble average desired records.

13. Calculate composite instantaneous time and frequency domain data as follows:

    A) Access the scaled time domain temperature data found in step 8B, stored on units 13 and 14.

    B) If data are to be uncompensated skip to step J.

    C) Obtain a second data block and shift as described below in Description of Subroutines (CSFN).

    D) Multiply data by the Hanning window.

    E) Perform an FFT to the frequency domain.

    F) Perform complex division by the compensation spectrum.

    G) Subject data to an inverse Fourier transform back to the time domain.

    H) De-Hann data by dividing by the Hanning Window.

    I) Combine two data blocks into one as described in Description of Subroutines (CSFN) below.

    J) Apply the P301 window to time domain data.

    K) Perform an FFT to convert to frequency domain.

    L) Self-conjugate multiply and scale according to Table 1.

Detailed flowcharts of the major subroutines are presented in the Appendix.

Description of Subroutines

The Dynamic Gas Temperature Measurement System program was written in such a way that all major functions are performed by distinct and separate subroutines. The function of the MAIN routine and all subroutines is described in this section.

MAIN

The primary function of the MAIN routine is to pass control of the program to the major subroutines. However, if the program is using a previously calculated compensation spectrum, the MAIN routine reads in this data from unit 12. The compensation spectrum will be written to unit 6 (if IBUG2 is turned on), and then converted to rectangular coordinates (it is stored as polar gain and phase) before execution continues.

CHECK

Because of the various user options available, required inputs change according to the IFLAGS entries. Subroutine CHECK checks that the user has input all needed information. If an error has occured, control is passed to subroutine TERM which terminates execution with an appropriate message. See Input Description in the Input/Output Section for a description of variables.

CSFN

CSFN evaluates instantaneous compensated time and frequency spectra for plotting. This is done by reading in two records of scaled, digitized data from unit 13 or 14 (depending on the thermocouple used). Compensated time domain data are derived by applying the Hanning windowing function,

$$H(x) = 1-\cos\left(2\Pi(x-1)/IBLSZ\right) \text{ for } x = 1,IBLSZ \tag{23}$$

followed by performing an FFT. The user supplied threshold level (see step 7 on page 11) is then applied before dividing by the compensation spectrum. Data are then subjected to an inverse Fourier transform followed by de-Hanning. Application of the Hanning Window causes some invalidities at the ends of the data blocks, therefore the two data blocks are shifted to provide one block of 'good' data. The shifting is performed as shown in Figure 9.

Frequency domain data are found by applying the P301 window to the time domain data, performing an FFT, scaling the data according to Table 1, and self-conjugate multiplying. If IBUG2 is turned on, these functions are written to the output file, unit 6.

Variables

- ALSS — Area line shape squared, needed for scaling of frequency data (input)
- COMP — Compensation spectrum (input)
- DATA — Time domain data (output to plotting routine)
- DATAF — Frequency domain data (output to plotting routine)
- GS — Array containing gas stream parameters, needed only for the plotting routine (input)
- WINDO — P301 windowing function (input).

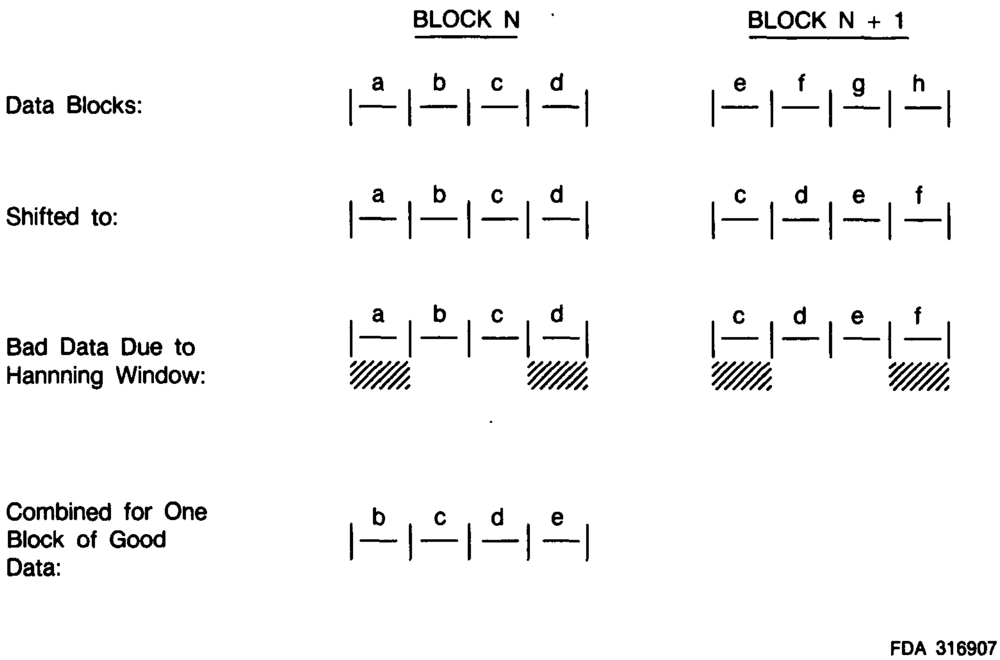


Figure 9. Data Block Shifting

FFT

Subroutine FFT calculates finite complex Fourier transform or the inverse transform of a complex input array. Input to the routine is a complex array of size N. (Note: N must be an integer power of 2.) Complex data consist of a real part equal to the time domain data and an imaginary part of zero. The FFT is returned from the routine in the same complex array. The first element of the direct transform is the mean (zero frequency) value. The second through (N/2 + 1) values are for positive frequencies, and the remaining (N/2 - 2) values are for negative frequencies. Results from these negative frequencies are folded about the (N/2 + 1) point. Thus, the frequency for the (N/2 + 2) point is the negative of the frequency for the N/2 point, and so on.

Variables

A	— Array of N complex values to which the transform is applied and put back into the array (input and output)
ISET	— Type of transform requested (input) 1 = Direct transform −1 = Inverse transform
N	— Number of data points or block size — *must be an integer power of* 2 (input).

GET

Subroutine GET accesses the Fourier transformed input data stored on unit 15 or unit 16, depending on the thermocouple for which the compensation spectrum was found, and self-conjugate multiplies to obtain the auto-power spectrum in (Peak deg)^2.

Variables

WW — Array to contain the auto-power spectrum (output).

GSPARM

This subroutine calculates the gas stream parameters which follow:

- Density (RHO)
- Thermal conductivity (XK)
- Specific heat (CP)
- Specific heat ratio (GA)
- Viscosity (XMU)
- Sonic velocity (SONVL)
- Kinetic viscosity (G)
- Prandtl number (PR)
- Mean gas velocity (U)
- Aerodynamic parameter, Γ(GMA).

When the aerodynamic parameter is found, it is written to unit 6 if IDEBUG is turned on. The equations used for these calculations are listed in Section II.B of Volume II, Final Report (NAS3-23154, FR-17145)

Variables

GS — Array of gas stream parameters (output)
TS — Array of thermocouple parameters (input).

INPUT

Subroutine INPUT reads all of the user inputs from unit 5. For a description of the variables see Input Description in Input/Output Section.

INTERP

Subroutine INTERP uses the measured transfer function (see MEASUR) to interpolate within a family of estimated transfer functions of the two thermocouples (see TRANTC) at $0.2\Gamma_e$ to $1.8\Gamma_e$ to find a measured gamma, Γ_m. The interpolation is performed at all user desired frequencies and the results are averaged to find the measured value of gamma. If the measured coherence function is not within the acceptable range of 0.8 to 1.005 at any frequency, that frequency point is not used. Failure to obtain at least one frequency with an acceptable coherence will result in an appropriate message being written to unit 6 and termination of program execution.

Variables

COHR — Coherence function (input)
GS — Array containing the gamma value that is updated in this routine (input and output)
RTRAN — Measured transfer function (input)
TRAN — Estimated transfer functions (input).

7189C

INTEST

This subroutine strictly reads in the digitized test data from unit 4.

Variables

IFLAG	— Flag to signal end of data (output)
DATA3	— One record of small diameter thermocouple data (output)
DATA10	— One record of large diameter thermocouple data (output)
DATADC	— One record of dc channel data (output).

MEASUR

Subroutine MEASUR calculates the measured transfer function of the two thermocouples, and the coherence function which is used to ensure adequate signal to noise ratio. The transfer function is computed as the FFT cross-spectral density function between the small and large diameter thermocouple divided by the FFT auto-spectral density function of the small diameter thermocouple. The coherence function is the self-conjugate multiplication of XY divided by the product of XX and YY.

Variables

COHR	— Coherence function (output)
RTRAN	— Measured transfer function array (output)
XX	— Small diameter thermocouple auto-power spectrum (input)
XY	— Cross-power spectrum of small to large diameter thermocouple (input)
YY	— Large diameter thermocouple auto-power spectrum (input).

OUT

Subroutine OUT prints the user inputs with a description of their meaning to unit 6. See Input Description in Input/Output Section for variable definitions.

PLT1

Subroutine PLT1 plots one input array on a page. This plot can either be the averaged frequency domain data (frequency vs temperature) or the composite instantaneous time waveform (time vs temperature). Which function is being plotted is determined by the value of ICODE, sent through the argument list. The information necessary for labeling the plot is transferred through the argument list and the two commons, INPUTS and PLOTTR. Examples of the type of plots generated can be found in the Test Cases Section, Test Case 2 Plot 1 and Test Case 3 Plot 2.

Variables

ARRAY	— The array to be plotted (input)
ICODE	— Code for function being plotted (input) 1 = Power spectral density function 2 = Composit instantaneous time waveform
NUM	— Number of record being plotted if ICODE=2 (input)
TMEAN	— The mean dc temperature for labeling plots (input).

PLT2

PLT2 generates two plots on the same page. These plots are either the compensation spectrum (frequency vs gain and phase), or the composite instantaneous spectra (time and frequency vs temperature). Which function is being plotted is determined by the value of ICODE, sent through the argument list. The information necessary for labeling the plots is transferred through the argument list and the two commons, INPUTS and PLOTTR. Examples of the plots generated can be found in the Test Cases Section, Test Case 1 Plot 1 and Test Case 2 Plot 2.

Variables

ARR1	—The first array to be plotted (input)
ARR2	—The second array to be plotted (input)
GS	—Array containing the value of gamma for which the compensation spectrum was found. Needed for plot labeling (input).
ICODE	—Code for function being plotted (input) 1 = Compensation spectrum 2 = Composit instantaneous spectra
NUM	—Number of the record being plotted if ICODE=2 (input)
TMEAN	—The mean dc temperature for labeling plots (input).

POWER

Subroutine POWER calculates auto-power spectrums of large and small diameter thermocouples, and the cross-power spectrum of small vs large diameter thermocouples. This is accomplished by taking digitized data for each record specified in IAVDAT (input variable), scaling to Kelvin, applying the P301 window, and performing an FFT.

The auto-power and cross-power spectrums are the self-conjugate multiplication and the conjugate multiplication of the data respectively. These are averaged together for specified records to arrive at the desired functions. The FFTs are written to units 15 and 16 (depending on the thermocouple) for future executions of the program with the same data.

Variables

DATA3	—Small diameter digitized data (output)
DATA10	—Large diameter digitized data (output)
DATADC	—DC channel digitized data (output)
XX	—Array containing the small wire auto-power spectrum (output)
XY	—Array containing the cross-power spectrum of small to large wire thermocouples (output)
YY	—Array containing the large wire auto-power spectrum (output).

PRNTIN

PRNTIN prints out (to unit 6) the user's input data in card image format.

Variables

IIN	—Unit number of the input file to be printed (input)
IOUT	—Unit number of the output file into which to write the data (input).

PSDFN

Subroutine PSDFN calculates the Power Spectral Density function (averaged frequency domain data) for plotting. This is accomplished by accessing FFTs for the desired records from units 15 and 16, and self-conjugate multiplying to form the auto-power spectrum. If the data are to be compensated, they are then divided by the power form of the compensation spectrum. To complete the computations, data are scaled according to Table 1 and plotted and/or printed depending on the user's options. Also performed in PSDFN is the calculation of ALSS, which is the sum of the squares of the P301 window at each channel, divided by the block size. ALSS is used in scaling frequency domain data.

Variables

ALSS	—Area line shape squared (output)
COMP	—Compensation spectrum (input)
WINDO	—P301 windowing function (input).

SCALER

Subroutine SCALER takes digitized data and converts it to temperature (Kelvin). SCALER first removes the amplifier dc offset and scales the data to volts prior to conversion to temperature. Two records are ac data (large and small wire thermocouples) and the third is the dc channel. After removal of dc offset and scaling, the program adds the dc to the ac, converts it to temperature, and then removes the dc, leaving peak temperature. This is done to both ac channels. Scaled data are written onto units 13 and 14 for future access.

Variables

DATA3	—Data for the small wire thermocouple, contains digitized data as input and scaled data as output
DATA10	—Data for the large wire thermocouple, contains digitized data as input and scaled data as output
DATADC	—Data for the dc channel, contains digitized data as input and scaled data as output.

SPCY

Subroutine SPCY determines the sampling frequency as a function of the input frequency.

Variables

A	—Array which contains the sampling frequency (output)
FRQ	—Frequency for which the sampling frequency is being found (input).

TCALC

Subroutine TCALC calculates temperature from digitized data for the routine SCALER. The equations used depend on the thermocouple material code, and involve coefficients stored in the array TCF.

The coefficients are from thermocouple curve equations which were derived from NBS curves, where the independent variable (millivolts) was normalized between −1 and +1. The reference junction is 32°F.

7189C

Variables

T — Contains scaled data as input and the derived temperature as output.

TCPARM

Subroutine TCPARM calculates the following thermocouple wire parameters:

- Density (RHO)
- Thermal conductivity (XK)
- Specific heat (CP)
- Thermal diffusity (AL).

Equations for type B thermocouples can be found in Section II.B (4.1), Final Report Volume II (NAS3-23154 FR-17145). Equations used for type K thermocouples are as follows:

RHO $= 540.95$ lb/ft^3

CP $= (0.0001129 \times T + 0.21454)/2.0$ btu/lb°F

XK $= (0.01547 \times T + 24.505)/7200.0$ btu/ft-sec°F

AL $= XK/(RHO \times CP)$ ft^2/sec

Variables

TC — Array containing the thermocouple parameters (output).

TERM

Subroutine TERM terminates the program due to an illegal user entry, or because a calculated gamma was not able to be found. Before terminating execution, TERM writes an appropriate message to output unit 6.

Variables

I — Code to determine which error has occured (input).

TRFM

Subroutine. TRFM performs the actual evaluation of the transfer function of the thermouple wire and the gas stream using the finite difference method described in detail in Section III.C of Volume I, Final Report (Reference 1).

Variables

A — Array in which gain and phase are stored (output)
FRQ — Frequency (input)
TP — Parameters found by TRFP, needed to evaluate transfer function (input).

TRFP

Subroutine TRFP calculates the following parameters needed for evaluation of the transfer function.

- Delta
- Deltat
- Sigma
- CN
- A1
- B
- C
- E
- F
- G

The equations for these parameters are located in Section II.B (4.3), Volume II, Final Report (NAS3-23154 FR-17145), and their description is in Section V.F, Volume I, Final Report (NAS3-23154 FR-17145).

Variables

A	— Array containing the sampling frequency (output)
FRQ	— Frequency (input)
J	— Code for which thermocouple is being evaluated (input)
TC	— Array containing thermocouple parameters (input)
TP	— Array containing above parameters (output).

TRANGS

Subroutine TRANGS evaluates the transfer function between the thermocouple and the gas stream. This routine is used twice, once for the estimated transfer functions, and once for the compensation spectrum. When finding the estimated transfer functions, the test gammas of $0.2\Gamma_e$ to $1.8\Gamma_e$, and the user requested channels (FREQ of input) are used for evaluation. For the compensation spectrum, the measured value of gamma is used, and a piecewise transform is performed (i.e., the transfer function is found at the first 50 channels, followed by every tenth, and filled in linearly). The compensation spectrum is plotted if requested, and then written to unit 12 for future use. If IBUG2 is turned on, it is also written to output file 6. The compensation spectrum is computed, stored, and plotted as polar gain and phase, but converted to rectangular coordinates before execution continues.

Variables

COMP	— Compensation spectrum (output)
GS	— Array which contains the aerodynamic parameter, gamma (input)
J	— Code for thermocouple for which transfer function is desired (input) 1 = Large diameter 2 = Small diameter
NGAM	— Number of gamma values for which transfer function is desired (input) NGAM = 17 $\rightarrow$ estimated transfer functions NGAM = 1 $\rightarrow$ compensation spectrum
TC	— Array containing thermocouple parameters (input)
TRAN	— Array into which the transfer function is placed (output).

TRANTC

TRANTC evaluates the estimated transfer functions between large and small wire thermocouples, for all test gammas and all desired frequencies.

Gain	=	Gain large/Gain small
Phase	=	Phase large-Phase small

Variables

NGAM	— Number of gamma values for which to evaluate the transfer function, always equal to 17 (input).
TRAN	— Array containing transfer functions of large and small thermocouples vs gas stream, and the transfer function found in this routine (input and output).

WINDOW

Subroutine WINDOW applies the P301 windowing function to desired data.

Variables

WINDO	— The P301 window (input)
DATA	— The data to which the window is applied (input and output).

WINGEN

WINGEN generates the P301 windowing function and places it in the variable WINDO.

$$\begin{aligned} P\ (x) = 0.9994484\ &+ 2(-0.955728 \cos(2\ \pi\ (x\text{-}1)/IBLSZ) \\ &+ 0.539289 \cos(4\ \pi\ (x\text{-}1)/IBLSZ) \\ &- 0.091581 \cos(6\ \pi\ (x\text{-}1)/IBLSZ) \\ &\text{for } x = 1,\ IBLSZ \end{aligned}$$

Subroutine Interaction

The interaction between program subroutines is shown in Table 2.

Common Blocks

The Dynamic Gas Temperature Measurement System Program contains only three common blocks. Common /DATAS/ contains two arrays, both of which are coefficients needed for the finite difference method, initialized in BLOCKDATA and never altered. Common /INPUTS/ contains all user input data read in subroutine INPUT and needed throughout the program. The last common, /PLOTTR/, consists of only one array containing any data that the user wishes to have printed on all plots generated. Table 3 describes common block variables (type, dimension, units, and definition); Table 4 is a common block/subroutine cross reference; and Table 5 shows treatment of common block variables within the subroutines in which they are contained.

TABLE 2. — SUBROUTINE INTERACTION

Subroutine	*Calls*	*Referenced by*
CHECK	TERM	MAIN
CSFN	FFT PLT1 PLT2	MAIN
FFT	—	CSFN
GET	—	PSDFN
GSPARM	—	MAIN
INPUT	OUT PRNTIN	MAIN
INTERP	TERM	MAIN
INTEST	—	POWER
MAIN	CHECK CSFN GSPARM INPUT INTERP MEASUR PLT2 POWER PSDFN TCPARM TRANGS TRANTC WINGEN	—
MEASUR	—	MAIN
OUT	—	INPUT
PLT1	—	CSFN PSDFN
PLT2	—	CSFN MAIN TRANGS
POWER	FFT INTEST SCALER WINDOW	MAIN
PRNTIN	—	INPUT
PSDFN	GET PLT1	MAIN
SCALER	TCALC	POWER
SPCY	—	TRFM TRFP
TCALC	—	SCALER
TCPARM	—	MAIN
TERM	—	CHECK INTERP
TRANGS	PLT2 TRFM TRFP	MAIN
TRANTC	—	MAIN
TRFM	SPCY	TRANGS
TRFP	SPCY	TRANGS
WINDOW	—	
WINGEN	—	MAIN

7222C

7189C

TABLE 3. — COMMON BLOCK VARIABLES

Common Block	Variables	Type	Dimension	Units	Description
DATAS	C	R	45	—	Array containing the coefficients of the equations for the thermocouple wire parameters.
	TCF	R	(11,9)	—	Array containing the coefficients of the equations used to calculate temperature from digitized data.
INPUTS	IFLAGS	I	12	—	See Input Description section for definition of IFLAGS.
	TCDATA	R	(4,2)	cm	Thermocouple dimensions
	GAS	R	4	—	Fuel/air ratio
				K	Mean gas temperature
				N/m^2	Mean gas pressure
				—	Mach number
	FREQ	R	4	Sec	Delta-T setting
				Hz	Starting frequency, ending frequency, and frequency increment for which to find the transfer
	CHANL	R	9	Volts	Contains gain, record level, and offset for small, large, and dc thermocouples
	IAVDAT	I	2	—	Starting record number and number of records to use in ensemble averaging
	IBLSZ	I	—	—	Data block size
	IREC	I	10	—	Records desired for plotting of instantaneous spectrum
	TIME	R	2	Sec	Starting and ending time (with respect to data block) for partial time range plots
	IBSZ	I	—	—	Half the block size plus one
	GAMMA	R	—	—	User input value of gamma
	NREC	I	10	—	Records for plotting of frequency domain data when only one record at a time is averaged
	NRECS	I	2	—	Starting record and number records to use in averaging of frequency domain data
	PLTFRQ	R	—	Hz	Frequency at which to end plots of frequency domain data
	TIMTEM	R	—	K	Temperature to which the instantaneous time domain data is to be scaled
	IDEBUG	I	—	—	Intermediate write option 1 0 — no writes 1 — write out the interpolated gammas
	IBUG2	I	—	—	Intermediate write option 2 0 — no writes 1 — print out all of the functions generated
	ITHRSH	I	—	dB	Relative threshold level for instantaneous time domain plots
PLOTTR	PLOTIT	A	(20,3)	—	User supplied headings for plots

7222C

7189C

TABLE 4. — COMMON BLOCK/SUBROUTINE CROSS REFERENCE

Common Block	*Subroutines Containing the Common Block*
INPUTS	CHECK
	CSFN
	GET
	GSPARM
	INPUT
	INTERP
	INTEST
	MAIN
	MEASUR
	OUT
	PLT1
	PLT2
	POWER
	PSDFN
	SCALER
	TCALC
	TCPARM
	TRANGS
	TRANTC
	TRFP
	WINDOW
	WINGEN
DATAS	BLOKDATA
	GSPARM
	MAIN
	POWER
	SCALER
	TCALC
	TCPARM
PLOTTR	INPUT
	PLT1
	PLT2

7222C

7189C

TABLE 5. — TREATMENT OF COMMON BLOCK VARIABLES WITHIN SUBROUTINES

Subroutines and Usage Codes

Common Block	Variable	CHECK	CSFN	GET	GSPARM	INPUT	INTERP	INTEST	MAIN	MEASUR
INPUTS	IFLAGS	R	R	R	R	A	U	U	R A	U
	TCDATA	U	U	U	U	A	U	U	A	U
	GAS	U	U	U	R	A	U	U	U	U
	FREQ	U	R	U	U	A	R	U	U	U
	CHANL	U	U	U	U	A	U	U	U	U
	IAVDAT	U	U	U	U	A	U	U	U	U
	IBLSZ	U	R	U	U	A	U	R	U	U
	IREC	R	R	U	U	A	U	U	U	U
	TIME	R	U	U	U	A	U	U	U	U
	IBSZ	U	R	R	U	A	U	U	R	R
	GAMMA	R	U	U	U	A	U	U	R	U
	NREC	R	U	U	U	A	U	U	U	U
	NRECS	R	U	U	U	A	U	U	U	U
	PLTFRQ	U	U	U	U	A	U	U	U	U
	TIMTEM	R	U	U	U	A	U	U	U	U
	IDEBUG	U	U	U	U	A	R	U	U	U
	IBUG2	U	R	U	U	A	U	U	R	U
	ITHRSH	U	R	U	U	A	U	U	U	U

Common Block	Variable	OUT	PLT1	PLT2	POWER	PSDFN	SCALER	TCALC	TCPARM	TRFP	TRANGS	TRANTC	WINDO	WINGEN
INPUTS	IFLAGS	R	R	R	U	R	U	R	R	U	R	U	U	U
	TCDATA	R	U	R	U	U	U	U	U	R	R	U	U	U
	GAS	R	U	U	U	U	U	U	R	U	U	U	U	U
	FREQ	R	R	R	U	R	U	U	U	U	R	R	U	U
	CHANL	R	U	U	U	U	R	U	U	U	U	U	U	U
	IAVDAT	R	U	U	R	R	U	U	U	U	U	U	U	U
	IBLSZ	R	R	R	R	R	R	U	U	U	U	U	R	R
	IREC	R	U	U	U	U	U	U	U	U	U	U	U	U
	TIME	R	R	R	U	U	U	U	U	U	U	U	U	U
	IBSZ	U	R	R	R	R	U	U	U	U	R	U	U	R
	GAMMA	R	U	U	U	U	U	U	U	U	U	U	U	U
	NREC	R	U	U	U	R	U	U	U	U	U	U	U	U
	NRECS	R	R	U	U	R	U	U	U	U	U	U	U	U
	PLTFRQ	R	R	R	U	U	U	U	U	U	U	U	U	U
	TIMTEM	R	R	R	U	U	U	U	U	U	U	U	U	U
	IDEBUG	R	U	U	U	U	U	U	U	U	U	U	U	U
	IBUG2	R	U	U	U	R	U	U	R	U	R	U	U	U
	ITHRSH	R	U	U	U	U	U	U	U	U	U	U	U	U

Common Block	Variable	BLOKDATA	GSPARM	MAIN	POWER	SCALER	TCALC	TCPARM
DATAS	C	A	R	U	U	U	U	R
	TCF	A	U	U	U	U	R	U

Common Block	Variable	INPUT	PLT1	PLT2
PLOTTR	PLOTIT	A	R	R

Usage Codes
A— Altered
R— Referenced
U— Unreferenced

INPUT/OUTPUT

The program requires two different types of files as input. The first of these files is test data in raw voltage. A description of how this file is created and its correct format is included in the Data Preprocessing section. The second input file is a user input file including information such as thermocouple wire dimensions, gas stream properties, test data recording information, and many user options.

Program output consists of five 'intermediate' output files, one 'final' output file, and graphical output. This output provides both graphic and tabulated data for the compensation spectrum, averaged frequency spectrum, and instantaneous time waveform and frequency spectrum. Table 1 lists specific frequency and time domain functions available for output. Plotting routines contain both CALCOMP and DISSPLA calls.

Input Description

The input description in Table 6 illustrates the required form for card input of the specifying parameters in the user input file. Lines 1 through 13 of the input file must always be present, while depending on the various user options, some of lines 14 through 23 may be absent as will be seen in the "Test Cases" section.

TABLE 6. — USER INPUT DESCRIPTION

Variable Name	*Dimensions*	*Type*	*Input Format*	*Units*	*Definition and Options*
Line 1					
TCDATA	(4,2)	R	8F10.0	cm	Thermocouple dimensions as below (1,1) Length of support wire for large diameter thermocouple (2,1) ½ total length of the smaller wire for large thermocouple (3,1) Diameter of support wire for large diameter thermocouple (4,1) Diameter of smaller wire for large diameter thermocouple (1,2), (2,2), (3,2), and (4,2) Same as above for small diameter thermocouple
Line 2					
GAS(1)		R	3F10.0	—	Fuel/air ratio
GAS(2)				K	Mean gas temperature
GAS(4)				—	Mach number
Line 3					
GAS(3)		R	E20.0	N/m^2	Mean gas pressure
Line 4					
FREQ(1)		R	E10.0	Sec	Delta-T setting
Line 5					
FREQ(2) FREQ(3) FREQ(4)		R	3F10.0	Hz	Starting frequency, ending frequency, and frequency increment for which to find the transfer function
Lines 6 and 7					
CHANL	9	R	8F10.0 F10.0	Volts	Contains the gain, output/input, and dc offset values for small, large, and dc thermocouples (i.e., CHANL(1) — CHANL(3) contain information for the small diameter thermocouple)
Line 8					
IAVDAT	2	I	215	—	Starting record number and number of records to use in ensemble averaging
Line 9					
IBLSZ	—	I	15	—	Data block size*
Line 10					
PLTFRQ	—	R	F10.0	Hz	Frequency at which to end plots of frequency domain data
Line 11					
IFLAGS	12	I	1215	—	1) Gamma used for Comp spectrum 1 — Calculate measured gamma 2 — Use a user entered gamma 3 — Use existing compensation spectrum 2) Thermocouple material code 0 — N/A : IFLAGS(1)=3 1 — PT / 6%RH 2 — PT / 30%RH 3 — CR / AL 3) Thermocouple to use for comp spectrum 0 — N/A : IFLAGS(1)=3 1 — Small wire thermocouple 2 — Large wire thermocouple 4) Plot of compensation spectrum? 1 — Yes 2 — No 5) Plot of instantaneous data? 1 — Yes 2 — No

TABLE 6. — USER INPUT DESCRIPTION (Continued)

Variable Name	*Dimensions*	*Type*	*Input Format*	*Units*	*Definition and Options*
					6) Scaling technique to use 1 — Regular power spectral density (PSD) K^2/Hz 2 — Log PSD (10log(PSD)) dB — 0 dB ref 1 K^2/Hz 3 — Linear PSD ($\sqrt{PSD}$) rms K/$\sqrt{Hz}$ 4 — Narrowband frequency rms K
					7) Plot of averaged frequency domain data? 1 — Yes 2 — No
					8) Type averaging desired** 0 — N/A: no plots or printouts 1 — Use user specified number of records 2 — Average only one at a time (user specifies the one)
					9) Plotting option 1 0 — N/A: no plots or printouts 1 — Plot compensated data 2 — Plot uncompensated data
					10) Plotting option 2 0 — N/A: no plots or printouts 1 — Plot time and frequency data 2 — Plot time data only
					11) Plotting option 3 0 — N/A: no plots or printouts 1 — Plot full time range 2 — Plot partial time range
					12) Temperature scaling flag 0 — N/A: no instantaneous plots 1 — Scale each record to its own maximum or minimum temperature 2 — Scale all records to a user specified temperature
Line 12					
IDEBUG	—	I	I5	—	Intermediate write option 1 0 — No writes 1 — Write out the interpolated gammas
Line 13					
IBUG2	—	I	I5	—	Intermediate write option 2*** 0 — No writes 1 — Print out all of the functions generated
Line 14 — Input if IFLAGS(1) = 2					
GAMMA	—	R	E20.0	SI	User input value of gamma ($m^{1.5}$/sec)
Line 15 — Input if IFLAGS(5) = 1 or IBUG2 = 1					
IREC	10	I	10I5	—	Records desired for plotting or printout of the instantaneous spectrum
Line 16 — Input if IFLAGS(11) = 2					
TIME	2	R	2F10.0	Sec	Starting and ending time (with respect to data block) for partial time range plots
Line 17 — Input if IFLAGS(8) = 2					
NREC	10	I	10I5	—	Records of frequency domain data to use when only one record at a time is averaged**

TABLE 6. — USER INPUT DESCRIPTION (Continued)

Variable Name	*Dimensions*	*Type*	*Input Format*	*Units*	*Definition and Options*
Line 18 — Input if IFLAGS(8) = 1					
NRECS	2	I	2I5	—	Starting record and number of records to use in averaging of frequency domain data
Line 19 — Input if IFLAGS(12) = 2					
TIMTEM	—	R	F10.0	K	Temperature to which the instantaneous time domain data are to be scaled
Line 20 — Input if IFLAGS(9) = 1 and IFLAGS(5) = 1 or IBUG2 = 1					
ITHRSH	—	I	I5	dB	Relative threshold level for instantaneous time domain plots
Line 21 — Line 23					
PLOTIT	(20,3)	A	3*20A4	—	User supplied headings for plots

*The block size must be an integer power of 2. Arrays are dimensioned such that the data block size must be less than or equal to 2048. This may be bumped up if necessary.

**'Averaged' frequency domain data are available either averaged or instantaneous. If averaged, input desired information to NRECS. If instantaneous, put record numbers desired into NREC (maximum of 10).

***Note: A peculiarity of the program — If IBUG2 is turned on, at least one record must be requested for instantaneous data. A plot need not be requested (IFLAGS(5) can be set to 2), but a printout will be generated. Therefore, IFLAGS(9), IFLAGS(10), IFLAGS(11), and IFLAGS(12) must all be non-zero.

7222C

Output Description

Program output involves graphic and tabular form of any functions calculated, and files 'intermediate' output files saving time for any information desired after initial run of the program.

Upon initial execution of the program (IFLAGS(1) = 1), five unformatted files are created. These files contain the compensation spectrum (unit 12), scaled data for small and large thermocouples (units 13 and 14 respectively), and Fourier transformed data for small and large thermocouples (units 15 and 16). Once these five files exist, plots may be generated at minimal cost by executing the program with IFLAGS(1)=3.

The main output file (unit 6) will always contain a card copy of the user input file along with a 'summary' of the input. This 'summary' aids in locating incorrect inputs. Depending on the options in effect, the file might also contain step by step details of the interpolation process for calculating a measured gamma (IDEBUG=1), and tabular forms of all functions calculated (IBUG2=1).

Plots of the compensation spectrum, averaged frequency domain data, and instantaneous time and frequency domain data may be generated. These functions may be either compensated or uncompensated and the plot will denote this decision. Frequency domain graphs may be plotted to any desired frequency, and may be either averaged or instantaneous (both may be generated at the same time). Instantaneous plots may include either time and frequency on the same plot, or just the time domain. Other input options exist allowing several different presentations for the plots. See the Test Cases Section for examples of the types of plots that may be generated.

PROGRAM PECULIARITIES AND RESTRICTIONS

Input Restrictions

The Dynamic Gas Temperature Measurement System only models one element or leg of a thermocouple. Therefore, for TCDATA, the user should input the average of the measured dimensions of the two elements of each thermocouple.

The average of the properties of the two elements or legs comprising the thermocouple is used when finding the thermocouple parameters. (Example: the average of properties for PT/6% RH element and PT/30% RH element for type B thermocouples.) Thus, specification of 1 or 2 for the thermocouple material code (IFLAGS(2)) will produce identical results.

If IBUG2 is set to 1, at least one record of instantaneous data must be requested.

Program Peculiarities

If a partial time block is plotted for the Instantaneous Time Waveform, the rms value (printed on the plot) is only calculated over the displayed portion of the data block.

The absolute time of the Instantaneous Time Waveform is different for compensated and uncompensated data. For compensated data, the plot is shifted up in time by one quarter of a record. For example if Record 1 is plotted uncompensated, absolute time is 0 to 0.5 second (for delta T = 0.24415E-3 seconds or delta F = 2 Hz) while for compensated data the absolute time is 0.125 to 0.625 second. See CSFN in the Subroutines section for details of the shifting technique.

There are three different starting points for the Dynamic Gas Temperature Measurement System (IFLAGS(1) = 1, 2, or 3). The initial run for a test point must have IFLAGS(1) = 1. This setting allows the program to find a measured gamma, calculate the compensation spectrum, and create the five 'intermediate' files discussed in Output Description in the previous section. Once these files exist, the user may specify the gamma value for which to calculate the compensation spectrum by setting IFLAGS(1) = 2. For additional plots without any time consuming calculations, the program may be run with IFLAGS(1) = 3.

Restriction on Test Data

Measured data must contain dynamic temperature fluctuations at one or more frequencies lying generally between the corner frequencies of the large and small diameter thermocouples to ensure adequate sensitivity in the in situ measurement of gamma. The analysis technique is based on the use of thermocouples with differing frequency responses. At frequencies much greater than the thermocouple corner frequencies, the frequency response gain functions approach constant slopes (approximately 6 dB/octave). This results in approximately constant valued transfer functions (i.e., ratio of their frequency response functions) as a function of gamma.

Table 7 provides a list of various sizes of thermocouple elements with recommended usable frequency range (compensated) along with sampling frequency, anti-aliasing filter setting, and the spectral line separation of the frequency analysis (i.e., delta F). This table is intended for use as a guideline only, and is based on type B thermocouples operating at the following conditions:

Mean temperature	— 1400K
Pressure	— 1.02E + 6 N/m^2
Mach number	— 0.2
Fuel/air ratio	— 0.02

TABLE 7. — SAMPLE USABLE COMPENSATED FREQUENCY RANGE AND DIGITIZING PARAMETERS FOR VARIOUS DIAMETER TYPE B THERMOCOUPLES

Thermocouple Diameter (µm)	*F_{usable} (Compensated) (Hz)*	*Δt Sampling (sec)*	*Anti-Aliasing Filter Setting (−3 dB) (Hz)*	*ΔF (2048 Block Size) (Hz)*
25.4	6000	40E-6	7500	12.207
50.8	2000	125E-6	2500	3.906
76.2	1000	250E-6	1250	1.953
101.6	800	300E-6	1000	1.628
127.0	600	400E-6	750	1.221
152.4	450	500E-6	563	0.977
177.8	350	625E-6	438	0.781
203.2	300	800E-6	375	0.610
228.6	250	1E-3	312	0.488
254.0	200	1.25E-3	250	0.391
304.8	150	1.5E-3	188	0.326
355.6	125	2.0E-3	156	0.244
381.0	100	2.5E-3	125	0.195
406.4	100	2.5E-3	125	0.195
457.2	90	2.75E-3	113	0.178
508.0	80	3.0E-3	100	0.163
635.0	50	4.5E-3	63	0.109
762.0	40	6.0E-3	50	0.081

7222C

Table 7 was derived using a first order approximation to compute the corner frequency of the thermocouple for specific test conditions, then determining the frequency where the frequency response gain ratio is −35 dB. The usuable frequency (F_{usable}) was then arbitrarily selected to be a convenient value less than the −35 dB frequency point. The sampling frequency was selected to be a convenient value at least four times greater and less than five times the usable frequency (delta t = 1/sampling frequency). The anti-aliasing filter was set to be 1.25 times F_{usable} (based on a minimum of 48 dB/octave roll-off rate for the anti-aliasing filter). The formulas for computing the first order corner frequency are:

$$\tau = \frac{1.4\rho_w C_w D_w^{1.5}}{\sqrt{MP_s}} \left(\frac{T_w}{556}\right)^{-0.18} \tag{24}$$

Where: ρ_w = Density of thermocouple wire (grams/m^3)
C_w = Specific heat of thermocouple wire (Joules/kg K)
D_w = Diameter of thermocouple element (m)
T_w = Mean thermocouple wire temperature (K)
M = Mach number
P_s = Gas stream pressure (N/m^2)

$$\text{Then: } F_c = \frac{1}{2\pi\,\tau} \tag{25}$$

Where: F_c = Corner frequency (i.e., -3 dB gain in Hz).

For a first order system, the gain ratio (i.e., output/input) as a function of frequency is given by the expression:

$$G(f) = \frac{1}{\sqrt{1 + (f/F_c)^2}} \tag{26}$$

Where: f = Frequency (Hz)
F_c = Corner frequency (Hz).

This expression can be manipulated to normalize f as a function of F_c to define f at any desired gain ratio (expressed in dB):

$$f = F_c \sqrt{10^{(-dB/10)} - 1} \tag{27}$$

Note: Observe sign of dB value. It will always be negative for thermocouples.

Thus, the frequency at which the thermocouple gain ratio is −35 dB (a practical limit of compensation) would be computed as:

$$f(-35\ \mathrm{dB}) = F_c \sqrt{10^{(-(-35)/10)} - 1} = F_c\,(56.225) \tag{28}$$

The recommended F_{usable} is an arbitrarily convenient frequency less than f(−35 dB).

Example: 76 µm diameter type B thermocouple operation at the following conditions:

T_w = 1400K
M = 0.2
P_s = 1.02E + 6 N/m²

Type B thermocouple properties at these conditions are:

C_w = 1.933E + 2 Joules/kg K
ρ_w = 2.009E + 4 kg/m³

1) Compute τ:

$$\tau = \frac{(1.4)\,(2.009E + 4)\,(1.933E + 2)\,(76E - 6)^{1.5}}{\sqrt{(0.2)\,(1.02E + 6)}} \left(\frac{1400}{556}\right)^{-0.18} \tag{29}$$

τ = 6.754E-3 seconds

2) Compute F_c:

$$F_c = \frac{1}{2\pi\tau} = \frac{1}{2\pi(6.754E - 3)} = 23.6\ \mathrm{Hz} \tag{30}$$

3) Compute f(−35 dB) and select F_{usable}:

$$f(-35\ \text{dB}) = (23.6)(56.225) = 1327\ \text{Hz} \tag{31}$$

Select F_{usable} 1000 Hz

4) Compute sampling frequency:

$$F_s = 4 \times 1000 = 4000\ \text{Hz} \tag{32}$$

or

$$\Delta t = 1/F_s = 1/4000 = 250.E - 6\ \text{seconds} \tag{33}$$

5) Compute anti-aliasing filter setting:

$$F_a = (1.25)(F_{usable}) = 1250\ \text{Hz} \tag{34}$$

6) Compute FFT spectral line separation (ΔF)

$$\Delta F = F_s/\text{block size} = 4000\ \text{Hz}/2048 = 1.953125\ \text{Hz} \tag{35}$$

The selection of dual thermocouple combinations for the measurement of the aerodynamic parameter, gamma, has not been evaluated in depth other than the 76 µm/254 µm combination developed for specific application to meet the original Contract Statement of Work for 1 kHz compensated bandwidth in an F100 engine environment. It is imperative that the combination be selected judiciously to provide adequate sensitivity in the ratio of their amplitude response as a function of gamma.

Accuracy of Results

A rigorous analysis of all components contributing to the measurement error was beyond the scope of this contract. Table 8 depicts dynamic component errors (expressed as percent of reading) identified and evaluated in the original contract and reported in NASA CR-168267. Notable potential sources of errors that could not be (or were not) evaluated were: (1) dynamic velocity effects, (2) use of the average properties from both thermocouple legs for the single leg finite difference model, and (3) frequency dependency of the aerodynamic parameter gamma.

Results of comparisons made between the dual thermocouple and compensated resistance thermometer data on the rotating wheel and subscale combustor rig experiments (described in Volume I) showed typical dynamic component errors of about −22 percent and +33 percent (time domain 1 kHz bandwidth). The true errors would lie between these and those presented in Table 8. Note that these errors are based on percents of reading of very small dynamic temperature fluctuations. Inclusion of the mean gas path temperature into the error statements would reduce the errors to less than 4 percent of reading.

TABLE 8. — OVERALL ERROR

	Error Due to Data System SNR (%)	Error Due to FFT Compensation Technique (%)	* Error (diameters — %)	** Error (H(f) — %)	Total Error (rss — %)
Averaged Frequency Spectrum					
1000K (1800°F) p-p at 200 Hz	<0.1	1.1	4.9	3.3	6.0
1000K (1800°F) p-p at 1000 Hz	<0.1	2.0	5.2	3.5	6.6
≃ 15K (27°F) p-p/$\sqrt{Hz}$ 4 Hz to 200 Hz	0.7	1.2	4.9	3.3	6.1
≃ 15K (27°F) p-p/$\sqrt{Hz}$ 200 Hz to 1000 Hz	3.8	1.9	5.2	3.5	7.6
≃ 6.7K (12°F) p-p/spectral line at 200 Hz	2.2	1.2	4.9	3.3	6.4
≃ 6.7K (12°F) p-p/spectral line at 1000 Hz	10.0	1.9	5.2	3.5	12.0
Instantaneous Time Waveform					
1000K (1800°F) p-p at 200 Hz	<0.1	1.6	4.9	3.3	6.1
1000K (1800°F) p-p at 1000 Hz	<0.1	2.5	5.2	3.5	6.7
≃ 15K (27°F) p-p/$\sqrt{Hz}$ 4 Hz to 200 Hz	0.7	4.4	4.9	3.3	7.4
≃ 15K (27°F) p-p/$\sqrt{Hz}$ 200 Hz to 1000 Hz	3.8	10.6	5.2	3.5	12.9
≃ 6.7K (12°F) p-p/spectral line 4 Hz to 200 Hz	2.2	4.4	4.9	3.3	7.7
≃ 6.7K (12°F) p-p/spectral line 200 Hz to 1000 Hz	10.0	10.6	5.2	3.5	15.9

$$\frac{\text{rms residule}}{\text{rms input}} \times 100\%$$

* Error due to perturbation of measurement error in thermocouple diameters
** Error due to perturbation of measurement error in H(f)

6854C

PROGRAM LISTING

```
C                                                                       E249
C***********************************************************************E249
C                                                                      *E249
C        THIS IS THE MAIN PROGRAM FOR THE DYNAMIC GAS                  *E249
C             TEMPERATURE MEASUREMENT SYSTEM                           *E249
C                                                                      *E249
C***********************************************************************E249
C                                                                       E249
C              FOR EARLY DOMESTIC DISSEMINATION                         E249
C              --------------------------------                         E249
C                                                                       E249
C         BECAUSE OF ITS SIGNIFICANT EARLY COMMERCIAL POTENTIAL, THIS   E249
C    INFORMATION, WHICH HAS BEEN DEVELOPED UNDER A U.S. GOVERNMENT      E249
C    PROGRAM, IS BEING DISSEMINATED WITHIN THE UNITED STATES IN         E249
C    ADVANCE OF GENERAL PUBLICATION.  THIS INFORMATION MAY BE DUP-      E249
C    LICATED AND USED BY THE RECIPIENT WITH THE EXPRESS LIMITATION      E249
C    THAT IT NOT BE PUBLISHED.  RELEASE OF THIS INFORMATION TO OTHER    E249
C    DOMESTIC PARTIES BY THE RECIPIENT SHALL BE MADE SUBJECT TO THESE   E249
C    LIMITATIONS.                                                       E249
C         FORIEGN RELEASE MAY BE MADE ONLY WITH PRIOR NASA APPROVAL     E249
C    AND APPROPRIATE EXPORT LICENSES.  THIS LEGEND SHALL BE MARKED ON   E249
C    ANY REPRODUCTION OF THIS INFORMATION IN WHOLE OR IN PART.          E249
C                                                                       E249
C***********************************************************************E249
C                                                                       E249
C    CALLS - INPUT:  INPUTS ALL THERMOCOUPLE AND GAS STREAM             E249
C                    INFORMATION, ALONG WITH THE USER OPTIONS.          E249
C            CHECK:  CHECKS FOR ERRORS IN THE USER INPUT AND            E249
C                    TERMINATES EXECUTION IF ANY ARE FOUND.             E249
C            WINGEN: GENERATES THE P301 WINDOWING FUNCTION.             E249
C            TCPARM: CALCULATES THE THERMOCOUPLE PARAMETERS.            E249
C            GSPARM: CALCULATES THE GAS STREAM PARAMETERS.              E249
C            PLT2:   PLOTS THE COMPENSATION SPECTRUM.                   E249
C            TRANGS: GENERATES THE TRANSFER FUNCTION OF THE             E249
C                    THERMOCOUPLE TO THE GAS STREAM.                    E249
C            TRANTC: GENERATES THE TRANSFER FUNCTION BETWEEN            E249
C                    THE TWO THERMOCOUPLES.                             E249
C            POWER:  CALCULATES THE AUTO AND CROSS POWER                E249
C                    SPECTRUMS.                                         E249
C            MEASUR: CALCULATES THE MEASURED TRANSFER FUNCTION          E249
C                    AND THE COHERENCE FUNCTION.                        E249
C            INTERP: INTERPOLATES BETWEEN THE ESTIMATED TRANSFER        E249
C                    FUNCTIONS FOR A MEASURED VALUE OF GAMMA.           E249
C            PSDFN:  GENERATES AND PLOTS THE AVERAGED FREQUENCY         E249
C                    DOMAIN DATA (POWER SPECTRAL DENSITY FUNCTIONS).    E249
C            CSFN:   GENERATES AND PLOTS THE INSTANTANEOUS TIME AND     E249
C                    FREQUENCY DOMAIN DATA.                             E249
C                                                                       E249
C    FILES USED:                                                        E249
C                 6 - WRITES THE COMPENSATION SPECTRUM TO THIS FILE     E249
C                     IF THE USER SO DISIRES (IBUG2 = 1)                E249
```

```
C                  12 - READS THE COMPENSATION SPECTRUM FROM THIS FILE   E249
C                       IF USER STARTS AT THIS POINT (IFLAGS(1) = 3)     E249
C                                                                        E249
C***********************************************************************E249
C                                                                        E249
      DIMENSION TC(4),GS(10),TRAN(17,3,1024,2),XX(1025),YY(1025),        E249
     *          XY(1025,2),WINDO(2048),RTRAN(1025,2),COHR(1025),         E249
     *          COMP(1024,2),IBUFF(1000),REA(1024),RIMA(1024),           E249
     *          DATA3(2048),DATA10(2048),DATADC(2048)                    E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),    E249
     *   IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),  E249
     *   PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                               E249
      COMMON /DATAS/ C(45),TCF(11,9)                                     E249
C                                                                        E249
C*** INPUT ALL USER EDITS NEEDED FOR RUNNING THE PROGRAM.                E249
      CALL INPUT                                                         E249
C                                                                        E249
C*** CHECK USER INPUT FOR ERRORS                                         E249
      CALL CHECK                                                         E249
C                                                                        E249
C*** GENERATE THE P301 WINDOWING FUNCTION                                E249
      CALL WINGEN(WINDO)                                                 E249
C                                                                        E249
C*** EVALUATE THE T/C AND GAS STREAM PARAMETERS NEEDED TO FIND           E249
C*** THE TRANSFER FUNCTION                                               E249
      IF(IFLAGS(1).EQ.3) GO TO 5                                         E249
      CALL TCPARM(TC)                                                    E249
      CALL GSPARM(TC,GS)                                                 E249
      GO TO 50                                                           E249
C                                                                        E249
C*** IF USER WISHES OLD COMPENSATION SPECTRUM TO BE USED, READ IN        E249
C*** AND PLOT THE COMPENSATION SPECTRUM AND ALL RELATIVE DATA            E249
    5 CALL PLOTSP(IBUFF,1000,0,4)                                        E249
      REWIND 12                                                          E249
      ISZ = IBSZ - 1                                                     E249
      READ(12) (COMP(J,1),J=1,ISZ), (COMP(J,2),J=1,ISZ),                 E249
     *         IFLAGS(2),IFLAGS(3),GMAMET,((TCDATA(I,J),I=1,4),J=1,2)    E249
      GS(10) = GMAMET * 5.9425                                           E249
      G = GS(10)                                                         E249
      CALL TCPARM(TC)                                                    E249
      CALL GSPARM(TC,GS)                                                 E249
      GS(10) = G                                                         E249
      DO 10 I = 1,ISZ                                                    E249
      REA(I) = COMP(I,1)                                                 E249
      RIMA(I) = COMP(I,2)                                                E249
   10 CONTINUE                                                           E249
      IF(IFLAGS(4).EQ.1) CALL PLT2(REA,RIMA,1,GS,0,0.0)                  E249
      IF(IBUG2.EQ.0) GO TO 30                                            E249
      WRITE(6,80)                                                        E249
      ISTOP = ISZ/2                                                      E249
      DO 20 I = 1,ISTOP                                                  E249
      II = I + ISTOP                                                     E249
      WRITE(6,90) I,REA(I),RIMA(I),II,REA(II),RIMA(II)                   E249
```

```
   20 CONTINUE                                                           E249
C-------- CHANGE TO RECTANGULAR COORDINATES                              E249
   30 DO 40 I = 1,ISZ                                                    E249
      GAIN = COMP(I,1)                                                   E249
      COMP(I,1) = GAIN • COS(COMP(I,2) * 3.1415927 / 180.)               E249
      COMP(I,2) = GAIN • SIN(COMP(I,2) * 3.1415927 / 180.)               E249
   40 CONTINUE                                                           E249
      GO TO 70                                                           E249
C                                                                        E249
C*** IF GAMMA IS BEING INPUT BY USER, JUMP DIRECTLY TO THE               E249
C*** COMPENSATION SPECTRUM                                               E249
   50 IF(IFLAGS(1).EQ.2) GS(10) = GAMMA                                  E249
      IF(IFLAGS(1).EQ.2) GO TO 60                                        E249
C                                                                        E249
C*** FIND THE TRANSFER FUNCTIONS FOR THE T/C'S -VS- GAS STREAM.          E249
      CALL TRANGS(1,17,GS,TC,TRAN,COMP)                                  E249
      CALL TRANGS(2,17,GS,TC,TRAN,COMP)                                  E249
C                                                                        E249
C*** FIND THE TRANSFER FUNCTION BETWEEN THE TWO T/C'S.                   E249
      CALL TRANTC(17,TRAN)                                               E249
C                                                                        E249
C*** READ IN THE DIGITIZED TEST DATA AND FIND THE AUTO AND CROSS POWER   E249
C*** SPECTRUMS NEEDED FOR THE MEASURED TRANSFER FUNCTION.                E249
      CALL POWER(DATA3,DATA10,DATADC,XX,YY,XY,WINDO)                     E249
C                                                                        E249
C*** EVALUATE THE MEASURED TRANSFER FUNCTION.                            E249
      CALL MEASUR(XX,YY,XY,RTRAN,COHR)                                   E249
C                                                                        E249
C*** INTERPOLATE FOR A MEASURED VALUE OF GAMMA.                          E249
      CALL INTERP(TRAN,RTRAN,COHR,GS)                                    E249
C                                                                        E249
C*** INITIALIZE THE CALCOMP PLOTTER                                      E249
   60 CALL PLOTSP(IBUFF,1000,0,4)                                        E249
C                                                                        E249
C*** FIND THE COMPENSATION SPECTRUM FOR THE USER SPECIFIED T/C           E249
      IF(IFLAGS(3).EQ.1) CALL TRANGS(2,1,GS,TC,TRAN,COMP)                E249
      IF(IFLAGS(3).EQ.2) CALL TRANGS(1,1,GS,TC,TRAN,COMP)                E249
C                                                                        E249
C*** EVALUATE THE POWER SPECTRAL DENSITY FUNCTION FOR PLOTTING           E249
   70 CALL PSDFN(COMP,WINDO,ALSS)                                        E249
C                                                                        E249
C*** EVALUATE THE COMPENSATION SPECTRA FOR PLOTTING                      E249
      CALL CSFN(COMP,WINDO,GS,ALSS)                                      E249
C                                                                        E249
C*** DONE! DISCONECT PLOTTER                                             E249
      CALL PLOT(0.,0.,999)                                               E249
      WRITE(6,100)                                                       E249
      STOP                                                               E249
   80 FORMAT('1','THE COMPENSATION SPECTRUM IN POLAR GAIN AND PHASE')    E249
   90 FORMAT(' ',2(2X,'I = ',I4,3X,E13.7,3X,E13.7,6X))                  E249
  100 FORMAT('0',/,' ','EXECUTION OF THE PROGRAM HAS BEEN COMPLETED!!')  E249
      END                                                                E249
C                                                                        E249
C                                                                        E249
```

```
      BLOCK DATA                                                        E249
      COMMON /DATAS/ C(45),TCF(11,9)                                    E249
C                                                                       E249
      DATA C /38.926,  1.8746E-3,  2.1226E-6,  -2.7962E-10, 3.2070E-2,  E249
     *     4.8648E-6, -3.8201E-13,-1.0204E-13,  2.6336E-4, -2.4880E-8,  E249
     *     1.4592E-11,-1.5870E-15, 30.239   ,   1.0526E-2, -1.8102E-6,  E249
     *     1.1490E-10, 3.9228E-2,  4.8327E-6,   3.3457E-9, -1.7809E-12, E249
     *     2.2544E-16, 1.9764E-4,  3.2121E-8,  -1.5888E-11, 2.9097E-15, E249
     *     1.8998E-5,  1.4023E-2,  2.7857E-5,   2.4733E-1, -3.4000E-5,  E249
     *     1.3750   ,  1.9429E-5,  1.4041E-2,   2.7400E-5,  2.4413E-1,  E249
     *    -3.4500E-5,  1.369    ,  1.9859E-5,   1.4062E-2,  2.7091E-5,  E249
     *     0.23937  , -3.5000E-5,  1.3630   ,   9.0420E-9,  1.6100E-5/  E249
C                                                                       E249
      DATA TCF/ 6.0920   ,    16.3490   ,    52.9390   ,    1.4780   ,  E249
     *          6.9130   ,    18.6120   ,    27.4339E-3,  479.1847E-3,  E249
     *        310.3278E-2,   137.6271E-1,    17.4160   ,   29.4811E-2,  E249
     *         19.4989E-2,    54.6597E-3,    13.4771E-1,   36.7985E-2,  E249
     *         17.0955E-2,   683.6671E-1,   438.4414E-2,  742.7122E-3,  E249
     *        182.0053E-3,  1105.5832E-4,    -7.9599E-1,  -21.8787E-1,  E249
     *        -18.9363E-1,   -99.1914E-2,   -15.4338E-1,  -21.8181E-1,  E249
     *       -885.0557E-3,  -110.0944E-2,  -130.4843E-2, -150.4887E-2,  E249
     *       -925.4837E-3,    15.1489E+1,    52.8841E+1,   15.3262E+2,  E249
     *         23.4314E+1,    92.5658E+1,    23.1347E+2,  163.3668   ,  E249
     *        455.2585   ,   110.4443E+1,   244.3852E+1, 3606.8799E-1,  E249
     *         14.7235E+1,    22.5012E+1,    80.7975E+1,   17.7489E+1,  E249
     *         49.4519E+1,    86.9542E+1,   404.1203E-1,  164.9036   ,  E249
     *         40.5648E+1,   891.8105   , 31215.7227E-2,  -14.4194E-1,  E249
     *        -40.4444E-1,    42.3674   ,   -18.7673   ,  -24.9707   ,  E249
     *        -14.0391   ,  -192.8691E-1,  -561.0958E-1, -911.3308E-1,  E249
     *       -101.6005   ,  -299.8243E-1,    27.6855E-1,    0.0      ,  E249
     *         16.7773   ,    73.3276E-1,    53.3789E-1,   30.3267   ,  E249
     *        152.4649E-1,   391.1915E-1,   435.5711E-1,  677.5431E-1,  E249
     *       1035.7422E-2, 10*0.0,  -569.6789E-2,  10*0.0,  2.5058594/  E249
      END                                                               E249
C                                                                       E249
C                                                                       E249
      SUBROUTINE CHECK                                                  E249
C                                                                       E249
C***********************************************************************E249
C     SUBROUTINE 'CHECK' CHECKS THE USER INPUT FLAGS FOR ERRORS AND    *E249
C           TERMINATES THE PROGRAM IF ERRORS ARE FOUND                 *E249
C***********************************************************************E249
C                                                                       E249
C     CALLED BY MAIN PROGRAM                                            E249
C     CALLS - TERM:  TERMINATES THE PROGRAM DUE TO ERROR IN IFLAGS      E249
C                    WRITES OUT AN APPROPRIATE MESSAGE                  E249
C                                                                       E249
C***********************************************************************E249
C                                                                       E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),   E249
     *   IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),E249
     *   PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                              E249
C                                                                       E249
      IF(IFLAGS(1).GT.3.OR.IFLAGS(1).LT.1) CALL TERM(1)                 E249
```


7197C

```
      IF(IFLAGS(2).GT.3.OR.IFLAGS(2).LT.0) CALL TERM(2)                 E249
      IF(IFLAGS(3).GT.2.OR.IFLAGS(3).LT.0) CALL TERM(3)                 E249
      IF(IFLAGS(4).GT.2.OR.IFLAGS(4).LT.1) CALL TERM(4)                 E249
      IF(IFLAGS(5).GT.2.OR.IFLAGS(5).LT.1) CALL TERM(5)                 E249
      IF(IFLAGS(6).GT.4.OR.IFLAGS(6).LT.1) CALL TERM(6)                 E249
      IF(IFLAGS(7).GT.2.OR.IFLAGS(7).LT.1) CALL TERM(7)                 E249
      IF(IFLAGS(8).GT.2.OR.IFLAGS(8).LT.0) CALL TERM(8)                 E249
      IF(IFLAGS(9).GT.2.OR.IFLAGS(9).LT.0) CALL TERM(9)                 E249
      IF(IFLAGS(10).GT.2.OR.IFLAGS(10).LT.0) CALL TERM(10)              E249
      IF(IFLAGS(11).GT.2.OR.IFLAGS(11).LT.0) CALL TERM(11)              E249
      IF(IFLAGS(12).GT.2.OR.IFLAGS(12).LT.0) CALL TERM(12)              E249
      IF(IFLAGS(1).EQ.2.AND.GAMMA.EQ.0) CALL TERM(13)                   E249
      IF(IFLAGS(5).EQ.1.AND.IREC(1).EQ.0) CALL TERM(14)                 E249
      IF(IFLAGS(11).EQ.2.AND.TIME(2).EQ.0) CALL TERM(15)                E249
      IF(TIME(2).LT.TIME(1)) CALL TERM(16)                              E249
      IF(IFLAGS(8).EQ.2.AND.NREC(1).EQ.0) CALL TERM(17)                 E249
      IF(IFLAGS(8).EQ.1.AND.NRECS(2).EQ.0) CALL TERM(18)                E249
      IF(IFLAGS(12).EQ.2.AND.TIMTEM.EQ.0) CALL TERM(19)                 E249
      RETURN                                                            E249
      END                                                               E249
C                                                                       E249
C                                                                       E249
      SUBROUTINE CSFN(COMP,WINDO,GS,ALSS)                               E249
C                                                                       E249
C***********************************************************************E249
C  CSFN EVALUATES THE COMPENSTED SPECTRA FUNCTION FOR PLOTTING         *E249
C***********************************************************************E249
C                                                                       E249
C ---- IDENTIFICATION ----                                              E249
C                                                                       E249
C                                                                       E249
C** COMP - COMPENSATION SPECTRUM                                        E249
C     -- INPUT                                                          E249
C                                                                       E249
C** WINDO - P301 WINDOWING FUNCTION                                     E249
C     -- INPUT                                                          E249
C                                                                       E249
C** GS - ARRAY CONTAINING THE GAS STREAM PARAMETERS, NEEDED IN THE      E249
C        PLOTTING ROUTINE                                               E249
C     -- INPUT                                                          E249
C                                                                       E249
C** ALSS - AREA LINE SHAPE SQUARED                                      E249
C     -- INPUT                                                          E249
C                                                                       E249
C     CALLED BY MAIN PROGRAM                                            E249
C     CALLS - PLT1: PLOTS THE INPUT ARRAY                               E249
C             PLT2: PLOTS THE TWO INPUT ARRAYS ON ONE PAGE              E249
C             FFT:  A ROUTINE THAT PERFORMS THE DIRECT AND              E249
C                   THE INVERSE FOURIER TRANSFORMS                      E249
C                                                                       E249
C     FILES USED:                                                       E249
C                    6 - WRITES THE FUNCTIONS EVALUATED TO THIS FILE    E249
C                        IF THE USED SO DESIRES (IBUG2 = 1)             E249
C                   13 - READS THE SCALED DIGITIZED 3 MIL DATA FROM     E249
```

```
C                            THIS FILE                                   E249
C                     14 - READS THE SCALED DIGITIZED 10 MIL DATA FROM    E249
C                            THIS FILE                                   E249
C                                                                        E249
C***********************************************************************E249
C                                                                        E249
      DIMENSION COMP(1024,2),HANN(2048),DATA(2048),GS(10),               E249
     * WINDO(2048),DATAF(1025),DATA2(2048),TEMP(2048),DUMIMG(1025),      E249
     * CT(1025),ST(1025),POLAR(1024),POLAR2(1024)                        E249
      COMPLEX ARRAY(2048),CCOMP(1024),ARRAY2(2048)                       E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),    E249
     *    IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),E249
     *    PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                              E249
C                                                                        E249
      IF(IBUG2.EQ.0.AND.IFLAGS(5).EQ.2) RETURN                           E249
C                                                                        E249
C*** SET UP THE HANNING WINDOWING FUNCTION                               E249
      DO 10 I = 1,IBLSZ                                                  E249
      HANN(I) = 1.0 + COS(2.0*3.1415927*(I-1)/IBLSZ)                     E249
   10 CONTINUE                                                           E249
      ISIZ = IBSZ - 1                                                    E249
      DO 15 I = 1,ISIZ                                                   E249
      II = I + ISIZ                                                      E249
      TEM = HANN(II)                                                     E249
      HANN(II) = HANN(I)                                                 E249
      HANN(I) = TEM                                                      E249
   15 CONTINUE                                                           E249
C                                                                        E249
C*** LOOP TO READ IN THE SCALED DIGITIZED TEST DATA                      E249
      REWIND 13                                                          E249
      REWIND 14                                                          E249
      IEND = 0                                                           E249
      IREAD = 0                                                          E249
      DO 260 I = 1,400                                                   E249
      IF(IEND.EQ.1) GO TO 270                                            E249
      IF(IREAD.EQ.1) GO TO 20                                            E249
      IF(IFLAGS(3).EQ.1) READ(13,END=270) (DATA(J),J=1,IBLSZ),TMEAN      E249
      IF(IFLAGS(3).EQ.2) READ(14,END=270) (DATA(J),J=1,IBLSZ),TMEAN      E249
      IREAD = 0                                                          E249
      GO TO 40                                                           E249
   20 DO 30 J = 1,IBLSZ                                                  E249
      DATA(J) = TEMP(J)                                                  E249
   30 CONTINUE                                                           E249
      TMEAN = TMN2                                                       E249
      IREAD = 0                                                          E249
C                                                                        E249
C*** CHECK TO SEE IF THIS RECORD IS IN THE DESIRED DATA                  E249
   40 DO 50 J = 1,10                                                     E249
      IF(IREC(J).EQ.I) GO TO 60                                          E249
   50 CONTINUE                                                           E249
      GO TO 260                                                          E249
C                                                                        E249
C*** RECORD IS DESIRED: FIND COMPOSIT INSTANTANEOUS TIME WAVEFORM        E249
```

```
60   IF(IFLAGS(9).EQ.2) GO TO 180                                          E249
C                                                                          E249
C* COMPENSATED DATA DESIRED — TIME DOMAIN                                   E249
C—— READ IN ANOTHER DATA BLOCK FOR SHIFTING                                 E249
     IF(IFLAGS(3).EQ.1) READ(13,END=90) (DATA2(J),J=1,IBLSZ),TMN2          E249
     IF(IFLAGS(3).EQ.2) READ(14,END=90) (DATA2(J),J=1,IBLSZ),TMN2          E249
C                                                                          E249
     IREAD = 1                                                             E249
     DO 70 J = 1,IBLSZ                                                     E249
     TEMP(J) = DATA2(J)                                                    E249
70   CONTINUE                                                              E249
     ISZ = IBSZ - 1                                                        E249
     DO 80 J = IBSZ,IBLSZ                                                  E249
     DATA2(J) = DATA2(J-1024)                                              E249
80   CONTINUE                                                              E249
     DO 85 J = 1,ISZ                                                       E249
     DATA2(J) = DATA(J+1024)                                               E249
85   CONTINUE                                                              E249
     GO TO 100                                                             E249
90   IEND = 1                                                              E249
     DO 95 J = 1,IBLSZ
     DATA2(J) = 0.0
95   CONTINUE
C—— WINDOW THE DATA WITH THE HANNING WINDOW                                 E249
100  DO 110 J = 1,IBLSZ                                                    E249
     DATA(J) = DATA(J) * HANN(J)                                           E249
     DATA2(J) = DATA2(J) * HANN(J)                                         E249
     ARRAY(J) = CMPLX(DATA(J),0.0)                                         E249
     ARRAY2(J) = CMPLX(DATA2(J),0.0)                                       E249
110  CONTINUE                                                              E249
C—— PERFORM THE FORWARD TRANSFORM                                           E249
     CALL FFT(1,IBLSZ,ARRAY)                                               E249
     CALL FFT(1,IBLSZ,ARRAY2)                                              E249
C                                                                          E249
C—— APPLY THE THRESHOLD LEVEL                                               E249
     ISZ = IBSZ - 1                                                        E249
     PMAX = 0.0                                                            E249
     PMAX2 = 0.0                                                           E249
     DO 115 J = 2,ISZ                                                      E249
     CT1 = REAL(ARRAY(J))                                                  E249
     CT2 = REAL(ARRAY2(J))                                                 E249
     ST1 = AIMAG(ARRAY(J))                                                 E249
     ST2 = AIMAG(ARRAY2(J))                                                E249
     POLAR(J) = CT1**2 + ST1**2                                            E249
     POLAR2(J) = CT2**2 + ST2**2                                           E249
     IF(POLAR(J).GT.PMAX) PMAX = POLAR(J)                                  E249
     IF(POLAR2(J).GT.PMAX2) PMAX2 = POLAR2(J)                              E249
115  CONTINUE                                                              E249
     X = PMAX * 10.0**(ITHRSH/10.0)                                        E249
     X2 = PMAX2 * 10.0**(ITHRSH/10.0)                                      E249
C—— DIVIDE BY THE COMPENSATION SPECTRUM                                     E249
     DO 120 J = 2,ISZ                                                      E249
     CCOMP(J) = CMPLX(COMP(J,1),COMP(J,2))                                 E249
     IF(POLAR(J).GE.X) ARRAY(J) = ARRAY(J) / CCOMP(J)                      E249
```

```
      IF(POLAR(J).LT.X) ARRAY(J) = CMPLX(0.0,0.0)                     E249
      IF(POLAR2(J).GE.X2) ARRAY2(J) = ARRAY2(J) / CCOMP(J)            E249
      IF(POLAR2(J).LT.X2) ARRAY2(J) = CMPLX(0.0,0.0)                  E249
  120 CONTINUE                                                        E249
      DO 130 J = 2,ISZ                                                E249
      DUMIMG(J) = - COMP(J,2)                                         E249
      CCOMP(J) = CMPLX(COMP(J,1),DUMIMG(J))                           E249
      JJ = IBLSZ + 2 - J                                              E249
      IF(POLAR(J).GE.X) ARRAY(JJ) = ARRAY(JJ) / CCOMP(J)              E249
      IF(POLAR(J).LT.X) ARRAY(JJ) = CMPLX(0.0,0.0)                    E249
      IF(POLAR2(J).GE.X2) ARRAY2(JJ) = ARRAY2(JJ) / CCOMP(J)          E249
      IF(POLAR2(J).LT.X2) ARRAY2(JJ) = CMPLX(0.0,0.0)                 E249
  130 CONTINUE                                                        E249
C---- INVERSE FOURIER TRANSFORM THE DATA                               E249
      CALL FFT(-1,IBLSZ,ARRAY)                                        E249
      CALL FFT(-1,IBLSZ,ARRAY2)                                       E249
C---- DE-HANNING THE DATA (IN CHANNELS THAT ARE TO BE USED)            E249
      IST = IBLSZ / 4                                                 E249
      IEN = 3 * IST                                                   E249
      DO 140 J = IST,IEN                                              E249
      DATA(J) = REAL(ARRAY(J)) / HANN(J)                              E249
      DATA2(J) = REAL(ARRAY2(J)) / HANN(J)                            E249
  140 CONTINUE                                                        E249
      IF(IEND.EQ.1) GO TO 160                                         E249
C---- SHIFTING THE DATA FOR 'GOOD' DATA BLOCK                          E249
      DO 150 J = 1,ISZ                                                E249
      DATA(J) = DATA(J + ISZ/2)                                       E249
  150 CONTINUE                                                        E249
      DO 155 J = 1,ISZ                                                E249
      DATA(J+ISZ) = DATA2(J + ISZ/2)                                  E249
  155 CONTINUE                                                        E249
      GO TO 180                                                       E249
  160 ISIZ = ISZ/2                                                    E249
      DO 170 J = 1,ISIZ                                               E249
      DATA(J) = 0.0                                                   E249
      DATA(J+3*ISIZ) = 0.0                                            E249
  170 CONTINUE                                                        E249
  180 IF(IFLAGS(10).EQ.2) GO TO 230                                   E249
C                                                                     E249
C*** FREQUENCY DOMAIN                                                  E249
      DO 190 J = 1,IBLSZ                                              E249
      DUMM = DATA(J) * WINDO(J)                                       E249
      ARRAY(J) = CMPLX(DUMM,0.0)                                      E249
  190 CONTINUE                                                        E249
      CALL FFT(1,IBLSZ,ARRAY)                                         E249
      DO 200 J = 1,IBSZ                                               E249
      CT(J) = REAL(ARRAY(J)) * 2.0                                    E249
      ST(J) = AIMAG(ARRAY(J)) * 2.0                                   E249
  200 CONTINUE                                                        E249
C                                                                     E249
C*** PERFORM THE APPROPRIATE SCALING                                   E249
      DO 220 J = 2,IBSZ                                               E249
      DATAF(J) = (CT(J)**2 + ST(J)**2) / 2.0                          E249
      IF(IFLAGS(6).NE.4) GO TO 210                                    E249
```

```
      DATAF(J) = SQRT(DATAF(J))                                          E249
      GO TO 220                                                          E249
  210 DATAF(J) = DATAF(J) / (FREQ(1)*ALSS)                               E249
      IF(IFLAGS(6).EQ.2) DATAF(J) = 10.0* ALOG10(DATAF(J))               E249
      IF(IFLAGS(6).EQ.3) DATAF(J) = SQRT(DATAF(J))                       E249
  220 CONTINUE                                                           E249
C                                                                        E249
C*** PLOT THE DATA                                                       E249
  230 IF(IFLAGS(5).EQ.1.AND.IFLAGS(10).EQ.2) CALL PLT1(DATA,2,I,TMEAN)   E249
      IF(IFLAGS(5).EQ.1.AND.IFLAGS(10).EQ.1)                             E249
     *                              CALL PLT2(DATA,DATAF,2,GS,I,TMEAN)   E249
      IF(IBUG2.EQ.0) GO TO 260                                           E249
      WRITE(6,280) I                                                     E249
      JSTOP = IBLSZ/4                                                    E249
      DO 240 J = 1,JSTOP                                                 E249
      JJ = J + JSTOP                                                     E249
      JJJ = JJ + JSTOP                                                   E249
      JJJJ = JJJ + JSTOP                                                 E249
      WRITE(6,290) J,DATA(J),JJ,DATA(JJ),JJJ,DATA(JJJ),JJJJ,DATA(JJJJ)   E249
  240 CONTINUE                                                           E249
      IF(IFLAGS(10).EQ.2) GO TO 260                                      E249
      WRITE(6,300) I                                                     E249
      JSTOP = IBLSZ/8                                                    E249
      DO 250 J = 1,JSTOP                                                 E249
      JJ = J + JSTOP                                                     E249
      JJJ = JJ + JSTOP                                                   E249
      JJJJ = JJJ + JSTOP                                                 E249
      WRITE(6,290) J,DATAF(J),JJ,DATAF(JJ),                              E249
     *                              JJJ,DATAF(JJJ),JJJJ,DATAF(JJJJ)      E249
  250 CONTINUE                                                           E249
  260 CONTINUE                                                           E249
  270 RETURN                                                             E249
  280 FORMAT('1','THE INSTANTANEOUS TIME DATA FOR RECORD ',I4)           E249
  290 FORMAT(' ',4(2X,'I = ',I4,3X,E13.7,6X))                            E249
  300 FORMAT('1','THE INSTANTANEOUS FREQUENCY DATA FOR RECORD ',I4)      E249
      END                                                                E249
C                                                                        E249
C                                                                        E249
      SUBROUTINE FFT(ISET,N,A)                                           E249
C                                                                        E249
C*********************************************************************** E249
C SUBPROGRAM FFT CALCULATES FINITE COMPLEX FOURIER TRANSFORM OR        * E249
C     THE INVERSE TRANSFORM OF A COMPLEX INPUT ARRAY                   * E249
C*********************************************************************** E249
C                                                                        E249
C —— IDENTIFICATION ——                                                   E249
C                                                                        E249
C** ISET = TYPE OF TRANSFORM                                             E249
C           1 = DIRECT TRANSFORM                                         E249
C          -1 = INVERSE TRANSFORM                                        E249
C     — INPUT                                                            E249
C                                                                        E249
C** N = NUMBER OF DATA POINTS — MUST BE AN INTEGER POWER OF TWO          E249
C       OR THE TRANSFORM WILL NOT BE COMPUTED                            E249
```


7197C

```
C     — INPUT                                                          E249
C                                                                      E249
C** A = ARRAY OF N COMPLEX VALUES TO WHICH TO TRANSFORM IS APPLIED      E249
C        AND PUT BACK INTO THE ARRAY                                   E249
C     — INPUT AND OUTPUT                                               E249
C                                                                      E249
C     CALLED FROM POWER SUBPROGRAM                                     E249
C                 CSFN SUBPROGRAM                                      E249
C                                                                      E249
C ---- IDIOSYNCRACIES ----                                             E249
C                                                                      E249
C     1.  THE FIRST ELEMENT OF THE DIRECT TRANSFORM IS THE MEAN        E249
C         (ZERO FREQUENCY) VALUE.  THE SECOND THROUGH (N/2 + 1)        E249
C         VALUES ARE FOR POSITIVE FREQUENCIES, AND THE REMAINING       E249
C         (N/2 - 2) VALUES ARE FOR NEGATIVE FREQUENCIES.               E249
C                                                                      E249
C     2.  FOR THE DIRECT TRANSFORM, THE RESULTS FOR NEGATIVE           E249
C         FREQUENCIES ARE FOLDED ABOUT THE (N/2 + 1) POINT.            E249
C         THUS THE FREQUENCY FOR THE (N/2 + 2) POINT IS THE            E249
C         NEGATIVE OF THE FREQUENCY FOR THE N/2 POINT, AND SO ON.      E249
C                                                                      E249
C     3.  THE EQUIVALENCE IS USED ONLY FOR DATA SHIFTING WITHIN        E249
C         THIS PARTICULAR SUBROUTINE                                   E249
C                                                                      E249
C********************************************************************* E249
C                                                                      E249
      COMPLEX     A(2048),WN,WJ,T                                      E249
      DIMENSION   W(2)                                                 E249
      EQUIVALENCE (W(1),WN),(W1,W(1)),(W2,W(2))                        E249
C                                                                      E249
C CHECK NUMBER OF DATA                                                 E249
      M=IFIX(ALOG(FLOAT(N))/ALOG(FLOAT(2)) + .01 )                     E249
      IF(2**M.EQ.N) GO TO 10                                           E249
      WRITE(6,110) N                                                   E249
      RETURN                                                           E249
C                                                                      E249
   10 FN = FLOAT(N)                                                    E249
      IF ( ISET ) 40,20,20                                             E249
C                                                                      E249
C DIRECT TRANSFORM IS REQUESTED                                        E249
   20 CONTINUE                                                         E249
      W1 = COS(6.283185/FN)                                            E249
      W2 =-SIN(6.283185/FN)                                            E249
      DO 30 J=1,N                                                      E249
      A(J) = A(J)/FN                                                   E249
   30 CONTINUE                                                         E249
      GO TO 50                                                         E249
C                                                                      E249
C INVERSE TRANSFORM IS REQUESTED                                       E249
   40 CONTINUE                                                         E249
      W1 = COS(6.283185/FN)                                            E249
      W2 = SIN(6.283185/FN)                                            E249
   50 CONTINUE                                                         E249
C                                                                      E249
```

```
C SHUFFLE INPUT DATA ACCORDING TO REVERSE BIT                      E249
C  PATTERN OF ITS SUBSCRIPTS BEFORE TRANSFORMING                   E249
      DO 70 J=2,N                                                  E249
      JJ  = J-1                                                    E249
      JI  = 1                                                      E249
      MML = N                                                      E249
   60 MML = MML/2                                                  E249
      IF ( MOD(JJ,2).EQ.1 ) JI = JI + MML                          E249
      JJ  = JJ/2                                                   E249
      IF ( JJ.NE.0 ) GO TO 60                                      E249
      IF ( JI.LE.J ) GO TO 70                                      E249
      T  = A(J)                                                    E249
      A(J)  = A(JI)                                                E249
      A(JI) = T                                                    E249
   70 CONTINUE                                                     E249
C                                                                  E249
      LM1  = 1                                                     E249
      MML  = N/2                                                   E249
      KLIM = N-1                                                   E249
C                                                                  E249
C DO CASE FOR L=1                                                  E249
      DO 80 K0=1,KLIM,2                                            E249
      K1 = K0+1                                                    E249
      T  = A(K1)                                                   E249
      A(K1) = A(K0) - T                                            E249
      A(K0) = A(K0) + T                                            E249
   80 CONTINUE                                                     E249
C                                                                  E249
C DO CASES FOR REMAINING L'S                                       E249
      DO 100 L=2,M                                                 E249
      LM1  = LM1*2                                                 E249
      LP0  = LM1*2                                                 E249
      MML  = MML/2                                                 E249
      KLIM = N - LP0 +1                                            E249
      JLIM = LM1 - 1                                               E249
C                                                                  E249
C DO CASE FOR J=0                                                  E249
      DO 90 K0 = 1,KLIM,LP0                                        E249
      K1 = K0 + LM1                                                E249
      T  = A(K1)                                                   E249
      A(K1) = A(K0) - T                                            E249
      A(K0) = A(K0) + T                                            E249
   90 CONTINUE                                                     E249
C                                                                  E249
C DO CASE FOR REMAINING J'S                                        E249
      DO 100 J=1,JLIM                                              E249
      J1    = J+1                                                  E249
      KLIM = KLIM + 1                                              E249
      WJ    = WN**(J*MML)                                          E249
      DO 100 K0 = J1,KLIM,LP0                                      E249
      K1 = K0 + LM1                                                E249
      T  = A(K1)*WJ                                                E249
      A(K1) = A(K0) - T                                            E249
      A(K0) = A(K0) + T                                            E249
```

```
 100  CONTINUE                                                          E249
C                                                                       E249
 110  FORMAT(//'******* ERROR IN FFT ** NUMBER OF DATA NOT A POWER OF', E249
     *    ' 2'//'**** N = ',I5,' **** TRANSFORM NOT CALCULATED ****'//)  E249
      RETURN                                                            E249
      END                                                               E249
C                                                                       E249
C                                                                       E249
      SUBROUTINE GET(WW)                                                E249
C                                                                       E249
C*********************************************************************** *E249
C SUBROUTINE GET ACCESSES THE FOURIER TRANSFORMS DESIRED OFF OF DISK   *E249
C  15 OR 16 DEPENDING ON THE TC FOR WHICH THE COMPENSATION SPECTRUM    *E249
C   WAS DESIRED. 'GET' ALSO PERFORMS THE APPROPRIATE SCALEING FOR      *E249
C    THE DESIRED PSD FUNCTION AND FORMS THE AUTO POWER SPECTRUM.       *E249
C************************************************************************E249
C                                                                       E249
C ---- IDENTIFICATION ----                                              E249
C                                                                       E249
C** WW - ARRAY INTO WHICH THE AUTO POWER SPECTRUM WILL BE PLACED        E249
C      --- OUTPUT                                                       E249
C                                                                       E249
C      CALLED BY PDSFN SUBPROGRAM                                       E249
C                                                                       E249
C      FILES USED:                                                      E249
C                   15 - READS THE FOURIER TRANSFORM OF THE 3 MIL       E249
C                        THERMOCOUPLE FROM THIS FILE.                   E249
C                   16 - READS THE FOURIER TRANSFORM OF THE 10 MIL      E249
C                        THERMOCOUPLE FROM THIS FILE.                   E249
C                                                                       E249
C************************************************************************E249
C                                                                       E249
      DIMENSION WW(1025),CT(1025),ST(1025)                              E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),   E249
     *    IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),E249
     *    PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                             E249
C                                                                       E249
C*** LOOP TO READ ALL THE DATA OFF OF THE DISK                          E249
      IF(IFLAGS(3).EQ.1) READ(15) (CT(J),J=1,IBSZ),(ST(J),J=1,IBSZ)     E249
      IF(IFLAGS(3).EQ.2) READ(16) (CT(J),J=1,IBSZ),(ST(J),J=1,IBSZ)     E249
C                                                                       E249
C*** PERFORM THE APPROPRIATE SCALING AND THE PSD FUNCTION               E249
      DO 10 J = 1,IBSZ                                                  E249
      WW(J) = (CT(J)**2 + ST(J)**2) / 2.0                               E249
   10 CONTINUE                                                          E249
      RETURN                                                            E249
      END                                                               E249
C                                                                       E249
C                                                                       E249
      SUBROUTINE GSPARM(TC,GS)                                          E249
C                                                                       E249
C************************************************************************E249
C         GSPARM CALCULATES THE GAS STREAM PARAMETERS AND              *E249
C                PUTS THEM INTO THE ARRAY "GS"                         *E249
```

```
C***********************************************************************E249
C                                                                       E249
C ---- IDENTIFICATION ----                                              E249
C                                                                       E249
C                                                                       E249
C** TC - ARRAY OF THERMOCOUPLE PARAMETERS, NEEDED TO FIND GAS STREAM    E249
C        PARAMETERS                                                     E249
C       -- INPUT                                                        E249
C                                                                       E249
C** GS - THE ARRAY OF GAS STREAM PARAMETERS CALCULATED, LISTED BELOW    E249
C       -- OUTPUT                                                       E249
C                                                                       E249
C  1. DENSITY (RHO)                                                     E249
C  2. THERMAL CONDUCTIVITY (XK)                                         E249
C  3. SPECIFIC HEAT (CP)                                                E249
C  4. SPECIFIC HEAT RATIO (GA)                                          E249
C  5. VISCOSITY (XMU)                                                   E249
C  6. SONIC VELOCITY (SONVL)                                            E249
C  7. KINETIC VISCOSITY (G)                                             E249
C  8. PRANDTL NUMBER (PR)                                               E249
C  9. MEAN GAS VELOCITY (U)                                             E249
C 10. AERODYNAMIC PARAMETER (GMA)                                       E249
C                                                                       E249
C     CALLED BY MAIN PROGRAM                                            E249
C                                                                       E249
C     FILES USED:                                                       E249
C                  6 - WRITES THE ESTIMATED VALUE OF GAMMA TO THIS FILE E249
C                                                                       E249
C ---- IDIOSYNCRACIES ----                                              E249
C                                                                       E249
C EQUATIONS USED TO CALCULATE PARAMETERS 2, 3, AND 4 ARE DEPENDENT      E249
C ON THE FUEL TO AIR RATIO, GAS(1).                                     E249
C                                                                       E249
C***********************************************************************E249
C                                                                       E249
      DIMENSION GS(10),TC(4)                                            E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),   E249
     *    IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),E249
     *    PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                             E249
      COMMON /DATAS/ C(45),TCF(11,9)                                    E249
C                                                                       E249
      T=GAS(2)                                                          E249
      P=GAS(3)                                                          E249
      XM=GAS(4)                                                         E249
      FA=GAS(1)                                                         E249
C                                                                       E249
C*** SETTING THE LIMITS AND CHECKING ON THE FUEL/AIR RATIO              E249
      RLIM1=.015                                                        E249
      RLIM2=.025                                                        E249
      IF(FA.LE.RLIM1) GO TO 10                                          E249
      IF(FA.GT.RLIM1.AND.FA.LE.RLIM2) GO TO 20                          E249
      IF(FA.GT.RLIM2) GO TO 30                                          E249
C                                                                       E249
C*** GS PARAMETERS 2,3,&4 FOR  FUEL/AIR < .015                          E249
```

```
   10  XK=C(26)*T+C(27)                                                 E249
       CP=C(28)*T+C(29)                                                 E249
       GA=C(30)*T+C(31)                                                 E249
       GO TO 40                                                         E249
C                                                                       E249
C*** GS PARAMETERS 2,3,&4 FOR .015 < FUEL/AIR < .025                     E249
   20  XK=C(32)*T+C(33)                                                 E249
       CP=C(34)*T+C(35)                                                 E249
       GA=C(36)*T+C(37)                                                 E249
       GO TO 40                                                         E249
C                                                                       E249
C*** GS PARAMETERS 2,3,&4 FOR  FUEL/AIR > .025                           E249
   30  XK=C(38)*T+C(39)                                                 E249
       CP=C(40)*T+C(41)                                                 E249
       GA=C(42)*T+C(43)                                                 E249
C                                                                       E249
C*** CALCULATING THE REST OF THE GAS STREAM PARAMETERS                  E249
   40  XMU=C(44)*T+C(45)                                                E249
       RHO=2.6983*(P/(T+460.))                                          E249
       SONVL=41.454*(SQRT(GA*(T+460.)))                                 E249
       G=XMU/RHO                                                        E249
       PR=3600.*XMU*CP/XK                                               E249
       U=XM*SONVL                                                       E249
       GMA=.48*XK*(PR**(.33333))*(SQRT(U))                              E249
       GMA=GMA/(SQRT(G)*TC(1)*TC(3))                                    E249
C                                                                       E249
C*** PLUGGING THE PARAMETERS INTO THE GS ARRAY                          E249
       GS(1)=RHO                                                        E249
       GS(2)=XK                                                         E249
       GS(3)=CP                                                         E249
       GS(4)=GA                                                         E249
       GS(5)=XMU                                                        E249
       GS(6)=SONVL                                                      E249
       GS(7)=G                                                          E249
       GS(8)=PR                                                         E249
       GS(9)=U                                                          E249
       GS(10)=GMA/3600.                                                 E249
C                                                                       E249
       GMAMET = GS(10) * .168279                                        E249
       IF(IFLAGS(1).EQ.1) WRITE(6,50) GMAMET                            E249
   50  FORMAT('1','THE ESTIMATED GAMMA IS ',E13.7)                      E249
       RETURN                                                           E249
       END                                                              E249
C                                                                       E249
C                                                                       E249
       SUBROUTINE INPUT                                                 E249
C                                                                       E249
C***********************************************************************E249
C      THIS SUBROUTINE INPUTS ALL USER EDITS NEEDED FOR THE PROGRAM    *E249
C***********************************************************************E249
C                                                                       E249
C —— IDENTIFICATION ——                                                   E249
C                                                                       E249
C                                                                       E249
```

```
C   1) TCDATA - DATA FOR THE LARGE (COL1) AND SMALL (COL2) WIRE T/C       E249
C               (1) - LENGTH OF SUPPROT WIRE (CM)                         E249
C               (2) - HALF THE TOTAL LENGTH OF SMALLER WIRE (CM)          E249
C               (3) - DIAMETER OF THE SUPPORT WIRE (CM)                   E249
C               (4) - DIAMETER OF THE SMALLER WIRE (CM)                   E249
C   2) GAS - GAS STREAM DATA                                              E249
C               (1) - FUEL TO AIR RATIO                                   E249
C               (2) - MEAN GAS TEMPREATURE (K)                            E249
C               (3) - MEAN GAS PRESSURE (PA - N/M**2)                     E249
C               (4) - MACH NUMBER                                         E249
C   3) FREQ - FREQUENCY INFORMATION FOR TRANSFER FUNCTIONS                E249
C               (1) - DELTA-T SETTING (SEC)                               E249
C               (2) - STARTING FREQUENCY (HZ)                             E249
C               (3) - ENDING FREQUENCY (HZ)                               E249
C               (4) - FREQUENCY INCREMENT (HZ)                            E249
C        NOTE:  THE PROGRAM CONVERTS DELTA-T TO A DELTA-F SETTING AND     E249
C               COMES AS CLOSE TO THE FREQUENCIES SPECIFIED AS POSSIBLE.E249
C   4) CHANL - CHANNEL INFORMATION                                        E249
C               (1) - CHANNEL A (3MIL) AMPLIFIER GAIN (VOLTS)             E249
C               (2) - VOLTAGE RATIO OF FM TAPE INPUT -TO- FM TAPE OUTPUT E249
C               (3) - CHANNEL A DC OFFSET (VOLTS)                         E249
C               (4) - (9)  -  SAME AS ABOVE FOR CHANNELS B (10MIL)        E249
C                             AND C (DC)                                  E249
C   5) IAVDAT - DATA REQUIRED FOR ENSEMBLE AVERAGING                      E249
C               (1) - STARTING RECORD FOR THE AVERAGING (1,2,ETC)         E249
C               (2) - NUMBER OF RECORDS DESIRED IN THE AVERAGING          E249
C   6) IBLSZ - DATA BLOCK SIZE                                            E249
C   7) PLTFRQ - FREQUENCY AT WHICH TO END PLOTS (HZ)                      E249
C               NOTES: PLOTS START AT ZERO FREQUENCY.                     E249
C                      IF PLTFRQ IS INVALID, PLOTS WILL COVER             E249
C                      THE ENTIRE DATA BLOCK                              E249
C   8) IFLAGS - FLAGS FOR VARIOUS USER OPTIONS                            E249
C               (1) - GAMMA USED FOR COMPENSATION SPECTRUM                E249
C                     1 = CALCULATE MEASURED GAMMA                        E249
C                     2 = USE A USER ENTERED ESTIMATED GAMMA              E249
C                     3 = USE A PREVIOUSLY EVALUATED COMPENSATION         E249
C                         SPECTRUM                                        E249
C               (2) - THE T/C MATERIAL CODE                               E249
C                     0 = NOT APPLICABLE (IFLAGS(1) = 3)                  E249
C                     1 = PT / 6%RH                                       E249
C                     2 = PT / 30%RH                                      E249
C                     3 = CR / AL                                         E249
C               (3) - T/C USED FOR COMPENSATION SPECTRUM                  E249
C                     0 = NOT APPLICABLE (IFLAGS(1) = 3)                  E249
C                     1 = 3 MIL                                           E249
C                     2 = 10 MIL                                          E249
C               (4) - PLOT OF COMPENSATION SPECTRUM DESIRED?              E249
C                     1 = YES                                             E249
C                     2 = NO                                              E249
C               (5) - PLOT OF INSTANTANEOUS DATA DESIRED?                 E249
C                     1 = YES                                             E249
C                     2 = NO                                              E249
C               (6) - TYPE OF FREQUENCY DOMAIN DATA DESRIED FOR PLOTS     E249
C                     1 = REGULAR POWER SPECTRAL DENSITY                  E249
```

```
C                    2 = LOG POWER SPECTRAL DENSITY ( 10*LOG(PSD) )      E249
C                    3 = LINEAR POWER SPECTRAL DENSITY ( SQRT(PSD) )     E249
C                    4 = NARROWBAND FREQUENCY SPECTRUM                   E249
C               (7) - PLOT OF AVERAGED FREQUENCY DOMAIN DATA DESIRED?    E249
C                    1 = YES                                             E249
C                    2 = NO                                              E249
C               (8) - TYPE OF AVERAGING DESIRED FOR FREQUENCY DATA       E249
C                    0 = NOT APPLICABLE (NO PLOT OR PRINT-OUT DESIRED)   E249
C                    1 = AVERAGE IN USER SPECIFIED NUMBER OF RECORDS     E249
C                    2 = AVERAGE ONLY ONE AT A TIME                      E249
C               (9) - GENERAL PLOTTING FLAG                              E249
C                    0 = NOT APPLICABLE (NO PLOTS OR PRINT-OUTS DESIRED)E249
C                    1 = PLOT COMPENSATED DATA                           E249
C                    2 = PLOT UNCOMPENSATED DATA                         E249
C              (10) - GENERAL PLOTTING FLAG                              E249
C                    0 = NOT APPLICABLE (NO PLOTS OR PRINT-OUTS DESIRED)E249
C                    1 = PLOT TIME AND FREQUENCY DATA                    E249
C                    2 = PLOT TIME ONLY                                  E249
C              (11) - GENERAL PLOTTING FLAG                              E249
C                    0 = NOT APPLICABLE (NO PLOTS OR PRINT-OUTS DESIRED)E249
C                    1 = PLOT FULL TIME RANGE                            E249
C                    2 = PLOT PARTIAL TIME RANGE                         E249
C              (12) - PLOTTING FLAG FOR SCALING OF INSTANTANEOUS DATA    E249
C                    0 = NOT APPLICABLE (NO PLOTS OR PRINT-OUTS DESIRED)E249
C                    1 = SCALE EACH RECORD TO IT'S OWN MAX TEMPERATURE   E249
C                    2 = SCALE ALL RECORDS TO A USER INPUT TEMPERATURE   E249
C      9) IDEBUG - FLAG FOR INTERMEDIATE WRITES                          E249
C               0 = NO                                                   E249
C               1 = YES                                                  E249
C     10) IBUG2 - FLAG FOR WRITING ALL FUNCTIONS THAT ARE GENERATED      E249
C                 (COMPENSATION SPECTRUM,TIME AND FREQUENCY DOMAIN DATA) E249
C               0 = NO                                                   E249
C               1 = YES                                                  E249
C     11) GAMMA - VALUE OF GAMMA TO USE IF ONE IS INPUT                  E249
C     12) IREC - RECORDS DESIRED FOR PLOTTING OF INSTANTANEOUS SPECTRUM  E249
C                MAX OF 10 RECS, NEED NOT BE CONSECUTIVE.                E249
C                NOTE: IF IBUG2 = 1, AT LEAST ONE RECORD IS REQUIRED     E249
C     13) TIME - TIMES FOR PLOTTING IF PARTIAL TIME RANGE                E249
C               (1) - STARTING TIME (WITH RESPECT TO THE DATA BLOCK)     E249
C               (2) - ENDING TIME (WITH RESPECT TO THE DATA BLOCK)       E249
C     14) NREC - RECORDS DESIRED FOR INSTANTANEOUS PLOTTING FREQUENCY    E249
C                DOMAIN DATA (IFLAGS(8) = 2)                             E249
C                MAX OF 10 RECORDS, NEED NOT BE CONSECUTIVE              E249
C                NOTE: THE ONLY ALLOWABLE RECORDS ARE THE ONES USED      E249
C                      FOR THE ENSEMBLE AVERAGING (IAVDAT)               E249
C     15) NRECS - RECORDS FOR AVERAGING OF FREQUENCY DOMAIN DATA         E249
C               (1) - STARTING RECORD NUMBER                             E249
C               (2) - NUMBER OF RECORDS IN AVERAGING                     E249
C                NOTE: THE ONLY ALLOWABLE RECORDS ARE THE ONES USED      E249
C                      FOR THE ENSEMBLE AVERAGING (IAVDAT)               E249
C     16) TIMTEM - MAX TEMPERATURE TO WHICH TO SCALE THE TIME DOMAIN DATAE249
C               NOTE: ZERO K IS IN CENTER OF AXIS AND PLOT IS            E249
C                     SCALED TO POS AND NEG TIMTEM ON 4 INCH AXIS.       E249
C     17) ITHRSH - RELATIVE THRESHOLD LEVEL NEEDED FOR THE INSTANTANEOUS E249
```

```
C                 TIME WAVEFORM (ENTERED IN DB)                          E249
C    18) PLOTIT - ARRAY INTO WHICH THE USER ENTERS ANY GENERAL            E249
C                 INFORMATION THAT HE/SHE WISHES TO HAVE PRINTED         E249
C                 OUT ON THE PLOTS.                                      E249
C                                                                        E249
C     CALLED BY MAIN PROGRAM                                             E249
C     CALLS : PRNTIN - PRINTS USER INPUT IN CARD IMAGE FORMAT            E249
C             OUT    - PRINTS THE INPUT WITH DISCRIPTIONS                E249
C                                                                        E249
C     FILES USED:                                                        E249
C                  5 - READS THE USER INPUTS FROM THIS FILE              E249
C                                                                        E249
C --- IDIOSYNCRACIES ---                                                 E249
C                                                                        E249
C NOTE: IN ORDER FOR THE USER INPUT GAMMA (IFLAGS(1) = 2), OR THE        E249
C       PREVIOUSLY EVALUATED COMPENSATION SPECTRUM (IFLAGS(1) = 3)       E249
C       OPTIONS TO BE USED, THIS PROGRAM MUST HAVE BEEN RUN BEFORE       E249
C       SO THAT VALUES OF THE FOURIER TRANSFORMS HAVE BEEN               E249
C       STORED ON DISK FILES 15(3MIL) AND 16(10MIL), THE TIME DOMAIN     E249
C       DIGITIZED DATA ON DISK FILES 13(3MIL) AND 14(10MIL), AND THE     E249
C       COMPENSATION SPECTRUM ON DISK FILE 12.                           E249
C                                                                        E249
C***********************************************************************E249
C                                                                        E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),    E249
     *    IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),E249
     *    PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                              E249
      COMMON /PLOTTR/ PLOTIT(20,3)                                       E249
      DATA BLNK/'    '/                                                  E249
C                                                                        E249
      CALL PRNTIN(5,6)                                                   E249
      READ (5,60) ((TCDATA(I,J),I=1,4),J=1,2)                            E249
      READ (5,60) (GAS(I),I=1,2),GAS(4)                                  E249
      READ (5,90) GAS(3)                                                 E249
      READ (5,70) FREQ(1)                                                E249
      READ (5,60) (FREQ(I),I=2,4)                                        E249
      READ (5,60) (CHANL(I),I=1,9)                                       E249
      READ (5,50) (IAVDAT(I),I=1,2)                                      E249
      READ (5,50) IBLSZ                                                  E249
      READ (5,60) PLTFRQ                                                 E249
      READ (5,50) (IFLAGS(I),I=1,12)                                     E249
      READ (5,50) IDEBUG                                                 E249
      READ (5,50) IBUG2                                                  E249
C*  CHECK TO SEE IF REST OF DATA IS REQUIRED                             E249
      IF(IFLAGS(1).EQ.2) READ (5,90) GAMMA                               E249
      IF(IFLAGS(5).EQ.1.OR.IBUG2.EQ.1) READ (5,50) (IREC(I),I=1,10)      E249
      IF(IFLAGS(11).EQ.2) READ (5,60) (TIME(I),I=1,2)                    E249
      IF(IFLAGS(8).EQ.1) READ (5,50) (NRECS(I),I=1,2)                    E249
      IF(IFLAGS(8).EQ.2) READ (5,50) (NREC(I),I=1,10)                    E249
      IF(IFLAGS(12).EQ.2) READ (5,60) TIMTEM                             E249
      IF((IFLAGS(9).EQ.1.AND.IFLAGS(5).EQ.1).OR.(IFLAGS(9).EQ.1.AND.     E249
     *  IBUG2.EQ.1)) READ(5,50) ITHRSH                                   E249
      DO 10 I = 1,3                                                      E249
      DO 10 J = 1,20                                                     E249
```

```
      PLOTIT(J,I) = BLNK                                                 E249
   10 CONTINUE                                                           E249
      DO 20 I = 1,3                                                      E249
      READ(5,80,END=25) (PLOTIT(J,I),J=1,20)                             E249
   20 CONTINUE                                                           E249
   25 CALL OUT                                                           E249
      IF(IFLAGS(1).EQ.2) GAMMA = GAMMA * 5.9425                          E249
      DO 30 I = 1,4                                                      E249
      TCDATA(I,1) = TCDATA(I,1)*.032808                                  E249
      TCDATA(I,2) = TCDATA(I,2)*.032808                                  E249
   30 CONTINUE                                                           E249
      GAS(2) = 9.0 * (GAS(2)-233.15) / 5.0 - 40.0                        E249
      GAS(3) = GAS(3) / 6894.8                                           E249
      FREQ(1) = 1.0 / (FREQ(1) * IBLSZ)                                  E249
      IBSZ = IBLSZ/2 + 1                                                 E249
   40 RETURN                                                             E249
   50 FORMAT (20I5)                                                      E249
   60 FORMAT (8F10.0)                                                    E249
   70 FORMAT (E10.0)                                                     E249
   80 FORMAT (20A4)                                                      E249
   90 FORMAT (E20.0)                                                     E249
      END                                                                E249
C                                                                        E249
C                                                                        E249
      SUBROUTINE INTERP(TRAN,RTRAN,COHR,GS)                              E249
C                                                                        E249
C*********************************************************************** E249
C  INTERP INTERPOLATES BETWEEN THE ESTIMATED TRANSFER FUNCTIONS OF     * E249
C  THE 10MIL -VS- 3MIL T/C'S IN ORDER TO EVALUATE A MEASURED GAMMA     * E249
C*********************************************************************** E249
C                                                                        E249
C ---- IDENTIFICATION ----                                               E249
C                                                                        E249
C                                                                        E249
C** TRAN - ARRAY CONTAINING THE ESTIMATED TRANSFER FUNCTIONS             E249
C      -- INPUT                                                          E249
C                                                                        E249
C** RTRAN - ARRAY CONTAINING THE MEASURED TRANSFER FUNCTION              E249
C      -- INPUT                                                          E249
C                                                                        E249
C** COHR - ARRAY CONTAINING THE COHERENCE FUNCTION                       E249
C      -- INPUT                                                          E249
C                                                                        E249
C** GS - ARRAY CONTAINING GAMMA VALUE THAT IS UPDATED IN THIS ROUTINE    E249
C      -- INPUT AND OUTPUT                                               E249
C                                                                        E249
C     CALLED BY MAIN PROGRAM                                             E249
C     CALLS - TERM: TERMINATES THE PROGRAM DUE TO NO CALCULATED GAMMA    E249
C                                                                        E249
C     FILES USED:                                                        E249
C                 6 - WRITES OUT THE GAINS OF THE ESTIMATED TRANSFER     E249
C                     FUNCTIONS AT EACH FREQUENCY AND THE INTERPOLATED   E249
C                     VALUE OF GAMMA IF IDEBUG = 1.                      E249
C                                                                        E249
```

```
C                                                                       E249
C --- IDIOSYNCRACIES ---                                                 E249
C                                                                       E249
C INTERP PERFORMS THE FOLLOWING DATA CHECKS:                             E249
C                                                                       E249
C 1. DETERMINES IF THE MEASURED GAIN CROSSES THE THEORETICAL X-FER      E249
C    FUNCTION CURVE.                                                    E249
C                                                                       E249
C 2. DETERMINES IF THE COHERENCE IS WITHIN SPECIFIED LIMITS OF          E249
C    .8<Y**2<1.005.                                                     E249
C                                                                       E249
C 3. DETERMINES IF A COMBINATION OF ABOVE ERRORS WOULD RESULT IN        E249
C    NOT HAVING A MEASURED VALUE OF GAMMA.                              E249
C***********************************************************************E249
C                                                                       E249
      DIMENSION RTRAN(1025,2),COHR(1025),GS(10),TRAN(17,3,1024,2)       E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),   E249
     *    IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),E249
     *    PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                             E249
      AVC=0.                                                            E249
      AVGMA=0.                                                          E249
C                                                                       E249
C*** SET LIMITS FOR COHERENCE FUNCTION                                  E249
      Y2L=.8                                                            E249
      Y2U=1.005                                                         E249
C                                                                       E249
C*** FIND THE APPROPRIATE CHANNELS FOR MEASURED FUNCTION FREQUENCIES    E249
      IL = FREQ(2) / FREQ(1) + 1.49                                     E249
      IU = FREQ(3) / FREQ(1) + 1.49                                     E249
      IN = FREQ(4) / FREQ(1) + .05                                      E249
C                                                                       E249
C*** LOOP THROUGH THE FREQUENCIES                                       E249
      DO 30 IC = IL,IU,IN                                               E249
C                                                                       E249
C*** ACCESS GAIN FROM THE MEASURED AND ESTIMATED TRANSFER FUNCTIONS     E249
      GN = RTRAN(IC,1)                                                  E249
      IF(IDEBUG.EQ.1) WRITE(6,70) IC,GN                                 E249
      DO 10 IR = 1,16                                                   E249
      IRT=IR+1                                                          E249
      PERGAM = FLOAT(IR + 1) / 10.0                                     E249
      PERGM1 = PERGAM + .1                                              E249
      V1 = TRAN(IR,3,IC,1)                                              E249
      V2 = TRAN(IRT,3,IC,1)                                             E249
      IF(IDEBUG.EQ.1) WRITE(6,80) PERGAM,V1                             E249
C                                                                       E249
C*** COMPARE THE MEASURED GAIN TO THOSE OF THE ESTIMATED FUNCTIONS      E249
      IF(GN.LT.V1) GO TO 10                                             E249
      IF(GN.LE.V2) GO TO 20                                             E249
   10 CONTINUE                                                          E249
      GO TO 30                                                          E249
C                                                                       E249
C*** CHECK THE COHERENCE FUNCTION AT THIS FREQUENCY                     E249
   20 IF(IDEBUG.EQ.1) WRITE(6,80) PERGM1,V2                             E249
      Y2 = COHR(IC)                                                     E249
```

```
      IF(Y2.GE.Y2L.AND.Y2.LE.Y2U) GO TO 25                              E249
      IF(IDEBUG.EQ.1) WRITE(6,90) Y2                                    E249
      GO TO 30                                                          E249
C                                                                       E249
C*** FIND THE FRACTION OF GAMMA WHERE THE MEASURED GAIN FALLS            E249
   25 R1 = .2 +.1*(IR-1)                                                 E249
      FRGMA = R1 + (GN-V1)*(.1) / (V2-V1)                               E249
      AVGMA = AVGMA + FRGMA                                             E249
      AVC=AVC+1.                                                        E249
C                                                                       E249
C*** IF DEBUG IS SET, WRITE OUT THE INTERPOLATED GAMMAS                  E249
      IF(IDEBUG.EQ.0) GO TO 30                                          E249
      WRITE(6,40)                                                       E249
      WRITE(6,45)                                                       E249
      FRGMA = FRGMA * GS(10)                                            E249
      FGMAMT = FRGMA * .168279                                          E249
      WRITE(6,50) FGMAMT                                                E249
C                                                                       E249
C*** NEXT FREQUENCY                                                      E249
   30 CONTINUE                                                          E249
      IF(AVC.LE.0.0) CALL TERM(20)                                      E249
C                                                                       E249
C*** FIND AND PRINT THE AVERAGED GAMMA IF ONE WAS FOUND                  E249
      AVGMA=AVGMA/AVC                                                   E249
      GS(10)=GS(10)*AVGMA                                               E249
      GMAMET = GS(10) * .168279                                         E249
      IF(IDEBUG.EQ.1) WRITE(6,60) GMAMET                                E249
      RETURN                                                            E249
   40 FORMAT('0',T5,'INTERPOLATED GAMMA')                               E249
   45 FORMAT('+',T4,'____________________')                             E249
   50 FORMAT(' ',T10,E13.7)                                             E249
   60 FORMAT('0',T5,'AVERAGED GAMMA FOUND = ',E13.7)                    E249
   70 FORMAT(' ','CHANNEL (',I3,')  MEASURED GAIN IS ',E13.7)           E249
   80 FORMAT(' ',5X,'ESTIMATED GAIN FOR ',F4.2,' GAMMA IS ',E13.7)      E249
   90 FORMAT(' ',T3,'UNACCEPTABLE COHERENCE OF ',F10.5)                 E249
      END                                                               E249
C                                                                       E249
C                                                                       E249
      SUBROUTINE INTEST(DATA3,DATA10,DATADC,IFLAG)                      E249
C                                                                       E249
C***********************************************************************E249
C     INTEST READS IN THE DIGITIZED TEST DATA FOR THE 3MIL, 10MIL,     *E249
C               AND DC CHANNEL T/C.                                    *E249
C***********************************************************************E249
C                                                                       E249
C —— IDENTIFICATION ——                                                  E249
C                                                                       E249
C                                                                       E249
C** DATA3 - ARRAY FOR THE 3MIL TEST DATA                                 E249
C     — OUTPUT                                                          E249
C                                                                       E249
C** DATA10 - ARRAY FOR THE 10MIL TEST DATA                               E249
C     — OUTPUT                                                          E249
C                                                                       E249
```

```
C** DATADC - ARRAY FOR THE DC CHANNEL TEST DATA                         E249
C     — OUTPUT                                                          E249
C                                                                       E249
C** IFLAG - FLAG TO SIGNAL THE END OF DATA                              E249
C     — OUTPUT                                                          E249
C                                                                       E249
C     CALLED BY SUBPROGRAM POWER                                        E249
C                                                                       E249
C     FILES USED:                                                       E249
C                  4 - READS THE DIGITIZED TEST DATA FROM THIS FILE.    E249
C                                                                       E249
C***********************************************************************E249
C                                                                       E249
      DIMENSION DATA3(2048),DATA10(2048),DATADC(2048)                   E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),   E249
     *   IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2), E249
     *   PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                              E249
C                                                                       E249
      READ (4,20,END=10) (DATA3(I),I=1,IBLSZ)                           E249
      READ (4,20,END=10) (DATA10(I),I=1,IBLSZ)                          E249
      READ (4,20,END=10) (DATADC(I),I=1,IBLSZ)                          E249
      GO TO 30                                                          E249
   10 IFLAG = 1                                                         E249
   20 FORMAT (1X,6E13.0)                                                E249
   30 RETURN                                                            E249
      END                                                               E249
C                                                                       E249
C                                                                       E249
      SUBROUTINE MEASUR(XX,YY,XY,RTRAN,COHR)                            E249
C                                                                       E249
C********************************************************************** E249
C    MEASUR CALCULATES THE MEASURED TRANSFER FUNCTION OF THE TWO T/C  * E249
C      ALONG WITH THE COHERENCE FUNCTION TO MEASURE ACCURACY.         * E249
C********************************************************************** E249
C                                                                       E249
C —— IDENTIFICATION ——                                                  E249
C                                                                       E249
C                                                                       E249
C** XX - ARRAY CONTAINING THE INPUT AUTO POWER SPECTRUM G 3-3           E249
C     — INPUT                                                           E249
C                                                                       E249
C** YY - ARRAY CONTAINING THE OUTPUT AUTO POWER SPECTRUM G 10-10        E249
C     — INPUT                                                           E249
C                                                                       E249
C** XY - ARRAY CONTAINING THE CROSS POWER SPECTRUM G 3-10               E249
C     — INPUT                                                           E249
C                                                                       E249
C** RTRAN - ARRAY TO CONTAIN THE MEASURED TRANSFER FUNCTION             E249
C     — OUTPUT                                                          E249
C                                                                       E249
C** COHR - ARRAY TO CONTAIN THE COHERENCE FUNCTION                      E249
C     — OUTPUT                                                          E249
C                                                                       E249
C     CALLED BY MAIN PROGRAM                                            E249
```

```
C                                                                         E249
C*********************************************************************** E249
C                                                                         E249
      DIMENSION XX(1025),YY(1025),XY(1025,2),RTRAN(1025,2),COHR(1025)    E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),    E249
     *   IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2), E249
     *   PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                               E249
C                                                                         E249
      DO 10 I = 1,IBSZ                                                    E249
      RTRAN(I,1) = XY(I,1) / XX(I)                                        E249
      RTRAN(I,2) = XY(I,2) / XX(I)                                        E249
      COHR(I) = (XY(I,1)**2 + XY(I,2)**2) / (XX(I) * YY(I))               E249
   10 CONTINUE                                                            E249
      DO 20 I = 1,IBSZ                                                    E249
      RTRAN(I,1) = SQRT(RTRAN(I,1)**2 + RTRAN(I,2)**2)                    E249
C---- UNUSED PHASE, THEREFORE SET TO ZERO                                  E249
      RTRAN(I,2) = 0.0                                                    E249
   20 CONTINUE                                                            E249
      RETURN                                                              E249
      END                                                                 E249
C                                                                         E249
C                                                                         E249
      SUBROUTINE OUT                                                      E249
C                                                                         E249
C*********************************************************************** E249
C   THIS SUBROUTINE PRINTS OUT ALL OF THE USER INPUTS                   * E249
C*********************************************************************** E249
C                                                                         E249
C     CALLED BY INPUT SUBROUTINE                                          E249
C                                                                         E249
C     FILES USED:                                                         E249
C                 6 - WRITES OUT THE USER INPUT ALONG WITH                E249
C                     DISCRIPTIONS OF THE INPUTS.                         E249
C                                                                         E249
C*********************************************************************** E249
C                                                                         E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),    E249
     *    IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),E249
     *    PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                              E249
C                                                                         E249
      WRITE(6,65)                                                         E249
      WRITE(6,70) ((TCDATA(I,1),TCDATA(I,2)),I=1,4)                       E249
      WRITE(6,80) GAS                                                     E249
      WRITE(6,90) FREQ                                                    E249
      WRITE(6,100) CHANL                                                  E249
      WRITE(6,110) IAVDAT                                                 E249
      WRITE(6,120) IBLSZ                                                  E249
      WRITE(6,130) PLTFRQ                                                 E249
      WRITE(6,140)                                                        E249
      WRITE(6,150) IFLAGS                                                 E249
      WRITE(6,160) IDEBUG                                                 E249
      WRITE(6,170) IBUG2                                                  E249
      IF(IFLAGS(1).EQ.2) WRITE(6,180) GAMMA                               E249
      IF(IFLAGS(5).EQ.2.AND.IBUG2.EQ.0) GO TO 30                          E249
```

```
      WRITE(6,190)                                                      E249
      DO 10 I = 1,10                                                    E249
      IF(IREC(I).EQ.0) NUM = I-1                                        E249
      IF(IREC(I).EQ.0) GO TO 20                                         E249
   10 CONTINUE                                                          E249
      NUM = 10                                                          E249
   20 WRITE(6,200) (IREC(I),I = 1,NUM)                                  E249
   30 IF(IFLAGS(11).EQ.2) WRITE(6,210) TIME                             E249
      IF(IFLAGS(8).EQ.1) WRITE(6,220) NRECS                             E249
      IF(IFLAGS(8).NE.2) GO TO 60                                       E249
      WRITE(6,230)                                                      E249
      DO 40 I = 1,10                                                    E249
      IF(NREC(I).EQ.0) NUM = I-1                                        E249
      IF(NREC(I).EQ.0) GO TO 50                                         E249
   40 CONTINUE                                                          E249
      NUM = 10                                                          E249
   50 WRITE(6,200) (NREC(I),I=1,NUM)                                    E249
   60 IF(IFLAGS(12).EQ.2) WRITE(6,240) TIMTEM                           E249
      IF((IFLAGS(9).EQ.1.AND.IFLAGS(5).EQ.1).OR.                        E249
     * (IFLAGS(9).EQ.1.AND.IBUG2.EQ.1)) WRITE(6,250) ITHRSH             E249
      RETURN                                                            E249
   65 FORMAT('1',T15,'10 MIL',T62,'3 MIL')                              E249
   70 FORMAT('+',T15,'______',T62,'_____',//,2(5X,'LENGTH OF SUPPORT WIRE249
     *E   = ',F10.5,' CM',4X),/,2(5X,'LENGTH OF SMALLER WIRE   = ',F10.5E249
     *,' CM',4X),/,2(5X,'DIAMETER OF SUPPORT WIRE = ',F10.5,' CM',4X),/ E249
     *2(5X,'DIAMETER OF SMALLER WIRE = ',F10.5,' CM',4X))               E249
   80 FORMAT('0',T5,'FUEL TO AIR RATIO IS ',T35,F10.5,/,T5,'MEAN GAS TEME249
     *PERATURE (K) IS ',T35,F10.5,/,T5,'MEAN GAS PRESSURE (PA) IS ',T35,E249
     *E13.7,/,T5,'MACH NUMBER IS ',T35,F10.5)                           E249
   90 FORMAT('0',T5,'DELTA-T = ',E13.7,/,T5,'START FREQ = ',F10.4,5X,   E249
     *'END FREQ = ',F10.4,5X,'FREQ INCREMENT = ',F10.4)                 E249
  100 FORMAT('0',T13,'GAIN',T24,'INPUT/OUTPUT',T42,'OFFSET',/,T13,      E249
     *'----',T24,'------------',T42,'------',/,T2,'3 MIL',T10,F10.4,    E249
     *T25,F10.5,T40,F10.6,/,T2,'10 MIL',T10,F10.4,T25,F10.5,T40,F10.6,  E249
     */,T2,'DC',T10,F10.4,T25,F10.5,T40,F10.6)                          E249
  110 FORMAT('0','THE ENSEMBLE AVERAGING STARTS WITH RECORD ',I3,' AND  E249
     *USES ',I3,' RECORDS.')                                            E249
  120 FORMAT(' ','WE HAVE A BLOCKSIZE OF ',I5)                          E249
  130 FORMAT(' ','ALL FREQUENCY DOMAIN PLOTS WILL END AS CLOSE TO ',    E249
     * F10.3,' HZ AS POSSIBLE')                                         E249
  140 FORMAT('0',T6,'FLAGS',T23,'DESCRIPTION',T62,'VALUE')              E249
  150 FORMAT('+',T6,'_____',T23,'___________',T62,'_____',/,T8,'1',T15, E249
     * 'WHERE TO BEGIN PROGRAM CALCULATIONS',T63,I2,/,T8,'2',T15,'T/C MAE249
     *TERIAL CODE',T63,I2,/,T8,'3',T15,'T/C USED FOR COMPENSATION SPECTRE249
     *UM',T63,I2,/,T8,'4',T15,'PLOT OF COMPENSATION SPECTRUM DESIRED?', E249
     *T63,I2,/,T8,'5',T15,'PLOT OF INSTANTANEOUS DATA DESIRED?',T63,I2,/E249
     *,T8,'6',T15,'TYPE OF SCALING DONE TO FREQUENCY DATA',T63,I2,/,T8, E249
     *'7',T15,'PLOT OF AVERGED FREQUENCY DATA DESIRED?',T63,I2,/,T8,'8',E249
     *T15,'AVERAGE ONE OR MANY RECORDS?',T63,I2,/,T8,'9',T15,'COMPENSATEE249
     *D DATA?',T63,I2,/,T8,'10',T15,'PLOT TIME AND FREQUENCY DOMAINS?', E249
     *T63,I2,/,T8,'11',T15,'PLOT FULL TIME RANGE?',T63,I2,/,T8,'12',T15,E249
     *'TEMPERATURE TO SCALE DATA ON PLOTS',T63,I2)                      E249
  160 FORMAT('0','IDEBUG IS SET TO ',I1)                                E249
  170 FORMAT('0','IBUG2 IS SET TO ',I1)                                 E249
```


7197C

```
 180  FORMAT('0','USER INPUT GAMMA IS ',E13.7,' M**1.5/SEC')                 E249
 190  FORMAT('0','THE RECORDS OF INSTANTANEOUS FREQUENCY DOMAIN DATA PLOE249
     *TTED ARE AS FOLLOWS:')                                                E249
 200  FORMAT(' ',10(5X,I3))                                                 E249
 210  FORMAT('0','TIME DOMAIN DATA IS PLOTTED FROM TIME ',F10.7,' TO TIME249
     *E ',F10.7,'    (WITH RESPECT TO THE DATA BLOCK)')                     E249
 220  FORMAT('0','THE AVERAGING OF THE FREQUENCY DOMAIN DATA STARTS WITHE249
     * RECORD ',I3,' AND USES ',I3,' RECORDS.')                             E249
 230  FORMAT('0','RECORDS PLOTTED OF AVERAGED FREQUENCY DOMAIN DATA ARE:E249
     *')                                                                    E249
 240  FORMAT('0','THE TEMPERATURE TO WHICH THE INSTANTANEOUS TIME DOMAINE249
     * DATA IS SCALED IS ',F10.3,' K')                                      E249
 250  FORMAT('0','THE THRESHOLD LEVEL USED FOR TIME DOMAIN DATA IS ',I5,E249
     * ' DB')                                                               E249
      END                                                                   E249
C                                                                           E249
C                                                                           E249
      SUBROUTINE PLT1(ARRAY,ICODE,NUM,TMEAN)                                E249
C                                                                           E249
C***********************************************************************E249
C            SUBROUTINE PLT1 WILL PLOT THE ONE INPUT                       *E249
C                ARRAY ALONG WITH SOME HEADERS                             *E249
C***********************************************************************E249
C                                                                           E249
C ---- IDENTIFICATION ----                                                  E249
C                                                                           E249
C** ARRAY - THE ARRAY FOR WHICH PLOTTING IS DESIRED                         E249
C      -- INPUT                                                             E249
C                                                                           E249
C** ICODE - CODE OF THE FUNCTION TO BE PLOTTED SO THAT THE APPROPRIATE      E249
C            LABELING CAN BE DONE                                           E249
C                   1 = POWER SPECTRAL DENSITY FUNCTION                     E249
C                   2 = COMPENSATED INSTANTANEOUS TIME WAVEFORM             E249
C      -- INPUT                                                             E249
C                                                                           E249
C** NUM - RECORD NUMBER THAT IS BEING PLOTTED                               E249
C      -- INPUT                                                             E249
C                                                                           E249
C** TMEAN - THE MEAN DC TEMPERATURE (NEEDED TO PUT ON PLOTS)                E249
C      -- INPUT                                                             E249
C                                                                           E249
C     CALLED FROM PSDFN SUBPROGRAM                                          E249
C                 CSFN SUBPROGRAM                                           E249
C     CALLS - MANY CALCOMP PLOTTER ROUTINES                                 E249
C                                                                           E249
C                                                                           E249
C***********************************************************************E249
C                                                                           E249
      DIMENSION ARRAY(2048)                                                 E249
C     DIMENSION XARRAY(2048)                                                E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),       E249
     *   IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),     E249
     *   PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                                  E249
```

```
      COMMON /PLOTTR/ PLOTIT(20,3)                                      E249
C                                                                       E249
C*** FIND MAX AND MIN VALUES SO THE ARRAYS CAN BE WINDOWED PROPERLY      E249
      ISIZ = IBSZ - 1                                                   E249
      N = PLTFRQ / FREQ(1) + 1.5                                        E249
      IF(N.GT.ISIZ) N = ISIZ                                            E249
      IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.2) GO TO 20                       E249
      ARRMAX = ARRAY(1)                                                 E249
      ARRMIN = ARRAY(1)                                                 E249
      DO 10 I = 1,IBLSZ                                                 E249
      IF(ARRAY(I).GT.ARRMAX) ARRMAX = ARRAY(I)                          E249
      IF(ARRAY(I).LT.ARRMIN) ARRMIN = ARRAY(I)                          E249
      IF(ICODE.EQ.1.AND.I.EQ.N) GO TO 40                                E249
   10 CONTINUE                                                          E249
      GO TO 40                                                          E249
   20 ICH1 = TIME(1)*IBLSZ*FREQ(1) + 1.49                               E249
      ICH2 = TIME(2)*IBLSZ*FREQ(1) + 1.49                               E249
      ARRMIN = ARRAY(ICH1)                                              E249
      ARRMAX = ARRAY(ICH1)                                              E249
      DO 30 I = ICH1,ICH2                                               E249
      IF(ARRAY(I).LT.ARRMIN) ARRMIN = ARRAY(I)                          E249
      IF(ARRAY(I).GT.ARRMAX) ARRMAX = ARRAY(I)                          E249
   30 CONTINUE                                                          E249
C                                                                       E249
C*** FINDING THE LIMITS OF THE PLOTTING WINDOW                           E249
   40 AMAX = ABS(ARRMIN)                                                E249
      IF(ARRMAX.GT.AMAX) AMAX = ARRMAX                                  E249
      RANGE = ARRMAX - ARRMIN                                           E249
      IF(ICODE.EQ.1.AND.N.EQ.ISIZ) XINC = 400.0                         E249
      IF(ICODE.EQ.1.AND.N.NE.ISIZ) XINC = (N-1) * FREQ(1) / 5.0         E249
      IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.1) XINC = 1.0/(5.0*FREQ(1))       E249
      IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.2) XINC = (TIME(2)-TIME(1))/5.0   E249
      IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.1) YINC = AMAX / 2.0              E249
      IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.2) YINC = TIMTEM / 2.0            E249
      IF(ICODE.EQ.1.AND.IFLAGS(6).NE.2) YINC = ARRMAX / 4.0             E249
      IF(ICODE.EQ.2.OR.IFLAGS(6).NE.2) GO TO 50                         E249
      INTN = (ARRMAX + 9.9) / 10.0                                      E249
      IAMAX = INTN * 10                                                 E249
      YINC = 10.0                                                       E249
      IF(RANGE.GT.40.) YINC = 20.0                                      E249
      YSTRT = IAMAX - 4.0 * YINC                                        E249
C                                                                       E249
C*** START THE PLOTTING                                                  E249
   50 IF(ICODE.EQ.1) CALL PLOT(0.5,2.0,-3)                              E249
      IF(ICODE.EQ.2) CALL PLOT(0.0,4.0,-3)                              E249
C                                                                       E249
      IF(ICODE.EQ.1) CALL AXIS(0.0,0.0,14HFREQUENCY (HZ),-14,5.5,0.,0.0,E249
     * XINC)                                                            E249
      IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.1) CALL AXIS(0.0,-2.,10HTIME (SEC)E249
     *,-10,5.,0.,0.0,XINC)                                              E249
      IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.2) CALL AXIS(0.0,-2.,10HTIME (SEC)E249
     *,-10,5.,0.,TIME(1),XINC)                                          E249
C                                                                       E249
      IF(ICODE.EQ.1.AND.IFLAGS(6).EQ.2) CALL AXIS(0.,0.,8HDECIBELS,8,   E249
```


7197C

```
     *4.,90.,YSTRT,YINC)                                               E249
      IF(ICODE.EQ.1.AND.IFLAGS(6).NE.2) CALL AXIS(0.0,0.0,11HTEMPERATUREE249
     *,11,4.,90.,0.,YINC)                                              E249
C                                                                      E249
      IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.1) CALL AXIS(0.0,-2.,15HTEMPERATURE249
     *E (K),15,4.,90.,-AMAX,YINC)                                      E249
      IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.2) CALL AXIS(0.0,-2.,15HTEMPERATURE249
     *E (K),15,4.,90.,-TIMTEM,YINC)                                    E249
C______________________________________________________________________E249
C     CALLS I WOULD EXPECT TO PREPARE PLOTTING WINDOW IN DISSPLA        E249
C----------------------------------------------------------------------E249
C  50 IF(ICODE.EQ.1) CALL PHYSOR (1.5,2.25)                            E249
C     IF(ICODE.EQ.2) CALL PHYSOR (1.,2.25)                             E249
C     CALL AREA2D(5.,4.)                                               E249
C     IF(ICODE.EQ.1) CALL XNAME(14HFREQUENCY (HZ),14)                  E249
C     IF(ICODE.EQ.2) CALL XNAME(10HTIME (SEC),10)                      E249
C     IF(ICODE.EQ.1.AND.IFLAGS(6).EQ.2) CALL YNAME(8HDECIBELS,8)       E249
C     IF(ICODE.EQ.1.AND.IFLAGS(6).NE.2) CALL YNAME(11HTEMPERATURE,11)  E249
C     IF(ICODE.EQ.2) CALL YNAME(15HTEMPERATURE (K),15)                 E249
C                                                                      E249
C     IF(ICODE.EQ.1.AND.IFLAGS(6).EQ.2)                                E249
C    *               CALL GRAF(0.,XINC,N*FREQ(1),YSTRT,YINC,IAMAX)     E249
C     IF(ICODE.EQ.1.AND.IFLAGS(6).NE.2)                                E249
C    *               CALL GRAF(0.,XINC,N*FREQ(1),0.,YINC,ARRMAX)       E249
C     IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.1.AND.IFLAGS(12).EQ.1)           E249
C    *               CALL GRAF(0.,XINC,.5,-AMAX,YINC,AMAX)             E249
C     IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.1.AND.IFLAGS(12).EQ.2)           E249
C    *               CALL GRAF(0.,XINC,.5,-TIMTEM,YINC,TIMTEM)         E249
C     IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.2.AND.IFLAGS(12).EQ.1)           E249
C    *               CALL GRAF(TIME(1),XINC,TIME(2),-AMAX,YINC,AMAX)   E249
C     IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.2.AND.IFLAGS(12).EQ.2)           E249
C    *               CALL GRAF(TIME(1),XINC,TIME(2),-TIMTEM,YINC,TIMTEM)E249
C______________________________________________________________________E249
C                                                                      E249
C---- PLOT FOR FULL TIME RANGE                                         E249
      RMS = 0.0                                                        E249
      IPEN = 3                                                         E249
      IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.2) GO TO 70                      E249
      DO 60 I = 1,IBLSZ                                                E249
      IF(ICODE.EQ.2) RMS = RMS + ARRAY(I)**2                           E249
      IF(ICODE.EQ.1) XVAL = (I - 1.0)*FREQ(1) / XINC                   E249
      IF(ICODE.EQ.2) XVAL = (I - 1.0) * (1.0 / (FREQ(1)*IBLSZ)) / XINC E249
      IF(ICODE.EQ.1.AND.IFLAGS(6).EQ.2.AND.ARRAY(I).LT.YSTRT)          E249
     *                                                ARRAY(I) = YSTRT E249
      IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.2.AND.ARRAY(I).LT.-TIMTEM)       E249
     *                                                ARRAY(I) = -TIMTEM E249
      IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.2.AND.ARRAY(I).GT.TIMTEM)        E249
     *                                                ARRAY(I) = TIMTEM E249
      YVAL = ARRAY(I) / YINC                                           E249
      IF(ICODE.EQ.1.AND.IFLAGS(6).EQ.2) YVAL = (ARRAY(I) - YSTRT) / YINCE249
      CALL PLOT(XVAL,YVAL,IPEN)                                        E249
      IF(ICODE.EQ.1.AND.I.EQ.N) GO TO 90                               E249
      IPEN = 2                                                         E249
   60 CONTINUE                                                         E249
```

```
      RMS = SQRT(RMS/IBLSZ)                                             E249
      CALL NUMBER(5.07,-1.65,.07,RMS,0.,3)                              E249
      CALL SYMBOL(5.77,-1.65,.07,5HK RMS,0.,5)                          E249
      CALL NUMBER(5.07,-1.8,.07,TMEAN,0.,3)                             E249
      CALL SYMBOL(5.77,-1.8,.07,6HK MEAN,0.,6)                          E249
      GO TO 90                                                          E249
C_______________________________________________________________________E249
C     CALLS FOR DISSPLA TO PLOT THE FULL TIME RANGE PLOTS               E249
C-----------------------------------------------------------------------E249
C     RMS = 0.0                                                         E249
C     IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.2) GO TO 70                       E249
C     DO 60 I = 1,IBLSZ                                                 E249
C     IF(ICODE.EQ.1) XARRY(I) = (I - 1) * FREQ(1)                       E249
C     IF(ICODE.EQ.2) XARRY(I) = (I - 1) * (1.0 / (FREQ(1)*IBLSZ))       E249
C     IF(ICODE.EQ.2) RMS = RMS + ARRAY(I)**2                            E249
C     IF(ICODE.EQ.1.AND.IFLAGS(6).EQ.2.AND.ARRAY(I).LT.YSTRT)           E249
C    *                                            ARRAY(I) = YSTRT      E249
C     IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.2.AND.ARRAY(I).LT.-TIMTEM)        E249
C    *                                            ARRAY(I) = -TIMTEM    E249
C     IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.2.AND.ARRAY(I).GT.TIMTEM)         E249
C    *                                            ARRAY(I) = TIMTEM     E249
C     IF(ICODE.EQ.1.AND.I.EQ.N) GO TO 65                                E249
C 60  CONTINUE                                                          E249
C 65  IF(ICODE.EQ.1) CALL CURVE(XARRY,ARRAY,N,0)                        E249
C     IF(ICODE.EQ.2) CALL CURVE(XARRY,ARRAY,IBLSZ,0)                    E249
C     IF(ICODE.EQ.1) GO TO 90                                           E249
C     RMS = SQRT(RMS/IBLSZ)                                             E249
C     CALL HEIGHT(.07)                                                  E249
C     CALL REALNO(RMS,3,5.07,.35)                                       E249
C     CALL MESSAG(9H    K RMS,9,'ABUT','ABUT')                          E249
C     CALL REALNO(TMEAN,3,5.07,.2)                                      E249
C     CALL MESSAG(9H   K MEAN,9,'ABUT','ABUT')                          E249
C     CALL RESET('HEIGHT')                                              E249
C     GO TO 90                                                          E249
C_______________________________________________________________________E249
C                                                                       E249
C--- PARTIAL TIME RANGE                                                 E249
  70  DO 80 I = ICH1,ICH2                                               E249
      RMS = RMS + ARRAY(I)**2                                           E249
      XVAL = (I-ICH1) * (1.0 /(FREQ(1)*IBLSZ)) / XINC                   E249
      IF(IFLAGS(12).EQ.2.AND.ARRAY(I).LT.-TIMTEM) ARRAY(I) = -TIMTEM    E249
      IF(IFLAGS(12).EQ.2.AND.ARRAY(I).GT.TIMTEM) ARRAY(I) = TIMTEM      E249
      YVAL = ARRAY(I) / YINC                                            E249
      CALL PLOT(XVAL,YVAL,IPEN)                                         E249
      IPEN = 2                                                          E249
  80  CONTINUE                                                          E249
      RMS = SQRT(RMS / (ICH2-ICH1+1))                                   E249
      CALL NUMBER(5.07,-1.65,.07,RMS,0.,3)                              E249
      CALL SYMBOL(5.77,-1.65,.07,5HK RMS,0.,5)                          E249
      CALL NUMBER(5.07,-1.8,.07,TMEAN,0.,3)                             E249
      CALL SYMBOL(5.77,-1.8,.07,6HK MEAN,0.,6)                          E249
C_______________________________________________________________________E249
C     CALLS FOR DISSPLA TO PLOT THE PARTIAL TIME RANGE PLOTS            E249
C-----------------------------------------------------------------------E249
```

```
C                                                                       E249
C 70  DO 80 I = ICH1,ICH2                                               E249
C     II = I - ICH1 + 1                                                 E249
C     XARRY(II) = (I - 1) * (1.0 / (FREQ(1)*IBLSZ))                     E249
C     RMS = RMS + ARRAY(I)**2                                           E249
C     IF(IFLAGS(12).EQ.2.AND.ARRAY(I).LT.-TIMTEM) ARRAY(I) = -TIMTEM    E249
C     IF(IFLAGS(12).EQ.2.AND.ARRAY(I).GT.TIMTEM) ARRAY(I) = TIMTEM      E249
C 80  CONTINUE                                                          E249
C     M = ICH2 - ICH1 + 1                                               E249
C     CALL CURVE(XARRY,ARRAY,M,0)                                       E249
C     RMS = SQRT(RMS/FLOAT(M))                                          E249
C     CALL HEIGHT(.07)                                                  E249
C     CALL REALNO(RMS,3,5.07,.35)                                       E249
C     CALL MESSAG(9H     K RMS,9,'ABUT','ABUT')                         E249
C     CALL REALNO(TMEAN,3,5.07,.2)                                      E249
C     CALL MESSAG(9H    K MEAN,9,'ABUT','ABUT')                         E249
C     CALL RESET('HEIGHT')                                              E249
C_______________________________________________________________________E249
C                                                                       E249
C*** HEADERS FOR THE PLOTS AND THE UNITS OF TEMPERATURE                 E249
  90  IF(ICODE.EQ.1) CALL PLOT(-.5,-2.,-3)                              E249
      IF(ICODE.EQ.2) CALL PLOT(0.0,-4.,-3)                              E249
      IF(ICODE.EQ.2) GO TO 100                                          E249
      CALL SYMBOL(0.,9.5,.2,30HAVERAGED FREQUENCY DOMAIN DATA,0.,30)    E249
      IF(IFLAGS(6).EQ.1) CALL SYMBOL(.5,6.5,.1,9HK**2 / HZ,0.,9)        E249
      IF(IFLAGS(6).EQ.2) CALL SYMBOL(.5,6.5,.1,21H0 DB REF  1 K**2 / HZ,E249
     *  0.,21)                                                          E249
      IF(IFLAGS(6).EQ.3) CALL SYMBOL(.5,6.5,.1,16HRMS K / SQRT(HZ),0.,  E249
     * 16)                                                              E249
      IF(IFLAGS(6).EQ.4) CALL SYMBOL(.5,6.5,.1,5HRMS K,0.,5)            E249
      GO TO 110                                                         E249
 100  CALL SYMBOL(-.3,9.5,.2,36HCOMPOSIT INSTANTANEOUS TIME WAVEFORM,0.,E249
     *36)                                                               E249
C_______________________________________________________________________E249
C     CALLS TO DISSPLA FOR HEADERS                                      E249
C-----------------------------------------------------------------------E249
C 90  CALL ENDGR(0)                                                     E249
C     IF(ICODE.EQ.1) CALL OREL(-.5,-2.25)                               E249
C     IF(ICODE.EQ.2) CALL OREL(0.0,-2.25)                               E249
C     CALL AREA2D(7.,10.5)                                              E249
C     CALL HEIGHT(.2)                                                   E249
C     IF(ICODE.EQ.2) GO TO 100                                          E249
C     CALL MESSAG(30HAVERAGED FREQUENCY DOMAIN DATA,30,0.,9.5)          E249
C     CALL RESET('HEIGHT')                                              E249
C     CALL HEIGHT(.1)                                                   E249
C     IF(IFLAGS(6).EQ.1) CALL MESSAG(9HK**2 / HZ,9,.5,6.5)              E249
C     IF(IFLAGS(6).EQ.2) CALL MESSAG(21H0 DB REF  1 K**2 / HZ,21,       E249
C    *.5,6.5)                                                           E249
C     IF(IFLAGS(6).EQ.3) CALL MESSAG(16HRMS K / SQRT(HZ),16,.5,6.5)     E249
C     IF(IFLAGS(6).EQ.4) CALL MESSAG(5HRMS K,5,.5,6.5)                  E249
C     CALL RESET('HEIGHT')                                              E249
C     GO TO 110                                                         E249
C100  CALL MESSAG(36HCOMPOSIT INSTANTANEOUS TIME WAVEFORM,36,-.3,9.5)   E249
C     CALL RESET('HEIGHT')                                              E249
```

```
C______________________________________________________________E249
C---- TC PLOT IS DONE FOR                                       E249
 110  IF(IFLAGS(3).EQ.1) CALL SYMBOL(1.,9.0,.14,9HSMALL T/C,0.,9)    E249
      IF(IFLAGS(3).EQ.2) CALL SYMBOL(1.,9.0,.14,9HLARGE T/C,0.,9)    E249
C---- COMPENSATED?                                              E249
      IF(IFLAGS(9).EQ.1) CALL SYMBOL(4.,9.0,.14,16HCOMPENSATED DATA,0., E249
     *16)                                                       E249
      IF(IFLAGS(9).EQ.2) CALL SYMBOL(4.,9.0,.14,19HUN-COMPENSATED DATA, E249
     *0.,19)                                                    E249
C---- INSTANTANEOUS DATA                                        E249
      IF(ICODE.EQ.1) GO TO 120                                  E249
      CALL SYMBOL(0.5,8.6,.14,33HINSTANTANEOUS DATA, RECORD NUMBER,0.,  E249
     *33)                                                       E249
      RNUM = NUM + .05                                          E249
      CALL NUMBER(5.2,8.6,.14,RNUM,0.,0)                        E249
      GO TO 130                                                 E249
C_____________________________________________________________ E249
C     CALLS FOR DISSPLA                                         E249
C------------------------------------------------------------- E249
C---- TC PLOT IS DONE FOR                                       E249
C110  IF(IFLAGS(3).EQ.1) CALL MESSAG(9HSMALL T/C,9,1.,9.)       E249
C     IF(IFLAGS(3).EQ.2) CALL MESSAG(9HLARGE T/C,9,1.,9.)       E249
C---- COMPENSATED?                                              E249
C     IF(IFLAGS(9).EQ.1) CALL MESSAG(16HCOMPENSATED DATA,16,4.,9.)     E249
C     IF(IFLAGS(9).EQ.2) CALL MESSAG(19HUN-COMPENSATED DATA,19,4.,9.)  E249
C---- INSTANTANEOUS DATA                                        E249
C     IF(IFLAGS(8).EQ.1) GO TO 120                              E249
C     CALL MESSAG(33HINSTANTANEOUS DATA, RECORD NUMBER,33,.5,8.6)      E249
C     CALL INTNO(NUM,5.2,8.6)                                   E249
C     GO TO 130                                                 E249
C_____________________________________________________________ E249
C---- AVERAGED DATA                                             E249
 120  CALL SYMBOL(4.5,6.5,.07,20HSTARTING REC NUMBER ,0.,20)    E249
      CALL SYMBOL(4.5,6.1,.07,19HRECORDS IN AVERAGE ,0.,19)     E249
      IF(IFLAGS(8).EQ.1) STREC = NRECS(1) + .05                 E249
      IF(IFLAGS(8).EQ.2) STREC = NUM + .05                      E249
      IF(IFLAGS(8).EQ.1) RECNUM = NRECS(2) + .05                E249
      IF(IFLAGS(8).EQ.2) RECNUM = 1.0                           E249
      CALL NUMBER(6.0,6.5,.07,STREC,0.,0)                       E249
      CALL NUMBER(6.0,6.1,.07,RECNUM,0.,0)                      E249
 130  CALL SYMBOL(0.0,8.0,.07,PLOTIT(1,1),0.,80)                E249
      CALL SYMBOL(0.0,7.8,.07,PLOTIT(1,2),0.,80)                E249
      CALL SYMBOL(0.0,7.6,.07,PLOTIT(1,3),0.,80)                E249
      CALL PLOT(0.,0.,-999)                                     E249
C______________________________________________________________E249
C     CALLS FOR DISSPLA                                         E249
C--------------------------------------------------------------E249
C---- AVERAGED DATA                                             E249
C120  CALL HEIGHT(.07)                                          E249
C     CALL MESSAG(20HSTARTING REC NUMBER ,20,4.5,6.5)           E249
C     CALL MESSAG(19HRECORDS IN AVERAGE ,19,4.5,6.1)            E249
C     CALL INTNO(NRECS(1),6.0,6.5)                              E249
C     CALL INTNO(NRECS(2),6.0,6.1)                              E249
C     CALL RESET('HEIGHT')                                      E249
```

```
C130  CALL HEIGHT(.07)                                                  E249
C     CALL MESSAG(PLOTIT(1,1),80,0.0,8.0)                               E249
C     CALL MESSAG(PLOTIT(1,2),80,0.0,7.8)                               E249
C     CALL MESSAG(PLOTIT(1,3),80,0.0,7.6)                               E249
C     CALL RESET('HEIGHT')                                              E249
C     CALL ENDPL(0)                                                     E249
C______________________________________________________________________ E249
      RETURN                                                            E249
      END                                                               E249
C                                                                       E249
C                                                                       E249
      SUBROUTINE PLT2(ARR1,ARR2,ICODE,GS,NUM,TMEAN)                     E249
C                                                                       E249
C********************************************************************** E249
C  SUBROUTINE PLT2 WILL PLOT THE TWO INPUT ARRAYS ON THE SAME PAGE  *   E249
C********************************************************************** E249
C                                                                       E249
C --- IDENTIFICATION ---                                                E249
C                                                                       E249
C** ARR1 - THE FIRST ARRAY TO BE PLOTTED                                E249
C     -- INPUT                                                          E249
C                                                                       E249
C** ARR2 - THE SECOND ARRAY TO BE PLOTTED                               E249
C     -- INPUT                                                          E249
C                                                                       E249
C** ICODE - CODE OF THE FUNCTION TO BE PLOTTED SO THAT THE APPROPRIATE  E249
C           LABELING CAN BE DONE                                        E249
C                  1 = COMPENSATION SPECTRUM                            E249
C                  2 = COMPENSATED INSTANTANEOUS SPECTRUM               E249
C     -- INPUT                                                          E249
C                                                                       E249
C** GS - ARRAY CONTAINING THE VALUE OF GAMMA FOR WHICH THE COMPENSATION E249
C        SPECTRUM WAS FOUND                                             E249
C     -- INPUT                                                          E249
C                                                                       E249
C** NUM - THE RECORD NUMBER BEING PLOTTED (INSTANTANEOUS DATA)          E249
C     -- INPUT                                                          E249
C                                                                       E249
C** TMEAN - THE MEAN DC TEMPERATURE (NEEDED TO PUT ON PLOT)             E249
C     -- INPUT                                                          E249
C                                                                       E249
C     CALLED FROM TRANGS                                                E249
C                 CSFN                                                  E249
C     CALLS - MANY CALCOMP PLOTTER ROUTINES                             E249
C                                                                       E249
C********************************************************************** E249
C                                                                       E249
      DIMENSION ARR1(2048),ARR2(1025),GS(10)                            E249
C     DIMENSION XARRY(2048)                                             E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),   E249
     *   IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2), E249
     *   PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                              E249
      COMMON /PLOTTR/ PLOTIT(20,3)                                      E249
C                                                                       E249
```

```
      YSTRT = 0.0                                                       E249
      ISIZ = IBSZ - 1                                                   E249
      N = PLTFRQ / FREQ(1) + 1.5                                        E249
      IF(N.GT.ISIZ) N = ISIZ                                            E249
C                                                                       E249
C*** CONVERSIONS FOR COMPENSATION SPECTRUM GAIN TO DB'S                 E249
      IF(ICODE.EQ.2) GO TO 20                                           E249
      DO 10 I = 2,ISIZ                                                  E249
      ARR1(I) = 20.0 * ALOG10(ARR1(I))                                  E249
   10 CONTINUE                                                          E249
C                                                                       E249
C*** FIND MAX AND MIN VALUES OF THE ARRAYS FOR WINDOWING REASONS        E249
C------- FOR FULL TIME RANGE                                            E249
   20 IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.2) GO TO 40                       E249
      IF(ICODE.EQ.1) NUMST = 2                                          E249
      IF(ICODE.EQ.1) IEDCH = ISIZ                                       E249
      IF(ICODE.EQ.2) NUMST = 1                                          E249
      IF(ICODE.EQ.2) IEDCH = IBLSZ                                      E249
      AR1MIN = ARR1(NUMST)                                              E249
      AR2MIN = ARR2(NUMST)                                              E249
      AR1MAX = ARR1(NUMST)                                              E249
      AR2MAX = ARR2(NUMST)                                              E249
      DO 30 I = 2,IEDCH                                                 E249
      IF(ARR1(I).LT.AR1MIN) AR1MIN = ARR1(I)                            E249
      IF(ARR1(I).GT.AR1MAX) AR1MAX = ARR1(I)                            E249
      IF(I.GT.N) GO TO 30                                               E249
      IF(ARR2(I).LT.AR2MIN) AR2MIN = ARR2(I)                            E249
      IF(ARR2(I).GT.AR2MAX) AR2MAX = ARR2(I)                            E249
      IF(ICODE.EQ.1.AND.I.EQ.N) GO TO 70                                E249
   30 CONTINUE                                                          E249
      GO TO 70                                                          E249
C------- FOR PARTIAL TIME RANGE                                         E249
   40 ICH1 = TIME(1)*IBLSZ*FREQ(1) + 1.49                               E249
      ICH2 = TIME(2)*IBLSZ*FREQ(1) + 1.49                               E249
      AR1MIN = ARR1(ICH1)                                               E249
      AR2MIN = ARR2(2)                                                  E249
      AR1MAX = ARR1(ICH1)                                               E249
      AR2MAX = ARR2(2)                                                  E249
      DO 60 I = 2,IBLSZ                                                 E249
      IF(I.LT.ICH1.OR.I.GT.ICH2) GO TO 50                               E249
      IF(ARR1(I).LT.AR1MIN) AR1MIN = ARR1(I)                            E249
      IF(ARR1(I).GT.AR1MAX) AR1MAX = ARR1(I)                            E249
   50 IF(I.GT.ICH2.AND.I.GT.N) GO TO 70                                 E249
      IF(I.GT.N) GO TO 60                                               E249
      IF(ARR2(I).LT.AR2MIN) AR2MIN = ARR2(I)                            E249
      IF(ARR2(I).GT.AR2MAX) AR2MAX = ARR2(I)                            E249
   60 CONTINUE                                                          E249
C                                                                       E249
C*** SETTING THE INCREMENTS FOR THE PLOTTED DOMAIN                      E249
   70 AMAX = ABS(AR1MIN)                                                E249
      IF(AR1MAX.GT.AMAX) AMAX = AR1MAX                                  E249
      RANGE = AR2MAX - AR2MIN                                           E249
C                                                                       E249
      IF(ICODE.EQ.1.AND.N.EQ.IBSZ) XINC1 = 400.0                        E249
```

```
      IF(ICODE.EQ.1.AND.N.NE.IBSZ) XINC1 = (N-1) * FREQ(1) / 5.0        E249
      IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.1) XINC1 = 1.0/(5.0*FREQ(1))       E249
      IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.2) XINC1 - (TIME(2)-TIME(1))/5.0  E249
C                                                                        E249
      IF(N.EQ.IBSZ) XINC2 = 400.0                                        E249
      IF(N.NE.IBSZ) XINC2 = (N-1) * FREQ(1) / 5.0                        E249
C                                                                        E249
      IF(ICODE.EQ.1.AND.AR1MIN.LT.-45.) YINC1 = 20.0                     E249
      IF(ICODE.EQ.1.AND.AR1MIN.GE.-45.) YINC1 = 10.0                     E249
      IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.1) YINC1 = AMAX / 2.               E249
      IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.2) YINC1 = TIMTEM / 2.0            E249
C                                                                        E249
      IF(ICODE.EQ.1) YINC2 = 25                                          E249
      IF(ICODE.EQ.2.AND.IFLAGS(6).NE.2) YINC2 = AR2MAX / 4.0             E249
      IF(ICODE.EQ.1.OR.IFLAGS(6).NE.2) GO TO 80                          E249
      INTN = (AR2MAX + 9.9) / 10.0                                       E249
      IAMAX = INTN * 10                                                  E249
      YINC2 = 10.0                                                       E249
      IF(RANGE.GT.40.) YINC2 = 20.0                                      E249
      YSTRT = IAMAX - 4.0 * YINC2                                        E249
C                                                                        E249
C*** BEGINNING THE PLOTTING                                              E249
   80 IF(ICODE.EQ.1) CALL PLOT(0.,4.2,-3)                                E249
      IF(ICODE.EQ.2) CALL PLOT(0.,.5,-3)                                 E249
      IF(ICODE.EQ.2) CALL AXIS(0.0,0.0,14HFREQUENCY (HZ),-14,5.5,0.,0.0,E249
     *XINC2)                                                             E249
      IF(ICODE.EQ.1) CALL AXIS(0.0,0.0,14HFREQUENCY (HZ),14,5.5,0.,0.0, E249
     *XINC2)                                                             E249
      IF(ICODE.EQ.1) CALL AXIS(0.0,-4.0,11HPHASE (DEG),11,4.,90.,-100.0,E249
     *YINC2)                                                             E249
      IF(ICODE.EQ.2.AND.IFLAGS(6).NE.2) CALL AXIS(0.0,0.0,11HTEMPERATUREE249
     *,11,4.,90.,0.0,YINC2)                                              E249
      IF(ICODE.EQ.2.AND.IFLAGS(6).EQ.2) CALL AXIS(0.,0.,8HDECIBELS,8,   E249
     *4.,90.,YSTRT,YINC2)                                                E249
      CALL PLOT(0.,0.,3)                                                 E249
      IPEN = 2                                                           E249
      DO 90 I = 2,IBSZ                                                   E249
      IF(ICODE.EQ.1.AND.I.EQ.IBSZ) GO TO 90                              E249
      IF(I.GT.N) GO TO 100                                               E249
      IF(ICODE.EQ.2.AND.ARR2(I).LT.YSTRT) ARR2(I) = YSTRT                E249
      XVAL = FREQ(1) * (I-1.0) / XINC2                                   E249
      YVAL = ARR2(I) / YINC2                                             E249
      IF(ICODE.EQ.2.AND.IFLAGS(6).EQ.2) YVAL = (ARR2(I) - YSTRT) / YINC2E249
      CALL PLOT(XVAL,YVAL,IPEN)                                          E249
   90 CONTINUE                                                           E249
  100 IF(ICODE.EQ.1) GO TO 110                                           E249
      IF(IFLAGS(6).EQ.1)CALL SYMBOL(.5,4.1,.07,9HK**2 / HZ,0.,9)         E249
      IF(IFLAGS(6).EQ.2) CALL SYMBOL(.5,4.1,.07,21H0 DB REF  1 K**2 / HZE249
     *,0.,21)                                                            E249
      IF(IFLAGS(6).EQ.3) CALL SYMBOL(.5,4.1,.07,16HRMS K / SQRT(HZ),    E249
     *0.,16)                                                             E249
      IF(IFLAGS(6).EQ.4) CALL SYMBOL(.5,4.1,.07,5HRMS K,0.,5)            E249
C________________________________________________________________________E249
C     EXPECTED CALLS TO DISSPLA TO SET UP PLOTTING                       E249
```


7197C

```
C-------------------------------------------------------------------- E249
C 80  IF(ICODE.EQ.1) CALL PHYSOR(1.,.2)                                E249
C     IF(ICODE.EQ.2) CALL PHYSOR(1.,.5)                                E249
C     CALL AREA2D(5.,4.)                                               E249
C     IF(ICODE.EQ.1) CALL CROSS                                        E249
C     IF(ICODE.EQ.1) CALL XNAME(14HFREQUENCY (HZ),-14)                 E249
C     IF(ICODE.EQ.2) CALL XNAME(14HFREQUENCY (HZ),14)                  E249
C     IF(ICODE.EQ.1) CALL YNAME(11HPHASE (DEG),11)                     E249
C     IF(ICODE.EQ.2.AND.IFLAGS(6).EQ.2) CALL YNAME(8HDECIBELS,8)       E249
C     IF(ICODE.EQ.2.AND.IFLAGS(6).NE.2) CALL YNAME(11HTEMPERATURE,11)  E249
C                                                                      E249
C     XMX = N * FREQ(1)                                                E249
C     IF(ICODE.EQ.1) CALL GRAF(0.,XINC2,XMX,-100.,YINC2,0.0)           E249
C     IF(ICODE.EQ.2.AND.IFLAGS(6).NE.2)                                E249
C    *               CALL GRAF(0.,XINC2,XMX,0.0,YINC2,AR2MAX)          E249
C     IF(ICODE.EQ.2.AND.IFLAGS(6).EQ.2)                                E249
C    *               CALL GRAF(0.,XINC2,XMX,YSTRT,YINC2,IAMAX)         E249
C     XARRY(1) = 0.0                                                   E249
C     ARR2(1) = 0.0                                                    E249
C     DO 90 I = 2,IBSZ                                                 E249
C     IF(I.GT.N) GO TO 100                                             E249
C     IF(ICODE.EQ.1.AND.I.EQ.IBSZ) GO TO 90                            E249
C     XARRY(I) = (I - 1.0) * FREQ(1)                                   E249
C     IF(ICODE.EQ.2.AND.ARR2(I).LT.YSTRT) ARR2(I) = YSTRT              E249
C 90  CONTINUE                                                         E249
C100  CALL CURVE(XARRY,ARR2,N,0)                                       E249
C     IF(ICODE.EQ.1) GO TO 110                                         E249
C     CALL HEIGHT(.07)                                                 E249
C     IF(IFLAGS(6).EQ.1)CALL MESSAG(9HK**2 / HZ,9,.5,4.1)              E249
C     IF(IFLAGS(6).EQ.2) CALL MESSAG(21HO DB REF  1 K**2 / HZ,21,      E249
C    *.5,4.1)                                                          E249
C     IF(IFLAGS(6).EQ.3) CALL MESSAG(16HRMS K / SQRT(HZ),16,.5,4.1)    E249
C     IF(IFLAGS(6).EQ.4) CALL MESSAG(5HRMS K,5,.5,4.1)                 E249
C     CALL RESET('HEIGHT')                                             E249
C____________________________________________________________________  E249
C                                                                      E249
 110  IF(ICODE.EQ.1) CALL PLOT(0.,4.9,-3)                              E249
      IF(ICODE.EQ.2) CALL PLOT(0.,6.9,-3)                              E249
      IF(ICODE.EQ.1) CALL AXIS(0.0,0.0,14HFREQUENCY (HZ),14,5.5,0.,    E249
     *0.0,XINC1)                                                       E249
      IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.1) CALL AXIS(0.0,-2.,10HTIME (SEC)E249
     *,-10,5.0,0.,0.0,XINC1)                                           E249
      IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.2) CALL AXIS(0.0,-2.,10HTIME (SEC)E249
     *,-10,5.,0.,TIME(1),XINC1)                                        E249
      IF(YINC1.EQ.10) YST = -40.                                       E249
      IF(YINC1.EQ.20) YST = -80.                                       E249
      IF(ICODE.EQ.1) CALL AXIS(0.0,-4.,9HGAIN (DB),9,4.,90.,YST,YINC1) E249
      IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.1) CALL AXIS(0.0,-2.,15HTEMPERATURE249
     *E (K),15,4.,90.,-AMAX,YINC1)                                     E249
      IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.2) CALL AXIS(0.0,-2.,15HTEMPERATURE249
     *E (K),15,4.,90.,-TIMTEM,YINC1)                                   E249
C____________________________________________________________________  E249
C     DISSPLA CALLS                                                    E249
C-------------------------------------------------------------------- E249
```

```
C110  CALL ENDGR(0)                                                     E249
C      IF(ICODE.EQ.1) CALL OREL(0.,4.7)                                 E249
C      IF(ICODE.EQ.2) CALL OREL(0.,4.9)                                 E249
C      CALL AREA2D(5.,4.)                                               E249
C      IF(ICODE.EQ.1) CALL XNAME(14HFREQUENCY (HZ),-14)                 E249
C      IF(ICODE.EQ.2) CALL XNAME(10HTIME (SEC),10)                      E249
C      IF(ICODE.EQ.1) CALL YNAME(9HGAIN (DB),9)                         E249
C      IF(ICODE.EQ.2) CALL YNAME(15HTEMPERATURE (K),15)                 E249
C      IF(YINC1.EQ.10) YST = -40.                                       E249
C      IF(YINC1.EQ.20) YST = -80.                                       E249
C      XMX = N * FREQ(1)                                                E249
C      IF(ICODE.EQ.1) CALL GRAF(0.0,XINC1,XMX,YST,YINC1,0.0)            E249
C      IF(ICODE.EQ.1) GO TO 115                                         E249
C      IF(IFLAGS(11).EQ.1.AND.IFLAGS(12).EQ.1)                          E249
C     *              CALL GRAF(0.0,XINC1,.5,-AMAX,YINC1,AMAX)           E249
C      IF(IFLAGS(11).EQ.1.AND.IFLAGS(12).EQ.2)                          E249
C     *              CALL GRAF(0.0,XINC1,.5,-TIMTEM,YINC1,TIMTEM)       E249
C      IF(IFLAGS(11).EQ.2.AND.IFLAGS(12).EQ.1)                          E249
C     *              CALL GRAF(TIME(1),XINC1,TIME(2),-AMAX,YINC1,AMAX)  E249
C      IF(IFLAGS(11).EQ.2.AND.IFLAGS(12).EQ.2)                          E249
C     *              CALL GRAF(TIME(1),XINC1,TIME(2),-TIMTEM,YINC1,TIMTEME249
C_______________________________________________________________________E249
C---- PLOT FOR FULL TIME RANGE                                          E249
      RMS = 0.0                                                         E249
      IF(ICODE.EQ.2.AND.IFLAGS(11).EQ.2) GO TO 130                      E249
      IF(ICODE.EQ.2) IST = 1                                            E249
      IF(ICODE.EQ.1) IST = 2                                            E249
      IF(ICODE.EQ.1) CALL PLOT(0.,0.,3)                                 E249
      IPEN = 2                                                          E249
      DO 120 I = IST,IBLSZ                                              E249
      IF(ICODE.EQ.1.AND.I.GT.N) CALL SYMBOL(.1,-4.,.1,19H0 DB REF UNITY E249
     *GAIN,0.,19)                                                       E249
      IF(ICODE.EQ.1.AND.I.GT.N) GO TO 150                               E249
      IF(ICODE.EQ.2) RMS = RMS + ARR1(I)**2                             E249
      IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.2.AND.ARR1(I).GT.TIMTEM)          E249
     *                                              ARR1(I) = TIMTEM    E249
      IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.2.AND.ARR1(I).LT.-TIMTEM)         E249
     *                                              ARR1(I) = -TIMTEM   E249
      IF(ICODE.EQ.1) XVAL = FREQ(1) * (I-1.0) / XINC1                   E249
      IF(ICODE.EQ.2) XVAL = (I-1.0) * (1.0 / (FREQ(1)*IBLSZ)) / XINC1   E249
      YVAL = ARR1(I) / YINC1                                            E249
      CALL PLOT(XVAL,YVAL,IPEN)                                         E249
  120 CONTINUE                                                          E249
      RMS = SQRT(RMS/IBLSZ)                                             E249
      CALL NUMBER(5.07,-1.65,.07,RMS,0.,3)                              E249
      CALL SYMBOL(5.77,-1.65,.07,5HK RMS,0.,5)                          E249
      CALL NUMBER(5.07,-1.8,.07,TMEAN,0.,3)                             E249
      CALL SYMBOL(5.77,-1.8,.07,6HK MEAN,0.,6)                          E249
      GO TO 150                                                         E249
C_______________________________________________________________________E249
C     CALLS TO DISSPLA                                                  E249
C-----------------------------------------------------------------------E249
C---- PLOT FOR FULL TIME RANGE                                          E249
C115  RMS = 0.0                                                         E249
```

```
C      IF(IFLAGS(11).EQ.2.AND.ICODE.EQ.2) GO TO 130                      E249
C      IF(ICODE.EQ.2) IST = 1                                            E249
C      IF(ICODE.EQ.1) IST = 2                                            E249
C      IF(ICODE.EQ.1) XARRY(1) = 0.0                                     E249
C      IF(ICODE.EQ.1) ARR1(1) = 0.0                                      E249
C      DO 120 I = IST,IBLSZ                                              E249
C      IF(ICODE.EQ.1.AND.I.GT.N) GO TO 125                               E249
C      IF(ICODE.EQ.2) RMS = RMS + ARR1(I)**2                             E249
C      IF(ICODE.EQ.1) XARRY(I) = (I - 1.) * FREQ(1)                      E249
C      IF(ICODE.EQ.2) XARRY(I) = (I-1.0) * (1.0 / (FREQ(1)*IBLSZ))       E249
C      IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.2.AND.ARR1(I).GT.TIMTEM)          E249
C     *                                                ARR1(I) = TIMTEM  E249
C      IF(ICODE.EQ.2.AND.IFLAGS(12).EQ.2.AND.ARR1(I).LT.-TIMTEM)         E249
C     *                                                ARR1(I) = -TIMTEM E249
C120   CONTINUE                                                          E249
C125   IF(ICODE.EQ.1) CALL CURVE(XARRY,ARR1,N,0)                         E249
C      IF(ICODE.EQ.2) CALL CURVE(XARRY,ARR1,IBLSZ,0)                     E249
C      IF(ICODE.EQ.1) CALL MESSAG(19H0 DB REF UNITY GAIN,19,.1,0.0)      E249
C      IF(ICODE.EQ.1) GO TO 150                                          E249
C      CALL HEIGHT(.07)                                                  E249
C      RMS = SQRT(RMS/IBLSZ)                                             E249
C      CALL REALNO(RMS,3,5.07,.35)                                       E249
C      CALL MESSAG(5HK RMS,5,5.77,.35)                                   E249
C      CALL REALNO(TMEAN,3,5.07,.2)                                      E249
C      CALL MESSAG(6HK MEAN,6,5.77,.2)                                   E249
C      CALL RESET('HEIGHT')                                              E249
C      GO TO 150                                                         E249
C_______________________________________________________________________ E249
C---- PARTIAL TIME RANGE                                                 E249
 130   IPEN = 3                                                          E249
       DO 140 I = ICH1,ICH2                                              E249
       RMS = RMS + ARR1(I)**2                                            E249
       IF(IFLAGS(12).EQ.2.AND.ARR1(I).GT.TIMTEM) ARR1(I) = TIMTEM        E249
       IF(IFLAGS(12).EQ.2.AND.ARR1(I).LT.-TIMTEM) ARR1(I) = -TIMTEM      E249
       XVAL = (I-ICH1) * (1.0 / (FREQ(1)*IBLSZ)) / XINC1                 E249
       YVAL = ARR1(I) / YINC1                                            E249
       CALL PLOT(XVAL,YVAL,IPEN)                                         E249
       IPEN = 2                                                          E249
 140   CONTINUE                                                          E249
       RMS = SQRT(RMS / (ICH2-ICH1+1))                                   E249
       CALL NUMBER(5.07,-1.65,.07,RMS,0.,3)                              E249
       CALL SYMBOL(5.77,-1.65,.07,5HK RMS,0.,5)                          E249
       CALL NUMBER(5.07,-1.8,.07,TMEAN,0.,3)                             E249
       CALL SYMBOL(5.77,-1.8,.07,6HK MEAN,0.,6)                          E249
C_______________________________________________________________________ E249
C      DISSPLA CALLS                                                     E249
C----------------------------------------------------------------------- E249
C---- PARTIAL TIME RANGE                                                 E249
C130   DO 140 I = ICH1,ICH2                                              E249
C      RMS = RMS + ARR1(I)**2                                            E249
C      XARRY(I) = (I-1.0) * (1.0 / (FREQ(1)*IBLSZ))                      E249
C      IF(IFLAGS(12).EQ.2.AND.ARR1(I).GT.TIMTEM) ARR1(I) = TIMTEM        E249
C      IF(IFLAGS(12).EQ.2.AND.ARR1(I).LT.-TIMTEM) ARR1(I) = -TIMTEM      E249
C140   CONTINUE                                                          E249
```

```
C      M = ICH2 - ICH1 + 1                                                   E249
C      CALL CURVE(XARRY,ARR1,M,0)                                            E249
C      CALL HEIGHT(.07)                                                      E249
C      RMS = SQRT(RMS / M)                                                   E249
C      CALL REALNO(RMS,3,5.07,.35)                                           E249
C      CALL MESSAG(5HK RMS,5,5.77,.35)                                       E249
C      CALL REALNO(TMEAN,3,5.07,.2)                                          E249
C      CALL MESSAG(6HK MEAN,6,5.77,.2)                                       E249
C      CALL RESET('HEIGHT')                                                  E249
C_____________________________________________________________________________E249
C                                                                            E249
C*** HEADERS FOR THE PLOTS                                                   E249
 150   IF(ICODE.EQ.1) CALL PLOT(0.,-9.1,-3)                                  E249
       IF(ICODE.EQ.2) CALL PLOT(0.,-7.4,-3)                                  E249
       IF(ICODE.EQ.1) CALL SYMBOL(0.8,10.4,.1,21HCOMPENSATION SPECTRUM,      E249
     * 0.,21)                                                                E249
       IF(ICODE.EQ.2) CALL SYMBOL(1.2,10.4,.1,18HINSTANTANEOUS DATA,0.,      E249
     *18)                                                                    E249
C---- TC PLOT IS FOR                                                         E249
       IF(IFLAGS(3).EQ.1) CALL SYMBOL(3.3,10.4,.07,9HSMALL T/C,0.,9)         E249
       IF(IFLAGS(3).EQ.2) CALL SYMBOL(3.3,10.4,.07,9HLARGE T/C,0.,9)         E249
C---- MATERIAL                                                               E249
       IF(IFLAGS(2).EQ.1) CALL SYMBOL(4.5,10.4,.07,10HPT / 6% RH,0.,10)      E249
       IF(IFLAGS(2).EQ.2) CALL SYMBOL(4.5,10.4,.07,11HPT / 30% RH,0.,11)     E249
       IF(IFLAGS(2).EQ.3) CALL SYMBOL(4.5,10.4,.07,7HCR / AL,0.,7)           E249
C_____________________________________________________________________________E249
C      DISSPLA CALLS                                                         E249
C-----------------------------------------------------------------------------E249
C*** HEADERS FOR THE PLOTS                                                   E249
C150   CALL ENDGR(0)                                                         E249
C      IF(ICODE.EQ.1) CALL OREL(0.,-4.9)                                     E249
C      IF(ICODE.EQ.2) CALL OREL(0.,-5.4)                                     E249
C      CALL AREA2D(7.,10.5)                                                  E249
C      CALL HEIGHT(.1)                                                       E249
C      IF(ICODE.EQ.1) CALL MESSAG(21HCOMPENSATION SPECTRUM,21,1.2,10.4)      E249
C      IF(ICODE.EQ.2) CALL MESSAG(18HINSTANTANEOUS DATA,18,1.2,10.4)         E249
C      CALL RESET('HEIGHT')                                                  E249
C---- TC PLOT IS FOR                                                         E249
C      CALL HEIGHT(.07)                                                      E249
C      IF(IFLAGS(3).EQ.1) CALL MESSAG(9HSMALL T/C,9,3.3,10.4)                E249
C      IF(IFLAGS(3).EQ.2) CALL MESSAG(9HLARGE T/C,9,3.3,10.4)                E249
C---- MATERIAL                                                               E249
C      IF(IFLAGS(2).EQ.1) CALL MESSAG(10HPT / 6% RH,10,4.5,10.4)             E249
C      IF(IFLAGS(2).EQ.2) CALL MESSAG(11HPT / 30% RH,11,4.5,10.4)            E249
C      IF(IFLAGS(2).EQ.3) CALL MESSAG(7HCR / AL,7,4.5,10.4)                  E249
C_____________________________________________________________________________E249
C---- GAMMA VALUE FOR COMPENSATION SPECTRUM                                  E249
       IF(ICODE.EQ.2) GO TO 160                                              E249
       CALL SYMBOL(0.0,10.25,.07,8HGAMMA = ,0.,8)                            E249
       GMAMET = GS(10) * .168279                                             E249
       CALL NUMBER(0.5,10.25,.07,GMAMET,0.,9)                                E249
C---- TC DIMENSIONS FOR COMPENSATION SPECTRUM                                E249
       CALL SYMBOL(2.0,10.25,.07,3HL1 ,0.,3)                                 E249
       CALL SYMBOL(3.1,10.25,.07,3HL2 ,0.,3)                                 E249
```

```
      CALL SYMBOL(4.2,10.25,.07,3HD1 ,0.,3)                            E249
      CALL SYMBOL(5.3,10.25,.07,3HD2 ,0.,3)                            E249
      IF(IFLAGS(3).EQ.1) ICOL = 2                                      E249
      IF(IFLAGS(3).EQ.2) ICOL = 1                                      E249
      SULN = TCDATA(1,ICOL) / .032808 + .0000001                       E249
      SMLN = TCDATA(2,ICOL) / .032808 + .0000001                       E249
      SUDI = TCDATA(3,ICOL) / .032808 + .0000001                       E249
      SMDI = TCDATA(4,ICOL) / .032808 + .0000001                       E249
      CALL NUMBER(2.21,10.25,.07,SULN,0.,7)                            E249
      CALL NUMBER(3.31,10.25,.07,SMLN,0.,7)                            E249
      CALL NUMBER(4.41,10.25,.07,SUDI,0.,7)                            E249
      CALL NUMBER(5.51,10.25,.07,SMDI,0.,7)                            E249
      GO TO 170                                                        E249
C--- COMPENSATED?                                                       E249
 160  IF(IFLAGS(9).EQ.2) CALL SYMBOL(0.,10.25,.07,19HUN-COMPENSATED DATAE249
     *,0.,19)                                                          E249
      IF(IFLAGS(9).EQ.1) CALL SYMBOL(0.,10.25,.07,16HCOMPENSATED DATA,0.E249
     *,16)                                                             E249
C--- INSTANTANEOUS DATA                                                 E249
      CALL SYMBOL(3.5,10.25,.07,13HRECORD NUMBER,0.,13)                E249
      RNUM = NUM + .05                                                 E249
      CALL NUMBER(4.5,10.25,.07,RNUM,0.,0)                             E249
 170  CALL SYMBOL(0.0,9.95,.07,PLOTIT(1,1),0.,80)                      E249
      CALL SYMBOL(0.0,9.75,.07,PLOTIT(1,2),0.,80)                      E249
      CALL SYMBOL(0.0,9.55,.07,PLOTIT(1,3),0.,80)                      E249
      CALL PLOT(0.,0.,-999)                                            E249
C______________________________________________________________________E249
C     DISSPLA CALLS                                                    E249
C--------------------------------------------------------------------- E249
C--- GAMMA VALUE FOR COMPENSATION SPECTRUM                              E249
C     IF(ICODE.EQ.2) GO TO 160                                         E249
C     CALL MESSAG(8HGAMMA = ,8,0.0,10.25)                              E249
C     GMAMET = GS(10) * .168279                                        E249
C     CALL REALNO(GMAMET,-5,'ABUT','ABUT')                             E249
C--- TC DIMENSIONS FOR COMPENSATION SPECTRUM                            E249
C     CALL MESSAG(3HL1 ,3,2.0,10.25)                                   E249
C     CALL MESSAG(3HL2 ,3,3.1,10.25)                                   E249
C     CALL MESSAG(3HD1 ,3,4.2,10.25)                                   E249
C     CALL MESSAG(3HD2 ,3,5.3,10.25)                                   E249
C     IF(IFLAGS(3).EQ.1) ICOL = 2                                      E249
C     IF(IFLAGS(3).EQ.2) ICOL = 1                                      E249
C     SULN = TCDATA(1,ICOL) / .032808 + .0000001                       E249
C     SMLN = TCDATA(2,ICOL) / .032808 + .0000001                       E249
C     SUDI = TCDATA(3,ICOL) / .032808 + .0000001                       E249
C     SMDI = TCDATA(4,ICOL) / .032808 + .0000001                       E249
C     CALL REALNO(SULN,7,2.21,10.25)                                   E249
C     CALL REALNO(SMLN,7,3.31,10.25)                                   E249
C     CALL REALNO(SUDI,7,4.41,10.25)                                   E249
C     CALL REALNO(SMDI,7,5.51,10.25)                                   E249
C     GO TO 170                                                        E249
C--- COMPENSATED?                                                       E249
C160  IF(IFLAGS(9).EQ.2) CALL MESSAG(19HUN-COMPENSATED DATA,19,0.,10.25)E249
C     IF(IFLAGS(9).EQ.1) CALL MESSAG(16HCOMPENSATED DATA,16,0.,10.25)  E249
C--- INSTANTANEOUS DATA                                                 E249
```

```
C      CALL MESSAG(13HRECORD NUMBER,13,3.5,10.25)                      E249
C      CALL INTNO(NUM,4.5,10.25)                                       E249
C170   CALL MESSAG(PLOTIT(1,1),80,0.0,9.95)                            E249
C      CALL MESSAG(PLOTIT(1,2),80,0.0,9.75)                            E249
C      CALL MESSAG(PLOTIT(1,3),80,0.0,9.55)                            E249
C      CALL RESET('HEIGHT')                                            E249
C      CALL RESET('CROSS')                                             E249
C      CALL ENDPL(0)                                                   E249
C______________________________________________________________________E249
       IF(ICODE.EQ.2) GO TO 190                                        E249
       DO 180 I = 1,ISIZ                                               E249
       ARR1(I) = 10.0 ** (ARR1(I)/20.)                                 E249
 180   CONTINUE                                                        E249
 190   RETURN                                                          E249
       END                                                             E249
C                                                                      E249
C                                                                      E249
       SUBROUTINE POWER(DATA3,DATA10,DATADC,XX,YY,XY,WINDO)            E249
C                                                                      E249
C**********************************************************************E249
C  POWER EVALUATES THE AUTO POWER SPECTRUMS OF THE 3MIL AND 10MIL DATA *E249
C      AND THE CROSS POWER SPECTRUM OF THE 3MIL VS 10MIL              *E249
C**********************************************************************E249
C                                                                      E249
C ---- IDENTIFICATION ----                                             E249
C                                                                      E249
C                                                                      E249
C** DATA3 - ARRAY FOR THE 3MIL TEST DATA READ IN INTEST                E249
C      -- OUTPUT                                                       E249
C                                                                      E249
C** DATA10 - ARRAY FOR THE 10MIL TEST DATA READ IN INTEST              E249
C      -- OUTPUT                                                       E249
C                                                                      E249
C** DATADC - ARRAY FOR THE DC CHANNEL TEST DATA READ IN INTEST         E249
C      -- OUTPUT                                                       E249
C                                                                      E249
C** XX - ARRAY CONTAINING THE INPUT AUTO POWER SPECTRUM G 3-3          E249
C      -- OUTPUT                                                       E249
C                                                                      E249
C** YY - ARRAY CONTAINING THE OUTPUT AUTO POWER SPECTRUM G 10-10       E249
C      -- OUTPUT                                                       E249
C                                                                      E249
C** XY - ARRAY CONTAINING THE CROSS POWER SPECTRUM G 3-10              E249
C      -- OUTPUT                                                       E249
C                                                                      E249
C** WINDO - ARRAY CONTAINING THE P301 WINDOW                           E249
C      -- OUTPUT                                                       E249
C                                                                      E249
C      CALLED BY MAIN PROGRAM                                          E249
C      CALLS - INTEST:  INPUTS THE DIGITIZED TEST DATA                 E249
C              SCALER:   SCALES THE DATA TO K                          E249
C              WINDOW:  APPLYS THE P301 TO INPUT DATA                  E249
C              FFT:     ROUTINE TO PERFORM FOURIER TRANSFORM           E249
C                       OF A COMPLEX VALUED SEQUENCE                   E249
```

```
C                                                                        E249
C     FILES USED:                                                        E249
C                   15 - WRITES OUT THE FOURIES TRANSFORM OF THE 3 MIL   E249
C                        THERMOCOUPLE TO THIS FILE.                      E249
C                   16 - WRITES OUT THE FOURIES TRANSFORM OF THE 10 MIL  E249
C                        THERMOCOUPLE TO THIS FILE.                      E249
C                                                                        E249
C***********************************************************************E249
C                                                                        E249
      COMPLEX CDATA3(2048),CDAT10(2048)                                  E249
      DIMENSION XX(1025),YY(1025),XY(1025,2),WINDO(2048),DATA3(2048),    E249
     *          DATA10(2048),DATADC(2048),                               E249
     *          ST3(1025),CT3(1025),ST10(1025),CT10(1025)                E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),    E249
     *   IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),  E249
     *   PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                               E249
      COMMON /DATAS/ C(45),TCF(11,9)                                     E249
C                                                                        E249
C*** INITIALIZE DATA                                                     E249
      REWIND 13                                                          E249
      REWIND 14                                                          E249
      REWIND 15                                                          E249
      REWIND 16                                                          E249
      KOUNT = 0                                                          E249
      IFLAG = 0                                                          E249
      DO 10 J = 1,IBSZ                                                   E249
      XX(J) = 0.0                                                        E249
      YY(J) = 0.0                                                        E249
      XY(J,1) = 0.0                                                      E249
      XY(J,2) = 0.0                                                      E249
   10 CONTINUE                                                           E249
C                                                                        E249
C*** LOOP TO READ IN DIGITIZED TEST DATA (MAX OF 400 REC)                E249
      DO 50 I=1,400                                                      E249
      CALL INTEST(DATA3,DATA10,DATADC,IFLAG)                             E249
C                                                                        E249
C** END OF DATA?                                                         E249
      IF(IFLAG.EQ.1) GO TO 60                                            E249
      IF(I.LT.IAVDAT(1)) GO TO 50                                        E249
      KOUNT = KOUNT + 1                                                  E249
C                                                                        E249
C*** CONVERT DATA FROM MV TO MV-PK AND THEN TO DEG-F                     E249
      CALL SCALER(DATA3,DATA10,DATADC)                                   E249
C                                                                        E249
C*** APPLY P301 WINDOW TO TEST DATA                                      E249
      CALL WINDOW(WINDO,DATA3)                                           E249
      CALL WINDOW(WINDO,DATA10)                                          E249
C                                                                        E249
C*** PERFORM FAST FOURIER TRANSFORMS ON DATA                             E249
      DO 20 J = 1,IBLSZ                                                  E249
      CDATA3(J) = CMPLX(DATA3(J),0.0)                                    E249
      CDAT10(J) = CMPLX(DATA10(J),0.0)                                   E249
   20 CONTINUE                                                           E249
      CALL FFT(1,IBLSZ,CDATA3)                                           E249
```

```
      CALL FFT(1,IBLSZ,CDAT10)                                          E249
      DO 30 J = 1,IBSZ                                                  E249
      CDATA3(J) = CDATA3(J)                                             E249
      CDAT10(J) = CDAT10(J)                                             E249
      CT3(J) = REAL(CDATA3(J)) * 2.0                                    E249
      ST3(J) = AIMAG(CDATA3(J)) * 2.0                                   E249
      CT10(J) = REAL(CDAT10(J)) * 2.0                                   E249
      ST10(J) = AIMAG(CDAT10(J)) * 2.0                                  E249
   30 CONTINUE                                                          E249
C                                                                       E249
C*** KEEP RUNNING SUMS OF AUTO AND CROSS POWER SPECTRUMS                E249
      DO 40 J = 1,IBSZ                                                  E249
      XX(J) = XX(J) + CT3(J)**2 + ST3(J)**2                             E249
      YY(J) = YY(J) + CT10(J)**2 + ST10(J)**2                           E249
      XY(J,1) = XY(J,1) + CT3(J)*CT10(J) + ST3(J)*ST10(J)               E249
      XY(J,2) = XY(J,2) + ST3(J)*CT10(J) - CT3(J)*ST10(J)               E249
   40 CONTINUE                                                          E249
C                                                                       E249
C*** WRITE OUT FOURIER TRANSFORMS ON DISK FILES FOR TEMPORARY STORAGE   E249
      WRITE(15) (CT3(J),J=1,IBSZ),(ST3(J),J=1,IBSZ)                     E249
      WRITE(16) (CT10(J),J=1,IBSZ),(ST10(J),J=1,IBSZ)                   E249
      IF(KOUNT.EQ.IAVDAT(2)) GO TO 60                                   E249
   50 CONTINUE                                                          E249
C                                                                       E249
C*** AVERAGE OUT THE POWER SPECTRUMS                                    E249
   60 DO 70 I = 1,IBSZ                                                  E249
      XX(I) = XX(I) / KOUNT                                             E249
      YY(I) = YY(I) / KOUNT                                             E249
      XY(I,1) = XY(I,1) / KOUNT                                         E249
      XY(I,2) = XY(I,2) / KOUNT                                         E249
   70 CONTINUE                                                          E249
      RETURN                                                            E249
      END                                                               E249
C                                                                       E249
C                                                                       E249
      SUBROUTINE PRNTIN (IIN, IOUT)                                     E249
C********************************************************************** E249
C     THIS ROUTINE PRINTS THE CARD IMAGE OF INPUT DATA SETS           * E249
C********************************************************************** E249
C                                                                       E249
C --- IDENTIFICATION ---                                                E249
C                                                                       E249
C** IIN - THE INPUT FILE TO BE PRINTED                                  E249
C     -- INPUT                                                          E249
C                                                                       E249
C** IOUT - THE OUTPUT FILE TO WHICH TO WRITE THE DATA                   E249
C     -- INPUT                                                          E249
C                                                                       E249
C     CALLED BY INPUT SUBROUTINE                                        E249
C                                                                       E249
C     FILES USED:                                                       E249
C                  THE FILES DESCRIBED ABOVE, IN THIS CASE 5 AND 6      E249
C                                                                       E249
```

```
C*********************************************************************** E249
C                                                                        E249
      DIMENSION ARRAY(200)                                               E249
      DATA BLANK /4H    /                                                E249
C                                                                        E249
   10 KOUNT = 0                                                          E249
   50 DO 100 I=1,200                                                     E249
  100 ARRAY(I) = BLANK                                                   E249
      IFLAG = 0                                                          E249
      KOUNT = KOUNT + 1                                                  E249
      READ (IIN, 125 ,END=150) ARRAY                                     E249
      IFLAG = 1                                                          E249
  150 IF (KOUNT .GT. 1) GO TO 250                                        E249
      WRITE (IOUT, 175)                                                  E249
      WRITE (IOUT, 180)                                                  E249
      WRITE (IOUT, 200)                                                  E249
  250 IF (IFLAG .EQ. 0) GO TO 400                                        E249
      WRITE (IOUT, 300) ARRAY                                            E249
      IF (KOUNT .NE. 4) GO TO 50                                         E249
      WRITE (IOUT, 200)                                                  E249
      WRITE (IOUT, 350)                                                  E249
      GO TO 10                                                           E249
  400 WRITE (IOUT, 300) ARRAY                                            E249
      REWIND IIN                                                         E249
      WRITE (IOUT, 200)                                                  E249
      WRITE (IOUT, 350)                                                  E249
      RETURN                                                             E249
  125 FORMAT (  20A4 )                                                   E249
  175 FORMAT (14H1INPUT LISTING )                                        E249
  180 FORMAT (/ 35X, 11HCARD COLUMN )                                    E249
  200 FORMAT (/ 10X,40H1111111111222222222233333333334444444444,         E249
     * 31H5555555555666666666677777777778 / 1X, 10H1234567890 ,          E249
     * 50H12345678901234567890123456789012345678901234567890,            E249
     * 20H12345678901234567890 / )                                       E249
  300 FORMAT ( 1X, 20A4 )                                                E249
  350 FORMAT (35X,11HCARD COLUMN )                                       E249
      END                                                                E249
C                                                                        E249
C                                                                        E249
      SUBROUTINE PSDFN(COMP,WINDO,ALSS)                                  E249
C                                                                        E249
C*********************************************************************** E249
C  PSDFN CREATES THE POWER SPECTRAL DENSITY FUNCTION FOR PLOTTING      * E249
C*********************************************************************** E249
C                                                                        E249
C --- IDENTIFICATION ---                                                 E249
C                                                                        E249
C** COMP - THE COMPENSATION SPECTRUM                                     E249
C      -- INPUT                                                          E249
C                                                                        E249
C** WINDO - P301 WINDOW, NEEDED TO FIND AREA OF THE LINE SHAPE SQUARED   E249
C      -- INPUT                                                          E249
C                                                                        E249
C** ALSS - AREA LINE SHAPE SQUARED                                       E249
```

```
C      — OUPUT                                                         E249
C                                                                      E249
C      CALLED BY MAIN PROGRAM                                          E249
C      CALLS PLT1: THIS SUBPROGRAM PLOTS THE INPUT ARRAY               E249
C            GET   : ACCESSES THE FFT DATA AND FORMS THE APPROPRIATE   E249
C                    AUTO POWER SPECTRUM                               E249
C                                                                      E249
C      FILES USED:                                                     E249
C                   6 - WRITES OUT ALL FUNCTIONS EVALUATED IF IBUG2 = 1 E249
C                                                                      E249
C*********************************************************************** E249
C                                                                      E249
      DIMENSION WW(1025),COMP(1024,2),PSD(1025),WINDO(2048),TEMP(1025) E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),  E249
     •    IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),E249
     •    PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                           E249
C                                                                      E249
      ISIZ = IBSZ - 1                                                  E249
      KOUNT = 0                                                        E249
C                                                                      E249
C*** FIND THE AREA OF THE LINE SHAPE SQUARED (ALSO INITIALIZE TEMP)    E249
      ALSS = 0.0                                                       E249
      DO 10 I = 1,IBLSZ                                                E249
      ALSS = ALSS + WINDO(I)**2                                        E249
      IF(I.GT.IBSZ) GO TO 10                                           E249
      TEMP(I) = 0.0                                                    E249
      PSD(I) = 0.0                                                     E249
   10 CONTINUE                                                         E249
      ALSS = ALSS / IBLSZ                                              E249
      IF(IFLAGS(7).EQ.2) RETURN                                        E249
C                                                                      E249
C*** ACCESS THE FFT DATA AND FIND THE AUTO POWER SPECTRUM              E249
      REWIND 15                                                        E249
      REWIND 16                                                        E249
      ISTP = IAVDAT(2)                                                 E249
      DO 110 I = 1,ISTP                                                E249
      CALL GET(WW)                                                     E249
      NUM = IAVDAT(1) + (I-1)                                          E249
      IF(IFLAGS(8).EQ.1) GO TO 30                                      E249
C                                                                      E249
C*** INSTANTANEOUS PLOTS: CHECK TO SEE IF THIS RECORD IS DESIRED       E249
      DO 20 J = 1,10                                                   E249
      IF(NUM.EQ.NREC(J)) GO TO 40                                      E249
   20 CONTINUE                                                         E249
      GO TO 110                                                        E249
C                                                                      E249
C*** PSD IS TO BE AVERAGED                                             E249
   30 IF(NUM.LT.NRECS(1)) GO TO 110                                    E249
      KOUNT = KOUNT + 1                                                E249
      IF(KOUNT.GT.NRECS(2)) GO TO 112                                  E249
C                                                                      E249
C*** RECORD IS WANTED                                                  E249
   40 IF(IFLAGS(8).EQ.1) GO TO 90                                      E249
```

```
      IF(IFLAGS(9).EQ.2) GO TO 60                                       E249
C                                                                       E249
C* COMPENSATED                                                          E249
      DO 50 J = 2,ISIZ                                                  E249
      WW(J) =(WW(J)/(FREQ(1)*ALSS))/(COMP(J,1)**2 + COMP(J,2)**2)       E249
      IF(IFLAGS(6).EQ.1) TEMP(J) = WW(J)                                E249
      IF(IFLAGS(6).EQ.2) TEMP(J) = 10. * ALOG10(WW(J))                  E249
      IF(IFLAGS(6).EQ.3) TEMP(J) = SQRT(WW(J))                          E249
      IF(IFLAGS(6).EQ.4) TEMP(J) = SQRT(WW(J)*FREQ(1)*ALSS)             E249
   50 CONTINUE                                                          E249
      GO TO 80                                                          E249
C                                                                       E249
C* UNCOMPENSATED                                                        E249
   60 DO 70 J = 2,IBSZ                                                  E249
      WW(J) = WW(J) / (FREQ(1)*ALSS)                                    E249
      IF(IFLAGS(6).EQ.1) TEMP(J) = WW(J)                                E249
      IF(IFLAGS(6).EQ.2) TEMP(J) = 10. * ALOG10(WW(J))                  E249
      IF(IFLAGS(6).EQ.3) TEMP(J) = SQRT(WW(J))                          E249
      IF(IFLAGS(6).EQ.4) TEMP(J) = SQRT(WW(J)*FREQ(1)*ALSS)             E249
   70 CONTINUE                                                          E249
   80 IF(IFLAGS(7).EQ.1) CALL PLT1(TEMP,1,NUM,0.0)                      E249
      IF(IBUG2.EQ.1) WRITE(6,180) NUM                                   E249
      ISTOP = IBSZ/4                                                    E249
      DO 85 J = 1,ISTOP                                                 E249
      JJ = J + ISTOP                                                    E249
      JJJ = JJ + ISTOP                                                  E249
      JJJJ = JJJ + ISTOP                                                E249
      IF(IBUG2.EQ.1) WRITE(6,200) J,TEMP(J),                            E249
     *                      JJ,TEMP(JJ),JJJ,TEMP(JJJ),JJJJ,TEMP(JJJJ)   E249
   85 CONTINUE                                                          E249
      GO TO 110                                                         E249
C                                                                       E249
C* SUM IF AVERAGED                                                      E249
   90 DO 100 J = 2,IBSZ                                                 E249
      TEMP(J) = TEMP(J) + WW(J)                                         E249
  100 CONTINUE                                                          E249
  110 CONTINUE                                                          E249
C                                                                       E249
C*** LOOP IS FINISHED, IF DATA WAS INSTANTANEOUS YOU ARE DONE.          E249
      IF(IFLAGS(8).EQ.2) RETURN                                         E249
  112 DO 115 I = 2,IBSZ                                                 E249
      TEMP(I) = TEMP(I) / NRECS(2)                                      E249
  115 CONTINUE                                                          E249
      IF(IFLAGS(9).EQ.2) GO TO 140                                      E249
C                                                                       E249
C* COMPENSATED                                                          E249
      DO 130 J = 2,ISIZ                                                 E249
      TEMP(J) =(TEMP(J)/(FREQ(1)*ALSS))/(COMP(J,1)**2 + COMP(J,2)**2)   E249
      IF(IFLAGS(6).EQ.1) PSD(J) = TEMP(J)                               E249
      IF(IFLAGS(6).EQ.2) PSD(J) = 10. * ALOG10(TEMP(J))                 E249
      IF(IFLAGS(6).EQ.3) PSD(J) = SQRT(TEMP(J))                         E249
      IF(IFLAGS(6).EQ.4) PSD(J) = SQRT(TEMP(J)*FREQ(1)*ALSS)            E249
  130 CONTINUE                                                          E249
```

```
      GO TO 160                                                          E249
C                                                                        E249
C* UNCOMPENSATED                                                         E249
 140  DO 150 J = 2,IBSZ                                                  E249
      TEMP(J) = TEMP(J) / (FREQ(1)*ALSS)                                 E249
      IF(IFLAGS(6).EQ.1) PSD(J) = TEMP(J)                                E249
      IF(IFLAGS(6).EQ.2) PSD(J) = 10. * ALOG10(TEMP(J))                  E249
      IF(IFLAGS(6).EQ.3) PSD(J) = SQRT(TEMP(J))                          E249
      IF(IFLAGS(6).EQ.4) PSD(J) = SQRT(TEMP(J)*FREQ(1)*ALSS)             E249
 150  CONTINUE                                                           E249
 160  IF(IFLAGS(7).EQ.1) CALL PLT1(PSD,1,0,0.0)                          E249
      IF(IBUG2.EQ.1) WRITE(6,190) NRECS(2)                               E249
      ISTOP = IBSZ/4                                                     E249
      DO 175 J = 1,ISTOP                                                 E249
      JJ = J + ISTOP                                                     E249
      JJJ = JJ + ISTOP                                                   E249
      JJJJ = JJJ + ISTOP                                                 E249
      IF(IBUG2.EQ.1) WRITE(6,200) J,PSD(J),JJ,PSD(JJ),                   E249
     *                                 JJJ,PSD(JJJ),JJJJ,PSD(JJJJ)       E249
 175  CONTINUE                                                           E249
      RETURN                                                             E249
C                                                                        E249
 180  FORMAT('1','THE FREQUENCY DOMAIN DATA FOR RECORD NUMBER ',I3)      E249
 190  FORMAT('1','THE FREQUENCY DOMAIN DATA FOR USING ',I3,2X,'RECORDS IE249
     *N THE AVERAGING')                                                  E249
 200  FORMAT(' ',4(2X,'I = ',I4,3X,E13.7,6X))                            E249
      END                                                                E249
C                                                                        E249
C                                                                        E249
      SUBROUTINE SCALER(DATA3,DATA10,DATADC)                             E249
C                                                                        E249
C*********************************************************************** E249
C       SCALER CONVERTS THE DATA TO DEG FAHRENHEIT                     * E249
C*********************************************************************** E249
C                                                                        E249
C ---- IDENTIFICATION ----                                               E249
C                                                                        E249
C                                                                        E249
C** DATA3 - TEST DATA FOR 3MIL TC READ IN INTEST, INPUT TO SCALE, AND    E249
C           CONTIANING THE SCALED DATA AS OUTPUT FROM THE ROUTINE        E249
C     -- INPUT AND OUTPUT                                                E249
C                                                                        E249
C** DATA10 - TEST DATA FOR 10MIL TC READ IN INTEST, INPUT TO SCALE, AND  E249
C            CONTAINING THE SCALED DATA AS OUTPUT FROM THE ROUTINE       E249
C     -- INPUT AND OUTPUT                                                E249
C                                                                        E249
C** DATADC - TEST DATA FOR DC CHANNEL READ IN INTEST, INPUT TO SCALE,    E249
C            AND CONTAINING THE SCALED DATA AS OUTPUT FROM THE ROUTINE   E249
C     -- INPUT AND OUTPUT                                                E249
C                                                                        E249
C     CALLED BY POWER SUBPROGRAM                                         E249
C     CALLS - TCALC: ACTUALLY CALCULATES THE TEMPERATURE                 E249
C                                                                        E249
C     FILES USED:                                                        E249
```

```
C                  13 - WRITES OUT THE SCALED DIGITIZED 3 MIL DATA       E249
C                       TO THIS FILE.                                    E249
C                  14 - WRITES OUT THE SCALED DIGITIZED 10 MIL DATA      E249
C                       TO THIS FILE.                                    E249
C                                                                        E249
C ---- IDIOSYNCRACIES ----                                               E249
C                                                                        E249
C SCALER REMOVES AMPLIFIER DC OFFSET AND SCALES THE DATA PRIOR TO        E249
C LINEARIZATION.  TWO RECORDS ARE AC DATA (LARGE & SMALL WIRE T/C)       E249
C AND THE THIRD IS THE DC CHANNEL.  AFTER REMOVAL OF DC OFFSET AND       E249
C SCALING, THE PROGRAM ADDS THE DC TO THE AC, CONVERTS IT TO TEMP-       E249
C ERATURE AND THEN REMOVES THE DC, LEAVING PEAK TEMPERATURE.  THIS       E249
C IS DONE TO BOTH AC CHANNELS.                                           E249
C THE SCALED DATA IS WRITTEN ONTO DISK FILE 17 FOR FUTURE ACCESS.        E249
C*********************************************************************** E249
C                                                                        E249
      DIMENSION DATA3(2048),DATA10(2048),DATADC(2048)                    E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),    E249
     *   IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),  E249
     *   PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                               E249
      COMMON /DATAS/ C(45),TCF(11,9)                                     E249
C                                                                        E249
C*** SCALE THE DATA                                                      E249
      TMEAN = 0.0                                                        E249
      DO 10 I=1,IBLSZ                                                    E249
      D3M = (DATA3(I)-CHANL(3)) * CHANL(2) * 1000. / CHANL(1)            E249
      D10M = (DATA10(I)-CHANL(6)) * CHANL(5) * 1000. / CHANL(4)          E249
      DDC = (DATADC(I)-CHANL(9)) * CHANL(8) * 1000. / CHANL(7)           E249
C                                                                        E249
C***CALCULATE DC TEMPERATURE                                             E249
      T = DDC                                                            E249
      CALL TCALC(T)                                                      E249
      TDC = T                                                            E249
      TMEAN = TMEAN + TDC                                                E249
C                                                                        E249
C*** ADD THE DC TO THE AC'S                                              E249
      D3MDDC = D3M + DDC                                                 E249
      D10MDC = D10M + DDC                                                E249
C                                                                        E249
C*** CALCULATE TEMPERATURE                                               E249
      T = D3MDDC                                                         E249
      CALL TCALC(T)                                                      E249
      T3MDDC = T                                                         E249
      T = D10MDC                                                         E249
      CALL TCALC(T)                                                      E249
      T10MDC = T                                                         E249
C                                                                        E249
C*** REMOVE THE DC                                                       E249
      T3M=T3MDDC-TDC                                                     E249
      T10M=T10MDC-TDC                                                    E249
C                                                                        E249
C*** PEAK TEMPERATURE                                                    E249
      DATA3(I) = T3M * 5./9.                                             E249
      DATA10(I) = T10M * 5./9.                                           E249
```

```
      DATADC(I) = TDC                                                     E249
   10 CONTINUE                                                            E249
      TMEAN = TMEAN / IBLSZ                                               E249
      TMEAN = 5.0*(TMEAN+40.)/9.0 + 233.15                                E249
      WRITE(13) (DATA3(I),I=1,IBLSZ),TMEAN                                E249
      WRITE(14) (DATA10(I),I=1,IBLSZ),TMEAN                               E249
      RETURN                                                              E249
      END                                                                 E249
C                                                                         E249
C                                                                         E249
      SUBROUTINE SPCY(FRQ,A)                                              E249
C                                                                         E249
C************************************************************************E249
C "SPCY" IS A SUBPROGRAM THAT DETERMINES THE SAMPLING FREQUENCY         *E249
C        AS A FUNCTION OF THE INPUT FREQUENCY.                          *E249
C************************************************************************E249
C                                                                         E249
C --- IDENTIFICATION ---                                                  E249
C                                                                         E249
C                                                                         E249
C** FRQ - FREQUENCY FOUND IN TRANGS, NEEDED HERE FOR COMPUTATIONS         E249
C     -- INPUT                                                            E249
C                                                                         E249
C** A - ARRAY WHICH CONTAINS THE SAMPLING FREQUENCY                       E249
C     -- OUTPUT                                                           E249
C                                                                         E249
C     CALLED BY TRFP SUBPROGRAM                                           E249
C               TRFM SUBPROGRAM                                           E249
C                                                                         E249
C************************************************************************E249
C                                                                         E249
      DIMENSION A(2)                                                      E249
C                                                                         E249
      F = FRQ +.00001                                                     E249
      P=1./F                                                              E249
      T1=P/(.0005 * 4)                                                    E249
      I1 = T1                                                             E249
      T1 = I1                                                             E249
      A(1)=(T1+1.)*4.                                                     E249
      IF(A(1).LT.128.0) A(1) = 128.0                                      E249
      RETURN                                                              E249
      END                                                                 E249
C                                                                         E249
C                                                                         E249
      SUBROUTINE TCALC(T)                                                 E249
C                                                                         E249
C*********************************************************************** E249
C "TCALC" IS A SUBPROGRAM THAT CALCULATES TEMPERATURE FROM THE         * E249
C COEFFICIENTS IN THE ARRAY "TCF" AND SPECIFIED THE TC CODE FLAG       * E249
C*********************************************************************** E249
C                                                                         E249
C --- IDENTIFICATION ---                                                  E249
C                                                                         E249
C                                                                         E249
```

```
C** T - VARIABLE CONTAINING THE SCALED DATA TO BE INPUT TO TCALC.        E249
C        WILL CONTAIN THE TEMPERATURE DERIVED IN THIS ROUTINE.            E249
C      — INPUT AND OUTPUT                                                 E249
C                                                                         E249
C      CALLED BY SCALER SUBPROGRAM                                        E249
C                                                                         E249
C***********************************************************************E249
C                                                                         E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),    E249
     *   IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),  E249
     *   PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                                E249
      COMMON /DATAS/ C(45),TCF(11,9)                                      E249
C                                                                         E249
C*** DIFFERENT INCREMENTS FOR THE DIFFERENT MATERIAL CODES                E249
      IF(IFLAGS(2).EQ.3) GO TO 10                                         E249
      IF(IFLAGS(2).EQ.2.OR.IFLAGS(2).EQ.1) GO TO 20                       E249
   10 IRL=1                                                               E249
      IRH=IRL+2                                                           E249
      GO TO 30                                                            E249
   20 IRL=7                                                               E249
      IRH=IRL+3                                                           E249
   30 DO 40 IR=IRL,IRH                                                    E249
      IF(TCF(IR,1).GT.T) GO TO 50                                         E249
   40 CONTINUE                                                            E249
      IR = IRH                                                            E249
C                                                                         E249
C*** CALCULATE TEMPERATURE                                                E249
   50 XN = TCF(IR,2)*T + TCF(IR,3)                                        E249
      DF = TCF(IR,4) + TCF(IR,5)*XN + TCF(IR,6)*XN**2 + TCF(IR,7)*XN**3 E249
     *     + TCF(IR,8)*XN**4 + TCF(IR,9)*XN**5                            E249
      T = DF                                                              E249
      RETURN                                                              E249
      END                                                                 E249
C                                                                         E249
C                                                                         E249
      SUBROUTINE TCPARM(TC)                                               E249
C                                                                         E249
C***********************************************************************E249
C        TCPARM CALCULATES THE THERMOCOUPLE WIRE PARAMETERS AND         *E249
C                 PUTS THEM INTO THE ARRAY "TC"                         *E249
C***********************************************************************E249
C                                                                         E249
C —— IDENTIFICATION ——                                                    E249
C                                                                         E249
C                                                                         E249
C** TC IS THE ARRAY OF THERMOCOUPLE PARAMETERS CALCULATED, LISTED BELOW E249
C        — OUTPUT                                                         E249
C                                                                         E249
C 1. DENSITY (RHO)                                                        E249
C 2. THERMAL CONDUCTIVITY (XK)                                            E249
C 3. SPECIFIC HEAT (CP)                                                   E249
C 4. THERMAL DIFFUSITY (AL)                                               E249
C                                                                         E249
C      CALLED BY MAIN PROGRAM                                             E249
```

```
C                                                                        E249
C ---- IDIOSYNCRACIES ----                                               E249
C                                                                        E249
C THE VALUE OF "IFLAGS(2)" DETERMINES WHICH SET OF EQUATIONS WILL BE USEE249
C                                                                        E249
C 1. WHEN IFLAGS(2) IS 1, THE EQUATIONS FOR PT/6%RH TC'S WILL BE USED   E249
C                                                                        E249
C 2. WHEN IFLAGS(2) IS 2, THE EQUATIONS FOR PT/30%RH TC'S WILL BE USED  E249
C                                                                        E249
C 3. WHEN IFALGS(2) IS 3, THE EQUATIONS FOR CR/AL TC'S WILL BE USED     E249
C                                                                        E249
C***********************************************************************E249
C                                                                        E249
      DIMENSION TC(4)                                                    E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),    E249
     *   IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2), E249
     *   PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                               E249
      COMMON /DATAS/ C(45),TCF(11,9)                                     E249
C                                                                        E249
      T=GAS(2)                                                           E249
C                                                                        E249
C*** CHECKING ON T/C MATERIAL CODE                                       E249
      IF(IFLAGS(2).EQ.1 .OR. IFLAGS(2).EQ.2) GO TO 10                    E249
      IF(IFLAGS(2).EQ.3) GO TO 30                                        E249
C                                                                        E249
C*** EQUATIONS FOR MATERIAL PT / 6% RH  -OR-  PT /30% RH                 E249
   10 RHO1=1278.7                                                        E249
      XK1=C(1)+C(2)*T+C(3)*T**2+C(4)*T**3                                E249
      CP1=C(5)+C(6)*T+C(7)*T**2+C(8)*T**3                                E249
      AL1=C(9)+C(10)*T+C(11)*T**2+C(12)*T**3                             E249
C                                                                        E249
      RHO2=1092.1                                                        E249
      XK2=C(13)+C(14)*T+C(15)*T**2+C(16)*T**3                            E249
      CP2=C(17)+C(18)*T+C(19)*T**2+C(20)*T**3+C(21)*T**4                 E249
      AL2=C(22)+C(23)*T+C(24)*T**2+C(25)*T**3                            E249
C                                                                        E249
      RHO = (RHO1 + RHO2) / 2.0                                          E249
      XK = (XK1 + XK2) / 2.0                                             E249
      CP = (CP1 + CP2) / 2.0                                             E249
      AL = (AL1 + AL2) / 2.0                                             E249
      GO TO 40                                                           E249
C                                                                        E249
C*** EQUATIONS FOR MATERIAL CU / AL                                      E249
   30 RHO = 540.95                                                       E249
      XK = (0.01547 * T + 24.505) / 2.0                                  E249
      CP = (.0001129 * T + .21454) / 2.0                                 E249
      AL = XK / (RHO * CP * 3600.)                                       E249
C                                                                        E249
C*** EVALUATING THE PARAMETERS                                           E249
   40 TC(1)=RHO                                                          E249
      TC(2)=XK/3600.                                                     E249
      TC(3)=CP                                                           E249
      TC(4)=AL                                                           E249
      RETURN                                                             E249
```

```
      END                                                                E249
C                                                                        E249
C                                                                        E249
      SUBROUTINE TERM(I)                                                 E249
C                                                                        E249
C*********************************************************************** E249
C 'TERM' TERMINATES THE PROGRAM DUE TO AN ILLEGAL USER ENTRY, OR       * E249
C     BECAUSE A CALCULATED GAMMA WAS NOT ABLE TO BE FOUND              * E249
C*********************************************************************** E249
C                                                                        E249
C ---- IDENTIFICATION ----                                               E249
C                                                                        E249
C** I - CODE PASSED TO DETERMINE WHICH ERROR CAUSED TERMINATION OF       E249
C        THE PROGRAM                                                     E249
C                                                                        E249
C     CALLED FROM CHECK SUBPROGRAM                                       E249
C                 INTERP SUBPROGRAM                                      E249
C                                                                        E249
C     FILES USED:                                                        E249
C                  6 - WRITES OUT APPROPRIATE REASONS FOR ANY PREMATURE  E249
C                      TERMINATION OF THE PROGRAMS EXECUTION.            E249
C                                                                        E249
C*********************************************************************** E249
C                                                                        E249
      II = I - 15                                                        E249
      IF(I.LT.20) WRITE(6,210)                                           E249
      GO TO (10,20,30,40,50,60,70,80,90,100,110,120,130,140,150),I       E249
      GO TO (160,170,180,190,200),II                                     E249
   10 WRITE(6,220)                                                       E249
      WRITE(6,420)                                                       E249
      STOP                                                               E249
   20 WRITE(6,230)                                                       E249
      WRITE(6,420)                                                       E249
      STOP                                                               E249
   30 WRITE(6,240)                                                       E249
      WRITE(6,420)                                                       E249
      STOP                                                               E249
   40 WRITE(6,250)                                                       E249
      WRITE(6,420)                                                       E249
      STOP                                                               E249
   50 WRITE(6,260)                                                       E249
      WRITE(6,420)                                                       E249
      STOP                                                               E249
   60 WRITE(6,270)                                                       E249
      WRITE(6,420)                                                       E249
      STOP                                                               E249
   70 WRITE(6,280)                                                       E249
      WRITE(6,420)                                                       E249
      STOP                                                               E249
   80 WRITE(6,290)                                                       E249
      WRITE(6,420)                                                       E249
      STOP                                                               E249
   90 WRITE(6,300)                                                       E249
      WRITE(6,420)                                                       E249
```

```
      STOP                                                              E249
  100 WRITE(6,310)                                                      E249
      WRITE(6,420)                                                      E249
      STOP                                                              E249
  110 WRITE(6,320)                                                      E249
      WRITE(6,420)                                                      E249
      STOP                                                              E249
  120 WRITE(6,330)                                                      E249
      WRITE(6,420)                                                      E249
      STOP                                                              E249
  130 WRITE(6,340)                                                      E249
      WRITE(6,420)                                                      E249
      STOP                                                              E249
  140 WRITE(6,350)                                                      E249
      WRITE(6,420)                                                      E249
      STOP                                                              E249
  150 WRITE(6,360)                                                      E249
      WRITE(6,420)                                                      E249
      STOP                                                              E249
  160 WRITE(6,370)                                                      E249
      WRITE(6,420)                                                      E249
      STOP                                                              E249
  170 WRITE(6,380)                                                      E249
      WRITE(6,420)                                                      E249
      STOP                                                              E249
  180 WRITE(6,390)                                                      E249
      WRITE(6,420)                                                      E249
      STOP                                                              E249
  190 WRITE(6,400)                                                      E249
      WRITE(6,420)                                                      E249
      STOP                                                              E249
  200 WRITE(6,410)                                                      E249
      STOP                                                              E249
  210 FORMAT(' ','EXECUTION TERMINATED DUE TO ILLEGAL USER INPUT.')     E249
  220 FORMAT(' ','** INVALID SELECTION FOR IFLAGS(1) **')               E249
  230 FORMAT(' ','** INVALID SELECTION FOR IFLAGS(2) **')               E249
  240 FORMAT(' ','** INVALID SELECTION FOR IFLAGS(3) **')               E249
  250 FORMAT(' ','** INVALID SELECTION FOR IFLAGS(4) **')               E249
  260 FORMAT(' ','** INVALID SELECTION FOR IFLAGS(5) **')               E249
  270 FORMAT(' ','** INVALID SELECTION FOR IFLAGS(6) **')               E249
  280 FORMAT(' ','** INVALID SELECTION FOR IFLAGS(7) **')               E249
  290 FORMAT(' ','** INVALID SELECTION FOR IFLAGS(8) **')               E249
  300 FORMAT(' ','** INVALID SELECTION FOR IFLAGS(9) **')               E249
  310 FORMAT(' ','** INVALID SELECTION FOR IFLAGS(10) **')              E249
  320 FORMAT(' ','** INVALID SELECTION FOR IFLAGS(11) **')              E249
  330 FORMAT(' ','** INVALID SELECTION FOR IFLAGS(12) **')              E249
  340 FORMAT(' ','USER IMPLIED A PRE-DETERMINED VALUE OF GAMMA WAS TO BEE249
     * USED',/,' ','(IFLAGS(1)), AND NO VALUE WAS INPUT TO THE VARIABLE E249
     *"GAMMA"')                                                         E249
  350 FORMAT(' ','USER IMPLIED HE WANTED PLOTS OF THE INSTANTANEOUS COMPE249
     *ENSATED SPECTRA',/,' ','(IFLAGS(5)), AND NO RECORDS WERE INPUT TO E249
     *THE ARRAY "IREC"')                                                E249
  360 FORMAT(' ','USER IMPLIER HE WANTED A PARTIAL TIME RANGE TO BE PLOTE249
     *TED'/' ','(IFLAGS(11)), AND NO TIME VALUES WERE INPUT TO THE ARRAYE249
```

```
     *"TIME"')                                                          E249
  370 FORMAT(' ','FOR PARTIAL TIME RANGE, ENDING TIME MUST BE GREATER THE249
     *AN STARTING TIME')                                                E249
  380 FORMAT(' ','USER IMPLIED HE WANTED INSTANTANEOUS PLOTS OF THE FREQE249
     *UENCY DOMAIN DATA'/' ','(IFLAGS(8)), AND NO RECORDS WERE INPUT TO E249
     *THE ARRAY "NREC"')                                                E249
  390 FORMAT(' ','USER IMPLIED HE WANTED AVERAGED PLOTS OF THE FREQUENCYE249
     * DOMAIN DATA'/' ','(IFLAGS(8)), AND NO RECORDS WERE INPUT TO THE AE249
     *RRAY "NRECS"')                                                    E249
  400 FORMAT(' ','USER IMPLIED HE WANTED ALL INSTANTANEOUS PLOTS SCALED E249
     *TO THE SAME',/,' ',5X,'TEMPERATURE (IFLAGS(12)=2), BUT NO TEMPERA E249
     *TURE WAS ENTERED TO TIMTEM')                                      E249
  410 FORMAT(' ','EXECUTION TERMINATED - NO CALCULATED VALUE OF GAMMA WAE249
     *S FOUND')                                                         E249
  420 FORMAT(' ','PLEASE CHECK USER INPUTS AND TRY AGAIN')              E249
      END                                                               E249
C                                                                       E249
C                                                                       E249
      SUBROUTINE TRANGS(J,NGAM,GS,TC,TRAN,COMP)                         E249
C                                                                       E249
C***********************************************************************E249
C  TRANGS EVALUATES THE TRANSFER FUNCTIONS OF THE T/C -VS- GS FOR     *E249
C     NGAM VALUES OF GAMMA AND STORES THEM IN TRAN(NGAM,J,LOC,TYPE)   *E249
C***********************************************************************E249
C                                                                       E249
C --- IDENTIFICATION ---                                                E249
C                                                                       E249
C                                                                       E249
C** J - T/C FOR WHICH THE TRANSFER FUNCTION IS DESIRED: 1=LARGE, 2=SMALLE249
C      -- INPUT                                                         E249
C                                                                       E249
C** NGAM - NUMBER OF GAMMA VALUES FOR WHICH THE TRANSFER FUNCTION IS    E249
C          DERIVED. NOTE: NGAM SHOULD BE 17 IF USING THE THEORETICAL    E249
C          VALUE OF GAMMA, OR 1 IF USING THE MEASURED VALUE OF GAMMA.   E249
C      -- INPUT                                                         E249
C                                                                       E249
C** GS - ARRAY CONTAINING THE AERODYNAMIC PARAMETER, GAMMA.             E249
C      -- INPUT                                                         E249
C                                                                       E249
C** TC - ARRAY CONTAINING THE THERMOCOUPLE PARAMETERS REQUIRED IN TRFP  E249
C      -- INPUT                                                         E249
C                                                                       E249
C** TRAN - ARRAY INTO WHICH THE TRANSFER FUNCTION IS PUT                E249
C      -- OUTPUT                                                        E249
C                                                                       E249
C** COMP - ARRAY INTO WHICH THE COMPENSATION SPECTRUM WILL BE PLACED    E249
C      -- OUTPUT                                                        E249
C                                                                       E249
C     CALLED BY MAIN PROGRAN                                            E249
C     CALLS - TRFP:  THIS SUBPROGRAM CALCULATES THE COEFFICIENTS FOR THEE249
C                    TRANSFER PROGRAM "TRFM"                            E249
C             TRFM:  THIS SUBPROGRAM EVALUATES THE TRANSFER FUNCTION.   E249
C             PLT2: THIS SUBPROGRAM PLOTS THE INPUT ARRAYS              E249
C                                                                       E249
```

```
C     FILES USED:                                                       E249
C                  6 - WRITES THE COMPENSATION SPECTRUM TO THIS FILE    E249
C                      IF THE USER SO DISIRES (IBUG2 = 1)               E249
C                 12 - WRITES THE COMPENSATION SPECTRUM TO THIS FILE    E249
C                                                                       E249
C***********************************************************************E249
C                                                                       E249
      DIMENSION TRAN(17,3,1024,2),A(2),GS(10),COMP(1024,2),             E249
     * REA(1024),RIMA(1024),TP(10),TC(4)                                E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),   E249
     *    IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),E249
     *    PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                             E249
C                                                                       E249
      A(1) = GS(10)                                                     E249
C                                                                       E249
C*** SETTING A TEMPORARY VARIABLE FOR THE APPROPRIATE DELTA-F           E249
      T=FREQ(1)+.00001                                                  E249
C                                                                       E249
C*** SETTING THE START CHANNEL, END CHANNEL, AND STEPPING INCREMENT FOR E249
C*** THE PIECEWISE TRANSFORM IN THE COMPENSATION SPECTRUM               E249
      IF(NGAM.EQ.17) GO TO 10                                           E249
      ISTCH = 2                                                         E249
      IEDCH = IBSZ - 1                                                  E249
      ISTEP = 1                                                         E249
      GO TO 20                                                          E249
C                                                                       E249
C*** SETTING THE START CHANNEL, END CHANNEL, AND STEPPING INCREMENT FOR E249
C*** THE USER SPECIFIED FREQUENCIES                                     E249
   10 CHN = FREQ(2) / FREQ(1)                                           E249
      ISTCH = CHN +1.49                                                 E249
      CHN = FREQ(3) / FREQ(1)                                           E249
      IEDCH = CHN +1.49                                                 E249
      CHN = FREQ(4) / FREQ(1)                                           E249
      ISTEP = CHN +.05                                                  E249
C                                                                       E249
   20 DELGMA=.2                                                         E249
      IC1=ISTCH                                                         E249
C                                                                       E249
C*** LOOP FOR EACH 'TEST' GAMMA USED FOR AN ESTIMATED TRANSFER FUNCTION E249
      DO 110 IG=1,NGAM                                                  E249
      OLDPHS=0.                                                         E249
      IF(NGAM.EQ.1) GO TO 30                                            E249
      TGMA=GS(10)*DELGMA                                                E249
      GO TO 40                                                          E249
   30 TGMA=GS(10)                                                       E249
C                                                                       E249
C*** LOOPING THROUGH THE CORRECT FREQUENCIES                            E249
   40 DO 100 IC=ISTCH,IEDCH,ISTEP                                       E249
      IF(NGAM.EQ.17) GO TO 60                                           E249
C                                                                       E249
C** CHECKING ON CHANNELS FOR PIECEWISE TRANSFORM                        E249
      IF(IC.GE.IEDCH) GO TO 60                                          E249
      IF(IC.GE.51) GO TO 50                                             E249
      IC1=IC1+1                                                         E249
```

```
      GO TO 60                                                          E249
   50 IF(IC.LT.IC1) GO TO 100                                           E249
      IC1=IC1+10                                                        E249
C                                                                       E249
C*** CALLING ROUTINES TO EVALUATE THE TRANSFER FUNCTION                  E249
   60 A(2)=TGMA                                                         E249
      FRQ=T*(IC - 1)                                                    E249
      CALL TRFP(J,TC,FRQ,A,TP)                                          E249
      CALL TRFM(TP,FRQ,A)                                               E249
      IF(OLDPHS.GT.A(2)) GO TO 70                                       E249
      A(2)=OLDPHS                                                       E249
      GO TO 80                                                          E249
   70 OLDPHS=A(2)                                                       E249
C                                                                       E249
C*** STORING GAIN AND PHASE AS A TRANSFER FUNCTION OR AS COMPENSATION   E249
C*** SPECTRUM DEPENDING ON GAMMA (TEST GAMMAS OR MEASURED GAMMA)        E249
   80 IF(NGAM.EQ.1) GO TO 90                                            E249
      TRAN(IG,J,IC,1) = A(1)                                            E249
      TRAN(IG,J,IC,2) = A(2)                                            E249
      GO TO 100                                                         E249
   90 COMP(IC,1) = A(1)                                                 E249
      COMP(IC,2) = A(2)                                                 E249
  100 CONTINUE                                                          E249
C-- NEXT FREQUENCY                                                      E249
      DELGMA = DELGMA +.1                                               E249
  110 CONTINUE                                                          E249
C-- NEXT GAMMA                                                          E249
      IF(NGAM.EQ.17) RETURN                                             E249
C                                                                       E249
C*** FILL IN THE COMPENSATION SPECTRUM LINEARLY BETWEEN EVALUATED POINTSE249
      INT = IEDCH / 10                                                  E249
      ISTP = INT*10 - 9                                                 E249
      DO 130 I = 51,ISTP,10                                             E249
      IX1 = I + 1                                                       E249
      IX2 = I + 9                                                       E249
      DO 120 JJ = IX1,IX2                                               E249
      COMP(JJ,1) = (COMP(I+10,1) - COMP(I,1)) * (JJ-I) / 10.+ COMP(I,1) E249
      COMP(JJ,2) = (COMP(I+10,2) - COMP(I,2)) * (JJ-I) / 10.+ COMP(I,2) E249
  120 CONTINUE                                                          E249
  130 CONTINUE                                                          E249
      ISZ = IBSZ - 1                                                    E249
      ISTP = ISTP + 10                                                  E249
      I1 = ISTP + 1                                                     E249
      I2 = IBSZ - 2                                                     E249
      DO 140 JJ = I1,I2                                                 E249
      COMP(JJ,1) = (COMP(ISZ,1)-COMP(ISTP,1))*(JJ-ISTP) / (ISZ-ISTP)    E249
     *                                       + COMP(ISTP,1)             E249
      COMP(JJ,2) = (COMP(ISZ,2)-COMP(ISTP,2))*(JJ-ISTP) / (ISZ-ISTP)    E249
     *                                       + COMP(ISTP,2)             E249
  140 CONTINUE                                                          E249
C                                                                       E249
C*** STORE VARIOUS DATA ON DISK FILES FOR FUTURE ACCESS                 E249
      REWIND 12                                                         E249
      GMAMET = GS(10) * .168279                                         E249
```

```
      WRITE(12) (COMP(JJ,1),JJ=1,ISZ),(COMP(JJ,2),JJ=1,ISZ),             E249
     *          IFLAGS(2),IFLAGS(3),GMAMET,((TCDATA(I,JJ),I=1,4),JJ=1,2)E249
C                                                                        E249
C*** PLUG COMPENSATION SPECTRUM INTO ARRAYS FOR PLOTTING                 E249
      ISIZ = IBSZ - 1                                                    E249
      DO 150 I = 1,ISIZ                                                  E249
      REA(I) = COMP(I,1)                                                 E249
      RIMA(I) = COMP(I,2)                                                E249
  150 CONTINUE                                                           E249
      IF(IFLAGS(4).EQ.1) CALL PLT2(REA,RIMA,1,GS,0,0.0)                  E249
      IF(IBUG2.EQ.0) GO TO 170                                           E249
      WRITE(6,190)                                                       E249
      ISTOP = ISZ/2                                                      E249
      DO 160 I = 1,ISTOP                                                 E249
      II = I + ISTOP                                                     E249
      WRITE(6,200) I,REA(I),RIMA(I),II,REA(II),RIMA(II)                  E249
  160 CONTINUE                                                           E249
C------------ CHANGE TO RECTANGULAR COORDINATES                          E249
  170 DO 180 I = 1,IBSZ                                                  E249
      GAIN = COMP(I,1)                                                   E249
      COMP(I,1) = GAIN * COS(COMP(I,2) * 3.1415927 / 180.)               E249
      COMP(I,2) = GAIN * SIN(COMP(I,2) * 3.1415927 / 180.)               E249
  180 CONTINUE                                                           E249
      RETURN                                                             E249
  190 FORMAT('1','THE COMPENSATION SPECTRUM IN POLAR GAIN AND PHASE')    E249
  200 FORMAT(' ',2(2X,'I = ',I4,3X,E13.7,3X,E13.7,6X))                   E249
      END                                                                E249
C                                                                        E249
C                                                                        E249
      SUBROUTINE TRANTC(NGAM,TRAN)                                       E249
C                                                                        E249
C***********************************************************************E249
C        TRANTC EVALUATES THE TRANSFER FUNCTION BETWEEN                *E249
C                 THE LARGE AND SMALL T/C                              *E249
C***********************************************************************E249
C                                                                        E249
C --- IDENTIFICATION ---                                                 E249
C                                                                        E249
C                                                                        E249
C** NGAM - NUMBER OF GAMMA VALUES TRANSFER FUNCTION TO BE EVALUATED FOR E249
C     -- INPUT                                                           E249
C                                                                        E249
C** TRAN - ARRAY CONTAINING THE TRANSFER FUNCTIONS OF THE LARGE AND      E249
C          SMALL T/C -VS- GAS STREAM (FOR INPUT), ALONG WITH THE         E249
C          OUTPUT OF THIS SUBROUTINE                                     E249
C     -- INPUT AND OUTPUT                                                E249
C                                                                        E249
C     CALLED BY MAIN PROGRAM                                             E249
C                                                                        E249
C********************************************************************** E249
C                                                                        E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),    E249
     *    IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),E249
     *    PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                              E249
```

```
      DIMENSION TRAN(17,3,1024,2)                                       E249
C                                                                       E249
C*** START AND STOP CHANNELS AND THE STEPPING INCREMENT                  E249
      IST=FREQ(2)/FREQ(1) + 1.49                                        E249
      IED=FREQ(3)/FREQ(1) + 1.49                                        E249
      INC=FREQ(4)/FREQ(1) + .05                                         E249
C                                                                       E249
C*** LOOPING THROUGH THE 'TEST' GAMMAS AND THE CHANNELS                  E249
      DO 40 IG = 1,NGAM                                                 E249
      DO 30 IC=IST,IED,INC                                              E249
      G3 = TRAN(IG,2,IC,1)                                              E249
      P3 = TRAN(IG,2,IC,2)                                              E249
      G10 = TRAN(IG,1,IC,1)                                             E249
      P10 = TRAN(IG,1,IC,2)                                             E249
C                                                                       E249
C*** EVALUATING 10MIL -VS- 3MIL TRANSFER FUNCTION                        E249
      TF10T3=G10/G3                                                     E249
      PH10T3=P10-P3                                                     E249
C                                                                       E249
C*** CHECKING AND CORRECTING FOR QUADRANT OF PHASE                       E249
      T=ABS(PH10T3)                                                     E249
      IF(180.0.GE.T) GO TO 20                                           E249
      IF(PH10T3.GE.0.0) GO TO 10                                        E249
      PH10T3=PH10T3+360.                                                E249
      GO TO 20                                                          E249
   10 PH10T3=PH10T3-360.                                                E249
C                                                                       E249
   20 TRAN(IG,3,IC,1) = TF10T3                                          E249
      TRAN(IG,3,IC,2) = PH10T3                                          E249
   30 CONTINUE                                                          E249
   40 CONTINUE                                                          E249
C                                                                       E249
      RETURN                                                            E249
      END                                                               E249
C                                                                       E249
C                                                                       E249
      SUBROUTINE TRFM(TP,FRQ,A)                                         E249
C                                                                       E249
C***********************************************************************E249
C "TRFM" IS A SUBPROGRAM THAT EVALUATES THE TRANSFER FUNCTION          *E249
C      BETWEEN THE THERMOCOUPLE WIRE AND THE GAS STREAM.               *E249
C***********************************************************************E249
C                                                                       E249
C —— IDENTIFICATION ——                                                  E249
C                                                                       E249
C                                                                       E249
C** TP - PARAMETERS FOUND BY TRFP NEEDED TO EVALUATE TRANSFER FUNCTION  E249
C      — INPUT                                                          E249
C                                                                       E249
C** FRQ - FREQUENCY FOUND IN TRANGS, NEEDED IN SPCY                     E249
C      — INPUT                                                          E249
C                                                                       E249
C** A - ARRAY INTO WHICH THE GAIN AND PHASE ARE STORED                  E249
C      — OUTPUT                                                         E249
```

```
C                                                                        E249
C     CALLED BY TRANGS SUBPROGRAM                                        E249
C     CALLS - SPCY:  DETERMINES THE SAMPLING FREQUENCY AS A FUNCTION     E249
C                    OF THE ANALYSIS FREQUENCY                           E249
C                                                                        E249
C***********************************************************************E249
C                                                                        E249
      DIMENSION Z(10),ZP(10),TP(10),A(2)                                 E249
C                                                                        E249
C*** INITIALIZING VARIABLES                                              E249
      CALL SPCY(FRQ,A)                                                   E249
      XN2=A(1)                                                           E249
      N2=XN2                                                             E249
      LAP=0                                                              E249
      P1=0.                                                              E249
      P2=0.                                                              E249
      P3=0.                                                              E249
      ZC1=0.                                                             E249
      ZC2=0.                                                             E249
      PKPOS=0.                                                           E249
      PKNEG=0.                                                           E249
      DO 10 I=1,10                                                       E249
      Z(I)=0.                                                            E249
      ZP(I)=0.                                                           E249
   10 CONTINUE                                                           E249
      DELTAT=TP(2)                                                       E249
      T=0.0                                                              E249
      CN=TP(4)                                                           E249
      A1=TP(5)                                                           E249
      B=TP(6)                                                            E249
      C=TP(7)                                                            E249
      E=TP(8)                                                            E249
      F=TP(9)                                                            E249
      G=TP(10)                                                           E249
C                                                                        E249
C*** FINITE DIFFERENCE METHOD FOR TRANSFER FUNCTION (UNTILL CONVERGENCE)E249
      DO 70 I=1,32000                                                    E249
      T1=Z(1)                                                            E249
      T2=Z(2)                                                            E249
      T3=Z(3)                                                            E249
      T4=Z(4)                                                            E249
      T5=Z(5)                                                            E249
      T6=Z(6)                                                            E249
      T7=Z(7)                                                            E249
      T8=Z(8)                                                            E249
      T9=Z(9)                                                            E249
      T0=Z(10)                                                           E249
      TR=SIN(CN*T)                                                       E249
      ZP(1)=(.5)*(C*(T0+T2-2*T1))/(A1)+T1                                E249
      ZP(2)=(.5)*(C*(T1+T3-2*T2))/(A1)+T2                                E249
      ZP(3)=(.5)*(C*(T2+T4-2*T3)+F*(TR-T3))/(A1)+T3                      E249
      ZP(4)=(.5)*(C*(T3+T5-2*T4)+(2*F)*(TR-T4))/(A1)+T4                  E249
      ZP(5)=(.5)*(C*(T4+T6-2*T5)+(2*F)*(TR-T5))/(A1)+T5                  E249
      ZP(6)=(C*(T5-T6)+E*(T7-T6)+(F+G)*(TR-T6))/(A1+B)+T6                E249
```

```
      ZP(7)=(.5)*(E*(T6+T8-2*T7)+(2*G)*(TR-T7))/(B)+T7                  E249
      ZP(8)=(.5)*(E*(T7+T9-2*T8)+(2*G)*(TR-T8))/(B)+T8                  E249
      ZP(9)=(E*(T8-T9)+G*(TR-T9))/(B)+T9                                E249
      P1=P2                                                             E249
      P2=Z(9)                                                           E249
      P3=ZP(9)                                                          E249
      T=T+DELTAT                                                        E249
      LAP=LAP+1                                                         E249
      IF(N2.GT.LAP) GO TO 20                                            E249
      T=0.                                                              E249
      LAP=0                                                             E249
C                                                                       E249
C*** RESETTING THE VARIABLES                                            E249
   20 DO 30 IP=1,9                                                      E249
      Z(IP)=ZP(IP)                                                      E249
   30 CONTINUE                                                          E249
      IF(P2.GT.0.0) GO TO 40                                            E249
      IF(P3.LT.0.0) GO TO 40                                            E249
      ZC1=P2                                                            E249
      ZC2=P3                                                            E249
      XIC=I                                                             E249
      XIC=XIC-2.                                                        E249
   40 IF(ABS(P1).GT.ABS(P2)) GO TO 70                                   E249
      IF(ABS(P2).LE.ABS(P3)) GO TO 70                                   E249
      IF(P2.GE.0.0) GO TO 50                                            E249
      PKNEG=P2                                                          E249
      GO TO 60                                                          E249
   50 PKPOS=P2                                                          E249
C                                                                       E249
C*** CHECKING ON CONVERGENCE                                            E249
   60 PKDIF=PKPOS-ABS(PKNEG)                                            E249
      IF(PKNEG.EQ.0.0) PKQT = PKDIF                                     E249
      IF(PKNEG.NE.0.0) PKQT=PKDIF/ABS(PKNEG)                            E249
      IF(PKQT.LT.0.001) GO TO 80                                        E249
      XXX = 200.0*XN2                                                   E249
      IF(XIC.LT.XXX) GO TO 70                                           E249
      PKPOS=(PKPOS-PKNEG)/2.                                            E249
      GO TO 80                                                          E249
   70 CONTINUE                                                          E249
   80 A(1)=PKPOS                                                        E249
      XNC=XIC/XN2                                                       E249
      NC = XNC                                                          E249
      YNC = NC                                                          E249
      CHR=YNC*XN2                                                       E249
      FRC=XIC-CHR                                                       E249
      ZC = ZC2-ZC1                                                      E249
      IF(ZC.EQ.0.0) GO TO 90                                            E249
C                                                                       E249
      FRC=FRC-ZC1/(ZC2-ZC1)                                             E249
   90 PHSLAG=(FRC/XN2)*(-360.)                                          E249
      A(2)=PHSLAG                                                       E249
      RETURN                                                            E249
```


7197C

```
      END                                                               E249
C                                                                       E249
C                                                                       E249
      SUBROUTINE TRFP(J,TC,FRQ,A,TP)                                    E249
C                                                                       E249
C***********************************************************************E249
C        "TRFP"  CALCULATES THE PARAMETERS REQUIRED BY THE             *E249
C               TRANSFER FUNCTION PROGRAM "TRFM".                      *E249
C***********************************************************************E249
C                                                                       E249
C ---- IDENTIFICATION ----                                              E249
C                                                                       E249
C                                                                       E249
C** J - CODE FOR WHICH T/C IS BEING EVALUATED                           E249
C     -- INPUT                                                          E249
C                                                                       E249
C** TC - THE ARRAY HOLDING THE T/C PARAMETERS                           E249
C     --INPUT                                                           E249
C                                                                       E249
C** FRQ - FREQUENCY FOUND IN TRANGS, NEEDED IN SPCY                     E249
C     -- INPUT                                                          E249
C                                                                       E249
C** A - ARRAY INTO WHICH THE SAMPLING FREQUENCY FOUND BY                E249
C       SUBROUTINE SPCY WILL BE PUT                                     E249
C     -- OUTPUT                                                         E249
C                                                                       E249
C** TP - ARRAY CONTAINING THE PARAMETERS EVALUATED BY TRFP              E249
C     -- OUTPUT                                                         E249
C                                                                       E249
C  1. DELTA                                                             E249
C  2. DELTAT                                                            E249
C  3. SIGMA                                                             E249
C  4. CN                                                                E249
C  5. A1                                                                E249
C  6. B                                                                 E249
C  7. C                                                                 E249
C  8. E                                                                 E249
C  9. F                                                                 E249
C 10. G                                                                 E249
C                                                                       E249
C REFERENCE "DYNAMIC GAS TEMPERATURE MEASURING SYSTEM - SYSTEM          E249
C DESIGN AND TEST PLAN (FR-16381)" FOR DEFINITION OF ABOVE TERMS.       E249
C                                                                       E249
C     CALLED BY TRANGS SUBPROGRAM                                       E249
C     CALLS - SPCY:  DETERMINES THE SAMPLING FREQUENCY AS A FUNCTION    E249
C                    OF THE ANALYSIS FREQUENCY                          E249
C                                                                       E249
C***********************************************************************E249
C                                                                       E249
      DIMENSION A(2),TC(4),TP(10)                                       E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),   E249
     *    IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),E249
```

```
     •    PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                                 E249
C                                                                           E249
C*** FIND THE SAMPLING FREQUENCY                                            E249
      CALL SPCY(FRQ,A)                                                      E249
      XN2=A(1)                                                              E249
C                                                                           E249
C*** CALCULATE THE TRANSFER FUNCTION PARAMETERS                             E249
      DELTA = TCDATA(1,J) / 3.0                                             E249
      DELTAT = 1./(XN2*FRQ)                                                 E249
      SIGMA = TCDATA(2,J) / 3.0                                             E249
      CN = 2.*3.1415*FRQ                                                    E249
      A1 = (TCDATA(3,J)**2 * DELTA) / (8.*TC(4)*DELTAT)                     E249
      B = (TCDATA(4,J)**2 * SIGMA) / (8.*TC(4)*DELTAT)                      E249
      C = (TCDATA(3,J)**2) / (4.*DELTA)                                     E249
      E = (TCDATA(4,J)**2) / (4.*SIGMA)                                     E249
      F = (A(2) * SQRT(TCDATA(3,J)) * DELTA) / (2.*TC(4))                   E249
      G = (A(2) * SQRT(TCDATA(4,J)) * SIGMA) / (2.*TC(4))                   E249
C                                                                           E249
C*** STORE THE PARAMETERS INTO THE ARRAY TP                                 E249
      TP(1) = DELTA                                                         E249
      TP(2) = DELTAT                                                        E249
      TP(3) = SIGMA                                                         E249
      TP(4) = CN                                                            E249
      TP(5) = A1                                                            E249
      TP(6) = B                                                             E249
      TP(7) = C                                                             E249
      TP(8) = E                                                             E249
      TP(9) = F                                                             E249
      TP(10) = G                                                            E249
      RETURN                                                                E249
      END                                                                   E249
C                                                                           E249
C                                                                           E249
      SUBROUTINE WINDOW(WINDO, DATA)                                        E249
C                                                                           E249
C*********************************************************************** E249
C    WINDOW APPLYS THE P301 WINDOW TO THE INPUT DATA BLOCK               * E249
C*********************************************************************** E249
C                                                                           E249
C --- IDENTIFICATION ---                                                    E249
C                                                                           E249
C                                                                           E249
C** WINDO - THE P301 WINDOW FOUND IN WINGEN                                 E249
C     --- INPUT                                                             E249
C                                                                           E249
C** DATA - ARRAY TO WHICH THE WINDOW IS TO BE APPLIED                       E249
C     --- INPUT AND OUTPUT                                                  E249
C                                                                           E249
C     CALLED BY POWER SUBPROGRAM                                            E249
C                                                                           E249
C************************************************************************E249
C                                                                           E249
      DIMENSION WINDO(2048), DATA(2048)                                     E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),       E249
```

```
     *   IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),  E249
     *   PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                                E249
C                                                                         E249
      DO 10 I = 1,IBLSZ                                                   E249
      DATA(I) = DATA(I) * WINDO(I)                                        E249
   10 CONTINUE                                                            E249
      RETURN                                                              E249
      END                                                                 E249
C                                                                         E249
C                                                                         E249
      SUBROUTINE WINGEN(WINDO)                                            E249
C                                                                         E249
C**********************************************************************   E249
C    WINGEN GENERATES THE P301 WINDOWING FUNCTION                       * E249
C**********************************************************************   E249
C                                                                         E249
C ---- IDENTIFICATION ----                                                E249
C                                                                         E249
C                                                                         E249
C** WINDO - ARRAY TO CONTAIN THE WINDOWING FUNCTION                       E249
C      --- OUTPUT                                                         E249
C                                                                         E249
C      CALLED BY POWER SUBPROGRAM                                         E249
C                                                                         E249
C**********************************************************************   E249
C                                                                         E249
      DIMENSION WINDO(2048)                                               E249
      COMMON /INPUTS/ IFLAGS(12),TCDATA(4,2),GAS(4),FREQ(4),CHANL(9),     E249
     *   IAVDAT(2),IBLSZ,IREC(10),TIME(2),IBSZ,GAMMA,NREC(10),NRECS(2),  E249
     *   PLTFRQ,TIMTEM,IDEBUG,IBUG2,ITHRSH                                E249
C                                                                         E249
      TWOPI = 2.0 * 3.1415927                                             E249
      DO 10 I = 1,IBLSZ                                                   E249
      WINDO(I) = 0.9994484 + 2.0 * (0.955728*COS(TWOPI*(I-1)/IBLSZ) +     E249
     * 0.539289*COS(2.0*TWOPI*(I-1)/IBLSZ) +                              E249
     * 0.091581*COS(3.0*TWOPI*(I-1)/IBLSZ))                               E249
   10 CONTINUE                                                            E249
      ISIZ = IBSZ - 1                                                     E249
      DO 20 I = 1,ISIZ                                                    E249
      II = I + ISIZ                                                       E249
      TEMP = WINDO(II)                                                    E249
      WINDO(II) = WINDO(I)                                                E249
      WINDO(I) = TEMP                                                     E249
   20 CONTINUE                                                            E249
      RETURN                                                              E249
      END                                                                 E249
```


7197C

TEST CASES

The program was developed and tested on an IBM 3090 computer with a VM/CMS operating system. The CMS environment facilitated interactive execution and testing of the program. The program was written in the IBM System/370 FORTRAN IV language.

Seven test cases were run to show the available user options. A sample EXEC with comments as to the statements functions is shown in Figure 10.

```
&TRACE off
GLOBAL TXTLIB FGHLIB CMSLIB PWALIB SUBLIB               - needed txtlibs
 EXEC CCPU
 &READ VARS &MINUTES &SECONDS &HUNDRED                  - accounting info
CLEAR
EXEC PACKSCAN ENG005 STACK LINK                         - accessing data
&READ VARS &VAD &MOD                                      input file from TSO
FI 4 DISK E249SIN DATA &MOD DSN E100827 $E249 SIN DATA   - test data input
FI 5 DISK E249SIN INPUT A (LRECL 80 BLKSIZE 80)          - user input
FI 6 DISK E249SIN OUTPUT A (LRECL 133 BLKSIZE 133)       - output file
FI 12 DISK E249COM3 SIN    A                             - comp spectrum
FI 13 DISK E249DATS SIN    A                             - small T/C scaled
                                                           data
FI 14 DISK E249DATL SIN    A                             - large T/C scaled
                                                           data
FI 15 DISK E249FFTS SIN    A                             - small T/C FFT
FI 16 DISK E249FFTL SIN    A                             - large T/C FFT
FI PLOTPARM DISK VECINR DATA 0                          - needed filedefs
FI PLOTLOG TERMINAL
FI VECTR1 DISK VECTR1 DATA A
FI VECTR2 DISK VECTR2 DATA A                            - calcomp filedefs
&TYPE EXECUTING ........
E249                                                    - execute the module
STATE VECTR1 DATA A
*
&IF &RETCODE NE 0 &GOTO -CONT
EXEC USE SYNCSORT
&STACK 1 4 CH A
SSORT VECTR2 DATA A SECTR2 DATA A (OUTPUT REP
FILEDEF PLOTLOG TERMINAL
FILEDEF RJEVECTR PRINTER (LRECL 132 BLOCK 132 RECFM FB)
FILEDEF SECTR2 DISK SECTR2 DATA A
FILEDEF VECTR1 DISK VECTR1 DATA A
EXEC ROUTE PRINT LOCAL SYSOUT=6
CP SPOOL PRT CONT
LOADM VTPLOT
START
CP SPOOL PRT CLOSE
CP SPOOL PRT NOCONT
ERASE VECTR1 DATA A
ERASE VECTR2 DATA A
ERASE SECTR2 DATA A                                     - calcomp plotting
****  ACCOUNTING INFORMATION  ***********
      EXEC CCPU &MINUTES &SECONDS &HUNDRED
      &READ VARS &MINUTES &SECONDS &CRU &COST
      EXEC ACCNT2 183 E249 &CRU &COST
      &TYPE VIRTUAL CPU USED : &MINUTES MINUTE(S) &SECONDS SECOND(S)
      &TYPE ESTIMATED CRU : &CRU
      &TYPE ESTIMATED COST : $ &COST                     - accounting info
*************
EXEC USE SYNCSORT (DET)
*REL &MODE (DET)
&EXIT
```

FD 316914

Figure 10. Sample EXEC Program

Test Case 1

The input file that was used to run the first test case is shown in Figure 11. Please note that lines 14 through 20 are not necessary.

```
    .3048   .140589    .0508  .025999   .20828    .0635    .0381  .0077787
      .02   1266.48     .355
      1.0893784E+06
.24414E-03
      8.0       8.0      2.0
   2500.0       2.0     .125   1500.0      0.5      -.2     50.0       1.0
      -.4
   1    3
 2048
   2000.0
   1    1    1    1    2    4    2    0    0    0    0    0
   1
   0
 CHECK-OUT TEST CASE B
 8 HZ SIN WAVE  USING  PT / 6% RH
 6/6/86
```

FD 316915

Figure 11. Input for Test Case 1

7197C

Since this case is a first time run (IFLAGS(1)=1), a measured gamma and compensation spectrum will be calculated. IDEBUG has been set to one so a printout of the interpolation for a measured gamma will be produced. Printed output generated from this test case is shown in Figure 12. The only plot generated by this execution is of the compensation spectrum and is shown in Figure 13.

```
FILE: RUN1      OUTPUT   A            P R A T T   A N D   W H I T N E Y   A I R C R A F T          PAGE 00001

1INPUT LISTING

                                        CARD COLUMN

         1111111111222222222233333333334444444444555555555566666666667777777778
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

    .3048   .140589    .0508   .025999    .20828     .0635     .0381  .0077787
      .02   1266.48     .355
     1.0893784E+06
.24414E-03
      8.0       8.0      2.0
   2500.0       2.0     .125    1500.0       0.5        -.2     50.0       1.0
      -.4
   1    3
2048
   2000.0
   1    1    1    1    2    4    2    0    0    0    0    0
   1
   0
CHECK-OUT TEST CASE B
8 HZ SIN WAVE  USING  PT / 6% RH
6/6/86


         1111111111222222222233333333334444444444555555555566666666667777777778
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

                                        CARD COLUMN
1               10 MIL                                                3 MIL
+               ------                                                -----

     LENGTH OF SUPPORT WIRE    =    0.30480 CM       LENGTH OF SUPPORT WIRE    =    0.20828 CM
     LENGTH OF SMALLER WIRE    =    0.14059 CM       LENGTH OF SMALLER WIRE    =    0.06350 CM
     DIAMETER OF SUPPORT WIRE  =    0.05080 CM       DIAMETER OF SUPPORT WIRE  =    0.03810 CM
     DIAMETER OF SMALLER WIRE  =    0.02600 CM       DIAMETER OF SMALLER WIRE  =    0.00778 CM
0    FUEL TO AIR RATIO IS             0.02000
     MEAN GAS TEMPERATURE (K) IS  1266.47998
     MEAN GAS PRESSURE (PA) IS    0.1089378E+07
     MACH NUMBER IS                   0.35500
0    DELTA-T = 0.2441400E-03
     START FREQ =      8.0000    END FREQ =     8.0000     FREQ INCREMENT =     2.0000
0             GAIN        INPUT/OUTPUT       OFFSET
              ----        ------------       ------
 3 MIL    2500.0000          2.00000        0.125000
 10 MIL   1500.0000          0.50000       -0.200000
 DC         50.0000          1.00000       -0.400000
OTHE ENSEMBLE AVERAGING STARTS WITH RECORD    1 AND  USES   3 RECORDS.
 WE HAVE A BLOCKSIZE OF  2048
 ALL FREQUENCY DOMAIN PLOTS WILL END AS CLOSE TO   2000.000 HZ AS POSSIBLE
0    FLAGS              DESCRIPTION                            VALUE
+    -----              -----------                            -----
       1     WHERE TO BEGIN PROGRAM CALCULATIONS                  1
```

FD 316916

Figure 12. Output Generated by Test Case 1

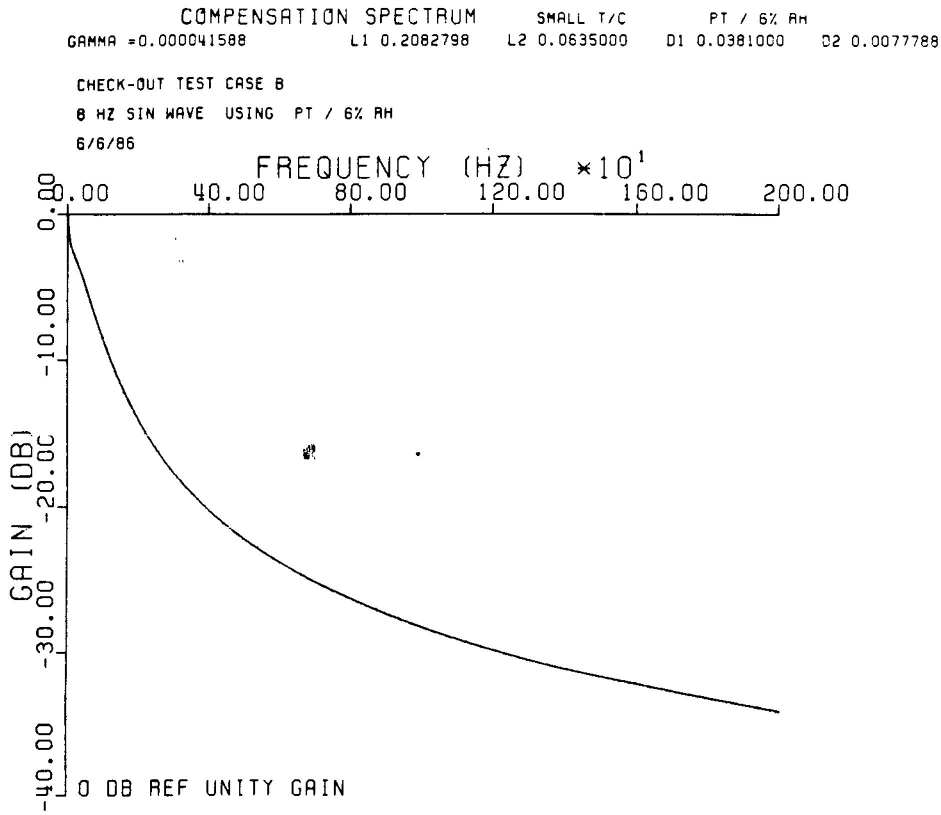
```
FILE: RUN1      OUTPUT   A             P R A T T   A N D   W H I T N E Y   I R C R A F T        PAGE 00002

      2       T/C MATERIAL CODE                                  1
      3       T/C USED FOR COMPENSATION SPECTRUM                 1
      4       PLOT OF COMPENSATION SPECTRUM DESIRED?             1
      5       PLOT OF INSTANTANEOUS DATA DESIRED?                2
      6       TYPE OF SCALING DONE TO FREQUENCY DATA             4
      7       PLOT OF AVERGED FREQUENCY DATA DESIRED?            2
      8       AVERAGE ONE OR MANY RECORDS?                       0
      9       COMPENSATED DATA?                                  0
     10       PLOT TIME AND FREQUENCY DOMAINS?                   0
     11       PLOT FULL TIME RANGE?                              0
     12       TEMPERATURE TO SCALE DATA ON PLOTS                 0
0IDEBUG IS SET TO 1
0IBUG2 IS SET TO 0
1THE ESTIMATED GAMMA IS 0.3907835E-04
 CHANNEL (  5)  MEASURED GAIN IS 0.7042887E+00
     ESTIMATED GAIN FOR 0.20 GAMMA IS 0.4448995E+00
     ESTIMATED GAIN FOR 0.30 GAMMA IS 0.4843767E+00
     ESTIMATED GAIN FOR 0.40 GAMMA IS 0.5213147E+00
     ESTIMATED GAIN FOR 0.50 GAMMA IS 0.5557104E+00
     ESTIMATED GAIN FOR 0.60 GAMMA IS 0.5873098E+00
     ESTIMATED GAIN FOR 0.70 GAMMA IS 0.6164644E+00
     ESTIMATED GAIN FOR 0.80 GAMMA IS 0.6434885E+00
     ESTIMATED GAIN FOR 0.90 GAMMA IS 0.6678530E+00
     ESTIMATED GAIN FOR 1.00 GAMMA IS 0.6908519E+00
     ESTIMATED GAIN FOR 1.10 GAMMA IS 0.7117718E+00
0   INTERPOLATED GAMMA
+   __________________
        0.4158830E-04
0   AVERAGED GAMMA FOUND = 0.4158830E-04
0
 EXECUTION OF THE PROGRAM HAS BEEN COMPLETED!!
```

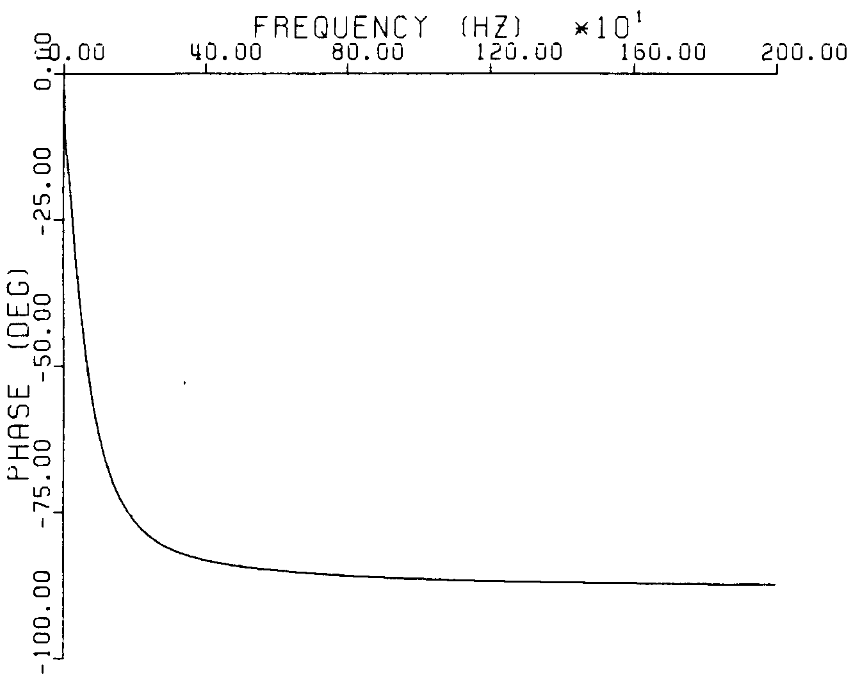
FD 316916

Figure 12. Output Generated by Test Case 1 (Continued)

7197C

ORIGINAL PAGE IS
OF POOR QUALITY





FD 316917

Figure 13. Test Case 1 Plot 1

7197C

Test Case 2

The second test case uses the compensation spectrum, scaled data, and FFT files generated in Test Case 1. This greatly reduces the execution time. This case requests an instantaneous, compensated plot of Record 1 using a threshold level of −80 dB. The averaged frequency domain plot is generated using Records 1 through 3, and scaling in K^2/Hz. The input to Test Case 2 is shown in Figure 14, noting that input lines 14, 16, 17, and 19 are not necessary.

```
     .3048   .140589     .0508   .025999    .20828     .0635     .0381  .0077787
       .02   1266.48     .355
       1.0893784E+06
 .24414E-03
       8.0       8.0       2.0
    2500.0       2.0      .125    1500.0       0.5       -.2      50.0       1.0
       -.4
   1    3
 2048
    2000.0
   3    0    0    2    1    1    1    1    1    1    1    1
   0
   0
   1
   1    3
  -80
 CHECK-OUT TEST CASE B    COMPENSATED WITH THRESHOLD OF -80DB
 8 HZ SIN WAVE  USING  PT / 6% RH
 6/6/86
```

FD 316918

Figure 14. Input for Test Case 2

Printed output for Test Case 2 is a card copy of the input along with a summary of the input variables shown in Figure 15. When IDEBUG and IBUG2 are both set to zero, this is the only output generated. Figures 16 and 17 show the plots generated. Note that the frequency domain plots could be 'spread out' so as to see the data better by changing the value of PLTFRQ input on line 10.

```
FILE: RUN2          INPUT   A                  P R A T T   A N D   W H I T N E Y   A I R C R A F T              PAGE 00001

1INPUT LISTING

                                                  CARD COLUMN

          111111111122222222223333333333444444444455555555556666666666777777777778
 12345678901234567890123456789012345678901234567890123456789012345678901234567890

     .3048   .140589    .0508   .025999    .20828     .0635      .0381  .0077787
       .02   1266.48     .355
      1.0893784E+06
 2.4414E-04
       8.0       8.0      2.0
    2500.0       2.0     .125    1500.0       0.5       -.2       50.0       1.0
       -.4
   1    3
  2048
    2000.0
    3    0    0    2    1    1    1    1    1    1    1    1
    0
    0
    1
    1    3
   -80
 CHECK-OUT TEST CASE B    COMPENSATED WITH THRESHOLD OF -80DB
 8 HZ SIN WAVE  USING  PT / 6% RH
 6/6/86


          111111111122222222223333333333444444444455555555556666666666777777777778
 12345678901234567890123456789012345678901234567890123456789012345678901234567890

                                                  CARD COLUMN
 1              10 MIL                                               3 MIL
 +              ------                                               -----

      LENGTH OF SUPPORT WIRE    =     0.30480 CM           LENGTH OF SUPPORT WIRE    =     0.20828 CM
      LENGTH OF SMALLER WIRE    =     0.14059 CM           LENGTH OF SMALLER WIRE    =     0.06350 CM
      DIAMETER OF SUPPORT WIRE  =     0.05080 CM           DIAMETER OF SUPPORT WIRE  =     0.03810 CM
      DIAMETER OF SMALLER WIRE  =     0.02600 CM           DIAMETER OF SMALLER WIRE  =     0.00778 CM
 0   FUEL TO AIR RATIO IS               0.02000
     MEAN GAS TEMPERATURE (K) IS     1266.47998
     MEAN GAS PRESSURE (PA) IS       0.1089378E+07
     MACH NUMBER IS                     0.35500
 0   DELTA-T = 0.2441400E-03
     START FREQ =       8.0000      END FREQ =       8.0000      FREQ INCREMENT =       2.0000
 0            GAIN         INPUT/OUTPUT        OFFSET
              ----         ------------        ------
  3 MIL     2500.0000        2.00000          0.125000
  10 MIL    1500.0000        0.50000         -0.200000
  DC          50.0000        1.00000         -0.400000
 0THE ENSEMBLE AVERAGING STARTS WITH RECORD    1 AND   USES    3 RECORDS.
  WE HAVE A BLOCKSIZE OF   2048
  ALL FREQUENCY DOMAIN PLOTS WILL END AS CLOSE TO    2000.000 HZ AS POSSIBLE
```

FD 316919

Figure 15. Output Generated by Test Case 2

```
FILE: RUN      OUTPUT   A              P R A T T   A N D   W H I T N E Y   I R C R A F T          PAGE 00002

0    FLAGS            DESCRIPTION                               VALUE
+    _____            ___________                               _____
       1      WHERE TO BEGIN PROGRAM CALCULATIONS                 3
       2      T/C MATERIAL CODE                                   0
       3      T/C USED FOR COMPENSATION SPECTRUM                  0
       4      PLOT OF COMPENSATION SPECTRUM DESIRED?              2
       5      PLOT OF INSTANTANEOUS DATA DESIRED?                 1
       6      TYPE OF SCALING DONE TO FREQUENCY DATA              1
       7      PLOT OF AVERGED FREQUENCY DATA DESIRED?             1
       8      AVERAGE ONE OR MANY RECORDS?                        1
       9      COMPENSATED DATA?                                   1
      10      PLOT TIME AND FREQUENCY DOMAINS?                    1
      11      PLOT FULL TIME RANGE?                               1
      12      TEMPERATURE TO SCALE DATA ON PLOTS                  1
0IDEBUG IS SET TO 0
0IBUG2 IS SET TO 0
0
 EXECUTION OF THE PROGRAM HAS BEEN COMPLETED!!
```

FD 316919

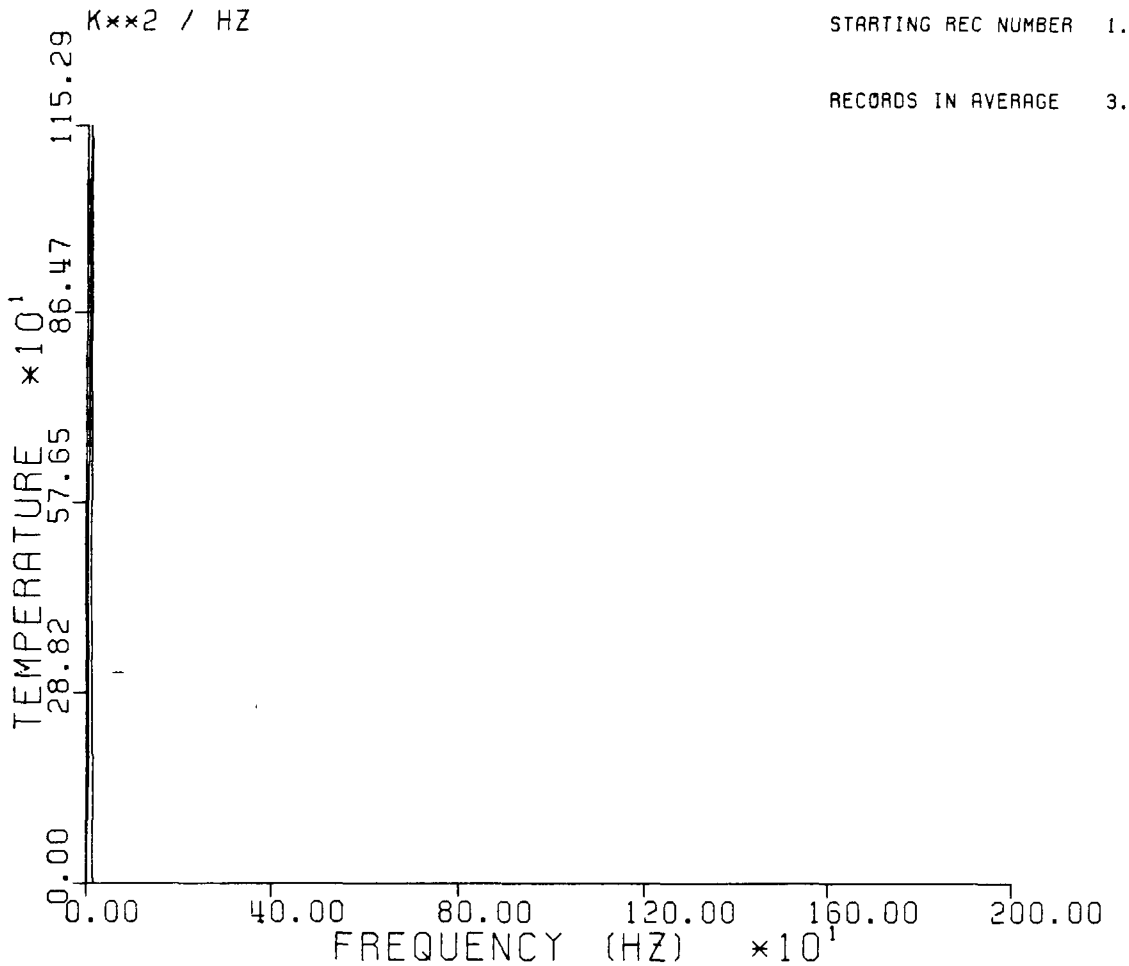
Figure 15. Output Generated by Test Case 2 (Continued)

ORIGINAL PAGE IS
OF POOR QUALITY

AVERAGED FREQUENCY DOMAIN DATA

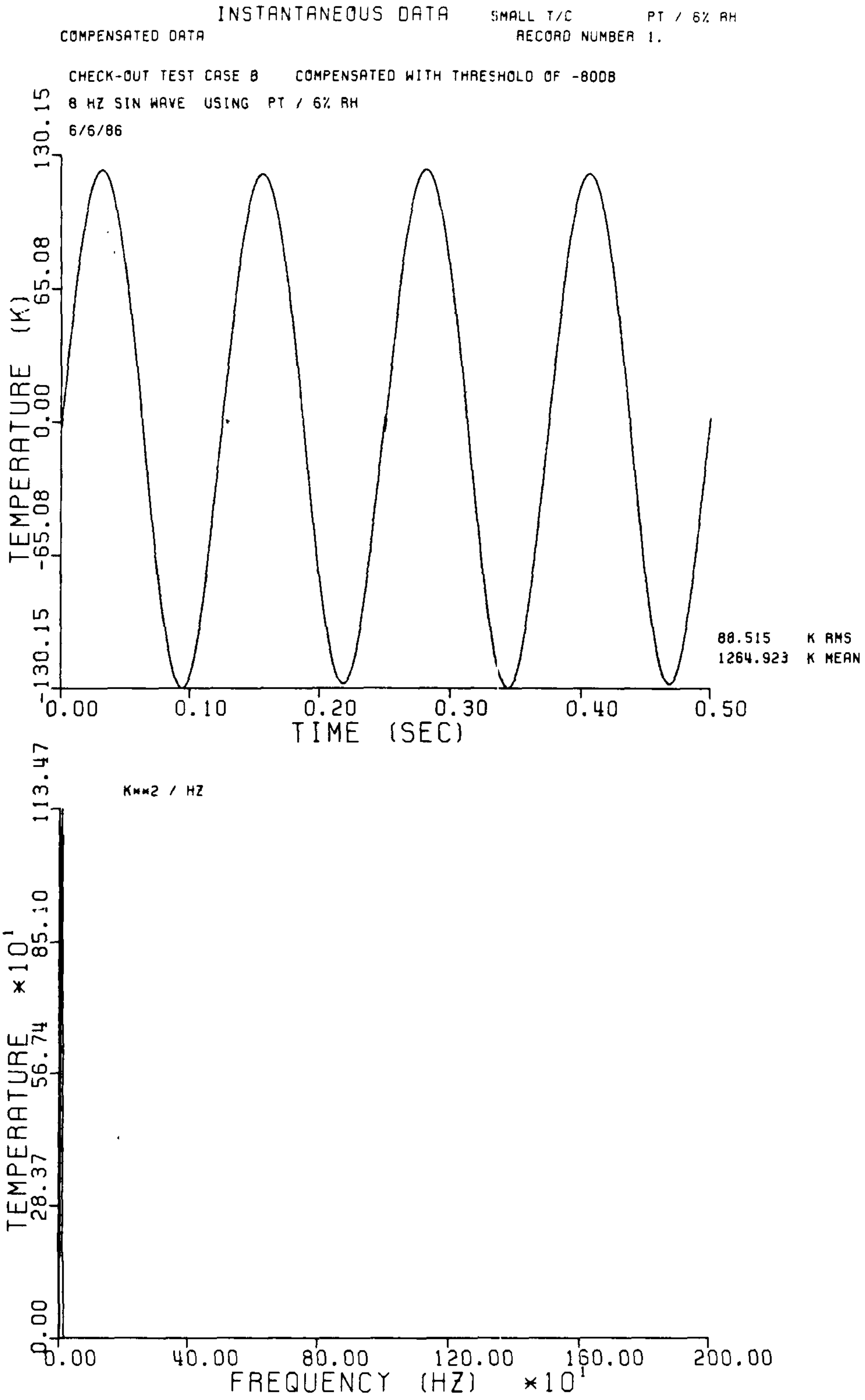
SMALL T/C   COMPENSATED DATA

CHECK-OUT TEST CASE B   COMPENSATED WITH THRESHOLD OF -80DB
8 HZ SIN WAVE USING PT / 6% RH
6/6/86



FD 316920

Figure 16. Test Case 2 Plot 1



FD 316921

Figure 17. Test Case 2 Plot 2

7197C

Test Case 3

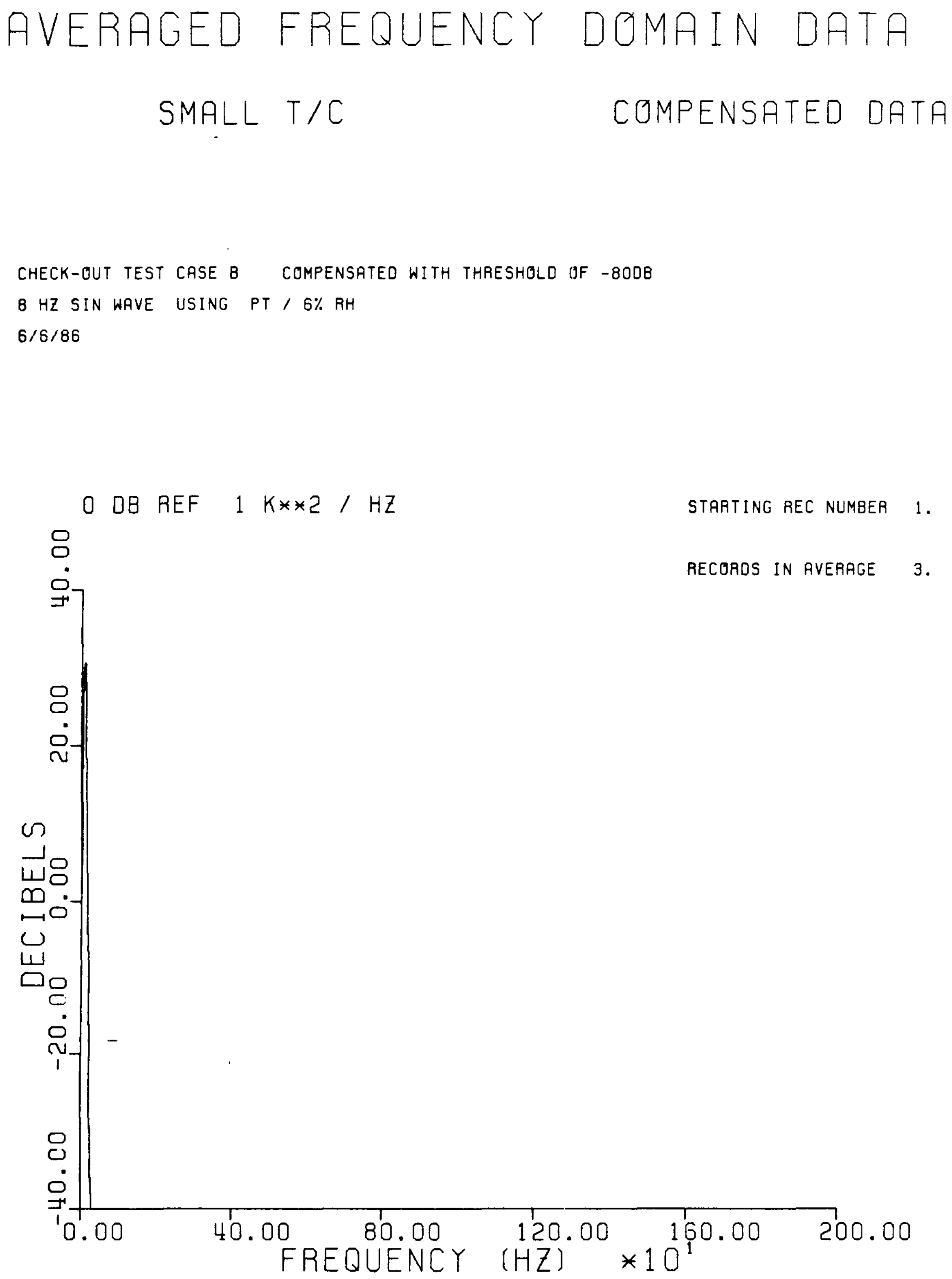
Test Case 3 demonstrates how partial time may be requested for instantaneous time domain plots. A compensated plot of Record 1 for time 0.125 second to 0.375 second is requested using a threshold of −80 dB. The instantaneous frequency domain plot is no longer needed so IFLAGS(10) has been set to 2. The averaged frequency domain is now scaled in decibels. Figure 18 shows input for Test Case 3, while Figures 19 and 20 show plots generated by this input. Please note that lines 14, 17, and 19 are not needed in the input file.

```
      .3048    .140589      .0508   .025999    .20828      .0635     .0381  .0077787
        .02    1266.48       .355
       1.0893784E+06
 .24414E-03
        8.0        8.0        2.0
     2500.0        2.0       .125    1500.0       0.5        -.2      50.0       1.0
        -.4
    1   3
 2048
     2000.0
     3   0    0    2    1    2    1    1    1    2    2    1
     0
     0
     1
       .125       .375
    1   3
  -80
 CHECK-OUT TEST CASE B    COMPENSATED WITH THRESHOLD OF -80DB
 8 HZ SIN WAVE  USING  PT / 6% RH
 6/6/86
```

FD 316922

Figure 18. Input for Test Case 3

7197C



FD 316923

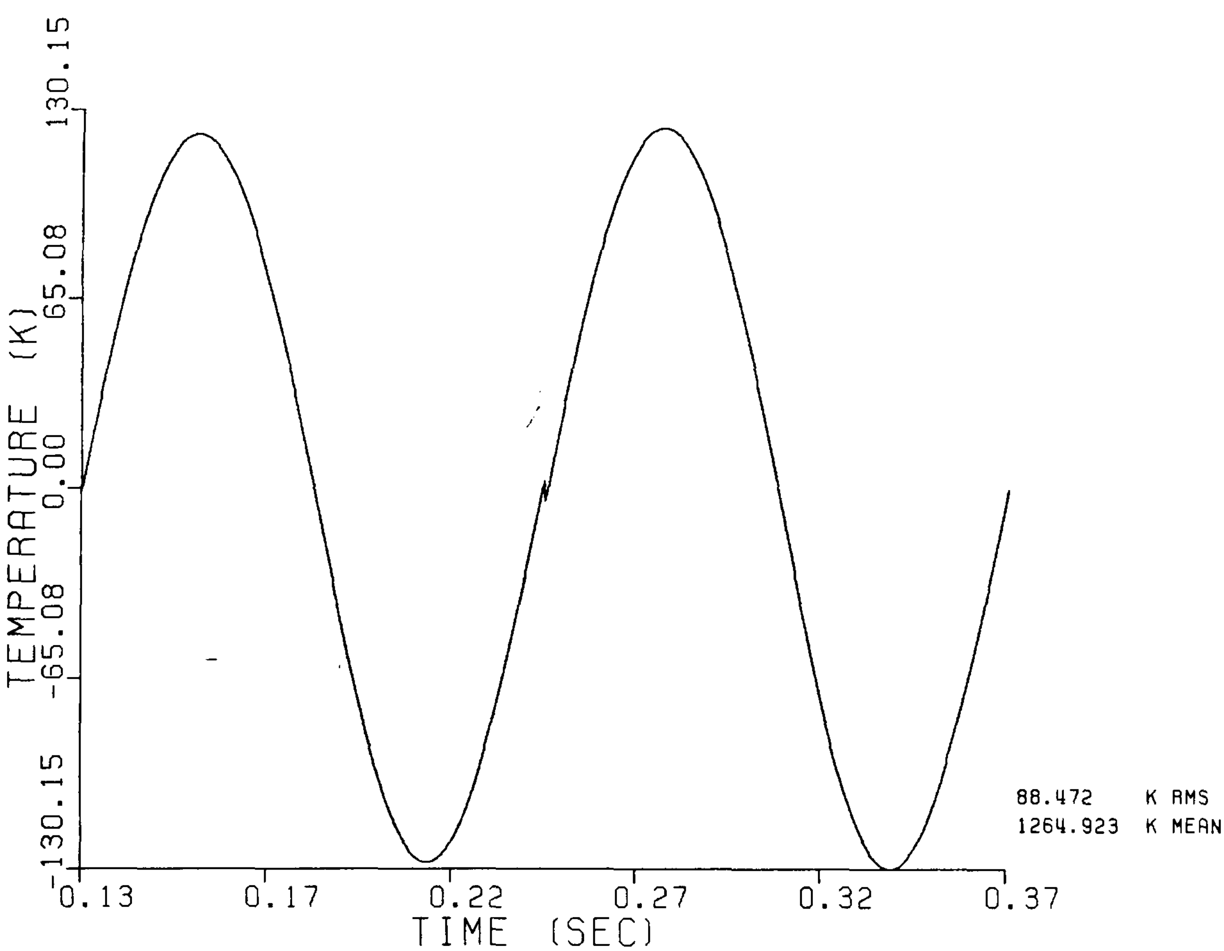
Figure 19. Test Case 3 Plot 1

COMPOSIT INSTANTANEOUS TIME WAVEFOR

SMALL T/C COMPENSATED DATA

INSTANTANEOUS DATA, RECORD NUMBER 1.

CHECK-OUT TEST CASE B COMPENSATED WITH THRESHOLD OF -80DB
8 HZ SIN WAVE USING PT / 6% RH
6/6/86



FD 316924

Figure 20. Test Case 3 Plot 2

7197C

Test Case 4

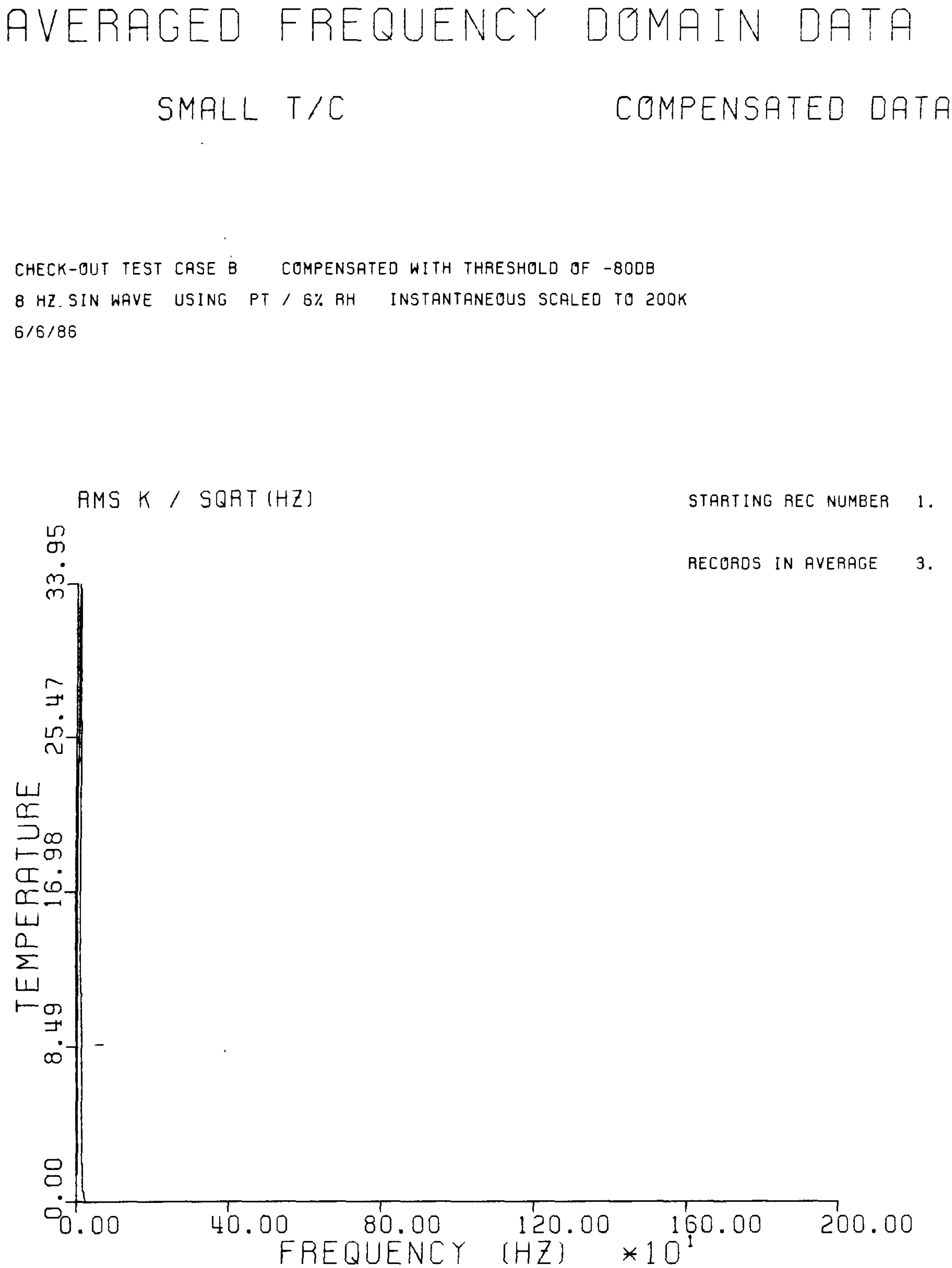
Test Case 4, whose input is shown in Figure 21, again requests a compensated, instantaneous plot of Record 1. This run now has the time domain scaled to a user input temperature of 200K. The averaged frequency domain is now scaled linearly (rms $K/\sqrt{Hz}$). The plots generated are shown in Figures 22 and 23. Please note that lines 14, 16, and 17 of the input description are not necessary.

```
    .3048   .140589     .0508   .025999     .20828      .0635      .0381  .0077787
      .02   1266.48      .355
      1.0893784E+06
.24414E-03
      8.0       8.0       2.0
   2500.0       2.0      .125    1500.0        0.5         -.2      50.0        1.0
      -.4
 1    3
2048
   2000.0
   3    0    0    2    1    3    1    1    1    2    1    2
   0
   0
   1
   1    3
     200.
  -80
CHECK-OUT TEST CASE B    COMPENSATED WITH THRESHOLD OF -80DB
8 HZ SIN WAVE  USING  PT / 6% RH   INSTANTANEOUS SCALED TO 200K
6/6/86
```

FD 316925

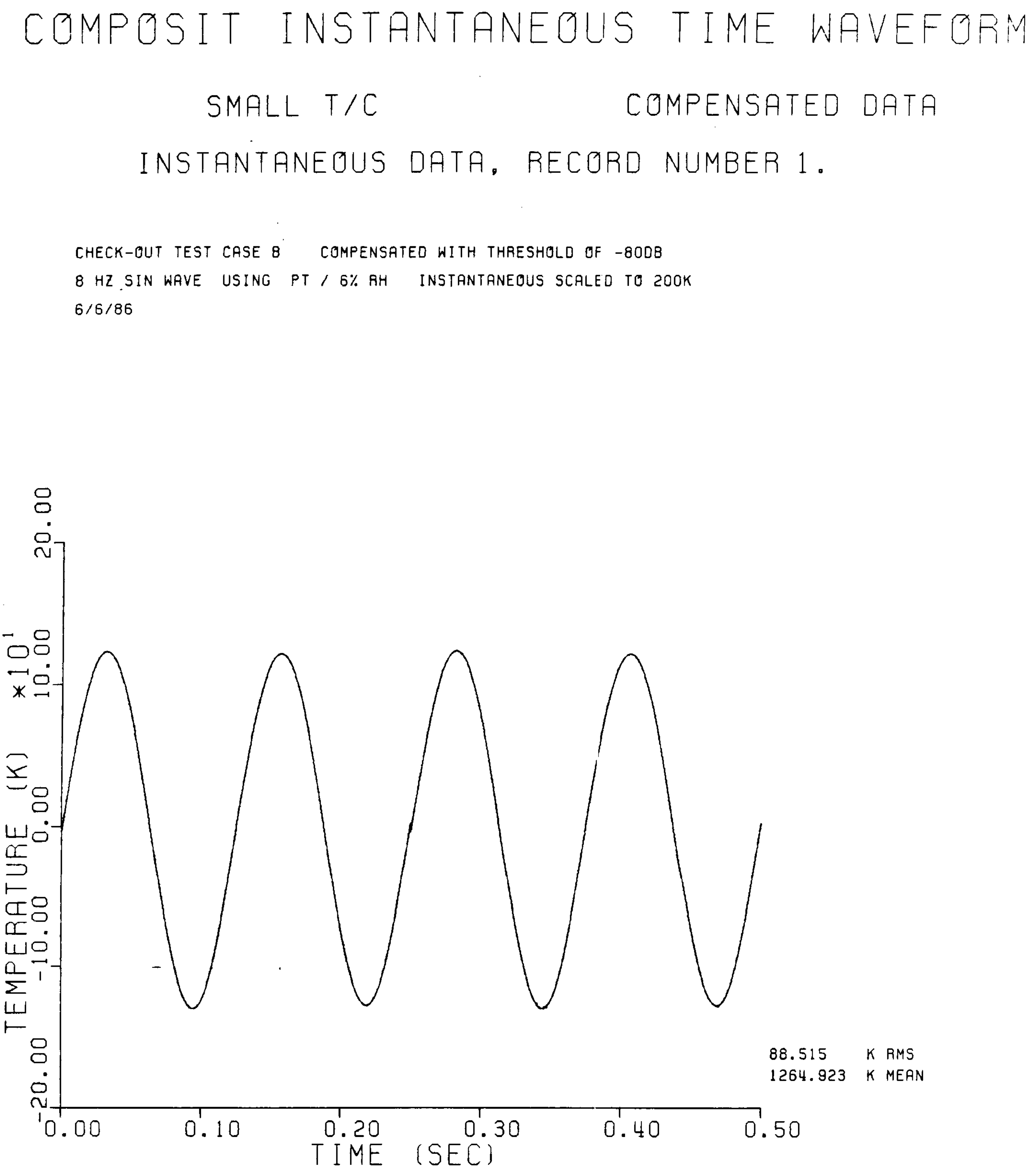
Figure 21. Input for Test Case 4

7197C



FD 316926

Figure 22. Test Case 4 Plot 1



FD 316927

Figure 23. Test Case 4 Plot 2

Test Case 5

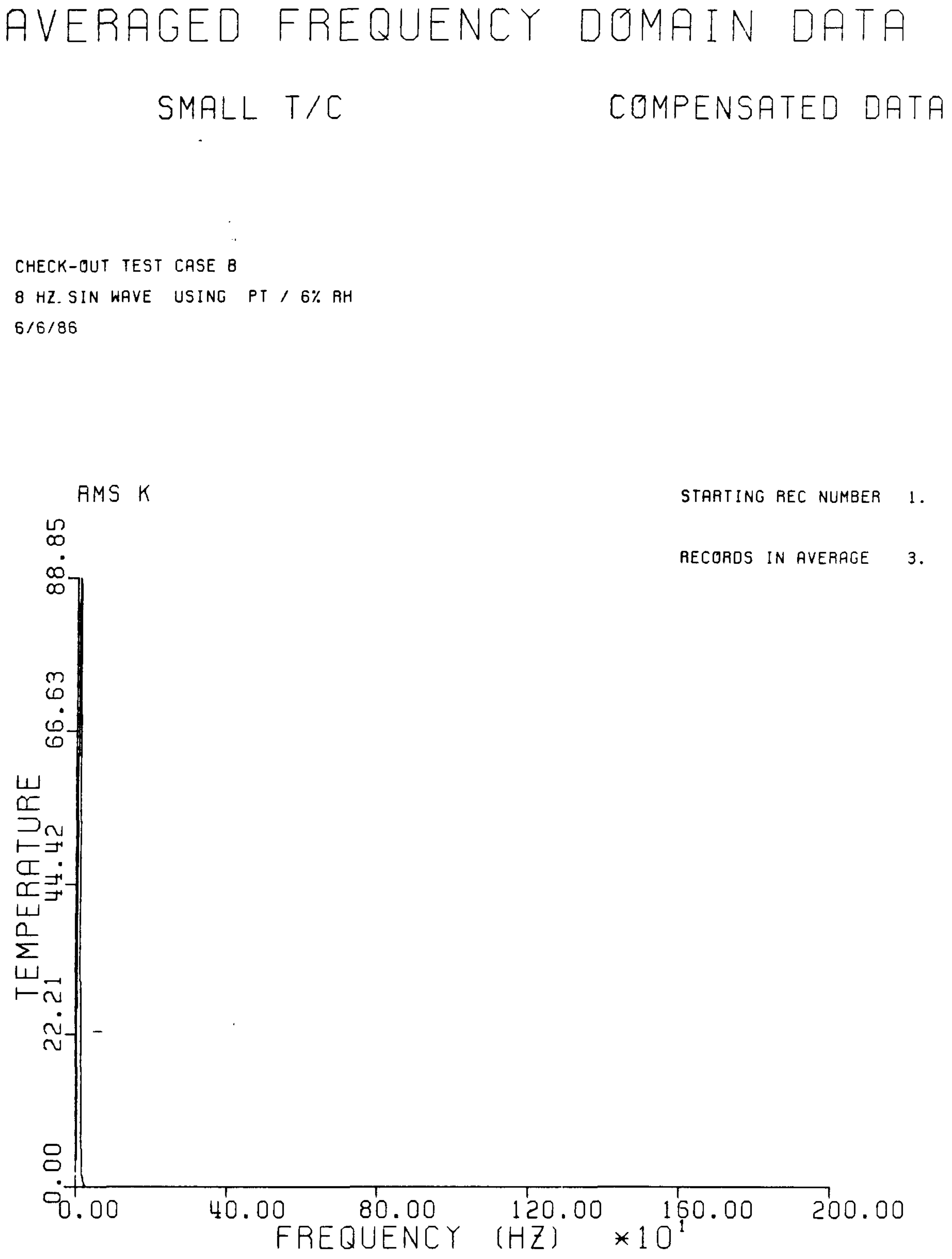
Test Case 5 requests only a plot of the averaged frequency domain data, scaled in narrowband (rms K). The plot is compensated, but since no instantaneous data are requested, a threshold level should not be input. Figure 24 shows input for Test Case 5 (lines 14, 15, 16, 17, and 19 are not needed) while Figure 25 is the plot generated by this input.

```
      .3048   .140589     .0508  .025999    .20828      .0635      .0381  .0077787
        .02   1266.48      .355
        1.0893784E+06
 .24414E-03
        8.0       8.0       2.0
     2500.0       2.0      .125   1500.0       0.5        -.2       50.0       1.0
        -.4
   1    3
  2048
     2000.0
     3    0    0    2    2    4    1    1    1    1    1    1
     0
     0
     1    3
  CHECK-OUT TEST CASE 8
  8 HZ SIN WAVE  USING  PT / 6% RH
  6/6/86
```

FD 316928

Figure 24. Input for Test Case 5

7197C



FD 316929

Figure 25. Test Case 5 Plot 1

7197C

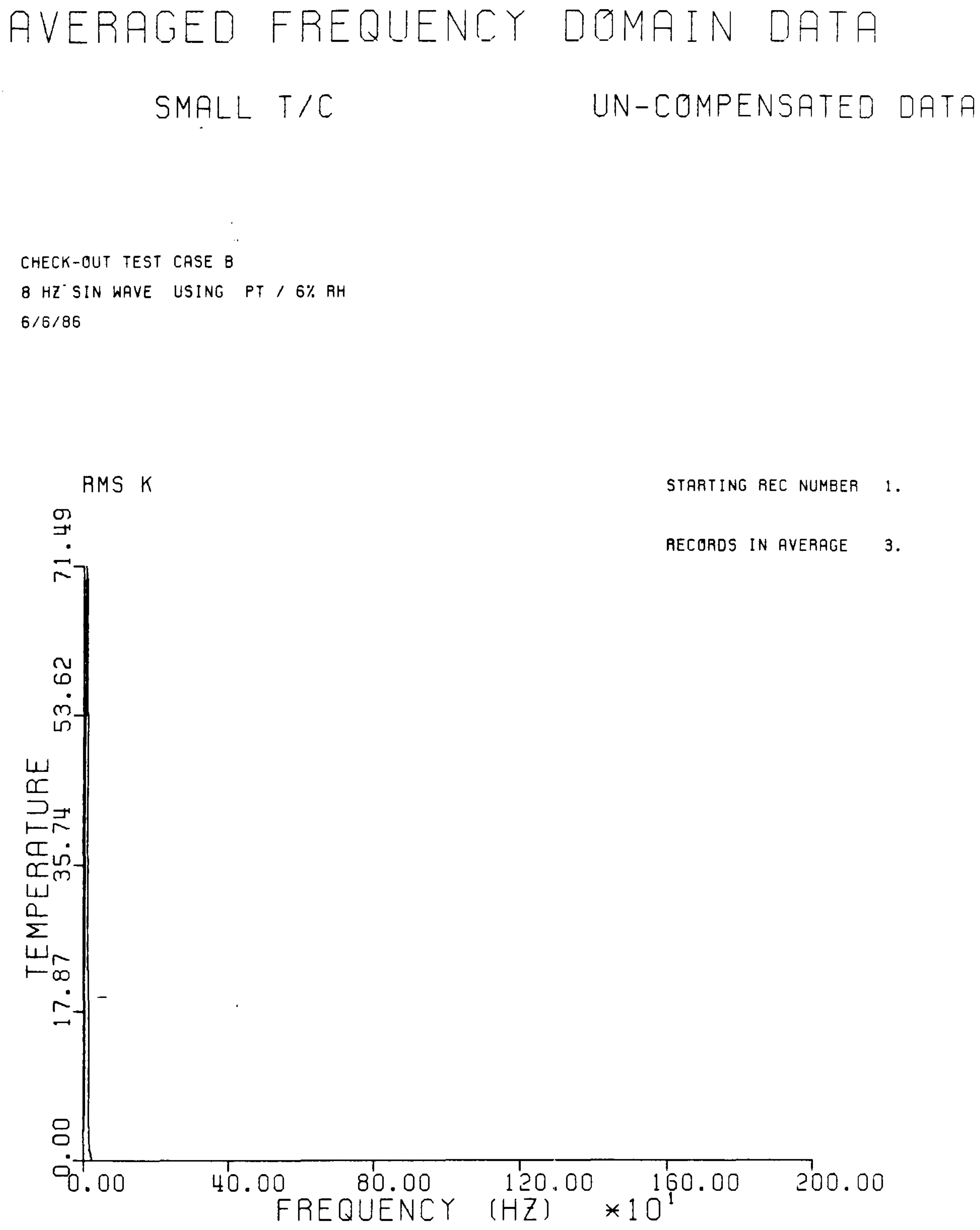
Test Case 6

Test Case 6 requests uncompensated plots of instantaneous data for Record 1 and averaged frequency domain using Records 1 through 3. Frequency domain is again scaled in rms K. Test Case 6 input is shown in Figure 26. Please note lines 14, 16, 17, and 19 are not present. Figures 27 and 28 show plots generated.

```
     .3048   .140589    .0508  .025999   .20828    .0635    .0381  .0077787
       .02   1266.48     .355
      1.0893784E+06
 .24414E-03
       8.0       8.0      2.0
    2500.0       2.0     .125   1500.0      0.5      -.2     50.0       1.0
       -.4
   1    3
 2048
    2000.0
   3    0    0    2    1    4    1    1    2    1    1    1
   0
   0
   1
   1    3
 CHECK-OUT TEST CASE B
 8 HZ SIN WAVE  USING  PT / 6% RH
 6/6/86
```

FD 316930

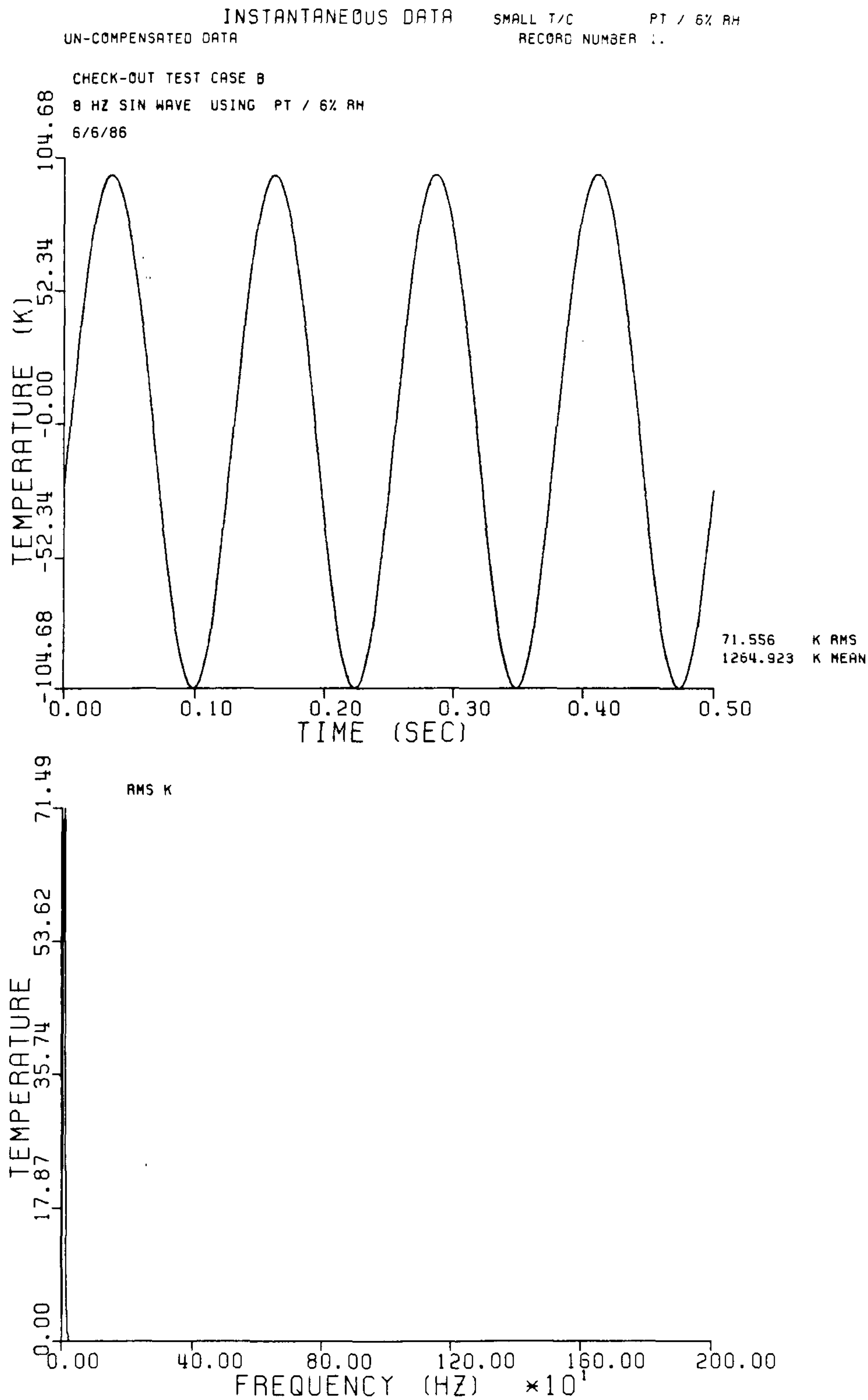
Figure 26. Input for Test Case 6



FD 316931

Figure 27. Test Case 6 Plot 1

7197C



FD 316932

Figure 28. Test Case 6 Plot 2

Test Case 7

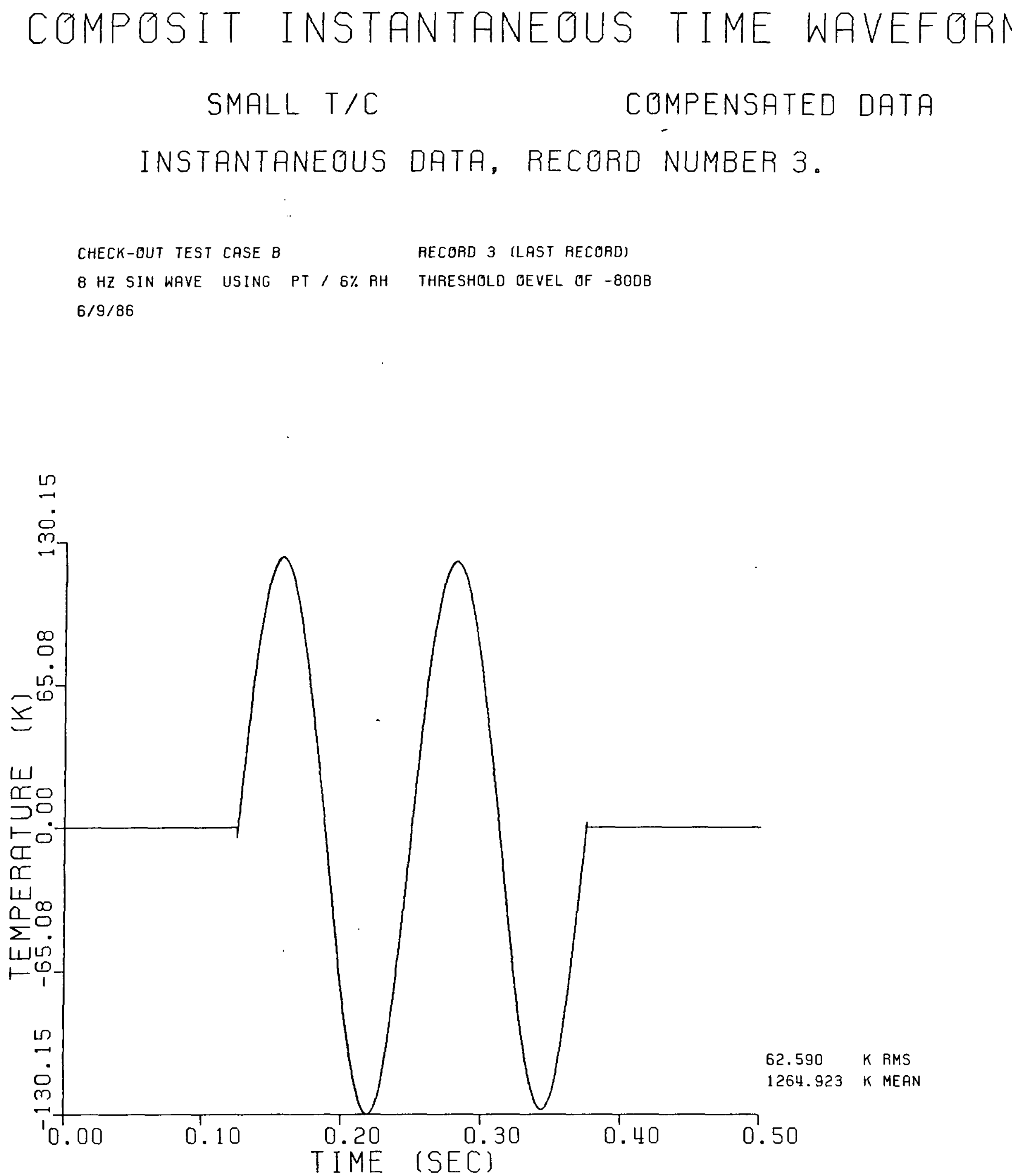
Test Case 7 illustrates the effect of the data block shifting (described in Figure 9) on the last record of data. The compensated time waveform for Record 3 was requested. Since only 3 records were generated for this test data, there was not another record to merge with Record 3. In this case the first and last quarter of the record is set to zero. This input to Test Case 7 is shown in Figure 29, and Figure 30 shows the plot generated. Please note that lines 14, 16, 17, 18, and 19 of the input description are not needed.

```
   .3048   .140589    .0508  .025999   .20828    .0635    .0381  .0077787
     .02   1266.48    .355
     1.0893784E+06
.24414E-03
     8.0       8.0      2.0
  2500.0       2.0     .125   1500.0      0.5      -.2     50.0       1.0
     -.4
  1  3
2048
  2000.0
  3    0    0    2    1    4    2    0    1    2    1    1
  0
  0
  3
 -80
CHECK-OUT TEST CASE B                RECORD 3 (LAST RECORD)
8 HZ SIN WAVE  USING  PT / 6% RH     THRESHOLD LEVEL OF -80DB
6/6/86
```

FD 316933

Figure 29. Input for Test Case 7

7197C



FD 316934

Figure 30. Test Case 7 Plot 1

7197C

PROGRAM EXECUTION ON THE TSS OPERATING SYSTEM USING THE DISSPLA GRAPHICS PACKAGE

Set-Up

The first time user of GASTEMP must issue the following commands in order to access the object code for GASTEMP. These commands need only be issued once. From then on, the user has access to the required code.

```
SHARE D1100LI$,UUDALIB,D1100LI$
SHARE GPMLIB1$,UUDALIB,GPMLIB1$
SHARE DISSPLA,XXISSCO,DISSPLA
SHARE MIG.GASLIB,IOPRATT,MIG.GASLIB
SHARE DDUM,IOPRATT,USERLIB
PROCTRAN DDUM,GASTEMP,TAPEREAD,TAPEINFO
```

Digitizing Analog Data

Only three channels of data should exist on the digitized tape. These are 1) small diameter thermocouple ac, 2) large diameter thermocouple ac, and 3) dc reading of the large or small diameter thermocouple. If more than these three channels exist on the digital tape, an error will occur while reading the tape.

Four items of information concerning digitizing analog data will be needed in the user input file. These items are:

1) Digitizing rate (delta T)
2) Preston amplifier gain
3) Output/input ratio
4) DC offset.

Items 1 and 2 are information the user should give the technician digitizing the data. If the user does not know items 3 and 4, they can be found by using TAPEINFO if the required information was included on the analog tape and digitized along with the real data. (See TAPEINFO below.)

Execution of GASTEMP

The execution of GASTEMP is a two part process. The first run sets up the compensation spectrum, scaled data, and FFTs necessary for all plots to be generated. This run takes a large amount of connect time and the user may wish to run it as a batch job. Due to system restrictions, no plots should be requested for a first time run if it is submitted as a batch job. Two files must exist on the user's ID before GASTEMP may be executed for the first time. These files are: 1) digitized data in volts, and 2) user input file. Digitized data is a migrated dataset called MIG.NAME.DATA where NAME is user supplied to specify this particular test case. The user input file must be named NAME.INPUT and should not be a migrated dataset.

To create plots, GASTEMP should be run interactively on a Tektronics or a Solanar terminal. If hardcopies are desired, make sure the terminal is hooked up to a hardcopy device. Required input for these runs consists of the user input file, NAME.INPUT, along with migrated files created on the first time run containing compensation spectrum, scaled data, and FFTs.

The following command will initiate execution of the program: GASTEMP NAME = name, STATUS = status. For a first time run use STATUS = new, and once the generated files exist, STATUS = old may be used. Files that are created within the first execution of GASTEMP, and used for successive runs to generate plots, are named as follows:

MIG.NAME.COMP —	compensation spectrum (computed only if a measured gamma is found)
MIG.NAME.DATS —	scaled data for small diameter thermocouples
MIG.NAME.DATL —	scaled data for large diameter thermocouples
MIG.NAME.FFTS —	FFT for small diameter thermocouples
MIG.NAME.FFTL —	FFT for large diameter thermocouples.

These files are created as temporary datasets and migrated in order to save storage.

Creating MIG.NAME.DATA

A procedure exists that will create MIG.NAME.DATA for the user. The user must know the tape number that contains the digitized data, the digital reading number of the desired event, and a unique name that describes this particular test case. This name should be simple, consisting of one to eight alphanumeric characters beginning with an alphabetic. The command that begins execution of this procedure is TAPEREAD VOL=tape#,RDG=rdg#,NAME=name. The program will prompt the user for the number of records to be processed beginning with the first data point of RDG. The amount of connect time required for this procedure varies according to if the tape is already mounted, and the number of records the user wishes to process. If the user wants to submit this job to background, it may be done after entering the desired number of records. (See Submitting to Background.) The program reads in the desired amount of data, converts this data to volts, and writes it to a temporary file, NAME.DATA. This file is then migrated under the name MIG.NAME.DATA.

Creating NAME.INPUT

NAME.INPUT is the user input describing the options for a particular run. Two sample input files exist on IOPRATT called SAMPLE.INNEW and SAMPLE.INOLD. These files may be copied to the user's ID and edited for his/her own use. These files will help the user to fit data in the correct columns and to pick appropriate options for a particular type of run. SAMPLE.INNEW is an input file for a first time run to be submitted to batch, and requesting no plots. SAMPLE.INOLD is a sample of an input file for plot generating runs, showing several of the available options.

See Description of the Computer Program for a complete input description.

TAPEINFO

TAPEINFO is a routine that will give the user information to calculate two input variables, output/input ratio, and dc offset. The command is TAPEINFO VOL=tape#,RDG=rdg#. The user is prompted for block size (see Input Description) and the variable for which information is sought.

For the output/input ratio, a reading of known input is included on the FM analog tape and digitized along with the raw data. The program calculates the rms voltage of the digitized reading and prints it on the terminal screen for the user. For example, an input of 400 mv rms is included on the analog tape and digitized. The output value from the program is 2.9446 volts rms. The output/input ratio is then calculated by dividing 0.4 volt rms into 2.9446 volts rms.

For the dc offset calculation, an input of zero volts is digitized and TAPEINFO finds the average over an entire data block. This output value is the dc offset and should be close to zero.

Submitting to Background

A job that is begun interactively can be submitted to finish running background by issuing the command BACK DSNAME. DSNAME is the name of a file on the user's ID which will tell the system how to finish the job. For TAPEREAD (and GASTEMP if the user desires), the only commands necessary are 'go' and 'logoff'. A sample file is stored on IOPRATT as SAMPLE.GOBACK. Once the user has issued the command BACK, if he/she interactively uses any of the files needed by the background job, an error will occur. To be safe, the user should logoff after submitting the job to background.

Running GASTEMP as a Batch Job

When GASTEMP is run with a status of 'new', the user may want to run it as a batch job due to the time required, or submit it to background once execution has begun (see above section). If a total batch job is desired, the user enters the command EXECUTE DSNAME. In this case, DSNAME is a file telling the machine which commands to execute, beginning with 'logon' and ending with 'logoff'. Since, in this case, the procdef GASTEMP controls all desired actions, the only other command needed in DSNAME is GASTEMP NAME = name,STATUS = new. A sample dataset for submitting to batch exists on IOPRATT under the name SAMPLE.BATCH.

Warning

The user should be aware of the correlation between the procdef parameter STATUS, and the input variable, IFLAGS(1). The input variable IFLAGS(1) determines order of program execution. The choices are:

1. The program calculates a measured value of gamma and uses that value to find the compensation spectrum (long run with IFLAGS(1) = 1).

2. The user inputs the value of gamma to be used in the calculation of the compensation spectrum (some time saved with IFLAGS(1) = 2).

3. The compensation spectrum, scaled data, and FFTs already exist in migrated files and are used to create all plots (short run with IFLAGS(1) = 3).

The procdef parameter, STATUS, controls only the allocation of datasets that contain or will contain the compensation spectrum, scaled data, and FFTs. With a status of 'new', the program assumes the files do not exist. They are created with execution of the program and migrated upon termination of the run. If a status of 'old' is specified, the program assumes the files already exist as migrated datasets. The files are restored to temporary storage and used to create plots. Upon termination of the program, the files are erased from temporary storage but still exist as migrated datasets.

The proper way to combine these two parameters is as follows: A status of 'new' should be used when the files are being created and no migrated files exist under the NAME specified (except for the raw data, MIG.NAME.DATA). IFLAGS(1) should be 1 in order to use a status of 'new'. When a status of 'old' is specified, scaled data and FFTs should exist as migrated datasets. The compensation spectrum may or may not exist depending on whether a measured gamma was found when the case was run as 'new'. IFLAGS(1) should be either two or three in order to use a status of 'old'.

If a status of 'new' is specified with an IFLAGS(1) value of three, the program will terminate with a message stating the end of the record was encountered while trying to read the compensation spectrum (file 12, dsname=MIG.NAME.COMP). The reason for this is that the procdef did not restore the migrated datasets because of the status of 'new', and the program tries to read restored files which are empty. If a status of 'old' is specified with an IFLAGS(1) value of 1, the following things may happen.

- If the files do not exist as migrated datasets, messages will arise which state the files could not be restored.

- The files created within the program are erased upon its termination and all the data calculated is lost. This will cause no permanent harm, but will waste time.

There is no way for the procdef to check if an error occured within the execution of GASTEMP. Therefore if a status of 'new' is specified, the compensation spectrum, scaled data, and FFT files will still be migrated even if an error occured within execution of the program. If the program is then rerun with a status of 'new', the program will execute properly but an error may occur when trying to migrate the files, since they already exist (even though they are wrong). The files will not be migrated and will exist in temporary storage under the names NAME.COMP, NAME.DATL, etc. The user may manually correct this by erasing the migrated files MIG.NAME.____, renaming NAME.____ to MIG.NAME.____, and migrating these files. The user may also prevent this from happening by making a habit of looking at the program output, NAME.OUTPUT. This file is automatically printed for the user and stored as a permanent dataset. If the output shows an error has occured, and the program is to be rerun, make sure all datasets that were created and migrated with the original run are erased before starting over.

Order of Execution

The following order of execution is recommended for use with the program GASTEMP and related programs.

TAPEREAD VOL=tape#,RDG=rdg#,NAME=name

Execution of this program is required before any data can be processed. Execution must begin as interactive, but may be submitted to background after entering the number of records to be processed.

TAPEINFO VOL=tape#,RDG=rdg#

Execution of this program is optional. The program's only use is to supply the user with information as to the values of input variables, output/input ratio, and dc offset. If the user knows these values, this program need not be run.

NAME.INPUT

Creation of the input dataset should be accomplished at this time. Please make use of the sample datasets that exist on IOPRATT.

GASTEMP NAME=name,STATUS=new

The first run of GASTEMP will create the data necessary for subsequent runs of the program. This run will take a large amount of time. The user can handle this in one of two ways.

First, the user may begin execution interactively with the above statement and then submit to background once it looks as though it is working. (See Submitting to Background, above.) Secondly, the user may submit the job as total batch. (See Running GASTEMP as a Batch Job, above.)

GASTEMP NAME=name,STATUS=old

The plotting runs of GASTEMP should be run interactively on a graphics terminal such as the Tektronics or the Solanar. The plotting package, DISSPLA, is used, and at this time, DISSPLA is restricted to interactive runs on the above terminals.

If a hardcopy of a plot is desired, the run must be on a Tektronics that is hooked to a hardcopy device. Note: To continue execution once a plot has been displayed on the screen, the user must hit 'break' followed by a 'return'. Watch the computing newsletter to keep abreast of DISSPLA changes.

APPENDIX
SUBROUTINE FLOWCHARTS

The following subroutine flowcharts are shown in Figures 31 through 34 in this appendix:

- POWER
- INTERP
- PSDFN
- CFSN

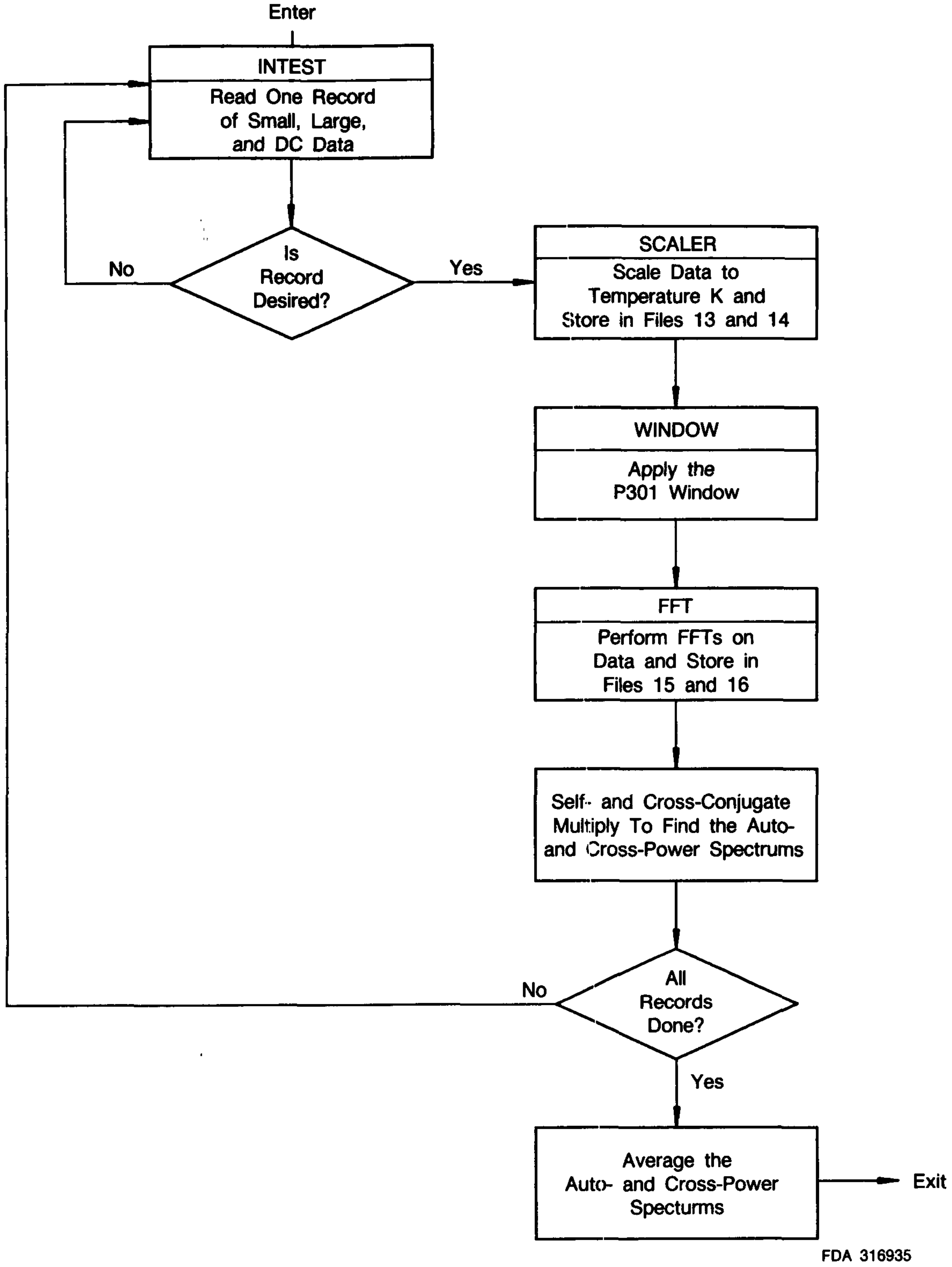


Figure 31. Subroutine POWER

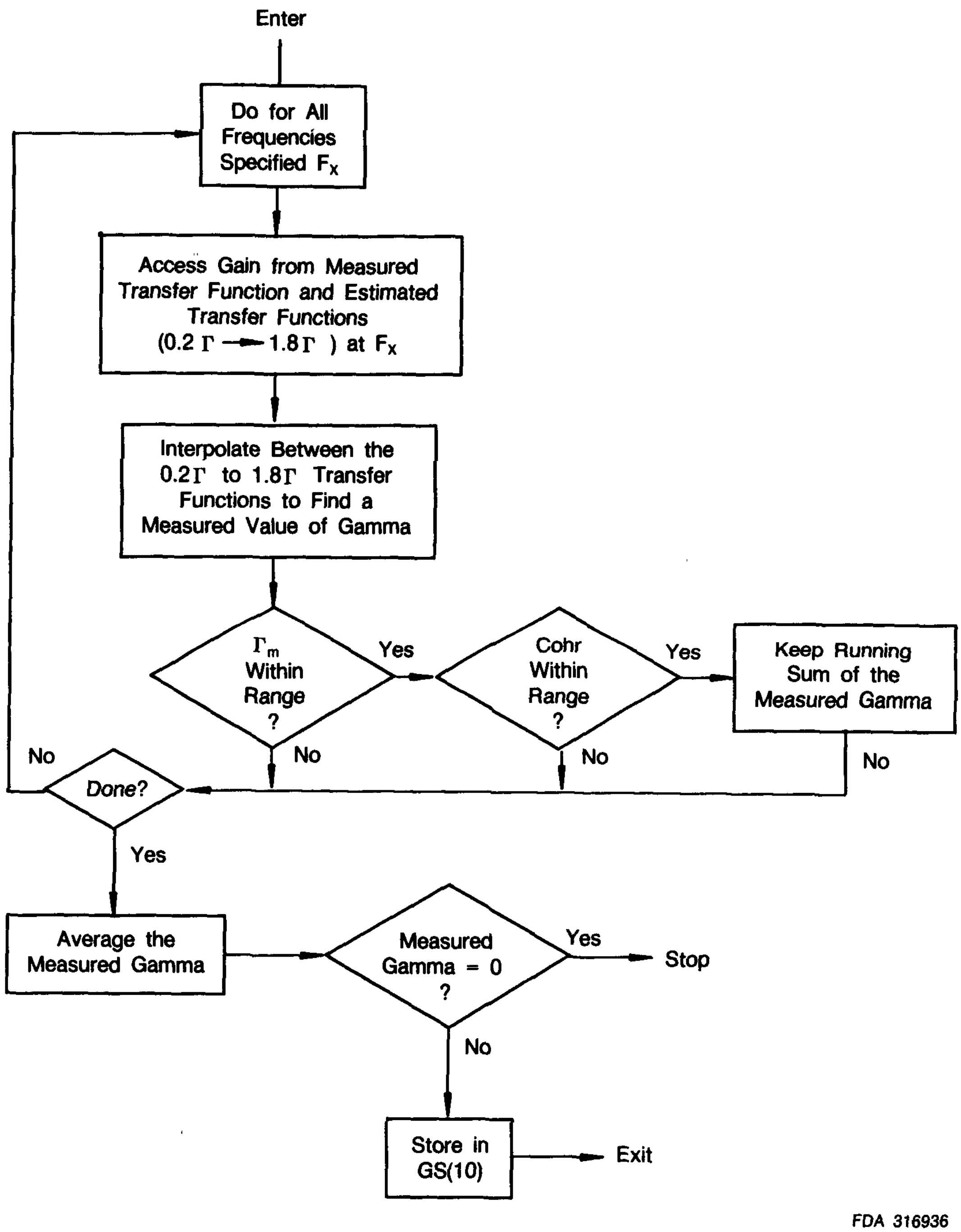


Figure 32. Subroutine INTERP

7219C

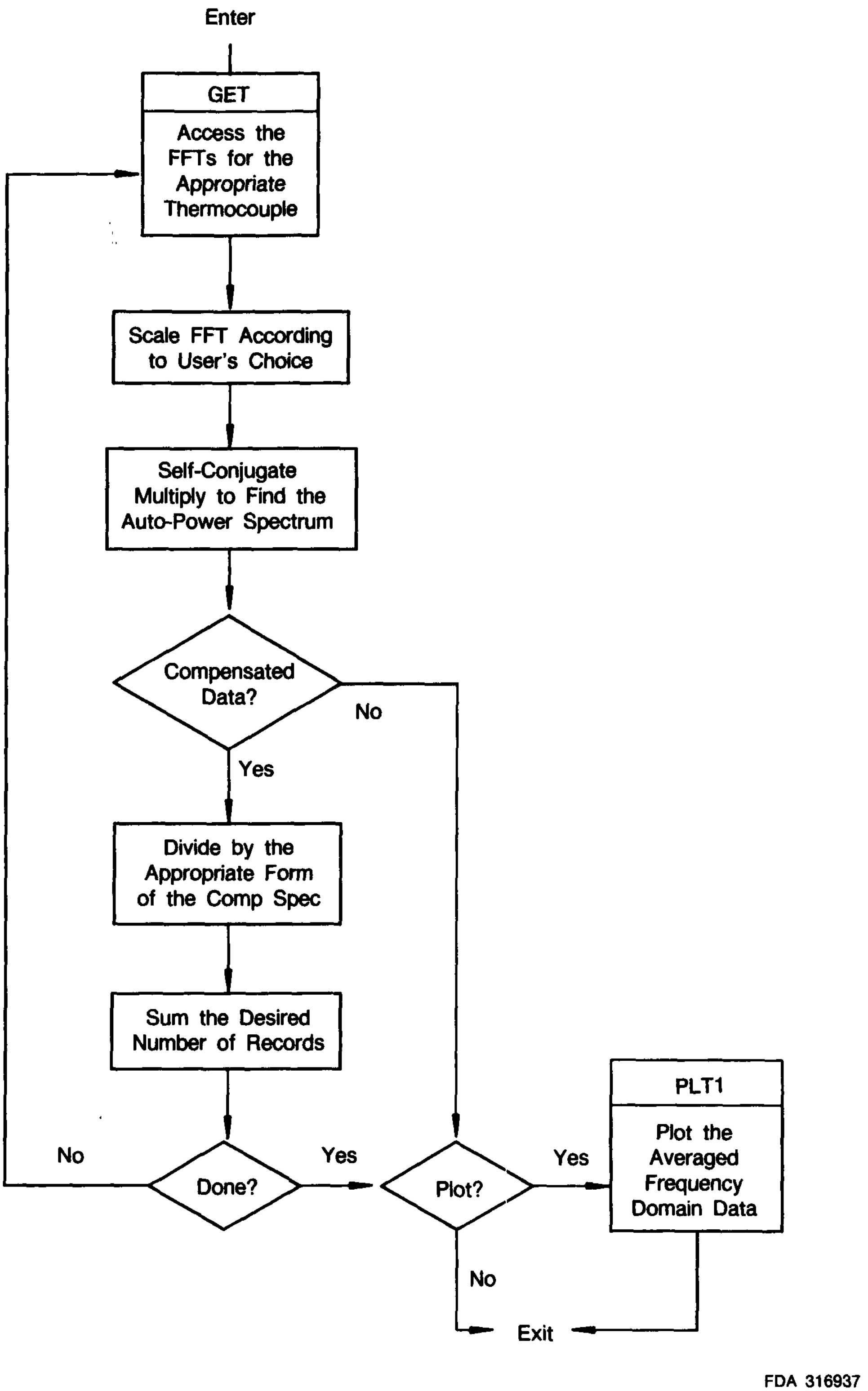


Figure 33. Subroutine PSDFN

7219C

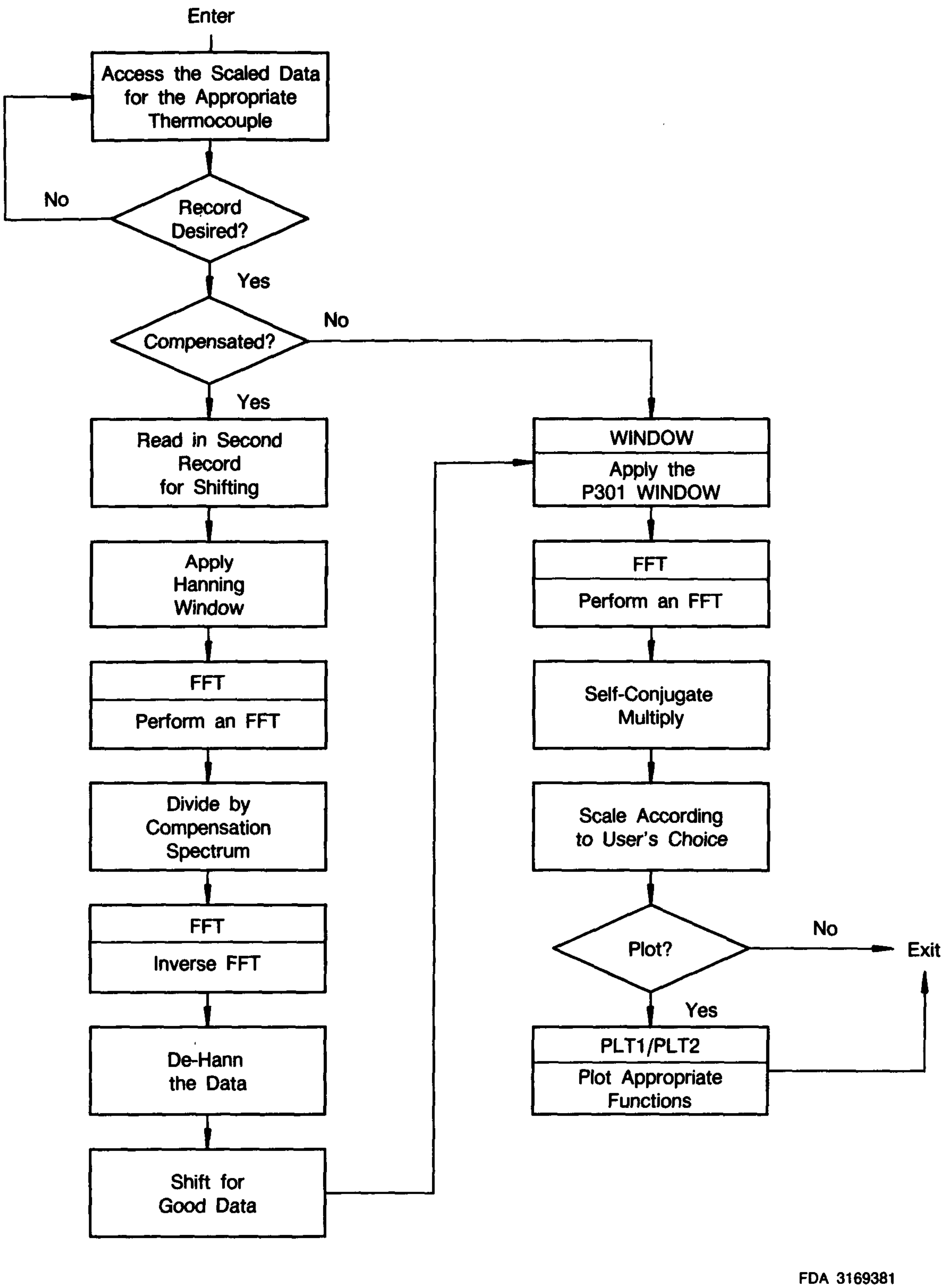


Figure 34. Subroutine CSFN

7219C

1. Report No. [redacted] CR-179513-vol 2

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle
FURTHER DEVELOPMENT OF THE DYNAMIC GAS TEMPERATURE MEASUREMENT SYSTEM
Volume II — Computer Program User's Manual

5. Report Date
August 1986

6. Performing Organization Code

7. Author(s)
Dana R. Stocks

8. Performing Organization Report No.
P&W/GPD FR-19381

9. Performing Organization Name and Address
Pratt & Whitney
Government Products Division
P.O. Box 109600, West Palm Beach, FL 33410-9600

10. Work Unit No.

11. Contract or Grant No.
NAS3-24228

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
NASA-Lewis Research Center
Cleveland, Ohio 44135

13. Type of Report and Period Covered
Draft Final Report

14. Sponsoring Agency Code

15. Supplementary Notes
Program Manager, Denny L. Elmore, Pratt & Whitney
Project Manager, G. C. Fralick, Instrumentation and Fluid Mechanics Division, NASA-Lewis Research Center, Cleveland, Ohio 44135

16. Abstract

The Dynamic Gas Temperature Measurement System compensation software accepts digitized data from two different diameter thermocouples and computes a compensated frequency response spectrum for one of the thermocouples. Detailed discussions of the physical system, analytical model, and computer software are presented in this volume and in Volume I of this report under Task III. Computer program software restrictions and test cases are also presented. Compensated and uncompensated data may be presented in either the time or frequency domain. Time domain data are presented as instantaneous temperature vs time. Frequency domain data may be presented in several forms such as power spectral density vs frequency.

17. Key Words (Suggested by Author(s))

Date for general release Feb. 1989

19. Security Classif. (of this report)
Unclassified

20. Security Classif. (of this page)
Unclassified

21. No. of Pages
136

22. Price*

[redacted]