A PERSONAL COMPUTER VERSION OF PHAS20, FOR THE SIMULTANEOUS MULTIPLE REGRESSION OF THERMOCHEMICAL DATA

By James W. Ball[1], George A. Parks[2], John L. Haas, Jr.[3], and D. Kirk Nordstrom[1]

[1]U.S. Geological Survey, Menlo Park, California

[2]Department of Applied Earth Sciences, Stanford University, Stanford, California

[3]U.S. Geological Survey, Reston, Virginia

U.S. GEOLOGICAL SURVEY

Open-File Report 88-489-A

Menlo Park, California

1988

DEPARTMENT OF THE INTERIOR

DONALD PAUL HODEL, Secretary

U.S. GEOLOGICAL SURVEY

Dallas L. Peck, Director

For additional information
write to:

Regional Hydrologist
U.S. Geological Survey
345 Middlefield Road
Menlo Park, California 94025

Copies of this report can
be purchased from:

U.S. Geological Survey
Books and Open-File Reports Section
Federal Center, Building 810
Box 25425,
Denver, Colorado 80225

The source code listed in Attachment G is available as Open-File Report 88-489-B, and can also be purchased from the U.S. Geological Survey, Books and Open-File Reports Section.

CONTENTS

CONVERSION FACTORS

For the use of readers who prefer to use inch-pound units, rather than the metric (International System) units used in this report, the following conversion factors may be used:

Multiply metric unit	By	To obtain inch-pound unit
Calorie (cal)	0.003968	British thermal unit (Btu)
Joule (J)	0.01660	British thermal unit (Btu)

Temperature in degrees Fahrenheit (Kelvins, K) as follows:

$$°F = 1.8 \times (K - 273.15) + 32$$

Temperature in degrees Fahrenheit (°C) as follows:

$$°F = 1.8 \times °C + 32$$

A PERSONAL COMPUTER VERSION OF PHAS20, FOR THE SIMULTANEOUS MULTIPLE REGRESSION OF THERMOCHEMICAL DATA

By James W. Ball, George A. Parks, John L. Haas, Jr., and D. Kirk Nordstrom

ABSTRACT

The previously developed computer program PHAS20 has been modified to execute on a Prime[1] 850 minicomputer and on an IBM or compatible personal microcomputer. PHAS20 (20 PHASes) performs simultaneous multiple regression for developing the functional relations among experimentally determined thermodynamic quantities as a function of temperature for a group of chemically related species. The thermodynamic quantities included are heat capacity, entropy, enthalpy, Gibbs free energy, equilibrium constant, electrochemical potential, and relative heat content. PHAS20PC (PHAS20 for Personal Computers) has about one-eighth the data-handling capacity of the mainframe and minicomputer versions of PHAS20, but will perform simultaneous multiple regressions on a significantly large number of input parameters. Without further modification to the FORTRAN source code, PHAS20PC will accept as many as 100 sets of data, 1,500 independent observations, and 20 chemically related species. Each data set can contain a measured thermodynamic property involving a maximum of 16 phases. The required microcomputer configuration is an IBM PC or compatible with at least 512,000 bytes of random access memory, at least one million bytes of disk storage, and a printer. For recompilation of the source code, Microsoft FORTRAN, Version 4.00 or its equivalent also is required.

---

[1]The use of trade, brand or product names in this report is for identification purposes only and does not constitute endorsement by the U.S. Geological Survey.

INTRODUCTION

The development of coherent sets of accurate, internally consistent thermodynamic data for the evaluation of the geochemistry of natural systems is a formidable task. Fortunately, the inherently numerical nature of this undertaking lends itself readily to the use of high-speed computers to assist in the effort. The computer program PHAS20 (20 PHASes) has been used for the evaluation of thermodynamic data for at least 13 years (Haas, 1974). The generation of one popular tabulation of mineral thermodynamic data was facilitated by use of PHAS20 (Robie and others, 1978); recent refinements of mineral thermodynamic data have used PHAS20 (Haas and others, 1981; Hemingway and others, 1982; Robinson and others, 1982); and Nordstrom and Jenne (1977) used PHAS20 in an evaluation of the thermodynamic properties of fluorite and its solubility. The original version was developed on the U. S. Geological Survey's IBM 360 and 370 computers in the early 1970's. PHAS20 has since been installed on the U.S. Geological Survey's Honeywell MULTICS computer (1976), and with only minor changes on the U.S. Geological Survey's VAX/VMS 11/780 computer (1982). This report announces the installation of PHAS20 on the U.S. Geological Survey's Prime 850 minicomputer, and describes its adaptation to run on a stand-alone personal microcomputer.

Documentation for PHAS20 consists of a user's manual (Haas, 1974), and a publication (Haas and Fisher, 1976). This report describes implementation of a recent version (PHAS20PC) of the program on an IBM PC/XT, AT or compatible with at least 512,000 bytes (512 KB) of random access memory (RAM) and at least one million bytes (1 MB) of disk storage. This report also describes an updated description of the underlying heat capacity equation and its parameters, a volume equation for pressure corrections, an updated list of variables, and some additions to the list of error messages found in Haas (1974), including some that may be unique to the microcomputer environment.

Diskette copies of program PHAS20PC (Ball and others, 1988), at cost, may be ordered from the U.S. Geological Survey's Books and Open-File Reports Section at the address listed on Page ii of the preliminary pages of this report. Specify Open-File Report 88-489-B, diskette when ordering.

CHANGES TO THE COMPUTER CODE

The VAX/VMS version of PHAS20 was transported to the Prime 850 of the U.S. Geological Survey office at Menlo Park, California by use of a Tektronix 4050 series DC-300 cassette tape. After installation of PHAS20 on the Prime computer, the updated FORTRAN 77 source code was downloaded to an IBM PC/AT and separated into main program and individual subroutines. Each of these was compiled separately, until all the code conformed with the syntax of Microsoft FORTRAN, Version 4.00. The object code was then linked and run on a test data set, with recompilation as necessary to obtain a correct run, as demonstrated by equivalence of the output to that supplied with the source code.

There are several differences between the two FORTRAN 77 versions. Most microcomputer FORTRAN compilers do not allow areas of COMMON to contain the mixtures of character and numeric data that are found in the original code. Therefore, new COMMON blocks were created to contain all character variables.

Personal computer FORTRAN compilers typically require an explicit delimiter following a character string in a FORMAT statement. Commas therefore were inserted where necessary in FORMAT statements. Character data cannot be stored in REAL variables. Accordingly, all REAL variables being used to store character data were redeclared as CHARACTER variables. A new subroutine, CVDATE, was written to read the personal computer's system clock and convert the parameters to a form suitable for placement in the printed output. The system clock functions are nonstandard, and therefore may not be available on other microcomputer FORTRAN versions. The dimensions of all the major variables were reduced by a factor of about 8, in order to reduce the executable code to a packed size (110 KB) that could be copied onto a 360 KB diskette. This change reduces the number of phases that can be refined simultaneously from 150 to 20, the number of independent observations from 8499 to 1500, and the number of data sets from 999 to 100. Since program RECKON was not available to us, all WRITE statements to the file designated as the RECKON input file (unit 7) have been commented out.

TEMPERATURE AND PRESSURE FUNCTIONS AND PARAMETERS

For solids and gases, the original PHAS20 and the user's manual (Haas, 1974) assume an extension of the Maier-Kelley function for the temperature dependence of heat capacity (C_p) and corresponding functions for entropy (S), enthalpy (H), and Gibbs free energy (G). For aqueous ions, an alternate function was given by Haas and Fisher (1976) and Haas (1974). No modification was made in the alternate function and it will not be discussed in this report. For solids and gases, these functions are (Haas and Fisher, 1976):

$$C_p = a + 2bT + cT^{-2} + fT^2 + gT^{-0.5} \quad (1)$$

$$S = a\ln(T) + 2bT - 0.5cT^{-2} + 0.5fT^2 - 2gT^{-0.5} + e \quad (2)$$

$$H = aT + bT^2 - cT^{-1} + 0.333fT^3 + 2gT^{0.5} + d \quad (3)$$

$$G = aT(1-\ln(T)) - bT^2 - 0.5cT^{-1} + d - eT - 0.167fT^3 + 4gT^{0.5}, \quad (4)$$

where a, b, c, d, e, f and g are constants and T is temperature in Kelvins.

The mainframe version of the program currently being distributed is different from the original code in the following ways:

(1) The mathematical model for the temperature dependence of the heat capacity has been expanded, and:

$$C_p = a_1T^{-3} + a_2T^{-2} + a_3T^{-1} + a_4T^{-0.5} + a_5 + a_6T + a_7T^2 + a_8T^3 + C_M \quad (5)$$

where C_M is the magnetic contribution to the heat capacity and is given by:

$$C_M = a_{12} \sum_{k=1}^{n} \frac{\tau^{j(2k-1)}}{2k-1} \qquad \text{for } \tau = T/T_c \leq 1 \quad (6)$$

and

$$C_M = a_{13} \sum_{k=1}^{n} \frac{\tau^{j(2k-1)}}{2k-1} \qquad \text{for } \tau = T/T_c > 1. \tag{7}$$

The term j is commonly set to +3 for $T<T_c$ and to -5 for $T>T_c$. The term n should be 7 or greater. The critical temperature at the reference pressure is set to a_{11}:

$$T_c = a_{11}. \tag{8}$$

The new and corresponding original parameters are:

new parameter name		original parameter name	term in Cp equation (1)
a_1	=	none	X T^{-3}
a_2	=	c	X T^{-2}
a_3	=	none	X T^{-1}
a_4	=	g	X $T^{-0.5}$
a_5	=	a	constant
a_6	=	2b	X T_2
a_7	=	f	X T^3
a_8	=	none	X T^3
a_9	=	d	integration constant, enthalpy
a_{10}	=	e	integration constant, entropy
a_{11}	=	none	T_c at the reference pressure, P_0
a_{12}	=	none	refer to equation (6) for C_M, the contribution due to a magnetic anomaly at T_c
a_{13}	=	none	refer to equation (7) for C_M.

(2) Pressure dependence has been added, including an expression relating pressure to volume and temperature through parameters a_{14} to a_{18}:

$$V = a_{14} + a_{15}T + a_{16}\exp(-T/300) + a_{17}P + a_{18}\exp(-P/35000) \tag{9}$$

We have not taken advantage of these extensions in PHAS20PC; they are mentioned only because some variables they introduce must be dealt with in the input file.

RANGE OF APPLICATION

PHAS20 was originally designed for use with temperatures above about 200 K. The first four terms in equation 5 must be used with care to avoid serious computational error and overflow or zero-divide computer errors when lower temperatures are used. The program can be used with temperatures to zero Kelvins if the heat capacity function invoked includes no terms in which temperature has a negative exponent.

Ignoring nuclear contributions, heat capacities at temperatures from zero to 50 K are described by the function:

$$C_p \approx C_v = a_5 T + a_8 T^3 \ , \qquad (10)$$

in which the a_5 and a_8 terms correspond to lattice and electronic contributions (Touloukian and Buyco, 1970). We have used PHAS20 successfully to verify estimates of a_5 and a_8 made by others using data in the range 0 to 50 K by allowing only a_5 and a_8 to vary during regression.

PROGRAM OPERATION FOR MULTIPLE REGRESSION OF THERMOCHEMICAL DATA

PHAS20PC is operated in much the same way as its minicomputer and mainframe counterparts, except that input and output must be handled in a manner compatible with a personal microcomputer disk operating system. To facilitate handling of the input and output, a DOS batch file has been created which handles all necessary file operations, including queueing of output files to the default print device. The program has been executed correctly on an IBM PC/XT, AT or compatible with 640 KB of RAM, one flexible disk drive, and a 20 MB hard disk.

The packed executable code occupies about 110 KB of disk storage. The test case provided with the program was solved in about 30 seconds, including output of the results to disk files. Execution time will vary significantly as a function of the complexity of the input data set and the options specified. The calculations done on the PC are virtually identical to those done by the same code on the U.S. Geological Survey's Prime 850 minicomputer.

Input

User-supplied input of thermodynamic data and individual run options is complex, and is best prepared using an interactive program (PH20INPT, An interactive program for the preparation of input data for thermodynamic fitting program PHAS20) written for that purpose (J. W. Ball, U.S. Geological Survey, and G. A. Parks, Stanford University, written commun., 1987).

Input Variables

The input file consists of a series of lines representing FORTRAN card images. Input variables are listed and defined in groups corresponding to the line in which they appear in the input file.

Line: Format:
Variable Definition and comments

Problem description

Line: 1 Format: A8
DATE Date or any string identifier.

Line: 2 Format: I5

NREG Number of separate problems to be entered with this job.

Line: 3 Format: 20A4

TITLE A descriptive title for this job.

Line: 4 Format: 5I5,2A8,I5

IDO Tasks requested of PHAS20.
IDO=1, regression plus error plots;
IDO=2, regression only;
IDO=3, plots only.

NC Maximum number of cycles of refinement in regression, commonly NC=3.

IW Obsolete flag to set error or weighting style.
Automatically set to 0.

IL Output control.
IL=0, print final result only;
IL=1, print results after each cycle of refinement.

IFMOUT Format control for card punch.

UNIT Select thermodynamic units.
String="calories", "CALORIES", "joules " or "JOULES ".

RECKON Prepare output for program RECKON.
String "reckon ", "RECKON ", "noreckon" or "NORECKON".
Program as modified does not create RECKON output files.

ITEK Output files 11 and 15 are generated for use on a plot facility. File 11 contains:
"JSET;X(1,I),X(2,I),X(3,I),X(4,I),YO(I),YC"
JSET is the data set reference; YC is the calculated Y.
File 15 contains:
"JSET;X(1,I),DELH"
DELH is the Third-Law enthalpy of reaction as calculated during error plotting. The remainder of the variables written into files 11 and 15 are described later, in the section entitled Data Entry.
Output file 16 contains the data set reference, the number of observations, the average Third-Law enthalpy of reaction, and the mean scatter of the enthalpies of reaction about the average for each set. Semicolon and commas are included in the ITEK output files for potential editing purposes.
ITEK = 0 produces no file;
ITEK = 1 produces file 11 during regression, and files 15 and 16 during error plotting.

Line: 5 Format: 4I5

LISTP Number of distinct chemical species in problem.
Maximum=20.

ICY Maximum number of fitting parameters permitted to drop during step-backward elimination if determined to be nonsignificant.

NHOLD Number of fitting parameters to be retained or forced during step backward elimination, even if they are determined to be nonsignificant.

If NHOLD>0, an IHOLD line is required in trial parameter section to list the specific parameters to be forced.

To retain all parameters in regression, regardless of significance, set both ICY, NHOLD=0.

IDAMP — Set to 1 to cause changes in the damping factor (successive undamping) of a refinement during successive cycles. Set to 0 if damping is not used or if the same damping factor is to be used throughout the refinement. The option to use damping factors was inserted in the ORGLS subroutine in December, 1986. KI(I) is varied from 1 through 9 and causes the calculated change in a parameter to be multiplied by the following factors, respectively:
1.0, 0.63, 0.40, 0.25, 0.16, 0.10, 0.063, 0.04, and 0.025

Line: 6 — Format: 10A8

PNAME — Contains one 8-character identifier-name for each of the LISTP species in the problem. Continue on additional lines if LISTP>10.

Line: 7 — Format: 40I2

ION — Species type identifiers. One integer for each of the LISTP species, listed in the same sequence as in PNAME.
ION= 1 for reference elemental species;
ION=-1 for water and all aqueous species;
ION= 0 for all other types of species.
Use additional lines if LISTP>40.

Line: 8 — Format: 16(L1,2I2)

LAMBDA — LAMBDA=T if the system has a lambda point;
=F if the system has no lambda point.

JLOW — =Exponent in the empirical function for the contribution to the heat capacity of a phase with a lambda anomaly.

JHIGH — =Exponent in the empirical function for the contribution to the heat capacity of a phase with a lambda anomaly.

JLOW, JHIGH=0 if system has no lambda point.
Include one set of LAMBDA, JLOW, JHIGH for each LISTP species. Use additional lines if LISTP>16.

Line: 9 — Format: I5

NSETS — Number of datasets to follow. Maximum=100.

Data set description

For each of the NSETS sets of data:

Line: D1 — Format: 10A8

REF — String. Alphanumeric description of source and type of data.

Line: D2 — Format: 16I5

NPHASE — Number of phases or species involved in this dataset (maximum=20). For relative heat content, $H_{T_2} - H_{T_1}$, set NPHASE to 2.

IKOUNT — Number of observations in this dataset. Sum of all IKOUNT for all sets of data cannot exceed 1500.

IGO — Type of data.

If	IGO=	
C_p	1	(heat capacity)
S	2	(entropy)
H	3	(enthalpy)
G	4	(Gibbs free energy)
$Log_{10}K$	5	(equilibrium constant)
E	6	(electrochemical potential)
$H_{T_2} - H_{T_1}$	7	(relative heat content)
Special cases as programmed in UNIQUE by the user.	8 through 20	

ITFACT — Temperature conversion flag.
If temperatures entered are Kelvins, ITFACT=1;
If temperature conversion is needed, ITFACT=0.
Conversion factor, to be added to observed temperature, is entered with data.

IPARA — Data conversion flag. Default units are joules or calories (selected by UNIT), or volts (for E). Equilibrium constants are expected to be based on molal activities and atmosphere pressures. If data are entered in these units, IPARA=1. If conversion is needed, set IPARA=0 and enter factors with data.

ISIG — Type-of-error flag.
If experimental errors are relative, ISIG=0;
If experimental errors are absolute, ISIG=1.

I4X — Data-entry-format flag. In default mode, I4X=0, and you can enter two dependent variables in the first and last columns of the data line below. If you are fitting volumetric data, P-T equilibria, or using subroutine UNIQUE to incorporate non-standard constraints, two more dependent variables can be added by setting I4X=1.

For each of the NPHASE phases or species involved in this dataset, enter the following lines:

Line: D3 — Format: A8,2X,D10.3,I5,A5,2I5

ANAME — 8-character string identifier for one of the NPHASE phases or species. ANAME must be chosen from the list entered in list PNAME and entered exactly as entered there, character for character, space for space, case for case.

COEF — Either (1) the stoichiometric coefficient of species ANAME in the reaction or phase change (positive for products, negative for reactants) or (2) the number of moles upon which the measurement was made.

ISPECL — For most data, set ISPECL = 0. Exceptions:

1. Volume data, types 24, 25, 26, and 27. Some data in the literature are presented as shown above. These alternative data will be fit properly if ISPECL = -1.
2. Solution calorimetry where the solid is dropped from one temperature (T_1) to a final dissolving temperature (T_2). Set ISPECL = -1 for the solid.
3. UNIQUE as programmed by the user may require nonzero values for ISPECL.

ANOM — String. Lambda point flag.
Set ANOM="nolam" or "NOLAM" to ignore. Set to any other string or leave blank to fit lambda anomaly.

ISTATE — Type-of-species flag.
ISTATE= 1 for elements in their reference state as they exist at 298.15 K and 1.01 bar, where the standard free energy of formation is 0.0;
ISTATE=-1 for water and all aqueous species;
ISTATE= 0 for elements at temperatures higher than the reference temperature and for all other types of species.

NINVER — Number of inversions, equal to the number of <u>lower temperature</u> modifications of the species to be considered between the temperature of the observation and the reference temperature. The maximum NINVER allowed is 4. <u>Important!</u> If NINVER is zero, no more lines are needed for this species. If this is the last species in the set, skip to DATA ENTRY. If NINVER is not zero two more lines are needed, as follows:

Line: INV1 — Format: 4D12.5

TINV — One to four inversion temperatures, in Kelvins, at which PHAS20 should assume Gibbs free energy of transformation is zero.

Line: INV2 — Format: I5,I3,5A8

INSTAT — Flag; if ISTATE=1, INSTAT=1, else INSTAT=0.

INVSC — A change in stoichiometry sometimes accompanies a phase change. When this happens, INVSC contains the index of the stoichiometry change stored in the vector STCOEF(10) in BLOCK DATA. The reaction is STCOEFF(INVSC)*PHASE1=PHASE2, e.g. $2FeCl_3(s)=Fe_2Cl_6(\ell)$, for which STCOEFF(INVSC)=2.

PHINV — List the sequence of phases, beginning with the phase stable at 298.15 K and continuing with increasing temperature up to and including ANAME, using the same names as recorded in PNAME.

<u>Data entry</u>

Enter one line for each of the IKOUNT observations in this dataset.

Line: DATA(IKOUNT) Format: 6D12.5

X(1,NO) — Temperature of observation, must be entered in Kelvins or degrees Celsius.

TFACT — Additive factor to convert degrees Celsius to Kelvins. TFACT=0.0D+00 if temperature entered in Kelvins, =273.15D+00 if temperature entered in degrees Celsius.

YO(NO) — Dependent variable, e.g., heat capacity or relative heat content. YO(NO) can be entered in any units.

PARA — Conversion factor to convert YO(NO) to one of the following units, by multiplication: cal or J $mol^{-1} K^{-1}$; cal or J mol^{-1}; $log_{10}K$; or volts.

SIGYO — Estimated or assigned precision or probable error of measurement YO.

X(2,NO) — Appropriate second independent variable. Enter 0.0D+00 if inappropriate. When a reference temperature is needed but was set to zero, the reference temperature 298.15 K is used. For example, for the relative heat content, X(2,NO) is the reference temperature, 298.15 K, by default, unless set otherwise.

X(3,NO), X(4,NO) — Additional independent variables for use when volume data or phase equilibria are being fit, or when subroutine UNIQUE is invoked, and I4X=1. These two values are placed on a new line.

Formats and trial parameters

Line: T1 — Format: 80I1

IFMIN — Formatting flag. Used to select format independently for each of the LISTP species. IFMIN=0 invokes 6D15.8 (formerly 6D12.5/D12.5). IFMIN=1 invokes 7A9 (formerly 7A8). Roundoff errors are eliminated with the A format.

Line: T2(a,b,..etc.) — Format: Selected by IFMIN

P — Trial values for each of the 18 parameters for each of the LISTP species, entered in the same sequence as in PNAME. When IFMIN=0, enter 6 sets of parameter estimates on each of three lines for each species. All P may be set to zero. If a parameter is to be fixed, the corresponding P may be set to a predetermined value or ignored by setting to zero.

Line: T3(a,b,..etc.) — Format: 80I1

KI — Flag determining whether or not corresponding parameter is fixed during regression. Enter one KI for each P. If KI=0, parameter is fixed. If KI≥1, parameter is varied. To select the Haas-Fisher (1976) heat capacity function for a reference element, for example, this line should read "010111000100000000" to set each of the 18 parameters. For large data sets, sets using UNIQUE, or sets fitting volume and high-pressure phase equilibria, damping may be needed to avoid singularity. Values of KI = 1 to KI = 9 will reduce the calculated change in the corresponding parameter P by factors of 1.0, 0.63, 0.40, 0.25, 0.16, 0.10, 0.063, 0.040, or 0.025, respectively.

If NHOLD is not zero, include the following line. Omit If NHOLD is zero.

Line: T4 Format: 16I5

IHOLD Indexes of parameters P to be forced to remain in regression regardless of statistical significance. Parameters a_9, a_{10}, and a_{14} should remain in the regression. Serious error will result if they are dropped.

Plotting parameters

PHAS20 plots z=residual/(standard error), or the discrepancy between calculated and observed dependent variables, normalized to the standard error of the regression, for each dataset. The Y-axis is the z and centers on Y=0. The X-axis is temperature.

Line: PC1 Format 4I5

NHL Number of horizontal grid lines desired, less one. Conveniently set to 4.

NSBH Number of spaces between horizontal grid lines, plus one. Conveniently set to 5 or 10.

NVL Number of vertical grid lines desired, less one. If 250<=T<=2000, set NVL to 7 to cause vertical scale to correspond to 250 degrees per division.

NSBV Number of spaces between vertical grid lines, plus one. Conveniently set to 5 or 10.

Line: PC2 Format: 3E10.3

XMAX Upper bound of temperature axis.

XMIN Lower bound of temperature axis.

YMAX Upper bound of z. YMIN=-YMAX by default. Setting YMAX= 10.0E+00 allows weighted Z=±10 standard errors, retaining enough resolution near Y=0 to be useful; however, until you are familiar with the data, it is safer to set YMAX=100.0E+00 to ensure that large errors will be seen on the plot.

------ END OF INPUT FILE ------

Running PHAS20PC

User-supplied input of thermodynamic data and run options must reside in a disk file with the extension .DAT. Execution of PHAS20PC is invoked by entering:

PH20 {filename} {print_option}

The .DAT extension of {filename} must not be entered. The {print_option} parameter may be N, D, or left blank. Leave out {print_option} to convert and print the output files immediately upon completion of PHAS20PC execution. Enter N to skip conversion and printing of the output files, or D to convert the output files to printable form but defer printing until later. Batch

program PH20.BAT then issues all remaining commands necessary to complete execution of PHAS20PC and print the generated files in the manner specified by the user. PH20.BAT must have access to the specified .DAT file, PHAS20.EXE, P20PRINT.BAS and the DOS commands, including your personal computer's Advanced BASIC interpreter, PRINT.COM and COMMAND.COM. The simplest means for providing this access is to place the pathname of the location of your DOS commands in your PATH setting, prior to executing PH20, and to invoke PH20.BAT from within the directory containing all the remaining files. Use caution in invoking PH20.BAT, because all previously existing files with the extensions .OUT and .PRN are erased during execution.

Output

The following output files are created by PHAS20: {filename}.OUT, PLOTS.OUT, UNIT9.OUT, UNIT10.OUT, UNIT11.OUT, UNIT15.OUT, and UNIT16.OUT. File UNIT10.OUT is erased as soon as all input operations have been successfully completed. File {filename}.OUT is directly printable, and can be immediately routed to the default print device. Files PLOTS.OUT and UNIT9.OUT are prepared for printing by the PH20.BAT batch program, which invokes BASIC program P20PRINT to interpret the first character of each line in each output file as a carriage control character, then write the remainder of the line into files named PLOTS.PRN and UNIT9.PRN, respectively. The batch program then queues the .PRN files to the default print device using the DOS PRINT command. The contents of the PHAS20 output files are described in detail by Haas (1974), and such a description here is beyond the scope of this report. The reader is referred to Haas (1974) for a more complete discussion of the objectives of PHAS20 as a tool for fitting thermodynamic data to a model and for the interpretation of the program's output.

Attachment A contains test data used as input to PHAS20PC, and Attachments B through F contain output from PHAS20PC for this test case. This data set was deliberately oversimplified to reduce the volume of output that was produced, and several constants were fit even though the constants are not significant statistically. The least significant constant that was fit in the example was parameter 24, a_6 for As(g). A rule of thumb to use is, "Do not let the error of the constant exceed 30 percent." Several other constants are also not significant. The user should try other combinations in subsequent reruns of this data through PHAS20PC to obtain the best fit.

Error Messages

The discussion of error messages in Haas' (1974) user's manual is very helpful. In addition to those, we have encountered the following PHAS20 and FORTRAN problems and error messages:

PHAS20 errors

ZERO DIAGONAL ERROR MESSAGE

A zero diagonal error message can be caused by allowing parameter a_{10} to vary during regression when no data have been provided to constrain entropy or Gibbs free energy.

EMPTY ERROR PLOTS

Empty error plots can be caused by setting XMAX, XMIN to exclude range of data and/or setting YMAX, YMIN to exclude range of calculated errors.

EXTREME CALCULATION ERRORS

When running a trial dataset, difficulty may be encountered in reproducing known C_p equations or entropy or enthalpy data. When attempting use of unfamiliar data, the computed error, or (observed-calculated)/error, or (YO(I)-YCALC(I))/SIGYO(I), may be very large. These problems can be caused by:

1. Erroneous conversion parameters, IPARA.
2. Extrapolation beyond experimental data, e.g., attempting to estimate S_{298} when data for very low temperatures are lacking.
3. Very wide temperature gaps between datasets.
4. Outlier data. Problems can be minimized by weighting suspect data lightly, that is, assigning very large error, perhaps 100 percent, in variable SIGYO.
5. Attempts to use a_1, a_2, a_3, or a_4 with temperatures approaching 0 K.
6. Use of $H_{T_2} - H_{T_1}$ data (IGO=7) as simple heat content. The change in heat content between two temperatures represents a change in state, for example, $As(c,T_1) = As(c,T_2)$. PHAS20 treats this as a reaction between two species, so NPHASE must be set to 2.

FORTRAN errors

FAILURE TO FIND DOS VERSION

This failure can be triggered by incomplete path description. Be sure that the DOS path allows your process to access PHAS20.EXE, the FORTRAN library, and DOS, as well as your input file. Some applications programs change the DOS path and do not restore it, so rebooting or redirecting of the path may be necessary when PHAS20 is run after such an application program.

INCORRECT DSQRT ARGUMENT and/or
INCORRECT INTEGER IN INPUT

These errors can be triggered by ignoring case when entering either integer or floating point data in exponential format, that is, some FORTRAN compilers expect 4D-03, not 4d-03. These errors can also be caused by discrepancies involving IKOUNT, see next error.

INCORRECT EXPONENT IN INPUT

Causes of this error include disagreement between the number of data*sets* claimed in NSETS and the number input, or between the number of data*points* claimed in variable IKOUNT and the number actually entered in the corresponding dataset.

SUMMARY

The simultaneous multiple regression program PHAS20 has been installed on the U.S. Geological Survey's Prime 850 minicomputer system. A second version has been reduced in size and adapted to execute on an IBM or compatible microcomputer having at least 512 KB RAM, one flexible disk drive, a printer, and at least 1 MB of disk storage. The resulting computer code, PHAS20PC, performs calculations that are virtually identical to those of the mainframe/minicomputer version. Adjunct programs also are available to facilitate the construction of input data sets (PH20INPT.EXE), and to handle the housekeeping associated with execution of the program (PH20.BAT), and printing of the output (P20PRINT.BAS). All the computer codes mentioned in this report are intended for use in conjunction with the previously published computer program documentation of Haas (1974) to perform the necessary calculations on sets of thermodynamic data supplied by the user.

REFERENCES

Ball, J. W., Parks, G. A., Haas, J. L., Jr., and Nordstrom, D. K., 1988, A personal computer version of PHAS20, for the simultaneous multiple regression of thermochemical data: Part B - diskette: U.S. Geological Survey Open-File Report 88-489-B.

Haas, J. L., Jr., 1974, PHAS20, A program for simultaneous multiple regression of a mathematical model to thermochemical data: National Technical Information Service, Report AD-780 301, 158 p.

Haas, J. L., Jr., and Fisher, J. R., 1976, Simultaneous evaluation and correlation of thermodynamic data: American Journal of Science, v. 276, p. 525-545.

Haas, J. L., Jr., Robinson, G. R., Jr., and Hemingway, B. S., 1981, Thermodynamic tabulations for selected phases in the system CaO-Al_2O_3-SiO_2-H_2O at 101.325 kPa (1 atm) between 273.15 and 1800 K: Journal of Physical and Chemical Reference Data, v. 10, p. 575-669.

Hemingway, B. S., Haas, J. L., Jr., and Robie, R. A., 1982, Thermodynamic properties of selected minerals in the system Al_2O_3-CaO-SiO_2-H_2O at 298.15 K and 1 bar (10^5 pascals) pressure and at higher temperatures: U.S. Geological Survey Bulletin 1544, 70 p.

Nordstrom, D. K., and Jenne, E. A., 1977, Fluorite solubility equilibria in selected geothermal waters: Geochimica et Cosmochimica Acta, v. 41, p. 175-188.

Robie, R. A., Hemingway, B. S., and Fisher, J. R., 1978, Thermodynamic properties of minerals and related substances at 298.15 K and one atmosphere pressure and at higher temperatures: U.S. Geological Survey Bulletin 1259, 465 p.

Robinson, G. R., Haas, J. L., Jr., Schafer, C. M., and Hazleton, H. T., Jr., 1982, Thermodynamic and thermophysical properties of selected phases in the MgO-SiO_2-H_2O-CO_2, CaO-Al_2O_3-SiO_2-H_2O-CO_2, and Fe-FeO-Fe_2O_3-SiO_2 chemical systems, with special emphasis on the properties of basalts and their mineral components: U.S. Geological Survey Open-File Report 83-79, 429 p.

Touloukian, Y. S., and Buyco, E. H., 1970, Thermophysical properties of matter. Volume 4, Specific heat, metallic elements and alloys. Volume 5, Specific heat, non-metallic solids: New York, Plenum, 2568 p.

Attachment A. Example input data set for PHAS20PC

```
! Aug 22, 1985
    1
====    Arsenic (after HULTGREN AND OTHERS, 1973)    =====
    1    1    0    0    0joules  noreckon    1
    6    0    0    0
as   (r)as   (g)as2  (g)as3  (g)as4  (g)as   (l)
 1 0 0 0 0 0
F 0 0F 0 0F 0 0F 0 0F 0 0F 0 0
   16
!{arsenic, heat capacity     HULTGREN AND OTHERS (1973)
    1   13    1    1    0    1    0
as   (r)   1.000D+00    0NOLAM    1    0
  200.00D+00    0.00D+00   5.368D+00  4.1840D+00   0.010D+00    0.00D+00
  250.00D+00    0.00D+00   5.625D+00  4.1840D+00   0.010D+00    0.00D+00
  298.15D+00    0.00D+00   5.892D+00  4.1840D+00   0.010D+00    0.00D+00
  400.00D+00    0.00D+00   6.068D+00  4.1840D+00   0.010D+00    0.00D+00
  500.00D+00    0.00D+00   6.201D+00  4.1840D+00   0.010D+00    0.00D+00
  600.00D+00    0.00D+00   6.334D+00  4.1840D+00   0.010D+00    0.00D+00
  700.00D+00    0.00D+00   6.466D+00  4.1840D+00   0.010D+00    0.00D+00
  800.00D+00    0.00D+00   6.599D+00  4.1840D+00   0.010D+00    0.00D+00
  876.00D+00    0.00D+00   6.700D+00  4.1840D+00   0.010D+00    0.00D+00
  900.00D+00    0.00D+00   6.732D+00  4.1840D+00   0.010D+00    0.00D+00
 1000.00D+00    0.00D+00   6.865D+00  4.1840D+00   0.010D+00    0.00D+00
 1100.00D+00    0.00D+00   6.998D+00  4.1840D+00   0.010D+00    0.00D+00
 1200.00D+00    0.00D+00   7.130D+00  4.1840D+00   0.010D+00    0.00D+00
!{as, gas, heat capacity     HULTGREN AND OTHERS (1973)
    1   11    1    1    0    1    0
as   (g)   1.000D+00    0NOLAM    0    0
  298.15D+00    0.00D+00   4.968D+00  4.1840D+00   0.010D+00    0.00D+00
  400.00D+00    0.00D+00   4.968D+00  4.1840D+00   0.010D+00    0.00D+00
  500.00D+00    0.00D+00   4.968D+00  4.1840D+00   0.010D+00    0.00D+00
  600.00D+00    0.00D+00   4.968D+00  4.1840D+00   0.010D+00    0.00D+00
  700.00D+00    0.00D+00   4.968D+00  4.1840D+00   0.010D+00    0.00D+00
  800.00D+00    0.00D+00   4.968D+00  4.1840D+00   0.010D+00    0.00D+00
  876.00D+00    0.00D+00   4.968D+00  4.1840D+00   0.010D+00    0.00D+00
  900.00D+00    0.00D+00   4.968D+00  4.1840D+00   0.010D+00    0.00D+00
 1000.00D+00    0.00D+00   4.968D+00  4.1840D+00   0.010D+00    0.00D+00
 1100.00D+00    0.00D+00   4.969D+00  4.1840D+00   0.010D+00    0.00D+00
 1200.00D+00    0.00D+00   4.970D+00  4.1840D+00   0.010D+00    0.00D+00
!{as2, gas, heat capacity     HULTGREN AND OTHERS (1973)
    1   11    1    1    0    1    0
as2  (g)   0.500D+00    0NOLAM    0    0
  298.15D+00    0.00D+00   4.183D+00  4.1840D+00   0.010D+00    0.00D+00
  400.00D+00    0.00D+00   4.297D+00  4.1840D+00   0.010D+00    0.00D+00
  500.00D+00    0.00D+00   4.355D+00  4.1840D+00   0.010D+00    0.00D+00
  600.00D+00    0.00D+00   4.389D+00  4.1840D+00   0.010D+00    0.00D+00
  700.00D+00    0.00D+00   4.410D+00  4.1840D+00   0.010D+00    0.00D+00
  800.00D+00    0.00D+00   4.424D+00  4.1840D+00   0.010D+00    0.00D+00
  876.00D+00    0.00D+00   4.432D+00  4.1840D+00   0.010D+00    0.00D+00
  900.00D+00    0.00D+00   4.434D+00  4.1840D+00   0.010D+00    0.00D+00
 1000.00D+00    0.00D+00   4.441D+00  4.1840D+00   0.010D+00    0.00D+00
 1100.00D+00    0.00D+00   4.446D+00  4.1840D+00   0.010D+00    0.00D+00
 1200.00D+00    0.00D+00   4.450D+00  4.1840D+00   0.010D+00    0.00D+00
!{as3, gas, heat capacity     HULTGREN AND OTHERS (1973)
    1   11    1    1    0    1
as3  (g)   .33333D00    0NOLAM    0    0
  298.15D+00    0.00D+00   4.704D+00  4.1840D+00   0.010D+00    0.00D+00
  400.00D+00    0.00D+00   4.813D+00  4.1840D+00   0.010D+00    0.00D+00
  500.00D+00    0.00D+00   4.867D+00  4.1840D+00   0.010D+00    0.00D+00
  600.00D+00    0.00D+00   4.897D+00  4.1840D+00   0.010D+00    0.00D+00
  700.00D+00    0.00D+00   4.915D+00  4.1840D+00   0.010D+00    0.00D+00
  800.00D+00    0.00D+00   4.927D+00  4.1840D+00   0.010D+00    0.00D+00
  876.00D+00    0.00D+00   4.934D+00  4.1840D+00   0.010D+00    0.00D+00
  900.00D+00    0.00D+00   4.936D+00  4.1840D+00   0.010D+00    0.00D+00
 1000.00D+00    0.00D+00   4.942D+00  4.1840D+00   0.010D+00    0.00D+00
 1100.00D+00    0.00D+00   4.946D+00  4.1840D+00   0.010D+00    0.00D+00
 1200.00D+00    0.00D+00   4.950D+00  4.1840D+00   0.010D+00    0.00D+00
!{as4, gas, heat capacity     HULTGREN AND OTHERS (1973)
    1   11    1    1    0    1
as4  (g)   0.250D+00    0NOLAM    0    0
  298.15D+00    0.00D+00   4.614D+00  4.1840D+00   0.010D+00    0.00D+00
  400.00D+00    0.00D+00   4.764D+00  4.1840D+00   0.010D+00    0.00D+00
```

```
  500.00D+00    0.00D+00   4.835D+00  4.1840D+00   0.010D+00    0.00D+00
  600.00D+00    0.00D+00   4.875D+00  4.1840D+00   0.010D+00    0.00D+00
  700.00D+00    0.00D+00   4.899D+00  4.1840D+00   0.010D+00    0.00D+00
  800.00D+00    0.00D+00   4.915D+00  4.1840D+00   0.010D+00    0.00D+00
  876.00D+00    0.00D+00   4.923D+00  4.1840D+00   0.010D+00    0.00D+00
  900.00D+00    0.00D+00   4.926D+00  4.1840D+00   0.010D+00    0.00D+00
 1000.00D+00    0.00D+00   4.934D+00  4.1840D+00   0.010D+00    0.00D+00
 1100.00D+00    0.00D+00   4.940D+00  4.1840D+00   0.010D+00    0.00D+00
 1200.00D+00    0.00D+00   4.944D+00  4.1840D+00   0.010D+00    0.00D+00
!{{as, entropy     HULTGREN AND OTHERS (1973)
    1    1    2    1    0    1
as   (r)   1.000D+00    0NOLAM    1    0
  298.15D+00    0.00D+00   8.530D+00  4.1840D+00   0.010D+00    0.00D+00
!{{as, gas, entropy     HULTGREN AND OTHERS (1973)
    1    1    2    1    0    1
as   (g)   1.000D+00    0NOLAM    0    0
  298.15D+00    0.00D+00  41.611D+00  4.1840D+00   0.010D+00    0.00D+00
!{{as2, gas, entropy     HULTGREN AND OTHERS (1973)
    1    1    2    1    0    1
as2  (g)   0.500D+00    0NOLAM    0    0
  298.15D+00    0.00D+00  28.773D+00  4.1840D+00   0.010D+00    0.00D+00
!{{as3, gas, entropy     HULTGREN AND OTHERS (1973)
    1    1    2    1    0    1
as3  (g)  .33333D+00    0NOLAM    0    0
  298.15D+00    0.00D+00  24.707D+00  4.1840D+00   0.010D+00    0.00D+00
!{{as4, gas, entropy     HULTGREN AND OTHERS (1973)
    1    1    2    1    0    1
as4  (g)   0.250D+00    0NOLAM    0    0
  298.15D+00    0.00D+00  19.558D+00  4.1840D+00   0.010D+00    0.00D+00
!}}}as(c) = as(g)     HULTGREN AND OTHERS (1973)
    2    1    3    1    0    1
as   (r)  -1.000D+00    0NOLAM    1    0
as   (g)   1.000D+00    0NOLAM    0    0
  298.15D+00    0.00D+00  72.120D+03  4.1840D+00   0.550D+03    0.00D+00
!}}}as(c) = .5 as2(g)     HULTGREN AND OTHERS (1973)
    2    1    3    1    0    1
as   (r)  -1.000D+00    0NOLAM    1    0
as2  (g)   0.500D+00
  298.15D+00    0.00D+00  26.410D+03  4.1840D+00   0.850D+03    0.00D+00
!}}}as(c) = 1/3 as3(g)     HULTGREN AND OTHERS (1973)
    2    1    3    1    0    1
as   (r)  -1.000D+00    0NOLAM    1    0
as3  (g)  .33333D+00    0NOLAM    0    0
  298.15D+00    0.00D+00  20.827D+02  4.1840D+00   3.000D+03    0.00D+00
!}}}as(c) = .25 as4(g)     HULTGREN AND OTHERS (1973)
    2    1    3    1    0    1
as   (r)  -1.000D+00    0NOLAM    1    0
as4  (g)   0.250D+00    0NOLAM    0    0
  298.15D+00    0.00D+00   9.160D+03  4.1840D+00   0.100D+03    0.00D+00
!}}}}as(c) = .25 as4(g)     HULTGREN AND OTHERS (1973)
    2    1    4    1    1    1
as   (r)  -1.000D+00    0NOLAM    1    0
as4  (g)   0.250D+00    0NOLAM    0    0
  876.00D+00    0.00D+00    0.00D+00    0.00D+00   0.001D+00    0.00D+00
!}}}}}4 as(c) = as4(g)     HULTGREN AND OTHERS (1973)
    2   12    5    1    1    1
as   (r)  -4.000D+00    0NOLAM    1    0
as4  (g)   1.000D+00    0NOLAM    0    0
  407.00D+00    0.00D+00  -10.00D+00    0.00D+00    0.01D+00    0.00D+00
  431.00D+00    0.00D+00  - 9.00D+00    0.00D+00    0.01D+00    0.00D+00
  457.00D+00    0.00D+00  - 8.00D+00    0.00D+00    0.01D+00    0.00D+00
  485.00D+00    0.00D+00  - 7.00D+00    0.00D+00    0.01D+00    0.00D+00
  518.00D+00    0.00D+00  - 6.00D+00    0.00D+00    0.01D+00    0.00D+00
  554.00D+00    0.00D+00  - 5.00D+00    0.00D+00    0.01D+00    0.00D+00
  597.00D+00    0.00D+00  - 4.00D+00    0.00D+00    0.01D+00    0.00D+00
  647.00D+00    0.00D+00  - 3.00D+00    0.00D+00    0.01D+00    0.00D+00
  708.00D+00    0.00D+00  - 2.00D+00    0.00D+00    0.01D+00    0.00D+00
  783.00D+00    0.00D+00  - 1.00D+00    0.00D+00    0.01D+00    0.00D+00
  876.00D+00    0.00D+00    0.00D+00    0.00D+00    0.01D+00    0.00D+00
  999.00D+00    0.00D+00    1.00D+00    0.00D+00    0.01D+00    0.00D+00
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
```

```
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00 0.00000000D+00
01001100010000000000001100110000000001001100110000000001001100110000000001001100
1100000000000000000000000000
    4    5    4   10
2000.     0.        10.
```

Attachment B. Primary output from PHAS2OPC

=== Arsenic (after HULTGREN AND OTHERS, 1973) ====

THESE RESULTS WERE OBTAINED IN A RUN AT 2/10/88 10:29

NUMBER OF CYCLES IN THIS JOB IS 1

NUMBER OF PARAMETERS TO BE VARIED IS 23

NUMBER OF INDEPENDENT VARIABLES PER OBSERVATION IS 4

DERIVATIVES PROGRAMMED IN SUBROUTINE EAFWC .

WEIGHTS TO BE SUPPLIED BY USER

NUMBER OF PARAMETERS READ IS 108

NUMBER OF OBSERVATIONS READ IS 79

TRIAL CONSTANTS

I	P(I)	KI(I)
	as (r)	
1	0.0000D+00	0
2	0.0000D+00	1
3	0.0000D+00	0
4	0.0000D+00	0
5	0.0000D+00	1
6	0.0000D+00	1
7	0.0000D+00	0
8	0.0000D+00	0
9	0.0000D+00	0
10	0.0000D+00	1
11	0.0000D+00	0
12	0.0000D+00	0
13	0.0000D+00	0
14	0.0000D+00	0
15	0.0000D+00	0
16	0.0000D+00	0
17	0.0000D+00	0
18	0.0000D+00	0
	as (g)	
19	0.0000D+00	0
20	0.0000D+00	0
21	0.0000D+00	0
22	0.0000D+00	0
23	0.0000D+00	1
24	0.0000D+00	1
25	0.0000D+00	0
26	0.0000D+00	0
27	0.0000D+00	1
28	0.0000D+00	1
29	0.0000D+00	0
30	0.0000D+00	0
31	0.0000D+00	0
32	0.0000D+00	0
33	0.0000D+00	0
34	0.0000D+00	0
35	0.0000D+00	0
36	0.0000D+00	0
	as2 (g)	
37	0.0000D+00	0
38	0.0000D+00	1
39	0.0000D+00	0
40	0.0000D+00	0
41	0.0000D+00	1
42	0.0000D+00	1
43	0.0000D+00	0
44	0.0000D+00	0

45	0.0000D+00	1
46	0.0000D+00	1
47	0.0000D+00	0
48	0.0000D+00	0
49	0.0000D+00	0
50	0.0000D+00	0
51	0.0000D+00	0
52	0.0000D+00	0
53	0.0000D+00	0
54	0.0000D+00	0
	as3 (g)	
55	0.0000D+00	0
56	0.0000D+00	1
57	0.0000D+00	0
58	0.0000D+00	0
59	0.0000D+00	1
60	0.0000D+00	1
61	0.0000D+00	0
62	0.0000D+00	0
63	0.0000D+00	1
64	0.0000D+00	1
65	0.0000D+00	0
66	0.0000D+00	0
67	0.0000D+00	0
68	0.0000D+00	0
69	0.0000D+00	0
70	0.0000D+00	0
71	0.0000D+00	0
72	0.0000D+00	0
	as4 (g)	
73	0.0000D+00	0
74	0.0000D+00	1
75	0.0000D+00	0
76	0.0000D+00	0
77	0.0000D+00	1
78	0.0000D+00	1
79	0.0000D+00	0
80	0.0000D+00	0
81	0.0000D+00	1
82	0.0000D+00	1
83	0.0000D+00	0
84	0.0000D+00	0
85	0.0000D+00	0
86	0.0000D+00	0
87	0.0000D+00	0
88	0.0000D+00	0
89	0.0000D+00	0
90	0.0000D+00	0
	as (l)	
91	0.0000D+00	0
92	0.0000D+00	0
93	0.0000D+00	0
94	0.0000D+00	0
95	0.0000D+00	0
96	0.0000D+00	0
97	0.0000D+00	0
98	0.0000D+00	0
99	0.0000D+00	0
100	0.0000D+00	0
101	0.0000D+00	0
102	0.0000D+00	0
103	0.0000D+00	0
104	0.0000D+00	0
105	0.0000D+00	0
106	0.0000D+00	0
107	0.0000D+00	0
108	0.0000D+00	0

```
===    Arsenic (after HULTGREN AND OTHERS, 1973)    ====
    1   !{arsenic, heat capacity     HULTGREN AND OTHERS (1973)
    2   !{as, gas, heat capacity     HULTGREN AND OTHERS (1973)
    3   !{as2, gas, heat capacity     HULTGREN AND OTHERS (1973)
    4   !{as3, gas, heat capacity     HULTGREN AND OTHERS (1973)
    5   !{as4, gas, heat capacity     HULTGREN AND OTHERS (1973)
    6   !{{as, entropy     HULTGREN AND OTHERS (1973)
    7   !{{as, gas, entropy     HULTGREN AND OTHERS (1973)
    8   !{{as2, gas, entropy     HULTGREN AND OTHERS (1973)
    9   !{{as3, gas, entropy     HULTGREN AND OTHERS (1973)
   10   !{{as4, gas, entropy     HULTGREN AND OTHERS (1973)
   11   !}}}as(c) = as(g)     HULTGREN AND OTHERS (1973)
   12   !}}}as(c) = .5 as2(g)     HULTGREN AND OTHERS (1973)
   13   !}}}as(c) = 1/3 as3(g)     HULTGREN AND OTHERS (1973)
   14   !}}}as(c) = .25 as4(g)     HULTGREN AND OTHERS (1973)
   15   !}}}}as(c) = .25 as4(g)     HULTGREN AND OTHERS (1973)
   16   !}}}}}4 as(c) = as4(g)     HULTGREN AND OTHERS (1973)

AGREEMENT FACTORS BASED ON PARAMETERS BEFORE CYCLE 1

SUM(W*(O-C)**2) IS     .380D+04

SQRTF(SUM(W*(O-C)**2)/(NO-NV)) IS  .8235D+01
```

=== Arsenic (after HULTGREN AND OTHERS, 1973) ====

PARAMETERS AFTER LEAST SQUARES CYCLE 1

	OLD	CHANGE	NEW	ERROR	PCT. CHANGE	PCT. ERROR
as	(r): CONSTRAINTS (ELEM = 1, OTHER = 0) = 1 AND LAMBDA ANOMALY = F, Ja = 0 and Jb = 0.					
1	0.0000000000D+00					
2	0.0000000000D+00	-9.5529926461D+04	-9.5529926461D+04	5.2582923064D+03	1.0000000000D+02	5.5043
3	0.0000000000D+00					
4	0.0000000000D+00					
5	0.0000000000D+00	2.3951479730D+01	2.3951479730D+01	1.0668544365D-01	1.0000000000D+02	.4454
6	0.0000000000D+00	4.8602121786D-03	4.8602121786D-03	1.1684651902D-04	1.0000000000D+02	2.4041
7	0.0000000000D+00					
8	0.0000000000D+00					
9	0.0000000000D+00					
10	0.0000000000D+00	-1.0271754533D+02	-1.0271754533D+02	6.1293677456D-01	1.0000000000D+02	.5967
11	0.0000000000D+00					
12	0.0000000000D+00					
13	0.0000000000D+00					
14	0.0000000000D+00					
15	0.0000000000D+00					
16	0.0000000000D+00					
17	0.0000000000D+00					
18	0.0000000000D+00					
as	(g): CONSTRAINTS (ELEM = 1, OTHER = 0) = 0 AND LAMBDA ANOMALY = F, Ja = 0 and Jb = 0.					
19	0.0000000000D+00					
20	0.0000000000D+00					
21	0.0000000000D+00					
22	0.0000000000D+00					
23	0.0000000000D+00	2.0782647836D+01	2.0782647836D+01	7.0852129180D-02	1.0000000000D+02	.3409
24	0.0000000000D+00	6.0493069907D-06	6.0493069907D-06	8.7476798184D-05	1.0000000000D+02	1446.0630
25	0.0000000000D+00					
26	0.0000000000D+00					
27	0.0000000000D+00	3.0620777371D+05	3.0620777371D+05	4.4126327302D+03	1.0000000000D+02	1.4411
28	0.0000000000D+00	5.5687474348D+01	5.5687474348D+01	3.8766979225D-01	1.0000000000D+02	.6962
29	0.0000000000D+00					
30	0.0000000000D+00					
31	0.0000000000D+00					
32	0.0000000000D+00					
33	0.0000000000D+00					
34	0.0000000000D+00					
35	0.0000000000D+00					
36	0.0000000000D+00					
as2	(g): CONSTRAINTS (ELEM = 1, OTHER = 0) = 0 AND LAMBDA ANOMALY = F, Ja = 0 and Jb = 0.					
37	0.0000000000D+00					
38	0.0000000000D+00	-1.9789508482D+05	-1.9789508482D+05	2.9882443936D+04	1.0000000000D+02	15.1001
39	0.0000000000D+00					
40	0.0000000000D+00					
41	0.0000000000D+00	3.7148093028D+01	3.7148093028D+01	3.3417921394D-01	1.0000000000D+02	.8996
42	0.0000000000D+00	2.0748551902D-04	2.0748551902D-04	3.3017098858D-04	1.0000000000D+02	159.1297
43	0.0000000000D+00					
44	0.0000000000D+00					
45	0.0000000000D+00	2.3055882872D+05	2.3055882872D+05	1.3640076137D+04	1.0000000000D+02	5.9161
46	0.0000000000D+00	2.7942646287D+01	2.7942646287D+01	1.9681184717D+00	1.0000000000D+02	7.0434
47	0.0000000000D+00					
48	0.0000000000D+00					
49	0.0000000000D+00					
50	0.0000000000D+00					
51	0.0000000000D+00					
52	0.0000000000D+00					
53	0.0000000000D+00					
54	0.0000000000D+00					
as3	(g): CONSTRAINTS (ELEM = 1, OTHER = 0) = 0 AND LAMBDA ANOMALY = F, Ja = 0 and Jb = 0.					
55	0.0000000000D+00					
56	0.0000000000D+00	-2.8106549242D+05	-2.8106549242D+05	4.4824114146D+04	1.0000000000D+02	15.9479
57	0.0000000000D+00					
58	0.0000000000D+00					

```
 59     0.0000000000D+00     6.2138946854D+01     6.2138946854D+01     5.0127383364D-01     1.0000000000D+02          .8067
 60     0.0000000000D+00     1.6994009433D-04     1.6994009433D-04     4.9526143548D-04     1.0000000000D+02        291.4330
 61     0.0000000000D+00
 62     0.0000000000D+00
 63     0.0000000000D+00     3.8628579995D+04     3.8628579995D+04     7.2206770035D+04     1.0000000000D+02        186.9258
 64     0.0000000000D+00    -4.5548874650D+01    -4.5548874650D+01     2.9522072296D+00     1.0000000000D+02          6.4814
 65     0.0000000000D+00
 66     0.0000000000D+00
 67     0.0000000000D+00
 68     0.0000000000D+00
 69     0.0000000000D+00
 70     0.0000000000D+00
 71     0.0000000000D+00
 72     0.0000000000D+00

as4  (g):  CONSTRAINTS (ELEM = 1, OTHER = 0) =  0 AND LAMBDA ANOMALY = F, Ja =  0 and Jb =  0.
 73     0.0000000000D+00
 74     0.0000000000D+00    -5.1213724856D+05    -5.1213724856D+05     5.9720704305D+04     1.0000000000D+02         11.6611
 75     0.0000000000D+00
 76     0.0000000000D+00
 77     0.0000000000D+00     8.2866670490D+01     8.2866670490D+01     6.6424500905D-01     1.0000000000D+02          .8016
 78     0.0000000000D+00     2.1633965626D-04     2.1633965626D-04     6.5653186756D-04     1.0000000000D+02        303.4727
 79     0.0000000000D+00
 80     0.0000000000D+00
 81     0.0000000000D+00     1.6931756004D+05     1.6931756004D+05     3.9381845185D+02     1.0000000000D+02          .2326
 82     0.0000000000D+00    -1.4794379550D+02    -1.4794379550D+02     3.9289715837D+00     1.0000000000D+02          2.6557
 83     0.0000000000D+00
 84     0.0000000000D+00
 85     0.0000000000D+00
 86     0.0000000000D+00
 87     0.0000000000D+00
 88     0.0000000000D+00
 89     0.0000000000D+00
 90     0.0000000000D+00

as   (1):  CONSTRAINTS (ELEM = 1, OTHER = 0) =  0 AND LAMBDA ANOMALY = F, Ja =  0 and Jb =  0.
 91     0.0000000000D+00
 92     0.0000000000D+00
 93     0.0000000000D+00
 94     0.0000000000D+00
 95     0.0000000000D+00
 96     0.0000000000D+00
 97     0.0000000000D+00
 98     0.0000000000D+00
 99     0.0000000000D+00
100     0.0000000000D+00
101     0.0000000000D+00
102     0.0000000000D+00
103     0.0000000000D+00
104     0.0000000000D+00
105     0.0000000000D+00
106     0.0000000000D+00
107     0.0000000000D+00
108     0.0000000000D+00

ESTIMATED AGREEMENT FACTORS BASED ON PARAMETERS AFTER CYCLE 1

SUM(W*(O-C)**2) IS     .141D-01

SQRTF(SUM(W*(O-C)**2)/(NO-NV)) IS  .1586D-01
```

=== Arsenic (after HULTGREN AND OTHERS, 1973) ====

CALCULATED Y BASED ON PARAMETERS BEFORE CYCLE 2

INDEX	X(1,I)	X(2,I)	X(3,I)	X(4,I)	Y(OBS)	Y(CALC)	OBS-CALC	SIG(O)	(O-C)/SIG(O)
1	!{arsenic, heat capacity			HULTGREN AND OTHERS (1973)					
1	200.000		.00	.00	22.4597	22.5353	-.0756	.0418	-1.8060
2	250.000		.00	.00	23.5350	23.6381	-.1031	.0418	-2.4630
3	298.150		.00	.00	24.6521	24.3259	.3262	.0418	7.7972
4	400.000		.00	.00	25.3885	25.2985	.0900	.0418	2.1513
5	500.000		.00	.00	25.9450	25.9995	-.0545	.0418	-1.3022
6	600.000		.00	.00	26.5015	26.6022	-.1008	.0418	-2.4089
7	700.000		.00	.00	27.0537	27.1587	-.1049	.0418	-2.5078
8	800.000		.00	.00	27.6102	27.6904	-.0802	.0418	-1.9161
9	876.000		.00	.00	28.0328	28.0845	-.0517	.0418	-1.2365
10	900.000		.00	.00	28.1667	28.2077	-.0410	.0418	-.9810
11	1000.000		.00	.00	28.7232	28.7162	.0070	.0418	.1673
12	1100.000		.00	.00	29.2796	29.2188	.0609	.0418	1.4548
13	1200.000		.00	.00	29.8319	29.7174	.1145	.0418	2.7372
2	!{as, gas, heat capacity			HULTGREN AND OTHERS (1973)					
14	298.150		.00	.00	20.7861	20.7845	.0017	.0418	.0397
15	400.000		.00	.00	20.7861	20.7851	.0010	.0418	.0250
16	500.000		.00	.00	20.7861	20.7857	.0004	.0418	.0105
17	600.000		.00	.00	20.7861	20.7863	-.0002	.0418	-.0040
18	700.000		.00	.00	20.7861	20.7869	-.0008	.0418	-.0184
19	800.000		.00	.00	20.7861	20.7875	-.0014	.0418	-.0329
20	876.000		.00	.00	20.7861	20.7879	-.0018	.0418	-.0439
21	900.000		.00	.00	20.7861	20.7881	-.0020	.0418	-.0473
22	1000.000		.00	.00	20.7861	20.7887	-.0026	.0418	-.0618
23	1100.000		.00	.00	20.7903	20.7893	.0010	.0418	.0238
24	1200.000		.00	.00	20.7945	20.7899	.0046	.0418	.1093
3	!{as2, gas, heat capacity			HULTGREN AND OTHERS (1973)					
25	298.150		.00	.00	17.5017	17.4919	.0098	.0418	.2342
26	400.000		.00	.00	17.9786	17.9971	-.0185	.0418	-.4415
27	500.000		.00	.00	18.2213	18.2301	-.0088	.0418	-.2105
28	600.000		.00	.00	18.3636	18.3614	.0021	.0418	.0511
29	700.000		.00	.00	18.4514	18.4447	.0067	.0418	.1603
30	800.000		.00	.00	18.5100	18.5024	.0076	.0418	.1812
31	876.000		.00	.00	18.5435	18.5360	.0075	.0418	.1794
32	900.000		.00	.00	18.5519	18.5453	.0066	.0418	.1577
33	1000.000		.00	.00	18.5811	18.5788	.0023	.0418	.0550
34	1100.000		.00	.00	18.6021	18.6064	-.0043	.0418	-.1034
35	1200.000		.00	.00	18.6188	18.6298	-.0110	.0418	-.2635
4	!{as3, gas, heat capacity			HULTGREN AND OTHERS (1973)					
36	298.150		.00	.00	19.6815	19.6757	.0058	.0418	.1387
37	400.000		.00	.00	20.1376	20.1499	-.0123	.0418	-.2938
38	500.000		.00	.00	20.3635	20.3663	-.0028	.0418	-.0674
39	600.000		.00	.00	20.4890	20.4865	.0025	.0418	.0604
40	700.000		.00	.00	20.5644	20.5612	.0031	.0418	.0748
41	800.000		.00	.00	20.6146	20.6117	.0029	.0418	.0684
42	876.000		.00	.00	20.6439	20.6403	.0035	.0418	.0848
43	900.000		.00	.00	20.6522	20.6481	.0041	.0418	.0987
44	1000.000		.00	.00	20.6773	20.6757	.0016	.0418	.0381
45	1100.000		.00	.00	20.6941	20.6977	-.0036	.0418	-.0859
46	1200.000		.00	.00	20.7108	20.7157	-.0049	.0418	-.1169
5	!{as4, gas, heat capacity			HULTGREN AND OTHERS (1973)					
47	298.150		.00	.00	19.3050	19.2925	.0125	.0418	.2986
48	400.000		.00	.00	19.9326	19.9381	-.0055	.0418	-.1317
49	500.000		.00	.00	20.2296	20.2316	-.0019	.0418	-.0462
50	600.000		.00	.00	20.3970	20.3935	.0035	.0418	.0844
51	700.000		.00	.00	20.4974	20.4932	.0042	.0418	.1000
52	800.000		.00	.00	20.5644	20.5599	.0045	.0418	.1070

53	876.000	.00	.00	20.5978	20.5972	.0006	.0418	.0151
54	900.000	.00	.00	20.6104	20.6073	.0031	.0418	.0743
55	1000.000	.00	.00	20.6439	20.6427	.0011	.0418	.0272
56	1100.000	.00	.00	20.6690	20.6703	-.0014	.0418	-.0332
57	1200.000	.00	.00	20.6857	20.6927	-.0070	.0418	-.1664

6 !{{as, entropy HULTGREN AND OTHERS (1973)

58	298.150	.00	.00	35.6895	35.7347	-.0452	.0418	-1.0805

7 !{{as, gas, entropy HULTGREN AND OTHERS (1973)

59	298.150	.00	.00	174.1004	174.1004	.0000	.0418	.0000

8 !{{as2, gas, entropy HULTGREN AND OTHERS (1973)

60	298.150	.00	.00	120.3862	120.3862	.0000	.0418	.0000

9 !{{as3, gas, entropy HULTGREN AND OTHERS (1973)

61	298.150	.00	.00	103.3741	103.3741	.0000	.0418	.0000

10 !{{as4, gas, entropy HULTGREN AND OTHERS (1973)

62	298.150	.00	.00	81.8307	81.7856	.0451	.0418	1.0784

11 !}}}as(c) = as(g) HULTGREN AND OTHERS (1973)

63	298.150	.00	.00	301750.1000	301750.1000	.0000	2301.2000	.0000

12 !}}}as(c) = .5 as2(g) HULTGREN AND OTHERS (1973)

64	298.150	.00	.00	110499.4000	110499.4000	.0000	3556.4000	.0000

13 !}}}as(c) = 1/3 as3(g) HULTGREN AND OTHERS (1973)

65	298.150	.00	.00	8714.0170	8714.0170	.0000	12552.0000	.0000

14 !}}}as(c) = .25 as4(g) HULTGREN AND OTHERS (1973)

66	298.150	.00	.00	38325.4400	38283.5900	41.8515	418.4000	.1000

15 !}}}}as(c) = .25 as4(g) HULTGREN AND OTHERS (1973)

67	876.000	.00	.00	.0000	.0000	.0000	.0010	-.0000

16 !}}}}}4 as(c) = as4(g) HULTGREN AND OTHERS (1973)

68	407.000	.00	.00	-10.0000	-10.0782	.0782	.0100	7.8248
69	431.000	.00	.00	-9.0000	-9.0019	.0019	.0100	.1883
70	457.000	.00	.00	-8.0000	-7.9672	-.0328	.0100	-3.2824
71	485.000	.00	.00	-7.0000	-6.9809	-.0191	.0100	-1.9079
72	518.000	.00	.00	-6.0000	-5.9602	-.0398	.0100	-3.9802
73	554.000	.00	.00	-5.0000	-4.9907	-.0093	.0100	-.9349
74	597.000	.00	.00	-4.0000	-3.9923	-.0077	.0100	-.7745
75	647.000	.00	.00	-3.0000	-3.0060	.0060	.0100	.5981
76	708.000	.00	.00	-2.0000	-2.0013	.0013	.0100	.1265
77	783.000	.00	.00	-1.0000	-.9934	-.0066	.0100	-.6630
78	876.000	.00	.00	.0000	.0000	.0000	.0100	.0000
79	999.000	.00	.00	1.0000	1.0061	-.0061	.0100	-.6092

AGREEMENT FACTORS BASED ON PARAMETERS BEFORE CYCLE 2

SUM(W*(O-C)**2) IS .138D-01

SQRTF(SUM(W*(O-C)**2)/(NO-NV)) IS .1569D-01

THESE RESULTS WERE COMPLETED IN A RUN AT 2/10/88 10:30

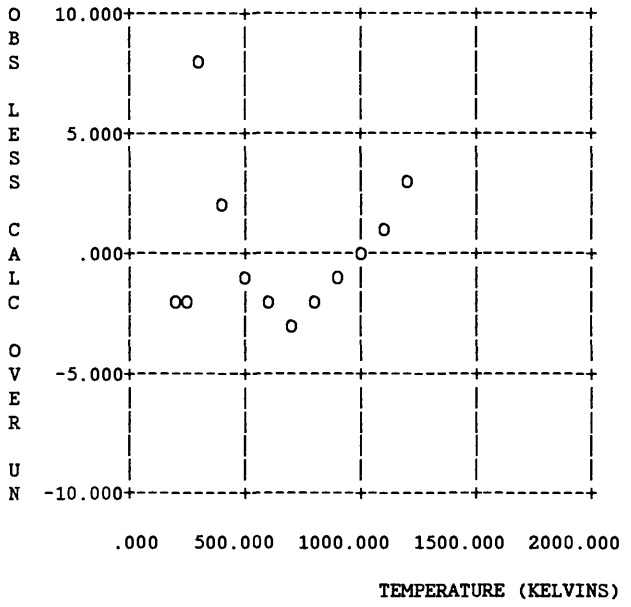
Attachment C. PLOTS Output from PHAS20PC

==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

INDEPENDENT VARIABLES				HEAT CAP		SIGYO	ERROR		
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG
1 !{arsenic, heat capacity				HULTGREN AND OTHERS (1973)					
200.00	.00	.00	.00	2.24597D+01	2.25353D+01	4.1840D-02	-.0756	-.3364	-1.8060
250.00	.00	.00	.00	2.35350D+01	2.36381D+01	4.1840D-02	-.1031	-.4379	-2.4630
298.15	.00	.00	.00	2.46521D+01	2.43259D+01	4.1840D-02	.3262	1.3233	7.7972
400.00	.00	.00	.00	2.53885D+01	2.52985D+01	4.1840D-02	.0900	.3545	2.1513
500.00	.00	.00	.00	2.59450D+01	2.59995D+01	4.1840D-02	-.0545	-.2100	-1.3022
600.00	.00	.00	.00	2.65015D+01	2.66022D+01	4.1840D-02	-.1008	-.3803	-2.4089
700.00	.00	.00	.00	2.70537D+01	2.71587D+01	4.1840D-02	-.1049	-.3878	-2.5078
800.00	.00	.00	.00	2.76102D+01	2.76904D+01	4.1840D-02	-.0802	-.2904	-1.9161
876.00	.00	.00	.00	2.80328D+01	2.80845D+01	4.1840D-02	-.0517	-.1846	-1.2365
900.00	.00	.00	.00	2.81667D+01	2.82077D+01	4.1840D-02	-.0410	-.1457	-.9810
1000.00	.00	.00	.00	2.87232D+01	2.87162D+01	4.1840D-02	.0070	.0244	.1673
1100.00	.00	.00	.00	2.92796D+01	2.92188D+01	4.1840D-02	.0609	.2079	1.4548
1200.00	.00	.00	.00	2.98319D+01	2.97174D+01	4.1840D-02	.1145	.3839	2.7372
THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE-------							-.0010	-.0061	-.0241
THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE--------							.1185	.4683	2.8329

THE NUMBER OF OBSERVATIONS IS 13

1 !{arsenic, heat capacity HULTGREN AND OTHERS (1973)



==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

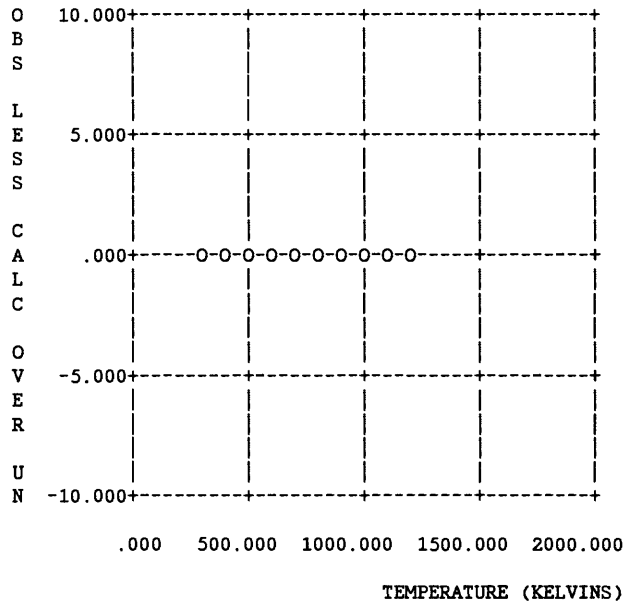
INDEPENDENT VARIABLES				HEAT CAP		SIGYO	ERROR		
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG
2 !{as, gas, heat capacity HULTGREN AND OTHERS (1973)									
298.15	.00	.00	.00	2.07861D+01	2.07845D+01	4.1840D-02	.0017	.0080	.0397
400.00	.00	.00	.00	2.07861D+01	2.07851D+01	4.1840D-02	.0010	.0050	.0250
500.00	.00	.00	.00	2.07861D+01	2.07857D+01	4.1840D-02	.0004	.0021	.0105
600.00	.00	.00	.00	2.07861D+01	2.07863D+01	4.1840D-02	-.0002	-.0008	-.0040
700.00	.00	.00	.00	2.07861D+01	2.07869D+01	4.1840D-02	-.0008	-.0037	-.0184
800.00	.00	.00	.00	2.07861D+01	2.07875D+01	4.1840D-02	-.0014	-.0066	-.0329
876.00	.00	.00	.00	2.07861D+01	2.07879D+01	4.1840D-02	-.0018	-.0088	-.0439
900.00	.00	.00	.00	2.07861D+01	2.07881D+01	4.1840D-02	-.0020	-.0095	-.0473
1000.00	.00	.00	.00	2.07861D+01	2.07887D+01	4.1840D-02	-.0026	-.0124	-.0618
1100.00	.00	.00	.00	2.07903D+01	2.07893D+01	4.1840D-02	.0010	.0048	.0238
1200.00	.00	.00	.00	2.07945D+01	2.07899D+01	4.1840D-02	.0046	.0220	.1093

THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE------- .0000 .0000 .0000

THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE-------- .0020 .0094 .0469

THE NUMBER OF OBSERVATIONS IS 11

2 !{as, gas, heat capacity HULTGREN AND OTHERS (1973)

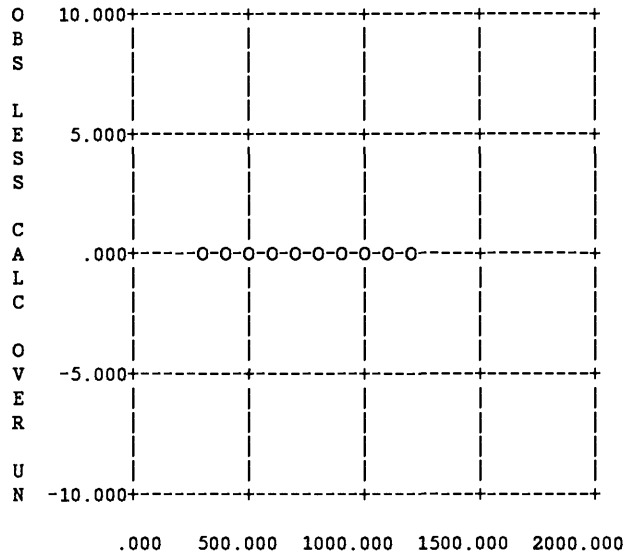


==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

INDEPENDENT VARIABLES				HEAT CAP		SIGYO	ERROR		
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG
3 !{as2, gas, heat capacity HULTGREN AND OTHERS (1973)									
298.15	.00	.00	.00	1.75017D+01	1.74919D+01	4.1840D-02	.0098	.0560	.2342
400.00	.00	.00	.00	1.79786D+01	1.79971D+01	4.1840D-02	-.0185	-.1028	-.4415
500.00	.00	.00	.00	1.82213D+01	1.82301D+01	4.1840D-02	-.0088	-.0483	-.2105
600.00	.00	.00	.00	1.83636D+01	1.83614D+01	4.1840D-02	.0021	.0116	.0511
700.00	.00	.00	.00	1.84514D+01	1.84447D+01	4.1840D-02	.0067	.0364	.1603
800.00	.00	.00	.00	1.85100D+01	1.85024D+01	4.1840D-02	.0076	.0410	.1812
876.00	.00	.00	.00	1.85435D+01	1.85360D+01	4.1840D-02	.0075	.0405	.1794
900.00	.00	.00	.00	1.85519D+01	1.85453D+01	4.1840D-02	.0066	.0356	.1577
1000.00	.00	.00	.00	1.85811D+01	1.85788D+01	4.1840D-02	.0023	.0124	.0550
1100.00	.00	.00	.00	1.86021D+01	1.86064D+01	4.1840D-02	-.0043	-.0232	-.1034
1200.00	.00	.00	.00	1.86188D+01	1.86298D+01	4.1840D-02	-.0110	-.0592	-.2635
THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE-------							.0000	-.0000	.0000
THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE--------							.0089	.0489	.2122

THE NUMBER OF OBSERVATIONS IS 11

3 !{as2, gas, heat capacity HULTGREN AND OTHERS (1973)

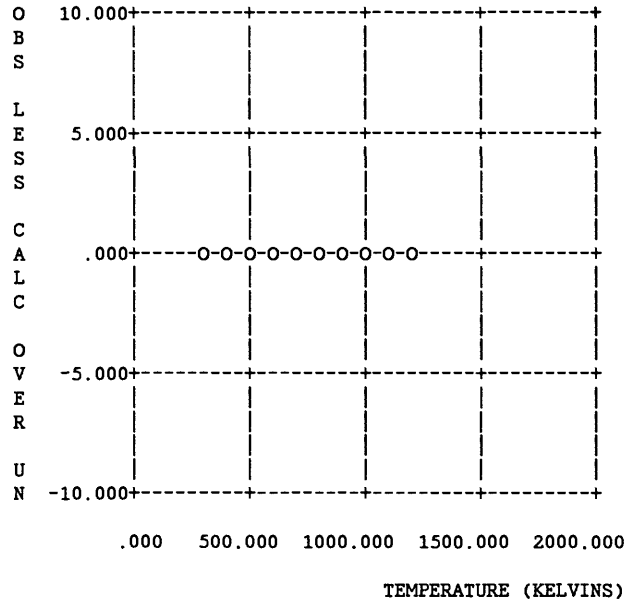


==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

INDEPENDENT VARIABLES				HEAT CAP		SIGYO	ERROR		
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG
4 !{as3, gas, heat capacity HULTGREN AND OTHERS (1973)									
298.15	.00	.00	.00	1.96815D+01	1.96757D+01	4.1840D-02	.0058	.0295	.1387
400.00	.00	.00	.00	2.01376D+01	2.01499D+01	4.1840D-02	-.0123	-.0611	-.2938
500.00	.00	.00	.00	2.03635D+01	2.03663D+01	4.1840D-02	-.0028	-.0138	-.0674
600.00	.00	.00	.00	2.04890D+01	2.04865D+01	4.1840D-02	.0025	.0123	.0604
700.00	.00	.00	.00	2.05644D+01	2.05612D+01	4.1840D-02	.0031	.0152	.0748
800.00	.00	.00	.00	2.06146D+01	2.06117D+01	4.1840D-02	.0029	.0139	.0684
876.00	.00	.00	.00	2.06439D+01	2.06403D+01	4.1840D-02	.0035	.0172	.0848
900.00	.00	.00	.00	2.06522D+01	2.06481D+01	4.1840D-02	.0041	.0200	.0987
1000.00	.00	.00	.00	2.06773D+01	2.06757D+01	4.1840D-02	.0016	.0077	.0381
1100.00	.00	.00	.00	2.06941D+01	2.06977D+01	4.1840D-02	-.0036	-.0174	-.0859
1200.00	.00	.00	.00	2.07108D+01	2.07157D+01	4.1840D-02	-.0049	-.0236	-.1169
THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE-------							.0000	.0000	.0000
THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE--------							.0051	.0252	.1219

THE NUMBER OF OBSERVATIONS IS 11

4 !{as3, gas, heat capacity HULTGREN AND OTHERS (1973)

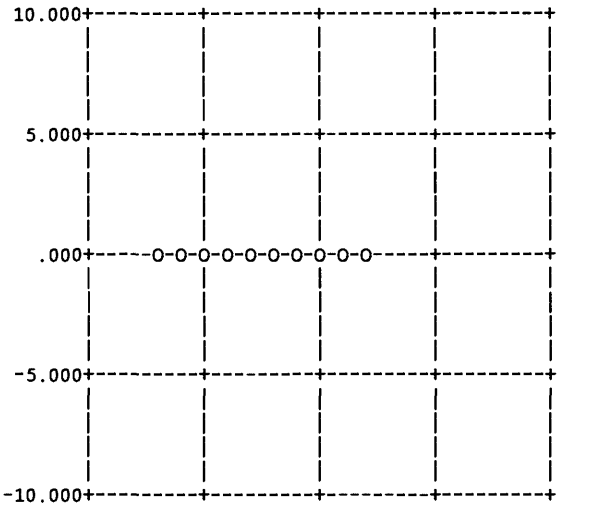


==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

INDEPENDENT VARIABLES				HEAT CAP		SIGYO	ERROR		
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG
5 !{as4, gas, heat capacity HULTGREN AND OTHERS (1973)									
298.15	.00	.00	.00	1.93050D+01	1.92925D+01	4.1840D-02	.0125	.0647	.2986
400.00	.00	.00	.00	1.99326D+01	1.99381D+01	4.1840D-02	-.0055	-.0276	-.1317
500.00	.00	.00	.00	2.02296D+01	2.02316D+01	4.1840D-02	-.0019	-.0096	-.0462
600.00	.00	.00	.00	2.03970D+01	2.03935D+01	4.1840D-02	.0035	.0173	.0844
700.00	.00	.00	.00	2.04974D+01	2.04932D+01	4.1840D-02	.0042	.0204	.1000
800.00	.00	.00	.00	2.05644D+01	2.05599D+01	4.1840D-02	.0045	.0218	.1070
876.00	.00	.00	.00	2.05978D+01	2.05972D+01	4.1840D-02	.0006	.0031	.0151
900.00	.00	.00	.00	2.06104D+01	2.06073D+01	4.1840D-02	.0031	.0151	.0743
1000.00	.00	.00	.00	2.06439D+01	2.06427D+01	4.1840D-02	.0011	.0055	.0272
1100.00	.00	.00	.00	2.06690D+01	2.06703D+01	4.1840D-02	-.0014	-.0067	-.0332
1200.00	.00	.00	.00	2.06857D+01	2.06927D+01	4.1840D-02	-.0070	-.0337	-.1664
THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE-------							.0013	.0064	.0299
THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE--------							.0051	.0256	.1216

THE NUMBER OF OBSERVATIONS IS 11

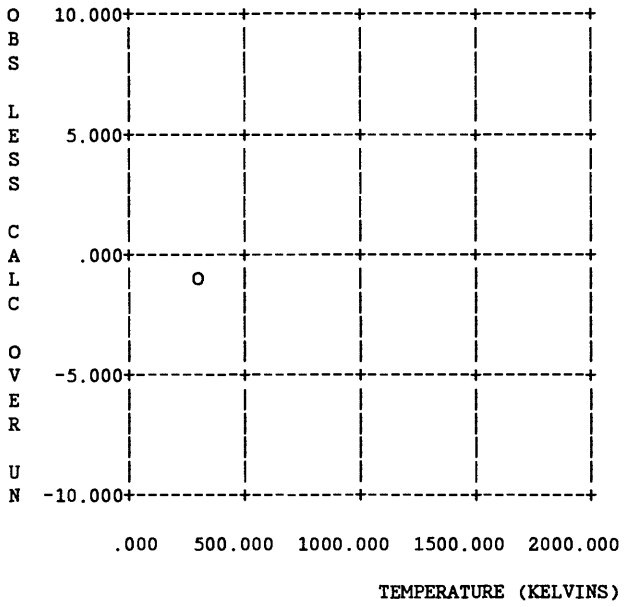
5 !{as4, gas, heat capacity HULTGREN AND OTHERS (1973)



==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

INDEPENDENT VARIABLES				ENTROPY		SIGYO	ERROR		
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG
6 !{{as, entropy HULTGREN AND OTHERS (1973)									
298.15	.00	.00	.00	3.56895D+01	3.57347D+01	4.1840D-02	-.0452	-.1267	-1.0805
THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE-------							-.0452	-.1267	-1.0805
THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE--------							.0000	.0000	.0000
THE NUMBER OF OBSERVATIONS IS 1									

6 !{{as, entropy HULTGREN AND OTHERS (1973)

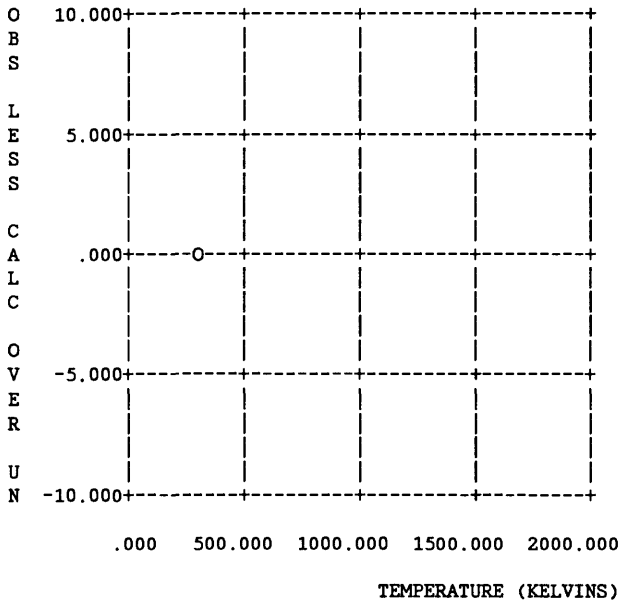


==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

INDEPENDENT VARIABLES				ENTROPY		SIGYO	ERROR		
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG
7 !{{as, gas, entropy HULTGREN AND OTHERS (1973)									
298.15	.00	.00	.00	1.74100D+02	1.74100D+02	4.1840D-02	.0000	.0000	.0000
THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE-------							.0000	.0000	.0000
THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE--------							.0000	.0000	.0000

THE NUMBER OF OBSERVATIONS IS 1

7 !{{as, gas, entropy HULTGREN AND OTHERS (1973)

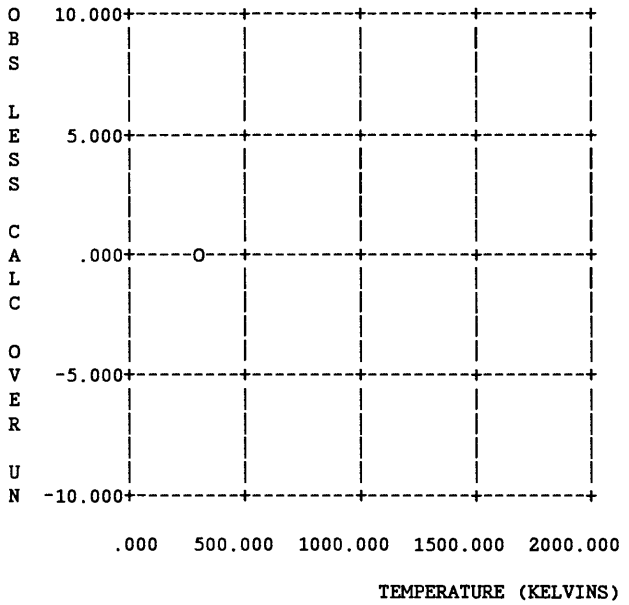


==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

INDEPENDENT VARIABLES				ENTROPY		SIGYO	ERROR		
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG
8 !{{as2, gas, entropy HULTGREN AND OTHERS (1973)									
298.15	.00	.00	.00	1.20386D+02	1.20386D+02	4.1840D-02	.0000	.0000	.0000
THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE-------							.0000	.0000	.0000
THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE--------							.0000	.0000	.0000

THE NUMBER OF OBSERVATIONS IS 1

8 !{{as2, gas, entropy HULTGREN AND OTHERS (1973)

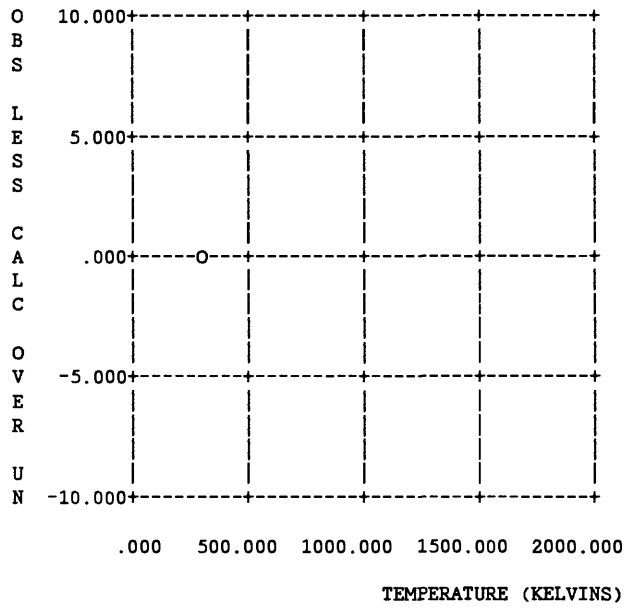


==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

INDEPENDENT VARIABLES				ENTROPY		SIGYO	ERROR		
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG
9 !{{as3, gas, entropy HULTGREN AND OTHERS (1973)									
298.15	.00	.00	.00	1.03374D+02	1.03374D+02	4.1840D-02	.0000	.0000	.0000
THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE-------							.0000	.0000	.0000
THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE--------							.0000	.0000	.0000

THE NUMBER OF OBSERVATIONS IS 1

9 !{{as3, gas, entropy HULTGREN AND OTHERS (1973)

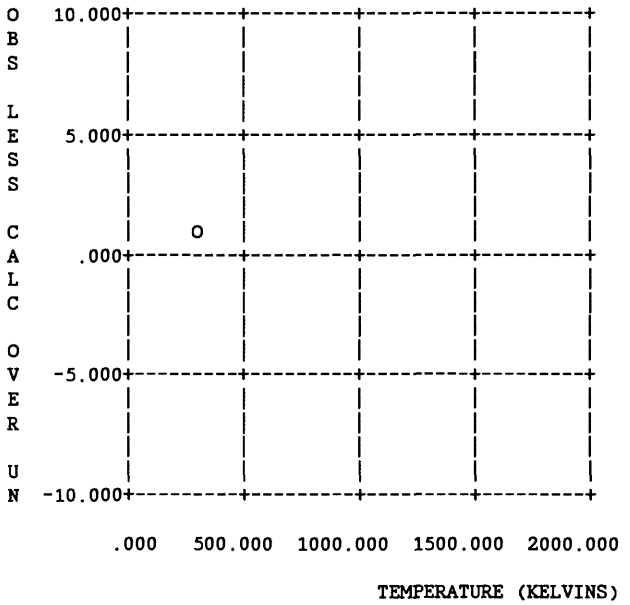


==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

INDEPENDENT VARIABLES				ENTROPY		SIGYO	ERROR		
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG
10 !{{as4, gas, entropy HULTGREN AND OTHERS (1973)									
298.15	.00	.00	.00	8.18307D+01	8.17856D+01	4.1840D-02	.0451	.0551	1.0784
THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE-------							.0451	.0551	1.0784
THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE--------							.0000	.0000	.0000

THE NUMBER OF OBSERVATIONS IS 1

10 !{{as4, gas, entropy HULTGREN AND OTHERS (1973)



==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

INDEPENDENT VARIABLES				ENTHALPY		SIGYO	ERROR		
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG

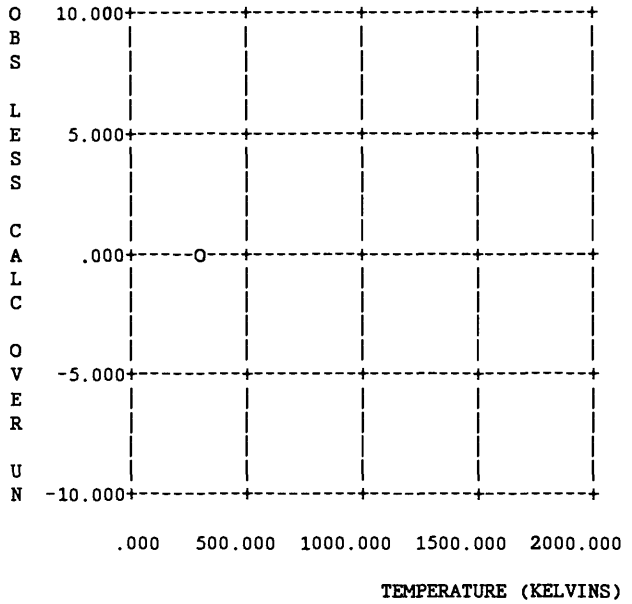
11 !}}}as(c) = as(g) HULTGREN AND OTHERS (1973)

298.15	.00	.00	.00	3.01750D+05	3.01750D+05	2.3012D+03	.0000	.0000	.0000

THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE-------	.0000	.0000	.0000
THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE--------	.0000	.0000	.0000

THE NUMBER OF OBSERVATIONS IS 1

11 !}}}as(c) = as(g) HULTGREN AND OTHERS (1973)

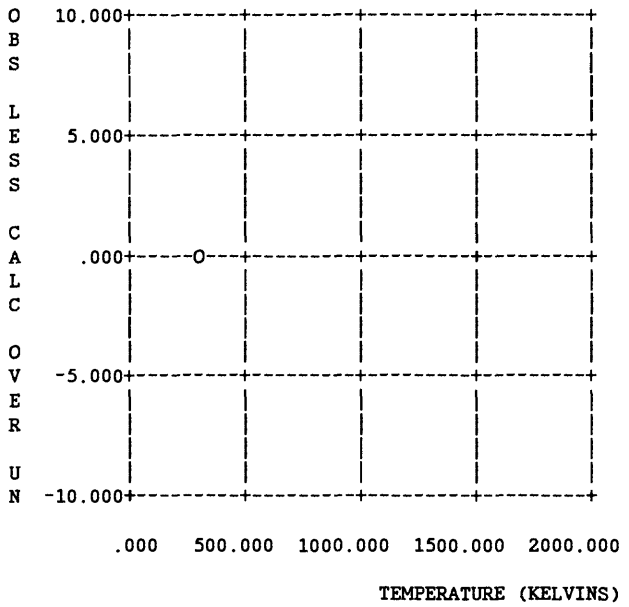


==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

INDEPENDENT VARIABLES				ENTHALPY		SIGYO	ERROR		
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG
12 !}}}as(c) = .5 as2(g) HULTGREN AND OTHERS (1973)									
298.15	.00	.00	.00	1.10499D+05	1.10499D+05	3.5564D+03	.0000	.0000	.0000
THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE-------							.0000	.0000	.0000
THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE--------							.0000	.0000	.0000

THE NUMBER OF OBSERVATIONS IS 1

12 !}}}as(c) = .5 as2(g) HULTGREN AND OTHERS (1973)

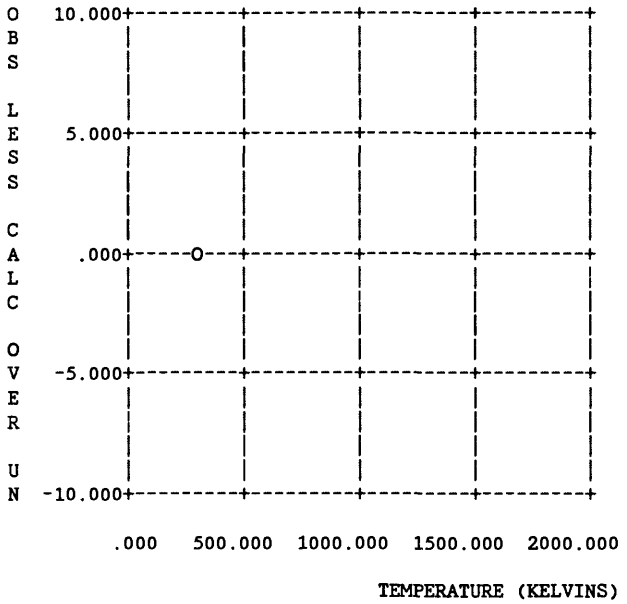


==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

INDEPENDENT VARIABLES				ENTHALPY		SIGYO	ERROR		
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG
13 !}}}as(c) = 1/3 as3(g) HULTGREN AND OTHERS (1973)									
298.15	.00	.00	.00	8.71402D+03	8.71402D+03	1.2552D+04	.0000	.0000	.0000
THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE-------							.0000	.0000	.0000
THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE--------							.0000	.0000	.0000

THE NUMBER OF OBSERVATIONS IS 1

13 !}}}as(c) = 1/3 as3(g) HULTGREN AND OTHERS (1973)



==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

INDEPENDENT VARIABLES				ENTHALPY		SIGYO	ERROR		
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG

14 !}}}as(c) = .25 as4(g) HULTGREN AND OTHERS (1973)

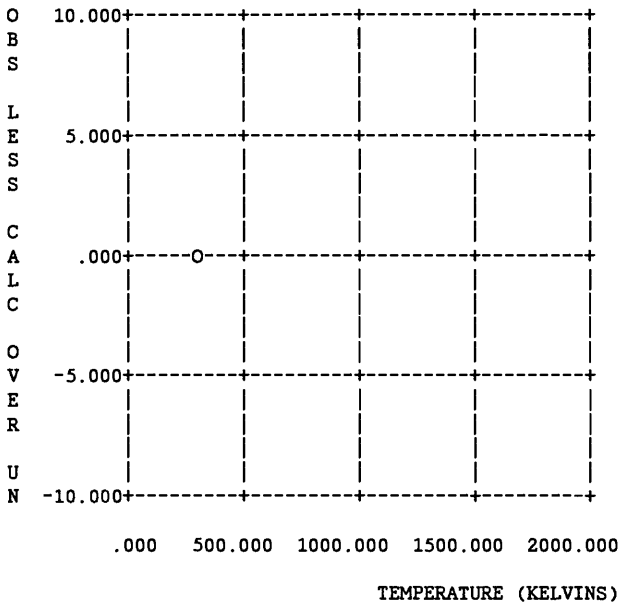
298.15	.00	.00	.00	3.83254D+04	3.82836D+04	4.1840D+02	41.8515	.1092	.1000

THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE------- 41.8515 .1092 .1000

THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE-------- .0000 .0000 .0000

THE NUMBER OF OBSERVATIONS IS 1

14 !}}}as(c) = .25 as4(g) HULTGREN AND OTHERS (1973)

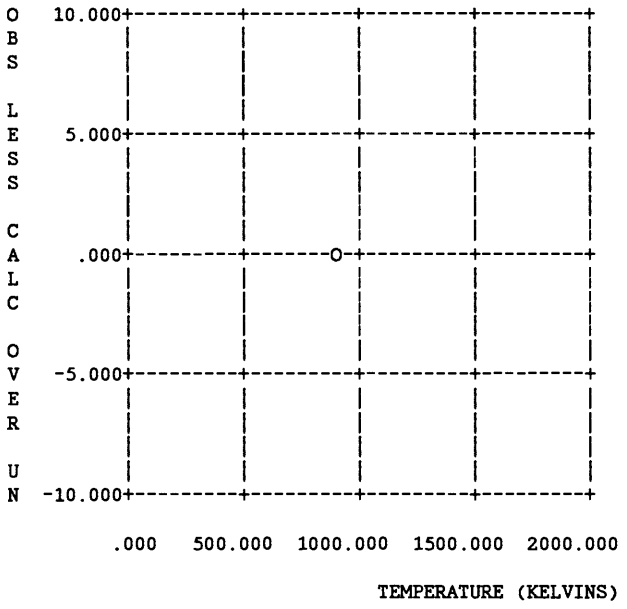


==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

INDEPENDENT VARIABLES				GIBBS EN		SIGYO	ERROR			DELTA H(2
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG	
15 !}}}}as(c) = .25 as4(g) HULTGREN AND OTHERS (1973)										
876.00	.00	.00	.00	0.00000D+00	5.24141D-08	1.0000D-03	.0000	.0000	-.0000	3.828359
THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE-------							.0000	.0000	-.0000	3.828359D
THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE--------							.0000	.0000	.0000	0.000000D

THE NUMBER OF OBSERVATIONS IS 1

15 !}}}}as(c) = .25 as4(g) HULTGREN AND OTHERS (1973)

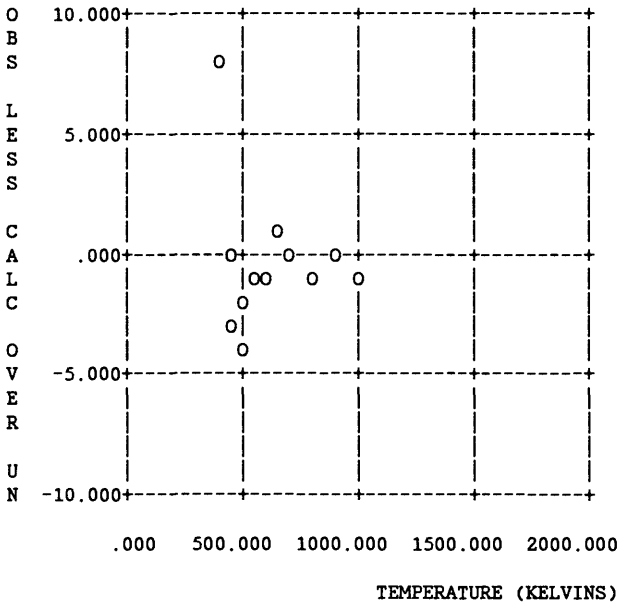


==== Arsenic (after HULTGREN AND OTHERS, 1973) ==== 2/10/88 10:29

INDEPENDENT VARIABLES				LOG K		SIGYO	ERROR			DELTA H(2
X(1,I)	X(2,I)	X(3,I)	X(4,I)	OBSERVED	CALCULATED	(WT)**-.5	O-C	%(O-C)/O	(O-C)/SIG	
16 !}}}}}4 as(c) = as4(g) HULTGREN AND OTHERS (1973)										
407.00	.00	.00	.00	-1.00000D+01	-1.00782D+01	1.0000D-02	.0782	.7825	7.8248	1.525247
431.00	.00	.00	.00	-9.00000D+00	-9.00188D+00	1.0000D-02	.0019	.0209	.1883	1.531188
457.00	.00	.00	.00	-8.00000D+00	-7.96718D+00	1.0000D-02	-.0328	-.4103	-3.2824	1.534215
485.00	.00	.00	.00	-7.00000D+00	-6.98092D+00	1.0000D-02	-.0191	-.2726	-1.9079	1.533115
518.00	.00	.00	.00	-6.00000D+00	-5.96020D+00	1.0000D-02	-.0398	-.6634	-3.9802	1.535291
554.00	.00	.00	.00	-5.00000D+00	-4.99065D+00	1.0000D-02	-.0093	-.1870	-.9349	1.532335
597.00	.00	.00	.00	-4.00000D+00	-3.99226D+00	1.0000D-02	-.0077	-.1936	-.7745	1.532229
647.00	.00	.00	.00	-3.00000D+00	-3.00598D+00	1.0000D-02	.0060	.1994	.5981	1.530603
708.00	.00	.00	.00	-2.00000D+00	-2.00126D+00	1.0000D-02	.0013	.0632	.1265	1.531172
783.00	.00	.00	.00	-1.00000D+00	-9.93370D-01	1.0000D-02	-.0066	-.6630	-.6630	1.532337
876.00	.00	.00	.00	0.00000D+00	-1.25036D-11	1.0000D-02	.0000	.0000	.0000	1.531344
999.00	.00	.00	.00	1.00000D+00	1.00609D+00	1.0000D-02	-.0061	-.6092	-.6092	1.532509
THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE-------							-.0028	-.1611	-.2845	1.531799D
THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN ARE--------							.0279	.3990	2.7870	2.351740D

THE NUMBER OF OBSERVATIONS IS 12

16 !}}}}}4 as(c) = as4(g) HULTGREN AND OTHERS (1973)



Attachment D. UNIT9 output from PHAS2OPC

THESE RESULTS WERE OBTAINED IN A RUN ON 2/10/88 10:29

==== Arsenic (after HULTGREN AND OTHERS, 1973) ====

as (r)

	HEAT CAP	ENTROPY	ENTHALPY	GIBBS EN	LOG K	CELL EMF	HT2-HT1
273.15	23.999	33.619	10050.	867.	-.1658	-.0090	-604.
298.15	24.326	35.735	10654.	2.	.0000	.0000	9.
400.00	25.299	43.030	13185.	-4027.	.5258	.0417	2531.
500.00	25.999	48.753	15751.	-8625.	.9011	.0894	5097.
600.00	26.602	53.547	18382.	-13747.	1.1967	.1425	7727.
700.00	27.159	57.690	21070.	-19313.	1.4411	.2002	10416.
800.00	27.690	61.352	23813.	-25269.	1.6498	.2619	13158.
900.00	28.208	64.643	26608.	-31571.	1.8323	.3272	15953.
1000.00	28.716	67.641	29454.	-38188.	1.9947	.3958	18800.
1100.00	29.219	70.402	32351.	-45091.	2.1412	.4673	21696.
1200.00	29.717	72.966	35297.	-52261.	2.2748	.5417	24643.
1300.00	30.213	75.364	38294.	-59679.	2.3979	.6185	27640.
1400.00	30.707	77.621	41340.	-67329.	2.5120	.6978	30686.
1500.00	31.199	79.756	44435.	-75199.	2.6186	.7794	33781.
1600.00	31.691	81.786	47580.	-83277.	2.7187	.8631	36926.
1700.00	32.181	83.722	50773.	-91553.	2.8130	.9489	40119.
1800.00	32.670	85.575	54016.	-100019.	2.9024	1.0366	43362.
1900.00	33.159	87.354	57308.	-108666.	2.9874	1.1262	46653.
2000.00	33.648	89.068	60648.	-117487.	3.0684	1.2177	49994.

'A'	'B'	'C'	'D'	'E'	'F'	'G'
0.0000000D+00	-9.5529926D+04	0.0000000D+00	0.0000000D+00	2.3951480D+01	4.8602122D-03	0.0000000D+00
0.0000000D+00	2.9767460D+03	-1.0271755D+02	0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00
0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00			

THESE RESULTS WERE OBTAINED IN A RUN ON 2/10/88 10:29

==== Arsenic (after HULTGREN AND OTHERS, 1973) ====

as (g)

	HEAT CAP	ENTROPY	ENTHALPY	GIBBS EN	LOG K	CELL EMF	HT2-HT1
273.15	20.784	172.280	311885.	264826.	-50.6416	-2.7447	-520.
298.15	20.784	174.100	312404.	260496.	-45.6367	-2.6999	5.
400.00	20.785	180.208	314521.	242438.	-31.6584	-2.5127	2117.
500.00	20.786	184.847	316600.	224177.	-23.4190	-2.3234	4195.
600.00	20.786	188.636	318678.	205497.	-17.8896	-2.1298	6274.
700.00	20.787	191.840	320757.	186469.	-13.9141	-1.9326	8353.
800.00	20.787	194.616	322836.	167143.	-10.9130	-1.7323	10431.
900.00	20.788	197.065	324915.	147556.	-8.5637	-1.5293	12510.
1000.00	20.789	199.255	326993.	127738.	-6.6722	-1.3239	14589.
1100.00	20.789	201.236	329072.	107712.	-5.1147	-1.1164	16668.
1200.00	20.790	203.045	331151.	87497.	-3.8085	-.9068	18747.
1300.00	20.791	204.709	333230.	67108.	-2.6964	-.6955	20826.
1400.00	20.791	206.250	335309.	46559.	-1.7371	-.4826	22905.
1500.00	20.792	207.685	337389.	25862.	-.9006	-.2680	24984.
1600.00	20.792	209.027	339468.	5025.	-.1641	-.0521	27063.
1700.00	20.793	210.287	341547.	-15941.	.4898	.1652	29143.
1800.00	20.794	211.476	343626.	-37030.	1.0745	.3838	31222.
1900.00	20.794	212.600	345706.	-58234.	1.6009	.6036	33301.
2000.00	20.795	213.666	347785.	-79548.	2.0775	.8245	35381.

'A'	'B'	'C'	'D'	'E'	'F'	'G'
0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00	2.0782648D+01	6.0493070D-06	0.0000000D+00
0.0000000D+00	3.0620777D+05	5.5687474D+01	0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00
0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00			

THESE RESULTS WERE OBTAINED IN A RUN ON 2/10/88 10:29

==== Arsenic (after HULTGREN AND OTHERS, 1973) ====

as2 (g)

	HEAT CAP	ENTROPY	ENTHALPY	GIBBS EN	LOG K	CELL EMF	HT2-HT1
273.15	34.552	237.727	241438.	176503.	-33.7519	-1.8293	-869.
298.15	34.984	240.772	242308.	170521.	-29.8738	-1.7673	0.
400.00	35.994	251.216	245929.	145443.	-18.9925	-1.5074	3622.
500.00	36.460	259.303	249555.	119903.	-12.5259	-1.2427	7247.
600.00	36.723	265.976	253215.	93629.	-8.1510	-.9704	10907.
700.00	36.889	271.650	256896.	66741.	-4.9802	-.6917	14589.
800.00	37.005	276.584	260591.	39324.	-2.5675	-.4076	18284.
900.00	37.091	280.948	264296.	11443.	-.6641	-.1186	21989.
1000.00	37.158	284.859	268009.	-16850.	.8802	.1746	25701.
1100.00	37.213	288.403	271727.	-45516.	2.1613	.4717	29420.
1200.00	37.260	291.643	275451.	-74521.	3.2437	.7724	33143.
1300.00	37.301	294.627	279179.	-103836.	4.1721	1.0762	36871.
1400.00	37.338	297.393	282911.	-133439.	4.9785	1.3830	40603.
1500.00	37.371	299.970	286646.	-163309.	5.6868	1.6926	44339.
1600.00	37.403	302.383	290385.	-193428.	6.3146	2.0047	48078.
1700.00	37.432	304.651	294127.	-223781.	6.8758	2.3193	51819.
1800.00	37.460	306.792	297871.	-254354.	7.3810	2.6362	55564.
1900.00	37.487	308.818	301619.	-285135.	7.8387	2.9552	59311.
2000.00	37.514	310.741	305369.	-316114.	8.2558	3.2763	63061.

'A'	'B'	'C'	'D'	'E'	'F'	'G'
0.0000000D+00	-1.9789508D+05	0.0000000D+00	0.0000000D+00	3.7148093D+01	2.0748552D-04	0.0000000D+00
0.0000000D+00	2.3055883D+05	2.7942646D+01	0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00
0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00			

THESE RESULTS WERE OBTAINED IN A RUN ON 2/10/88 10:29

==== Arsenic (after HULTGREN AND OTHERS, 1973) ====

as3 (g)

	HEAT CAP	ENTROPY	ENTHALPY	GIBBS EN	LOG K	CELL EMF	HT2-HT1
273.15	58.418	304.982	56637.	-26669.	5.0997	.2764	-1468.
298.15	59.028	310.125	58106.	-34358.	6.0193	.3561	7.
400.00	60.450	327.701	64200.	-66880.	8.7334	.6932	6095.
500.00	61.100	341.267	70281.	-100352.	10.4835	1.0401	12176.
600.00	61.460	352.442	76411.	-135054.	11.7572	1.3997	18305.
700.00	61.684	361.934	82569.	-170785.	12.7438	1.7701	24463.
800.00	61.836	370.181	88745.	-207400.	13.5415	2.1495	30640.
900.00	61.945	377.471	94935.	-244789.	14.2068	2.5371	36829.
1000.00	62.028	384.002	101134.	-282869.	14.7752	2.9317	43028.
1100.00	62.094	389.917	107340.	-321569.	15.2697	3.3328	49234.
1200.00	62.148	395.323	113552.	-360835.	15.7064	3.7398	55446.
1300.00	62.194	400.299	119769.	-400620.	16.0967	4.1521	61663.
1400.00	62.233	404.909	125990.	-440883.	16.4491	4.5694	67885.
1500.00	62.269	409.204	132216.	-481591.	16.7701	4.9913	74110.
1600.00	62.301	413.224	138444.	-522714.	17.0644	5.4176	80339.
1700.00	62.331	417.002	144676.	-564228.	17.3362	5.8478	86570.
1800.00	62.358	420.565	150910.	-606108.	17.5883	6.2819	92805.
1900.00	62.384	423.938	157147.	-648334.	17.8235	6.7195	99042.
2000.00	62.409	427.138	163387.	-690890.	18.0437	7.1606	105281.

'A'	'B'	'C'	'D'	'E'	'F'	'G'
0.0000000D+00	-2.8106549D+05	0.0000000D+00	0.0000000D+00	6.2138947D+01	1.6994009D-04	0.0000000D+00
0.0000000D+00	3.8628580D+04	-4.5548875D+01	0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00
0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00			

THESE RESULTS WERE OBTAINED IN A RUN ON 2/10/88 10:29

==== Arsenic (after HULTGREN AND OTHERS, 1973) ====

as4 (g)

	HEAT CAP	ENTROPY	ENTHALPY	GIBBS EN	LOG K	CELL EMF	HT2-HT1
273.15	76.062	320.431	193836.	106310.	-20.3292	-1.1018	-1916.
298.15	77.170	327.142	195752.	98214.	-17.2063	-1.0179	2.
400.00	79.752	350.236	203762.	63668.	-8.3139	-.6599	8010.
500.00	80.926	368.173	211802.	27716.	-2.8954	-.2873	16051.
600.00	81.574	382.990	219930.	-9864.	.8587	.1022	24178.
700.00	81.973	395.596	228109.	-48809.	3.6421	.5059	32357.
800.00	82.240	406.561	236320.	-88928.	5.8063	.9217	40569.
900.00	82.429	416.259	244554.	-130079.	7.5494	1.3482	48803.
1000.00	82.571	424.951	252805.	-172147.	8.9918	1.7842	57053.
1100.00	82.681	432.827	261067.	-215042.	10.2112	2.2288	65316.
1200.00	82.771	440.025	269340.	-258690.	11.2602	2.6811	73589.
1300.00	82.845	446.653	277621.	-303028.	12.1755	3.1407	81869.
1400.00	82.908	452.795	285909.	-348004.	12.9839	3.6068	90157.
1500.00	82.964	458.517	294202.	-393573.	13.7051	4.0791	98451.
1600.00	83.013	463.873	302501.	-439695.	14.3542	4.5571	106750.
1700.00	83.057	468.907	310805.	-486337.	14.9429	5.0405	115053.
1800.00	83.098	473.655	319113.	-533467.	15.4804	5.5290	123361.
1900.00	83.136	478.149	327424.	-581059.	15.9740	6.0223	131673.
2000.00	83.171	482.414	335740.	-629089.	16.4297	6.5201	139988.

'A'	'B'	'C'	'D'	'E'	'F'	'G'
0.0000000D+00	-5.1213725D+05	0.0000000D+00	0.0000000D+00	8.2866670D+01	2.1633966D-04	0.0000000D+00
0.0000000D+00	1.6931756D+05	-1.4794380D+02	0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00
0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00			

THESE RESULTS WERE OBTAINED IN A RUN ON 2/10/88 10:29

==== Arsenic (after HULTGREN AND OTHERS, 1973) ====

as (l)

	HEAT CAP	ENTROPY	ENTHALPY	GIBBS EN	LOG K	CELL EMF	HT2-HT1
273.15	.000	.000	0.	0.	.0000	.0000	0.
298.15	.000	.000	0.	0.	.0000	.0000	0.
400.00	.000	.000	0.	0.	.0000	.0000	0.
500.00	.000	.000	0.	0.	.0000	.0000	0.
600.00	.000	.000	0.	0.	.0000	.0000	0.
700.00	.000	.000	0.	0.	.0000	.0000	0.
800.00	.000	.000	0.	0.	.0000	.0000	0.
900.00	.000	.000	0.	0.	.0000	.0000	0.
1000.00	.000	.000	0.	0.	.0000	.0000	0.
1100.00	.000	.000	0.	0.	.0000	.0000	0.
1200.00	.000	.000	0.	0.	.0000	.0000	0.
1300.00	.000	.000	0.	0.	.0000	.0000	0.
1400.00	.000	.000	0.	0.	.0000	.0000	0.
1500.00	.000	.000	0.	0.	.0000	.0000	0.
1600.00	.000	.000	0.	0.	.0000	.0000	0.
1700.00	.000	.000	0.	0.	.0000	.0000	0.
1800.00	.000	.000	0.	0.	.0000	.0000	0.
1900.00	.000	.000	0.	0.	.0000	.0000	0.
2000.00	.000	.000	0.	0.	.0000	.0000	0.

'A'	'B'	'C'	'D'	'E'	'F'	'G'
0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00
0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00
0.0000000D+00	0.0000000D+00	0.0000000D+00	0.0000000D+00			

Attachment E. UNIT11 output from PHAS20PC

```
1; 2.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.24597D+01, 2.25353D+01
1; 2.50000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.35350D+01, 2.36381D+01
1; 2.98150D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.46521D+01, 2.43259D+01
1; 4.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.53885D+01, 2.52985D+01
1; 5.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.59450D+01, 2.59995D+01
1; 6.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.65015D+01, 2.66022D+01
1; 7.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.70537D+01, 2.71587D+01
1; 8.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.76102D+01, 2.76904D+01
1; 8.76000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.80328D+01, 2.80845D+01
1; 9.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.81667D+01, 2.82077D+01
1; 1.00000D+03, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.87232D+01, 2.87162D+01
1; 1.10000D+03, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.92796D+01, 2.92188D+01
1; 1.20000D+03, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.98319D+01, 2.97174D+01
2; 2.98150D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.07861D+01, 2.07845D+01
2; 4.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.07861D+01, 2.07851D+01
2; 5.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.07861D+01, 2.07857D+01
2; 6.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.07861D+01, 2.07863D+01
2; 7.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.07861D+01, 2.07869D+01
2; 8.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.07861D+01, 2.07875D+01
2; 8.76000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.07861D+01, 2.07879D+01
2; 9.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.07861D+01, 2.07881D+01
2; 1.00000D+03, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.07861D+01, 2.07887D+01
2; 1.10000D+03, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.07903D+01, 2.07893D+01
2; 1.20000D+03, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.07945D+01, 2.07899D+01
3; 2.98150D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.75017D+01, 1.74919D+01
3; 4.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.79786D+01, 1.79971D+01
3; 5.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.82213D+01, 1.82301D+01
3; 6.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.83636D+01, 1.83614D+01
3; 7.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.84514D+01, 1.84447D+01
3; 8.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.85100D+01, 1.85024D+01
3; 8.76000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.85435D+01, 1.85360D+01
3; 9.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.85519D+01, 1.85453D+01
3; 1.00000D+03, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.85811D+01, 1.85788D+01
3; 1.10000D+03, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.86021D+01, 1.86064D+01
3; 1.20000D+03, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.86188D+01, 1.86298D+01
4; 2.98150D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.96815D+01, 1.96757D+01
4; 4.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.01376D+01, 2.01499D+01
4; 5.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.03635D+01, 2.03663D+01
4; 6.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.04890D+01, 2.04865D+01
4; 7.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.05644D+01, 2.05612D+01
4; 8.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.06146D+01, 2.06117D+01
4; 8.76000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.06439D+01, 2.06403D+01
4; 9.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.06522D+01, 2.06481D+01
4; 1.00000D+03, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.06773D+01, 2.06757D+01
4; 1.10000D+03, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.06941D+01, 2.06977D+01
4; 1.20000D+03, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.07108D+01, 2.07157D+01
5; 2.98150D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.93050D+01, 1.92925D+01
5; 4.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.99326D+01, 1.99381D+01
5; 5.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.02296D+01, 2.02316D+01
5; 6.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.03970D+01, 2.03935D+01
5; 7.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.04974D+01, 2.04932D+01
5; 8.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.05644D+01, 2.05599D+01
5; 8.76000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.05978D+01, 2.05972D+01
5; 9.00000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.06104D+01, 2.06073D+01
```

```
 5; 1.00000D+03, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.06439D+01, 2.06427D+01
 5; 1.10000D+03, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.06690D+01, 2.06703D+01
 5; 1.20000D+03, 0.00000D+00, 0.00000D+00, 0.00000D+00, 2.06857D+01, 2.06927D+01
 6; 2.98150D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 3.56895D+01, 3.57347D+01
 7; 2.98150D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.74100D+02, 1.74100D+02
 8; 2.98150D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.20386D+02, 1.20386D+02
 9; 2.98150D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.03374D+02, 1.03374D+02
10; 2.98150D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 8.18307D+01, 8.17856D+01
11; 2.98150D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 3.01750D+05, 3.01750D+05
12; 2.98150D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.10499D+05, 1.10499D+05
13; 2.98150D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 8.71402D+03, 8.71402D+03
14; 2.98150D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 3.83254D+04, 3.82836D+04
15; 8.76000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 0.00000D+00, 5.24141D-08
16; 4.07000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00,-1.00000D+01,-1.00782D+01
16; 4.31000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00,-9.00000D+00,-9.00188D+00
16; 4.57000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00,-8.00000D+00,-7.96718D+00
16; 4.85000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00,-7.00000D+00,-6.98092D+00
16; 5.18000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00,-6.00000D+00,-5.96020D+00
16; 5.54000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00,-5.00000D+00,-4.99065D+00
16; 5.97000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00,-4.00000D+00,-3.99226D+00
16; 6.47000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00,-3.00000D+00,-3.00598D+00
16; 7.08000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00,-2.00000D+00,-2.00126D+00
16; 7.83000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00,-1.00000D+00,-9.93370D-01
16; 8.76000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 0.00000D+00,-1.25036D-11
16; 9.99000D+02, 0.00000D+00, 0.00000D+00, 0.00000D+00, 1.00000D+00, 1.00609D+00
```

Attachment F. UNIT15 and UNIT16 output from PHAS2OPC

UNIT15 output from test run of PHAS20PC using data in Attachment A:

```
15;  876.0000,      38284.
16;  407.0000,     152525.
16;  431.0000,     153119.
16;  457.0000,     153422.
16;  485.0000,     153312.
16;  518.0000,     153529.
16;  554.0000,     153234.
16;  597.0000,     153223.
16;  647.0000,     153060.
16;  708.0000,     153117.
16;  783.0000,     153234.
16;  876.0000,     153134.
16;  999.0000,     153251.
```

UNIT16 output from test run of PHAS20PC using data in Attachment A:

```
15;      38284.,          0.
16;     153180.,        235.
```

Attachment G. Program listing of PHAS20PC

```
C     ------------------------------------------------------------
C
C                        PHAS20 (CODATA VERSION)
C
C          THIS VERSION WILL HANDLE:
C
C                    20 PHASES BEING REFINED,
C                    1500 INDEPENDENT OBSERVATIONS, AND
C                    100 DATA SETS. EACH SET CAN CONTAIN A MEASURED
C                        PROPERTY INVOLVING 16 PHASES.
C
C          VERSION TESTED AND RUN ON IBM 360 & 370  8/15/73 BY HAAS.
C
C          WEIGHT NORMALIZATION BY FISHER 4/11/74
C
C          THIRD LAW TEST AND GIBBS ENERGY FUNCTION BY HAAS, 8/9/74.
C
C          UNITS FOR PROBLEM (JOULES OR CALORIES) BE SELECTED
C             AS OPTION BY HAAS 9/1/76
C
C          FORMAT CHANGES FOR OUTPUT BY HAAS, 9/1/76
C
C          MODIFIED TO ALLOW FOR LAMBDA ANOMALIES BY HAAS, FALL, 1976.
C             DISCONTINUED BY INSERTION OF DUMMY SUBROUTINES,
C             WINTER, 1981-82.  FUNCTION FOR LAMBDA ANOMALIES REINSERTED
C             BY HAAS DURING OCTOBER, 1984.  REFER
C             TO THE SUBROUTINE  LDERIV.
C
C          INSTALLED ON HONEYWELL MULTICS SUMMER 1976 BY HAAS.
C          ONLY MINOR MODIFICATIONS NEEDED.
C
C          INSTALLED ON VAX/VMS 11/780, DECEMBER, 1982 BY HAAS.
C          ONLY MINOR MODIFICATIONS NEEDED.
C
C          DIMENSIONS INCREASED TO INCLUDE 16 PHASES PER REACTION
C             SUMMER, 1982 BY HAAS.
C
C          CALCULATIONS OF DELTA H(REACTION) AT 298 HAVE BEEN CORRECTED
C             TO REFLECT THE REACTION WITH THE MOST STABLE PHASES AT
C             298 AS THE REACTANTS OR PRODUCTS.  JUNE, 1984 BY HAAS.
C
C          REVISED TO ACCEPT THE CODATA THERMAL FUNCTION BY HAAS, 9/31/84.
C
C          REVISED TO USE THE CODATA RECOMMENDED VALUES FOR:
C             R = GAS CONSTANT = 8.31451 J/(MOL K)
C             F = FARADAY = 96485.309 COULOMBS/MOL
C                REFERENCE:  COHEN & TAYLOR, 1986, CODATA BULL 63.
C
C          OPTION TO USE DAMPING FACTORS INSERTED IN ORGLS SUBROUTINE
C             BY HAAS, DEC, 1986.  KI(I) IS VARIED FROM 1 THRU 9 AND CAUSES
C             THE CALCULATED CHANGE IN A PARAMETER TO BE MULTIPLIED
C             BY THE FOLLOWING FACTORS, RESPECTIVELY:
C             1.0, 0.63, 0.40, 0.25, 0.16, 0.10, 0.063, 0.040, AND 0.025.
C
```

```
C          INSTALLED ON IBM PC/AT AND AT&T 6300, JULY, 1986, BY J W BALL.
C          NUMBER OF PHASES, OBSERVATIONS, AND DATASETS REDUCED BY A
C          FACTOR OF ABOUT 7.5 (AVERAGE), TO REDUCE THE OBJECT CODE SIZE
C          TO ABOUT 440K.  OTHERWISE, ONLY MINOR MODIFICATIONS NEEDED.
C
C     ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*1 LABEL,BCD,IMAGE
      CHARACTER*5 ANOM,NOANOM
      CHARACTER*8 UNIT,UNITS,DUMMY
      CHARACTER*23 DATE
      LOGICAL*1 UNDAMP
      LOGICAL*1 LAMBDA,NOLAM,KAPPA
      REAL*4 XMAX,XMIN,YMAX,YMIN,XI,E,TITLE,ACOEF,T,TO,AVAL,RELERR,SERR,
     1  SERRSQ,EBAR,STDEV,ERR,AN
      DIMENSION AVAL(7),SERR(3),SERRSQ(3),EBAR(3),STDEV(3),IMAGE(5000),
     1  XI(1),E(1),PHINV(6),VAL(7),IFMIN(20),TRUE(2)
      DIMENSION COEF(10,100),PNAME(20),TINV(10,100,4),IPHASE(10,100),
     1  NPHASE(100),IKOUNT(100),IGO(100),ISTATE(10,100),NINVER(10,100),
     2  INSTAT(10,100),INVPH(10,100,4),INVSC(10,100),ISPECL(10,100)
      DIMENSION X(4,1500),YO(1500),SIGYO(1500),P(360),KI(360),
     1  DC(360),PD(360),TITLE(20),SIGYON(1500)
      DIMENSION REF(20,100),ERRP(360),IHOLD(20)
      DIMENSION SCINV(2),STCOEF(10),YESNO(2),TK(19,2),TYPE(28),NSCALE(5)
     1  ,LABEL(50)
      DIMENSION DYDC(10),DDDC(10),DLDC(2),DLDDC(2)
      DIMENSION AA(360),ION(20)
      DIMENSION UNITS(4)
      DIMENSION LAMBDA(20),NOLAM(10,100),JLOW(20),JHIGH(20)
      DIMENSION NOANOM(2)
C     ------------------------------------------------------------------
C          COMMON BLOCKS
C     ------------------------------------------------------------------
      COMMON /EARTH/ COEF,PNAME,TINV,IPHASE,NPHASE,IKOUNT,IGO,NSETS,
     1  ISTATE,NINVER,INSTAT,INVPH,INVSC,LISTP,ISPECL
      COMMON /AIR/X,P,DC,TITLE,YO,SIGYO,SIGYON,PD,KI,NC,IC,NV,
     1  NX,IW,NP,NO,ISING,ISTOP,IL,JN,NPARM,NVPARM,NCPARM,NMPARM
      COMMON /FIRE/ REF, ERRP, IWRITE, ICY, IICY, IREDO, IREG, NHOLD,
     1 IHOLD,UNDAMP
      COMMON /WATER/ZERO, ONE, TWO, THREE, FOUR, SIX, R, F,
     1  SCINV, TREF, PREF, STCOEF, DIEO, ADIE, BDIE, THETA, YESNO, TK,
     2  ASTAR, TYPE, NL, NSCALE
      COMMON /WATCHR/LABEL, BCD
      COMMON /TIME/ DATE
      COMMON /SPACE/ DYDC, DDDC, DLDC, DLDDC, SC, TLOW
      COMMON /MAN/ AA, ION
      COMMON /MINERL/RJ,RC,FJ,FC
      COMMON /MINCHR/UNITS
      COMMON /HOTAIR/ RECKON,ITEK
      COMMON /AETHER/ LAMBDA,NOLAM,JLOW,JHIGH
      COMMON /DUMP/ IDUMP
c     ..... next line was added by MKo, 10/10/85 for Prime computer.
      common /ming01/image
      DATA NOANOM/'NOLAM','nolam'/
```

```
      DATA NTYPE/28/
C      THE FOLLOWING THREE CALLS HAVE BEEN REPLACED ON THE PRIME BY CALLS
C      TO THE ROUTINE CALLED ERRPR$.
C     CALL ERRSET(63,1,0,0,0,999)
C     CALL ERRSET(74,1,0,0,0,999)
C     CALL ERRSET(89,1,0,0,0,999)
C     ------------------------------------------------------------------
C     OPEN THE INPUT/OUTPUT FILES--
C          5 -- DATA INPUT FILE (IN USER'S DIRECTORY)
C          6 -- MAIN OUTPUT AREA - OPENED EXTERNAL TO PROGRAM
C          7 -- CARD PUNCH FILE (FROM HAAS, NTS PUB # AD 780 301)
C          8 -- PRINTER PLOTS GO HERE
C          9 -- DO SOME SORT OF 'TABLES' GO HERE??
C         10 -- RESULTS OF READING OF INPUT FILE GO HERE, TO ASSIST
C               TRACING OF DATA ERRORS
C         11 -- OUTPUT DIRECTED HERE WHEN 'ITEK' FLAG IS TURNED 'ON'
C         15 -- OUTPUT DIRECTED HERE WHEN 'ITEK' FLAG IS TURNED 'ON'
C         16 -- OUTPUT DIRECTED HERE WHEN 'ITEK' FLAG IS TURNED 'ON'
C     OUTPUTS 11 AND 15 ARE FILES FOR INPUT TO A PLOTTER AS FOLLOWS:
C     FILE 11:
C        "JSET;X(1,I),X(2,I),X(3,I),X(4,I),YO(I),YC"
C     THE SEMICOLON AND THE COMMAS ARE INCLUDED IN THE OUTPUT STRING.
C     FILE 15:
C        "JSET;X(1,I),DELH"
C     AGAIN, THE SEMICOLON AND COMMA ARE INCLUDED IN THE OUTPUT STRING.
C        DELH IS THE THIRD-LAW ENTHALPY OF THE REACTION FOR THE OBSERVED
C        GIBBS ENERGY DATA, EQUILIBRIUM CONSTANT, OR ELECTROCHEMICAL
C        POTENTIAL.
C     OUTPUT FILE 16 GIVES A QUICK FILE OF THE DATA SOURCE, THE
C        AVERAGE ENTHALPY OF REACTION FOR THE DATA SET, AND THE
C        MEAN SCATTER OF THE ENTHALPY OF REACTION ABOUT THE AVERAGE.
C     ------------------------------------------------------------------
C     OPEN(UNIT=7,FILE='UNIT7.OUT')
      OPEN(UNIT=8,FILE='PLOTS.OUT')
      OPEN(UNIT=9,FILE='UNIT9.OUT')
      OPEN(UNIT=10,FILE='UNIT10.OUT')
      OPEN(UNIT=11,FILE='UNIT11.OUT')
      OPEN(UNIT=15,FILE='UNIT15.OUT')
      OPEN(UNIT=16,FILE='UNIT16.OUT')
C     ------------------------------------------------------------------
C     READ IN AND STORE INPUT DATA--
C          DATE -- TODAY'S DATE
C          NREG -- NUMBER OF SEPARATE PROBLEMS IN THE DATA SET
C     ------------------------------------------------------------------
      CALL CVDATE
C     OPEN(UNIT=5,STATUS='OLD')
      READ (5,5600) DUMMY
      READ (5,5700) NREG
      DO 5400 IREG=1,NREG
C     ------------------------------------------------------------------
C     INITIALIZE CONSTANTS
C     WRITE DATE
C     ------------------------------------------------------------------
      SSQRSG=ZERO
```

```
      NCPARM=10
      NMPARM=3
      NVPARM=5
      NPARM=NCPARM+NMPARM+NVPARM
      NX=4
      IICY=0
      KOUNT=0
      IWRITE=1
      WRITE (10,6600) DATE
C
C     ----------------------------------------------------------------
C     READ IN AND STORE --
C          TITLE (20) -- TITLE FOR CORRELATION
C          IDO -- 1 FOR REGRESSION AND ERROR PLOTS
C                 2 FOR REGRESSION ONLY
C                 3 FOR ERROR PLOTS ONLY
C          NC -- NUMBER OF CYCLES. GENERALLY 2 IS SUFFICIENT.
C                           SUBROUTINE TEST WILL TERMINATE REGRESSION
C                           IF THE RELATIVE CHANGE IN SUCCESSIVE
C                           PARAMETERS IS LESS THAN THAN 1.0D-8.
C          IW -- 0 FOR WEIGHTED DATA
C                 1 FOR UNWEIGHTED DATA
C          IL -- 0 FOR CALCULATED RESULTS AT PLOT TIME ONLY
C                1 FOR CALCULATED RESULTS WITH EACH CYCLE
C          IFMOUT -- PUNCH FORMAT FOR OUTPUT OF PARAMETERS--
C               0 GIVES PARAMETERS IN 6D12.5 FORMAT
C               1 GIVES PARAMETERS IN 6A8 FORMAT.
C          UNIT -- UNITS USED IN CALCULATIONS ('JOULES' OR 'CALORIES')
C          RECKON -- PUNCHED OUTPUT NEEDED FOR PROGRAM 'RECKON'
C               'RECKON  ' PRODUCES PUNCH.
C               'NORECKON' OR BLANK PRODUCES NO PUNCH.
C          ITEK -- PRINTS FILES CONTAINING THE FOLLOWING:
C               11 -- 'JSET;X(1,I),X(2,I),X(3,I),X(4,I),YO(I),YC'
C               15 -- 'JSET;X(1,I),DELH'
C               16 -- DATA SET REFERENCE, NUMBER OF OBSERVATIONS,
C                     AVERAGE AND MEAN SCATTER OF 3RD-LAW ENTHALPY
C               THESE FILES ARE FOR USE WITH PLOTTERS AND CONTAIN
C               THE SEMICOLON AND COMMAS TO SEPARATE THE DATA.
C          LISTP -- NUMBER OF PHASES IN CORRELATION
C          ICY -- NUMBER OF STEPS IN STEP-BACKWARD ELIMINATION.
C               DEFAULT IS 0.
C          NHOLD -- NUMBER OF PARAMETERS TO BE KEPT WITHIN THE
C               REGRESSION EVEN THOUGH 'TEST' WILL INDICATE THEY ARE
C               NON-SIGNIFICANT.  THIS IS IMPORTANT FOR THE
C               TWO CONSTANTS OF INTEGRATION, a(9) AND a(10) FOR EACH
C               PHASE.
C          IDAMP -- SET TO 1 IF YOU WANT TO CAUSE UNDAMPING OF A
C                   REFINEMENT DURING SUCCESSIVE CYCLES.  NOT NEEDED IF
C                   DAMPING IS NOT USED.
C                   SET TO 0 IF THE SAME DAMPING IS TO BE USED THRU
C                   THE REFINEMENT.
C          PNAME(LISTP) -- 8-CHARACTER LABEL FOR EACH PHASE IN
C                           CORRELATION
C          ION(LISTP) -- FLAG WHICH INDICATES THE SPECIES IS A
```

```
C              REFERENCE ELEMENT (1), A COMPOUND (0), OR AN ION (-1).
C          LAMBDA(LISTP) -- LOGICAL ARGUMENT TO SPECIFY WHETHER CP
C                           FUNCTION FOR PHASE HAS LAMBDA ANOMALY.
C          JLOW(LISTP) --  EXPONENT IN THE EMPIRICAL FUNCTION FOR THE
C               CONTRIBUTION TO THE HEAT CAPACITY OF A PHASE WITH A
C               LAMBDA ANOMALY.
C          JHIGH(LISTP) --  EXPONENT IN THE EMPIRICAL FUNCTION FOR THE
C               CONTRIBUTION TO THE HEAT CAPACITY OF A PHASE WITH A
C               LAMBDA ANOMALY.
C          NSETS -- NUMBER OF DATA SETS IN CORRELATION
C     ------------------------------------------------------------------
      READ (5,5800) (TITLE(I),I=1,20)
      READ (5,10000)IDO,NC,IW,IL,IFMOUT,UNIT,RECKON,ITEK
      UNDAMP=.TRUE.
      READ (5,5700) LISTP,ICY,NHOLD,IDAMP
      IF(IDAMP.EQ.1)  UNDAMP=.FALSE.
      READ (5,5600) (PNAME(M),M=1,LISTP)
      READ (5,5900) (ION(I),I=1,LISTP)
      READ (5,10100)(LAMBDA(I),JLOW(I),JHIGH(I),I=1,LISTP)
      READ (5,5700) NSETS
      WRITE (10,6700) (TITLE(I),I=1,20)
      WRITE (10,6800) (PNAME(M),M=1,LISTP)
      WRITE (10,6500)
      WRITE (10,6900) (TITLE(I),I=1,20),DATE
      WRITE (10,7000)
C     WRITE (7,5800) (TITLE(I),I=1,20)
C     WRITE (7,5700) LISTP
C     WRITE (7,5600) (PNAME(M),M=1,LISTP)
C     WRITE (7,10100)(LAMBDA(I),JLOW(I),JHIGH(I),I=1,LISTP)
      DO 1 I=1,4
      IF(UNITS(I).EQ.UNIT)  GO TO 2
    1 CONTINUE
      I=1
    2 CONTINUE
      GO TO (3,3,4,4),I
    3 CONTINUE
      R=RJ
      F=FJ
      GO TO 5
    4 CONTINUE
      R=RC
      F=FC
    5 CONTINUE
C     ------------------------------------------------------------------
C     READ IN AND STORE INPUT DATA FOR EACH DATA SET --
C          REF (10,J) -- REFERENCE FOR 'J'TH DATA SET (DATA SET
C                        IDENTIFICATION)
C          NPHASE(J) -- NUMBER OF PHASES IN REACTION, SET TO 1 WHEN
C                        DATA FOR ONE PHASE ONLY.
C          IKOUNT(J) -- NUMBER OF OBSERVATIONS IN DATA SET
C          IGO(J) -- DATA TYPE,
C                        FOR HEAT CAPACITY, IGO(J) IS  1
C                            ENTROPY                   2
C                            ENTHALPY                  3
```

```
C                                GIBBS ENERGY              4
C                                LOG(10) K                 5
C                                CELL DATA                 6
C                                DIFFERENTIAL HEAT,
C                                   H(T)-H(To)             7
C                              FOR ANY DATA NOT INCLUDED IN TYPES 1
C                                THROUGH 7, SET IGO(J) TO ANY INTEGER
C                                8 THRU 20.  SUPPLY SUBROUTINE  UNIQUE
C                                TO CALCULATE YC (AND THE DERIVATIVES
C                                DC(I) IF JN (OR JDFLAG) = ZERO) FROM
C                                THE  CURRENT VALUES OF THE PARAMETERS.
C                                IF MORE THAN ONE TYPE OF DATA ARE TO BE
C                                CONSIDERED, THESE ARE KEYED BY DIFFERENT
C                                VALUES OF IGO(J).
C          ITFACT -- THE VALUE IS 1 IF TEMPERATURE IS IN KELVINS,
C               0 IF THE TEMPERATURE IS IN CENTIGRADE DEGREES.
C          IPARA -- PROGRAM IS SET FOR UNITS IN JOULES OR CALORIES AS
C               SPECIFIED, VOLTS, OR LOG(10)K IN MOLAL UNITS
C               IF SO, SET IPARA TO 1
C               IF NOT, SET IPARA TO 0.  PROGRAM MULTIPLIES YO(I) AND
C               SIGYO(I)BY PARA TO BE READ LATER.
C          ISIG -- 0 IF WEIGHTS ARE RELATIVE.
C               1 IF WEIGHTS ARE ABSOLUTE.
C               (ABSOLUTE WEIGHTS ARE USED IN CALCULATIONS.)
C          I4X -- SET TO 1 IF 4 INDEPENDENT PARAMETERS ARE TO BE
C                 READ.  PLACE X(3,I) AND X(4,I) ON THE SECOND CARD
C                 IN 2D12.5 FORMAT
C                 SET TO 0 FOR ALL OTHER CASES.
C     ------------------------------------------------------------------
      DO 1800 J=1,NSETS
      KAPPA=.FALSE.
      READ (5,5601) (REF(I,J),I=1,15)
      WRITE (10,7100)
      WRITE (10,7200) ASTAR
      WRITE (10,7100)
      WRITE (10,7300) J,(REF(I,J),I=1,15)
      READ (5,5700) NPHASE(J),IKOUNT(J),IGO(J),ITFACT,IPARA,ISIG,I4X
      IF(IGO(J).NE.29)  GO TO 50
      KAPPA=.TRUE.
      IGO(J)=26
   50 CONTINUE
      WRITE (10,7400) NPHASE(J),IKOUNT(J),TYPE(IGO(J))
      NPHAS=NPHASE(J)
C     ------------------------------------------------------------------
C     FOR EACH DATA SET, READ AND STORE FOR ONE PHASE AT A TIME,
C          ANAME -- 8-CHARACTER PHASE LABEL. MUST BE THE SAME AS
C                          GIVEN IN PNAME(LISTP).
C          COEF(I,J) -- COEFFICIENT IN REACTION (+1.0D0 IF NOT
C                          REACTION)
C          ISPECL(I,J) -- FLAG USED TO ALLOW FOR SPECIAL CASES IN
C                          HEATS OF SOLUTION AND SOME VOLUME FUNCTIONS.
C          ANOM(I,J) --  INSERT 'NOLAM' IF THE CONTRIBUTION TO THE
C                         THERMAL FUNCTION IS TO BE INGNORED FOR THIS
C                         DATA SET.
```

```
C          ISTATE(I,J) -- FOR ELEMENTS IN LOWEST TEMPERATURE MODIFICA-
C                            TION, SET TO 1
C                           FOR COMPOUNDS AND ELEMENTS IN HIGHER TEMPER-
C                            ATURE MODIFICATIONS, SET TO 0
C          NINVER(I,J) -- NUMBER OF INVERSIONS(EQUALS NUMBER OF LOWER
C                            TEMPERATURE MODIFICATIONS BEING CONSIDERED
C                            FOR COMPONENT.)
C     ------------------------------------------------------------------
      DO 700 I=1,NPHAS
      READ (5,6000) ANAME,COEF(I,J),ISPECL(I,J),ANOM,ISTATE(I,J),
     1              NINVER(I,J)
C     ------------------------------------------------------------------
C     LOCATE ANAME IN LIST  PNAME(LISTP) AND ASIGN THE INDEX TO
C          IPHASE(I,J).
C     ------------------------------------------------------------------
      DO 100 K=1,LISTP
      IF (ANAME.EQ.PNAME(K)) GO TO 200
  100 CONTINUE
      GO TO 5200
  200 IPHASE(I,J)=K
      NOYES=(ISTATE(I,J)+3)/2
      ACOEF=COEF(I,J)
      WRITE (10,7500) PNAME(IPHASE(I,J)),ACOEF,YESNO(NOYES),NINVER(I,J)
      NOLAM(I,J)=.FALSE.
      IF((ANOM.EQ.NOANOM(1)).OR.(ANOM.EQ.NOANOM(2)))  NOLAM(I,J)=.TRUE.
      IF (NINVER(I,J).EQ.0) GO TO 700
C     ------------------------------------------------------------------
C     IF (NINVER(I,J).GT.0) READ AND STORE,
C          TINV(NINVER(I,J)) -- INVERSION TEMPERATURES, KELVINS
C          INSTAT(I,J) -- FOR ELEMENTS, ASSIGN 1
C                            FOR COMPOUNDS, ASSIGN 0
C          INVSC(I,J) -- IF UPON INVERSION THERE IS A STOICHIOMETRY
C               CHANGE (EG-- 2 FECL3 = FE2CL6), THEN FIND THE APPRO-
C               PRIATE REACTANT COEFFICIENT (HERE '2') IN THE VECTOR
C               STCOEF AND ENTER HERE.  DEFAULT GIVES A COEFFICIENT OF
C               '1'.
C          PHINV(NINVER(I,J)+1) -- PHASE NAME FOR ALL MODIFICATIONS,
C                            BEGINNING WITH LOWEST TEMPERATURE MODIFICA-
C                            TION FIRST.  NAME MUST CONFORM WITH NAMES
C                            IN PNAME(LISTP).
C     ------------------------------------------------------------------
      KINVER=NINVER(I,J)
      READ (5,6100) (TINV(I,J,K),K=1,KINVER)
      KINVER=KINVER+1
      READ (5,6200) INSTAT(I,J),INVSC(I,J),(PHINV(K),K=1,KINVER)
      IF (INVSC(I,J).EQ.0) INVSC(I,J)=1
C     ------------------------------------------------------------------
C     LOCATE PHINV(NINVER(I,J)+1) IN LIST PNAME(LISTP) AND ASIGN THE
C          INDEX TO INVPH(I,J).
C     ------------------------------------------------------------------
      DO 500 K=1,KINVER
      DO 300 L=1,LISTP
      IF (PHINV(K).EQ.PNAME(L)) GO TO 400
  300 CONTINUE
```

```
      GO TO 5200
  400 INVPH(I,J,K)=L
  500 CONTINUE
      KINVER=KINVER-1
      WRITE (10,7600)
      ACOEF=1.0
      DO 600 K=1,KINVER
      IF (K.EQ.KINVER) ACOEF=STCOEF(INVSC(I,J))
      T=TINV(I,J,K)
      WRITE (10,7700) ACOEF,PNAME(INVPH(I,J,K)),PNAME(INVPH(I,J,K+1)),T
  600 CONTINUE
      IF (INSTAT(I,J).EQ.10) WRITE (1,7800) PNAME(INVPH(I,J,1))
  700 CONTINUE
C     ------------------------------------------------------------------
C     INITIALIZE KO, KOUNT,AND IKOUNT(I). AFTER DATA IS STORED,
C          IKOUNT(I) CONTAINS THE LOCATION OF THE LAST ITEM IN THE
C          'J'TH DATA SET.
C     ------------------------------------------------------------------
      KO=KOUNT+1
      KOUNT=KOUNT+IKOUNT(J)
      IKOUNT(J)=KOUNT
      WRITE (10,7900)
      WRITE (10,8000) TYPE(IGO(J))
C     ------------------------------------------------------------------
C     READ AND STORE THE DATA IN THE 'J'TH DATA SET,
C          X(1,I) -- TEMPERATURE (SEE TFACT BELOW)
C          TFACT -- 0.0D0 IF TEMPERATURE IN KELVINS
C                   273.15D0 IF TEMPERATURE IN CENTIGRADE DEGREES
C          YO(I)  -- DEPENDENT OBSERVATION
C          PARA -- CONVERSION FACTOR TO CONVERT DATA TO CALORIES OR
C                            VOLTS  (1.0D3 FOR DATA IN KILOCALORIES,
C                            1.0D-3 FOR DATA IN MILLIVOLTS, --OR THE
C                            APPROPIATE CONVERSION FACTOR FOR DATA IN
C                            JOULES, BTU/LB,  QUART*STONES/ACRE,ETC.)
C          SIGYO(I)  -- SEE PROGRAM DESCRIPTION FOR ASSIGNING WEIGHTS.
C          X(2,I) -- FOR DATA TYPES (IGO(J)) FROM 1 THRU 6,
C                       X(2,I) = ZERO.
C                    FOR RELATIVE HEAT CONTENT DATA, X(2,I) IS THE BASE
C                       TEMPERATURE.   DEFAULT IS TREF IN THE BLOCKDATA
C                       ROUTINE.
C                    FOR DATA TYPES TO BE PROGRAMMED BY THE USER,
C                       X(2,I) IS AN OPTIONAL SECOND INDEPENDENT
C                       VARIABLE.
C          X(3,I) -- INDEPENDENT PARAMETER #3.  FOR ALL PREPROGRAMMED
C                    FUNCTIONS, THIS IS THE PRESSURE OF THE OBSERVATION.
C          X(4,I) -- INDEPENDENT PARAMETER #4.  FOR ALL PREPROGRAMMED
C                    FUNCTIONS, THIS IS THE REFERENCE PRESSURE.  THE
C                    DEFAULT IS GIVEN BY PREF IN THE BLOCKDATA ROUTINE.
C     ------------------------------------------------------------------
      DO 1700 I=KO,KOUNT
      IF(I4X.EQ.1)  GO TO 710
      READ (5,6100) X(1,I),TFACT,YO(I),PARA,SIGYO(I),X(2,I)
      GO TO 720
  710 CONTINUE
```

```
      READ (5,6100)X(1,I),TFACT,YO(I),PARA,SIGYO(I),X(2,I),X(3,I),X(4,I)
  720 CONTINUE
      IF(KAPPA)  YO(I)=1.0D0/YO(I)
      IF ((IW.EQ.0).AND.(SIGYO(I).EQ.0.0D0).AND.(ISIG.EQ.0)) SIGYO(I)=
     1      1.0D3*YO(I)
      IF (ITFACT.EQ.1) GO TO 1000
C     ------------------------------------------------------------------
C          IF ITFACT = 0     X(1,I) = X(1,I) + TFACT
C     ------------------------------------------------------------------
      X(1,I)=X(1,I)+TFACT
      IF ((IGO(J).EQ.7).OR.(IGO(J).EQ.3).OR.(IGO(J).GE.21)) GO TO 800
      IF((IGO(J).EQ.4).OR.(IGO(J).EQ.5).OR.(IGO(J).EQ.20))  GO TO 900
      X(2,I)=0.0D0
      GO TO 900
  800 X(2,I)=X(2,I)+TFACT
  900 CONTINUE
 1000 CONTINUE
      IF (IPARA.EQ.1) GO TO 1100
C     ------------------------------------------------------------------
C     IF IPARA = 0     YO(I) = YO(I)*PARA
C     ------------------------------------------------------------------
      YO(I)=YO(I)*PARA
 1100 CONTINUE
      IF (IW.EQ.1) GO TO 1300
      IF (ISIG.EQ.1) GO TO 1200
C     ------------------------------------------------------------------
C     IF ISIG = 0        SIGYO(I)= DABS (YO(I))*SIGYO(I)
C     ------------------------------------------------------------------
      SIGYO(I)=DABS(SIGYO(I)*YO(I))
      GO TO 1400
 1200 CONTINUE
C     ------------------------------------------------------------------
C     IF IPARA = 0      SIGYO(I)=SIGYO(I)*PARA
C     ------------------------------------------------------------------
      IF (IPARA.EQ.0) SIGYO(I)=DABS(SIGYO(I)*PARA)
      GO TO 1400
 1300 CONTINUE
      SIGYO(I)=DABS(YO(I)*1.0D-2)
 1400 CONTINUE
      IF (SIGYO(I).EQ.0.0D0) SIGYO(I)=1.0D3*YO(I)
      IF (SIGYO(I).EQ.0.0D0)  SIGYO(I)=1.0D9
      IF ((IGO(J).EQ.7).AND.(X(2,I).EQ.0.0D0)) X(2,I)=298.15D0
      T=X(1,I)
      TO=X(2,I)
      IF (TO.NE.0.0) GO TO 1500
      WRITE (10,8100) I,T,YO(I),SIGYO(I)
      GO TO 1600
 1500 CONTINUE
      WRITE (10,8200) I,T,TO,YO(I),SIGYO(I)
 1600 CONTINUE
      SSQRSG=SSQRSG+(1.D0/SIGYO(I))**2
 1700 CONTINUE
 1800 CONTINUE
      FKOUNT=DBLE(KOUNT)
```

```
      DO 1900 I=1,KOUNT
      SIGYON(I)=SIGYO(I)*DSQRT(SSQRSG/FKOUNT)
 1900 CONTINUE
C     ------------------------------------------------------------------
C     THE DATA HAS NOW BEEN STORED. INITIALIZE CONSTANTS NEEDED FOR
C          ORGLSC.
C     ------------------------------------------------------------------
      NO=KOUNT
      NP=LISTP*NPARM
      IW=0
C     ------------------------------------------------------------------
C     READ AND STORE,
C          IFMIN -- INPUT FORMAT FOR P(LISTP*NPARM)
C               0 INDICATES (6D12.5/D12.5)
C               1 INDICATES (7A8)
C               THE LATTER FORMAT HAS NO ROUNDOFF IN DATA TRANSFER.
C          P(LISTP*6)  -- TRIAL PARAMETERS. IF VARIED, MAY BE 0.0D0.
C                         IF NOT VARIED, 0.0D0 OR A
C                         PREDETERMINED VALUE MAY BE USED
C          KI(LISTP*6)  -- 0 IF PARAMETER IS HELD CONSTANT.
C                          1 IF PARAMETER IS TO BE VARIED
C                    KI(I) MAY ALSO BE SET TO AN INTEGER RANGING FROM 1
C                    TO 9.  IF SET TO AN INTEGER GREATER THAN 1,
C                    DAMPING OF THE REFINEMENT WILL OCCUR.  REFER TO
C                    INTRODUCTORY NOTES FOR THE DAMPING FACTORS.
C          IHOLD -- INDICES OF PARAMETERS IN P(LISTP*7) WHICH ARE
C               FORCED TO STAY IN REGRESSION DESPITE  'TEST' DETERMINA-
C               TION THAT THESE PARAMETERS MAY BE NON-SIGNIFICANT.
C     ------------------------------------------------------------------
      READ (5,6300) (IFMIN(I),I=1,LISTP)
      DO 2200 I=1,LISTP
      JO=1+NPARM*(I-1)
      JN=JO+NPARM-1
      IF (IFMIN(I).EQ.1) GO TO 2000
      READ (5,6110) (P(J),J=JO,JN)
      GO TO 2100
 2000 CONTINUE
      READ (5,5610) (P(J),J=JO,JN)
 2100 CONTINUE
      DO 2200 J=JO,JN
      AA(J)=P(J)
 2200 CONTINUE
      READ (5,6300) (KI(I),I=1,NP)
      IF (NHOLD.NE.0) READ (5,5700) (IHOLD(I),I=1,NHOLD)
C     ------------------------------------------------------------------
C     CLOSE AND DELETE UNIT 10.  WHEN THE INPUT FILE HAS BEEN READ,
C        THIS FILE IS OF NO FURTHER USE.  RELEASE THE SPACE TO OTHERS.
C     ------------------------------------------------------------------
      CLOSE (UNIT=10, STATUS='DELETE')
C     ------------------------------------------------------------------
C     BYPASS REGRESSION IF IDO IS 3.
C     ------------------------------------------------------------------
      IF (IDO.EQ.3) GO TO 2700
      IDUMP=6
```

```
 2300 CONTINUE
      NV=0
      DO 2400 K=1,NP
      IF (KI(K).EQ.0) GO TO 2400
      NV=NV+1
 2400 CONTINUE
      JN=0
C     ------------------------------------------------------------------
C     ENTER ORGLSC AND REFINE PARAMETERS
C     ------------------------------------------------------------------
      CALL ORGLSC
      IF (ISING.NE.0) GO TO 2700
C     ------------------------------------------------------------------
C     PUNCH REFINED PARAMETERS ON CARD, ONE PHASE AT A TIME AND INCLUDE
C          THE PHASE NAME ON THE CARD WITH THE RELATED PARAMETERS.
C     ------------------------------------------------------------------
      DO 2600 I=1,LISTP
      JO=1+NPARM*(I-1)
      IF (IFMOUT.EQ.1) GO TO 2500
      JN=JO+5
C     WRITE (7,9600) (P(J),J=JO,JN),PNAME(I)
      JO=JN+1
      JN=JN+6
C     WRITE (7,9600) (P(J),J=JO,JN),PNAME(I)
      JO=JN+1
      JN=JN+6
C     WRITE (7,9700) (P(J),J=JO,JN),PNAME(I),DATE
      GO TO 2600
 2500 CONTINUE
      JN=JO+NPARM-1
C     WRITE (7,9800) (P(J),J=JO,JN),PNAME(I),DATE
 2600 CONTINUE
      IF (IDO.EQ.2) GO TO 5100
 2700 CONTINUE
      IDUMP=8
      IF (IICY.GE.1) GO TO 2800
C     ------------------------------------------------------------------
C     SET UP AND PRINT ERROR PLOTS.
C     READ IN AND STORE
C          NHL -- NUMBER OF HORIZONTAL DIVISIONS ON ORDINATE LESS 1.
C          NSBH -- NUMBER OF HORIZONTAL LINES PER DIVISION.
C          NVL -- NUMBER OF VERTICAL DIVISIONS ON ABSCISSA LESS 1.
C          NSBV -- NUMBER OF VERTICAL LINES PER DIVISION.
C          XMAX -- MAXIMUM TERPERATURE ON ABSCISSA.
C          XMIN -- MINIMUM TERPERATURE ON ABSCISSA.
C          YMAX -- MAXIMUM ERROR ON ORDINATE.
C               YMIN = - YMAX.
C     ------------------------------------------------------------------
      READ (5,5700) NHL,NSBH,NVL,NSBV
      READ (5,6400) XMAX,XMIN,YMAX
      YMIN=-YMAX
 2800 CONTINUE
      IF ((ISING.NE.0).AND.(IDO.NE.3)) GO TO 5300
      IN=0
```

```
C     ------------------------------------------------------------------
C     CYCLE THROUGH EACH DATA SET.
C     ------------------------------------------------------------------
      DO 4400 J=1,NSETS
      WRITE (8,7100)
      WRITE (8,6900) (TITLE(I),I=1,20),DATE
      IF ((((IGO(J)-1)/3).EQ.1).OR.(IGO(J).EQ.28))  GO TO 2900
      WRITE (8,8400) TYPE(IGO(J))
      GO TO 3000
 2900 CONTINUE
      WRITE (8,8500) TYPE(IGO(J))
 3000 CONTINUE
C     ------------------------------------------------------------------
C     INITIALIZE THE PLOT IMAGE.
C     ------------------------------------------------------------------
      CALL PLOT1(NSCALE,NHL,NSBH,NVL,NSBV)
      CALL PLOT2(IMAGE,XMAX,XMIN,YMAX,YMIN)
C     ------------------------------------------------------------------
C     INITIALIZE CONSTANTS AND COUNTERS.
C     ------------------------------------------------------------------
      IO=IN+1
      IN=IKOUNT(J)
      DO 3100 I=1,3
      SERR(I)=ZERO
      SERRSQ(I)=ZERO
 3100 CONTINUE
      SDELH=ZERO
      SDELHS=ZERO
      CALL PUTOUT(J)
C     ------------------------------------------------------------------
C      CALCULATE AND PRINT YC.  CALCULATE AND SUM THE DIFFERENCE (ERR),
C        PERCENT ERROR (RELERR), AND THE WEIGHTED DIFFERENCE (E).
C     ------------------------------------------------------------------
      DO 4000 I=IO,IN
      CALL EAFWC (YC,I,JSET)
      ERRDP=YO(I)-YC
      ERR=ERRDP
      IF (YO(I).EQ.ZERO) GO TO 3200
      RELERR=ERRDP*1.0D2/DABS(YO(I))
      GO TO 3300
 3200 CONTINUE
      RELERR=ZERO
 3300 CONTINUE
      E(1)=ERR/SIGYO(I)
      XI(1)=X(1,I)
      TO=X(2,I)
      IF ((((IGO(J)-1)/3).EQ.1).OR.(IGO(J).EQ.28))  GO TO 3400
      WRITE (8,8600)XI(1),TO,X(3,I),X(4,I),YO(I),YC,SIGYO(I),ERR,RELERR,
     1            E(1)
      GO TO 3900
C     ------------------------------------------------------------------
C     CALCULATE THE ENTHALPY OF REACTION AT 298.15 K USING THE GIBBS
C         ENERGY FUNCTION AND THE DATA FOR EACH OBSERVATION.
C     ------------------------------------------------------------------
```

```
 3400 CONTINUE
      YY=YO(I)
      IF(IGO(J).EQ.28)  GO TO 3450
      IGOES=IGO(J)-3
      GO TO (3500,3600,3700), IGOES
 3450 CONTINUE
      YY=YY+PTGEF(X(1,I),X(3,I),J)
 3500 CONTINUE
      DELH=YY-GEF(X(1,I),J)*X(1,I)
      GO TO 3800
 3600 CONTINUE
      DELH=-(R*YY+GEF(X(1,I),J))*X(1,I)
      GO TO 3800
 3700 CONTINUE
      DELH=-(F*YY+GEF(X(1,I),J)*X(1,I))
 3800 CONTINUE
C     ------------------------------------------------------------------
C     CORRECT THE ENTHALPY OF REACTION AT 298.15 K, AS CALCULATED FROM
C         THE GIBBS ENERGY FUNCTION, FOR ANY HIGH TEMPERATURE
C         INVERSIONS.
C     ------------------------------------------------------------------
      CALL YDERIV(TREF,3)
      LPHASE=NPHASE(J)
      DH298=ZERO
      DO 3820 L=1,LPHASE
      IF(NINVER(L,J).EQ.0)  GO TO 3820
      IND298=INVPH(L,J,1)
      INDLST=INVPH(L,J,(NINVER(L,J)+1))
      DH=ZERO
      DO 3810 K=1,NCPARM
      INDEX=(IND298-1)*NPARM+K
      DH=DH+AA(INDEX)*DYDC(K)
 3810 CONTINUE
      DO 3811 K=1,NCPARM
      INDEX=(INDLST-1)*NPARM+K
      DH=DH-AA(INDEX)*DYDC(K)
 3811 CONTINUE
      DH298=DH298+COEF(L,J)*DH
 3820 CONTINUE
      DELH=DELH+DH298
      SDELH=SDELH+DELH
      SDELHS=SDELHS+DELH*DELH
      WRITE (8,8600)XI(1),TO,X(3,I),X(4,I),YO(I),YC,SIGYO(I),ERR,RELERR,
     1             E(1),DELH
      IF (ITEK.GE.1) WRITE (15,10300)JSET,X(1,I),DELH
 3900 CONTINUE
C     ------------------------------------------------------------------
C     ENTER THE POINT ON THE PLOT IMAGE.
C     SUM THE ERRORS.
C     ------------------------------------------------------------------
      CALL PLOT3(IMAGE,BCD,XI,E,1)
      SERR(1)=SERR(1)+ERR
      SERR(2)=SERR(2)+RELERR
      SERR(3)=SERR(3)+E(1)
```

```
      SERRSQ(1)=SERRSQ(1)+ERR*ERR
      SERRSQ(2)=SERRSQ(2)+RELERR*RELERR
      SERRSQ(3)=SERRSQ(3)+E(1)*E(1)
C     ------------------------------------------------------------------
C     CALCULATE AND WRITE THE STANDARD ERROR OF ESTIMATE FOR THE
C        DIFFERENCE, THE PERCENT ERROR, AND THE WEIGHTED DIFFERENCE.
C     ------------------------------------------------------------------
 4000 CONTINUE
      NDATA=1+(IN-IO)
      AN=NDATA
      DO 4100 I=1,3
      EBAR(I)=SERR(I)/AN
      IF(AN.EQ.1.)  GO TO 4050
      EBARSQ=EBAR(I)*EBAR(I)
      ANSESQ=SERRSQ(I)/AN
      IF(ANSESQ.LE.EBARSQ)  GO TO 4050
      STDEV(I)=SQRT(ANSESQ-EBARSQ)
      GO TO 4100
 4050 CONTINUE
      STDEV(I)=0.0
 4100 CONTINUE
      WRITE (8,7900)
      IF ((((IGO(J)-1)/3).EQ.1).OR.(IGO(J).EQ.28)) GO TO 4200
      WRITE (8,8700) (EBAR(I),I=1,3),(STDEV(I),I=1,3),NDATA
      GO TO 4300
 4200 CONTINUE
      DN=DBLE(NDATA)
      DELHAV=SDELH/DN
      IF(DN.EQ.1.0D0)  GO TO 4210
      DHAVSQ=DELHAV*DELHAV
      ANSDHS=SDELHS/DN
      IF(ANSDHS.LE.DHAVSQ)  GO TO 4210
      DELHBA=DSQRT(ANSDHS-DHAVSQ)
      GO TO 4220
 4210 CONTINUE
      DELHBA=0.0D0
 4220 CONTINUE
      WRITE (8,8800)(EBAR(I),I=1,3),DELHAV,(STDEV(I),I=1,3),DELHBA,NDATA
      IF (ITEK.GE.1) WRITE (16,10400)JSET,DELHAV,DELHBA
 4300 CONTINUE
C     ------------------------------------------------------------------
C     CALL PUTOUT TO CALCULATE AND WRITE THE COMPLETE SET OF CONSTANTS
C        FOR THE PHASES AND FOR THE DATA SET.  BYPASS THE LATTER SET
C        OF CONSTANTS IF IGO(J) IS GREATER THAN OR EQUAL TO 7 OR IF
C        THE J-TH DATA SET HAS ONLY ONE PHASE.
C     ------------------------------------------------------------------
      CALL PUTIT(J)
C     ------------------------------------------------------------------
C      PRINT THE PLOT IMAGE FOR THE J-TH DATA SET.
C     ------------------------------------------------------------------
      WRITE (8,6500)
      WRITE (8,6900) (TITLE(I),I=1,20),DATE
      WRITE (8,8900) J,(REF(I,J),I=1,15)
      CALL PLOT4(NL,LABEL,IMAGE)
```

```
      WRITE (8,9000)
 4400 CONTINUE
C     ------------------------------------------------------------------
C     PUT OUT TABLES FOR EACH SPECIES IN THE LIST PNAME(LISTP).
C        THIS OUTPUT WILL NOT HAVE THE CONTRIBUTIONS DUE TO THE MAGNETIC
C        ANOMALIES.  IT IS TO BE USED WITH CARE.
C     ------------------------------------------------------------------
      SC=1.0D0
      DO 5000 L=1,LISTP
C     ------------------------------------------------------------------
C      WRITE LEGEND FOR EACH TABLE
C     ------------------------------------------------------------------
      WRITE (9,6600) DATE
      WRITE (9,6700) (TITLE(I),I=1,20)
      WRITE (9,9100) PNAME(L)
      WRITE (9,9200) (TYPE(I),I=1,7)
      INDEX=1+NPARM*(L-1)
      KOUNT=19
      IF (ION(L).EQ.-1) KOUNT=15
      M=(ION(L)+4)/2
      DO 4900 K=1,KOUNT
      T=TK(K,M)
      DO 4700 J=1,7
      VAL(J)=0.0D0
      IGOES=J
      IF (ION(L).EQ.-1) IGOES=IGOES+7
      IF (IGOES.EQ.14) GO TO 4700
      CALL YDERIV(TK(K,M),IGOES)
      DO 4500 I=1,NCPARM
      VAL(J)=VAL(J)+DYDC(I)*AA(INDEX+I-1)
 4500 CONTINUE
      IF (J.NE.7) GO TO 4700
      CALL YDERIV(TREF,J)
      DO 4600 I=1,NCPARM
      VAL(J)=VAL(J)-DYDC(I)*AA(INDEX+I-1)
 4600 CONTINUE
 4700 CONTINUE
      DO 4800 I=1,7
      AVAL(I)=VAL(I)
 4800 CONTINUE
      WRITE (9,9300) T,(AVAL(I),I=1,7)
 4900 CONTINUE
      JO=INDEX
      JN=JO+NPARM-1
      WRITE (9,7900)
      WRITE (9,9400) (AA(J),J=JO,JN)
      IF (IFMOUT.EQ.1)  GO TO 4950
      JNN=JO+5
C     WRITE (7,9600) (AA(J),J=JO,JNN),PNAME(L)
      JO=JO+6
      JNN=JO+5
C     WRITE (7,9600) (AA(J),J=JO,JNN),PNAME(L)
      JO=JO+6
C     WRITE (7,9710) (AA(J),J=JO,JN),PNAME(L),DATE
```

```
      GO TO 5000
 4950 CONTINUE
C     WRITE (7,9900) (AA(J),J=JO,JN),PNAME(L),DATE
 5000 CONTINUE
      IF (IDO.EQ.3) GO TO 5300
C     ------------------------------------------------------------------
C     IF STEP-BACKWARD ELIMINATION OF NON-SIGNIFICANT PARAMETERS IS
C          IN EFFECT (ICY.NE.0), SET LEAST SIGNIFICANT PARAMETER TO
C          ZERO AND REPEAT REGRESSION.
C     ------------------------------------------------------------------
 5100 IF ((IREDO.EQ.0).OR.(IICY.EQ.ICY)) GO TO 5300
      KI(IREDO)=0
      P(IREDO)=0.0D0
      IICY=IICY+1
      GO TO 2300
 5200 CONTINUE
      WRITE (6,9500) J,ANAME
 5300 CONTINUE
 5400 CONTINUE
C     ------------------------------------------------------------------
C     PROGRAM PRESENTLY SET TO USE IBM PERSONAL COMPUTER PROFESSIONAL
C        FORTRAN SYSTEM CLOCK FUNCTIONS TO GET CURRENT DATE/TIME.
C        GO GET THE CURRENT DATE.
C     ------------------------------------------------------------------
      CALL CVDATE
      WRITE (6,10200)DATE
C     CLOSE (UNIT=7,STATUS='KEEP')
      CLOSE (UNIT=8,STATUS='KEEP')
      CLOSE (UNIT=9,STATUS='KEEP')
      CLOSE (UNIT=5,STATUS='KEEP')
      CLOSE (UNIT=6,STATUS='KEEP')
      CLOSE (UNIT=10,STATUS='KEEP')
      CLOSE (UNIT=11,STATUS='KEEP')
      CLOSE (UNIT=15,STATUS='KEEP')
      CLOSE (UNIT=16,STATUS='KEEP')
      STOP
C     ------------------------------------------------------------------
C     FORMAT STATEMENTS
C     ------------------------------------------------------------------
 5600 FORMAT (10A8)
 5601 FORMAT(15A8)
 5610 FORMAT(7A9)
 5700 FORMAT (16I5)
 5800 FORMAT (20A4)
 5900 FORMAT (40I2)
 6000 FORMAT (A8,2X,D10.3,I5,A5,2I5)
 6100 FORMAT (6D12.5)
 6110 FORMAT (6D15.8)
 6200 FORMAT (I5,I3,9A8/(10A8))
 6300 FORMAT (80I1)
 6400 FORMAT (8E10.3)
 6500 FORMAT (1H1)
 6600 FORMAT (42H1 THESE RESULTS WERE OBTAINED IN A RUN ON ,A23)
 6700 FORMAT (1H0,20A4/1H0)
```

```
6800 FORMAT (60H0     PHASES CONSIDERED IN THIS REGRESSION ARE AS FOLLO
    1WS---/1H0,10X,5A20/(1H ,10X,5A20))
6900 FORMAT (1H ,20A4,25X,A23/1H0)
7000 FORMAT (54H0THE FOLLOWING DATA SETS HAVE BEEN READ IN TO STORAGE:/
    111H0SET NUMBER,10X,9HREFERENCE)
7100 FORMAT (1H0)
7200 FORMAT (1H ,A8)
7300 FORMAT (1H ,I6,4X,15A8)
7400 FORMAT (25H0THE NUMBER OF PHASES IS:,I5/31H THE NUMBER OF OBSERVAT
    1IONS IS:,I5/21H THE TYPE OF DATA IS:,3X,A8/11H0PHASE NAME,10X,11HC
    2OEFFICIENT,9X,10HREF. STATE,10X,10HINVERSIONS/1H )
7500 FORMAT (1H ,1X,A8,11X,F8.3,15X,A3,14X,I6)
7600 FORMAT (1H0,10X,26HINVERSIONS ARE AS FOLLOWS:/1H )
7700 FORMAT (1H ,F15.2,3X,A8,5X,1H=,5X,A8,F20.3)
7800 FORMAT (1H0,10X,A8,5X,58HIS A REFERENCE PHASE AND HAS ZERO GIBBS E
    1NERGY AT 298.15 K/1H0)
7900 FORMAT (1H )
8000 FORMAT (1H ,15X,5HINDEX,21X,11HTEMPERATURE,18X,A8,13X,5HERROR/1H ,
    135X,2HT2,18X,2HT1/1H )
8100 FORMAT (1H ,I20,F20.3,16X,1H-,3X,1P2D20.5)
8200 FORMAT (1H ,I20,2F20.3,1P2D20.5)
8400 FORMAT(1H ,T12,21HINDEPENDENT VARIABLES,
    1       T52,A8,
    1       T74,5HSIGYO,
    1       T99,5HERROR/1H ,
    2       T3,38H--------------------------------------,
    2       T43,26H--------------------------,
    2       T71,11H-----------,
    2       T84,32H--------------------------------/1H ,
    3       T4,6HX(1,I),T14,6HX(2,I),T24,6HX(3,I),T34,6HX(4,I),
    3       T45,8HOBSERVED,T58,10HCALCULATED,
    3       T72,9H(WT)**-.5,
    3       T89,3HO-C,T97,8H%(O-C)/O,T107,9H(O-C)/SIG)
8500 FORMAT(1H ,T12,21HINDEPENDENT VARIABLES,
    1       T52,A8,
    1       T74,5HSIGYO,
    1       T99,5HERROR,
    1       T119,12HDELTA H(298)/1H ,
    2       T3,38H--------------------------------------,
    2       T43,26H--------------------------,
    2       T71,11H-----------,
    2       T84,32H--------------------------------,
    2       T118,14H--------------/1H ,
    3       T4,6HX(1,I),T14,6HX(2,I),T24,6HX(3,I),T34,6HX(4,I),
    3       T45,8HOBSERVED,T58,10HCALCULATED,
    3       T72,9H(WT)**-.5,
    3       T89,3HO-C,T97,8H%(O-C)/O,T107,9H(O-C)/SIG)
8600 FORMAT(1H ,F9.2,3F10.2,1P2D14.5,D13.4,0PF12.4,2F11.4,1PD16.6)
8700 FORMAT(59H0THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE----
    1---,T83,3F11.4/59H0THE STANDARD ERRORS OF ESTIMATE ABOUT THE MEAN
    2ARE--------,T83,3F11.4/31H0THE NUMBER OF OBSERVATIONS IS ,I5)
8800 FORMAT(59H0THE ARITHMETIC MEANS OF THE ERRORS FOR THIS SET ARE----
    1---,T83,3F11.4,1PD16.6/59H0THE STANDARD ERRORS OF ESTIMATE ABOUT T
    2HE MEAN ARE--------,T83,0P3F11.4,1PD16.6/31H0THE NUMBER OF OBSERVA
```

```
     3TIONS IS ,I5)
 8900 FORMAT (1H0,I5,3X,15A8/1H )
 9000 FORMAT (1H0,T35,21HTEMPERATURE (KELVINS))
 9100 FORMAT (1H0,A8)
 9200 FORMAT (1H ,15X,A8,8X,A8,8X,A8,8X,A8,8X,A8,8X,A8,8X,A8,8X,A8/1H )
 9300 FORMAT (1H ,F7.2,2F16.3,2F16.0,2F16.4,F16.0)
 9400 FORMAT (1H0,10X,3H'A',15X,3H'B',15X,3H'C',15X,3H'D',15X,3H'E',15X,
     13H'F',15X,3H'G'/1H0,1P7D18.7/1H0,1P7D18.7/1H0,1P7D18.7)
 9500 FORMAT (39H I,YOUR FRIENDLY COMPUTER, DO NOW QUIT./9H0IN YOUR ,I4,
     146HTH DATA SET YOU HAVE MISSTYPED THE PHASE NAME ,A8)
 9600 FORMAT (1P6D15.8,A8)
 9700 FORMAT (1P6D15.8,A8,A23)
 9710 FORMAT (1P6D15.8,A8,A23,8HCOMPLETE)
 9800 FORMAT (12A9,1X,A8,A23)
 9900 FORMAT (7A9,1H*,A8,A23)
10000 FORMAT(5I5,2A8,I5)
10100 FORMAT(16(L1,2I2))
10200 FORMAT (42H0THESE RESULTS WERE COMPLETED IN A RUN AT ,A23)
10300 FORMAT (I5,1H;,F10.4,1H,,F12.0)
10400 FORMAT (I5,1H;,F12.0,1H,,F12.0)
      END
      BLOCKDATA
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*1 LABEL,BCD
      CHARACTER*8 UNIT,UNITS
      LOGICAL*1 LAMBDA,NOLAM
      DIMENSION SCINV(2),STCOEF(10),YESNO(2),TK(19,2),TYPE(28),NSCALE(5)
     1  ,LABEL(50)
      DIMENSION UNITS(4)
      DIMENSION LAMBDA(20),NOLAM(10,100),JLOW(20),JHIGH(20)
C     ----------------------------------------------------------------
C         COMMON BLOCKS
C     ----------------------------------------------------------------
      COMMON /WATER/ZERO, ONE, TWO, THREE, FOUR, SIX, R, F,
     1  SCINV, TREF, PREF, STCOEF, DIEO, ADIE, BDIE, THETA, YESNO, TK,
     2  ASTAR, TYPE, NL, NSCALE
      COMMON /WATCHR/LABEL, BCD
      COMMON /MINERL/RJ,RC,FJ,FC
      COMMON /MINCHR/UNITS
      COMMON /AETHER/ LAMBDA,NOLAM,JLOW,JHIGH
      DATA UNITS/'joules  ','JOULES  ','calories','CALORIES'/
C     ----------------------------------------------------------------
C      1986 RECOMMENDED VALUES OF THE FUNDAMENTAL PHYSICAL CONSTANTS.
C
C         F = FARADAY
C         R = (MOLAR GAS CONSTANT)*(ln(10.0))
C     ----------------------------------------------------------------
      DATA FJ,FC/9.6485309D4,2.30605423D4/
      DATA RJ,RC/19.144866D0,4.5757329D0/
      DATA ZERO,ONE,TWO,THREE,FOUR,SIX/0.0D0,1.0D0,2.0D0,3.0D0,4.0D0,
     1  6.0D0/
      DATA SCINV/-1.0D0,1.0D0/
      DATA TREF,PREF/2.9815D2,1.01325D0/
      DATA  STCOEF /1.0D0,2.0D0,3.0D0,4.0D0,5.0D0,0.5D0,0.4D0,0.33333333
```

```
     133D0,0.25D0,0.2D0/
      DATA DIEO,ADIE,BDIE,THETA/3.057D2,1.875D-2,-1.2741D1,2.19D2/
      DATA YESNO    /3HNO ,3HYES/
      DATA TYPE     /8HHEAT CAP,8HENTROPY ,8HENTHALPY,8HGIBBS EN,
     1    8HLOG K   ,8HCELL EMF,8HHT2-HT1 ,13*8HSPECIAL ,
     1    8HVOLUME  ,8HDELTA V ,8HERROR!!!,8H(dV/dT)P,
     1    8HERROR!!!,8H(dV/dP)T,8HERROR!!!,8H G(T,P) /
      DATA ASTAR/8H********/
      DATA TK /273.15D0,298.15D0,323.15D0,348.15D0,373.15D0,398.15D0,
     1 423.15D0,448.15D0,473.15D0,498.15D0,523.15D0,548.15D0,573.15D0,
     2 598.15D0,623.15D0,0.0D0,0.0D0,0.0D0,0.0D0,273.15D0,298.15D0,
     3 4.0D2,5.0D2,6.0D2,7.0D2,8.0D2,9.0D2,1.0D3,1.1D3,1.2D3,1.3D3,
     4 1.4D3,1.5D3,1.6D3,1.7D3,1.8D3,1.9D3,2.0D3/
C     DATA LABEL /30HOBS LESS CALC OVER UNCERTAINTY/
      DATA LABEL /'O','B','S',' ','L','E','S','S',' ','C','A','L','C',
     C   ' ','O','V','E','R',' ','U','N','C','E','R','T','A','I','N',
     C   'T','Y',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',
     C   ' ',' ',' ',' ',' ',' ',' ',' '/
      DATA NL/30/
      DATA BCD/'O'/
      DATA NSCALE/5*0/
      END
      SUBROUTINE ORGLSC
C     ------------------------------------------------------------------
C
C                              OR GLS
C
C              A GENERAL FORTRAN LEAST SQUARES PROGRAM
C                                 BY
C                WILLIAM R. BUSING AND HENRI A. LEVY
C                          CHEMISTRY DIVISION
C                    OAK RIDGE NATIONAL LABORATORY
C                        OAK RIDGE, TENNESSEE
C
C     REFERENCE-- BUSING, W. R. AND LEVY, H. A., 1962, OR GLS, A GENERAL
C                     FORTRAN LEAST SQUARES PROGRAM, OAK RIDGE
C                     NATIONAL LABORATORY, OAK RIDGE, TENN.,
C                     ORNL-TM-271.
C
C
C                              ORGLSC
C
C          ORGLSC HAS BEEN ADAPTED FROM BUSING AND LEVY'S OR GLS
C     PROGRAM TO ALLOW USE OF THE STATISTICAL ROUTINES WITHOUT BEING
C     RESTRICTED TO THE RIGID INPUT-OUTPUT FEATURES OF THE PROGRAM.
C     THE CONVERSION HAS PERMITTED CHANGES WHICH CLARIFY THE OUTPUT
C     FOR THIS PECULIAR PROBLEM.
C
C     ADAPTATION BY HAAS, 8/20/72.
C
C     WEIGHT NORMALIZATION BY FISHER 4/11/74
C
C     THE DIMENSION OF THE VECTOR AM IS CALCULATED FROM 0.5*(NV*(NV+3)) = 17390
C     ------------------------------------------------------------------
```

```
      IMPLICIT REAL*8 (A-H,O-Z)
      CHARACTER*23 DATE
      LOGICAL*1 UNDAMP
      LOGICAL*1 LAMBDA,NOLAM
      REAL*4 TITLE,PCTERR,OUT
      DIMENSION AA(360),ION(20)
      DIMENSION LAMBDA(20),NOLAM(10,100),JLOW(20),JHIGH(20)
      DIMENSION OUT(9),TRUE(2),DAMP(9)
C     ----------------------------------------------------------------
C          DIMENSION OF AM(  ) = (NV*(NV+3))/2
C          CURRENTLY SET TO ALLOW 185 CONSTANTS TO VARY.
C     ----------------------------------------------------------------
      DIMENSION SQSIG(2),AM(17390),V(360),DV(360),DIAG(360),
     1          ROW(360)
      DIMENSION COEF(10,100),PNAME(20),TINV(10,100,4),IPHASE(10,100),
     1  NPHASE(100),IKOUNT(100),IGO(100),ISTATE(10,100),NINVER(10,100),
     2  INSTAT(10,100),INVPH(10,100,4),INVSC(10,100),ISPECL(10,100)
      DIMENSION X(4,1500),YO(1500),SIGYO(1500),P(360),KI(360),
     1  DC(360),PD(360),TITLE(20),SIGYON(1500)
      DIMENSION REF(20,100),ERRP(360),IHOLD(20)
C     ----------------------------------------------------------------
C         COMMON BLOCKS
C     ----------------------------------------------------------------
      COMMON /EARTH/ COEF,PNAME,TINV,IPHASE,NPHASE,IKOUNT,IGO,NSETS,
     1  ISTATE,NINVER,INSTAT,INVPH,INVSC,LISTP,ISPECL
      COMMON /AIR/X,P,DC,TITLE,YO,SIGYO,SIGYON,PD,KI,NC,IC,NV,
     1  NX,IW,NP,NO,ISING,ISTOP,IL,JDFLAG,NPARM,NVPARM,NCPARM,NMPARM
      COMMON /FIRE/ REF, ERRP, IWRITE, ICY, IICY, IREDO, IREG, NHOLD,
     1 IHOLD,UNDAMP
      COMMON /MAN/ AA, ION
      COMMON /AETHER/ LAMBDA,NOLAM,JLOW,JHIGH
      COMMON /HOTAIR/ RECKON,ITEK
      COMMON /ROCK/ AM
      COMMON /TIME/ DATE
      COMMON /DUMP/ IDUMP
c     ..... next line was added by MKo, 10/10/85 for Prime computer.
      common/ming02/diag,dv,row,v
      DATA TRUE/8Hreckon  ,8HRECKON  /
      DATA UNDER/30.0D0/
      DATA ZERO/0.0D0/
      DATA DAMP/1.D0,0.63D0,0.40D0,0.25D0,0.16D0,0.10D0,
     1          0.063D0,0.040D0,0.025D0/
C     FORMAT STATEMENTS
  100 FORMAT (1H1,20A4)
  150 FORMAT(42H0 THESE RESULTS WERE OBTAINED IN A RUN AT ,A23)
  200 FORMAT (32H0NUMBER OF CYCLES IN THIS JOB IS,I2/37H0NUMBER OF PARAM
     1ETERS TO BE VARIED IS,I3/51H0NUMBER OF INDEPENDENT VARIABLES PER O
     2BSERVATION IS,I2)
  300 FORMAT (46H0DERIVATIVES PROGRAMMED IN SUBROUTINE EAFWC . )
  400 FORMAT (31H0WEIGHTS TO BE SUPPLIED BY USER)
  500 FORMAT (34H0UNIT WEIGHTS TO BE SET BY PROGRAM)
  600 FORMAT (29H0NUMBER OF PARAMETERS READ IS,I4)
  700 FORMAT (31H0NUMBER OF OBSERVATIONS READ IS,I5)
  800 FORMAT (46H0CALCULATED Y BASED ON PARAMETERS BEFORE CYCLE,I2)
```

```
 900 FORMAT (1H0,5HINDEX,4X,6HX(1,I),4X,6HX(2,I),4X,6HX(3,I),4X,6HX(4,I
    1),9X,6HY(OBS),9X,7HY(CALC),8X,8HOBS-CALC,9X,6HSIG(O),6X,12H(O-C)/S
    1IG(O))
1000 FORMAT (1H ,I5,F10.3,10X,2F10.2,1P4D16.3,0PF16.4)
1100 FORMAT (1H ,I5,2F10.3,2F10.2,1P4D16.3,0PF16.4)
1200 FORMAT (1H ,I5,F10.3,10X,2F10.2,5F16.4)
1300 FORMAT (1H ,I5,2F10.3,2F10.2,5F16.4)
1350 FORMAT(1H ,I5,2F10.3,2F10.5,5F16.4)
1010 FORMAT (1H ,I5,F10.3,10X,2F10.2,1P3D16.3,2(11X,5H---- ))
1110 FORMAT (1H ,I5,2F10.3,2F10.2,1P3D16.3,2(11X,5H---- ))
1210 FORMAT (1H ,I5,F10.3,10X,2F10.2,3F16.4,2(11X,5H---- ))
1310 FORMAT (1H ,I5,2F10.3,2F10.2,3F16.4,2(11X,5H---- ))
1360 FORMAT(1H ,I5,2F10.3,2F10.5,3F16.4,2(11X,5H---- ))
1400 FORMAT (51H0AGREEMENT FACTORS BASED ON PARAMETERS BEFORE CYCLE,I2/
    120H0SUM(W*(O-C)**2) IS ,D11.3/35H0SQRTF(SUM(W*(O-C)**2)/(NO-NV)) I
    2S ,D10.4)
1500 FORMAT (60H0ESTIMATED AGREEMENT FACTORS BASED ON PARAMETERS AFTER
    1CYCLE,I2/20H0SUM(W*(O-C)**2) IS ,D11.3/35H0SQRTF(SUM(W*(O-C)**2)/(
    2NO-NV)) IS ,D10.4)
1600 FORMAT (62H MATRIX HAS A ZERO DIAGONAL ELEMENT CORRESPONDING TO PA
    1RAMETER,I3,16H OF THOSE VARIED)
1700 FORMAT (40H SINGULARITY RETURN FROM MATRIX INVERTER)
1800 FORMAT (37H0PARAMETERS AFTER LEAST SQUARES CYCLE,I2/1H0,T16,'OLD',
    1T35,'CHANGE',T57,'NEW',T77,'ERROR',T95,'PCT. CHANGE',T118,'PCT. ER
    2ROR'/1H )
1900 FORMAT (1H0,A8,':  CONSTRAINTS (ELEM = 1, OTHER = 0) = ',I2,
    1     ' AND LAMBDA ANOMALY = ',L1,', Ja = ',I2,' and Jb = ',I2,'.')
2000 FORMAT (1H ,I3,4X,1PD17.10)
2100 FORMAT (1H ,I3,1P5D21.10,0PF17.4)
2200 FORMAT (66H0SUBROUTINE TEST INDICATES THAT JOB IS TO BE TERMINATED
    1 FOR REASON,I2)
2300 FORMAT (16H0TRIAL CONSTANTS/1H0,2X,1HI,9X,4HP(I),6X,5HKI(I)/1H )
2400 FORMAT (1H0,9X,A8)
2500 FORMAT (1H ,I3,5X,1PD11.4,6X,I1)
2600 FORMAT (19H0CORRELATION MATRIX)
2700 FORMAT (1H0,I3,10D12.4/(1H ,3X,10D12.4))
2800 FORMAT (14H1***WARNING***/93H0THE TERM SIG/(NO-NV) IS NEGATIVE.  T
    1HE ABSOLUTE VALUE IS TAKEN AND THE REGRESSION CONTINUES.)
     id=1
     DO 2900 I=1,NP
     ERRP(I)=0.0D0
2900 CONTINUE
     WRITE (6,100) (TITLE(I),I=1,20)
     WRITE (6,150)DATE
     WRITE (6,200) NC,NV,NX
     WRITE (6,300)
     IF (IW) 3100,3000,3100
3000 WRITE (6,400)
     GO TO 3200
3100 WRITE (6,500)
3200 CONTINUE
     WRITE (6,600) NP
     WRITE (6,700) NO
     IF (NC) 3300,3300,3500
```

```
 3300 DO 3400 I=1,NP
 3400 KI(I)=0
C     INITIALIZE PROBLEM AND ENTER SUBROUTINE PRELIM IF PROVIDED
 3500 NM=(NV*(NV+1))/2
      SQSIG(1)=0.0D0
      CALL PRELIM
C     PUT OUT TRIAL PARAMETERS, AND KEY-INTEGERS.
      WRITE (6,2300)
      JO=1
      DO 3600 I=1,LISTP
      WRITE (6,2400) PNAME(I)
      JN=JO+NPARM-1
      WRITE (6,2500) (J,P(J),KI(J),J=JO,JN)
      JO=JO+NPARM
 3600 CONTINUE
C     START LOOP TO PERFORM NC CYCLES AND ONE FINAL CALCULATION OF Y
      NCY=NC+1
      DO 7900 IC=1,NCY
      IF(IC.LE.9)  CALL ADJUST
C        CLEAR ARRAYS AM AND V EXCEPT ON LAST CYCLE
      IF (IC-NCY) 3700,4000,4000
 3700 DO 3800 I=1,NM
 3800 AM(I)=0.0D0
      DO 3900 I=1,NV
 3900 V(I)=0.0D0
C        INITIALIZE FOR CYCLE IC AND PUT OUT CAPTION FOR LIST OF Y(CALC)
 4000 SQSIG(2)=SQSIG(1)
      SIG=0.0D0
      WRITE (6,100) (TITLE(I),I=1,20)
      IF (IL.NE.0) GO TO 4100
      IF (IC.NE.NCY) GO TO 4200
 4100 CONTINUE
      WRITE (6,800) IC
      WRITE (6,900)
 4200 CONTINUE
C        START LOOP THROUGH NO OBSERVATIONS
      IF (IW.EQ.0) SQRTNO=DSQRT(DBLE(NO))
      IDUMP=6
      DO 5900 I=1,NO
C           ENTER USERS SUBROUTINE TO COMPUTE Y(CALC) AND DERIVATIVES
      CALL EAFWC(YC,I,JSET)
C           OBTAIN WEIGHT AND CALCULATE QUANTITIES FROM Y(OBS)-Y(CALC)
      IF (IW) 4400,4300,4400
 4300 SQRTW=1.0D0/SIGYON(I)
      GO TO 4500
 4400 SIGYO(I)=1.0D0
      SIGYON(I)=SQRTNO
      SQRTW=1.0D0/SQRTNO
 4500 DY=YO(I)-YC
      WDY=SQRTW*DY
      SIG=SIG+WDY*WDY
C           PUT OUT Y(CALC) AND OTHER INFORMATION FOR ONE OBSERVATION
      IF (IL.NE.0) GO TO 4600
      IF (IC.NE.NCY) GO TO 5100
```

```
 4600 CONTINUE
      OUT(1)=X(1,I)
      OUT(2)=X(2,I)
      OUT(3)=X(3,I)
      OUT(4)=X(4,I)
      OUT(5)=YO(I)
      OUT(6)=YC
      OUT(7)=DY
      OUT(8)=SIGYO(I)
      OUT(9)=DY/SIGYO(I)
      TEMPO=YC
      IF(ITEK.GE.1)  WRITE (11,9100)JSET,(X(K,I),K=1,4),YO(I),YC
      IF(SIGYO(I).GE.1.0D10)  GO TO 4910
      IF((IGO(JSET).GT.7).AND.(IGO(JSET).LT.21))  GO TO 4650
      IF (X(2,I).EQ.0.0D0) GO TO 4800
      IF ((DABS(YO(I)).LT.1.0D-2).AND.(YO(I).NE.0.0D0)) GO TO 4700
      IF ((DABS(YC).LT.1.0D-2).AND.(YC.NE.0.0D0))  GO TO 4700
      WRITE (6,1300) I,(OUT(K),K=1,9)
      GO TO 5000
 4650 CONTINUE
      WRITE (6,1350)I,(OUT(K),K=1,9)
      GO TO 5000
 4700 CONTINUE
      WRITE (6,1100) I,(OUT(K),K=1,4),YO(I),YC,DY,SIGYO(I),OUT(9)
      GO TO 5000
 4800 CONTINUE
      IF ((DABS(YO(I)).LT.1.0D-2).AND.(YO(I).NE.0.0D0)) GO TO 4900
      WRITE (6,1200) I,OUT(1),(OUT(K),K=3,9)
      GO TO 5000
 4900 CONTINUE
      WRITE (6,1000) I,OUT(1),OUT(3),OUT(4),YO(I),YC,DY,SIGYO(I),OUT(9)
      GO TO 5000
 4910 CONTINUE
      IF((IGO(JSET).GT.7).AND.(IGO(JSET).LT.21))  GO TO 4920
      IF (X(2,I).EQ.0.0D0) GO TO 4940
      IF ((DABS(YO(I)).LT.1.0D-2).AND.(YO(I).NE.0.0D0)) GO TO 4930
      IF ((DABS(YC).LT.1.0D-2).AND.(YC.NE.0.0D0))  GO TO 4930
      WRITE (6,1310) I,(OUT(K),K=1,7)
      GO TO 5000
 4920 CONTINUE
      WRITE (6,1360)I,(OUT(K),K=1,7)
      GO TO 5000
 4930 CONTINUE
      WRITE (6,1110) I,(OUT(K),K=1,4),YO(I),YC,DY
      GO TO 5000
 4940 CONTINUE
      IF ((DABS(YO(I)).LT.1.0D-2).AND.(YO(I).NE.0.0D0)) GO TO 4950
      WRITE (6,1210) I,OUT(1),(OUT(K),K=3,7)
      GO TO 5000
 4950 CONTINUE
      WRITE (6,1010) I,OUT(1),OUT(3),OUT(4),YO(I),YC,DY
 5000 CONTINUE
 5100 CONTINUE
C           BY-PASS DERIVATIVE AND MATRIX SET-UP ON FINAL CALC OF Y
```

```
      IF (IC-NCY) 5200,5900,5900
C               START LOOP TO STORE AN ARRAY OF NV DERIVATIVES
 5200 J=1
      DO 5400 K=1,NP
      IF (KI(K)) 5400,5400,5300
 5300 CONTINUE
C                        OBTAIN DERIVATIVE FROM THOSE PROGRAMMED BY USER
      DV(J)=SQRTW*DC(K)
      J=J+1
 5400 CONTINUE
C               END OF LOOP TO OBTAIN DERIVATIVES
C               START LOOP TO STORE MATRIX AND VECTOR.
C               1604 OR GLS STORAGE SCHEME IS REVERSE OF 7090 OR GLS
      JK=1
      DO 5800 J=1,NV
      TEMP=DV(J)
      IF (TEMP) 5600,5500,5600
C                    BY-PASS IF DERIVATIVE IS ZERO
 5500 JK=JK+NV+1-J
      GO TO 5800
 5600 DO 5700 K=J,NV
      IF((TEMP.EQ.ZERO).OR.(DV(K).EQ.ZERO))  GO TO 5650
      IF(DABS(DLOG10(DABS(TEMP))+DLOG10(DABS(DV(K)))).GT.UNDER)
     1      GO TO 5650
      AM(JK)=AM(JK)+TEMP*DV(K)
 5650 CONTINUE
      JK=JK+1
 5700 CONTINUE
      V(J)=V(J)+TEMP*WDY
 5800 CONTINUE
C                    END LOOP TO STORE MATRIX AND VECTOR
 5900 CONTINUE
C               END LOOP THROUGH NO OBSERVATIONS
C        COMPUTE AND PUT OUT AGREEMENT FACTORS
      ONVN=NO-NV
      SQSIG(1)=DSQRT(SIG/(ONVN))
      WRITE (6,1400) IC,SIG,SQSIG(1)
C        BY-PASS MATRIX INVERSION AND PARAMETER OUTPUT ON FINAL CYCLE
      IF (IC-NCY) 6000,8000,8000
C           START LOOP TO TEST FOR ZERO DIAGONAL ELEMENT
 6000 ISING=0
      II=1
      IID=NV
      DO 6300 I=1,NV
      IF (AM(II)) 6200,6100,6200
 6100 ISING=1
      WRITE (6,1600) I
 6200 II=II+IID
      IID=IID-1
 6300 CONTINUE
C               END LOOP TO TEST FOR ZERO DIAGONAL ELEMENT
C           TERMINATE JOB IF ZERO DIAGONAL ELEMENT WAS FOUND
      IF (ISING) 8600,6400,8600
C               ENTER SUBROUTINE TO REPLACE MATRIX WITH INVERSE
```

```
 6400 CALL MINVC(NV,ISING)
      IF (ISING) 6500,6600,6500
C                 TERMINATE JOB IF SINGULAR MATRIX WAS FOUND
 6500 WRITE (6,1700)
      GO TO 8600
C                 START LOOP FOR MATRIX VECTOR MULTIPLICATION FOR
C                                                   PARAMETER CHANGES
 6600 DO 7100 I=1,NV
      PDI=0.0D0
      IJ=I
      IJD=NV-1
      DO 7000 J=1,NV
      PDI=PDI+AM(IJ)*V(J)
      IF (J-I) 6700,6800,6900
 6700 IJ=IJ+IJD
      IJD=IJD-1
      GO TO 7000
C                          SAVE DIAGONAL ELEMENTS OF INVERSE MATRIX
 6800 DIAG(I)=AM(IJ)
 6900 IJ=IJ+1
 7000 CONTINUE
      PD(I)=PDI
      SIG=SIG-PDI*V(I)
 7100 CONTINUE
C                    END LOOP FOR MATRIX VECTOR MULTIPLICATION
C                 RECOMPUTE AGREEMENT FACTOR USING MODIFIED SIG
      IF (SIG) 7200,7300,7300
 7200 CONTINUE
      WRITE (6,2800)
      SIG=DABS(SIG)
 7300 CONTINUE
      SQSIG(1)=DSQRT(SIG/(NO-NV))
C                 PUT OUT CAPTION FOR LIST OF CORRECTED PARAMETERS
      WRITE (6,100) (TITLE(I),I=1,20)
      WRITE (6,1800) IC
C                 START LOOP TO CORRECT AND PUT OUT PARAMETERS
      J=1
      JO=1
      DO 7700 IO=1,LISTP
      WRITE (6,1900) PNAME(IO),ION(IO),LAMBDA(IO),JLOW(IO),JHIGH(IO)
      JN=JO+NPARM-1
      DO 7600 I=JO,JN
      IF (KI(I)) 7400,7400,7500
 7400 WRITE (6,2000) I,P(I)
C     IF((RECKON.EQ.TRUE(1)).OR.(RECKON.EQ.TRUE(2)))
C    1       WRITE (7,8700) I,P(I),ZERO
      GO TO 7600
 7500 POLD=P(I)
      KII=KI(I)
      P(I)=POLD+PD(J)*DAMP(KII)
      SIGP=DSQRT(DIAG(J))*SQSIG(1)
      PCTCHG=1.D2*PD(J)*DAMP(KII)/P(I)
      PCTERR=DABS(1.0D2*SIGP/P(I))
      ERRP(I)=PCTERR
```

```
C     IF((RECKON.EQ.TRUE(1)).OR.(RECKON.EQ.TRUE(2)))
C    1       WRITE (7,8700) I,P(I),SIGP
      WRITE (6,2100) I,POLD,PD(J),P(I),SIGP,PCTCHG,PCTERR
      J=J+1
 7600 CONTINUE
      JO=JO+NPARM
 7700 CONTINUE
C                  THE END OF LOOP TO CORRECT AND PUT OUT PARAMETERS
C                  PUT OUT ESTIMATED AGREEMENT FACTORS
      WRITE (6,1500) IC,SIG,SQSIG(1)
C                  ENTER USERS SUBROUTINE TO TEST AND MODIFY PARAMETERS
C                                                        OR END JOB
      ISTOP=0
      CALL TESTC
C                  TERMINATE JOB IF INDICATED BY USERS SUBROUTINE TEST
      IF (ISTOP) 7800,7900,7800
 7800 WRITE (6,2200) ISTOP
      GO TO 8000
 7900 CONTINUE
C                  THE END OF LOOP THROUGH NC CYCLES AND FINAL CALC OF Y
C     TERMINATE JOB
 8000 IF (NC) 8600,8600,8100
 8100 CONTINUE
      IF((IL.EQ.2).AND.(.NOT.((RECKON.EQ.TRUE(1))
     1      .OR.(RECKON.EQ.TRUE(2)))))
     1      GO TO 8600
C        CALCULATE AND PUT OUT CORRELATION MATRIX
      IF(IL.EQ.2)  WRITE (6,100) (TITLE(I),I=1,20)
      IF(IL.EQ.2)  WRITE (6,2600)
      DO 8200 I=1,NV
      DIAG(I)=1.0D0/DSQRT(DIAG(I))
 8200 CONTINUE
      IJ=1
      DO 8500 I=1,NV
      DO 8300 J=1,NV
      ROW(J)=0.0D0
 8300 CONTINUE
      DO 8400 J=I,NV
      ROW(J)=AM(IJ)*DIAG(I)*DIAG(J)
      IJ=IJ+1
 8400 CONTINUE
      IF(IL.EQ.2)  WRITE (6,2700) I,(ROW(J),J=1,NV)
C     IF((RECKON.EQ.TRUE(1)).OR.(RECKON.EQ.TRUE(2)))
C    1       WRITE (7,8700) I,(ROW(J),J=1,NV)
 8500 CONTINUE
 8600 CONTINUE
      RETURN
 8700 FORMAT(I3,2X,1p5d15.8/(5X,1p5d15.8))
 8800 FORMAT(20A4)
 8900 FORMAT(16I5)
 9000 FORMAT(10A8)
 9100 FORMAT(I5,1H;,1PD12.5,5(1H,,1PD12.5))
      END
      SUBROUTINE PRELIM
```

```
C     ------------------------------------------------------------------
C     DUMMY SUBROUTINE PRELIM
C     ------------------------------------------------------------------
      RETURN
      END
      SUBROUTINE ADJUST
      IMPLICIT REAL*8(A-H,O-Z)
      LOGICAL*1 UNDAMP
      REAL*4 TITLE
      DIMENSION REF(20,100),ERRP(360),IHOLD(20)
      DIMENSION X(4,1500),YO(1500),SIGYO(1500),P(360),KI(360),
     1  DC(360),PD(360),TITLE(20),SIGYON(1500)
C     ------------------------------------------------------------------
C         COMMON BLOCKS
C     ------------------------------------------------------------------
      COMMON /FIRE/ REF, ERRP, IWRITE, ICY, IICY, IREDO, IREG, NHOLD,
     1 IHOLD,UNDAMP
      COMMON /AIR/X,P,DC,TITLE,YO,SIGYO,SIGYON,PD,KI,NC,IC,NV,
     1  NX,IW,NP,NO,ISING,ISTOP,IL,JN,NPARM,NVPARM,NCPARM,NMPARM
      IF(IC.EQ.1) GO TO 9999
      IF(.NOT.UNDAMP)  GO TO 9999
      DO 1000 I=1,NP
      IF(KI(I).LE.1)  GO TO 1000
      KI(I)=KI(I)-1
 1000 CONTINUE
 9999 CONTINUE
      RETURN
      END
      SUBROUTINE TESTC
C     ------------------------------------------------------------------
C
C                                   TESTC
C
C     TESTC  TERMINATES THE REGRESSION IF THE CHANGE IN ALL
C          PARAMETERS IS LESS THAN 1.0D-8.
C
C     TESTC   WILL ALSO CAUSE IREDO TO TAKE THE INDEX OF THE
C          PARAMETER WHICH HAS THE GREATEST ERROR, PROVIDED ICY HAS
C          AN NON-ZERO VALUE AND PROVIDED THE ERROR ON ONE OR MORE
C          PARAMETERS IS GREATER THAN 10 PERCENT.
C
C     ------------------------------------------------------------------
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*4 TITLE
      LOGICAL*1 UNDAMP
      DIMENSION X(4,1500),YO(1500),SIGYO(1500),P(360),KI(360),
     1  DC(360),PD(360),TITLE(20),SIGYON(1500)
      DIMENSION REF(20,100),ERRP(360),IHOLD(20)
C     ------------------------------------------------------------------
C         COMMON BLOCKS
C     ------------------------------------------------------------------
      COMMON /AIR/X,P,DC,TITLE,YO,SIGYO,SIGYON,PD,KI,NC,IC,NV,
     1  NX,IW,NP,NO,ISING,ISTOP,IL,JDFLAG,NPARM,NVPARM,NCPARM,NMPARM
      COMMON /FIRE/ REF, ERRP, IWRITE, ICY, IICY, IREDO, IREG, NHOLD,
```

```
     1 IHOLD,UNDAMP
       IREDO=0
       ISTAY=0
       J=1
       DO 400 I=1,NP
       IF (KI(I).EQ.0) GO TO 400
       IF (DABS(PD(J)/P(I)).GT.1.0D-8) ISTAY=1
       J=J+1
       IF (ICY.EQ.0) GO TO 400
       DERRP=ERRP(I)
       IF (DERRP.LE.1.0D1) GO TO 400
       IF (IREDO.EQ.0) GO TO 100
       IF (DERRP.LE.ERRP(IREDO)) GO TO 400
   100 CONTINUE
       IF (NHOLD.EQ.0) GO TO 300
       DO 200 K=1,NHOLD
       IF (I.EQ.IHOLD(K)) GO TO 400
   200 CONTINUE
   300 CONTINUE
       IREDO=I
   400 CONTINUE
       IF (ISTAY.EQ.1) GO TO 500
       ISTOP=1
   500 RETURN
       END
       SUBROUTINE MINVC(N,NFAIL)
       IMPLICIT REAL*8(A-H,O-Z)
       DIMENSION AM(17390)
C      ********* DIMENSION OF AM(    ) = (NV*(NV+3))/2 *********
C      ***** CURRENTLY SET TO ALLOW 185 CONSTANTS TO VARY *****
       COMMON /ROCK/ AM
        DATA UNDER/-30.0D0/
C      ************ SEGMENT 1 OF CHOLESKI INVERSION ************
C      ***** FACTOR MATRIX INTO LOWER TRIANGLE X TRANSPOSE *****
       K=1
       IF (N-1) 100,200,300
   100 NFAIL=K
       GO TO 2000
   200 AM(1)=1.0/AM(1)
       GO TO 1900
C      ***** LOOP M OF A(L,M) *****
   300 DO 1200 M=1,N
       IMAX=M-1
C      ***** LOOP L OF A(L,M) *****
       DO 1100 L=M,N
       SUMA=0.0
       KLI=L
       KMI=M
       IF (IMAX) 600,600,400
C      ***** SUM OVER I=1,M-1 A(L,I)*A(M,I) *****
   400 DO 500 I=1,IMAX
       IF((AM(KLI).EQ.0.0D0).OR.(AM(KMI).EQ.0.0D0))  GO TO 450
       IF(DLOG10(DABS(AM(KLI)))+DLOG10(DABS(AM(KMI))).LT.UNDER)
      1      GO TO 450
```

```
      SUMA=SUMA+AM(KLI)*AM(KMI)
  450 CONTINUE
      J=N-I
      KLI=KLI+J
  500 KMI=KMI+J
C     ***** TERM=C(L,M)-SUM *****
  600 TERM=AM(K)-SUMA
      IF (L-M) 700,700,1000
  700 IF (TERM) 900,900,800
C     ***** A(M,M)=SQRT(TERM) *****
  800 DENOM=DSQRT(TERM)
      AM(K)=DENOM
      GO TO 1100
  900 NFAIL=K
      GO TO 2000
C     ***** A(L,M)=TERM/A(M,M) *****
 1000 AM(K)=TERM/DENOM
 1100 K=K+1
 1200 CONTINUE
C     ********** SEGMENT 2 OF CHOLESKI INVERSION **********
C     ***** INVERSION OF TRIANGULAR MATRIX *****
      AM(1)=1.0/AM(1)
      KDM=1
C     ***** STEP L OF B(L,M) *****
      DO 1500 L=2,N
      KDM=KDM+N-L+2
C     ***** RECIPROCAL OF DIAGONAL TERM *****
      TERM=1.0/AM(KDM)
      AM(KDM)=TERM
      KMI=0
      KLI=L
      IMAX=L-1
C     ***** STEP M OF B(L,M) *****
      DO 1400 M=1,IMAX
      K=KLI
C     ***** SUM TERMS *****
      SUMA=0.0
      DO 1300 I=M,IMAX
      II=KMI+I
      IF((AM(KLI).EQ.0.0D0).OR.(AM(II).EQ.0.0D0))  GO TO 1250
      IF(DLOG10(DABS(AM(KLI)))+DLOG10(DABS(AM(II))).LT.UNDER)
     1     GO TO 1250
      SUMA=SUMA-AM(KLI)*AM(II)
 1250 CONTINUE
 1300 KLI=KLI+N-I
C     ***** MULT SUM * RECIP OF DIAGONAL *****
      AM(K)=SUMA*TERM
      J=N-M
      KLI=K+J
 1400 KMI=KMI+J
 1500 CONTINUE
C     ********** SEGMENT 3 OF CHOLESKI INVERSION **********
C     ***** PREMULTIPLY LOWER TRIANGLE BY TRANSPOSE *****
      K=1
```

```
      DO 1800 M=1,N
      KLI=K
      DO 1700 L=M,N
      KMI=K
      IMAX=N-L+1
      SUMA=0.0
      DO 1600 I=1,IMAX
      IF((AM(KLI).EQ.0.0D0).OR.(AM(KMI).EQ.0.0D0))  GO TO 1550
      IF(DLOG10(DABS(AM(KLI)))+DLOG10(DABS(AM(KMI))).LT.UNDER)
     1     GO TO 1550
      SUMA=SUMA+AM(KLI)*AM(KMI)
 1550 CONTINUE
      KLI=KLI+1
 1600 KMI=KMI+1
      AM(K)=SUMA
 1700 K=K+1
 1800 CONTINUE
 1900 NFAIL=0
 2000 RETURN
      END
      SUBROUTINE EAFWC (YC,I,J)
C     ------------------------------------------------------------------
C
C                                   EAFWC
C
C      THIS VERSION WRITTEN 3/09/72 BY HAAS.
C               MODIFIED 8/05/72 BY HAAS.
C                        9/11/74 BY HAAS.
C
C     ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      LOGICAL*1 UNDAMP
      CHARACTER*1 LABEL,BCD
      REAL*4 TITLE
      DIMENSION SGN(2)
      DIMENSION COEF(10,100),PNAME(20),TINV(10,100,4),IPHASE(10,100),
     1  NPHASE(100),IKOUNT(100),IGO(100),ISTATE(10,100),NINVER(10,100),
     2  INSTAT(10,100),INVPH(10,100,4),INVSC(10,100),ISPECL(10,100)
      DIMENSION X(4,1500),YO(1500),SIGYO(1500),P(360),KI(360),
     1  DC(360),PD(360),TITLE(20),SIGYON(1500)
      DIMENSION REF(20,100),ERRP(360),IHOLD(20)
      DIMENSION SCINV(2),STCOEF(10),YESNO(2),TK(19,2),TYPE(28),NSCALE(5)
     1  ,LABEL(50)
      DIMENSION DYDC(10),DDDC(10),DLDC(2),DLDDC(2)
C     ------------------------------------------------------------------
C         COMMON BLOCKS
C     ------------------------------------------------------------------
      COMMON /EARTH/ COEF,PNAME,TINV,IPHASE,NPHASE,IKOUNT,IGO,NSETS,
     1  ISTATE,NINVER,INSTAT,INVPH,INVSC,LISTP,ISPECL
      COMMON /AIR/X,P,DC,TITLE,YO,SIGYO,SIGYON,PD,KI,NC,IC,NV,
     1  NX,IW,NP,NO,ISING,ISTOP,IL,JDFLAG,NPARM,NVPARM,NCPARM,NMPARM
      COMMON /FIRE/ REF, ERRP, IWRITE, ICY, IICY, IREDO, IREG, NHOLD,
     1 IHOLD,UNDAMP
      COMMON /WATER/ZERO, ONE, TWO, THREE, FOUR, SIX, R, F,
```

```
     1  SCINV, TREF, PREF, STCOEF, DIEO, ADIE, BDIE, THETA, YESNO, TK,
     2  ASTAR, TYPE, NL, NSCALE
      COMMON /WATCHR/LABEL, BCD
      COMMON /SPACE/ DYDC, DDDC, DLDC, DLDDC, SC, TLOW
      COMMON /DUMP/ IDUMP
C     ------------------------------------------------------------------
C     DETERMINE J BY LOCATING I IN THE VECTOR IKOUNT.  IF NOT
C          LOCATED, PRINT ERROR MESSAGE AND STOP
C     ------------------------------------------------------------------
      DO 100 J=1,NSETS
      IF (I.LE.IKOUNT(J)) GO TO 200
  100 CONTINUE
      WRITE (6,600)
      STOP
  200 CONTINUE
C     ------------------------------------------------------------------
C     WRITE REFERENCE IF THE I-TH OBSERVATION IS THE FIRST IN THE
C          J-TH DATA SET.
C     ------------------------------------------------------------------
      IF (IWRITE.EQ.1.OR.I.EQ.1) GO TO 300
      GO TO 400
  300 CONTINUE
      WRITE (IDUMP,700) J,(REF(II,J),II=1,15)
      IWRITE=0
  400 CONTINUE
      IF (I.EQ.IKOUNT(J)) IWRITE=1
      IF (IGO(J).LE.7) GO TO 500
C     ------------------------------------------------------------------
C     IF IGO(J) IS GREATER THAN 7, CALL UNIQUE TO CALCULATE
C        YC (AND THE DERIVATIVES DC(I)) IF JDFLAG (JN IN MAIN) IS
C        ZERO.
C     ------------------------------------------------------------------
      IF(IGO(J).LE.20)  GO TO 450
      CALL VOLUME(YC,I,J)
      RETURN
  450 CONTINUE
      CALL UNIQUE(YC,I,J,JDFLAG)
      RETURN
  500 CONTINUE
C     ------------------------------------------------------------------
C     ENTER FNCALC AND CALCULATE Y(CALCULATED) AND THE DERIVATIVES
C     DC.
C     ------------------------------------------------------------------
      TLOW=X(2,I)
      YC=FNCALC(X(1,I),J,IGO(J))
C     ------------------------------------------------------------------
C     RETURN
C     ------------------------------------------------------------------
      RETURN
C     ------------------------------------------------------------------
C     FORMAT STATEMENTS
C     ------------------------------------------------------------------
  600 FORMAT (60H0IN EAFWC  I IS GREATER THAN NO, THE NUMBER OF OBSERVAT
     1IONS./64H THEREFORE, I (THE COMPUTER) AM QUITTING. THE PROBLEM IS
```

```
     2YOURS. )
  700 FORMAT (1H0,I5,3X,15A8/1H )
      END
      FUNCTION FNCALC(T,J,ICODE)
C     ------------------------------------------------------------------
C
C     THE FUNCTION FNCALC IS ADAPTED FROM THE ORIGINAL EAFW20
C        BY HAAS, 9/11/74.
C
C
C     MODIFIED TO ALLOW FOR LAMBDA ANOMALIES BY HAAS, FALL, 1976.
C     ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*1 LABEL,BCD
      LOGICAL*1 UNDAMP
      LOGICAL*1 LAMBDA,NOLAM
      REAL*4 TITLE
      DIMENSION LAMBDA(20),NOLAM(10,100),JLOW(20),JHIGH(20)
      DIMENSION SGN(2)
      DIMENSION COEF(10,100),PNAME(20),TINV(10,100,4),IPHASE(10,100),
     1  NPHASE(100),IKOUNT(100),IGO(100),ISTATE(10,100),NINVER(10,100),
     2  INSTAT(10,100),INVPH(10,100,4),INVSC(10,100),ISPECL(10,100)
      DIMENSION X(4,1500),YO(1500),SIGYO(1500),P(360),KI(360),
     1  DC(360),PD(360), TITLE(20),SIGYON(1500)
      DIMENSION REF(20,100),ERRP(360),IHOLD(20)
      DIMENSION DYDC(10),DDDC(10),DLDC(2),DLDDC(2)
      DIMENSION SCINV(2),STCOEF(10),YESNO(2),TK(19,2),TYPE(28),NSCALE(5)
     1  ,LABEL(50)
C     ------------------------------------------------------------------
C         COMMON BLOCKS
C     ------------------------------------------------------------------
      COMMON /EARTH/ COEF,PNAME,TINV,IPHASE,NPHASE,IKOUNT,IGO,NSETS,
     1  ISTATE,NINVER,INSTAT,INVPH,INVSC,LISTP,ISPECL
      COMMON /AIR/X,P,DC,TITLE,YO,SIGYO,SIGYON,PD,KI,NC,IC,NV,
     1  NX,IW,NP,NO,ISING,ISTOP,IL,JDFLAG,NPARM,NVPARM,NCPARM,NMPARM
      COMMON /FIRE/ REF, ERRP, IWRITE, ICY, IICY, IREDO, IREG, NHOLD,
     1  IHOLD,UNDAMP
      COMMON /WATER/ZERO, ONE, TWO, THREE, FOUR, SIX, R, F,
     1  SCINV, TREF, PREF, STCOEF, DIEO, ADIE, BDIE, THETA, YESNO, TK,
     2  ASTAR, TYPE, NL, NSCALE
      COMMON /WATCHR/LABEL, BCD
      COMMON /SPACE/ DYDC, DDDC, DLDC, DLDDC, SC, TLOW
      COMMON /AETHER/ LAMBDA,NOLAM,JLOW,JHIGH
C     ------------------------------------------------------------------
C     INITIALIZE VARIABLES
C     ------------------------------------------------------------------
      FNCALC=ZERO
      DO 100 K=1,NP
      DC(K)=ZERO
  100 CONTINUE
      LAST=NPHASE(J)
C     ------------------------------------------------------------------
C     FOR EACH PHASE IN THE J-TH DATA SET, CALCULATE THE
C          DERIVATIVES DC AND THE FUNCTION DCUSP IF THERE
```

```
C          IS A LAMBDA ANOMALY FOR THE PHASE.
C     ------------------------------------------------------------------
      DO 1300 L=1,LAST
C     ------------------------------------------------------------------
C     LOCATE THE FIRST CONSTANT FOR THE L-TH PHASE IN THE VECTOR P.
C          INITIALIZE SC, THE STOICHIOMETRIC COEFFICIENT.
C     ------------------------------------------------------------------
      INDEX=1+NPARM*(IPHASE(L,J)-1)
      SC=COEF(L,J)
C     ------------------------------------------------------------------
C     IF THE PHASE HAS A LAMBDA TRANSITION, SET THE PARAMETER TC
C     ------------------------------------------------------------------
      TC=P(INDEX+NCPARM)
C     ------------------------------------------------------------------
C     CALL YDERIV TO CALCULATE THE TEMPERATURE TERM IN THE DERIVATIVE.
C          IF THE DATA SET IS FOR ENTROPY AND HEAT CAPACITY,  GO TO
C          STATEMENT 130.
C     ------------------------------------------------------------------
      IGOES=ICODE
      IF (ISTATE(L,J).EQ.-1) IGOES=IGOES+7
      CALL YDERIV(T,IGOES)
      IF(ISPECL(L,J).EQ.-1)  CALL YDERIV(TLOW,IGOES)
      IF (ICODE.LE.2) GO TO 1100
C     ------------------------------------------------------------------
C     CALL DDERIV TO CALCULATE THE DERIVATIVE WITH RESPECT TO -D- AT
C          298.15 KELVIN IF COMPONENT IS AN ELEMENT.  PROCEED TO ADJUST
C          DC ACCORDINGLY
C     ------------------------------------------------------------------
      IF (ISTATE(L,J).NE.1) GO TO 300
      CALL DDERIV(TREF)
      DO 200 K=1,NCPARM
      DC(K+INDEX-1)=DC(K+INDEX-1)+SC*DDDC(K)*DYDC(9)
  200 CONTINUE
      IF(NOLAM(L,J))  GO TO 220
      IF(.NOT.LAMBDA(IPHASE(L,J))) GO TO 220
      JL=JLOW(IPHASE(L,J))
      JH=JHIGH(IPHASE(L,J))
      CALL LDDRIV(TREF,INDEX,JL,JH)
      DC(INDEX+NCPARM+1)=DC(INDEX+NCPARM+1)+SC*DYDC(9)*DLDDC(1)
      IF(TREF.LE.P(INDEX+NCPARM))  GO TO 220
      DC(INDEX+NCPARM+2)=DC(INDEX+NCPARM+2)+SC*DYDC(9)*DLDDC(2)
  220 CONTINUE
      GO TO 1100
  300 CONTINUE
C     ------------------------------------------------------------------
C     CHECK FOR INVERSIONS.  IF LOW TEMPERATURE PHASES ARE TO BE
C          CONSIDERED, INITIALIZE SGN.  IF LOW TEMPERATURE PHASES
C          ARE NOT TO BE CONSIDERED,  GO TO STATEMENT 130.
C     ------------------------------------------------------------------
      IF (NINVER(L,J).LE.0) GO TO 1100
      DO 400 LLL=1,2
      SGN(LLL)=SCINV(LLL)
  400 CONTINUE
      IF (INVSC(L,J).EQ.0) GO TO 500
```

```
      SGN(1)=SGN(1)*STCOEF(INVSC(L,J))
      SGN(2)=SGN(2)*STCOEF(INVSC(L,J))
  500 CONTINUE
C     ------------------------------------------------------------------
C     IF THE LOWEST TEMPERATURE PHASE IS AN ELEMENT, CALL DDERIV
C          THE ARGUMENT IS 298.15, AND PROCEED TO ADJUST DC
C          ACCORDINGLY.
C     ------------------------------------------------------------------
      IF (INSTAT(L,J).EQ.0) GO TO 700
      IINVR=1+NPARM*(INVPH(L,J,1)-1)
      CALL DDERIV(TREF)
      DO 600 K=1,NCPARM
      DC(K+IINVR-1)=DC(K+IINVR-1)+DDDC(K)*SGN(2)*SC*DYDC(9)
  600 CONTINUE
      IF(.NOT.LAMBDA(INVPH(L,J,1)))  GO TO 620
      JL=JLOW(INVPH(L,J,1))
      JH=JHIGH(INVPH(L,J,1))
      CALL LDDRIV(TREF,IINVR,JL,JH)
      DC(IINVR+NCPARM+1)=DC(IINVR+NCPARM+1)+SC*SGN(2)*DYDC(9)*DLDDC(1)
      IF(TREF.LE.P(IINVR+NCPARM))  GO TO 620
      DC(IINVR+NCPARM+2)=DC(IINVR+NCPARM+2)+SC*SGN(2)*DYDC(9)*DLDDC(2)
  620 CONTINUE
  700 CONTINUE
C     ------------------------------------------------------------------
C     CALL DDERIV TO CALCULATE THE DERIVATIVE WITH RESPECT TO D FOR
C          HIGH AND LOW TEMPERATURE PHASE AT TEMPERATURE OF
C          INVERSION.  PROCEED TO ADJUST DC ACCORDINGLY
C     ------------------------------------------------------------------
      LSTINV=NINVER(L,J)
      DO 1000 LL=1,LSTINV
      CALL DDERIV(TINV(L,J,LL))
C     ------------------------------------------------------------------
C     IF THERE IS A STOICHIOMETRY CHANGE FOR THE HIGHEST TRANSITION
C          (EG: 2 FECL2(L) = FE2CL4(G)), ADJUST SGN(2) ACCORDINGLY
C     ------------------------------------------------------------------
      IF (INVSC(L,J).EQ.0) GO TO 800
      IF (LL.NE.LSTINV) GO TO 800
      SGN(2)=SGN(2)/STCOEF(INVSC(L,J))
  800 CONTINUE
      DO 900 LLL=1,2
      IINVR=1+NPARM*(INVPH(L,J,(LL+LLL-1))-1)
      DO 810 K=1,NCPARM
      DC(K+IINVR-1)=DC(K+IINVR-1)+DDDC(K)*SGN(LLL)*SC*DYDC(9)
  810 CONTINUE
      IF(.NOT.LAMBDA(INVPH(L,J,(LL+LLL-1)))) GO TO 820
      JL=JLOW(INVPH(L,J,(LL+LLL-1)))
      JH=JHIGH(INVPH(L,J,(LL+LLL-1)))
      TINVER=TINV(L,J,LL)
      CALL LDDRIV(TINVER,IINVR,JL,JH)
      DC(IINVR+NCPARM+1)=DC(IINVR+NCPARM+1)+SC*SGN(LLL)*DYDC(9)*DLDDC(1)
      IF(TINVER.LE.P(IINVR+NCPARM))  GO TO 820
      DC(IINVR+NCPARM+2)=DC(IINVR+NCPARM+2)+SC*SGN(LLL)*DYDC(9)*DLDDC(2)
  820 CONTINUE
  900 CONTINUE
```

```
 1000 CONTINUE
 1100 CONTINUE
C     ------------------------------------------------------------------
C     COMPLETE THE CALCULATION OF DC BY ADDING THE DERIVATIVE AT T(OBS).
C     ------------------------------------------------------------------
      DO 1200 K=1,NCPARM
      DC(K+INDEX-1)=DC(K+INDEX-1)+SC*DYDC(K)
 1200 CONTINUE
C     ------------------------------------------------------------------
C     COMPLETE WITH THE CALCULATION OF DPERNT IF APPLICABLE.
C     ------------------------------------------------------------------
      IF(NOLAM(L,J))  GO TO 1220
      IF(.NOT.LAMBDA(IPHASE(L,J))) GO TO 1220
      JL=JLOW(IPHASE(L,J))
      JH=JHIGH(IPHASE(L,J))
      CALL LDERIV(T,IGOES,INDEX,JL,JH)
      DC(INDEX+NCPARM+1)=DC(INDEX+NCPARM+1)+SC*DLDC(1)
      IF(T.LE.TC)  GO TO 1220
      DC(INDEX+NCPARM+2)=DC(INDEX+NCPARM+2)+SC*DLDC(2)
 1220 CONTINUE
 1300 CONTINUE
C     ------------------------------------------------------------------
C     ALL DERIVATIVES HAVE BEEN CALCULATED.  NOW CALCULATE FNCALC BY
C          SUMMING THE PRODUCT DC*P.  ADD IN THE COMPONENT DUE TO
C          THE LAMBDA ANOMALY IF APPROPRIATE.
C     ------------------------------------------------------------------
      DO 1400 K=1,NP
      IF(MOD(K,NPARM).EQ.11)  GO TO 1400
      FNCALC=FNCALC+P(K)*DC(K)
 1400 CONTINUE
C     ------------------------------------------------------------------
C     RETURN TO THE CALLING PROGRAM
C     ------------------------------------------------------------------
      RETURN
      END
      SUBROUTINE LDERIV(T,IGOES,INDEX,JL,JH)
C     ------------------------------------------------------------------
C
C                           LDERIV
C
C     LDERIV CALCULATES THE DERIVATIVES FOR THE LAMBDA ANOMALY
C        FUNCTIONS WITH RESPECT TO THE CONSTANTS FOR BOTH LIMBS.
C
C     REFERENCE:  GUILLERMET & GUSTAFSON (1984)
C                 AGREN (1979)
C                 HILLERT & JARL (1975).
C
C     WRITTEN 10/84 BY HAAS.
C     ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*4 TITLE
      CHARACTER*1 LABEL,BCD
      DOUBLE PRECISION KM
      DIMENSION X(4,1500),YO(1500),SIGYO(1500),P(360),KI(360),
```

```
     1  DC(360),PD(360),TITLE(20),SIGYON(1500)
      DIMENSION SCINV(2),STCOEF(10),YESNO(2),TK(19,2),TYPE(28),NSCALE(5)
     1  ,LABEL(50)
      DIMENSION DYDC(10),DDDC(10),DLDC(2),DLDDC(2)
      COMMON /AIR/X,P,DC,TITLE,YO,SIGYO,SIGYON,PD,KI,NC,IC,NV,
     1  NX,IW,NP,NO,ISING,ISTOP,IL,JN,NPARM,NVPARM,NCPARM,NMPARM
      COMMON /WATER/ZERO, ONE, TWO, THREE, FOUR, SIX, R, F,
     1  SCINV, TREF, PREF, STCOEF, DIEO, ADIE, BDIE, THETA, YESNO, TK,
     2  ASTAR, TYPE, NL, NSCALE
      COMMON /WATCHR/LABEL, BCD
      COMMON /SPACE/ DYDC, DDDC, DLDC, DLDDC, SC, TLOW
      NTERM=7
      TC=P(INDEX+NCPARM)
      TAU=T/TC
      AJL=DBLE(JL)
      AJH=DBLE(-JH)
      DLDC(1)=ZERO
      DLDC(2)=ZERO
      GO TO (1000,2000,3000,4000,5000,6000,7000), IGOES
C     ------------------------------------------------------------------
C        DERIVATIVE OF THE LAMBDA ANOMALY FUNCTION WITH RESPECT TO
C        THE CONSTANTS FOR THE TWO LIMBS OF THE HEAT CAPACITY EQUATION.
C     ------------------------------------------------------------------
 1000 CONTINUE
      DO 1200 I=1,NTERM
      ITERM=2*I-1
      AITERM=DBLE(ITERM)
      IF(T.GT.TC)  GO TO 1100
      DLDC(1)=DLDC(1)+TAU**(JL*ITERM)/AITERM
      GO TO 1200
 1100 CONTINUE
      DLDC(2)=DLDC(2)+TAU**(-JH*ITERM)/AITERM
 1200 CONTINUE
      RETURN
C     ------------------------------------------------------------------
C        DERIVATIVE OF THE LAMBDA ANOMALY FUNCTION WITH RESPECT TO
C        THE CONSTANTS FOR THE TWO LIMBS OF THE ENTROPY EQUATION.
C     ------------------------------------------------------------------
 2000 CONTINUE
      DO 2200 I=1,NTERM
      ITERM=2*I-1
      AITERM=DBLE(ITERM)
      IF(T.GT.TC)  GO TO 2100
      DLDC(1)=DLDC(1)+TAU**(JL*ITERM)/(AJL*AITERM*AITERM)
      GO TO 2200
 2100 CONTINUE
      DLDC(2)=DLDC(2)+TAU**(-JH*ITERM)/(AJH*AITERM*AITERM)
      DLDC(1)=DLDC(1)+ONE/(AJL*AITERM*AITERM)
      DLDC(2)=DLDC(2)-ONE/(AJH*AITERM*AITERM)
 2200 CONTINUE
      RETURN
C     ------------------------------------------------------------------
C        DERIVATIVE OF THE LAMBDA ANOMALY FUNCTION WITH RESPECT TO
C        THE CONSTANTS FOR THE TWO LIMBS OF THE ENTHALPY EQUATION.
```

```
C     ---------------------------------------------------------------
 3000 CONTINUE
      DO 3200 I=1,NTERM
      ITERM=2*I-1
      AITERM=DBLE(ITERM)
      IF(T.GT.TC)  GO TO 3100
      DLDC(1)=DLDC(1)+TAU**(JL*ITERM+1)/(AITERM*(AJL*AITERM+ONE))
      GO TO 3200
 3100 CONTINUE
      DLDC(2)=DLDC(2)+TAU**(-JH*ITERM+1)/(AITERM*(AJH*AITERM+ONE))
      DLDC(1)=DLDC(1)+ONE/(AITERM*(AJL*AITERM+ONE))
      DLDC(2)=DLDC(2)-ONE/(AITERM*(AJH*AITERM+ONE))
 3200 CONTINUE
      DLDC(1)=TC*DLDC(1)
      DLDC(2)=TC*DLDC(2)
      RETURN
C     ---------------------------------------------------------------
C        DERIVATIVE OF THE LAMBDA ANOMALY FUNCTION WITH RESPECT TO
C        THE CONSTANTS FOR THE TWO LIMBS OF THE GIBBS ENERGY EQUATION.
C     ---------------------------------------------------------------
 4000 CONTINUE
      DO 4200 I=1,NTERM
      ITERM=2*I-1
      AITERM=DBLE(ITERM)
      IF(T.GT.TC)  GO TO 4100
      DLDC(1)=DLDC(1)+TC*TAU**(JL*ITERM+1)/(AITERM*(AJL*AITERM+ONE))
     1               -T*TAU**(JL*ITERM)/(AJL*AITERM*AITERM)
      GO TO 4200
 4100 CONTINUE
      DLDC(2)=DLDC(2)+TC*TAU**(-JH*ITERM+1)/(AITERM*(AJH*AITERM+ONE))
     1               -T*TAU**(-JH*ITERM)/(AJH*AITERM*AITERM)
      DLDC(1)=DLDC(1)+TC/(AITERM*(AJL*AITERM+ONE))
     1               -T/(AJL*AITERM*AITERM)
      DLDC(2)=DLDC(2)-TC/(AITERM*(AJH*AITERM+ONE))
     1               +T/(AJH*AITERM*AITERM)
 4200 CONTINUE
      RETURN
C     ---------------------------------------------------------------
C        DERIVATIVE OF THE LAMBDA ANOMALY FUNCTION WITH RESPECT TO
C        THE CONSTANTS FOR THE TWO LIMBS OF THE EQUILIBRIUM CONSTANT
C        EQUATION.
C     ---------------------------------------------------------------
 5000 CONTINUE
      DO 5200 I=1,NTERM
      ITERM=2*I-1
      AITERM=DBLE(ITERM)
      IF(T.GT.TC)  GO TO 5100
      DLDC(1)=DLDC(1)+TC*TAU**(JL*ITERM+1)/(AITERM*(AJL*AITERM+ONE))
     1               -T*TAU**(JL*ITERM)/(AJL*AITERM*AITERM)
      GO TO 5200
 5100 CONTINUE
      DLDC(2)=DLDC(2)+TC*TAU**(-JH*ITERM+1)/(AITERM*(AJH*AITERM+ONE))
     1               -T*TAU**(-JH*ITERM)/(AJH*AITERM*AITERM)
      DLDC(1)=DLDC(1)+TC/(AITERM*(AJL*AITERM+ONE))
```

```
     1              -T/(AJL*AITERM*AITERM)
      DLDC(2)=DLDC(2)-TC/(AITERM*(AJH*AITERM+ONE))
     1              +T/(AJH*AITERM*AITERM)
 5200 CONTINUE
      DLDC(1)=-DLDC(1)/(R*T)
      DLDC(2)=-DLDC(2)/(R*T)
      RETURN
C     ------------------------------------------------------------------
C        DERIVATIVE OF THE LAMBDA ANOMALY FUNCTION WITH RESPECT TO
C        THE CONSTANTS FOR THE TWO LIMBS OF THE EMF EQUATION.
C     ------------------------------------------------------------------
 6000 CONTINUE
      DO 6200 I=1,NTERM
      ITERM=2*I-1
      AITERM=DBLE(ITERM)
      IF(T.GT.TC)  GO TO 6100
      DLDC(1)=DLDC(1)+TC*TAU**(JL*ITERM+1)/(AITERM*(AJL*AITERM+ONE))
     1              -T*TAU**(JL*ITERM)/(AJL*AITERM*AITERM)
      GO TO 6200
 6100 CONTINUE
      DLDC(2)=DLDC(2)+TC*TAU**(-JH*ITERM+1)/(AITERM*(AJH*AITERM+ONE))
     1              -T*TAU**(-JH*ITERM)/(AJH*AITERM*AITERM)
      DLDC(1)=DLDC(1)+TC/(AITERM*(AJL*AITERM+ONE))
     1              -T/(AJL*AITERM*AITERM)
      DLDC(2)=DLDC(2)-TC/(AITERM*(AJH*AITERM+ONE))
     1              +T/(AJH*AITERM*AITERM)
 6200 CONTINUE
      DLDC(1)=-DLDC(1)/F
      DLDC(2)=-DLDC(2)/F
      RETURN
C     ------------------------------------------------------------------
C        DERIVATIVE OF THE LAMBDA ANOMALY FUNCTION WITH RESPECT TO
C        THE CONSTANTS FOR THE TWO LIMBS OF THE HEAT CONTENT EQUATION.
C     ------------------------------------------------------------------
 7000 CONTINUE
      IF(SC.LT.0.D0)  TAU=TLOW/TC
      DO 7200 I=1,NTERM
      ITERM=2*I-1
      AITERM=DBLE(ITERM)
      IF(TAU.GT.ONE)  GO TO 7100
      DLDC(1)=DLDC(1)+TAU**(JL*ITERM+1)/(AITERM*(AJL*AITERM+ONE))
      GO TO 7200
 7100 CONTINUE
      DLDC(2)=DLDC(2)+TAU**(-JH*ITERM+1)/(AITERM*(AJH*AITERM+ONE))
      DLDC(1)=DLDC(1)+ONE/(AITERM*(AJL*AITERM+ONE))
      DLDC(2)=DLDC(2)-ONE/(AITERM*(AJH*AITERM+ONE))
 7200 CONTINUE
      DLDC(1)=TC*DLDC(1)
      DLDC(2)=TC*DLDC(2)
      RETURN
      END
      SUBROUTINE VOLUME(Y,I,J)
C     ------------------------------------------------------------------
C                              VOLUME
```

```
C
C        VOLUME CALCULATES THE MOLAR VOLUME OF A SPECIES FROM
C        A GIVEN TEMPERATURE T AND/OR GIVEN PRESSURE P OR FROM A GIVEN
C        TEMPERATURE DIFFERENTIAL (T-TO) AND/OR A GIVEN PRESSURE
C        DIFFERENTIAL (P-PO).
C
C     WRITTEN BY HAAS IN JANUARY, 1980
C     REVISED BY ROBINSON IN MARCH, L982.
C
C     ----------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*4 TITLE
      REAL*8 NUM
      DIMENSION DVDC(5)
      DIMENSION COEF(10,100),PNAME(20),TINV(10,100,4),IPHASE(10,100),
     1  NPHASE(100),IKOUNT(100),IGO(100),ISTATE(10,100),NINVER(10,100),
     2  INSTAT(10,100),INVPH(10,100,4),INVSC(10,100),ISPECL(10,100)
      DIMENSION X(4,1500),YO(1500),SIGYO(1500),PC(360),KI(360),
     1  DC(360),PD(360),TITLE(20),SIGYON(1500)
      COMMON /EARTH/ COEF,PNAME,TINV,IPHASE,NPHASE,IKOUNT,IGO,NSETS,
     1  ISTATE,NINVER,INSTAT,INVPH,INVSC,LISTP,ISPECL
      COMMON /AIR/X,PC,DC,TITLE,YO,SIGYO,SIGYON,PD,KI,NC,IC,NV,
     1  NX,IW,NP,NO,ISING,ISTOP,IL,JDFLAG,NPARM,NVPARM,NCPARM,NMPARM
      COMMON /WATER/ZERO, ONE, TWO, THREE, FOUR, SIX, R, F,
     1  SCINV, TREF, PREF, STCOEF, DIEO, ADIE, BDIE, THETA, YESNO, TK,
     2  ASTAR, TYPE, NL, NSCALE
      DO 1 N=1,NP
      DC(N)=0.0D0
    1 CONTINUE
      Y=0.0D0
      T=X(1,I)
      TO=X(2,I)
      IF(TO.EQ.0.0D0)  TO=TREF
      P=X(3,I)
      IF(P.EQ.0.0D0)  P=PREF
      PO=X(4,I)
      IF(PO.EQ.0.0D0)  PO=PREF
      RTT=DEXP(-T/3.0D02)
      RTTO=DEXP(-TO/3.0D02)
      RTP=DEXP(-P/3.5D04)
      RTPO=DEXP(-PO/3.5D04)
      LAST=NPHASE(J)
      IGOES=IGO(J)-20
      IF(IGOES.NE.8)  GO TO 10
      Y=FNCALC(X(1,I),J,4)
      DELP=P-PO
      DVDC(1)=DELP
      DVDC(2)=DELP*T
      DVDC(3)=DELP*RTT
      DVDC(4)=(P*P-PO*PO)/2.0D0
      DVDC(5)=-3.50D04*(RTP-RTPO)
   10 CONTINUE
      DO 1000 L=1,LAST
      ACOEF=COEF(L,J)
```

```
      ITEM=NCPARM+NMPARM+NPARM*(IPHASE(L,J)-1)+1
      B1=PC(ITEM)
      B2=PC(ITEM+1)
      B3=PC(ITEM+2)
      B4=PC(ITEM+3)
      B5=PC(ITEM+4)
      GO TO (100,200,300,400,500,600,700,800),IGOES
C     ------------------------------------------------------------------
C     V = B1 + B2XT2 + B3XEXP(-T2/300) + B4XP2 + B5XEXP(-P2/35000)
C     ------------------------------------------------------------------
  100 CONTINUE
      DC(ITEM)=DC(ITEM)+ACOEF*1.0D0
      DC(ITEM+1)=DC(ITEM+1)+ACOEF*T
      DC(ITEM+2)=DC(ITEM+2)+ACOEF*RTT
      DC(ITEM+3)=DC(ITEM+3)+ACOEF*P
      DC(ITEM+4)=DC(ITEM+4)+ACOEF*RTP
      Y=Y+ACOEF*(B1+B2*T+B3*RTT+B4*P+B5*RTP)
      GO TO 1000
C     ------------------------------------------------------------------
C     V-VO=B2(T-TO) + B3(RTT-RTTO) + B4(P-PO) + B5(RTP-RTPO)
C     ------------------------------------------------------------------
  200 CONTINUE
      IF(ACOEF.LT.0.D0)  GO TO 250
      DC(ITEM)=DC(ITEM)+ACOEF
      DC(ITEM+1)=DC(ITEM+1)+ACOEF*T
      DC(ITEM+2)=DC(ITEM+2)+ACOEF*RTT
      DC(ITEM+3)=DC(ITEM+3)+ACOEF*P
      DC(ITEM+4)=DC(ITEM+4)+ACOEF*RTP
      Y=Y+ACOEF*(B1+B2*T+B3*RTT+B4*P+B5*RTP)
      GO TO 1000
  250 CONTINUE
      DC(ITEM)=DC(ITEM)+ACOEF
      DC(ITEM+1)=DC(ITEM+1)+ACOEF*TO
      DC(ITEM+2)=DC(ITEM+2)+ACOEF*RTTO
      DC(ITEM+3)=DC(ITEM+3)+ACOEF*PO
      DC(ITEM+4)=DC(ITEM+4)+ACOEF*RTPO
      Y=Y+ACOEF*(B1+B2*TO+B3*RTTO+B4*PO+B5*RTPO)
      GO TO 1000
C     ------------------------------------------------------------------
C     V      B1 + B2T + B3RTT + B4P + B5RTP
C     -- = ---------------------------------
C     VO   B1 = B2TO + B3RTTO + B4PO + B5RTPO
C     ------------------------------------------------------------------
  300 CONTINUE
      V=B1+B2*T+B3*RTT+B4*P+B5*RTP
      VO=B1+B2*TO+B3*RTTO+B4*PO+B5*RTPO
      VOSQ=1.D0/(VO*VO)
      DC(ITEM)=DC(ITEM)+ACOEF*VOSQ*(VO-V)
      DC(ITEM+1)=DC(ITEM+1)+ACOEF*VOSQ*(VO*T-V*TO)
      DC(ITEM+2)=DC(ITEM+2)+ACOEF*VOSQ*(VO*RTT-V*RTTO)
      DC(ITEM+3)=DC(ITEM+3)+ACOEF*VOSQ*(VO*P-V*PO)
      DC(ITEM+4)=DC(ITEM+4)+ACOEF*VOSQ*(VO*RTP-V*RTPO)
      Y=Y+ACOEF*V/VO
      RETURN
```

```
C     ------------------------------------------------------------------
C     (DV/DT)P =  B2 - (B3/3.0D02)*EXP(-T/300)
C     ------------------------------------------------------------------
  400 CONTINUE
      DC(ITEM)=DC(ITEM)+0.0D0
      DC(ITEM+1)=DC(ITEM+1)+ACOEF*1.0D0
      DC(ITEM+2)=DC(ITEM+2)-(ACOEF/3.0D02)*RTT
      DC(ITEM+3)=DC(ITEM+3)+0.0D0
      DC(ITEM+4)=DC(ITEM+4)+0.0D0
      Y=Y+ACOEF*(B2-B3*RTT/3.0D02)
      GO TO 1000
C     ------------------------------------------------------------------
C                        B2(T-TO) + B3(RTT-RTTO)          1
C     'A|P' =     --------------------------------    ------
C                B1 + B2TO + B3RTTO + B4P + B5RTP    T - TO
C     ------------------------------------------------------------------
  500 CONTINUE
      VO=B1+B2*TO+B3*RTTO+B4*P+B5*RTP
      DT=T-TO
      DC(ITEM)=DC(ITEM)-ACOEF/(VO*VO*DT)
      DC(ITEM+1)=DC(ITEM+1)+ACOEF*(T-TO-TO/(VO*VO*DT))
      DC(ITEM+2)=DC(ITEM+2)+ACOEF*(RTT-RTTO-RTTO/(VO*VO*DT))
      DC(ITEM+3)=DC(ITEM+3)-ACOEF*P/(VO*VO*DT)
      DC(ITEM+4)=DC(ITEM+4)-ACOEF*RTP/(VO*VO*DT)
      Y=ACOEF*(B2*(T-TO)+B3*(RTT-RTTO))/(VO*DT)
      RETURN
C     ------------------------------------------------------------------
C     (DV/DP)T = B4 - (B5/3.5D04)*EXP(-P/3.5D04)
C     ------------------------------------------------------------------
  600 CONTINUE
      DC(ITEM)=DC(ITEM)+0.0D0
      DC(ITEM+1)=DC(ITEM+1)+0.0D0
      DC(ITEM+2)=DC(ITEM+2)+0.0D0
      DC(ITEM+3)=DC(ITEM+3)+ACOEF*1.0D0
      DC(ITEM+4)=DC(ITEM+4)-(ACOEF/3.5D04)*RTP
      Y=Y+ACOEF*(B4-(B5/3.5D04)*RTP)
      GO TO 1000
C     ------------------------------------------------------------------
C                          B4(P-PO) + B5(RTP-RTPO)          1
C     'B|T' = -  ------------------------------------    ------
C                B1 + B2*T + B3*RTT + B4*PO + B5*RTPO    P - PO
C     ------------------------------------------------------------------
  700 CONTINUE
      VO=B1+B2*T+B3*RTT+B4*PO+B5*RTPO
      DP=P-PO
      DC(ITEM)=DC(ITEM)-ACOEF/(VO*VO*DP)
      DC(ITEM+1)=DC(ITEM+1)-ACOEF*T/(VO*VO*DP)
      DC(ITEM+2)=DC(ITEM+2)-ACOEF*RTT/(VO*VO*DP)
      DC(ITEM+3)=DC(ITEM+3)+ACOEF*(P-PO-PO/(VO*VO*DP))
      DC(ITEM+4)=DC(ITEM+4)+ACOEF*(RTP-RTPO-RTPO/(VO*VO*DP))
      Y=ACOEF*(B4*(P-PO)+B5*(RTP-RTPO))/(VO*DP)
      RETURN
C     ------------------------------------------------------------------
C
```

```
C     G(T,P) = G(T,PO) + INTEGRAL ( V DP ) FROM PO TO P.
C
C     ---------------------------------------------------------------
  800 CONTINUE
      DO 900 K=1,NVPARM
      DC(ITEM+K-1)=DC(ITEM+K-1)+0.1D0*DVDC(K)*ACOEF
      Y=Y+0.1D0*PC(ITEM+K-1)*DVDC(K)*ACOEF
  900 CONTINUE
 1000 CONTINUE
      RETURN
 1100 CONTINUE
      WRITE (6,60)IGO(J)
      STOP
   60 FORMAT('0*****   THE VOLUME FUNCTION FOR IGO(J) = ',I3,' IS NOT PR
     1OGRAMMED.   *****')
      END
      SUBROUTINE YDERIV(T,J)
C     ---------------------------------------------------------------
C
C                                   YDERIV
C     THIS VERSION WRITTEN 3/09/72 BY HAAS.
C          LAST REVISED 8/05/72 BY HAAS.
C
C     REVISED TO ACCEPT THE CODATA THERMAL FUNCTION BY HAAS, 8/31/84.
C
C     ---------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*1 LABEL,BCD
      DIMENSION DYDC(10),DDDC(10),DLDC(2),DLDDC(2)
      DIMENSION SCINV(2),STCOEF(10),YESNO(2),TK(19,2),TYPE(28),NSCALE(5)
     1  ,LABEL(50)
C     ---------------------------------------------------------------
C         COMMON BLOCKS
C     ---------------------------------------------------------------
      COMMON /WATER/ZERO, ONE, TWO, THREE, FOUR, SIX, R, F,
     1  SCINV, TREF, PREF, STCOEF, DIEO, ADIE, BDIE, THETA, YESNO, TK,
     2  ASTAR, TYPE, NL, NSCALE
      COMMON /WATCHR/LABEL, BCD
      COMMON /SPACE/ DYDC, DDDC, DLDC, DLDDC, SC, TLOW
C     ---------------------------------------------------------------
C     CALCULATE THE APPROPRIATE TEMPERATURE TERM IN THE DERIVATIVE
C          AND RETURN.  --APPROPRIATE AS INDICATED BY J.
C     ---------------------------------------------------------------
      GO TO (100,200,300,400,500,600,700,900,1000,1100,1200,1300,1400,15
     100), J
  100 CONTINUE
C     ---------------------------------------------------------------
C         THE DERIVATIVE FOR HEAT CAPACITY OF NON-IONIC SPECIES.
C     ---------------------------------------------------------------
      DYDC(1)=ONE/(T*T*T)
      DYDC(2)=ONE/(T*T)
      DYDC(3)=ONE/T
      DYDC(4)=ONE/DSQRT(T)
      DYDC(5)=ONE
```

```
      DYDC(6)=T
      DYDC(7)=T*T
      DYDC(8)=T*T*T
      DYDC(9)=ZERO
      DYDC(10)=ZERO
      RETURN
  200 CONTINUE
C     ------------------------------------------------------------------
C         THE DERIVATIVE FOR ENTROPY OF NON-IONIC SPECIES.
C     ------------------------------------------------------------------
      DYDC(1)=-ONE/(THREE*T*T*T)
      DYDC(2)=-ONE/(TWO*T*T)
      DYDC(3)=-ONE/T
      DYDC(4)=-TWO/DSQRT(T)
      DYDC(5)=DLOG(T)
      DYDC(6)=T
      DYDC(7)=T*T/TWO
      DYDC(8)=T*T*T/THREE
      DYDC(9)=ZERO
      DYDC(10)=ONE
      RETURN
  300 CONTINUE
C     ------------------------------------------------------------------
C         THE DERIVATIVE FOR THE ENTHALPY OF NON-IONIC SPECIES.
C     ------------------------------------------------------------------
      DYDC(1)=-ONE/(TWO*T*T)
      DYDC(2)=-ONE/T
      DYDC(3)=DLOG(T)
      DYDC(4)=TWO*DSQRT(T)
      DYDC(5)=T
      DYDC(6)=T*T/TWO
      DYDC(7)=T*T*T/THREE
      DYDC(8)=T**4/FOUR
      DYDC(9)=ONE
      DYDC(10)=ZERO
      RETURN
  400 CONTINUE
C     ------------------------------------------------------------------
C         THE DERIVATIVE FOR THE GIBBS ENERGY OF NON-IONIC SPECIES.
C     ------------------------------------------------------------------
      DYDC(1)=-ONE/(SIX*T*T)
      DYDC(2)=-ONE/(TWO*T)
      DYDC(3)=ONE+DLOG(T)
      DYDC(4)=FOUR*DSQRT(T)
      DYDC(5)=T-T*DLOG(T)
      DYDC(6)=-T*T/TWO
      DYDC(7)=-T*T*T/SIX
      DYDC(8)=-T**4/12.0D0
      DYDC(9)=ONE
      DYDC(10)=-T
      RETURN
  500 CONTINUE
C     ------------------------------------------------------------------
C         THE DERIVATIVE FOR THE EQUILIBRIUM CONSTANT OF NON-IONIC
```

```
C             SPECIES.
C     ------------------------------------------------------------------
      DYDC(1)=ONE/(SIX*R*T*T*T)
      DYDC(2)=ONE/(TWO*R*T*T)
      DYDC(3)=-(ONE+DLOG(T))/(R*T)
      DYDC(4)=-FOUR/(R*DSQRT(T))
      DYDC(5)=-(ONE-DLOG(T))/R
      DYDC(6)=T/(TWO*R)
      DYDC(7)=T*T/(SIX*R)
      DYDC(8)=T*T*T/(12.0D0*R)
      DYDC(9)=-ONE/(R*T)
      DYDC(10)=ONE/R
      RETURN
  600 CONTINUE
C     ------------------------------------------------------------------
C         THE DERIVATIVE FOR THE CELL POTENTIAL FOR NON-IONIC SPECIES.
C     ------------------------------------------------------------------
      DYDC(1)=ONE/(SIX*F*T*T)
      DYDC(2)=ONE/(TWO*F*T)
      DYDC(3)=-(ONE+DLOG(T))/F
      DYDC(4)=-FOUR*DSQRT(T)/F
      DYDC(5)=-(T-T*DLOG(T))/F
      DYDC(6)=T*T/(TWO*F)
      DYDC(7)=T*T*T/(SIX*F)
      DYDC(8)=(T**4)/(12.0D0*F)
      DYDC(9)=-ONE/F
      DYDC(10)=T/F
      RETURN
  700 CONTINUE
C     ------------------------------------------------------------------
C         THE DERIVATIVE FOR THE RELATIVE HEAT CONTENT OF NON-IONIC
C             SPECIES.
C     ------------------------------------------------------------------
      IF (SC.LT.ZERO)  GO TO 800
      DYDC(1)=-ONE/(TWO*T*T)
      DYDC(2)=-ONE/T
      DYDC(3)=DLOG(T)
      DYDC(4)=TWO*DSQRT(T)
      DYDC(5)=T
      DYDC(6)=T*T/TWO
      DYDC(7)=T*T*T/THREE
      DYDC(8)=T**4/FOUR
      DYDC(9)=ONE
      DYDC(10)=ZERO
      RETURN
  800 CONTINUE
      DYDC(1)=-ONE/(TWO*TLOW*TLOW)
      DYDC(2)=-ONE/TLOW
      DYDC(3)=DLOG(TLOW)
      DYDC(4)=TWO*DSQRT(TLOW)
      DYDC(5)=TLOW
      DYDC(6)=TLOW*TLOW/TWO
      DYDC(7)=TLOW*TLOW*TLOW/THREE
      DYDC(8)=TLOW**4/FOUR
```

```
      DYDC(9)=ONE
      DYDC(10)=ZERO
      RETURN
  900 CONTINUE
C     ------------------------------------------------------------------
C         THE DERIVATIVE FOR THE HEAT CAPACITY OF IONIC SPECIES.
C     ------------------------------------------------------------------
      DYDC(1)=ONE
      DYDC(2)=TWO*T
      DYDC(3)=ONE/(T*T)
      DYDC(4)=ZERO
      DYDC(5)=ZERO
      DYDC(6)=T*T
      FNT=FN(T)
      DYDC(7)=-T*DIE(T)*(ADIE*FNT*ADIE*FNT+ADIE*FNT*ADIE+ADIE*FNT*TWO/
     1THETA+ONE/(THETA*THETA))
      DYDC(8)=ZERO
      DYDC(9)=ZERO
      DYDC(10)=ZERO
      RETURN
 1000 CONTINUE
C     ------------------------------------------------------------------
C         THE DERIVATIVE FOR THE ENTROPY OF IONIC SPECIES.
C     ------------------------------------------------------------------
      DYDC(1)=DLOG(T)
      DYDC(2)=TWO*T
      DYDC(3)=-ONE/(TWO*T*T)
      DYDC(4)=ZERO
      DYDC(5)=ONE
      DYDC(6)=T*T/TWO
      DYDC(7)=-DIE(T)*(ADIE*FN(T)+ONE/THETA)
      DYDC(8)=ZERO
      DYDC(9)=ZERO
      DYDC(10)=ZERO
      RETURN
 1100 CONTINUE
C     ------------------------------------------------------------------
C         THE DERIVATIVE FOR THE ENTHALPY OF IONIC SPECIES.
C     ------------------------------------------------------------------
      DYDC(1)=T
      DYDC(2)=T*T
      DYDC(3)=-ONE/T
      DYDC(4)=ONE
      DYDC(5)=ZERO
      DYDC(6)=(T**3)/THREE
      DYDC(7)=DIE(T)*(ONE-ADIE*T*FN(T)-T/THETA)
      DYDC(8)=ZERO
      DYDC(9)=ZERO
      DYDC(10)=ZERO
      RETURN
 1200 CONTINUE
C     ------------------------------------------------------------------
C         THE DERIVATIVE FOR THE GIBBS ENERGY OF IONIC SPECIES.
C     ------------------------------------------------------------------
```

```
      DYDC(1)=T-T*DLOG(T)
      DYDC(2)=-T*T
      DYDC(3)=-ONE/(TWO*T)
      DYDC(4)=ONE
      DYDC(5)=-T
      DYDC(6)=-(T**3)/SIX
      DYDC(7)=DIE(T)
      DYDC(8)=ZERO
      DYDC(9)=ZERO
      DYDC(10)=ZERO
      RETURN
 1300 CONTINUE
C     ------------------------------------------------------------------
C         THE DERIVATIVE FOR THE EQUILIBRIUM CONSTANT OF IONIC
C             SPECIES.
C     ------------------------------------------------------------------
      DYDC(1)=(DLOG(T)-ONE)/R
      DYDC(2)=T/R
      DYDC(3)=ONE/(TWO*R*T*T)
      DYDC(4)=-ONE/(R*T)
      DYDC(5)=ONE/R
      DYDC(6)=(T*T)/(SIX*R)
      DYDC(7)=-DIE(T)/(R*T)
      DYDC(7)=-DIE(T)/(R*T)
      DYDC(8)=ZERO
      DYDC(9)=ZERO
      DYDC(10)=ZERO
      RETURN
 1400 CONTINUE
C     ------------------------------------------------------------------
C         THE DERIVATIVE FOR THE CELL POTENTIAL OF IONIC SPECIES.
C     ------------------------------------------------------------------
      DYDC(1)=(T*DLOG(T)-T)/F
      DYDC(2)=(T*T)/F
      DYDC(3)=ONE/(TWO*F*T)
      DYDC(4)=-ONE/F
      DYDC(5)=T/F
      DYDC(6)=(T**3)/(SIX*F)
      DYDC(7)=-DIE(T)/F
      DYDC(8)=ZERO
      DYDC(9)=ZERO
      DYDC(10)=ZERO
      RETURN
 1500 CONTINUE
C     ------------------------------------------------------------------
C         THE DERIVATIVE FOR THE RELATIVE HEAT CONTENT OF IONIC
C             SPECIES.
C     ------------------------------------------------------------------
      IF (SC.LT.0) GO TO 1600
      DYDC(1)=T
      DYDC(2)=T*T
      DYDC(3)=-ONE/T
      DYDC(4)=ONE
      DYDC(5)=ZERO
```

```
      DYDC(6)=(T**3)/THREE
      DYDC(7)=DIE(T)*(ONE-ADIE*T*FN(T)-T/THETA)
      DYDC(8)=ZERO
      DYDC(9)=ZERO
      DYDC(10)=ZERO
      RETURN
 1600 CONTINUE
      DYDC(1)=TLOW
      DYDC(2)=TLOW*TLOW
      DYDC(3)=-ONE/TLOW
      DYDC(4)=ONE
      DYDC(5)=ZERO
      DYDC(6)=(TLOW**3)/THREE
      DYDC(7)=DIE(TLOW)*(ONE-ADIE*TLOW*FN(TLOW)-TLOW/THETA)
      DYDC(8)=ZERO
      DYDC(9)=ZERO
      DYDC(10)=ZERO
      RETURN
C     ------------------------------------------------------------------
      END
      SUBROUTINE DDERIV(T)
C     ------------------------------------------------------------------
C
C                                   DDERIV
C
C     THIS VERSION WAS WRITTEN 3/09/72 BY HAAS.
C          LAST REVISED 8/05/72 BY HAAS.
C
C     REVISED TO ACCEPT THE CODATA THERMAL FUNCTION BY HAAS, 8/31/84.
C
C     ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*1 LABEL,BCD
      DIMENSION SCINV(2),STCOEF(10),YESNO(2),TK(19,2),TYPE(28),NSCALE(5)
     1  ,LABEL(50)
      DIMENSION DYDC(10),DDDC(10),DLDC(2),DLDDC(2)
C     ------------------------------------------------------------------
C         COMMON BLOCKS
C     ------------------------------------------------------------------
      COMMON /WATER/ZERO, ONE, TWO, THREE, FOUR, SIX, R, F,
     1  SCINV, TREF, PREF, STCOEF, DIEO, ADIE, BDIE, THETA, YESNO, TK,
     2  ASTAR, TYPE, NL, NSCALE
      COMMON /WATCHR/LABEL, BCD
      COMMON /SPACE/ DYDC, DDDC, DLDC, DLDDC, SC, TLOW
C     ------------------------------------------------------------------
C     CALCULATE THE DERIVATIVE OF -D- WITH RESPECT TO THE OTHER
C          CONSTANTS WHERE THE GIBBS ENERGY IS 0.0 AS IS THE
C          CASE FOR ELEMENTS AT 298.15 KELVIN OR AT THE TEMPERATURE OF
C          INVERSION.
C     ------------------------------------------------------------------
      DDDC(1)=ONE/(SIX*T*T)
      DDDC(2)=ONE/(TWO*T)
      DDDC(3)=-(ONE+DLOG(T))
      DDDC(4)=-FOUR*DSQRT(T)
```

```
      DDDC(5)=T*DLOG(T)-T
      DDDC(6)=T*T/TWO
      DDDC(7)=T*T*T/SIX
      DDDC(8)=T**4/12.0D0
      DDDC(9)=-ONE
      DDDC(10)=T
      RETURN
      END
      SUBROUTINE LDDRIV(T,INDEX,JL,JH)
C     ------------------------------------------------------------------
C
C                           LDDRIV
C
C     LDDRIV CALCULATES THE DERIVATIVES FOR THE LAMBDA ANOMALY
C        FUNCTIONS WITH RESPECT TO THE CONSTANTS FOR BOTH LIMBS
C        FOR INCLUSION AT TEMPERATURES OF TRANSITION OR FOR REFERENCE
C        STATE CALCULATIONS.
C
C     REFERENCE:  GUILLERMET & GUSTAFSON (1984)
C                 AGREN (1979)
C                 HILLERT & JARL (1975).
C
C     WRITTEN 10/84 BY HAAS.
C     ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*4 TITLE
      CHARACTER*1 LABEL,BCD
      DIMENSION X(4,1500),YO(1500),SIGYO(1500),P(360),KI(360),
     1  DC(360),PD(360),TITLE(20),SIGYON(1500)
      DIMENSION SCINV(2),STCOEF(10),YESNO(2),TK(19,2),TYPE(28),NSCALE(5)
     1  ,LABEL(50)
      DIMENSION DYDC(10),DDDC(10),DLDC(2),DLDDC(2)
      COMMON /AIR/X,P,DC,TITLE,YO,SIGYO,SIGYON,PD,KI,NC,IC,NV,
     1  NX,IW,NP,NO,ISING,ISTOP,IL,JN,NPARM,NVPARM,NCPARM,NMPARM
      COMMON /WATER/ZERO, ONE, TWO, THREE, FOUR, SIX, R, F,
     1  SCINV, TREF, PREF, STCOEF, DIEO, ADIE, BDIE, THETA, YESNO, TK,
     2  ASTAR, TYPE, NL, NSCALE
      COMMON /WATCHR/LABEL, BCD
      COMMON /SPACE/ DYDC, DDDC, DLDC, DLDDC, SC, TLOW
      NTERM=7
      TC=P(INDEX+NCPARM)
      TAU=T/TC
      AJL=DBLE(JL)
      AJH=DBLE(-JH)
      DLDDC(1)=ZERO
      DLDDC(2)=ZERO
      DO 2000 I=1,NTERM
      ITERM=2*I-1
      AITERM=DBLE(ITERM)
      IF(T.GT.TC)  GO TO 1000
      DLDDC(1)=DLDDC(1)-TC*TAU**(JL*ITERM+1)/(AITERM*(AJL*AITERM+ONE))
     1              +T*TAU**(JL*ITERM)/(AJL*AITERM*AITERM)
      GO TO 2000
 1000 CONTINUE
```

```
      DLDDC(2)=DLDDC(2)-TC*TAU**(-JH*ITERM+1)/(AITERM*(AJH*AITERM+ONE))
     1              +T*TAU**(-JH*ITERM)/(AJH*AITERM*AITERM)
      DLDDC(1)=DLDDC(1)-TC/(AITERM*(AJL*AITERM+ONE))
     1              +T/(AJL*AITERM*AITERM)
      DLDDC(2)=DLDDC(2)+TC/(AITERM*(AJH*AITERM+ONE))
     1              -T/(AJH*AITERM*AITERM)
 2000 CONTINUE
      RETURN
      END
      FUNCTION DIE(T)
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*1 LABEL,BCD
      DIMENSION SCINV(2),STCOEF(10),YESNO(2),TK(19,2),TYPE(28),NSCALE(5)
     1  ,LABEL(50)
C     ------------------------------------------------------------------
C         COMMON BLOCKS
C     ------------------------------------------------------------------
      COMMON /WATER/ZERO, ONE, TWO, THREE, FOUR, SIX, R, F,
     1  SCINV, TREF, PREF, STCOEF, DIEO, ADIE, BDIE, THETA, YESNO, TK,
     2  ASTAR, TYPE, NL, NSCALE
      COMMON /WATCHR/LABEL, BCD
      DIE=DEXP(FN(T)+T/THETA)/DIEO
      RETURN
      END
      FUNCTION FN(T)
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*1 LABEL,BCD
      DIMENSION SCINV(2),STCOEF(10),YESNO(2),TK(19,2),TYPE(28),NSCALE(5)
     1  ,LABEL(50)
C     ------------------------------------------------------------------
C         COMMON BLOCKS
C     ------------------------------------------------------------------
      COMMON /WATER/ZERO, ONE, TWO, THREE, FOUR, SIX, R, F,
     1  SCINV, TREF, PREF, STCOEF, DIEO, ADIE, BDIE, THETA, YESNO, TK,
     2  ASTAR, TYPE, NL, NSCALE
      COMMON /WATCHR/LABEL, BCD
      FN=DEXP(BDIE+ADIE*T)
      RETURN
      END
      FUNCTION GEF(T,J)
C     ------------------------------------------------------------------
C     FOR A DISCUSSION OF THE 'THIRD-LAW' TEST USING THE GIBBS
C        ENERGY FUNCTION, REFER TO--
C
C        STULL AND PROPHET, 1971, JANAF THERMOCHEMICAL TABLES,
C          NATL. BUR. STANDARDS, NSRDS-NBS 37, ESPECIALLY PARAGRAPH
C          B2, PAGE 4.
C
C        --BY HAAS, 9/11/74.
C
C     ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*1 LABEL,BCD
      DIMENSION SCINV(2),STCOEF(10),YESNO(2),TK(19,2),TYPE(28),NSCALE(5)
```

```
     1  ,LABEL(50)
C     ----------------------------------------------------------------
C         COMMON BLOCKS
C     ----------------------------------------------------------------
      COMMON /WATER/ZERO, ONE, TWO, THREE, FOUR, SIX, R, F,
     1  SCINV, TREF, PREF, STCOEF, DIEO, ADIE, BDIE, THETA, YESNO, TK,
     2  ASTAR, TYPE, NL, NSCALE
      COMMON /WATCHR/LABEL, BCD
      GEF=(FNCALC(T,J,4)-FNCALC(TREF,J,3))/T
      RETURN
      END
      FUNCTION PTGEF(T,P,J)
C     ----------------------------------------------------------------
C
C                 PTGEF
C
C     PTGEF CALCULATES THE GIBBS ENERGY DIFFERENCE BETWEEN P AND
C        PO(=PREF) AT T, THE TEMPERATURE OF THE OBSERVATION.  THE
C        FUNCTION IS NEEDED TO CORRECT THE GIBBS ENERGY TO PO SO
C        THAT THE THIRD-LAW ENTHALPY OF REACTION CAN BE CALCULATED AT
C        TREF.  THE CALCULATION IS DEPENDENT ONLY ON THE CURRENT VALUES
C        FOR THE VOLUME OF THE PHASES INVOLVED.
C
C     WRITTEN 6/82 BY HAAS.
C
C     ----------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*4 TITLE
      CHARACTER*1 LABEL,BCD
      DIMENSION SCINV(2),STCOEF(10),YESNO(2),TK(19,2),TYPE(28),NSCALE(5)
     1  ,LABEL(50)
      DIMENSION DVDC(5)
      DIMENSION COEF(10,100),PNAME(20),TINV(10,100,4),IPHASE(10,100),
     1  NPHASE(100),IKOUNT(100),IGO(100),ISTATE(10,100),NINVER(10,100),
     2  INSTAT(10,100),INVPH(10,100,4),INVSC(10,100),ISPECL(10,100)
      DIMENSION X(4,1500),YO(1500),SIGYO(1500),PC(360),KI(360),
     1  DC(360),PD(360),TITLE(20),SIGYON(1500)
      COMMON /EARTH/ COEF,PNAME,TINV,IPHASE,NPHASE,IKOUNT,IGO,NSETS,
     1  ISTATE,NINVER,INSTAT,INVPH,INVSC,LISTP,ISPECL
      COMMON /AIR/X,PC,DC,TITLE,YO,SIGYO,SIGYON,PD,KI,NC,IC,NV,
     1  NX,IW,NP,NO,ISING,ISTOP,IL,JDFLAG,NPARM,NVPARM,NCPARM,NMPARM
      COMMON /WATER/ZERO, ONE, TWO, THREE, FOUR, SIX, R, F,
     1  SCINV, TREF, PREF, STCOEF, DIEO, ADIE, BDIE, THETA, YESNO, TK,
     2  ASTAR, TYPE, NL, NSCALE
      COMMON /WATCHR/LABEL, BCD
      Y=0.0D0
      TO=TREF
      PO=PREF
      RTT=DEXP(-T/3.0D2)
      RTTO=DEXP(-TO/3.0D2)
      RTP=DEXP(-P/3.5D4)
      RTPO=DEXP(-PO/3.5D4)
      LAST=NPHASE(J)
      DELP=P-PO
```

```
      DVDC(1)=DELP
      DVDC(2)=DELP*T
      DVDC(3)=DELP*RTT
      DVDC(4)=(P*P-PO*PO)/2.0D0
      DVDC(5)=-3.50D4*(RTP-RTPO)
      DO 2000 L=1,LAST
      ACOEF=COEF(L,J)
      ITEM=NCPARM+NMPARM+NPARM*(IPHASE(L,J)-1)+1
      DO 1000 K=1,NVPARM
      Y=Y+0.1D0*PC(ITEM+K-1)*DVDC(K)*ACOEF
 1000 CONTINUE
 2000 CONTINUE
      PTGEF=-Y
      RETURN
      END
      SUBROUTINE PUTOUT(J)
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*23 DATE
      CHARACTER*1 BCD,LABEL
      REAL*4 TITLE,ACOEF
C     ------------------------------------------------------------------
C
C     NOTE:
C     TO CHANGE NPHASE(MAX), I.E. CHANGE "(10,", CHANGE A(11,... ALSO.
C
C     ------------------------------------------------------------------
      DIMENSION SGN(2),A(11,18)
      DIMENSION COEF(10,100),PNAME(20),TINV(10,100,4),IPHASE(10,100),
     1  NPHASE(100),IKOUNT(100),IGO(100),ISTATE(10,100),NINVER(10,100),
     2  INSTAT(10,100),INVPH(10,100,4),INVSC(10,100),ISPECL(10,100)
      DIMENSION X(4,1500),YO(1500),SIGYO(1500),P(360),KI(360),
     1  DC(360),PD(360),TITLE(20),SIGYON(1500)
      DIMENSION SCINV(2),STCOEF(10),YESNO(2),TK(19,2),TYPE(28),NSCALE(5)
     1  ,LABEL(50)
      DIMENSION DYDC(10),DDDC(10),DLDC(2),DLDDC(2)
      DIMENSION AA (360),ION(20)
C     ------------------------------------------------------------------
C         COMMON BLOCKS
C     ------------------------------------------------------------------
      COMMON /EARTH/ COEF,PNAME,TINV,IPHASE,NPHASE,IKOUNT,IGO,NSETS,
     1  ISTATE,NINVER,INSTAT,INVPH,INVSC,LISTP,ISPECL
      COMMON /AIR/X,P,DC,TITLE,YO,SIGYO,SIGYON,PD,KI,NC,IC,NV,
     1  NX,IW,NP,NO,ISING,ISTOP,IL,JDFLAG,NPARM,NVPARM,NCPARM,NMPARM
      COMMON /WATER/ZERO, ONE, TWO, THREE, FOUR, SIX, R, F,
     1  SCINV, TREF, PREF, STCOEF, DIEO, ADIE, BDIE, THETA, YESNO, TK,
     2  ASTAR, TYPE, NL, NSCALE
      COMMON/WATCHR/LABEL, BCD
      COMMON /TIME/ DATE
      COMMON /SPACE/ DYDC, DDDC, DLDC, DLDDC, SC, TLOW
      COMMON /MAN/ AA, ION
      CPRIME=ZERO
      LSTPHA=NPHASE(J)
      DO 1200 L=1,LSTPHA
      INDEX=1+NPARM*(IPHASE(L,J)-1)
```

```
      D=0.0D0
      IF (ISTATE(L,J).NE.1) GO TO 200
      CALL DDERIV(TREF)
      DO 100 K=1,NCPARM
      D=D+P(INDEX+K-1)*DDDC(K)
  100 CONTINUE
      GO TO 900
  200 CONTINUE
      IF (NINVER(L,J).EQ.0) GO TO 900
      DO 300 I=1,2
      SGN(I)=SCINV(I)
  300 CONTINUE
      IF (INVSC(L,J).EQ.0) GO TO 500
      DO 400 LLL=1,2
      SGN(LLL)=SGN(LLL)*STCOEF(INVSC(L,J))
  400 CONTINUE
  500 CONTINUE
      IF (INSTAT(L,J).EQ.0) GO TO 700
      INDEX2=1+NPARM*(INVPH(L,J,1)-1)
      CALL DDERIV(TREF)
      DO 600 K=1,NCPARM
      D=D+P(INDEX2+K-1)*DDDC(K)*SGN(2)
  600 CONTINUE
  700 CONTINUE
      LSTINV=NINVER(L,J)
      DO 800 LST=1,LSTINV
      IF ((LST.EQ.LSTINV).AND.(INVSC(L,J).NE.0)) SGN(2)=SGN(2)/STCOEF(IN
     1VSC(L,J))
      CALL DDERIV(TINV(L,J,LST))
      DO 800 LLL=1,2
      INDEX2=1+NPARM*(INVPH(L,J,LST+LLL-1)-1)
      DO 800 K=1,NCPARM
      D=D+P(INDEX2+K-1)*DDDC(K)*SGN(LLL)
  800 CONTINUE
  900 CONTINUE
      DO 1000 K=1,NPARM
      A(L,K)=P(INDEX+K-1)
 1000 CONTINUE
      A(L,9)=A(L,9)+D
      DO 1100 K=1,NPARM
      AA(INDEX+K-1)=A(L,K)
 1100 CONTINUE
 1200 CONTINUE
      LST=LSTPHA+1
      DO 1300 K=1,NPARM
      A(LST,K)=0.0D0
 1300 CONTINUE
      DO 1500 L=1,LSTPHA
      DO 1500 K=1,NPARM
      IF ((K.EQ.7).AND.(ISTATE(L,J).EQ.-1)) GO TO 1400
      A(LST,K)=A(LST,K)+A(L,K)*COEF(L,J)
      GO TO 1500
 1400 CPRIME=CPRIME+A(L,K)*COEF(L,J)
 1500 CONTINUE
```

```
      RETURN
      ENTRY PUTIT (J)
c     WRITE (8,1900) DATE
      DO 1600 L=1,LSTPHA
      ACOEF=COEF(L,J)
c     WRITE (8,2000) PNAME(IPHASE(L,J)),ACOEF,(A(L,K),K=1,NPARM)
 1600 CONTINUE
      IF ((NPHASE(J).EQ.1).OR.((IGO(J).GE.7).AND.(IGO(J).NE.28)))
     1   GO TO 1800
      IF (CPRIME.EQ.ZERO) GO TO 1700
c     WRITE (8,2200)
c     WRITE (8,2300) (A(LST,K),K=1,NPARM),CPRIME
      GO TO 1800
 1700 CONTINUE
c     WRITE (8,2200)
c     WRITE (8,2300) (A(LST,K),K=1,NPARM)
 1800 CONTINUE
      RETURN
 1900 FORMAT(1H0,A23,10H----------/6H0PHASE,8X,4HCOEF,9X,9HC(1/7/13),
     1  9X,9HC(2,8,14),9X,9HC(3,9,15),9X,10HC(4,10,16),8X,10HC(5,11,17),
     1  8X,10HC(6,12,18))
 2000 FORMAT (1H0,A8,4X,F6.3,1X,1P6D18.7/(1h ,19X,1P6D18.7))
 2100 FORMAT(1H0/1H ,26X,9HC(1/7/13),
     1  9X,9HC(2,8,14),9X,9HC(3,9,15),9X,10HC(4,10,16),8X,10HC(5,11,17),
     1  8X,10HC(6,12,18))
 2200 FORMAT(1H0/1H ,26X,9HC(1/7/13),
     1  9X,9HC(2,8,14),9X,9HC(3,9,15),9X,10HC(4,10,16),8X,10HC(5,11,17),
     1  8X,10HC(6,12,18))
 2300 FORMAT (20H0REACTION CONSTANTS ,1P6D18.7/(1H ,19X,1P6D18.7))
      END
      SUBROUTINE PRPLOT
C     ------------------------------------------------------------------
C
C          MODIFIED TO BE RUN WITH <PHAS20> BY HAAS.
C
C     ------------------------------------------------------------------
      IMPLICIT LOGICAL(W), LOGICAL(K)
      DIMENSION  NSCALE(5), ABNOS(26), X(1), Y(1)
      CHARACTER*1 NOS(10)
      CHARACTER*1 WL
      CHARACTER*1 IMAGE(5000),CH,LABEL(50)
      CHARACTER*1 VC,HC,FOR1(20),FOR2(17),FOR3(20),NC,BL,HF,HF1
      CHARACTER*20 FOX1,FOX3
      CHARACTER*17 FOX2
      CHARACTER*1 VCR
      COMMON/FMT/FOX1,FOX2,FOX3
      EQUIVALENCE (VC,VCR)
      DATA NOS/'0','1','2','3','4','5','6','7','8','9'/
      DATAHC/'-'/,NC/'+'/,BL/' '/,HF/'F'/,HF1/'.'/
C     DATA FOX1/20H(1X,A1,F9.2,  121A1)/
C     DATA FOX2/17H(1X,A1, 9X,121A1)/
C     DATA FOX3/20H(1H0,F  . ,  F   . )/
      DATA FOR1/'(','1','X',',','A','1',',','F','9','.','2',',',' ',
     1  ' ','1','2','1','A','1',')'/
```

```
      DATA FOR2/'(','1','X',',','A','1',',',' ','9','X',',','1','2',
     1  '1','A','1',')'/
      DATA FOR3/'(','1','H','0',',','F',' ',' ','.',' ',',',' ',' ',
     1  'F',' ',' ',' ','.',' ',')'/
      DATA VCR/'|'/
      DATA KPLOT1/.FALSE./,KPLOT2/.FALSE./
      DATA KABSC,KORD,KBOTGL/3*.FALSE./
C
      ENTRY PLOT1 (NSCALE,NHL,NSBH,NVL,NSBV)
      KPLOT1=.TRUE.
      KPLOT2=.FALSE.
      NH=IABS(NHL)
      NSH=IABS(NSBH)
      NV=IABS(NVL)
      NSV=IABS(NSBV)
      NSCL=NSCALE(1)
      IF (NH*NSH*NV*NSV.NE.0) GO TO 200
      WRITE (6,100)
100      FORMAT (T5,'SOME PLOT1 ARG. ILLEGALLY 0')
      KPLOT=.FALSE.
      RETURN
200      KPLOT=.TRUE.
      IF (NV.LE.25) GO TO 400
      WRITE (6,300)
      KPLOT=.FALSE.
300      FORMAT (T5,'NO. OF VERTICAL LINES >25')
      RETURN
400      CONTINUE
      NVM=NV-1
      NVP=NV+1
      NDH=NH*NSH
      NDHP=NDH+1
      NDV=NV*NSV
      NDVP=NDV+1
      NIMG=(NDHP*NDVP)
      IF (NDV.LE.120) GO TO 600
      KPLOT=.FALSE.
      WRITE (6,500)
500      FORMAT (T5,'WIDTH OF GRAPH >121')
      RETURN
600      CONTINUE
      IF (NSCL.EQ.0) GO TO 700
          FSY=10.**NSCALE(2)
          FSX=10.**NSCALE(4)
          IY=MIN0(IABS(NSCALE(3)),7)+1
          IX=MIN0(IABS(NSCALE(5)),9)+1
      GO TO 800
700      FSY=1.
      FSX=1.
      IY=4
      IX=4
800      FOR1(11)=NOS(IY)
      NA=MIN0(IX,NSV)-1
      NS=NA-MIN0(NA,120-NDV)
```

```
      NB=11-NS+NA
      I1=NB/10
      I2=NB-I1*10
      FOR3(7)=NOS(I1+1)
      FOR3(8)=NOS(I2+1)
      FOR3(10)=NOS(NA+1)
      IF (NV.GT.0) GO TO 1000
      DO 900 J=12,19
900      FOR3(J)=BL
      GO TO 1100
1000      I1=NV/10
      I2=NV-I1*10
      FOR3(12)=NOS(I1+1)
      FOR3(13)=NOS(I2+1)
      FOR3(14)=HF
      I1=NSV/100
      I3=NSV-I1*100
      I2=I3/10
      I3=I3-I2*10
      FOR3(15)=NOS(I1+1)
      FOR3(16)=NOS(I2+1)
      FOR3(17)=NOS(I3+1)
      FOR3(18)=HF1
      FOR3(19)=FOR3(10)
1100      CONTINUE
c     ENCODE (20,3400,FOX1) (FOR1(I),I=1,20)
c     ENCODE (17,3500,FOX2) (FOR2(I),I=1,17)
c     ENCODE (20,3400,FOX3) (FOR3(I),I=1,20)
      do 1110 i=1,20
        fox1(i:i)=for1(i)
        fox3(i:i)=for3(i)
 1110 continue
      do 1120 i=1,17
        fox2(i:i)=for2(i)
 1120 continue
      IF (KPLOT1) RETURN
      KPLOT1=.TRUE.
C
      ENTRY PLOT2(IMAGE,XMAX,XMIN,YMAX,YMIN)
      KPLOT2=.TRUE.
      IF (KPLOT1) GO TO 1200
      NSCL=0
      NH=5
      NSH=10
      NV=10
      NSV=10
      GO TO 200
1200      CONTINUE
      IF (.NOT.KPLOT) RETURN
      YMX=YMAX
      DH=(YMAX-YMIN)/FLOAT(NDH)
      DV=(XMAX-XMIN)/FLOAT(NDV)
      DO 1300 I=1,NVP
1300      ABNOS(I)=(XMIN+FLOAT((I-1)*NSV)*DV)*FSX
```

```
      DO 1400 I=1,NIMG
1400      IMAGE(I)=BL
      DO 1800 I=1,NDHP
      I2=I*NDVP
      I1=I2-NDV
      KNHOR=MOD(I-1,NSH).NE.0
      IF (KNHOR) GO TO 1600
      DO 1500 J=I1,I2
1500      IMAGE(J)=HC
1600      CONTINUE
      DO 1800 J=I1,I2,NSV
      IF (KNHOR) GO TO 1700
      IMAGE(J)=NC
      GO TO 1800
1700      IMAGE(J)=VC
1800      CONTINUE
      XMIN1=XMIN-DV/2.
      YMIN1=YMIN-DH/2.
      RETURN
C
      ENTRY PLOT3(IMAGE,CH,X,Y,N3)
      IF (KPLOT2) GO TO 2100
1900      WRITE (6,2000)
2000      FORMAT (T5,'PLOT2 MUST BE CALLED')
2100      CONTINUE
      IF (.NOT.KPLOT) RETURN
      IF (N3.GT.0) GO TO 2300
      KPLOT=.FALSE.
      WRITE (6,2200)
2200      FORMAT (T5,'PLOT3, ARG4 <.OR.= 0')
      RETURN
2300      DO 3000 I=1,N3
      IF (DV) 2500,2400,2500
2400      DUM1=0
      GO TO 2600
2500      CONTINUE
      DUM1=(X(I)-XMIN1)/DV
2600      IF (DH) 2800,2700,2800
2700      DUM2=0
      GO TO 2900
2800      CONTINUE
      DUM2=(Y(I)-YMIN1)/DH
2900      CONTINUE
      IF (DUM1.LT.0..OR.DUM2.LT.0.) GO TO 3000
      IF (DUM1.GE.NDVP.OR.DUM2.GE.NDHP) GO TO 3000
      NX=1+INT(DUM1)
      NY=1+INT(DUM2)
      J=(NDHP-NY)*NDVP+NX
      IMAGE(J)=CH
3000      CONTINUE
      RETURN
C
      ENTRY PLOT4(NL,LABEL,IMAGE)
      ENTRY FPLOT4(NL,LABEL,IMAGE)
```

```
      IF (.NOT.KPLOT) RETURN
      IF (.NOT.KPLOT2) GO TO 1900
      DO 3200 I=1,NDHP
      IF (I.EQ.NDHP.AND.KBOTGL) GO TO 3200
      WL=BL
      IF (I.LE.NL) WL=LABEL(I)
      I2=I*NDVP
      I1=I2-NDV
      IF (MOD(I-1,NSH).EQ.0.AND..NOT.KORD) GO TO 3100
      WRITE (8,FOX2) WL,(IMAGE(J),J=I1,I2)
      GO TO 3200
3100      CONTINUE
      ORDNO=(YMX-FLOAT(I-1)*DH)*FSY
      WRITE (8,FOX1) WL,ORDNO,(IMAGE(J),J=I1,I2)
3200      CONTINUE
      IF (KABSC) GO TO 3300
      WRITE (8,FOX3) (ABNOS(J),J=1,NVP)
3300      RETURN
C
      ENTRY OMIT(LSW)
      KABSC=MOD(LSW,2).EQ.1
      KORD=MOD(LSW,4).GE.2
      KBOTGL=LSW.GE.4
      RETURN
3400      FORMAT(20A1)
3500      FORMAT(17A1)
      END
      SUBROUTINE UNIQUE(YC,I,J,JDFLAG)
C     ------------------------------------------------------------------
C
C         UNIQUE......
C
C     ------------------------------------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION COEF(10,100),PNAME(20),TINV(10,100,4),IPHASE(10,100),
     1  NPHASE(100),IKOUNT(100),IGO(100),ISTATE(10,100),NINVER(10,100),
     2  INSTAT(10,100),INVPH(10,100,4),INVSC(10,100),ISPECL(10,100)
      COMMON /EARTH/ COEF,PNAME,TINV,IPHASE,NPHASE,IKOUNT,IGO,NSETS,
     1  ISTATE,NINVER,INSTAT,INVPH,INVSC,LISTP,ISPECL
      IGOES=IGO(J)-7
C     ------------------------------------------------------------------
C     IF YOU WISH TO PROGRAM SPECIAL RELATIONS, INSERT A "CALL"
C        STATEMENT AND A "RETURN" STATEMENT AFTER THE APPROPRIATE
C        CONTINUE STATEMENT AS INDICATED BY IGOES.
C        AN EXAMPLE IS GIVEN BY COMMENT STATEMENTS FOR IGOES = 8.
C     ------------------------------------------------------------------
      GO TO (8,9,10,11,12,13,14,15,16,17,18,19,20,9999),
     1          IGOES
    8 CONTINUE
C     CALL DUMMY(YC,I,J,JDFLAG)
C     RETURN
    9 CONTINUE
   10 CONTINUE
   11 CONTINUE
```

```
  12 CONTINUE
  13 CONTINUE
  14 CONTINUE
  15 CONTINUE
  16 CONTINUE
  17 CONTINUE
  18 CONTINUE
  19 CONTINUE
  20 CONTINUE
9999 CONTINUE
     WRITE (6,60) J,J,IGO(J)
     STOP
  60 FORMAT (6HOYOUR ,I3,39H-TH DATA SET CALLED UNIQUE FROM EAFWC ./5H
    1IGO(,I2,17H) HAS A VALUE OF ,I6,38H BUT UNIQUE IS CURRENTLY UNPROG
    2RAMMED.)
     END
     SUBROUTINE CVDATE
     INTEGER*2 IYEAR,IMONTH,IDAY,IHOUR,IMIN,ISEC,IHSEC
     CHARACTER*23 DATE
     DIMENSION NUM(14)
     COMMON /TIME/ DATE
     CALL GETDAT(IYEAR,IMONTH,IDAY)
     CALL GETTIM(IHOUR,IMIN,ISEC,IHSEC)
     IYEAR=IYEAR-1900
     NUM(1)=INT(IMONTH/10)
     NUM(2)=IMONTH-NUM(1)*10+48
     NUM(3)=47
     NUM(4)=INT(IDAY/10)
     NUM(5)=IDAY-NUM(4)*10+48
     NUM(6)=47
     NUM(7)=INT(IYEAR/10)
     NUM(8)=IYEAR-NUM(7)*10+48
     NUM(9)=32
     NUM(10)=INT(IHOUR/10)
     NUM(11)=IHOUR-NUM(10)*10+48
     NUM(12)=58
     NUM(13)=INT(IMIN/10)
     NUM(14)=IMIN-NUM(13)*10+48
     DO 10 I=1,13,3
       NUM(I)=NUM(I)+48
  10 CONTINUE
     IF (NUM(1).EQ.48) NUM(1)=32
     IF (NUM(10).EQ.48) NUM(10)=32
     DO 20 I=1,14
       DATE(I:I)=CHAR(NUM(I))
  20 CONTINUE
     DO 30 I=15,23
       DATE(I:I)=' '
  30 CONTINUE
     RETURN
     END
```