

ARMY RESEARCH LABORATORY



Preparation of a Revised Library for the BLAKE Thermodynamic Program

William Oberle
U.S. ARMY RESEARCH LABORATORY

Eli Freedman
ELI FREEDMAN AND ASSOCIATES

ARL-TR-1152

July 1996

19960729 028

DTIC QUALITY INSPECTED 2

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION IS UNLIMITED.

NOTICES

Destroy this report when it is no longer needed. DO NOT return it to the originator.

Additional copies of this report may be obtained from the National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Road, Springfield, VA 22161.

The findings of this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

The use of trade names or manufacturers' names in this report does not constitute indorsement of any commercial product.

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project(0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE July 1996	3. REPORT TYPE AND DATES COVERED Final, Jan 94 - Jun 95	
4. TITLE AND SUBTITLE Preparation of a Revised Library for the BLAKE Thermodynamic Program			5. FUNDING NUMBERS PR: 1L162618A1FL	
6. AUTHOR(S) William Oberle and Eli Freedman				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) U.S. Army Research Laboratory ATTN: AMSRL-WT-PA Aberdeen Proving Ground, MD 21005-5066			8. PERFORMING ORGANIZATION REPORT NUMBER ARL-TR-1152	
9. SPONSORING/MONITORING AGENCY NAMES(S) AND ADDRESS(ES)			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) This report details the modification of the equilibrium thermodynamic code BLAKE to implement a revised and updated library. This revision extends the standard BLAKE library to 6,000 K and results in improved run efficiency. <p style="text-align: right;">DTIC QUALITY INSPECTED 3</p>				
14. SUBJECT TERMS thermochemical, thermodynamic, equilibrium			15. NUMBER OF PAGES 61	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL	

INTENTIONALLY LEFT BLANK.

ACKNOWLEDGMENTS

The authors would like to acknowledge Anthony J. Kotlar for his valuable assistance, especially with the extension of the program to include many more species; and Kevin J. White for his many helpful discussions.

INTENTIONALLY LEFT BLANK.

TABLE OF CONTENTS

	<u>Page</u>
ACKNOWLEDGMENTS	iii
LIST OF TABLES	vii
1. INTRODUCTION	1
2. SOURCES OF DATA	2
3. ASSEMBLING THE PRODUCT DATA	3
4. SUMMARY	9
5. REFERENCES	11
APPENDIX A: LISTING OF THE INPUT FILE GASES	13
APPENDIX B: LISTING OF THE PROGRAM MAKSUBJG.BAS	19
APPENDIX C: LISTING OF THE PROGRAM MAKEJANG.BAS	25
APPENDIX D: LISTING OF THE INPUT FILE CONDENS D	31
APPENDIX E: LISTING OF THE PROGRAM MAKEJANC.BAS	37
APPENDIX F: LISTING OF THE PROGRAM TIGFIT8	43
APPENDIX G: LISTING OF THE PROGRAM ADDBLNK.BAS	57
APPENDIX H: LISTING OF THE DOS BATCH FILE MAKELIBS.BAT	61
DISTRIBUTION LIST	65

INTENTIONALLY LEFT BLANK.

LIST OF TABLES

<u>Table</u>		<u>Page</u>
1.	The Elements Included in the BLAKE Library	3
2.	Listing of the Input File Containing Various, C, H, O, and N Propellants	4
3.	The Different Species in the Output From the Input of Table 2 Sorted in Order of Decreasing Mole Fraction	5
4.	The Gaseous Species Included in the New BLAKE Library	6
5.	Gaseous Species for Which Thermodynamic Data Are Furnished, but Which Must First Be Processed by a User	7
6.	The Condensed Species Included in the BLAKE Library	7

INTENTIONALLY LEFT BLANK.

1. INTRODUCTION

The BLAKE computer program is a general equilibrium thermodynamics program derived from an older version of TIGER.* Although it applies to a wide range of chemical equilibrium calculations, BLAKE is specifically intended for computing the thermodynamic properties of gun propellants combustion products at chamber conditions. Its principal difference from other chemical equilibrium computer programs is that, like TIGER, it permits the use of several nonideal gaseous equations of state.

BLAKE was formally documented in a U.S. Army Ballistic Research Laboratory (BRL)[†] report (Freedman 1982). Since that time many undocumented changes and additions have been made to the code. Several years ago, the U.S. Army Research Laboratory (ARL), through Task Order 3 on Contract DAAA15-92-D-0001, requested that (1) the program's library be updated and (2) an updated *Guide* be prepared for the program. The objective of updating the library was: (1) to extend the data temperature range to 6,000 K, (2) verify and correct, based upon the latest available data, the Lennard-Jones parameters, and (3) streamline the library to facilitate improved computation/run times. The *Guide* has been completed and is now being reviewed by ARL personnel.

The present report documents the details of the work accomplished in achieving the first-listed goal. The methods adopted are straightforward applications of well-known techniques and applications. The primary purpose of this report is to furnish a "paper trail" so that any questions arising about the new library can be readily answered.

The principal changes in the program since the publication of the earlier *Guide* are as follows:

- The number of possible gaseous products has been enlarged from 29 to 55.
- The program's default input units have been changed from atmospheres and calories to megapascals and Joules for pressure and energy respectively.

* BLAKE is based on the original version of TIGER that was developed by Stanford Research International under Contract No. DA-04-200-AMC-3226(X) with the (former) U.S. Army Ballistic Research Laboratories. Dr. Stanley M. Taylor was the Contracting Officer's Technical Representative. This work was documented in an SRI report by Wibenson, W. E., Jr., Zwisler, W. H., Seely, L. B., and Brinkley, S. R., Jr., "Tiger Computer Program Documentation," 1968.

[†] On 30 September 1992, the U.S. Army Ballistic Research Laboratory (BRL) was deactivated and subsequently became part of the U.S. Army Research Laboratory (ARL) on 1 October 1992.

- Instructions may now be entered in either upper or lower case.
- The computation of the heat of explosion has been corrected and improved.
- The program now computes the mean formula weight of the input composition and its corresponding molar enthalpy.

This report is not completely self-contained. The discussion in some places assumes familiarity with the latest version of the BLAKE program and its documentation (Freedman 1995).

2. SOURCES OF DATA

The primary and unchallenged source of thermodynamic data of product species is the JANAF Tables (Chase 1985), which contain data for approximately 600 species. Of the 88 species chosen for the new library (Table 4), these tables contain complete data, e.g., c_p , from 298.15 K to 6,000 K for 80 of the species; there are either partial or no data for the remaining 8. Data for these eight species were obtained by back calculation from the coefficients supplied by NASA's Lewis Research Center (McBride, Reno, and Gordon 1994) for the 1989 version of the CEC thermodynamics program.

The necessary Lennard-Jones parameters (viz., the diameter and the well depth) were taken from Reid, Prausnitz, and Poling (1987), who cite Svehla and McBride (1973) as their source. These values were obtained by fitting viscosity data, not P-V-T data, and are open to question; unfortunately there is no better source.

The primary source of enthalpy data for the stored organic species was the compilation edited by Pedley and Rylance (1977). The primary source for enthalpies of inorganic species was the NBS Tables (Wagman *et al.* 1982). Data for propellant ingredients not contained in either of these two sources were taken from the ICT compilations (Volk, Bathelt, and Kuthe 1971; Bathelt and Volk 1981.)

Data for the formulas and enthalpies of formation of some widely used propellant ingredients are stored within the program. Formulas and enthalpies of other ingredients are stored in the file named INGRED.DAT contained on the program disk. Users can insert their own data into this file if they desire.

3. ASSEMBLING THE PRODUCT DATA

Owing to its inherent limitations, BLAKE cannot handle, in any one computation, more than 55 gaseous product species or 3 condensed-phase product species. This means that some scheme must be devised for selecting the species to be included. Fortunately, the NASA-Lewis thermodynamics programs CEC and its descendants can readily handle 600 gaseous species and 200 condensed ones. Therefore, the approach used was straightforward. The products computed by the NASA program for a selected composition were sorted in order of decreasing mole fraction of products.

A review of the elements appearing in various military propellants, including electrothermal-combustion compositions, led to the 21 elements listed in Table 1 being chosen for inclusion in the new library.

Table 1. The Elements Included in the BLAKE Library

Aluminum	Chlorine	Hydrogen	Magnesium	Sodium
Argon	Electron	Iron	Nitrogen	Sulfur
Barium	Fluorine	Lead	Oxygen	Titanium
Boron	Helium	Lithium	Potassium	Zirconium
Carbon				

- (1) The elements C, H, O, and N were considered first and as a group because they are the four most important elements in all propellants. Table 2 shows the input file containing some typical C, H, O, and N propellants that were used.

Table 2. Listing of the Input File Containing Various C, H, O, and N Propellants

FOR, CAB, -2.9900E+06, C, 147, H, 237, O, 77
FOR, ATEC, -4E5, C, 14, H, 22, O, 8
FOR, CEL, -230300., C, 6, H, 10, O, 5
FOR, PU, -8313000., C, 512, H, 958, N, 14, O, 159
COM, RDX, 72.38, NC1260, 3.8095, CAB, 11.43, ATEC, 7.238, EC, 0.381,
GUN, 0.2, 10, 0
TIT, A Triple Base Propellant
FOR, NC, -0.16591E9, C, 6000, H, 7416, O, 10168, N, 2584
FOR, DEGDN, -1.035E5, C, 4, H, 8, O, 7, N, 2
FOR, NG, -0.886E5, C, 3, H, 5, O, 9, N, 3
FOR, AKAR, -0.255E5, C, 14, H, 14, O, 1, N, 2
COM, AKAR, 0.8, NG, 14.9, DEGDN, 24.8, NC, 59.5
GUN, 0.2, 10, 0
TIT, A Lova Composition
FOR, CAB, -1.3906E8, C, 4529, H, 7009, O, 2358
FOR, NC, -1.6916E8, C, 6000, H, 7549, O, 9901, N, 2451
FOR, EC, -2.51E4, C, 17, H, 20, O, 1, N, 2
FOR, RDX, 1.469E4, C, 3, H, 6, O, 6, N, 6
FOR, BDNPA, -1.533E5, C, 8, H, 14, O, 10, N, 4
FOR, BDNPF, -1.483E5, C, 7, H, 12, O, 10, N, 4
COM, CAB, 12, NC, 4, EC, 0.4, BDNPA, 3.8, BDNPF, 3.8, RDX, 76
GUN, 0.2, 10, 0
FOR, OC501B, -0.7638E8, C, 2934, H, 4731, O, 1537
TITLE, Another Propellant
CM2, NC1315, 53.74, NG, 20.58, DEGDN, 18.74, OC501B, 5.41,
DPA, 1.12, C, 0.2, DBP, 0.11, H2O, 0.1
GUN, 0.2, 10, 0
STOP

The relevant output was sorted and led to Table 3.

(2) Similar calculations for all of the other elements led to the list of species in Table 4.

Table 3. The Different Species in the Output From the Input of Table 2
Sorted in Order of Decreasing Mole Fraction

1. CO	0.4106000000	20. NH	0.0000077510
2. H2O	0.2668000000	21. ACETYLENE	0.0000061440
3. H2	0.2559000000	22. HNO	0.0000041300
4. N2	0.2349000000	23. KETENE	0.0000032850
5. CO2	0.1291000000	24. H2O2	0.0000030450
6. OH	0.0037710000	25. HO2	0.0000024270
7. H	0.0030730000	26. METHANOL	0.0000014010
8. NH3	0.0011400000	27. ETHYLENE	0.0000010160
9. HCN	0.0007914000	28. N	0.0000010140
10. NO	0.0005504000	29. CN	0.0000007575
11. CH4	0.0003968000	30. N2O	0.0000006096
12. HCO RAD	0.0001392000	31. NCO	0.0000005651
13. FORMALDEHYDE	0.0001243000	32. HNO2	0.0000003488
14. FORMIC ACID	0.0000876600	33. CH2	0.0000002635
15. O2	0.0000745200	34. HYDROXYMETHYLEN	0.0000002620
16. O	0.0000691900	35. NO2	0.0000002260
17. HNCO	0.0000568000	36. METHYL CYANIDE	0.0000002125
18. NH2	0.0000337300	37. NH2OH	0.0000001063
19. CH3	0.0000222100		

Table 4. The Gaseous Species Included in the New BLAKE Library

Al ₂ O	CH ₂	FeCl ₂	K ₂	NH ₃
AlF ₂ O	CH ₂ CO	FeO ₂ H ₂	K ₂ H ₂ O ₂	NO
AlHO ₂	CH ₂ O	HCOOH	KH	NO ₂
AlO ₂	CH ₂ OH	H ₂	KO	NS
Ar	CH ₃	H ₂ O	KOH	O
B ₂ O ₂	CH ₃ Cl	H ₂ O ₂	LiOH	O ₂
B ₂ O ₃	CH ₃ CN	H ₂ S	LiBO ₂	OH
B ₃ H ₃ O ₃	CH ₄	HAIO	MgO ₂ H ₂	Pb
BaO ₂ H ₂	CH ₃ OH	HCl	N	PbS
BH ₂	CHO	HCN	N ₂	S
BH ₃	Cl	Helium	N ₂ O	S ₂
BHO	CN	HF	NAF	S ₂ O
BHO ₂	CO	HNCO	NaOH	SH
BO	CO ₂	HNO	NCO	SO
C	COS	HNO ₂	NH	SO ₂
C ₂ H ₂	CS	HO ₂	NH ₂	TiO ₂
C ₂ H ₃ N	CS ₂	K	NH ₂ OH	ZrO
C ₂ H ₄	Electron	KBO ₂		

- (3) Of the 88 species listed in Table 4, 38 contain at most C, H, O, and N; the remaining 50 contain one or more of the other 17 elements. Most of these 17 elements occur in only a few compositions, but potassium and sulfur, which are in the flash reducer potassium sulfate and igniter formulations of black powder, are frequently found. These six elements (C, H, N, O, S, and K) are found in a total of 55 of the compounds listed in Table 4, which means that any such composition can be immediately run by the present BLAKE without the use of any 'Reject' instructions.
- (4) Exploratory calculations showed that the concentrations of the seven species listed in Table 5 are almost always too low to be of any interest; their presence contributes nothing to the computed thermodynamic properties.

Table 5. Gaseous Species for Which Thermodynamic Data Are Furnished, but Which Must First Be Processed by a User

Hydroxymethylene	Hydrogen peroxide	Hydroperoxy
Hydroxylamine	Ketene	Methanol
Methyl cyanide		

To take care of the exceptional cases, however, it was decided to include thermodynamic data for these seven species, but to displace their CONStituent instructions three columns to the right in the alphanumeric library. This has the effect of keeping these species out of the binary library, but in such a way that a user can easily restore them with any ASCII editor. See the revised *Guide* (Freedman 1995) for details.

- (5) Of the 88 species listed in Table 4, the JANAF Tables contain thermodynamic data from 298.15 K to 6,000 K for 80 of them. The eight exceptions are formic acid, hydrogen peroxide, hydroperoxy, hydroxylamine, hydroxymethylene, ketene, methyl cyanide, and methanol, which either are not included in the JANAF Tables, or for which the JANAF data do not extend to 5,000 K. (Note that seven of these eight species are the ones listed in Table 5.) Thermodynamic data for these eight species were obtained by back-calculation from the data included in the thermodynamic tables that accompany the NASA-Lewis thermodynamics program, CET89.
- (6) The program library contains data for the 18 condensed-phase species listed in Table 6.

Table 6. The Condensed Species Included in the BLAKE Library

Species	Phase(s)	Species	Phase(s)
Al ₂ O ₃	solid and liquid	K ₂ S	solid and liquid
B	solid and liquid	Li ₂ O	solid and liquid
B ₂ O ₃	solid and liquid	LiOH	solid and liquid
BC ₄	solid only	MgO	solid and liquid
BN	solid only	NaF	liquid only
C(s)	solid only	Pb	solid and liquid
KF	liquid only	Ti ₄ O ₇	liquid only
K ₂ CO ₃	solid and liquid	TiO ₂	solid only
KOH	liquid only	ZrO ₂	solid and liquid

All told, the new library will contain 106 species, down from the 153 species contained in older libraries. The 47 deleted species are almost never encountered in gun propellant calculations or their mole fraction is sufficiently small so as to have negligible impact on calculated results. Deleting these species results not only in reduced computation/run times, but also significantly reduces the use of the 'Reject' instruction.

The next step was to extract their thermodynamic data from a tape of the JANAF Tables. Doing this was a tedious task prone to errors and was therefore automated.

First the gaseous species were placed in a file named GASES. For each species, this file contained its names, its JANAF JCODE number (an arbitrary reference number found only on the JANAF tape), its formula, and all of the information that was to be incorporated into the STG entry for that species. The order of species in this file was the order in which it was intended that they appear in the final library. The entire file is listed in Appendix A.

File GASES was the input for a QuickBASIC program named MAKESUBJG, which is listed in Appendix B. This program formed a new file in which the JANAF data for 80 species was listed completely in the order in which these species appear in the original JANAF tape. A second QuickBASIC program named MAKEJANG, listed in Appendix C, rearranged these data into the order in which they appear in GASES and placed them into a file named JAN_FILE.GAS.

Similarly, the condensed species were placed into a file named CONDENS, which is listed in Appendix D. A QuickBASIC program named MAKEJANC, listed in Appendix E, extracted the appropriate data from the JANAF tape and placed them into a file named JAN_FILE.CON.

The two files JAN_FILE.GAS and JAN_FILE.CON were the inputs for the FORTRAN fitting program TIGFIT8; this program is listed in Appendix F. TIGFIT8 always places its output in TIGFIT.OUT; the two output files were temporarily copied to GASES.OUT and CONDEN.OUT, where they were kept for later use.

The NASA coefficients for the eight species, formic acid, hydrogen peroxide, hydroperoxy, ketene, methyl cyanide, methanol, hydroxymethylene, and hydroxylamine, were extracted by hand (using an ASCII editor) from the library file FILE4.DAT that accompanied CET89, and were stored in CUT.LIS.

This file, in turn, became the input for the FORTRAN program CPHSJN, which expanded the coefficients and formed JANAF-like tables for these species in the temperature range 298.15 K to 5,000 K; these tables were stored in TIGFITJ.IN.

The CON instructions for seven of these eight species have three blanks placed in front of them. This was accomplished by the QuickBASIC program ADDBLNK, which is listed in Appendix G.

The previous fitting program TIGFIT8 operated on TIGFITJ.IN and formed another file named TIGFIT8.OUT. This file was then merged with GASES.OUT; the resultant file was also named GASES.OUT.

Finally, the two files GASES.OUT and CONDEN.OUT were merged, and an appropriate header was placed in front of them. This final file was named NEWLIB.LIB; it was the alphanumeric source file that contained the final library for BLAKE. The program FORMLIB made the binary file that is used by BLAKE.

After visual inspection and testing, NEWLIB.LIB was renamed SBLAKLYB.LIB and became the new library for the latest version of BLAKE. The program FORMLIB made the binary file that is used by BLAKE.

The overall process was executed by a DOS batch file named MAKELIBS, which is listed in Appendix H.

4. SUMMARY

The BLAKE program has been revised. The most significant improvement is the expansion of the program to handle 55 gaseous species and 3 condensed phase species in any one calculation. Twenty-one elements were selected for inclusion in the program's new library. An intensive survey was carried out to determine the most significant species formed by these 21 elements for inclusion in the library; 88 species were selected. Thermodynamic data for these 88 species were taken from either the JANAF Tables or back-calculated from fittings furnished by NASA's Lewis Research Center. The coefficients of the fitted (or re-fitted) data were assembled in a new library. The enthalpies of formation of the prestored propellant ingredients were reviewed and revised where necessary.

INTENTIONALLY LEFT BLANK.

5. REFERENCES

- Bathelt, H., and F. Volk. "Thermodynamische Daten von Raketentreibstoffen, Treibladungspulvern und Sprengstoffen sowie deren Komponenten. I. Ergänzungsband." Institut fuer Chemie der Treib- und Explosivstoffe (now called the Institut fuer Chemische Technologie), Berghausen/Karlsruhe, 1981.
- Chase, M. JANAF Thermochemical Tables, 3rd edn., ACS/AIP (1986); also, Supplement No. 1 to the Journal of Physical and Chemical Reference Data, vol. 14, 1985.
- Freedman, Eli. "BLAKE—A Thermodynamics Code Based on TIGER: Users' Guide to the Revised Program." ARL-CR-xxxx, U.S. Army Research Laboratory, Aberdeen Proving Ground, MD, 1995 (in press).
- Freedman, Eli. "Blake—A Thermodynamics Code Based on Tiger—Users' Guide and Manual." ARBRL-TR-02411, U.S. Army Ballistic Research Laboratory, Aberdeen Proving Ground, MD, 1982 [AD A121 259].
- McBride, B. J., M. A. Reno, and S. Gordon. "CET93 and CETPC: An Interim Updated Version of the NASA Lewis Computer Program for Calculating Complex Chemical Equilibria with Applications." NASA TM 4557, National Aeronautics and Space Administration, Lewis Research Center, Cleveland, OH, 1994.
- Pedley, J. B., and J. Rylance. "Sussex-N.P.L. Computer Analyzed Thermochemical Data: Organic and Organometallic Compounds." University of Sussex, Sussex, 1977.
- Reid, R. C., J. M. Prausnitz, and B. E. Poling. The Properties of Liquids and Gases. 4th ed., McGraw-Hill, New York, 1987.
- Svehla, R. A., and B. J. McBride. "Fortran IV Computer Program for Calculation of Thermodynamic and Transport Properties of Complex Chemical Systems." NASA TD 7056, National Aeronautics and Space Administration, Lewis Research Center, Cleveland, OH, 1973.
- Volk, F., H. Bathelt, and R. Kuthe. "Thermodynamische Daten von Raketentreibstoffen, Treibladungspulvern und Sprengstoffen sowie deren Komponenten." Institut fuer Chemie der Treib- und Explosivstoffe (now called the Institut fuer Chemische Technologie), Berghausen/Karlsruhe, 1971.
- Wagman, D. D., V. B. Parker, R. H. Schumm, I. Halow, S. M. Bailey, K. L. Churney, and R. L. Nuttal. "The NBS Tables of Chemical Thermodynamic Data." J. Phys. Chem., Ref. Data 11, Supplement 2, 1982.

INTENTIONALLY LEFT BLANK.

APPENDIX A:
LISTING OF THE INPUT FILE GASES

INTENTIONALLY LEFT BLANK.

1, 2, "CO" , 153, "C", 1, "O", 1,
 "STG, CO, GAS, 390, D, 3.69, T, 91.7", ""

2, 2, "H2O" , 309, "H", 2, "O", 1,
 "STG, H2O, GAS, 250, T, 542.5, D, 2.79", ""

3, 1, "H2" , 304, "H", 2,
 "STG, H2, GAS, 180, D, 2.827, T, 59.7", ""

4, 1, "N2" , 360, "N", 2,
 "STG, N2, GAS, 148, D, 3.798, T, 71.4", ""

5, 2, "CO2" , 155, "C", 1, "O", 2,
 "STG, CO2, GAS, 600, D, 3.941, T, 195.2", ""

6, 2, "OH" , 300, "O", 1, "H", 1,
 "STG, OH, GAS, 226", ""

7, 1, "H" , 288, "H", 1,
 "STG, H, GAS, 13.4", ""

8, 2, "NH3" , 314, "N", 1, "H", 3,
 "STG, NH3, GAS, 476., D, 2.9, T, 558.3", ""

9, 3, "HCN" , 132, "H", 1, "C", 1, "N", 1,
 "STG, HCN, GAS, 359, D, 3.339, T, 344.7", ""

10, 2, "NO" , 351, "N", 1, "O", 1,
 "STG, NO, GAS, 386., D, 3.492, T, 116.7", "*"

11, 2, "CH4" , 145, "C", 1, "H", 4,
 "STG, CH4, GAS, 528, D, 3.758, T, 148.6", ""

12, 3, "CHO" , 134, "C", 1, "H", 1, "O", 1,
 "STG, CHO, GAS, 699.987", "*"

13, 3, "CH2O" , 139, "C", 1, "H", 2, "O", 1, "", "*"

14, 3, "CH2O2" , -1, "C", 1, "H", 2, "O", 2, "", "*"

15, 1, "O2" , 389, "O", 2,
 "STG, O2, GAS, 350, D, 3.467, T, 106.7", ""

16, 1, "O" , 381, "O", 1,
 "STG, O, GAS, 212.8", ""

17, 4, "HNCO" , 133, "H", 1, "N", 1, "C", 1, "O", 1, "", "*"

18, 2, "NH2" , 308, "N", 1, "H", 2, "", "*"

19, 2, "CH3" , 140, "C", 1, "H", 3,
 "STG, CH3, GAS, 525", "*"

20, 2, "NH" , 295, "N", 1, "H", 1, "", "*"

21, 2, "C2H2" , 165, "C", 2, "H", 2, "", "*"

22, 3, "HNO" , 767, "H", 1, "N", 1, "O", 1, "", "*"

23, 2, "H2O2" , -1, "H", 2, "O", 2, "", "*"

24, 2, "HO2" , 810, "H", 1, "O", 2, "", "*"

25, 2, "C2H4" , 166, "C", 2, "H", 4,
 "STG, C2H4, GAS, 372, D, 4.163, T, 224.7", "*"

26, 1, "N" , 350, "N", 1,
 "STG, N, GAS, 148", "*"

27, 2, "CN" , 147, "C", 1, "N", 1, "", "*"

28, 2, "N2O" , 361, "N", 2, "O", 1, "", "*"

29, 3, "NCO" , 1109, "N", 1, "C", 1, "O", 1, "", "*"

30, 3, "HNO2" , 783, "H", 1, "N", 1, "O", 2, "", "*"

31, 2, "CH2" , 135, "C", 1, "H", 2, "", "*"


```

"STG, CL, GAS, 250", ""
77, 3, "CH3CL" , 141, "C", 1, "H", 3, "CL", 1, "", "*"
78, 2, "TIO2" , 398, "TI", 1, "O", 2, "", ""
79, 2, "ZRO" , 388, "ZR", 1, "O", 1, "", ""
80, 2, "FECL2" , 851, "FE", 1, "CL", 2, "", ""
81, 3, "FEO2H2" , 917, "FE", 1, "O", 2, "H", 2, "", ""
82, 3, "LiOH" , 292, "LI", 1, "O", 1, "H", 1, "", ""
83, 3, "LIBO2" , 653, "LI", 1, "B", 1, "O", 2, "", ""
84, 1, "ARGON" , 1353, "AR", 1,
"STG, ARGON, GAS, 532, D, 3.542, T, 93.3", ""
85, 1, "HELIUM", 1359, "HE", 1,
"STG, HELIUM, GAS, 0.0, D, 2.551, T, 10.22", ""
86, 3, "CH4O" , -1, "C", 1, "H", 4, "O", 1, "", "*"
87, 3, "CH3O" , -1, "C", 1, "H", 3, "O", 1, "", "*"
88, 3, "NH2OH" , -1, "N", 1, "H", 3, "O", 1, "", "*"
89, 1, "ELEC" , 831, "EL", 1, "", ""
0, 0, "END" , -1, "", ""

```

INTENTIONALLY LEFT BLANK.

APPENDIX B:
LISTING OF THE PROGRAM MAKSUBJG.BAS

INTENTIONALLY LEFT BLANK.

In this and all of the following QuickBASIC listings, the underscore '_' is used to indicate a break in lines that are too long to fit on the page. The QuickBASIC compiler accepts this symbol, but displays such lines on the screen without the break.

In order to minimize the introduction of errors, file names have been left unchanged.

```
' This program creates a mini-JANAF tape file that is restricted to  
' data on the *gaseous* species that are to be included in the revised  
' BLAKE.
```

```
' Written by EF&A in March/94  
' Saved as MAKSUBJG.BAS [This version: 12 November 1994]
```

```
WIDTH 80, 43
```

```
DEFINT I-N: DEFSTR S: begin = TIMER
```

```
DATA 298.15
```

```
countt = 0: lines& = 0: kk = 0
```

```
READ tstart
```

```
OPEN "c:\janaf\subsetg.lib" FOR OUTPUT AS #3
```

```
OPEN "jan_file.prt" FOR OUTPUT AS #4
```

```
PRINT #4, " Num.Species    KK        JCODE        Species"
```

```
OPEN "c:\janaf\tables" FOR INPUT AS #2 LEN = 32767
```

```
mode = 2
```

```
get.jan.line:
```

```
LINE INPUT #2, s.line
```

```
    IF (EOF(2) = -1) THEN GOTO endd ELSE lines& = lines& + 1
```

```
    IF (MID$(s.line, 3, 7) <> "JCODE =") THEN GOTO get.jan.line
```

```
    j.tab.code = VAL(MID$(s.line, 10, 6))
```

```
    ,                               PRINT " JCODE (from table) = "; j.tab.code
```

```
jcode.sel:
```

```
CLOSE #1: OPEN "gases" FOR INPUT AS #1: num.species = 0
```

```
get.a.species:
```

```
    INPUT #1, idiot.no, num.at, species, jcode
```

```
    ' 'idiot.num' is a running number that counts the number of  
' species.
```

```
    ' 'num.at' is the number of atoms in the species
```

```
    ' If 'num.at' is greater than 10, it means that that species is  
' one that will have its 'CON' instruction displaced by 3
```

```
    ' columns in the alphanumeric library.
```

```
    if (num.at > 10) then num.at = num.at - 10
```

```
    ,                               PRINT "<Check1: >"; idiot.no, num.at; species; jcode
```

```
    ,                               PRINT "num at., etc "; _  
                                     num.at, species, jcode
```

```

IF (species = "E N D" OR jcode = 0) THEN GOTO get.jan.line
num.species = num.species + 1
IF (j.tab.code <> jcode) THEN
    FOR ii = 1 TO num.at: INPUT #1, s.junk1, junk2
        PRINT s.junk1, junk2;
    NEXT: ' PRINT

    INPUT #1, stg, asterisk$
    GOTO get.a.species
END IF
SOUND 440, 5: kk = kk + 1
PRINT kk, num.species; jcode; species; " ";
PRINT #4, USING _
    "    ###      ###      ####      &"; _
    num.species; kk; jcode; species
PRINT #3, USING "\ \ GAS"; species;
PRINT #3, USING "          JCODE =####"; jcode
PRINT USING " JCODE =####"; jcode

```

```

next.line.1:
    LINE INPUT #2, s.line: lines& = lines& + 1
    PRINT "> "; LEFT$(s.line, 10)
    test.val = VAL(LEFT$(s.line, 10))
    IF (test.val < .995 * tstart) THEN GOTO next.line.1
    IF (test.val = tstart OR test.val < 1.005 * tstart)_
        THEN GOTO start.tab
    PRINT "Can't find T = TSTART for species "; species
    STOP

```

```

start.tab:
    PRINT #3, s.line: ' PRINT s.line

```

```

nexxt:
    LINE INPUT #2, s.line: lines& = lines& + 1
    PRINT "> "; LEFT$(s.line, 30)
    IF (LEFT$(s.line, 12) = " PREVIOUS:") THEN
        mode = 0
        PRINT #3, " -1 "
        GOTO get.jan.line
    ELSE
        PRINT #3, s.line
    END IF
    IF (VAL(LEFT$(s.line, 9)) = 6000!) THEN
        mode = 0
        GOTO get.jan.line
    ELSE
        GOTO nexxt
    END IF

```

```

trouble:

```

```

PRINT " P O T E N T I A L   T R O U B L E .
PRINT "Can't find T = 6000 K for species "; species
PRINT s.line: STOP

endd:
all.done = TIMER: e.time = all.done - begin
                                PRINT #4, : PRINT #4,
PRINT #4, "TOTAL No. of lines read = "; lines&
PRINT #4, "Program took "; e.time; "seconds"
e.mins = INT(e.time / 60): e.secs = e.time - 60! * e.mins
e.secs = (INT(10! * e.secs + .5)) / 10!
PRINT #4, "          = "; e.mins; " mins, "; e.secs; " secs"
FOR i = 1 TO 5: SOUND 250, 10: SOUND 500, 5: NEXT
SYSTEM

make.con.instr:

RETURN: ' Temporary--for code checking

FOR i = 1 TO num.at: INPUT #1, symb(i), at.no$(i): NEXT

FOR i = 1 TO num.at
  IF (ASC(symb(i)) > 64 AND ASC(symb(i)) < 91) _
                                THEN GOTO nexxt1
err1:
  PRINT "Error in CON info for species "; species
  PRINT "CHEK: "; symb(i); " "; at.no$(i); ASC(symb(i));_
                                " "; ASC(at.no$(i))
  STOP
nexxt1:
  IF (ASC(at.no$(i)) > 47 AND ASC(at.no$(i)) < 58) _
                                then GOTO nexxt2
  GOTO err1
nexxt2:
  NEXT

PRINT #3, "CON,"; species; ","; "GAS,";
  FOR i = 1 TO num.at: PRINT #3, symb(i); ","; at.no$(i); ",";
  NEXT: PRINT #3,
RETURN

```

INTENTIONALLY LEFT BLANK.

APPENDIX C:
LISTING OF THE PROGRAM MAKEJANG.BAS

INTENTIONALLY LEFT BLANK.

```
' This program takes the file SUBSETG.LIB, created by
' MAKSUBJG.BAS (q.v.), and rearranges it into the order of
' decreasing importance for the new BLAKE library as specified
' in the file GASES.
```

```
' Written by EF&A in OCT/94.
```

```
' Saved as MAKEJANG.BAS [11-OCT-94]
```

```
DEFINT I-N: DEFSTR S: COLOR 15, 0: begin = TIMER
DATA 298.15
countt = 0: no.rew = 0: lines& = 0: no.species = 0
READ tstart
```

```
OPEN "gases" FOR INPUT AS #1
```

```
' The file GASES contains the names of the *gaseous* species in
' the approximate order of decreasing concentration in typical
' outputs.
```

```
OPEN "jan_file.gas" FOR OUTPUT AS #3
```

```
OPEN "jan_file.prt" FOR OUTPUT AS #4
```

```
jcode.sel:
```

```
INPUT #1, idiot.no, num.at, species, jcode
  num = num.at: IF (num.at > 10) THEN num = num.at - 10
  IF (species = "E N D" OR jcode = 0) THEN GOTO endd
  IF (jcode < 0) THEN
    FOR i = 1 TO num: INPUT #1, s.junk, junk
    NEXT
    INPUT #1, s.junk, s.junk1
    GOTO jcode.sel
```

```
END IF
```

```
no.species = no.species + 1
GOSUB make.con.instr
```

```
k.rew = 0
```

```
PRINT no.species; " Looking for JCODE = "; jcode; _
      " species = "; species
PRINT #4, "Looking for JCODE = "; jcode; " species = "; _
      species
```

```
PRINT #3, USING "\ \ GAS"; species
```

```
rewindd:
```

```
CLOSE #2: OPEN "c:\janaf\subsetg.lib" _
      FOR INPUT AS #2 LEN = 32767
```

```
no.rew = no.rew + 1
```

```
tab.input:
```

```
LINE INPUT #2, s.line
```

```

IF (EOF(2) = -1) THEN
  SOUND 500, 5: SOUND 250, 10
  k.rew = k.rew + 1
  IF (k.rew > 1) THEN
    PRINT "Too many rewinds for species "; species
    STOP
  END IF
  mode = 1
  GOTO rewindd
END IF
lines& = lines& + 1
IF (MID$(s.line, 43, 7) <> "JCODE =") THEN
  GOTO tab.input
END IF

num = VAL(MID$(s.line, 50, 5))
IF (num <> jcode) THEN
  GOTO tab.input
END IF
PRINT "> "; LEFT$(s.line, 40): PRINT : lines& = lines& + 1
PRINT #4, "> "; LEFT$(s.line, 40): PRINT #4,
  mode = 0: k.rew = 0
next.line.1:
  LINE INPUT #2, s.line: lines& = lines& + 1

  test.no = VAL(LEFT$(s.line, 10))
  IF (test.no < tstart) THEN GOTO next.line.1
  IF (test.no < 1.005 * tstart) THEN GOTO start.tab
  PRINT "Can't find T = "; tstart; "for species "; species: STOP

start.tab:
  PRINT #3, s.line

next.line.1.5:
  LINE INPUT #2, s.line
  IF (VAL(LEFT$(s.line, 9)) = -1) THEN
    mode = 0
    PRINT #3, "9999., 999. 99., 9., .9, 999. 99. 9."
    GOTO jcode.sel
  ELSEIF (VAL(LEFT$(s.line, 9)) < 6000!) THEN
    PRINT #3, s.line
    GOTO next.line.1.5
  ELSEIF (VAL(LEFT$(s.line, 9)) = 6000!) THEN
    PRINT #3, s.line
    mode = 0
    GOTO jcode.sel
  END IF
next.line.2:

```

```

IF (VAL(LEFT$(s.line, 9)) > 6000!) THEN GOTO trouble
  LINE INPUT #2, s.line: PRINT #3, s.line
  IF (VAL(LEFT$(s.line, 9)) < 6000!) THEN
    GOTO next.line.2
  ELSE
    GOTO jcode.se1
END IF

```

trouble:

```

  PRINT " P O T E N T I A L   T R O U B L E .
  print "Can't find T = 6000 K for species "; species
  PRINT s.line: STOP

```

endd:

```

all.done = TIMER: e.time = all.done - begin
                                'PRINT #3, : PRINT #3,
PRINT "No. of rewinds = "; no.rew:
e.mins = INT(e.time / 60): e.secs = e.time - 60! * e.mins
e.secs = (INT(10! * e.secs + .5)) / 10!
'PRINT #3, "          = "; e.mins; " mins, "; e.secs; " secs"
PRINT #3, "E N D": PRINT #3, "E N D": CLOSE #3

```

```

PRINT #4, : PRINT #4,
PRINT #4, "No. of rewinds = "; no.rew:
PRINT #4, "TOTAL No. of lines read = "; lines&
PRINT #4, "Program took "; e.time; "seconds"
PRINT #4, "          = "; e.mins; " mins, "; e.secs; " secs"

```

```

FOR i = 1 TO 5: SOUND 250, 10: SOUND 500, 5: NEXT
SYSTEM

```

make.con.instr:

```

  FOR i = 1 TO num: INPUT #1, symb(i), at.no$(i): NEXT
                    INPUT #1, stg$, asterisk$

```

```

  FOR i = 1 TO num
    IF (ASC(symb(i)) > 64 AND ASC(symb(i)) < 91) THEN GOTO nexxt1
err1:
    PRINT "Error in CON info for species "; species
    PRINT "CHEK: "; symb(i); " "; at.no$(i); _
                    ASC(symb(i)); " "; ASC(at.no$(i))
    STOP

```

nexxt1:

```

  IF (ASC(at.no$(i)) > 47 AND ASC(at.no$(i)) < 58) GOTO nexxt2
  GOTO err1

```

nexxt2:

```

  NEXT

```

```

IF (num.at < 10) THEN

```

```
        PRINT #3, "CON,"; species; ",,"; "GAS,";
ELSE
        PRINT #3, "  CON,"; species; ",,"; "GAS,";
    END IF
FOR i = 1 TO num: PRINT #3, symb(i); ",,"; at.no$(i); ",,"; : NEXT
PRINT #3, asterisk$: PRINT #3, stg$
RETURN
```

APPENDIX D:
LISTING OF THE INPUT FILE CONDENS

INTENTIONALLY LEFT BLANK.

1, 2, "AL203\$", 33, "SOLID", "AL", 2, "O", 3
 "STC, AL203\$, SOLID, 1, 28.3, 0., 0"
 "STC, AL203\$, SOLID, 2, 0., 0., 0"
 "STC, AL203\$, SOLID, 3, 0., 0., 0"
 2, 2, "AL203\$", 34, "LIQUID", "AL", 2, "O", 3
 "STC, AL203\$, LIQUID, 1, 33.9, 0., 0"
 "STC, AL203\$, LIQUID, 2, 0., 0., 0"
 "STC, AL203\$, LIQUID, 3, 0., 0., 0"
 3, 2, "NAF\$", 247, "LIQUID", "NA", 1, "F", 1
 "STC, NAF\$, LIQUID, 1, 18, 0, 0"
 "STC, NAF\$, LIQUID, 2, 0, 0, 0"
 "STC, NAF\$, LIQUID, 3, 0, 0, 0"
 4, 2, "KF\$", 535, "LIQUID", "K", 1, "F", 1
 "STC, KF\$, LIQUID, 1, 26, 0, 0"
 "STC, KF\$, LIQUID, 2, 0, 0, 0"
 "STC, KF\$, LIQUID, 3, 0, 0, 0"
 5, 2, "BN\$", 60, "SOLID", "B", 1, "N", 1
 "STC, BN\$, SOLID, 1, 11.03, 0, 0"
 "STC, BN\$, SOLID, 2, 0., 0., 0"
 "STC, BN\$, SOLID, 3, 0., 0., 0"
 6, 2, "B203\$", 70, "SOLID", "B", 2, "O", 3
 "STC, B203\$, SOLID, 1, 28., 0., 0"
 "STC, B203\$, SOLID, 2, 0., 0., 0"
 "STC, B203\$, SOLID, 3, 0., 0., 0"
 7, 2, "B203\$", 71, "LIQUID", "B", 2, "O", 3
 "STC, B203\$, LIQUID, 1, 31., 0., 0"
 "STC, B203\$, LIQUID, 2, 0., 0., 0"
 "STC, B203\$, LIQUID, 3, 0., 0., 0"
 8, 1, "C(S)", 111, "SOLID", "C", 1
 "STC, C(S), SOLID, 1, 4.99259, 3.9628E-5, 1.191359E-9"
 "STC, C(S), SOLID, 2, -6.377527E-6, 1.1924995E-10, -3.7557816E-15"
 "STC, C(S), SOLID, 3, 3.58287E-12, -1.00976E-16, 0"
 9, 2, "BC4\$", 78, "SOLID", "B", 1, "C", 4
 "STC, BC4\$, SOLID, 1, 21.9, 0, 0"
 "STC, BC4\$, SOLID, 2, 0., 0., 0"
 "STC, BC4\$, SOLID, 3, 0., 0., 0"
 10, 1, "B\$", 38, "SOLID", "B", 1
 "STC, B\$, SOLID, 1, 21.9, 0, 0"
 "STC, B\$, SOLID, 2, 0., 0., 0"
 "STC, B\$, SOLID, 3, 0., 0., 0"
 11, 1, "B\$", 39, "LIQUID", "B", 1
 "STC, B\$, LIQUID, 1, 24., 0, 0"
 "STC, B\$, LIQUID, 2, 0., 0., 0"
 "STC, B\$, LIQUID, 3, 0., 0., 0"
 12, 3, "KOH\$", 541, "LIQUID", "K", 1, "O", 1, "H", 1
 "STC, KOH\$, LIQUID, 1, 28., 7.3E-3, 0"
 "STC, KOH\$, LIQUID, 2, 0., 0., 0"
 "STC, KOH\$, LIQUID, 3, 0., 0., 0"

13, 3, "K2CO3\$", 526, "SOLID", "K", 2, "C", 1, "O", 3
 "STC, K2CO3\$, SOLID, 1, 60.4, 0, 0"
 "STC, K2CO3\$, SOLID, 2, 0., 0., 0"
 "STC, K2CO3\$, SOLID, 3, 0., 0., 0"
 14, 3, "K2CO3\$", 527, "LIQUID", "K", 2, "C", 1, "O", 3
 "STC, K2CO3\$, LIQUID, 1, 72, .0175, 0"
 "STC, K2CO3\$, LIQUID, 2, 0., 0., 0"
 "STC, K2CO3\$, LIQUID, 3, 0., 0., 0"
 15, 2, "K2S\$", 1452, "SOLID", "K", 2, "S", 1
 "STC, K2S\$, SOLID, 1, 45., 0, 0"
 "STC, K2S\$, SOLID, 2, 0, 0, 0"
 "STC, K2S\$, SOLID, 3, 0, 0, 0"
 16, 2, "K2S\$", 1453, "LIQUID", "K", 2, "S", 1
 "STC, K2S\$, LIQUID, 1, 50, .005, 0"
 "STC, K2S\$, LIQUID, 2, 0, 0, 0"
 "STC, K2S\$, LIQUID, 3, 0, 0, 0"
 17, 1, "PB\$", 645, "SOLID", "PB", 1
 "STC, PB\$, SOLID, 1, 18.27, 0., 0"
 "STC, PB\$, SOLID, 2, 0., 0., 0"
 "STC, PB\$, SOLID, 3, 0., 0., 0"
 18, 1, "PB\$", 646, "LIQUID", "PB", 1
 "STC, PB\$, LIQUID, 1, 18.97, 1.791E-3, 0"
 "STC, PB\$, LIQUID, 2, 0, 0, 0"
 "STC, PB\$, LIQUID, 3, 0, 0, 0"
 19, 2, "MGO\$", 335, "SOLID", "MG", 1, "O", 1
 "STC, MGO\$, SOLID, 1, 11.25, .000326, 0"
 "STC, MGO\$, SOLID, 2, -0.000066, 0., 0"
 "STC, MGO\$, SOLID, 3, 0., 0., 0"
 20, 2, "MGO\$", 871, "LIQUID", "MG", 1, "O", 1
 "STC, MGO\$, LIQUID, 1, 12, 0., 0"
 "STC, MGO\$, LIQUID, 2, 0, 0., 0"
 "STC, MGO\$, LIQUID, 3, 0., 0., 0"
 21, 2, "ZRO2\$", 399, "SOLID", "ZR", 1, "O", 2
 "STC, ZRO2\$, SOLID, 1, 20.9, 0, 0"
 "STC, ZRO2\$, SOLID, 2, 0, 0, 0"
 "STC, ZRO2\$, SOLID, 3, 0, 0, 0"
 22, 2, "ZRO2\$", 400, "LIQUID", "ZR", 1, "O", 2
 "STC, ZRO2\$, LIQUID, 1, 23., 0, 0"
 "STC, ZRO2\$, LIQUID, 2, 0, 0, 0"
 "STC, ZRO2\$, LIQUID, 3, 0, 0, 0"
 23, 2, "TIO2\$", 396, "SOLID", "TI", 1, "O", 2
 "STC, TIO2\$, SOLID, 1, 19., 0, 0"
 "STC, TIO2\$, SOLID, 2, 0, 0, 0"
 "STC, TIO2\$, SOLID, 3, 0, 0, 0"
 24, 2, "TI4O7\$", 1246, "LIQUID", "TI", 4, "O", 7
 "STC, TI4O7\$, LIQUID, 1, 66., 0, 0"
 "STC, TI4O7\$, LIQUID, 2, 0, 0, 0"
 "STC, TI4O7\$, LIQUID, 3, 0, 0, 0"

25, 2, "LI2O\$", 327, "SOLID", "LI", 2, "O", 1
"STC, LI2O\$, SOLID, 1, 25., 0, 0"
"STC, LI2O\$, SOLID, 2, 0, 0, 0"
"STC, LI2O\$, SOLID, 3, 0, 0, 0"
26, 2, "LI2O\$", 328, "LIQUID", "LI", 2, "O", 1
"STC, LI2O\$, LIQUID, 1, 28., 0, 0"
"STC, LI2O\$, LIQUID, 2, 0, 0, 0"
"STC, LI2O\$, LIQUID, 3, 0, 0, 0"
27, 2, "LIOH\$", 290, "SOLID", "LI", 2, "O", 1
"STC, LIOH\$, SOLID, 1, 25., 0, 0"
"STC, LIOH\$, SOLID, 2, 0, 0, 0"
"STC, LIOH\$, SOLID, 3, 0, 0, 0"
26, 2, "LIOH\$", 291, "LIQUID", "LI", 2, "O", 1
"STC, LIOH\$, LIQUID, 1, 28., 0, 0"
"STC, LIOH\$, LIQUID, 2, 0, 0, 0"
"STC, LIOH\$, LIQUID, 3, 0, 0, 0"
0, 0, "END", -1

INTENTIONALLY LEFT BLANK.

APPENDIX E:
LISTING OF THE PROGRAM MAKEJANC.BAS

INTENTIONALLY LEFT BLANK.

' A program that selects from the JANAF Tables the species that
' are going to be refitted.

' In this version, only condensed species are selected.

' Written by EF&A in March/94

' Saved as MAKEJANC.BAS [10 November 1994]

WIDTH 80, 43

DEFINT I-N: DEFSTR S: begin = TIMER
countt = 0: no.rew = 0: lines& = 0

OPEN "condensd" FOR INPUT AS #1

OPEN "jan_file.con" FOR OUTPUT AS #3

OPEN "jan_file.prt" FOR OUTPUT AS #4

mode = 2

jcode.sel:

INPUT #1, idito.no, num.at, species, jcode

IF (species = "E N D" OR jcode < 0) THEN GOTO endd

INPUT #1, s.phase

GOSUB make.con.instr

INPUT #1, stc1: INPUT #1, stc2: INPUT #1, stc3

k.rew = 0

PRINT "Looking for JCODE = "; jcode; " species = "; species

PRINT #4, "Looking for JCODE = "; jcode; " species = "; species

PRINT #3, stc1: PRINT #3, stc2: PRINT #3, stc3

PRINT #3, USING "\ \ &"; species; s.phase

IF (mode = 0) THEN GOTO tab.input

rewindd:

CLOSE #2: OPEN "c:\janaf\tables" FOR INPUT AS #2 LEN = 32767

tab.input:

LINE INPUT #2, s.line: lines& = lines& + 1

' countt = countt + 1: 'if (countt > 55000) then system

IF (EOF(2) <> 0) THEN

SOUND 500, 5: SOUND 250, 10

no.rew = no.rew + 1

k.rew = k.rew + 1

IF (k.rew > 1) THEN

PRINT "Too many rewinds for species "; _
species

```

                                STOP
                                END IF
                                mode = 1
                                GOTO rewindd
END IF

IF (MID$(s.line, 3, 7) <> "JCODE =") THEN
                                GOTO tab.input
END IF
, PRINT "> "; s.line; " < ";
num = VAL(MID$(s.line, 10, 5))
IF (num <> jcode) THEN
                                GOTO tab.input
END IF
SOUND 440, 5
INPUT #2, s.line: PRINT "> "; LEFT$(s.line, 40)
PRINT lines& = lines& + 1

next.line.1:
LINE INPUT #2, s.line: lines& = lines& + 1
IF (VAL(LEFT$(s.line, 10)) < 298!) THEN GOTO next.line.1
IF (VAL(LEFT$(s.line, 10)) > 298.1) THEN
IF (VAL(LEFT$(s.line, 10)) < 298.2) THEN
PRINT #3, s.line: GOTO cont.table
ELSE
PRINT "Can't find T = 298.15 for species "; species: STOP
END IF
END IF
cont.table:
LINE INPUT #2, s.line
IF (VAL(LEFT$(s.line, 9)) = 6000!) THEN
PRINT #3, s.line
GOTO jcode.sel
ELSEIF ((MID$(s.line, 4, 9)) = "PREVIOUS:") THEN
PRINT #3, "999., 999. 99., 9., .9, 999. 99. 9."
mode = 0
GOTO jcode.sel
ELSE
PRINT #3, s.line
GOTO cont.table
END IF

trouble:
PRINT " P O T E N T I A L   T R O U B L E .
PRINT " Can't find T = 6000 K for species "; species
PRINT s.line: STOP

endd:

```

```

all.done = TIMER: e.time = all.done - begin: PRINT #3,
                                PRINT #3,
PRINT "No. of rewinds = "; no.rew:
PRINT #4, "No. of rewinds = "; no.rew:
PRINT #4, "TOTAL No. of lines read = "; lines&
PRINT #4, "Program took "; e.time; "seconds"
e.mins = INT(e.time / 60): e.secs = e.time - 60! * e.mins
e.secs = (INT(10! * e.secs + .5)) / 10!
PRINT #4, "          = "; e.mins; " mins, "; e.secs; " secs"

FOR i = 1 TO 5: SOUND 250, 10: SOUND 500, 5: NEXT
PRINT #3, "E N D": PRINT #3, "E N D"
SYSTEM

make.con.instr:
  FOR i = 1 TO num.at: INPUT #1, symb(i), at.no$(i): NEXT
  FOR i = 1 TO num.at

      IF (ASC(symb(i)) > 64 AND ASC(symb(i)) < 91) THEN GOTO nexxt1
err1:
  PRINT "Error in CON info for species "; species
  PRINT "CHEK: "; symb(i); " "; at.no$(i); ASC(symb(i)); _
      " "; ASC(at.no$(i))

  STOP
nexxt1:
  IF (ASC(at.no$(i)) > 47 AND ASC(at.no$(i)) < 58) GOTO nexxt2
  GOTO err1
nexxt2:
  NEXT

PRINT #3, "CON,"; species; ",,"; "CONDEN,";
  FOR i = 1 TO num.at: PRINT #3, symb(i); ",,"; at.no$(i); ",,";
      NEXT: PRINT #3,
RETURN

```

INTENTIONALLY LEFT BLANK.

APPENDIX F:
LISTING OF THE PROGRAM TIGFIT8

INTENTIONALLY LEFT BLANK.


```

C   CON,AL2O3$,CONDEN,AL,2,0,3,
C   STR, AL2O3$,SOLID , 1,  5.9912400D+00,  8.1126695D+00, -2.7030436D+00
C   STR, AL2O3$,SOLID , 2,  3.6493235D-01,  5.3460303D+00, -2.3637334D+00
C   STR, AL2O3$,SOLID , 3,  2.6465142D-01, -4.0466039D+05,  3.7836920D+01
C   STC, AL2O3$, SOLID, 1, 28.3, 0., 0
C   STC, AL2O3$, SOLID, 2, 0., 0., 0
C   STC, AL2O3$, SOLID, 3, 0., 0., 0

```

C The second output file is named TIGFIT8.AUX. For each input species it
C repeats the output contained in TIGFIT8.OUT. In addition it contains a
C comparison of the back-fitted results with the original JANAF entries, and
C some statistical measures of the goodness-of-fit.

C The TIGER coefficients B(1) - B(7) are dimensionless. The last two
C coefficients, B(8) and B(9), however, are not; they each have the units of
C calories/mole-K as required by the BLAKE program. The back-calculated
C values, however, are stored with the units Joules/mol-K for the comparison
C with the JANAF tabular values.

```

IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION A(7,7), B(9), KX(7)
DIMENSION TK(500), CPP(500), SS(500), HH(500)

CHARACTER XNAME(2)*3,PHASE*6
CHARACTER CONCARD*80,STCARD(80)*1,STCARD2(80)*1,STCARD3(80)*1
CHARACTER PREV*80
PARAMETER (N6=0)
LOGICAL HITEMP

COMMON /COEFS/ B, NUM
COMMON /FUNCS/ TK, CPP, SS, HH, TSTART, ENTRPO, HOFJ,
$   XNAME, PHASE
COMMON /IOSTUF/ LI, LO, LOO
COMMON /LSTSQ/ T, T2, T3, T4, T5, T6, TM2, TM3, TM4, TM5, TM6,
$   Y, YT2, YT3, TM, SN, YTM2, YT, YTM, YTM3
DATA PHASE/ 'GAS  ' /
DATA TSTART / 298.15D0 /
WRITE (0,*) "TSTART = ", TSTART
WRITE (0,*)
LLO = 3
LOO = 4
LI = 5
LO = 6
LII = 8
OPEN (LLO,FILE='TIGFIT8.OUT',STATUS='UNKNOWN')
OPEN (LO,FILE=' TIGFIT8.AUX',STATUS='UNKNOWN')
OPEN (LOO,FILE='SUMMARY', STATUS='UNKNOWN')
OPEN (LI,FILE=' ',STATUS='OLD')

```

```

REWIND LI

11 READ (LI,1019,END=999) CONCARD, STCARD
   IF (CONCARD .NE. 'E N D') THEN
       GOTO 12
   ELSE
       STOP
   ENDIF
1019 FORMAT (A80/80A1)
12 IF (PREV .NE. CONCARD) THEN
       PREV = CONCARD
       WRITE (LLO, 1018) CONCARD
   ENDIF
   IF (STCARD(3) .EQ. 'C') THEN
       READ (LI,1017,END=999) STCARD2
       READ (LI,1017,END=999) STCARD3
   ENDIF
1018 FORMAT (A80)
1017 FORMAT (80A1)

       READ(LI,1020,END=999) XNAME, PHASE
1020 FORMAT(2A3, 2X, A6)

       WRITE (N6,2008) XNAME, TSTART
2008  FORMAT(1X,'Processing ==> ', 2A3, ' from ', F7.2, ' K to 6000 K'
$ )
C Initialize the elements that will make up the matrix
  CALL FMATR (0, X1, X2)

499 FORMAT (A13)
C Read the next line of data; look for T = 298.15
15 READ(LI, *, END=33, ERR=98766) TEM, CP, ENTRP0, X,
$ HMH298, HOFJ
   IF (TEM .LT. 298.1D0) THEN
       GOTO 15
   ELSEIF (TEM .GT. 298.2D0) THEN
       GOTO 16
   ENDIF
   HOF298 = 1.D3*HOFJ
   S298 = ENTRP0

501 FORMAT (' CHEKK: ', 4(E16.6))

C Look for TSTART
16 IF (TEM .GT. TSTART-1.D-2) THEN
       IF (TEM .LT. TSTART+1.D-2) THEN
           GOTO 20
       ELSE

```

```

                GOTO 17
            ENDIF
        ENDIF
    17 WRITE (LO, 502) TEM
502 FORMAT (' T R O U B L E ! ', F7.2, ' is not TSTART. Abort.')
```

STOP

C Yes, it is TSTART

C Start saving data

```

20             KT=1
                TK(1) = TEM
                CPP(1) = CP
                SS(1) = ENTRPO
                HH(1) = 1.D3*HMH298
C             HF(1) = 1.D3*HOFJ
                CALL FMATR (1, TK(1), CP)
                HITEMP = .FALSE.
C Keep reading the table

35 KT = KT + 1
36 READ(LI,*, END=33, ERR=36) TEM, CP, ENTROP, X, HMH298, HOFT
C WRITE(LO,501) TEM, CP, ENTROP, X1, HMH298, HOFT, X2, X3
  IF (CP .EQ. 9.99D2 .OR. CP .EQ. 9.99D3) THEN
      WRITE (6,*) 'KT = ', KT, TEM
      write (0, *) "KT = ", KT, TEM
      GOTO 60
  END IF

  TK(KT) = TEM
  CPP(KT) = CP
  SS(KT) = ENTROP
  HH(KT) = 1.D3*HMH298
C HF(KT) = 1.D3*HOFT
C WRITE (LO,*) KT, TEM, CP, ENTROP
  IF (TEM .LE. 5999.5D0) THEN
      GOTO 49
  ELSE IF (TEM .GE. 6000.D0) THEN
      HITEMP = .TRUE.
      GOTO 60
  END IF

49 CALL FMATR (1, TK(KT), CP)
  IF (.NOT. HITEMP) GOTO 35
60 NUM=KT-1
  IF (HITEMP) NUM=KT

C Form elements of the matrix
  B(1) = Y
```

B(2) = YT
B(3) = YT2
B(4) = YT3
B(5) = YTM
B(6) = YTM2
B(7) = YTM3
A(1,1) = SN
A(2,5) = SN
A(3,6) = SN
A(4,7) = SN
A(5,2) = SN
A(6,3) = SN
A(7,4) = SN
A(2,1) = T
A(3,5) = T
A(4,6) = T
A(1,2) = T
A(5,3) = T
A(6,4) = T
A(3,1) = T2
A(2,2) = T2
A(4,5) = T2
A(1,3) = T2
A(5,4) = T2
A(4,1) = T3
A(2,3) = T3
A(1,4) = T3
A(3,2) = T3
A(5,1) = TM
A(2,6) = TM
A(3,7) = TM
A(1,5) = TM
A(6,2) = TM
A(7,3) = TM
A(6,1) = TM2
A(2,7) = TM2
A(5,5) = TM2
A(1,6) = TM2
A(7,2) = TM2
A(7,1) = TM3
A(5,6) = TM3
A(1,7) = TM3
A(6,5) = TM3
A(2,4) = T4
A(3,3) = T4
A(4,2) = T4
A(3,4) = T5
A(4,3) = T5

```

A(4,4) = T6
A(5,7) = TM4
A(6,6) = TM4
A(7,5) = TM4
A(6,7) = TM5
A(7,6) = TM5
A(7,7) = TM6
CALL SOLVEX(7, A, B, KX, 0, KERR)
IF (KERR .LE. 0) THEN
    GOTO 66
ELSE
    STOP
END IF
C Compute the integration constants by fitting at TSTART
66 X= 2.9815D-1
    B(8) = HOF298 - 298.15D0 * 8.31451 * ( ( ( B(4) * X / 4.0 +
$ B(3)/3.0 ) * X + B(2) / 2.0 ) * X + B(1) - ( ( B(7) /
$ ( X * 2.0) + B(6)) / X - B(5) * DLOG(X) ) / X )
    B(9) = S298 - 8.31451 * ( ( ( B(4) * X / 3.0 + B(3) / 2.0 )
A * X + B(2) ) * X + B(1) * DLOG(X) - ( ( B(7) / (3.0 * X) +
A B(6) / 2.0 ) / X + B(5) ) / X )

    BCAL8 = B(8)/4.184D0
    BCAL9 = B(9)/4.184D0
    WRITE (LO, 70) XNAME, PHASE
70 FORMAT(/5X, 2A3, /, 5X, 'The phase is ', A6, / )

    WRITE (LO, 80)
80 FORMAT (' The STR instructions for the TIGER code are (units = Jou
$les, mols, & K): '/')
    I=1
    WRITE (LO,90) XNAME, PHASE, I, B(1), B(2), B(3)
    WRITE (LLO, 95) XNAME, PHASE, I, B(1), B(2), B(3)
    I = 2
    WRITE (LO,90) XNAME, PHASE, I, B(4), B(5), B(6)
    WRITE (LLO, 95) XNAME, PHASE, I, B(4), B(5), B(6)
    I = 3
    WRITE (LO, 90) XNAME, PHASE, I, B(7), B(8), B(9)
    WRITE (LLO, 95) XNAME, PHASE, I, B(7), BCAL8, BCAL9
90 FORMAT(5X, 'STR, ', 2A3, ', ', A6, ', ', I2, 3(2H, ,1PD14.7))
95 FORMAT('STR, ', 2A3, ', ', A6, ', ', I2, 3(2H, ,1PD14.7))

    WRITE (LLO, 1017) STCARD
    IF (STCARD(3) .EQ. 'C') THEN
        WRITE (LLO, 1017) STCARD2
        WRITE (LLO, 1017) STCARD3
    ENDIF

```

```
CALL GOODF8
GO TO 11
```

```
999 WRITE (LLO, *) TEM, CP, X1, HMM298, ENTROP, X2
33 STOP
98766 WRITE (LO, 998) CONCARD
998 FORMAT (' Uh, OOOOH! T R O U B L E ' / A80 /)
WRITE (LO, *) "TEM, CP, ENTRPO, X, HMM298, HOFJ"
WRITE (LO, *)
WRITE (LO, *) TEM, CP, ENTRPO, X, HMM298, HOFJ
```

```
END
```

```
SUBROUTINE SOLVEX(N, A, B, KX, K, KERR)
```

```
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION A(7,7), B(9), KX(7), T(7)
EPS = 0.0
KERR = 0
IF (K) 10, 10, 160
10 DO 20 J = 1, N
20 KX(J) = J
DO 150 I = 1, N
IM1 = I - 1
IF (IM1) 50, 50, 30
30 DO 40 J = 1, IM1
KJ = KX(J)
Q = A(I, KJ)
JP1 = J + 1
DO 40 L = JP1, N
KL = KX(L)
40 A(I, KL) = A(I, KL) - Q * A(J, KL)
50 KI = KX(I)
Q = DABS(A(I, KI))
IF (N - I) 100, 100, 60
60 L = I
IP1 = I + 1
DO 80 J = IP1, N
KJ = KX(J)
AIKJ = DABS(A(I, KJ))
IF (AIKJ - Q) 80, 80, 70
70 Q = AIKJ
L = J
80 CONTINUE
IF (L - I) 90, 100, 90
90 J = KX(L)
KX(L) = KX(I)
KX(I) = J
```

```

100 IF (Q - EPS) 110, 110, 120
110 KERR = 1
    RETURN
120 KI = KX(I)
    Q = A(I, KI)
    IF (N - I) 150, 150, 130
130 DO 140 J = IP1, N
    KJ = KX(J)
140 A(I, KJ) = A(I, KJ) / Q
150 CONTINUE
160 DO 200 I = 1, N
    Q = B(I)
    IM1 = I - 1
    IF (IM1) 190, 190, 170
170 DO 180 J = 1, IM1
    KJ = KX(J)
180 Q = Q - A(I, KJ) * B(J)
190 KI = KX(I)
200 B(I) = Q / A(I, KI)
    DO 230 IC = 1, N
    I = N - IC + 1
    Q = B(I)
    IF (N - I) 230, 230, 210
210 IP1 = I + 1
    DO 220 J = IP1, N
    KJ = KX(J)
220 Q = Q - T(J) * A(I, KJ)
230 T(I) = Q
    DO 240 J = 1, N
    KJ = KX(J)
240 B(KJ) = T(J)
    RETURN
    END

```

SUBROUTINE CALC (T,HOFJ,CP,S,H)

IMPLICIT REAL*8 (A-H,O-Z)

COMMON /COEFS /B(9)

TH = T / 1000.0

TH2 = TH * TH

TH3 = TH2 * TH

RTH = 1.0 / TH

RTH2 = RTH * RTH

RTH3 = RTH2 * RTH

CP =(B(1) + B(2)*TH + B(3)*TH2 + B(4)*TH3 + B(5)*RTH + B(6)*RTH2
\$ + B(7)*RTH3)* 8.31451

X = DLOG(TH)

```

S = 8.31451 * ( B(1)*X + B(2)*TH + B(3)*TH2/2.0 + B(4)*TH3/3.0
$ - B(5)*RTH - B(6)*RTH2/2.0 - B(7)*RTH3/3.0 ) + B(9)
H = ( T * 8.31451 * ( B(1) + B(2)*TH/2.0 + B(3)*TH2/3.0 +
$ B(4)*TH3/4.0 + B(5)*RTH*X - B(6)*RTH2 - B(7)*RTH3/2.0)
$ + B(8) - HOFJ )
RETURN
END

```

```

SUBROUTINE FMATR (IOPT, TEM, CP)
IMPLICIT REAL*8 (A-H,O-Z)

```

```

COMMON /LSTSQ/ T, T2, T3, T4, T5, T6, TM2, TM3, TM4, TM5, TM6,
$ Y, YT2, YT3, TM, SN, YTM2, YT, YTM, YTM3

```

```

IF (IOPT .NE. 0) GOTO 10
Y = 0.0
YT = 0.0
YT2 = 0.0
YT3 = 0.0
YTM = 0.0
YTM2 = 0.0
YTM3 = 0.0
SN = 0.0
T = 0.0
T2 = 0.0
T3 = 0.0
T4 = 0.0
T5 = 0.0
T6 = 0.0
TM = 0.0
TM2 = 0.0
TM3 = 0.0
TM4 = 0.0
TM5 = 0.0
TM6 = 0.0
GOTO 20
10 X = CP / 8.31451

C CHECK print
C WRITE (LOO, 9999) TEM, CP, T, X, Y
C9999 FORMAT (' TEM & Cp = ', 2F10.2, 3F15.4)

```

```

TH = TEM / 1000.0
TH2 = TH * TH
TH3 = TH * TH2
RTH = 1.0 / TH
RTH2 = RTH * RTH
RTH3 = RTH * RTH2

```

```

Y = Y + X
YT = YT + X * TH
YT2 = YT2 + X * TH2
YT3 = YT3 + X * TH3
YTM = YTM + X * RTH
YTM2 = YTM2 + X * RTH2
YTM3 = YTM3 + X * RTH3
SN = SN + 1.0
T = T + TH
T2 = T2 + TH2
T3 = T3 + TH3
T4 = T4 + TH2 * TH2
T5 = T5 + TH2 * TH3
T6 = T6 + TH3 * TH3
TM = TM + RTH
TM2 = TM2 + RTH2
TM3 = TM3 + RTH3
TM4 = TM4 + RTH2 * RTH2
TM5 = TM5 + RTH2 * RTH3
TM6 = TM6 + RTH3 * RTH3
20 RETURN
END

```

SUBROUTINE GOODF8

```

C A subroutine for computing the goodness-of-fit of TIGFIT8
C The TIGER coefs B(8) and B(9) are converted to
C calories/mole-K for use in the thermo library.

C The back-calculated values are kept in Joules/mol-K for the
C comparison with the JANAF tabular values.

C 14 October 1994

```

IMPLICIT REAL*8 (A-H,O-Z)

CHARACTER XNAME(2)*3, PHASE*6

COMMON /COEFS/ B(9), NUM, NNUM

COMMON /FUNCS/ TK(500), CPP(500), SS(500), HH(500),

\$ TSTART, ENTRPO, HOFJ, XNAME, PHASE

COMMON /IOSTUF/ LI, LO, LOO

WRITE (4, 70) XNAME, PHASE

70 FORMAT(5X, 2A3, 8X, 'The phase is ', A6,)

SUMC1 = 0.0

SUMC2 = 0.0

SUMS1 = 0.0

```
SUMS2 = 0.0
SUMH1 = 0.0
SUMH2 = 0.0
SUMH3 = 0.0
```

```
WRITE (LO, 500)
500 FORMAT (/55X,'H(T)- H(C)-' /
$ '      T      Cp(T) Cp(C) Delta      S(T)      S(C)      Delta      '
$ 'Hstart Hstart  Delta')
```

```
HHZERO = HH(1)
CALL CALC (298.15D0, HOFJ, CP, S, H298)
DO 100 I=1,NUM
T=TK(I)
CALL CALC (T,HOFJ,CP,S,H)
```

```
501 FORMAT (' > ', I4, 2F15.2)
DELTACP = CPP(I) - CP
IF (HH(I) .NE. 0.0) THEN
NUMH = NUM
DELTAS = SS(I) - S
```

```
C IF (TK(I) .LT. 9.D2)
C $ WRITE (LO,*) "Chekk1: ", TK(I) , HH(I), H, HHZERO
```

```
HH(I) = HH(I) - HHZERO
```

```
H = H - HHZERO - H298
```

```
DELTAH = HH(I) - H
IF (HH(I) .NE. 0.) DELTHB = DELTAH/HH(I)
```

```
C WRITE (LO, 501) I, H, HH(I), H600
```

```
ELSE
```

```
DELTAS = 0.
```

```
DELTAH = 0.
```

```
NUMH=NUM-1
```

```
GOTO 80
```

```
END IF
```

```
80 WRITE (LO,1000) I, T, CPP(I), CP, DELTACP, SS(I), S, DELTAS,
$ HH(I), H, DELTAH
1000 FORMAT (I3, F7.0, F7.2, F7.2, F6.2, F9.2, F8.2, F6.2,
$ F9.0, F8.0, F7.1)
```

```
SUMC1 = SUMC1 + DELTACP
SUMC2 = SUMC2 + DELTACP**2
SUMS1 = SUMS1 + DELTAS
SUMS2 = SUMS2 + DELTAS**2
```

```

SUMH1 = SUMH1 + DELTAH
SUMH2 = SUMH2 + DELTAH**2
100 SUMH3 = SUMH3 + DELTHB**2

NUMC=NUM

120 ERRC1 = SUMC1/NUMC
ERRC2 = SQRT(SUMC2/(NUMC-1))
ERRS1 = SUMS1/NUMH
ERRS2 = SQRT(SUMS2/(NUMH-1))
ERRH1 = SUMH1/NUMH
ERRH2 = SQRT(SUMH2/(NUMH-1))
ERRH3 = 1.D2*SQRT(SUMH3/(NUMH-1))

WRITE (LO,1004) NUMC, NUMH
1004 FORMAT (/
$ ' Number of Cp's = ',I3,19X,'Number of sets of S & H =', I3)

WRITE (LO, 1002) ERRC1, TSTART, ERRC2, ENTRPO, ERRS1,
$ HOFJ, ERRS2, ERRH1, ERRH2, ERRH3

WRITE (LOO,1002) ERRC1, TSTART, ERRC2, ENTRPO, ERRS1,
$ HOFJ, ERRS2, ERRH1, ERRH2, ERRH3
1002 FORMAT (/ ' H & S fixed at T = 298.15'
$ ' Mean Deviation of Cp = ', F6.3 /,
$ ' Fit & comparison start at ', F 7.2,
$ ' RMS Deviation of Cp = ', F6.3 /,
$ ' Entropy at T Fix = ', F11.2,
$ ' Mean Deviation of S = ', F6.3 /,
$ ' Enthalpy at T Fix = ', F10.1,
$ ' RMS Deviation of S =',F9.3/,
$ 31X, ' Mean Deviation of H =',F9.3/,
$ 31X, ' RMS Deviation of H =',F9.3/,
$ 31X, ' RMS Percent Dev of H =',F9.3//)

RETURN
END

```

APPENDIX G:
LISTING OF THE PROGRAM ADDBLINK.BAS

INTENTIONALLY LEFT BLANK.

```
' A program that adds 3 leading blanks to the CON instruction
' for the species to be included in the new BLAKE, but that are
' not in the JANAF TABLES (data for them are obtained from the
' NASA coefficients).
```

```
' Written by EF&A in Nov/94
```

```
' Saved as ADDBLNK.BAS [13 November 1994]
```

```
DEFSTR S: DEFINT I-N
```

```
OPEN "tigfitj.in" FOR INPUT AS #1
```

```
OPEN "dummy" FOR OUTPUT AS #2
```

```
CLS
```

```
PRINT "12345678901234567890123456789012345678901234567890";
```

```
PRINT "123456789012345678901234567890"
```

```
PRINT "          1          2          3          4          5";
```

```
PRINT "          6          7          8"
```

```
inputt:
```

```
LINE INPUT #1, s.line
```

```
IF NOT EOF(1) = -1 THEN GOTO nextt ELSE GOTO endd
```

```
nextt:
```

```
' Look for CON instructions; ignore any other lines
```

```
IF LEFT$(s.line, 3) <> "CON" THEN
```

```
    PRINT #2, s.line
```

```
    GOTO inputt
```

```
' A CON instruction has been found.
```

```
' If it is for Formic Acid, skip it.
```

```
ELSEIF MID$(s.line, 5, 6) = "FORMAC" THEN
```

```
    PRINT #2, s.line
```

```
    GOTO inputt
```

```
END IF
```

```
' Only CON instructions for the other species are left.
```

```
' Put 3 blanks in front of them
```

```
s.line = SPACE$(3) + s.line
```

```
    PRINT s.line: ' Print the modified line on the screen.
```

```
PRINT #2, s.line: ' Put the modified line in the new output file.
```

```
GOTO inputt
```

```
,
```

```
endd:
```

```
SYSTEM
```

INTENTIONALLY LEFT BLANK.

APPENDIX H:
LISTING OF THE DOS BATCH FILE MAKELIBS.BAT

INTENTIONALLY LEFT BLANK.

```

@echo off
:: 21 March 1995
:: This batch file creates the files GASES.OUT and CONDEN.OUT
:: that contain the TIGER-type coefficients for the latest BLAKE
:: program.

if "%1" == "1" goto skip1
if "%1" == "2" goto skip2
if "%1" == "3" goto skip3
if "%1" == "4" goto skip4

:: Form file SUBSETG.LIB. It contains the JANAF data for the
:: species in file GASES.

Qb/run maksubjg
:skip1

:: Now form file JAN_FILE.GFAS. It rearranges the data in
:: SUBSETG.LIB to put them into the order of species in GASES.

Qb/run makejang
:skip2

:: Form file JAN_FILE.CON. It contains the JANAF data for the
:: condensed species that will be in the new library.
Qb/run makejanc

:: Run the fitting program TIGFIT8 on the file that contains the
:: thermo data for the gaseous species. It forms the file
:: TIGFIT8.OUT that contains the TIGER coefficients for the
:: gaseous species.
:skip3
Tigfit8 jan_file.gas

:: Rename the output file for later use
If exist gases.out del gases.out
Ren tigfit8.out gases.out

:: Run the fitting program on the file that contains the thermo
:: data for the condensed species.
Tigfit8 jan_file.con
:: Rename the output file to save it for later use.
If exist conden.out del conden.out
Ren tigfit8.out conden.out

:: Run the program CPHSJN. It uses the NASA coefficients to form
:: the thermo data for the seven species not in the JANAF Tables.
Cphsjn

```

```
:: Place blanks in cols 1-3 for the CON instructions for these
:: seven species. Its output is placed in file DUMMY.
Qb/run addblnk
```

```
:: DUMMY is the input for the fitting program TIGFIT8. It has to
:: be renamed TIGFITJ.IN to be used.
If exist tigfitj.in del tigfitj.in
Ren dummy tigfitj.in
```

```
:: Run the fitting program
Tigfit8 tigfitj.in
:: Merge the two outputs that contain the coefficients for the
:: gaseous species. The DOS COPY program is used to do this.
Copy gases.out + tigfit8.out dummy > nul:
```

```
:: Rename the merged files.
If exist gases.out del gases.out
Ren dummy gases.out
```

```
:skipp4
Cd\blake\tests
```

```
:: Save previous version of NEWLIB.LIB
If exist newlib.lib copy newlib.lib newlib.sav
If exist newlib.lib del newlib.lib
```

```
:: Merge all output files
Copy topper.lib + d:\Blake\gough\gases.out + d:\Blake\gough\conden.out +
ENDEND.ENDnewlib.lib
```

```
:: Use Wordstar to examine NEWLIB.LIB
Call WdStar Blake\tests newlib.lib n
Cd \Blake\tests
:: Form the new binary library
Formlib newlib.lib
```

```
:: Use WordStar to examine the alphanumeric output file created
:: by FORMLIB
Call WdStar Blake\tests blake.out n
```

```
:: Print an admonitory message on the screen
Echo.
Echo The new library is ready for testing.
Echo.
Echo Don't forget to copy Bynlyb.dat to Sbynlyb.dat
Echo.
Echo.
```

NO. OF
COPIES ORGANIZATION

2 DEFENSE TECHNICAL INFO CTR
ATTN DTIC DDA
8725 JOHN J KINGMAN RD
STE 0944
FT BELVOIR VA 22060-6218

1 DIRECTOR
US ARMY RESEARCH LAB
ATTN AMSRL OP SD TA
2800 POWDER MILL RD
ADELPHI MD 20783-1145

3 DIRECTOR
US ARMY RESEARCH LAB
ATTN AMSRL OP SD TL
2800 POWDER MILL RD
ADELPHI MD 20783-1145

1 DIRECTOR
US ARMY RESEARCH LAB
ATTN AMSRL OP SD TP
2800 POWDER MILL RD
ADELPHI MD 20783-1145

ABERDEEN PROVING GROUND

2 DIR USARL
ATTN AMSRL OP AP L (305)

<u>NO. OF COPIES</u>	<u>ORGANIZATION</u>
1	CHAIRMAN DOD EXPLOSIVES SAFETY BOARD RM 856 C HOFFMAN BLDG 1 2461 EISENHOWER AVE ALEXANDRIA VA 22331-0600
1	OPM PALADIN ATTN SFAE AR HIP IP MR R DE KLEINE DEPT OF ARMY PCTNY ARSNL NJ 07806-5000
1	COMMANDER ATTN AMSMC PBM E L LAIBSON USA ARDEC PBMA PCTNY ARSNL NJ 07806-5000
3	PEO ARMAMENTS ATTN AMCPM TMA K RUSSELL AMCPM TMA 105 AMCPM TMA 120 PM TMAS PCTNY ARSNL NJ 07806-5000
5	DIRECTOR ATTN SARWV RD G CARAFANO R THIERRY R HASOENBEIN BENET LABORATORIES WATERVLIET NY 12189
3	COMMANDER ATTN AMSMC IRC G COWAN SMCAR ESM R W FORTUNE R ZASTROW USA AMCCOM ROCK ISLAND IL 61299-7300
1	COMMANDER ATTN ASQNC ELC IS L R MYER CENTER R&D TECHNICAL LIBRARY USACECOM FT MONMOUTH NJ 07703-5301
1	COMMANDANT ATTN AVIATION AGENCY USA AVIATION SCHOOL FT RUCKER AL 36360

<u>NO. OF COPIES</u>	<u>ORGANIZATION</u>
1	DIRECTOR ATTN ATCD MA MAJ WILLIAMS HQ TRAC RPD FT MONROE VA 23651-5143
1	HEADQUARTERS ATTN AMCICP AD M F FISETTE 5001 EISENHOWER AVE USA MATERIEL COMMAND ALEXANDRIA VA 22333-0001
3	COMMANDER ATTN SMCAR CCS SMCAR CCH T L ROSENDORF SMCAR CCH V E FENNELL USA ARDEC PCTNY ARSNL NJ 07806-5000
5	COMMANDER ATTN SMCAR AEE B D DOWNS S EINSTEIN A BRACUTI A BEARDELL D CHIU USA ARDEC PCTNY ARSNL NJ 07806-5000
1	COMMANDER ATTN SMCAR AEE J LANNON USA ARDEC PCTNY ARSNL NJ 07806-5000
1	COMMANDER ATTN SMCAR AE J PICARD USA ARDEC PCTNY ARSNL NJ 07806-5000
1	COMMANDER ATTN SMCAR AES S KAPLOWITZ USA ARDEC PCTNY ARSNL NJ 07806-5000
2	COMMANDER ATTN TECHNICAL LIBRARY D MANN PO BOX 12211 USARO RSRCH TRI PK NC 27709-2211

<u>NO. OF COPIES</u>	<u>ORGANIZATION</u>
1	COMMANDER ATTN STRBE WC TECH LIB VAULT BLDG 315 USA BELVOIR R&D CTR FT BELVOIR VA 22060-5606
1	COMMANDER USA TRAC FT LEE DEFENSE LOGISTICS STUDIES FT LEE VA 23801-6140
1	PRESIDENT USA ARTILLERY BOARD FT SILL OK 73503
1	COMMANDANT USA COMMAND & GEN STAFF COLLEGE FT LEAVENWORTH KS 66027-5200
1	COMMANDANT ATTN REV & TNG LIT DIV USA SPECIAL WARFARE SCHOOL FT BRAGG NC 28307
1	COMMANDER ATTN SMCRA QA HI LIBRARY RADFORD ARMY AMMO PLANT RADFORD VA 24141
1	COMMANDANT ATTN STSF TSM CN USA FIELD ARTILLERY SCHOOL FT SILL OK 73503-5600
3	COMMANDER ATTN AMXST MC 3 S LEBEAU C BEITER 220 SEVENTH ST NE USA FOREIGN SCIENCE & TECH CTR CHARLOTTESVILLE VA 22901
1	COMMANDANT ATTN ATSF CO MW B WILLIS USA FIELD ARTILLERY CTR & SCHOOL FT SILL OK 73503
1	OFFICE OF NAVAL RESEARCH ATTN CODE 473 R S MILLER 800 N QUINCY ST ARLINGTON VA 22217

<u>NO. OF COPIES</u>	<u>ORGANIZATION</u>
2	COMMANDER ATTN SEA 62R SEA 64 NAVAL SEA SYSTEMS COMMAND WASHINGTON DC 20362-5101
1	COMMANDER ATTN AIR 954 TECHNICAL LIBRARY NAVAL AIR SYSTEMS COMMAND WASHINGTON DC 20360
1	NAVAL RESEARCH LABORATORY TECHNICAL LIBRARY WASHINGTON DC 20375
3	COMMANDER ATTN 610 C SMITH 6110J K RICE 6110C S PETERS NAVAL SURFACE WARFARE CTR INDIAN HEAD DIVISION INDIAN HEAD MD 20640-5035
6	COMMANDER ATTN CODE G33 T DORAN J COPLEY J L EAST CODE G30 GUNS & MUNITIONS DIV CODE G301 D WILSON CODE G32 GUNS SYSTEMS DIV CODE E23 TECHNICAL LIBRARY NAVAL SURFACE WARFARE CTR DAHLGREN VA 22448-5000
1	COMMANDER ATTN E CHAN CODE 270P1 NSWC IH DIV 101 STRAUS AVE INDIAN HEAD MD 20640
1	COMMANDER ATTN R RAST CODE 3120 NSWC IH DIV 101 STRAUS AVE INDIAN HEAD MD 20640
1	COMMANDER ATTN R SIMMONS CODE 210P1 NSWC IH DIV 101 STRAUS AVE INDIAN HEAD MD 20640

<u>NO. OF COPIES</u>	<u>ORGANIZATION</u>	<u>NO. OF COPIES</u>	<u>ORGANIZATION</u>
2	COMMANDER ATTN CODE 6210 S BOYLES N ALMEYDA NSWC IH DIV 101 STRAUS AVE INDIAN HEAD MD 20640	1	NORTH CAROLINA STATE UNIVERSITY ATTN J G GILLIGAN BOX 7909 1110 BURLINGTON ENG LABS RALEIGH NC 27695-7909
1	COMMANDER ATTN A ATWOOD CODE 3891 NAWC CHINA LAKE CA 93555	1	THE JOHNS HOPKINS UNIV CPIA ATTN T CHRISTIAN 10630 LITTLE PATUXENT PKWY SUITE 202 COLUMBIA MD 21044-3200
1	OLAC PL/TSTL ATTN D SHIPLETT EDWARDS AFB CA 93523-5000	3	INSTITUTE FOR ADVANCED STUDIES DR H FAIR DR T KIEHNE DR I McNAB 4030 2 W BAKER LN AUSTIN TX 78759-5329
10	CENTRAL INTELLIGENCE AGENCY OFC OF CENTRAL REF DISSEMINATION BR ROOM GE 47 HQS WASHINGTON DC 20502	1	SRI INTERNATIONAL ATTN TECHNICAL LIBRARY 333 RAVENSWOOD AVE PROPULSION SCIENCES DIVISION MENLO PK CA 94025
1	CENTRAL INTELLIGENCE AGENCY ATTN J E BACKOFEN HQ RM 5F22 WASHINGTON DC 20505	1	SPARTA ATTN DR M HOLLAND 9455 TOWNE CENTER DR SAN DIEGO CA 92121-1964
1	DIRECTOR ATTN MS L 355 A BUCKINGHAM PO BOX 808 LLNL LIVERMORE CA 94550	2	UNITED DEFENSE LP ATTN MR J DYVIK B GOODELL 4800 E RIVER RD MINNEAPOLIS MN 55421-1498
2	DIRECTOR ATTN R ARMSTRONG S VOSEN DIVISION 8357 SNL COMB RSCH FAC LIVERMORE CA 94551-0469	2	RADFORD ARMY AMMUN PLANT ATTN D WORRELL E SANFORD MNGR MFG ENGNNG DEPT PO BOX 1 RADFORD VA 24141
1	UNIVERSITY OF ILLINOIS ATTN PROF H KRIER 144 MEB 1206 N GREEN ST DEPT OF MECH INDUST ENG URBANA IL 61801	1	PAUL GOUGH ASSOCIATES ATTN P S GOUGH 1048 SOUTH ST PORTSMOUTH NH 03801-5423
2	PENNSYLVANIA STATE UNIVERSITY ATTN J BROWN 312 MECHANICAL ENGINEERING BLDG DEPT OF MECH ENG UNIVERSITY PK PA 16802	1	PHYSICS INTERNATIONAL LIBRARY ATTN H W WAMPLER PO BOX 5010 SAN LEANDRO CA 94577

<u>NO. OF COPIES</u>	<u>ORGANIZATION</u>
2	ROCKWELL INTERNATIONAL ATTN BA08 J E FLANAGAN J GRAY 6633 CANOGA AVE ROCKETDYNE DIVISION CANOGA PK CA 91304
2	SCIENCE APPLICATIONS INC ATTN J BATTEH L THORNHILL 1519 JOHNSON FERRY RD STE 300 MARIETTA GA 30062-6438
1	ELI FREEDMAN & ASSOCIATES ATTN E FREEDMAN 2411 DIANA RD BALTIMORE MD 21209
1	ROCKETDYNE ATTN O HEINEY MAIL STOP BA26 6633 CANOGA AVE CONOGA PARK CA 91304
1	THIOKOL ATTN DR R WARDLE M S 244 G DIXON M S 230 PO BOX 707 BRIGHAM CITY UT 84302-0707
1	THIOKOL ATTN DR D DILLEHAY MAIL STOP 703 11 PO BOX 1149 MARSHALL TX 75671
1	THIOKOL ATTN DR R WILLER 55 THIOKOL RD ELKTON MD 21922
1	VERITAY TECHNOLOGY INC ATTN E FISHER 4845 MILLERSPORT HWY E AMHERST NY 14051-0305
1	BATTELLE ATTN TACTEC LIBRARY J N HUGGINS 505 KING AVE COLUMBUS OH 43201-2693

<u>NO. OF COPIES</u>	<u>ORGANIZATION</u>
2	CA INSTITUTE OF TECHNOLOGY ATTN L D STRAND MS 512 102 D ELLIOT 4800 OAK GROVE DR JET PROPULSION LABORATORY PASADENA CA 91109
1	LOCHKEED MARTIN DEFENSE SYSTEMS ATTN DR J MANDZY MAIL DROP 43 220 100 PLASTICS AVE DEFENSE SYSTEMS DIVISION PITTSFIELD MA 01201
1	DR W J SARGEANT DEPT OF ELECTRICAL ENGINEERING BONNER HALL RM 312 BUFFALO NY 14260
1	SCIENCE APPLICATION INTL CORP ATTN DR G CHRYSOMALLIS 8400 NORMANDELE BLVD STE 939 MINNEAPOLIS MN 55437
1	CRAFT TECH ATTN DR S DASH 174 N MAIN ST BLDG 3 PO BOX 1150 DUBLIN PA 18917
1	SANDIA NATIONAL LAB ATTN M GRUBELICH DIV 2515 PO BOX 5800 ALBUQUERQUE NM 87185
1	IMI SERVICES USA ATTN G RASHBA 2 WISCONSIN CIRCLE SUITE 420 CHEVY CHASE MD 20815
2	ALLIANT TECHSYSTEMS INC ATTN R E TOMPKINS J KENNEDY 7225 NORTHLAND DR BROOKLYN PARK MN 55428

NO. OF
COPIES ORGANIZATION

3 OLIN ORDNANCE
ATTN V MCDONALD LIBRARY
T BOURGEOIS
D WORTHINGTON
PO BOX 222
ST MARKS FL 32355

ABERDEEN PROVING GROUND, MD

4 CDR USAATC
ATTN S WALTON
G RICE
D LACEY
C HERUD

1 DIR USAHEL
ATTN J WEISZ

40 DIR USARL
ATTN AMSRL WT P A W HORST
AMSRL WT T W F MORRISON
AMSRL WT PC R ANDERSON
D BEYER
J HEIMERL
T KOTLAR
AMSRL WT PA W OBERLE (10 CPS)
A BIRK
T COFFEE
J DeSPIRITO
ARPAD JUHASZ
J KNAPTON
C LEVERITT
M McQUAID
I STOBIE
P TRAN
K WHITE
G WREN
B BURNS
A BRANT
L-M CHANG
J COLBURN
P CONROY
G KELLER
D KOOKER
D KRUCZYNSKY
R LIEB
T MINOR
M NUSCA
T ROSENBERGER
AMSRL WT PB P PLOSTINS

<u>NO. OF COPIES</u>	<u>ORGANIZATION</u>
2	<p>RARDE GS2 Division Building R31 ATTN: Dr. C. Woodley Dr. G. Cook Fort Halstead Sevenoaks, Kent TN14 7BP ENGLAND</p>
1	<p>Materials Research Laboratory Salisbury Branch ATTN: Anna Wildegger Gaissmaier Explosives Ordnance Division Salisbury South Australia 5108</p>
1	<p>R&D Department ATTN: Dr. Pierre Archambault 5 Montee des Arsenaux Le Gardeur, Quebec, Canada J5Z 2P4</p>
1	<p>TZN Forschungs- und Entwicklungszentrum ATTN: Mr. T. Weise, Dr. -Ing. Abt. SE, Fachbereich: Hochleistungspulstechnik Neuensothriether Strabe 20 D-29345 UnterlüB, Germany</p>
1	<p>Ernst-Mach-Institut ATTN: Dr. Gustav-Adolf Schröder Hauptstrabe 18 D-7858 Weil am Rhein, Germany</p>
2	<p>Institut Franco-Allemand ATTN: Dr. M. Samirant Mr. D. Grune F 68301 SAINT-LOUIS Cédex, 12, rue de l'Industrie, B.P. 301, France</p>

INTENTIONALLY LEFT BLANK.

USER EVALUATION SHEET/CHANGE OF ADDRESS

This Laboratory undertakes a continuing effort to improve the quality of the reports it publishes. Your comments/answers to the items/questions below will aid us in our efforts.

1. ARL Report Number/Author ARL-TR-1152 (Oberle) Date of Report July 1996

2. Date Report Received _____

3. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which the report will be used.) _____

4. Specifically, how is the report being used? (Information source, design data, procedure, source of ideas, etc.) _____

5. Has the information in this report led to any quantitative savings as far as man-hours or dollars saved, operating costs avoided, or efficiencies achieved, etc? If so, please elaborate. _____

6. General Comments. What do you think should be changed to improve future reports? (Indicate changes to organization, technical content, format, etc.) _____

CURRENT
ADDRESS

Organization

Name

Street or P.O. Box No.

City, State, Zip Code

7. If indicating a Change of Address or Address Correction, please provide the Current or Correct address above and the Old or Incorrect address below.

OLD
ADDRESS

Organization

Name

Street or P.O. Box No.

City, State, Zip Code

(Remove this sheet, fold as indicated, tape closed, and mail.)
(DO NOT STAPLE)

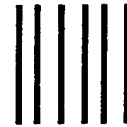
DEPARTMENT OF THE ARMY

OFFICIAL BUSINESS

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO 0001,APG,MD

POSTAGE WILL BE PAID BY ADDRESSEE

**DIRECTOR
U S ARMY RESEARCH LABORATORY
ATTN AMSRL WT PA
ABERDEEN PROVING GROUND MD 21005-5066**



**NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES**

