NWC TP 6449

Computer Program for Internal Aluminum-Fuel-Air Explosions

by
Richard A. Reinhardt
Naval Postgraduate School
Monterey, California
for the
Research Department

AUGUST 1983

NAVAL WEAPONS CENTER
CHINA LAKE, CALIFORNIA 93555

Naval Weapons Center
AN ACTIVITY OF THE NAVAL MATERIAL COMMAND

FOREWORD

This report documents the extension of the continuing research effort on internal blast at the Naval Weapons Center to include aluminized fuels in air. Work was performed during the period 1978–1982.

This effort was supported by the Naval Air Systems Command (NAVAIR) and was executed by the Naval Weapons Center under the Strike Warfare Weaponry Technology Block Program under AIRTASK A03W-03P2/008B/2F32-300-000 (appropriation 1721319.41AJ). This airtask provides for continued exploratory development in the air superiority and air-to-surface mission areas. Mr. H. B. Benefiel, AIR-350, was the cognizant NAVAIR Technology Administrator.

This report was reviewed for technical accuracy by K. J. Graham.

Approved by
E. B. ROYCE, *Head*
Research Department
5 August 1983

Under authority of
K. A. DICKERSON
Capt., USN
Commander

Released for publication by
B. W. HAYS
Technical Director

NWC Technical Publication 6449

Published by ..... Technical Information Department
Collation ..... Cover, 16 leaves
First printing ..... 200 unnumbered copies

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NWC TP 6449	2. GOVT ACCESSION NO. AD-A134 029	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) COMPUTER PROGRAM FOR INTERNAL ALUMINUM-FUEL-AIR EXPLOSIONS		5. TYPE OF REPORT & PERIOD COVERED Final report 1978–1982
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Richard A. Reinhardt		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS AIRTASK A03W-03P2/008B/2F32-300-000
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Weapons Center China Lake, California 93555		12. REPORT DATE August 1983
		13. NUMBER OF PAGES 30
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Aluminum	Internal Explosions
Fuel-Air Explosions	Overpressure
Internal Blast	Reactive Metals

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

See back of form.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

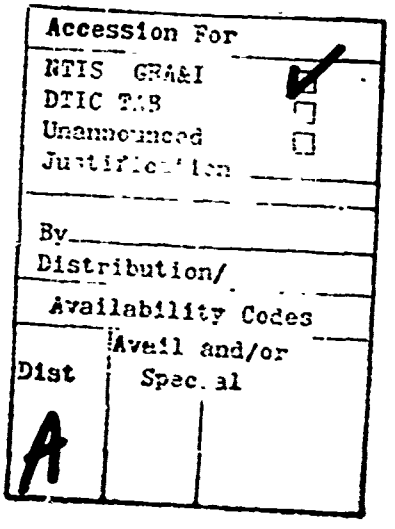
UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

(U) *Computer Program for Internal Aluminum-Fuel-Air Explosions*, by Richard A. Reinhardt, Naval Postgraduate School, China Lake, Calif., Naval Weapons Center, August 1983. 30 pp. (NWC TP 6449, publication UNCLASSIFIED.)

(U) This report documents the internal explosion computer program INAL, used to calculate overpressures, temperatures, and chemical species present in the internal explosion of aluminized fuels in air. A complete listing of the program in HP–BASIC is presented, as well as a discussion of the function performed in each major subroutine.

CONTENTS



INTRODUCTION

This report describes a program used to calculate overpressures in internal explosions of aluminized fuels in air. The program, formerly called INE, has been slightly modified and is called *INAL* (internal explosions with aluminum). An annotated copy of the *INAL* program appears in Appendix A.

The calculations are based on the assumptions that the process as a whole is adiabatic, that the products are uniformly distributed through the volume, that chemical and thermal equilibria prevail, and that the ideal gas law may be used throughout. The adiabatic assumption in a constant volume system is equivalent to the requirement that there be no net change in internal energy; that is, that the internal energy of the products (34 chemical species are considered) must be equal to that of the introduced fuel (inasmuch as all the other starting materials are elements in their thermochemical reference states).

A description of the program can rather logically be divided into two segments: (1) the main program, in which the general procedure is to find, by trial and error, a product temperature for which the adiabatic condition holds; and (2) the subroutine, called *Alcal*, which carries out the equilibrium calculations and then finds the internal energy change corresponding to the equilibrium set of products.

MAIN PROGRAM

INPUT SECTION

Fuel. enter formula and internal energy of formation. Computes formula mass; allows for zero C or H or no fuel.

Conc (accessible from line 1780) resets flags, counters, and amounts of solids to zero. Enter concentration (kg/m^3) of fuel and of Al. Computes moles of fuel and of atoms of Al, C, H, N, O. Computes total initial internal energy.

Temp (for entering trial temperature manually) may be accessed by use of special function key ***k4***. After at least two trials, ***Temp*** may be bypassed and interpolation used to find the new temperature.

COMPUTATION SECTION

Calc calls up the computational subroutines ***Eq***, ***Alcal***, and ***Ex***.

Eq is a subroutine of the main program to evaluate equilibrium constants of formation of each of the 34 chemical species. K_p is first computed from the stored parameters and then converted to K_n (expressed in mole numbers). K_p is defined for each chemical species as the ratio of the activity of the species to the product of the activities of C, Al vapor, H_2, N_2, and O_2, each raised to the power corresponding to the stoichiometric content of the element. In converting the K_p, the activities of gaseous species, including the elements, are changed to mole numbers. For the condensed species the standard state remains the pure phase, so that the activity is unity, except for C.[1]

Alcal is the master subroutine which carries out equilibrium and energy calculations. Results are displayed as "dU" (net) and "T high" or "T low"; dU = 0 is desired for convergence. A new temperature approximation is performed automatically by interpolation, based on former T and dU values; or else it is entered manually with ***k4***.

Ex prepares termination of calculation. Calls up ***Molsum***, finding the total moles of gases.

OUTPUT

Prints results of calculations.

C_v, the molar heat capacity at constant volume, is computed by finding dU at two nearby temperatures. An isentropic parameter, lambda, is found by using also dn/dT (change of mole number of gas with temperature).

Conclusion: various options allowed, as for new concentartions or a new fuel.

[1] Naval Weapons Center. *Adiabatic Computation of Internal Blast from Aluminum-Cased Charges in Air*, by R. A. Reinhardt and A. K. MacDonald, Naval Postgraduate School. China Lake, Calif., NWC, January 1982. (NWC TP 6287, publication UNCLASSIFIED.)

ALCAL SUBROUTINE

Alcal is the major computational subroutine whose task is to find the numbers of moles of the products present at equilibrium at the selected temperature. The conditions to be satisfied (other than for the trivial case of Ar) are the atom balance conditions for C, H, N, O, and Al and the establishment of chemical equilibrium between each compound and its component elements in their reference states at the prevailing temperature.

The master variables (all in mole numbers) are $X = \sqrt{O_2}$, $Y = \sqrt{H_2}$, $Z = \sqrt{N_2}$, Alv = Al metal vapor, Acc = activity of C (standard state = graphite). Of these, Y is always computed in closed form; from one to four of the remaining master variables are found as unknown parameters, using the Newton-Raphson method. The actual number of unknowns is equal to four, reduced by the number of condensed phases present. Possible condensed phases are: Al_2O_3 (solid or liquid), Al (liquid), AlN (solid), C (solid), Al_4C_3 (solid). The presence or absence of each condensed phase is indicated by a flag, to be set as described later. The set of condensed phases is referred to as a *regime*; allowance is made in the program for about 20 different regimes.

Based on the values of the master variables and the K_n of formation, the mole number of each species is computed. Then the material balance in the elements O, N, C, and Al is written in terms of these mole numbers. Thus, there results a set of up to four simultaneous non-linear equations. It is this set which is used as the basis of the Newton-Raphson scheme to find the unknown parameters.

At the conclusion of an iteration, the newly generated values of the master variables are used to repeat the calculations. In favorable situations each iteration results in improvement (although temporary divergence sometimes occurs). Iteration is repeated until errors in the material balances are less than one part in ten thousand.

INITIAL APPROXIMATION

To begin the calculation, an initial approximation of the master variables is made using the subroutine ***Approx***. In this approximation an arbitrary hierarchy of oxygen and nitrogen uptake is assumed. Oxygen is assumed to produce, in order, CO, Al_2O (g), Al_2O_3 (c), H_2O, CO_2, and O_2. When there is insufficient O to convert all Al to Al_2O_3, AlN is assumed present. If there is not sufficient O to convert all C to CO, Al_4C_3 is considered if the temperature is low enough. This rather long subroutine has given many more satisfactory initial conditions than the simpler scheme described in Reference 1. It is used once, or at most twice, for each new concentration that is run.

REGIMES IN *ALCAL*

It is necessary to assume which condensed phases are present, then perform the calculations previously outlined, and finally test for the presence of the condensed phases. The following criteria must be met: (1) the quantity of the phase must be positive and (2) the formation constant must be satisfied. The tests are carried out after the convergence of the Newton's method calculation. If the tests fail, a different regime (set of condensed phases) is tried--the assumptions now based on the currently computed sets of mole numbers.

An index *Nw* is used to identify the regime being considered. Eleven different values of *Nw* are allowed, which account for about 20 different regimes, since for most values of *Nw* graphite may be present or absent. A key to the *Nw* values is given at the end of the program in Appendix A.

LABEL *HOM*

The section of ***Alcal*** beginning at label ***Hom*** is the setup for the Newton's method calculation. (***Hom0*** is merely an early entrance into ***Hom***.) Depending on the regime selected, ***Ii*** (the number of variables) and *Nw* (the index identifying the regime) are evaluated by the subroutines ***Iiset*** and ***Nwset***. The regime is established in a variety of ways. Initially, ***Approx*** gives the first guess. Afterwards, the subroutine ***Almtest*** is used as criterion for all solids except Al_4C_3, which is looked for at ***Hom2***. At ***Exit*** and ***Exit1*** additional tests for alumina and graphite are made.

The various subdivisions of ***Hom*** are designated *Nwx*, where x is partly indicative of the value of *Nw* for the regime considered. The correspondence is imperfect since ***Nw1*** includes the cases of *Nw* = 1, 4, and 5 and ***Nw2*** includes *Nw* = 2 and 6. In each case, initial approximations are set for each variable to be solved, using latest results of the appropriate variable.

Subroutine ***Newt*** is called to carry out the calculations (except *Nw* = 9, for which solution in closed form is possible). Then either a new value of *Nw* is used (as called for by ***Almtest*** and ***Iiset***) or the program switches to ***Hom2***. At this point, if Al_4C_3 had not been presupposed, but was found present, it is necessary to pass once again through ***Approx***. Otherwise exit from the subroutine is prepared.

EXIT FROM *ALCAL*

At ***Exit*** and ***Exit1*** it is necessary to check for negative amounts of alumina and graphite. If found, the appropriate flags must be reset and a return to ***Hom*** is required. If those tests are satisfactory, next, at label ***Energy***, the internal energy, for each species is calculated making use of the stored parameters and the number of moles of the species. If T = 2315 K (the melting point of alumina), the relative

amounts of the two phases of alumina are found from the energy balance. Return to the main program then takes place.

SUBROUTINES IN *ALCAL*

Alntest sets flags for AlN, Al_2O_3, Lq (= liquid Al), and Gr (graphite) based on the present value of the formation expression, as related to the formation constant for the first three, and on whether ***Acc*** has been found less than unity (indicating no graphite).

Iiset and ***Nwset*** evaluate ***Ii*** (the number of variables to be solved for) and ***Nw*** (the index for the solids regime), based on the flags set by ***Alntest***.

Newt solves the set of simultaneous non-linear equations needed to find the values of the master variables, using the Newton-Raphson method.

Fx gives the fitting functions for ***Newt***. The subdivisions are labelled ***Fxy*** where y is a value appropriate to that of ***Nw***. The subroutine generates a variable designated ***Fx*** (returned to ***Newt***), which gives the fractional error in the stoichiometry for whichever element is being considered at the moment. Since ***Fx*** calls on ***Spec***, mole numbers of all species are found each time.

Diff first computes all the master variables which were not found in ***Fx*** and then calls ***Spec*** to find all the mole numbers of the gaseous species. ***Diff*** computes mole numbers of the condensed species; and finally the errors in the material balances for O, N, Al, and C are returned to ***Fx***.

Spec computes the mole numbers of all gaseous species, given the current values of X, Z, Alv, and ***Acc***. $Y = \sqrt{H_2}$ is computed in closed form in ***Spec***; it is needed for computations on the hydrogen-containing species. For each species the mole number is computed by using appropriate values of K_n, the formation constant.

DIAGNOSTIC SUBROUTINES

Sum1 is called at the end of each run to give first a check on the material balance in each element and then, at ***Testk***, a comparison of the computed amounts with the equilibrium constants of formation of each condensed phase and of several key gaseous species.

Printt is called whenever special function key ***k0*** has been depressed once. At each emergence from ***Newt*** the mole numbers of all species are given in a condensed table, this is followed by ***Sum1*** and ***Testk***, which give output as previously described. These various checks are especially valuable in troubleshooting.

Depressing special function key *k1* will cause execution of TRACE VARIABLES Yn(*). Then during each iteration in *Newt* the relative error functions, used to test convergence, will be displayed. Each Yn(J) must drop below 0.0001 in absolute value before convergence is realized. This feature may be turned off by executing NORMAL.

Appendix A
PROGRAM *INAL*

This program is written in Hewlett-Packard's HP-BASIC, and is intended for use on any of the HP 9845 series computers. In addition to the main program, the data file "THD" and the key file "INEK" are needed for operation.

KEY FILE *INEK*

```
KEY 0
    Check=(Check=0)
   -Execute

KEY 1
    TRACE VARIABLES Yn(*)
   -Execute

KEY 2
    CONT Er
   -Execute

KEY 3
    RE-STORE "
   -Left arrow
   -Insert character

KEY 4
    CONT Temp
   -Execute

KEY 5
    TRACE PAUSE Iiset
   -Execute

KEY 6
     TRACE PAUSE 5
   -Execute

KEY 7
     7,5,3,6
   -Continue

KEY 8
         CONT Fuel
   -Execute

KEY 9
   -Clear line
    LOAD

KEY10
   -Clear line
    SAVE

KEY11
   -Clear line
    STORE

KEY12
   -Clear line
    EDIT

KEY13
   -Clear line
    EDIT LINE

KEY14
   -Clear line
    LIST

KEY15
   -Clear line
    SCRATCH

KEY16-Undefined
KEY17-Undefined
KEY18-Undefined
KEY19-Undefined
KEY20-Undefined
KEY21-Undefined
KEY22-Undefined
KEY23
     -11750
   -Continue

KEY24-Undefined
KEY25-Undefined
KEY26-Undefined
KEY27-Undefined
KEY28-Undefined
KEY29-Undefined
KEY30-Undefined
KEY31-Undefined
```

```
10   !                                        FILE NAME "INAL"        18 FEB 83   Adiabatic overpressure calculation
20   !                                                                            Constant volume
30   !
40   OPTION BASE 1
50   OVERLAP
60   ASSIGN @3 TO "THD"
70   RLAD @3,1
80   INTEGER Alcflag,Alnflag,Aoflag,Lgflag,Grfl,Aln0,Af0,Al0,Grf0,I,J,Ii,Nw,Cv,Ccount,G(34)
90   SHOPT Ac,Ah,An,Aoq,Acp,Ahp,Anp,Aop,Fms,Mo,Mt,V,f,Deltfactor
100  DIM Xn(5),Yn(5),Yt(5),Del(5),D(5,5),In(5,5),Nold(34),Ch$(2),Ns(2)
110  DIM N(34),Ak(34,6),K(34),Bu(34,5),Fo$(34)[8],P(34)
120  MAT G=CON                                                                    Index for gases
130       FOR I=12 TO 32                                                          Condensed phases
140       G(I)=0
150  NEXT I
160  G(28)=0
170  DEF FNQd(A,B,C)=(-B+SQR(B^2-4*A*C))/2/A
180  DEF FNSal=Alv+Alh+Alch+2*Al2o+Alc+Alol+Alp                                   Sum gaseous Al atoms
190  DEF FNSo=Alch+Al2o+Co+Oh+H2o+No+O+Hnco+Hco+Ch2o+Alo+2*(Co2+O2+Alo2)          Sum gaseous O
200  DEF FNRat=3*(FNSal+Alo)/(2*Alc+2*FNSo)                                      FNRat = Alpha-1Lg - excess Al over O
210  DEF FNGrf=MIN(1,INT(Acc))

220  READ @3;Ak(*),Bu(*),Fo$(*)
230  LOAD KEY "INKEY"
240  PRINTER IS 7,1
250  DISP "FOR ALL N(I) PRESS K0:       FOR TRACE VARIABLES Yn(*): K1"
260  PAUSE
270   A$="Al"

280  Name: !
290  Fuel:  INPUT "NAME OF COMPOUND:",N$
300   INPUT "FORMULA, AS ATOMS PER MOLE OF C H,N,O, IN ORDER (use k7 for TNT)",Ac,Ah,An,Aoq
310   IF Ac=0 THEN Ac=1E-10
320   IF Ah=0 THEN Ah=1E-10
330   INPUT "Internal energy of formation, J/mol (use SHIFT17 - k23 FOR tnt)",Um
340   Fms=Ac*12.0115+Ah*1.008+An*14.0067+Aoq*15.9994                             ! Formula mass
350   IF Fms=0 THEN Fms=1E-10
360   !

370  Conc:   INPUT "MASS OF METAL, IN KG PER CU M OF AIR",Mm
380          INPUT "MASS OF FUEL IN KG/CU M",Mf
390          Lg=N(19)=Aln=N(20)=Ao=N(21)=Gr=N(23)=Alc=N(28)=0
400          Alcflag=Alnflag=Cv=Nw=Grfl=Ccount=Ii=0                                 as:  Alnflag FOR AlN, Alcflag FOR Al4C3;
410  !                                                                                   Alcflag = 1 for exc C ; = 2 for exc Al
420  !                                                                                   Aoflag for Al2O3; Lgflag for Lg;
430  !                                                                                   Grfl for Gr absent, =2 for Gr present.
440  !                                                                                   Cv = counter for Newton's iterations
450  !
460          Acp=PROUND(Ac,-2)
470          Ahp=PROUND(Ah,-3)
480          Anp=PROUND(An,-3)
490          Aop=PROUND(Aoq,-3)
500          Q=1
510  Conc1:    PRINT USING Imc;Mm,M$,Mf,Ms,N$,Acp,Ahp,Anp,Aop,"Mf/Mm",K*Q K2Y"C/M - ",DDD2.DDD..'
520  Imc:  IMAGE /,/,"KG OF ",K,2X,K,"KG OF ",K,"C ",K,"H ",K,"N ",K,"O",K2X"C/M - ",DDD2.DDD..'
530          Nal=Mm/.026982          ! MOLES OF Al
540  Ex1:   Mo=1000*Mf/Fms
550          IF Mols=0 THEN Mols=1E-10
560          Fms=PROUND(Fms,-3)
570          Mo=PROUND(Mols,5)
580          Mt=PROUND(Nal,5)
590          PRINT Mt;" moles of metal"
600          PRINT "Formula mass is ";Fms;"g/mole.  Present are ";Mo;" moles"
610          PRINT "Volume = ";V;" cu M"
620          N(6)=.4036
630            Nc=Ac*Mols
640            Nh=Ah*Mols
650            Nn=An*Mols+62.96                                                     Includes moles of N atoms from air
660            Noq=Aoq*Mols+16.951                                                  Includes moles of O atoms from air
670            PRINT "U0 = ";Um;" J/mol",LIN(1)                                   ! Initial total internal energy, in Joules
680            U0=Um*Mols
690            Ti=U1=0 !
```

```
700 Temp:  INPUT "Approximate temperature ?",T                          ! To be used in Cv and Cp calc.
710        Delfactor=Delvfactor=0                                       ! By repeating "Approx"
711        INPUT "Do you want to repeat 'Approx'?(Y\N)",Nw$ !           ! it may be possible to avoid
712        IF UPC$(Nw$)<>"Y" THEN Calc!                                 ! large excursions in running "Newt"
713        Nw$="N" !
720 !
730 Calc:  GOSUB Eq
740        DISP "T = ";T;
750        GOSUB Alcal !                                                ! Subroutine calculates mole numbers
760                                                                     !   and residual DelU.
770        D$=" T high  "
780        IF Du<0 THEN D$=" T low  "
790        DISP D$
800        FIXED 0
810        PRINT "T = ";T;" K","DelU = ";Du;" J";LIN(2)
820        DISP "T = ";T;" K";SPA(5);"DelU = ";Du;" J";
830        DISP D$
840        STANDARD
850        IF Du=0 THEN Ex !                                            ! When energy balance is at last achieved
860        T2=T1
870        U2=U1
880        T1=T
890        U1=Du
900        IF T2=0 THEN Temp
910        DISP "TO EXIT, PRESS K2;  MANUAL TEMPERATURE: K4; INTERPOLATE: CONT"
920        PAUSE
930        T=T2-U2*(T2-T1)/(U2-U1)      !                               ! linear interpolation
940        GOTO Calc
950 !
960
970 !

980 !

990 Molsum:   Nsum=Nt=0
1000          FOR I=1 TO 34
1010              Nt=Nt+N(I) !                                          ! Total moles
1020              Nsum=Nsum+G(I)*N(I)  !                                ! Moles of gas
1030          NEXT I
1040          RETURN !


1050 Ex: GOSUB Molsum !                                                 ! Moles of gas
1060        Ng=Nsum  !                                                  ! Total mass in grams
1070        Mass=(Mm+Ms+1.1692)*1000 !                                  ! Mass of gaseous products
1080        Masg=Mass-26.9815*N(19)-40.99*N(20) 101.96*(N(21)+N(22))-12.011*N(23)-143.96*N(28)  !
1090        M_g=Masg/Ng                                                 ! Avge molar mass of gases
1100        M_t=Mass/Nt                                                 ! Avge molar mass of all products
1110        PRINT USING Imcon;T
1120 Imcon: IMAGE /"T = ",4D.D,"K"
1130        FIXED 3
1140        PRINT "Avg Molar mass(gases) = ";M_g;" g/mol";SPA(2);"Avg Molar mass(tot..) = ";M_t;" g/mol";SPA(5);Ng;" moles of GAS ";Nt;
" moles TOTAL";LIN(1)
1150        P=8.3143E-5*Ng*T/V                                          !  nRT/V  value in bars

1160        PRINT LIN(1);"Overpressure = ";P-1;" bars";LIN(2) !         Output
1170        STANDARD
1180        PRINT "Species";SPA(8);"Moles";SPA(8);"Mole %";SPA(4);"Part. pres.";SPA(4);"LGT(X)";SPA(6);"LGT(P)";LIN(1)
1190        FOR I=1 TO 34
1200            IF G(I)=0 THEN Conc5 !                                  ! Skips condensed phases
1210            IF N(I)<=0 THEN N(I)=1E-90
1220            PRINT USING Imcon2;Fo$(I),N(I),100*N(I)/Ng,N(I)*P/Ng,LGT(N(I)/Ng),LGT(N(I)*P/Ng)
1230 Imcon2: IMAGE 8A,5X,Z.3DE,5X,2D.4D,5X,3D.4D,5X,MD2.3D,5X,MD2.3D
1240 Conc5: NEXT I
1250        FOR I=19 TO 23
1260            IF N(I)=0 THEN Conc6
1270            PRINT USING Imcon1:Fo$(I),N(I)
1280 Imcon1: IMAGE 8A,5X,Z.3DE
1290 Conc6: NEXT I
1300        IF N(28)>0 THEN PRINT USING Imcon1:Fo$(28),N(28)
1310        Delfactor=.001                                              !  dT /(1+Delfactor)*T
1320 !
```

```
1330 Ceev:!
1340       U1=Du
1350       T1=T    !                                                    Saving the original values
1360       MAT Nold=N
1370       T=(1+Deltfactor)*T1
1380       GOSUB Eq
1390       GOSUB Alcal
1400       GOSUB Molsum
1410       U2=Du                                                        !  For Cv calcn.
1420 Ceev2: PRINT "Deltfactor=";Deltfactor
1430        IF Check=1 THEN GOSUB Printt
1440        FOR I=1 TO 28
1450            IF G(I)=1 THEN Ceev1
1460            IF (N(I)<>0) EXOR (Nold(I)=0) THEN Ceev1                 ! Either both 0 or both >0
1470            PRINT Fo$(I);" CHANGES FROM ";Nold(I);" TO ";N(I);SPA(2)
1480 Ceev1: NEXT I
1490        T=T1
1500        Dn_dt=(Nsum-Ng)/Deltfactor/T                                 !  Const V
1510        Ceev=(U2-U1)/Deltfactor/T                                    !  Total Cv
1520 !
1530 Ceep: Lambda=1+(Ng+T*Dn_dt)*8.314/Ceev
1540       Ceev=Ceev/Mgas                                               !  per unit mass
1550       IF Check=0 THEN Ceep2
1560       FIXED 7
1570       PRINT "dn/dT=";Dn_dt,"Cv=";Ceev,LIN(1),"n'df)-";Nsum,"N(old)=";Ng,LIN(1),"U1=";U1,"U2=";U2
1580       STANDARD!
1590 Ceep2: Branch=3
1600        IF Deltfactor<0 THEN Ceep3
1610        C$=" "
1620        INPUT "DO YOU WANT TO REPEAT FOR dV AND dT NEGATIVE (Y/N)?",C$
1630        IF C$<>"Y" THEN Ceep3
1640        Deltfactor=-.001
1650        GOTO Ceev2
1660 Ceep3: Deltfactor=0
1670        PRINT USING Imce;Lambda,Ceev
1680 Imce: IMAGE /"Lambda = ",D.3D,2X,"Cv = ",3D.3D," J/K/Kg"
1690       PRINTER IS 7,1
1700       GOSUB Sum1
1710       PRINT USING Imc;Mn,M$,Ms,N$,Acp,Ahp,Anp,Aop,Ms/Mn
1720       PRINT USING Imcf;T,P-1.0
1730 Imcf: IMAGE "T = ",2D.D,"   Overpressure = ",3D.3D," bars",  "   Volume = ",3D.3D," cu m"
1740       PRINT LIN(4);PAGE
1750       LINPUT "DO YOU WISH TO RUN ANOTHER CONCLN/RATION FOR THIS SAME METAL-FUEL PAIR (Y/N)?",C$
1760       IF C$="Y" THEN Conc
1770       LINPUT "DO YOU WISH TO RUN A DIFFERENT FUEL WITH THE SAME METAL (Y/N)?",C$
1780       IF C$="Y" THEN Name
1790       PRINT "END OF ROUTINE"
1800 STOP
1810 !
1820 Eq:                                                                ! Computes Kn FROM LGT(Kp) parameters
1830 !                                                                  Kn is equil. const. of formation from elements,
1840 !                                                                  expressed in mole numbers
1850     V_rt=12188*V/T  !                                              V/RT FOR CONST VOL = V CU M.
1860     FOR I=1 TO 34
1870     Lgk=Ak(I,1)+Ak(I,2)/(T+Ak(I,3))+Ak(I,4)*T
1880     Lgk=Lgk+Ak(I,6)*LGT(V_rt)   !                                  Conversion from Kp (expressed in pressures) to Kn.
1881     IF (I=22) AND (T<10000) THEN Lgkao=Lgk                         At low T, avoids overflow in Kao
1890     IF Lgk>99 THEN Lgk=99                                          Ak(I,6) is delta n (gases) for formation reaction.
1900     K(I)=10^Lgk                                                    Ak(I,5) not used
1910     NEXT I
1920     RETURN
1930 !
```

```
1940 ! ************************************************************************
1950 Alcal: PRINT LIN(2)!****************    SUBROUTINE ALCAL    ****************
1960 !                                                                          Will find all mole numbers and then compute
1970 !                                                                          energy balance
1980       Kac=K(21)
1990       IF T(2315 THEN Kac=K(22)
2000       IF Nw()0 THEN Hem                                                  ! Nw set = 0 whenever a new concentration is run
2015       Alcflag=0
2030       Cor=0
2030       Alpha=3*Na1-2*Nac    !                                             Excess Al over O atoms
2040       Grf1=0
2050       Kac=K(21)
2060       IF T(2315 THEN Kac=K(22)
2070       Nec=Nc
2080       Nhc=Nh
2090       Nnc=Nn!

2100 Approx:                                                                  ! Initial approximations
2110       Aaflag=Aa=0
2120       IF Na1)Nc THEN Nagr                                                Graphite assumed present when C in excess over O
2130       Ca=Nac !
2140       Grf=Nc-Nac
2150       Accr1
2160       X=Ca/Acc/K(7)!                                                     Equil. with CO and Acc
2170       H2=.5*Nh
2180       Y=SQR(H2)
2190       Alnflag=1
2200       IF Nn)Na1 THEN Excn                                                ! AlN assumed present
2210       Aln=Nn                                                             Sumal is total Al, corrected for AlN and Al4C3
2220       Sumal=Na1-Aln !
2230       IF Alcflag=0 THEN Appr1                                             Excess Al case: a phase rule requirement
2240       Alcflag=2  !
2250       Grf=Acc=0
2260       Alc=(Nc-Nac)/3                                                     ! Excess carbon case
2270       Sumal=Na1-Nn *Alc
2280       IF Sumal)0 THEN Appr1
2290       Alcflag=Accr1
2300       Alc=(Na1-Nn)/4
2310       Grf=Nc-Nac-Alc/3
2320       Sumal=SQR(SQR(1/K(20)))  !                                         Make r Alv is equil. with Gr and Al4C3
2330 Appr1: W=Alv=MIN(Sumal,1/K(10))                                          Equil. with Al4C3 and Alv
2340       IF Alcflag=2 THEN Acc=(1/K(28)/Alv^4)^(1/3)!
2350       Lq=N(19)*Sumal Alv                                                 Equil. with AlN and Alv
2360       Laflag=(Lq)0)
2370       Z=1/Alv/K(20)!                                                     An exit from Approx
2380       N2=Z*Y
2390 Prep1: Nw=5+Laflag!
2400       Iir3-Iaflag-FNGrf
2410       GOTO Hem
2420 Excn: !                                                                  Nitrogen in excess over Al
2430       Aln=Na1
2440       N2=.5*(Nn-Aln)
2450       Z=SQR(N2)
2460       W=Alv=1/Z/K(20)!                                                   Equil. with AlN and N2
2470       Laflag=0  !                                                        Liquid metal assumed absent
2480       GOTO Prep1
2490 Nagr:Afactor=1+2*(T(2600)      !                                         Allows for alumina at lower T              ! Graphite as
sumed absent
2500       IF Nac Nc)Afactor*Na1/2 THEN Nagain
2510       Grf=0
2520       Cor=Nc
2530       Al2o=(T)=2600)*(Nac Nc)!                                           In this regime Al-O is assumed entirely  Al2O(g)
2540       Aa=(T(2600)*(Nac-Nc)/3 !                                           above 2600 and Al2O3 below
2541       Aaflag=(Aa)0)
2550       H2=Nh/2
2560       Y=SQR(H2)
2570       Alnflag=1
2580       IF Na)Afactor*Na1-2*(Nac-Nc) THEN Excn2
2590       Aln=Nn!                                                            Excess of Al over N
2600       Sumal=Na1-2*(Al2o+Aa)-Aln
2610       W=Alv=MIN(Sumal,1/K(19))
2620       Lq=Sumal-Alv
2630       Laflag=(Lq)0)
2640       Z=1/Alv/K(20)
2650       GOTO Nagr2!
```

```
2660 Excn2: Aln=Afactor*Nal-2*(Noa-Nc)!                    Excess of N over Al
2670        N2=(Nn-Aln)/2
2680        Z=SQR(N2)
2690        Alv=W=1/Z/K(20)
2700        Lqflag=Lq=0
2710 Negr2: X=Al2o/Alv^2/K(4)
2711        IF T(2600 THEN X=Alv^(-2/3)*Kao^(-1/3)
2720        Acc=Cc/X/K(7)
2730        GOTO Prep1
2740 Noaln:                                                ! AlN absent, alumina present
2750        Nw=1
2760        Aln=Lq=Cr=Alnflag=Lqflag=0
2770        Aoflag=1
2780        Ii=3
2790        N2=Nn/2
2800        Z=SQR(N2)
2810        IF Noa-Nc)1.5*Nal THEN Water                   H2O assumed absent
2820        Cc=Nc!
2830        Al2o=(Nc-Noa+1.5*Nal)/2                        Alumina and Al2O both at T\2600
2840        Ao=(Noa-Nc-Nal/2)/2  !                         ! These two eqns satisfy the Al and O balances
2850                                                       Equil. with Al2O and alumina
2860        O2=K(4)/Kao/Al2o!
2870        X=SQR(O2)                                      Equil. with CO and O2
2880        Acc=Cc/X/K(7)!
2890        H2=Nh/2
2900        Y=SQR(H2)                                      Equil. with O2 and Al2O
2910        Alv=W=SQR(Al2o/X/K(4))!
2920        GOTO Hom
2930 !

2940 Water: Ao=Nal/2!                                      H2O present
2950        IF Noa-Nc\1.5*Nal+Nh/2 THEN Carbdi             CO2 absent
2960        Cc=Nc!
2970        H2o=Noa-(Nc+1.5*Nal\
2980        H2=Nh/2-Noa+Nc+1.5*Nal                         These 2 eqns satisfy the H and O balance.
2990 !                                                     Equil. of O2 with H2O and H2
3000        X=H2o/H2/K(12)!                                Equil. with CO and O2
3010 Prep2: Acc=Cc/X/K(7)!                                 Equil. with O2 and alumina
3020        W=Alv=SQR(1/X^3/Kao)!
3030        GOTO Hom
3040 !

3050 Carbdi: H2o=Nh/2!                                     CO2 and H2O present
3060        IF Noa)1.5*Nal+Nh/2 THEN Oxy                   O2 assumed absent
3070        Co2=Noa-Nc-1.5*Nal-Nh/2!                       C balance
3080        Cc=Nc-Co2!                                     Equil. with CO and CO2
3090        X=K(7)*Co2/Cc/K(8)!                            Prepare exit from Approx
3100        GOTO Prep2!
3110 !

3120 Oxy:   Co2=Nc!                                        CO absent
3130        O2=(Noa-Nc-1.5*Nal-Nh/2)/2!                    Except O2
3140        X=SQR(O2)
3150        Acc=Co2/O2/K(8)!                               Equil. with CO2 and O2
3160        W=Alv=SQR(1/X^3/Kao)!                          Equil. with O2 and alumina
3170        GOTO Hom!                                      End of approximations
3180 !
```

```
3190 ! ##########################################################################
3200 ! ##########################################      HOM     ##################
3210 Hom0: GOSUB Alntest !
3220 !
3230 Hom:!                                                                        Set-up for Newton's method calculations
3240 !                                                                           Subdivisions of Hom are designated Nw with
3250 !                                                                           subscript, for the appropriate value
3260 !                                                                           value of the index Nw, identifying the regime.
3270 !                                                                           These routines must set all the flags for
3280 !                                                                           condensed phases, establish the number (Ii)
3290 !                                                                           and nature of the variables, and set initial
3300 !                                                                           approximationsfor the variables which are the
3310 !                                                                           parameters found by Newt.
3320      Grfl=1+FNGrf  !                                                         Normally 1, to consider graphite initially absent
3330      IF Alcflag)0 THEN Grfl=2
3340      GOSUB Nwset
3350      GOSUB Iiset
3360 !
3370 Hom1: ON Nw GOTO Nw1,Nw2,Nw3,Nw1,Nw1,Nw2,Nw7,Nw8,Nw9,Nw10,Nw11 !

3380 Hom2: IF (Alv^48Acc^32K(28)-1),001) AND (Alcflag=0) THEN Hom3!               Al4C3 found but presumed absent
3390      ON Nw GOTO Exit,Exit,Exit1,Exit1,Exit1,Exit1,Exit,Exit,Exit,Exit,Exit !

3400 Hom3: Alcflag=1  !                                                          For Al4C3 present, excess carbon
3410      IF Gr(L6+Alv THEN Alcflag=2                                            ! Alcflag=2 for Al4C3 present, excess Aluminum
3420      GOTO Aeprex!                                                           Entire process must be repeated

3430 Nw1: REDIM Xn(Ii)!                                                           Ii, number of variables, will have
3440 !                                                                            been given before entering Hom
3450 Nw1d: Xn(1)=X                                                               ! Variables are X,Z,(Acc);  Nw=1,4,5
3460      Xn(2)=Z
3470      IF Grfl=1 THEN Xn(3)=Acc!                                              Grfl=2 means Gr present, so that Acc=1
3480      GOSUB Newt
3490      GOSUB Alntest!                                                         Includes Iiset
3500      IF (Alnflag()Aln0) OR (Aeflag()Af0) OR (Lqflag()Al0) THEN Hom1!        If any changes, recompute
3510      IF Grf0=2 THEN Nw1b
3520      IF Acc(=1 THEN Nw1b                                                    If Acc were found )1, switch to
3530      Acc=1 !                                                                graphite-present case
3540 !
3550      Grfl=2
3560      GOSUB Iiset
3570      GOTO Nw1 !

3580 Nw1b: X=MIN(Xn(1),SQR(Neg/2))!                                              Avoids material balance errors
3590      Z=MIN(Xn(2),SQR(Nn/2.)
3600      IF (Lqflag=0) OR (Lq)() THEN Nw1c                                      Since Lq=0 but Lqflag was assumed=1;
3610      Lqflag=0  !                                                            when the preceding test fails
3620 !                                                                           Start over
3630      GOTO Hom3 !

3640 Nw1c: IF X()Xn(1) THEN Nw1d                                                 In these two cases, repeat iterations
3650      IF Z()Xn(2) THEN Nw1d!
3660      IF Grfl=1 THEN Acc=Xn(3)
3670      GOTO Nw2
3680 !

3690 Nw2: REDIM Xn(Ii) !                                                         Variables are X(Acc); Nw=2,6: AlN and Al2O3 present.
3700 Nw2a: Xn(1)=X
3710      IF Grfl=1 THEN Xn(2)=Acc
3720       GOSUB Newt
3730      GOSUB Alntest                                                          Recompute if any change
3740      IF (Alnflag()Aln0) OR (Aeflag()Af0) OR (Lqflag()Al0) THEN Hom1!
3750      IF Grf0=2 THEN Nw2c
3760      IF Acc(=1 THEN Nw2c                                                    Gr was assumed absent but found )1
3770      Acc=1!
3780      Grfl=2
3790      GOSUB Iiset
3800      GOTO Nw2 !

3810 Nw2c: X=MIN(Xn(1),SQR(Neg/2))!                                              Avoids material balance errors
3820      IF (Nw=2) OR (Lq)0) OR (Alcflag=1) THEN Nw2d                           For Nw=6 it had been assumed = 1
3830      Lqflag=0  !
3840      GOSUB Nwset                                                            Recompute
3850      GOSUB Iiset
3860      GOTO Hom  !

3870 Nw2d: IF X()Xn(1) THEN Nw2a!                                                Start over
3880      IF Grfl=1 THEN Acc=Xn(2)                                               Prepares exit
3890      GOTO Hom2!
3900 !
```

```
3910 Nw3: REDIM Xn(11) !                                  Nw=3;  variables are X,Z,W (Acc);
3920 !                                                    no condensed phase other than possible Gr
3930 Nw3a: Xn(1)=X
3940       Xn(2)=Z
3950       Xn(3)=W
3960       IF Grf1=1 THEN Xn(4)=Acc
3970 Nw3b: GOSUB Newt
3980       GOSUB Alntest
3990       IF (Alnflag()Aln0) OR (Aaflag()Af0) OR (Lqflag()Al0) THEN Hem1!     Restart if any change
4000       IF Grf0=2 THEN Nw3c
4010       IF Acc(=1 THEN Nw3c
4020       Acc=1!                                         For this case, Gr was assumed absent, but Acc found )1
4030       Grf1=2
4040       I1=3
4050       GOTO Nw3 !

4060 Nw3c: X=MIN(Xn(1),SQR(Nen/2))
4070       Z=MIN(Xn(2),SQR(Nn/2))!                        These three lines are to avoid material balance problem
4080       W=Alv=N(1)=MIN(Xn(3),Na1)!
4090
4100       IF (X()Xn(1)) OR (Z()Xn(2)) OR (W()Xn(3)) THEN Nw3a!     If any change, then recompute
4110       GOTO Hem2 !                                    to exit

4118 Nw7: REDIM Xn(11)  !                                 Variables are Z,(Acc).  Nw=7;
4120 !                                                    Alumina + Lq or (AlC + excess Gr).
4130       Xn(1)=Z
4140       IF Grf1=1 THEN Xn(2)=Acc
4150       GOSUB Newt
4160       GOSUB Alntest
4170       IF (Alnflag()Aln0) OR (Aaflag()Af0) OR (Lqflag()Al0) THEN Hem1!     If any change then recompute
4180       IF (Grf0=2) OR (Acc(=1) THEN Nw7b                  Gr had been assumed absent, but Acc found )1
4190       Acc=1!
4200       Grf1=2
4210       GOSUB I1
4220       GOTO Nw7 !

4230 Nw7b: Z=MIN(Xn(1),SQR(Naa/2))!                       Avoids material balance errors
4240       IF (Alcflag=1) OR (Lq?0) THEN Nw7c             !  Since Lq=0 but Lqflag was assumed = 1
4250       Lqflag=0
4260       GOTO Hem0 !

4270 Nw7c: IF X()Xn(1) THEN Nw7!                          recompute
4280       IF Grf1=1 THEN Acc=Xn(2)
4290       GOTO Hem2
4300 !

4310 Nw8:  REDIM Xn(1) !                                  Xn(1)=Acc, Nw=8;
4320 !                                                    Alumina, AlN and Lq (or AlC + Gr) present
4330       Gr=0
4340       Xn(1)=Acc
4350       GOSUB Newt
4360       IF Xn(1)(1 THEN Nw8a
4370       Acc=1!                                         Gr had been assumed absent, but Acc found )1
4380       Grf1=2
4390       GOTO Nw9 !

4400 Nw8a: Acc=Xn(1)
4410       GOTO Hem2!

4420 Nw9: !                                               Nw=9;  Solution in closed form;
4430 !                                                    Alumina, AlN, Gr and Lq or AlC present
4440       DISP "Nw=9"
4450       Acc=1
4460       IF Alcflag=2 THEN Acc=(1/K(2H)/Alv^4)^(1/3)!   Equil. with Alv and Gr
4470       GOSUB Diff
4480       GOSUB Alntest
4490       IF (Alnflag()Aln0) OR (Aaflag()Af0) OR (Lqflag()Al0) THEN Hem1!     If any change, then recompute
4500       GOTO Hem2
4510 !
```

```
4520 Nw10:  REDIM Xn(1)  !                                  Variable is Alv; Nw=10; Alcflag=2:
4530 !                                                      Alumina, Aln, AlC plus excess Al
4540        Gr=0
4550        Xn(1)=Alv
4560        GOSUB Newt
4570        GOSUB Alntest
4580        IF (Alnflag()Aln0) OR (Agflag()Af0) OR (Lqflag()Al0) THEN Hem1!      If any change then recompute
4590        W=Alv-N(1)*MIN(Na1,Xn(1))!                      Avoids material balance errors
4600        IF W()Xn(1) THEN Nw10
4610        GOTO Hem2 !

4620 Nw11:  REDIM Xn(2)!                                    Variables are Alv,Z;  Alcflag=2; Nw=11:
4630 !                                                      Alumina, AlC plus excess Al
4640        Gr=0!                                           Phase rule requirement
4650        Xn(1)=Alv
4660        Xn(2)=Z
4670        GOSUB Newt
4680        GOSUB Alntest
4690        IF (Alnflag()Aln0) OR (Agflag()Af0) OR (Lqflag()Al0) THEN Hem1!      If any change then recompute
4700        G=Alv-N(1)*MIN(Na1,Xn(1))                       These two to avoid material balance problems
4710        Z=MIN(Xn(2), SQR(Na/2))!
4720        IF (Alv()Xn(1)) OR (Z()Xn(2)) THEN Nw11
4730        GOTO Hem2
4740 !
```

```
4750 ! ************************************************************************
4760 Exit: ! ***************      EXIT FROM  Alcal     ***************************
4780       IF Ao)0 THEN Exit1
4790       Aoflag=0!
4800 !
4900       GOTO Hem4 !                                                    Alumina had been assumed present
                                                                          in the cases reached here
4910 Exit1: Gr=N(23)=Ncc-Co-Co2-3*Alc!
4920       IF (ABS(Gr)(1E-48Ncc) OR (Acc(1) THEN Gr=N(23)=0               The standard exit from Alcal
4932       Grfl=FNGrf+1
4938       IF (Gr)=0) OR (Ac=1E-10) THEN Exit2
4940       PRINT "GRAPHITE ABSENT"!                                       Gr(0 had been found
4950       Gr=N(23)=0
4951       Grfl=1
4960       IF Nw(9 THEN Exit5
4970       GOSUB Nyset
4980       GOSUB Yiset
4990       GOTO Nw8 !

5000 Exit5: Grfl=1
5010       GOSUB Car3
5020       I1=I1+1
5030       GOTO Hem!

5040 Exit2: IF ABS(Alc)(1E-48Ncc THEN Alc=N(28)=0                         Recompute
5050       IF ABS(Aln)(1E-48Nnc THEN Aln=N(28)=0
5060       IF Ao)0 THEN Exit3
5070       N(21)=N(22)=0 !                                                For the case of no alumina
5080       GOTO Energy !

5090 Exit3: N(21)=Ao*(T)2315)          !                                  Liquid alumina case
5100       N(22)=Ao-N(21)
5110       !

5120 Energy:!
5130 !                                                                    Computes the change in internal energy,
5140       Du=-U0                                                         dU = U(T) - U0, computed for entire system
5150       FOR I=1 TO 34
5160           Ui=Du(I,1)+Du(I,2)*T+Du(I,3)*T^2+Du(I,5)/T+Du(I,4)*LOG(T)   ! Molar int energy for component I
5170           Du=Du+Ui*N(I)
5180       NEXT I
5190           IF (T()2315) OR (Du(=0) THEN Exit6
5100           Sol=Du/118407                                              !  Delta U divided by molar energy of fusion
5110           Liq=Ao-Sol
5120           IF Liq(0 THEN Sol                                          Where both liq and sol alumina are present
5130           Du=0 !
5140           N(21)=Liq
5150           N(22)=Sol
5160           GOTO Exit6
5170 Sol:  N(21)=0 !                                                      The case where only solid is present
5180 !                                                                    and T=2315 is too high
5190       N(22)=Ao
5200       DISP "SELECT A LOWER TEMPERATURE"
5210       BEEP
5220 Exit6: IF Nw=9 THEN PRINT "Nw=9"
5230       IF Nw=9 THEN GOSUB Prefig
5240       IF Alcflag(0 THEN PRINT "Alcflag=";Alcflag
5250       RETURN !
5260 !                                                                    End of Alcal
```

```
5270 ! ***********************************************************************
5280 ! ************************    IISET, NWSET, ALNTEST    ********************
5290 Iiset: Iiv4-Aeflag-Lqflag-Alnflag-FNGrf!
5300       IF Alcflag=0 THEN Nwset
5310       Iiv2-Aeflag-Alnflag
5320       IF Alcflag=2 THEN Iiv3-Aeflag-Lqflag-Alnflag
5330 !
5340 Nwset: !
5350       IF (Lqflag=1) OR (Alcflag=1) THEN Nwset1
5360       Nw=2-2*Aeflag+(2-Aeflag)*Alnflag
5370       IF (Nw=2) AND (Alcflag=2) THEN Nw=10
5380       IF (Nw=1) AND (Alcflag=2) THEN Nw=11
5390       RETURN
5400 Nwset1: Nw=4+2*Alnflag
5410       IF Aeflag=0 THEN RETURN
5420       Nw=7
5430       IF Alnflag=0 THEN RETURN
5440       Nw=8
5450       IF (FNGrf=1) OR (Alcal=1) OR (Alcflag*Lqflag=2) THEN Nw=9
5460       RETURN !

5470 Alntest: Aln0=Alnflag!
5480 !
5490       Al0=Lqflag
5500       Af0=Aeflag
5510       Grf0=Grf1
5520       Lqflag=(Alv#K(17)-1)-.001)
5530       IF (Lq<0) OR (Alcflag=1) THEN Lqflag=0
5540       Grf1=1+FNGrf
5550       IF Ac>1E-10 THEN Grf1=0
5560       IF Alcflag>0 THEN Grf1=2
5570       Alnflag=(Alv#Z#K(20)-1)-.001)  !
5580       IF Aln<0 THEN Alnflag=0
5590       IF T>=1000 THEN Aeflag=(Alv^2*X^3*Kaq-1)-.001)  !
5600       IF T<1000 THEN Aeflag=(2*LOG(Alv)+3*LOG(X)+Lqkaq)-.001)
5601       IF Ae<0 THEN Aeflag=0 !

5620       GOSUB Iiset
5630       RETURN !
```

Sets number of variables for Newt
For Alcflag=1

Sets index to identify routine in Newt

! FOR Aeflag=0, Lqflag OR Alcflag=1

! Acc the variable

Sets flags for various condensed species
Formation constant must be exceeded and
and amount of phase must not be <0

-1 if Aln is present

-1 if Al2O3 is present
Using logs at low T where overflow is likely

```
5640 ! ************************************************************************
5650 Newt:  ! ********************    NEWT    ********************************
5660 !                                                                          A Newton's method for up to
5670 !                                                                          five simultaneous non-linear eqns
5680      REDIM Xn(I1),Yn(I1),Xt(I1),Delx(I1),D(I1,I1),In(I1,I1)
5690      DISP TAB(1);"NEWTON'S CALC., Nw = ";Nw;SPA(5);
5700      IF Aeflag THEN DISP "Ae";
5710      IF Alnflag THEN DISP "AlN";
5720      IF Lnflag THEN DISP "Ln";
5730      IF Grflg=2 THEN DISP "GP";
5740      IF Alcflag THEN DISP "Alcflag =";Alcflag
5750      Cv=0   !                                                              Counter for Newton's iterations
5760 Newt3:  MAT Xt=Xn                                                         Each Xt(J)=Xn(J), the trial value
5770      Cv=Cv+1
5780      FOR I=1 TO I1
5790        GOSUB Fx
5800        Yn(I)=Fx
5810        FOR J=1 TO I1
5820            Xt(J)=.99*Xn(J)    !                                           dX=-.01 X
5830            GOSUB Fx
5840            D(I,J)=-(Fx-Yn(I))/Xn(J)/.01     !                          Partial of Yn(I) wrt Xn(J), found by
5850 !                                                                         evaluating at Xn(J) and .99 Xn(J)
5860            Xt(J)=Xn(J)
5870        NEXT J
5880      NEXT I
5890      IF DET(D)=0 THEN Mater
5900      MAT In=INV(D)
5910      MAT Delx=In*Yn
5920      MAT Xn=Xn-Delx
5940      FOR I=1 TO I1
5950        IF Xn(I)>0 THEN Newt2
5960          Xn(I)=(Xn(I)+Delx(I))/2   !                                      If negative, use half the previous value
5970 Newt2: NEXT I
5980      FOR I=1 TO I1
5990          IF ABS(Delx(I)/Xn(I))>.001 THEN Newt3
6000          IF ABS(Yn(I))>1E-4 THEN Newt3
6010          GOSUB Fx
6020      NEXT I
6030      PRINT Cv;" ITERATIONS FOR Nw = ";Nw;";  ";I1;" VARIABLES"
6040      GOSUB Prtflg
6041      IF Check=1 THEN GOSUB Printt
6050      RETURN
6060 Mater: PRINT "Matrix singular"
6070        DISP "Matrix singular"
6071        PRINTER IS 0
6080        PRINT USING "/K/5X,K";"Matrix singular","D Matrix:"
6090        PRINT D(*)
6100        GOSUB Printt
6110        STOP

6120 Prtflg:IF Aeflag THEN PRINT "Ae ";
6130        IF Alnflag THEN PRINT "AlN ";
6140        IF Lnflag THEN PRINT "Ln ";
6150        IF Grflg=2 THEN PRINT "GP ";
6160        IF Alcflag THEN PRINT "Alcflag =";Alcflag
6170        PRINT TAB(1)
6180        RETURN
6190 !
```

```
6200 ! ************************************************************************
6210 Fx:   ! ************************     FX     ***************************
6220 !
6230 !                                                                          Fitting functions for Newt
6240 !                                                                        Fx computes errors in stoich. conditions at
6250 !                                                                        existing values of parameters.
6260 !                                                                        X(J) is used since both initial value
6270 !                                                                        and the 1% change are used
6280 !                                                                        to compute the derivative.
6290 !                                                                        In addition, since Fx calls on Spec,
6300      ON Nw GOTO Fx1,Fx2,Fx3,Fx1,Fx1,Fx2,Fx7,Fx8,Fx9,Fx10,Fxx1 !

6310 Fx11:Fx=De/Nea!                                                           These are the standard returns from Fx.
6320      RETURN !                                                             The variable Fx is the fractional error
6330      !                                                                    in the stoichiometry.
6340 Fx12:Fx=Deln/Nn
6350      IF Nw=5 THEN Fx=De1/Na1
6360      RETURN
6370 Fx13:Fx=Delc/Nc
6380      RETURN
6390 Fx31:Fx=De1/Na1
6400      IF Alcflag=2 THEN Fx=Delc/Nc
6410      RETURN !

6420 Fx1: X=Xt(1)!                                                             For Nw=1,4,5
6430      Z=Xt(2)
6440      IF Grfl=1 THEN Acc=Xt(3)
6450      GOSUB Diff
6460      IF Cv=0 THEN RETURN
6470 ON I GOTO Fx11,Fx12,Fx13 !

6480 !
6490 Fx2: X=Xt(1)!                                                             For Nw=2,6
6500      IF Grfl=1 THEN Acc=Xt(2)
6510      GOSUB Diff
6520      IF Cv=0 THEN RETURN
6530      ON I GOTO Fx11,Fx13
6540 !
6550 Fx3: X=Xt(1)!                                                             For Nw=3
6560      Z=Xt(2)
6570      W=Alv=N(1)=Xt(3)
6580      IF Grfl=1 THEN Acc=Xt(4)
6590      GOSUB Diff
6600      IF Cv=0 THEN RETURN
6610 ON I GOTO Fx11,Fx12,Fx31,Fx13
6620 !
6630 Fx7:  !                                                                   In Fx, for Nw=7
6640      Z=Xt(1)
6650      IF Grfl=1 THEN Acc=Xt(2)
6660      GOSUB Diff
6670      IF Cv=0 THEN RETURN
6680      Fx=Deln/Nn
6690      IF I=2 THEN Fx=Delc/Nc
6700      RETURN
6710 !

6720 Fx8: !                                                                    In Fx, for Nw=8
6730      Acc=Xt(1)
6740      GOSUB Diff
6750      IF Cv=0 THEN RETURN
6760 Fx=Delc/Nc !

6770 Fx9: RETURN
6780 !

6790 Fx10:  Alv=W=N(1)-Xt(1)
6800      GOSUB Diff
6810      IF Cv=0 THEN RETURN
6820      Fx=De1/Na1
6830 RETURN !

6840 Fxx1:Alv=W=Xt(1)!                                                         For Nw=1?
6850      Z=Xt(2)
6860      GOSUB Diff
6870      IF Cv=0 THEN RETURN
6880      Fx=De1/Na1
6890      IF I=2 THEN Fx=Deln/Nn
6900      RETURN !
```

```
6210 ! ***************************************************************************
6220 Diff: ! ********************     DIFF     ********************************
6230 !
6250       IF (Nw=5) OR (Nw=7) THEN W=Alv=N(1)=1/Z/K(20)                    Calculates error functions in Al,O,N,C balance,
6260       IF Lqflag=1 THEN W=Alv=N(1)=1/K(19)                              to be used in Newton's method computation
6270       IF (Crf1=2) AND (Alcflag(2) THEN Acc=1
6280       IF Alcflag=2 THEN Acc=(1/K(28)/Alv^2)^(1/3)                       Crf1=2 ALSO
6290       IF Alcflag=1 THEN W=Alv=SQR(SQR(1/K(28)/Acc^3))
6300       IF Acc<1E-10 THEN Acc=1E-10
6310       IF Nw)=7 THEN X=(1/Kno/Alv^2)^(1/3)
6320       GOSUB Sugc
6330       IF (Nw=6) OR (Nw=8) OR (Nw=9) OR (Nw=10) THEN Z=1/W/K(20)
6340       GOSUB Cor2
6350       GOSUB Nit
6360       Aln=N(20)=(Nnc-Na-2*N2)*Alcflag
6370       Ae=N(21)=(Neq-FNSe)*Aeflag^3
6380       Alc=N(28)=0
6390       IF Alcflag=1 THEN Alc=N(28)=(Nal-FNSal-Aln-2*Al)/4
6400       IF Alcflag=2 THEN Alc=N(28)=(Ncc-Ce-Ce2)/3
6410       Lq1=(Alpha-FNRgt)/3
6420       Lq=N(19)=MAX(0,Lq1)*Lqflag
6430       Lq=N(19)=Lq*sLqflag
6440       De=Neq-FNSe
6450       IF Aeflag=1 THEN De=(Alpha-FNRgt)/3                              Oxygen error
6460 !                                                                      Since Lig will be absent in all cases where Deq is use
6470       Dal=Nal-FNSal-Lq-Aln-2*Ae-4*Alc                                  Aluminum error
6480       Dan=Nnc-Na-2*N2                                                  Nitrogen error
6490       Dac=Ncc-Ce-Ce2                                                   Carbon error
6500 !                                                                      Note that H2 is computed exactly in Nit
6510 !     RETURN
6520 !
```

```
7220 ! ##########################################################################
7230 ! ##########################    SPEC    ###################################
7240 Spec: !                                                                    Input: X,Kae
7250       IF Aeflag=1 THEN W=Alv=N(1)=SQR(1/Kae/X^3)   !                       Al vapor
7251       IF (Aeflag=1) AND (T(1000) THEN W=Alv=N(1)=EXP((-3*LOG(X)-Lykae)/2)
7260       O2=N(18)=X^2
7270       O=N(17)=K(17)*X
7280       Al2e=N(4)=K(4)*Alv^2*X
7290       Alow=N(13)=K(13)*Alv*X
7300       Al2=N(14)=K(14)*Alv*X^2
7310       RETURN
7320 !
7330 Car: !                                                                     INPUT: Alcflag, Ncc + Acc IF Alc is present
7340       IF Alcflag>0 THEN Car2
7350       IF Grfl=1 THEN Car2     !                                           Graphite assumed absent
7360 Car4: Acc=1 ;
7370 Car2: Cg=N(7)=K(7)*Acc*X
7380       Co2=N(8)=K(8)*Acc*X^2
7390       Gr=N(23)=(Ncc-Cg-Co2)*INT(Acc)
7400       IF (Gr)=0) OR (Grfl()0) THEN RETURN !
7410 Car3: Ce=N(7)=Ncc/(1+K(8)*X/K(7))   !                                      No graphite
7420       IF (Grfl=1) AND (Ccount)50) THEN Grfl=2   !                          Limits trials for this case
7430       Acc=Cg/K(7)/X             !                                         Activity of C(s)
7440       IF (Acc)1) AND (Grfl=0) THEN Car4  !                                If Grfl=1, Acc)0 must be carried back to Newt
7450       Co2=N(8)=K(8)*Acc*X^2
7460       Gr=N(23)=0
7470       RETURN
7480 !
7490 Nit:  !                                                                    INPUT: Z=SQR(N2),Nnc,Acc
7500       Nh=K(9)+K(10)*X+Acc*(K(29)*Z+K(30)*Z*X+K(31)*X)+K(36)*Acc^2
7510       Nh=Nh+Alv*(K(2)+K(3)*X+K(5)*X^2)   !                                Coefficients of monohydrogen species
7520       Dh=1+K(12)*X+K(33)*Acc*X+K(33)*Acc^2   !                            Coefficients of dihydrogen species
7530       Y=FNQd(2*Dh,Nh,-Nnc)   !                                             Complete atom balance in H.  Y=SQR(H2)
7540       N2=N(16)=Z^2
7550       No=N(15)=K(15)*X*Z
7560       Alh=N(2)=Alv*K(2)*Y
7570       Aloh=N(3)=Alv*K(3)*X*Y
7580       Alp=N(5)=Cnn=N(24)=SQR(K(5)*K(24)*Alv*Acc*Z)
7590       H=N(9)=K(9)*Y
7600       Oh=N(10)=K(10)*X*Y
7610       H2=N(11)=Y^2
7620       H2o=N(12)=K(12)*X*Y^2
7630       Hcn=N(29)=K(29)*Acc*Y*Z
7640       Hnco=N(30)=K(30)*Acc*X*Y*Z
7650       Hco=N(31)=K(31)*Acc*X*Y
7660       Ch2o=N(32)=K(32)*Acc*X*Y^2
7670       C2h2=N(33)=K(33)*(Acc*Y)^2
7680       Cg=N(25)=K(25)*Acc*Z
7690       C2n=N(26)=K(26)*Acc^2*Y
7700       C2n=N(27)=K(27)*Acc^2*Z
7710       C3=N(34)=K(34)*Acc^3
7720       Ncc=Nn-Hcn-Nnco-Nco-2*Cnn-C2n-Cnn   !                                Corrected moles of N atoms
7730       IF Nnc<0 THEN Nnc=1E-50
7740       Ncc=Nc-Hcn-Nnco-Hco-Ch2o-Cn-2*(C2h2+C2h+C2n)-3*C3-Cnn   !            Corrected atom balance in C
7750       IF Ncc<0 THEN Ncc=1E-50
7760       RETURN
7770 !
```

```
7780 ! ********************************************************************************
7790 ! ********************************    DIAGNOSTICS    ********************************
7800 Printt: OUTPUT 0 USING "/K,K/";"T = ",T
7810 !
7820 Sum:  FOR I=1 TO 32 STEP 4
7830       OUTPUT 0 USING Imsum;Fo$(I),N(I),Fo$(I+1),N(I+1),Fo$(I+2),N(I+2),Fo$(I+3),N(I+3)
7840     NEXT I
7850     OUTPUT 0 USING Imsum;Fo$(33),N(33),Fo$(34),N(34)
7860 Imsum: IMAGE 4(8A,MZ.DC,3X)
7870       PRINTER IS 0!

7880 Sum1:  Sum=FNSol+2*Ao+Aln+4*Alc+Lq
7890       PRINT LIN(1)
7900       GOSUB Prtflg
7910       PRINT "Al BALANCE ",Nal,Sum
7920       Sum=Nc-Ncc+Co+Co2+Cr+3*Alc
7930       PRINT "C BALANCE ",Nc,Sum
7940       Sum=N(9)+N(10)+2*(N(11)+N(12))+Hcn+Hnco+Hco+2*(Ch2o+C2h2)+Alh+Aloh+Alho2+C2h
7950       PRINT "H BALANCE ",Nh,Sum
7960       Sum=FNSo+3*Ao
7970       PRINT "O BALANCE ",Noa,Sum
7980       Sum=Aln+2*N2+No+Hcn+Hnco+Cn+C2n+Cnm
7990       PRINT "N BALANCE ",Nn,Sum
8000       PRINT USING "K,K,//";"Acc = ",Acc
8010 Testk: !
8020       PRINT LIN(2);TAB(20);"Test for equilibrium";LIN(1)
8030       PRINT SPA(1);"K";TAB(10);"Numeric";TAB(30);"Functional(<=Numeric)",LIN(1)
8040       PRINT "Al(g)";TAB(10);1/K(19);TAB(30);Alv
8050       PRINT "CO";TAB(10);1/K(7);TAB(30);1/(Co/(Acc*SQR(O2)))
8060       PRINT "H2O";TAB(10);1/K(12);TAB(30);1/(H2o/(H2*SQR(O2)))
8070       IF T>=1000 THEN PRINT "Al2O3";TAB(10);1/Kao;TAB(30);Alv^2*O2^1.5
8075       IF T<1000 THEN PRINT "Al2O3";TAB(10);-Lgkao;TAB(30);2*LOG(Alv)+1.5*LOG(O2)          ! Log format below 1000 K
8080       PRINT "AlN";TAB(10);1/K(20);TAB(30);Alv*SQR(N2)
8090       PRINT "Al4C3";TAB(10);1/K(28);TAB(30);Alv^4*Acc^3
8100       PRINTER IS 7,1
8110       RETURN !
```

```
8120 ! **********************************************
8130 !  *****    CODE FOR Newt COMPUTATIONS:    ********
8140 !       Al2O3    Liq      AlN       VARIABLES      Nw
8150 !         +       -        -        X,Z,(Acc)      1
8160 !         +       -        +          X,(Acc)      3
8170 !         +       -        -        Y,Z,W (Acc)    3
8180 !         -       +        -        X,Z,(Acc)      4
8190 !         -       +        +        X,Z,(Acc)      5
8200 !         -       +        -          X,(Acc)      6
8210 !         +       +        -          Z,(Acc)      7
8220 !         +       +        +          Acc          8
8230 !         +       -        +       (Acc=1)         9
8240 !         +       -        -          Alw          10   Alcflag=2 (Al in excess)
8250 !         -       -        -          Alv          11   Alcflag=2 (  "  )
8260 !   Alcflag=1 (Al4C3 in presence of excess carbon) gives cases equivalent to liquid metal present
8270 !      Stoichiometric left in the material balance in the appropriate
8280 !       element except:  use Al balance when Z is variable for Nw=5 and
8290 !        use C balance when Alv is variable for Nw=3 and Alcflag=2 (probably never)
8300 !


8310 !  1     Al
8320 !  2     AlH
8330 !  3     AlOH
8340 !  4     Al2O
8350 !  5     Al+
8360 !  6     Ar
8370 !  7     CO
8380 !  8     CO2
8390 !  9     H
8400 ! 10      OH
8410 ! 11     H2
8420 ! 12     H2O
8430 ! 13     AlO
8440 ! 14     AlO2
8450 ! 15     N2
8460 ! 16     N2
8470 ! 17     O
8480 ! 18     O2
8490 ! 19     Al(l)        lq       Lqflag
8500 ! 20     AlN                   Alnflag
8510 ! 21     Al2O3(l)     Ao       Aoflag       Kao
8520 ! 22     Al2O3(s)     Ao       Aoflag       Kao
8530 ! 23     C(s)         Gr       Grfl         Acc=Activity of C (Gr= std. state)
8540 !                               Grfl=1 for no Gr; =2 for Gr present.
8550 ! 24     CN-
8560 ! 25     CN
8570 ! 26     C2H
8580 ! 27     C2N
8590 ! 28     Al4C3        Alc      Alcflag
8600 ! 29     HCN
8610 ! 30     HNCO
8620 ! 31     HCO
8630 ! 32     CH3O
8640 ! 33     C2H2
8650 ! 34     C3
```

DATA FILE *THD*

1. Equilibrium Constant of Formation Data

0	0	0	0	0	0
-2.04642317215	3095.08563709	-61.4431838074	-7.59713579408E-05	2.49078062430E-03	-.5
-6.82758724985	26914.8692052	1.94418164793	9.72325196131E-06	9.52727405371E-05	-1
-8.58206579927	40170.5376863	-8.77658908461	-1.74973854021E-04	4.98920268686E-03	-.5
2.25546558767	-32795.8697374	27.3534182517	9.80703937722E-05	2.37493938912E-03	1
0	0	0	0	0	0
4.57311031464	6188.44384152	29.708241155	-6.72129593889E-05	9.17169097642E-05	.5
6.74863151113E-02	28641.664597	1.18073929201	-2.43678722090E-05	6.32488877623E-05	0
3.13209978797	-12816.3695425	19.8339617644	1.48157371394E-05	8.41355776673E-05	.5
.769405272297	-1968.95209581	-16.3438133692	-1.47482381947E-05	4.41263920863E-05	0
0	0	0	0	0	0
-3.05603740471	13305.1928091	13.8679740254	-5.19832483354E-06	4.75084633874E-05	-.5
-2.94319764443	12989.3268156	5.26338084125	6.78638394752E-05	1.85115938584E-04	-.5
-6.14764465566	26185.834292	-7.41178328136	1.84832484614E-05	1.16220807803E-04	-1
.718103195616	-4088.84583459	7.78344272283	-9.88448254772E-06	3.92020867050E-05	0
0	0	0	0	0	0
3.49713846375	-13438.7266853	9.78086108778	6.17322768660E-06	6.93355557051E-05	.5
0	0	0	0	0	0
-5.95275584825	15658.2581898	-20.8673104972	9.84286188286E-05	1.71621469196E-04	-1
-10.9363518792	31054.2661256	-37.261473539	-1.17318808542E-04	2.75186302573E-03	-.5
-40.6869788932	154392.884779	282.327462644	1.92799147857E-03	6.13852865782E-03	-3.5
-29.4359333294	118807.365935	-6.88469862864	5.86473338935E-04	3.13818888764E-04	-3.5
0	0	0	0	0	0
.825767716985	4.76832635592	-2418.34572894	4.65494992481E-05	1.61847243788E-02	-.5
5.09215883192	-22218.8465815	-11.9581155333	-1.64335878389E-05	2.07761783011E-04	.5
6.52938298135	-24268.9161784	-12.4711153722	-4.29783319993E-05	1.86617359728E-04	.5
7.0682273836	-29343.3317977	-.378425411396	-6.61291975814E-05	9.54771877177E-05	.5
-24.9826257877	71142.8930524	-49.1382118137	-5.19454632333E-04	6.46687977746E-03	-4
1.68731167515	-6886.78852682	-12.8668891672	-7.69875947818E-06	5.53986588187E-05	0
-25.7483753981	96498.8463919	2562.82767313	2.86185217684E-03	1.15261813721E-02	-.5
2.2214151864	1289.81995973	219.335325689	-5.99683871438E-05	4.36611516989E-05	0
-2.03598836359	6769.63261573	18.2858441719	-1.62155531526E-05	9.66245762328E-05	-.5
2.65481667687	-11332.2620388	-15.5572448176	1.36720789929E-07	6.49226563872E-05	0
10.8495887765	-41395.8614497	-16.8531546361	-1.85667525968E-04	3.89781110914E-04	1

2. Internal Energy of Formation Data

318931.276915	11.6599712	.00007516556	1364.27608	664251.84
264815.420609	30.1511592	.00027214828	-2961.0168	1839598.48
-55065.7557	60.797312	-.0003553262	-22627.9008	-4116177.36
-149140.855536	50.095032	-1.207167680008E-05	-572.49672	1672177.44
824363.572854	11.395124	.000869554816	2733.78376	2271489.92
-3717.71863872	12.472504	0	0	0
-84114.5446491	31.1360992	-.000824608196	-6217.8424	23734.9952
-361013.397391	56.789432	.000866571624	-9200.616	1763807.84
213829.288681	12.472504	0	0	0
166504.893264	34.785776	-.000080058108	-20092.8232	-7842489.6
136535.182859	32.8095872	.0003125448	-21058.872	-9412744.8
50544.6435647	63.224032	-.00039459384	-45948.688	-16417179.2
844161.136898	86.474912	-.0036815872	-98479	-39617840.8
-229457.536972	51.802184	.00022566484	3388.2032	3817565.28
185581.17903	30.618512	.000018425482	-4353.452	630821.68
38785.1226616	31.6511232	-.000060705152	-7030.7744	-657139.84
214445.858485	9.684372	.00029773344	4427.5088	2058327.36
32184.6956292	29.47628	.00066230628	-6264.7032	-2102250.8
-794.96	31.748192	0	0	0
-297115.828612	37.5664624	.0001886442	-407.195248	858933.36
-1632491.4484	136.548632	0	0	0
-1471365.88672	146.423264	.00183298948	-37420.4488	-12025234.4
-48329.1736189	22.9848528	.0003199714	3488.71336	5429158.4
102398.354976	31.5587872	-.00006874312	-7172.2128	-392011.512
691553.916429	38.82752	.00074033708	-37192.4128	-24657567.2
758476.829892	66.613464	.0002391156	-43881.792	-28691553.6
538335.412899	54.37188	-.000015466156	-1052.18864	2483959.84
-89326.8112584	204.334808	.000136402584	-31427.2792	-2969381.12
284943.717866	62.76	-.0003023986	-26215.6888	-5946719.2
36616.8667367	83.572136	-.00047486812	-28280.9112	-4749676.8
90275.7482183	56.166816	-.0003527112	-19233.4296	-2344797.28
54534.8289981	85.5628	-.00060937868	-32361.5664	-1612811.52
474989.858316	92.818712	-.00002674622	-42593.12	-10538659.2
916856.388899	48.906776	.0002815832	-16185.3856	-3220898.88

3. Chemical Species Symbols

Al
AlH
AlOH
Al2O
Al+
e-
CO
CO2
H
H2
H2O
AlO
AlO2
HO
HO2
O
O2
Al(l)
AlN
Al2O3(l)
Al2O3(s)
C(s)
CH
CH2
CH3
Al4C3
HCN
HNCO
HCO
CH2O
CH2
C3

INITIAL DISTRIBUTION

7 Naval Air Systems Command
- AIR-00D4 (2)
- AIR-301 (2)
- AIR-350 (1)
- AIR-320E (1)
- AIR-541 (1)

5 Chief of Naval Operations
- OP-03 (2)
- OP-05 (1)
- OP-098 (1)
- OP-55 (1)

1 Chief of Naval Material (MAT-05)

3 Chief of Naval Research, Arlington
- ONR-102 (1)
- ONR-461 (1)
- ONR-473 (1)

2 Naval Sea Systems Command (SEA-99612)

1 Commander in Chief, U.S. Pacific Fleet (Code 325)

1 Air Test and Evaluation Squadron 5

1 Commander, Third Fleet, Pearl Harbor

1 Commander, Seventh Fleet, San Francisco

1 David W. Taylor Naval Ship Research and Development Center, Bethesda

1 Naval Air Engineering Center, Lakehurst

1 Naval Air Force, Atlantic Fleet

2 Naval Air Force, Pacific Fleet

1 Naval Air Station, North Island

1 Naval Air Test Center, Patuxent River (CT-252, Bldg. 405 (Aeronautical Publications Library))

1 Naval Avionics Center, Indianapolis (Technical Library)

2 Naval Civil Engineering Laboratory, Port Hueneme
- W. Keenan (1)
- Technical Library (1)

1 Naval Explosive Ordnance Disposal Technology Center, Indian Head

1 Naval Ocean Systems Center, San Diego (Code 131)

1 Naval Ordnance Station, Indian Head (Technical Library)

28 Naval Postgraduate School, Monterey
- Code 012, Dean of Research (1)
- Code 06, Dean of Science and Engineering (1)
- Code 1424, Library - Technical Reports (2)
- Code 33, Weapons Engineering Program Office (1)
- Code 61, Chairman, Physics and Chemistry Department (1)
- Code 61KY, G. F. Kinney (2)
- Code 61RI, R. A. Reinhardt (20)

3 Naval Ship Weapon Systems Engineering Station, Port Hueneme
- Code 5711, Repository (2)
- Code 5712 (1)

12 Naval Surface Weapons Center, Dahlgren
- G10 (1)
- G20 (1)
- G25, Hales (2)
- G30 (1)
- G35 (1)
- G43 (1)
- G51 (1)
- G103 (1)
- N12 (1)
- W. Marton (1)
- Technical Library (1)

14 Naval Surface Weapons Center, White Oak Laboratory, Silver Spring
- G42 (1)
- R10, S. Jacobs (1)
- R12
  - H. Adloph (1)
  - R. Bernecker (1)
  - K. Durrell (1)
  - J. Erkman (1)
  - W. Filler (1)
  - K. Kim (1)
  - Roslund (1)
  - Short (2)
- R13, R. Liddiard (1)
- Guided Missile Warhead Section (1)
- Technical Library (1)

1 Naval War College, Newport
1 Office of Naval Research, Pasadena Branch Office
1 Operational Test and Evaluation Force, Atlantic
1 Operational Test and Evaluation Force, Pacific
1 Pacific Missile Test Center, Point Mugu (Technical Library)
1 Marine Air Base Squadron 32, Beaufort
1 Army Armament Materiel Readiness Command, Rock Island (Technical Library)
4 Army Armament Research and Development Command, Dover
- DRDAR-LCU-SS, J. Pentel (1)
- DRDAR-TSS, Technical Library (3)

1 Aberdeen Proving Ground (Development and Proof Services)
2 Army Ballistic Research Laboratory, Aberdeen Proving Ground
- DRDAR-BLT, C. Kingery (1)
- DRDAR-BLV, Vitali (1)

1 Army Materiel Systems Analysis Activity, Aberdeen Proving Ground (J. Sperrazza)
1 Army Research Office, Research Triangle Park
1 Harry Diamond Laboratories, Adelphi (Technical Library)
1 Radford Army Ammunition Plant
1 Redstone Arsenal (Rocket Development Laboratory, Test and Evaluation Branch)
1 Rock Island Arsenal
1 White Sands Missile Range (STEWS-AD-L)
1 Yuma Proving Gounds (STEYT-GTE, M&W Branch)
1 Tactical Air Command, Langley Air Force Base (TPL-ROD-M)
1 Air University Library, Maxwell Air Force Base
1 Tactical Fighter Weapons Center, Nellis Air Force Base (COA)
1 57th Fighter Weapons Wing, Nellis Air Force Base (DTO)
1 554th Combat Support Group, Nellis Air Force Base (OT)
2 Defense Nuclear Agency
- Shock Physics Directorate (1)
- M. Frankel (1)

12 Defense Technical Information Center
1 Department of Defense Explosives Safety Board, Alexandria (Chairman)

1 Department of Defense-Institute for Defense Analyses Management Office (DIMO), Alexandria
1 Lewis Research Center (NASA), Cleveland
1 California Institute of Technology, Jet Propulsion Laboratory, Pasadena, CA
1 General Dynamics Corporation, Pomona Division, Pomona, CA (E. LaRocca)
2 Hercules Incorporated, Allegany Ballistics Laboratory, Cumberland, MD
1 IIT Research Institute, Chicago, IL (Department M, Document Librarian)
2 Johns Hopkins University, Applied Physics Laboratory, Laurel, MD (Document Library)
2 Johns Hopkins University, Applied Physics Laboratory, Chemical Propulsion Information Agency, Laurel, MD
2 Los Alamos National Laboratory, Los Alamos, NM
    C. Madgr, MS 214 (1)
    Reports Library (1)
1 Princeton University, Forrestal Campus Library, Princeton, NJ
1 R&D Associates, Marina del Rey, CA
1 Stanford Research Institute, Poulter Laboratories, Menlo Park, CA
1 The Rand Corporation, Santa Monica, CA (Technical Library)
1 University of California, Lawrence Livermore National Laboratory, Livermore, CA
1 University of Denver, Denver Research Institute, Denver, CO