

Chapter 1. Basic concepts of thermodynamics

[Problem 1.1A. External state variables](#)

[Problem 1.1B. External state variables](#)

[Problem 1.2. Internal state variables](#)

[Problem 1.3A. The first law of thermodynamics](#)

[Problem 1.3B. The first law of thermodynamics](#)

[Problem 1.4. Freezing-in conditions](#)

[Problem 1.5. Reversible and irreversible processes](#)

[Problem 1.6. The second law of thermodynamics](#)

[Problem 1.7. Condition of internal equilibrium](#)

[Problem 1.8. Driving force](#)

[Problem 1.9. The combined first and second law](#)

[Problem 1.10. General conditions of equilibrium](#)

1.1A. External state variables

- a) Consider a system of pure carbon. Define the conditions in sufficient detail to allow the state of equilibrium to be computed, using a thermodynamic data bank. Choose the conditions any way you like, except that $P=1$ bar should be chosen. Then, evaluate the volume.
- b) Use that volume when redefining the conditions and exclude another piece of information from the conditions. Then, evaluate the state of equilibrium, which should be the same as before. Check that by inspecting the value of the excluded property. It should be the same as before.

Hint

- 1) The conditions can be defined using the values of $c+2=1+2=3$ independent variables. So far we have discussed T , P and V but it is evident that another one is the amount of material, e.g. the number of moles of components (usually atoms), N . Since you are asked to evaluate V , it is evident that among those four variables you must give the values to all three of T , P and N to define the conditions. You may take 1000°C , 1 bar and 1 mol. 1 bar is 100000 Pa.
- 2) There can never be more than $c+1$ independent *intensive* variables, in this case T and P , whereas V and N are *extensive* variables and at least one of them is required in order to have a complete definition of the conditions. It serves to define the size of the system. However, you could use more than one extensive variable. When including V in the new set of independent variables, you could exclude N but it may be more interesting instead to exclude one of the potentials, say T , and use two extensive variables, V and N , together with P .

Instructions for using T-C

- 1) Go first to the database module to fetch the thermodynamic information and then to the equilibrium module POLY. It is constructed to compute a state of equilibrium by minimizing the Gibbs energy and then to give information on the computed state. It is thus necessary first to make the program perform an equilibrium computation even if it is trivial when the set of conditions is sufficient to define the state without any minimization. The reason is that POLY does not accept the given conditions as a description of a state. They are just treated as conditions for the equilibrium to be computed.
- 2) Explanations are inserted in the following print-out of Prompts, commands and responses and are marked with *) and printed with a different font.
- 3) All commands that you should give are printed **bold** but usually in a much abbreviated form.

Prompts, commands and responses

SYS:

*) Thermo-Calc (T-C) is composed of several modules. You should first **goto** the **database** module.

SYS: **go**

MODULE NAME: **da**

THERMODYNAMIC DATABASE module running on PC/WINDOWS NT

Current database: TCS Demo Al-Mg-Si Alloys TDB v1

VA DEFINED

TDB_DALMGSI:

*) You like to **switch** to another database. Press return after **sw** and you find the limited list of the databases available to you as a customer of the free-of-charge version of T-C. You should realize that it is really limited and there will be no ambition to use a wide variety of systems in the present set of problems. On the contrary, it may be of some pedagogical value that you get familiar to the systems you work with.

TDB_DALMGSI: **sw**

Use one of these databases

DALMGSI = TCS Demo Al-Mg-Si Alloys TDB v1
 DFECRC = TCS Demo Fe-Cr-C Alloys TDB v1
 PURE4 = SGTE Unary (Pure Elements) TDB v4
 PSUB = TCS Public Pure Substances TDB v1
 PBIN = TCS Public Binary Alloys TDB v1
 PKP = Kaufman Binary Alloys TDB v1
 PCHAT = Chatenay-Malabry Binary Alloys TDB v1
 PTERN = TCS Public Ternary Alloys TDB v1
 PG35 = G35 Binary Semi-Conductors TDB v1
 PION = TCS Public Ionic Solutions TDB v2
 PAQ2 = TCS Public Aqueous Solution TDB v2
 PGEO = Saxena Pure Minerals Database v1
 PFRIB = Fridberg Dilute Fe-Alloys MDB v1
 USER = User defined Database

DATABASE NAME /DALMGSI/: **PTERN**

Current database: TCS Public Ternary Alloys TDB v1

VA DEFINED

TDB_TERN:

*) You should now define the system. It is usually convenient to **define** the system through the **elements**. As stated just before the TDB_DALMGSI, you never need to define vacancies that some models use. Now your system should only contain carbon.

TDB_PTERN: **def-el**

ELEMENTS: **C**

C DEFINED

TDB_PTERN:

*) Before you are used to T-C you better list the **system** to see what you have accomplished so far.

TDB_TERN: **l-sys**

ELEMENTS, SPECIES, PHASES OR CONSTITUENTS: /CONSTITUENT/: **CONSTITUENT**

LIQUID:L :C:

> This is metallic liquid solution phase, with C species

GRAPHITE :C:

TDB_TERN:

*) You got two phases, liquid and graphite, but like to **reject** the **phase** called **liquid**. Then you are satisfied with the definition of the system and like to **get** the data.

TDB_TERN: **rej**

ELEMENTS, SPECIES, PHASES, CONSTITUENT OR SYSTEM: /PHASES/: **p**

PHASES: **liq**

LIQUID:L REJECTED

TDB_TERN: **get**

REINITIATING GES5

ELEMENTS

SPECIES

PHASES

PARAMETERS ...

Rewind to read functions 11

FUNCTIONS

List of references for assessed data

The list of references can be obtained in the Gibbs Energy System also by the command LIST_DATA and option R

-OK-

TDB_TERN:

*) Now you should **goto** the equilibration module called **POLY-3** (historically the 3rd version).

TDB_TERN: **go**

MODULE NAME: **pol**

POLY version 3.32, Aug 2001

POLY_3:

*) In the database you defined what phases and elements the system should contain. In POLY you should give values of the state variables with which you like to define the conditions for the equilibrium. The command is **set-conditions**. You will then be prompted to define your choices. If you accept the default, you just press return. Notice that N is the size of the system expressed as the number of moles of components (here atoms). If you like to accept a default value, just press return.

POLY_3: **s-c**

State variable expression: **P**

Value /100000/:

POLY_3: **s-c**

State variable expression: **T**

Value /1000/: **1273**

POLY_3: **s-c**

State variable expression: **N**

Value /0/: **1**

POLY_3:

*) Just to be sure, you may list your the conditions.

POLY_3: **1-c**

P=1E5, T=1273, N=1

DEGREES OF FREEDOM 0

POLY_3:

*) Degrees of freedom = 0 means that the equilibrium will be well defined. POLY can only compute equilibria with no degree of freedom. Evidently, you could now ask POLY to compute equilibrium.

POLY_3: **c-e**

Using global minimization procedure

Calculated 1 grid points in 0 s

POLY_3:

*) It seems that the computation was successful. Next you were asked to use the volume as a condition. In order to obtain the same equilibrium, you should require that V has the value of the state just computed. You don't need to inspect it in advance. The value of the current state will always be proposed to you as a default. You could type V or V=.

POLY_3: **s-c V**

POLY_3: Value /5.441691885E-06/:

*) Just press return.

POLY_3: Value /5.441691885E-06/:

POLY_3:

*) Just to be sure, you could again list the conditions before asking POLY to compute equilibrium.

POLY_3: **1-c**

P=1E5, T=1273, N=1, V=5.44169E-6

DEGREES OF FREEDOM -1

POLY_3:

*) There is one condition too many which is expected because you have added one. You should remove another one. You are free to remove any one, even the size of the system because the size is now defined by the volume.

POLY_3: **s-c N=none**

POLY_3:

*) You should note that you will not remove the condition based on N by typing N=0. On the contrary, that will just change the previous condition to the new N value, being 0. Now you can again try to compute equilibrium.

POLY_3: **c-e**

Normal POLY minimization, not global

Testing POLY result by global minimization procedure

Calculated 1 grid points in 0 s

6 ITS, CPU TIME USED 0 SECONDS

POLY_3:

*) You succeeded and like to show the N value in order to check that you obtained the same state of equilibrium.

POLY_3: **sh N**

N=1.

POLY_3:

*) Fine! Try to use both N and V to define conditions. Enter V instead of P.

POLY_3: **s-c P=none N=**

POLY_3: Value /1/:1

POLY_3: **c-e**

Normal POLY minimization, not global

Testing POLY result by global minimization procedure

Calculated 1 grid points in 0 s

6 ITS, CPU TIME USED 0 SECONDS

```
POLY_3: sh P
P=100002.13
    *)This is close enough.
POLY_3: exit
CPU time          0 seconds
```

Comments

- 1) You must give the value of at least one extensive state variable as condition. Here you saw that you could use more than one. In fact, you could give all the conditions with extensive state variables.
- 2) When giving commands it does not matter if you use upper or lower cases. Pressure could be typed with p and Fe with fe or even FE
- 3) The number of responses can be shortened when you learn to remember what prompts will automatically follow many main commands. You can type all those responses on the same line as the command. That will be demonstrated in the next Problem.
- 4) When you don't remember what would be a proper response to a prompt, you can get a list of all your options by typing **?** or **help**, followed by return.

1.1B. External state variables

Calculate and plot the function $V(P, T_1, N_1)$ for graphite between 1 bar and 1 kbar (i.e. 100000 to 1E8 Pa) at 1000°C and for 1 mol.

Hint

The method of computing and plotting a curve depends on your particular data bank system.

Instructions for using T-C

- 1) In order to plot a curve showing how some property of a system at equilibrium varies under changing conditions, you start by computing an initial equilibrium to be used as a starting point. Then you can let the computation be repeated at close intervals by varying one of the conditions. In the present case, let the P value vary between two limits.
- 2) The data should be fetched from a database as demonstrated in the previous Problem. If you have just solved that Problem, you may go back to the beginning of POLY and start from there using the command "reinitiate-module". Everything in POLY has been deleted but not data from the database.
- 3) Now you should begin typing responses on the same line as the command when possible.

Prompts, commands and responses

```
SYS: go da
THERMODYNAMIC DATABASE module running on PC/WINDOWS NT
```

Current database: TCS Demo Al-Mg-Si Alloys TDB v1

VA DEFINED

TDB_DALMGSI:

*) You need another database. You could type switch and return and get a list of databases available to you. You did that in the preceding Problem. If you remember that list, you can type it directly without consulting the list.

TDB_DALMGSI: **sw ptern**

Current database: TCS Public Ternary Alloys TDB v1

VA DEFINED

TDB_PTERN:

*) Now you should define the system and could choose the option to give the elements. You may remember that you will be prompted to name the element. Then you could do that on the same line.

TDB_PTERN: **def-el C**

C DEFINED

TDB_PTERN:

*) You may note that words belonging to the same command should be connected with a hyphen "-" (or underscore "_") but words being the responses to new prompts or questions, however not seen, will not have a hyphen. This explains "def-el C". Now you could list the system to see what it contains.

TDB_PTERN: **l-sys**

ELEMENTS, SPECIES, PHASES OR CONSTITUENTS: /CONSTITUENT/: **CONSTITUENT**

LIQUID:L :C:

> This is metallic liquid solution phase, with C species

GRAPHITE :C:

TDB_PTERN:

*) There are two phases but you only want graphite. Thus, you **reject** the phase **liquid**.

TDB_PTERN: **rej p liq**

LIQUID:L REJECTED

TDB_PTERN: **get**

REINITIATING GES5

ELEMENTS

SPECIES

PHASES

PARAMETERS ...

Rewind to read functions 11

FUNCTIONS

List of references for assessed data

The list of references can be obtained in the Gibbs Energy System also by the command LIST_DATA and option R

-OK-

TDB_PTERN: **go pol**

POLY version 3.32, Aug 2001

POLY_3:

*) Next you should set the conditions for an initial equilibrium as an introduction to stepping through the range of P values. Remember that you can give several items on the same line.

POLY_3: **s-c P=1E5 T=1273 N=1**

POLY_3:

*) You should notice that P and 1E5 were responses to separate prompts and should normally be separated by a blank. However, when the second one is a value to the first one,

you should use "=". Just for safety, you can now list the conditions. You will find that the degree of freedom is zero and could thus continue with computing equilibrium.

POLY_3: **1-c**

P=1E5, T=1273, N=1
DEGREES OF FREEDOM 0

POLY_3: **c-e**

Using global minimization procedure
Calculated 1 grid points in 0 s

POLY_3:

*) You can now make a series of calculations for the range of P and for the values of T and N used in defining the conditions for the initial equilibrium. First you specify the axes in the diagram to be plotted by setting the axis variables. Often one can accept the default value of the increment. Then just press return or type the value.

POLY_3: **s-a-v**

Axis number /1/: 1
Condition /NONE/: P
Min value /0/: 1e5
Max value /1/: 1e8
Increment /2497500/: 2*

POLY_3:

) 2 means logarithmic step. It is often used when the variable covers several orders of magnitude.

POLY_3: **step**

Option? /NORMAL/: NORMAL

looking for miscibility gaps..at:

QSTEPP 100000000.000000
QSTEPP 100000.000000000
QSTEPP 50050000.0000000

No new miscibility gap found!

Phase Region from 0.100000E+09 for:

GRAPHITE

Calculated 13 equilibria

*** Buffer saved on file: USERPROFILE\RESULT.POLY3

POLY_3:

*)The computation is finished and you should go to the postprocessor module. It is actually a submodule to POLY and you don't need to type "goto".

POLY_3: **post**

POLY-3 POSTPROCESSOR VERSION 3.2 , last update 2002-12-01

POST: **s-d-a**

AXIS (X, Y, or Z) :x

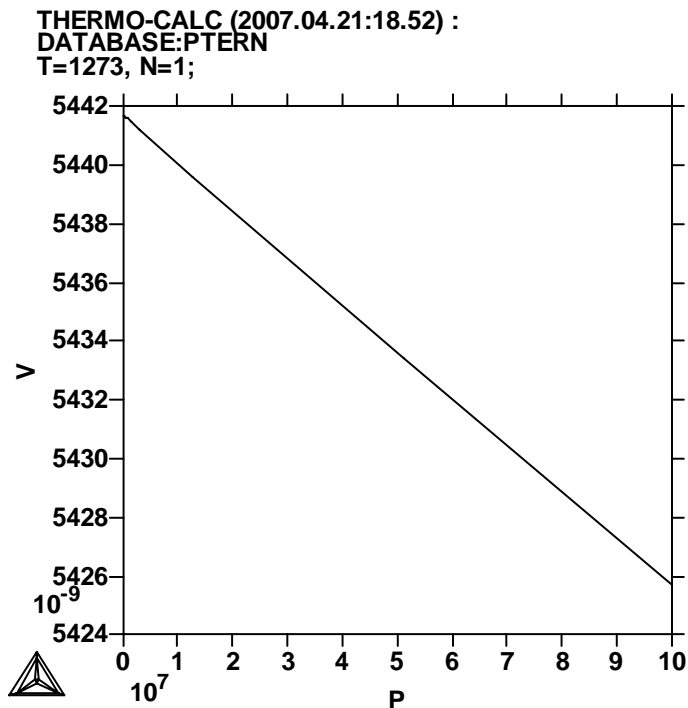
VARIABLE : P

POST: **s-d-a y**

VARIABLE : V

POST: **plot**

OUTPUT TO SCREEN OR FILE /SCREEN/:



POST: **exit**
 CPU time 0 seconds

Comments

Here you have learned how to compute and plot a curve.

1.2. Internal state variables

Consider an Ag-Cu alloy with 10 mass% Cu at 600°C and 1 atm (101325 Pa). Compute the equilibrium and find the values of some internal variables. Then, use one of those values when redefining the conditions and instead exclude one of the external variables. Again calculate the equilibrium and check that the excluded variable got the same value as it had before.

Hint

You can certainly make your program present the calculated state of equilibrium. It will then give a long list containing the external variables but also some internal variables, e.g. the fractions of phases and their compositions if there is more than one phase. Choose any of these values when redefining the conditions for a new computation of the same of equilibrium.

Instructions for using T-C

- 1) There may be other properties of the equilibrium state that are not included in the list. As described later, you can ask specifically for a large number of such properties with the command "show <variable>", whether it is included in the list or not.
- 2) When defining a system with more than one component, one can either give the amount as the number of moles, N(i), or the mass in gram, B(i), of a component or the total amount, N or B, and the composition expressed by the mole fractions, x(i) etc., but omitting one because $\sum x(i) = 0$.

Prompts, commands and responses

```

SYS: go da
THERMODYNAMIC DATABASE module running on PC/WINDOWS NT
Current database: TCS Demo Al-Mg-Si Alloys TDB v1

VA DEFINED
TDB_DALMGSI: sw
Use one of these databases

DALMGSI = TCS Demo Al-Mg-Si Alloys TDB v1
DFECRC = TCS Demo Fe-Cr-C Alloys TDB v1
PURE4 = SGTE Unary (Pure Elements) TDB v4
PSUB = TCS Public Pure Substances TDB v1
PBIN = TCS Public Binary Alloys TDB v1
PKP = Kaufman Binary Alloys TDB v1
PCHAT = Chatenay-Malabry Binary Alloys TDB v1
PTERN = TCS Public Ternary Alloys TDB v1
PG35 = G35 Binary Semi-Conductors TDB v1
PION = TCS Public Ionic Solutions TDB v2
PAQ2 = TCS Public Aqueous Solution TDB v2
PGEO = Saxena Pure Minerals Database v1
PFRIB = Fridberg Dilute Fe-Alloys MDB v1
USER = User defined Database

DATABASE NAME /DALMGSI/: pbin
Current database: TCS Public Binary Alloys TDB v1

VA /- DEFINED
IONIC_LIQ:Y L12_FCC B2_BCC
BCC_B2 REJECTED
TDB_PBIN: def-el Ag Cu
AG CU DEFINED
TDB_PBIN: 1-sys
ELEMENTS, SPECIES, PHASES OR CONSTITUENTS: /CONSTITUENT/:
LIQUID:L :AG CU:
> This is metallic liquid solution phase, with C-N species
FCC_A1 :AG CU:VA:
BCC_A2 :CU:VA:
HCP_A3 :CU:VA:
ALCU_EPSILON :CU:CU:
ALCU_ETA :CU:CU:
CUZN_EPS :CU:
TDB_PBIN:
*) As usual, the database contains more phases than you are interested in. This time accept
all the phases on this level but make sure that unwanted phases don't interfere later on.
TDB_PBIN: get

```

```

REINITIATING GES5 .....
ELEMENTS .....
SPECIES .....
PHASES .....
PARAMETERS ...
Rewind to read functions          3
FUNCTIONS ....

```

List of references for assessed data

```

91Din 'A.T. Dinsdale, SGTE Data for Pure Elements, CALPHAD, Vol.15,
      No.4, pp.317-425, (1991)'
HAY-AGCU 'F.H. Hayes, H.L. Lukas, G. Effenberg, G. Petzow,
      Z. fur Metallkde, Vol 77 (1986), No 11, p 749-754; AG-CU-PB'
NIG-ALCU 'Nigel Saunders, COST 507 round 1, (1993); Al-Cu'
KOW-CUZN 'M Kowalski and P Spencer, J Phase Equil, p 432-438 (1993); CU-ZN'

```

The list of references can be obtained in the Gibbs Energy System also by the command LIST_DATA and option R

-OK-

TDB_PBIN: **go pol**

POLY version 3.32, Aug 2001
POLY_3: **s-c T=873 P=101325 w(Cu)=0.1 N=1**

*) You may list the conditions. Then compute equilibrium if degrees of freedom=0.

POLY_3: **l-c**

T=873, P=1.01325E5, W(CU)=0.1, N=1
DEGREES OF FREEDOM 0

POLY_3: **c-e**

Using global minimization procedure
Calculated 279 grid points in 0 s
Found the set of lowest grid points in 0 s
Creating a new composition set FCC_A1#2
Calculated POLY solution 0 s, total time 0 s

POLY_3:

*) This time you like to inspect the computed equilibrium in detail by the command list-equilibrium.

POLY_3: **l-e**

OUTPUT TO SCREEN OR FILE /SCREEN/: **SCREEN**

Options /VWCS/:

*) There are two options for each position. W means mass fraction but it could be exchanged for X, meaning mole fractions, which you may prefer. You could inspect all the options by printing ? and return.

Options /VWCS/:

Output from POLY-3, equilibrium = 1, label A0 , database: PBIN

Conditions:

T=873, P=1.01325E5, W(CU)=0.1, N=1
DEGREES OF FREEDOM 0

Temperature 873.00 K (599.85 C), Pressure 1.013250E+05
Number of moles of components 1.00000E+00, Mass 1.00837E+02
Total Gibbs energy -4.56490E+04, Enthalpy 1.68797E+04, Volume 0.00000E+00

Component	Moles	W-Fraction	Activity	Potential	Ref.stat
AG	8.4132E-01	9.0000E-01	1.5374E-03	-4.7018E+04	SER
CU	1.5868E-01	1.0000E-01	5.0484E-03	-3.8388E+04	SER

FCC_A1#1 Status ENTERED Driving force 0.0000E+00

```
Number of moles 8.9013E-01, Mass 9.3780E+01
Mass fractions: AG 9.65787E-01 CU 3.42131E-02
```

```
FCC_A1#2                Status ENTERED      Driving force 0.0000E+00
Number of moles 1.0987E-01, Mass 7.0566E+00
Mass fractions: CU 9.74283E-01 AG 2.57167E-02
POLY_3:
```

*) First you see the conditions just as you would have seen them by the command list-conditions before computing equilibrium.

Then you see the main results of the computation, most of which is just a confirmation that the conditions were satisfied. The volume is given as 0 because this database does not include any data on volume.

Then you see the states of the two components, their amounts in mole as well as their mass fractions. Again, that confirms the conditions given. In addition, there is information on the activity and chemical potentials, both given relative to their references, which were here chosen as SER by default. You will later see how you could make different choices.

Then you see the states of the phases. There are two phases in the equilibrium. Both are fcc and are described with the same fundamental equation stored in the database because there is only one fcc phase in this database according to the response to the list-system command. Evidently, POLY was able to identify two fcc phases and decided to call them fcc#1 and fcc#2. Both are given as "entered" because you have not prevented them from taking part in the equilibrium. To the right you see that the driving force for more of them to form from the equilibrium is 0 because they are in equilibrium. The first number for a phase gives the fraction of that phase, measured in moles and then in mass fraction. The next line gives the composition of the phase and you can see that fcc#1 is Ag rich and fcc#2 is Cu rich. Unfortunately, on your screen the text "mass fractions" comes one line before the numbers. Here that text was moved to where it belongs.

You were next asked to use one of the internal variables as a condition. You could for instance use the composition of phase fcc#1, e.g. given as w(fcc#1,Cu). POLY knows the current numbers and you just have to type the symbol. However, you should also remove one of the old conditions and that *must be a realistic choice*. Here it would be reasonable to ask at what temperature the equilibrium fraction or composition of a phase should have the value given as a condition and you should thus remove the condition T=873. By computing equilibrium you should obtain the same equilibrium as before, a fact that you can easily check afterwards by showing the temperature.

```
POLY_3: sh w(fcc,Cu)
W(FCC_A1#1,CU)=3.4213068E-2
POLY_3: s-c w(fcc,Cu)
Value /.0342130676/:
POLY_3: s-c T=none
POLY_3: c-e
Normal POLY minimization, not global
Testing POLY result by global minimization procedure
Calculated          279 grid points in          0 s
  6 ITS, CPU TIME USED  0 SECONDS
POLY_3: sh T
T=873.
POLY_3: exit
CPU time          0 seconds
```

Comments

- 1) Internal variables can very well be used as conditions for equilibrium. However, you should be careful. The set of conditions should be logical.

- 2) Many details about the computed equilibrium can be shown by a single command, list-equilibrium. How the values of some quantities will be expressed is decided by what option is chosen. There are also options controlling what quantities will be shown.

1.3A. The first law of thermodynamics

1 kg of a steel (Fe+0.8 mass% C) is heated from a state of equilibrium at 500°C to a new state of equilibrium at 800°C. The pressure is kept at 1 atm. How much heat was needed for this operation?

Hint

Since there is no change of P, you should use the first law for the enthalpy, which yields $Q = \Delta H - \int VdP = \Delta H$ for constant P. You should thus compute the equilibria for the two sets of conditions, show the enthalpy and take the difference. It does not matter if you don't understand what reference state the values refer to because that does not affect the difference.

Instructions for using T-C

Remember that T-C expresses mass in gram.

Prompts, commands and responses

```
SYS: go da
THERMODYNAMIC DATABASE module running on PC/WINDOWS NT
Current database: TCS Demo Al-Mg-Si Alloys TDB v1

VA DEFINED
TDB_DALMGSI: sw pbin
Current database: TCS Public Binary Alloys TDB v1

VA                /- DEFINED
IONIC_LIQ:Y       L12_FCC                B2_BCC
BCC_B2 REJECTED
TDB_PBIN: def-el Fe C
FE                C DEFINED
TDB_PBIN: l-sys
ELEMENTS, SPECIES, PHASES OR CONSTITUENTS: /CONSTITUENT/:
LIQUID:L         :C FE:
  > This is metallic liquid solution phase, with C-N species
FCC_A1           :FE:C VA:
BCC_A2           :FE:VA C:
HCP_A3           :FE:C VA:
CBCC_A12         :FE:C VA:
CUB_A13          :FE:C VA:
CEMENTITE        :FE:C:
GRAPHITE         :C:
DIAMOND_FCC_A4  :C:
  > This is the Diamond phase for Si-C
TDB_PBIN:
```

*) There are many phases in the database. Try to keep them all. All may not be stable in the equilibrium you are going to compute and should automatically be excluded by POLY. On the other hand, graphite would be stable but by experience one knows that it may be slow to

form. In such a case one should in POLY take an action to stop it from taking part in the computation.

For each phase you can see the constituents in the various sublattices. Six of the phases have two sublattices and for five of them there are carbon and vacancies in the second sublattice. Those phases are interstitial solutions of C in Fe.

```
TDB_PBIN: get
REINITIATING GES5 .....
ELEMENTS .....
SPECIES .....
PHASES .....
PARAMETERS ...
Rewind to read functions          35
FUNCTIONS ....
```

List of references for assessed data

```
90Din 'Alan Dinsdale, SGTE Data for Pure Elements, NPL Report
      DMA(A)195, Rev. August 1990'
85Gus 'P. Gustafson, Scan. J. Metall. vol 14, (1985) p 259-267
      TRITA 0237 (1984); C-Fe'
89Din 'Alan Dinsdale, SGTE Data for Pure Elements, NPL Report
      DMA(A)195, September 1989'
91Din 'A.T. Dinsdale, SGTE Data for Pure Elements, CALPHAD, Vol.15,
      No.4, pp.317-425, (1991)'
```

The list of references can be obtained in the Gibbs Energy System also by the command LIST_DATA and option R

-OK-

```
TDB_PBIN: go pol
```

```
POLY version 3.32, Aug 2001
POLY_3: s-c T=773 P=101325 w(C)=.008 B=1000
POLY_3: c-e
Using global minimization procedure
Calculated          825 grid points in          0 s
Found the set of lowest grid points in          0 s
Calculated POLY solution          0 s, total time  0 s
POLY_3:
```

*) You should check if graphite has taken part in the equilibrium by listing the **status** of the **phases**.

```
POLY_3: l-st p
*** STATUS FOR ALL PHASES
PHASE                STATUS      DRIVING FORCE    MOLES
GRAPHITE              ENTERED      0.00000000E+00  6.65765437E-01
BCC_A2                ENTERED      0.00000000E+00  1.77631060E+01
FCC_A1                ENTERED     -2.30208472E-01  0.00000000E+00
CEMENTITE             ENTERED     -3.08111171E-01  0.00000000E+00
HCP_A3                ENTERED     -4.09496878E-01  0.00000000E+00
DIAMOND_FCC_A4       ENTERED     -7.63728102E-01  0.00000000E+00
CUB_A13               ENTERED     -8.97856313E-01  0.00000000E+00
CBCC_A12              ENTERED     -1.04907383E+00  0.00000000E+00
LIQUID                ENTERED     -1.11457154E+00  0.00000000E+00
```

```
POLY_3:
```

*) Graphite has taken part in the equilibrium. You should **change** the **status** of that **phase** by **suspending** it.

```
POLY_3: ch-st p gra=sus
POLY_3: c-e
Using global minimization procedure
```

```

Calculated          824 grid points in          0 s
Found the set of lowest grid points in          0 s
Calculated POLY solution          0 s, total time  0 s

```

POLY_3: **l-st p**

*** STATUS FOR ALL PHASES

PHASE	STATUS	DRIVING FORCE	MOLES
DIAMOND_FCC_A4	ENTERED	0.00000000E+00	6.65432313E-01
BCC_A2	ENTERED	0.00000000E+00	1.77634400E+01
CEMENTITE	ENTERED	-1.17193213E-01	0.00000000E+00
FCC_A1	ENTERED	-2.19640425E-01	0.00000000E+00
HCP_A3	ENTERED	-4.09208720E-01	0.00000000E+00
CUB_A13	ENTERED	-8.78849990E-01	0.00000000E+00
LIQUID	ENTERED	-9.96328024E-01	0.00000000E+00
CBCC_A12	ENTERED	-1.02536049E+00	0.00000000E+00

SUSPENDED PHASES:

GRAPHITE

POLY_3:

*) This time diamond has formed. It should also be suspended. It had been easier to reject all phases from the beginning and then restore only those phases you like to study, bcc, fcc and cementite. But that requires that you are familiar with the system.

POLY_3: **ch-st p dia=sus**

POLY_3: **c-e**

Using global minimization procedure

```

Calculated          824 grid points in          0 s
Found the set of lowest grid points in          0 s
Calculated POLY solution          0 s, total time  0 s

```

POLY_3: **l-st p**

*** STATUS FOR ALL PHASES

PHASE	STATUS	DRIVING FORCE	MOLES
CEMENTITE	ENTERED	0.00000000E+00	2.66068525E+00
BCC_A2	ENTERED	0.00000000E+00	1.57681870E+01
FCC_A1	ENTERED	-2.09112159E-01	0.00000000E+00
HCP_A3	ENTERED	-4.08888516E-01	0.00000000E+00
CUB_A13	ENTERED	-8.58331365E-01	0.00000000E+00
LIQUID	ENTERED	-9.14933394E-01	0.00000000E+00
CBCC_A12	ENTERED	-9.99646022E-01	0.00000000E+00

SUSPENDED PHASES:

GRAPHITE DIAMOND_FCC_A4

POLY_3: **ent-sym var H773=H;**

POLY_3: **sh H773**

H773=284251.95

POLY_3:

*) Change to the higher temperature and evaluate the enthalpy for that equilibrium. You only need to set the new condition. The old value will automatically be deleted.

POLY_3: **s-c T**

Value /773/: **1073**

POLY_3: **c-e**

Using global minimization procedure

```

Calculated          823 grid points in          0 s
Found the set of lowest grid points in          0 s
Calculated POLY solution          0 s, total time  0 s

```

POLY_3: **sh H**

H=585994.63

POLY_3: **ent-sym var deltaH=H-H773;**

POLY_3: **sh deltaH**

DELTAH=301742.68

POLY_3: **exit**

CPU time 0 seconds

Comments

- 1) DeltaH is given in J for the system, i.e. for 1 kg (B=1000 gram).
- 2) A phase, that for some reason should not take part in the equilibrium, can either be rejected before data are fetched from the database or by suspending it in POLY.
- 3) One can store the current value of a variable by entering a symbol for it.

1.3B. The first law of thermodynamics

A mixture of 2 mol of H₂ and 0.1 mol of O₂ is kept in a very strong cylinder at 25°C. The cylinder has a moveable piston, working against an outside atmosphere of 1 atm. The mixture is ignited and reacts quickly to a state of equilibrium, containing mostly H₂O molecules, and without giving time for any exchange of heat. Calculate the new temperature. In order to simplify the computation you may reject all species except for H₂, O₂ and H₂O.

Hint

The internal energy is not directly affected by an internal reaction. It can be changed only by interactions with the surroundings as described by the first law, $dU=dQ-PdV$. In the present case $dQ=0$ but $dV>0$. It would thus be more convenient to consider the enthalpy, $dH=dU+d(PV)=dQ+VdP=0$ since $dQ=0$ and $dP=0$. One should thus evaluate H for the initial state (which is not at equilibrium) and then search for an equilibrium state that has the same H value.

Instructions for using T-C

- 1) As a default, T-C recognizes H and O as the components also in the gas but H₂, O₂ and H₂O as the species that define the constitution. The formula unit of a gas is defined for one mole of species.
- 2) If the H₂O species is first suspended in POLY, then there can be no reaction and the state will not change if the equilibrium is computed. The only effect on the computation is that the constitution has been evaluated directly from the composition, which was entered. It is then possible to show the initial properties, which POLY always evaluates from the correct constitution.

Prompts, commands and responses

```
SYS: go da
THERMODYNAMIC DATABASE module running on PC/WINDOWS NT
Current database: TCS Demo Al-Mg-Si Alloys TDB v1
```

```
VA DEFINED
TDB_DALMGSI:
```

*) **Switch** to the PSUB database, which is primarily for stoichiometric substances but also has data for a gas with H, N, O and S. It includes a large number of species between those elements.

```
TDB_DALMGSI: sw psub
Current database: TCS Public Pure Substances TDB v1
```

```

VA DEFINED
TDB_PSUB: def-el H O
H O DEFINED
TDB_PSUB: l-sys
ELEMENTS, SPECIES, PHASES OR CONSTITUENTS: /CONSTITUENT/: CONSTITUENT
GAS:G :H H2 O O2 O3 H1O1 H1O2 H2O1 H2O2:
> Gaseous Mixture, using the ideal gas model
H2O_L :H2O1:
H2O2_L :H2O2:
TDB_PSUB: rej p *
GAS:G H2O_L H2O2_L
REJECTED
TDB_PSUB: rest p gas
GAS:G RESTORED
TDB_PSUB:

```

*) The gas phase is the first phase you meet with a constitution controlled not only by elements and by crystallography, which may define sublattices that are fixed for each phase. The constitution of the gas depends on the presence of species, usually molecules, and for various reasons one may like or not like a species to be present in a computation. The aqueous solution is another example. In the present case it is thus necessary to reject all species and then restore H2 and O2 for the first part and then to restore H2O1 for the second part.

```

TDB_PSUB: rej sp *
VA H O
H1O1 H1O2 H2
H2O1 H2O2 O2
O3 REJECTED
TDB_PSUB: rest sp H2 O2
H2 O2 RESTORED
TDB_PSUB: get
REINITIATING GES5 .....
ELEMENTS .....
SPECIES .....
PHASES .....
PARAMETERS ...
FUNCTIONS ....

```

List of references for assessed data

'TCS public data set for gaseous species, stoichiometric solids and liquids in the Cu-Fe-H-N-O-S system.'

The list of references can be obtained in the Gibbs Energy System also by the command LIST_DATA and option R

-OK-

```
TDB_PSUB: go pol
```

POLY version 3.32, Aug 2001

```
POLY_3:
```

*) When setting the conditions you should remember that you must define the values of state variables but for the composition you can only give values of the components. The database for the gas regards the atoms as the components, not the molecules, and a gas with 2 mol of H₂ and 0.1 mol of O₂ should thus be defined as 4 mol of H and 0.2 mole of O. However, POLY offers another possibility. You can set the initial amount of a species and POLY will immediately dissociate it and register the amounts of atoms. If you then give the

initial amount of another species that has an element in common with the first species, the new amount will be added to the previous amount.

```
POLY_3: s-c P=101325 T=298
POLY_3: s-i-a N(H2)=2
POLY_3: s-i-a N(O2)=.1
POLY_3: c-e
Using global minimization procedure
Calculated          137  grid points in          0  s
Found the set of lowest grid points in          0  s
Calculated POLY solution          0 s, total time  0  s
POLY_3: sh H
H=-9.091584
POLY_3:
```

*) It is interesting to note that the H value is very small. In fact it should have been exactly equal to zero if you had used the exact value of 298.15 K for 25°C. The reason is that the database uses the elements in their stable states at 1 atm and 25°C as references (called SER) and the enthalpy of mixing in the gas is zero according to the database that uses the ideal gas model. Now you should go back to the database and add the species H2O1. You can go back by simply typing **b**.

```
POLY_3: b
TDB_PSUB: def-sp H2O1
H2O1 DEFINED
TDB_PSUB: l-sys
ELEMENTS, SPECIES, PHASES OR CONSTITUENTS: /CONSTITUENTS/: CONSTITUENTS
GAS:G          :H2 O2 H2O1:
  > Gaseous Mixture, using the ideal gas model
H2O_L          :H2O1:
TDB_PSUB:
```

*) With H2O1 you also introduced a new phase, water, and like to reject it.

```
TDB_PSUB: rej ph H2O_L
H2O_L REJECTED
TDB_PSUB: get
REINITIATING GES5 .....
ELEMENTS .....
SPECIES .....
PHASES .....
PARAMETERS ...
FUNCTIONS ....
```

List of references for assessed data

```
'TCS public data set for gaseous species, stoichiometric solids and
liquids in the Cu-Fe-H-N-O-S system.'
```

The list of references can be obtained in the Gibbs Energy System also by the command LIST_DATA and option R

-OK-

```
TDB_PSUB:
```

*) You should notice the first line in the response, which says "REINITIATING GES5". That means that what you just did in POLY has been erased and when you again go to POLY you must give all the conditions again and the value of H has been forgotten. You would thus have to type it in by hand. In the present case that is no problem because it is practically 0 and should have been exactly equal to 0 if the correct value T=298.15 had been used. You can again type **b** to get back to POLY.

```
TDB_PSUB: b
POLY version  3.32,  Aug 2001
POLY_3: s-c P=101325 H=0
```

```

POLY_3: s-i-a N(H2)=2
POLY_3: s-i-a N(O2)=.1
POLY_3: c-e
Normal POLY minimization, not global
Testing POLY result by global minimization procedure
Calculated      8409  grid points in      0  s
  13 ITS, CPU TIME USED  0 SECONDS
POLY_3: sh T
T=1094.1025
POLY_3: exit
CPU time          0  seconds

```

Comments

- 1) 1094 K is the end of the adiabatic reaction.
- 2) Sometimes one must make sure that the required phases are included and only those. That is particularly important when one is interested in a metastable equilibrium. For the gas phase the same is true for species. There it is important when one is interested in a restricted equilibrium.
- 3) If one goes back to the database to amend the data, then all information about the preceding session on POLY will be erased.

1.4 Freezing-in conditions

0.5 kg of a white cast iron with 3.5 mass % C (which contains no graphite due to insufficient rate of reaction during fast cooling) has been heat treated at 1100°C to equilibrium (without graphite). Then it is cooled to 800°C. Calculate the amount of “liberated” heat during the cooling under two experimental conditions. (A) The state at 1100°C is completely frozen-in during the cooling. (B) A new state of full equilibrium has been established when 800°C is reached due to slow cooling. Also (C) evaluate the heat evolution if the frozen-in state equilibrates isothermally at 800°C if it were first retained during cooling to 800°C.

Hint

- 1) Suppose the pressure is the same. Then $\Delta H = Q + \int VdP = Q$, where ΔH is the difference of H between the initial and final states.
- 2) After the equilibrium at 1100°C has been computed, you should like to freeze-in the constitution and only change T. Thus, you should not compute equilibrium before evaluating H of the frozen-in state at 800°C. The question is what facility your data bank system has for frozen-in states.
- 3) For (B) it does not matter how close to equilibrium the system was at various temperatures during the cooling because H is a state function.

Instructions for using T-C

After giving the conditions for the state of the system, find the equilibrium at 1100°C and store H of the system as H1100. In order to evaluate H for the frozen-in alloy at 800°C, consider each one of the phases separately at 800°C but don't let them react with each other. Then, add the H values for the phases taking into account their actual amounts.

Prompts, commands and responses

SYS: **go da**
 THERMODYNAMIC DATABASE module running on PC/WINDOWS NT
 Current database: TCS Demo Al-Mg-Si Alloys TDB v1

VA DEFINED
 TDB_DALMGSI: **sw ptern**
 Current database: TCS Public Ternary Alloys TDB v1

VA DEFINED
 TDB_PTERN: **def-el Fe C**
 FE C DEFINED
 TDB_PTERN:

*) You should list the content of the system now defined in order to check that it does not contain a lot of unnecessary data.

TDB_PTERN: **l-sys**
 ELEMENTS, SPECIES, PHASES OR CONSTITUENTS: /CONSTITUENT/:
 LIQUID:L :C FE:
 > This is metallic liquid solution phase, with C species
 BCC_A2 :FE:C VA:
 FCC_A1 :FE:C VA:
 HCP_A3 :FE:VA C:
 CEMENTITE :FE:C:
 M7C3 :FE:C:
 M23C6 :FE:FE:C:
 V3C2 :FE:C:
 GRAPHITE :C:

TDB_PTERN:
) You got many more phases than you are interested in. You could reject one phase after another or reject all () phases and then **restore** the phases you want.

TDB_PTERN: **rej p ***

LIQUID:L	BCC_A2	FCC_A1
HCP_A3	CEMENTITE	M7C3
M23C6	V3C2	GRAPHITE

 REJECTED

TDB_PTERN:
 *) You can see that all the phases are now rejected. You like to **restore** the phases **fcc** and **cementite**, which are required for the equilibrium.

TDB_PTERN: **rest p fcc cem**
 FCC_A1 CEMENTITE RESTORED

TDB_PTERN:
 *) The wanted phases are now shown. Accept the choice and **get** the data.

TDB_PTERN: **get**
 REINITIATING GES5
 ELEMENTS
 SPECIES
 PHASES
 PARAMETERS ...
 Rewind to read functions 21
 FUNCTIONS

List of references for assessed data

The list of references can be obtained in the Gibbs Energy System also by the command LIST_DATA and option R

-OK-
TDB_PTERN: go pol

POLY version 3.32, Aug 2001
POLY_3: s-c P=101325 T=1373 B=500 w(C)=.035
POLY_3: l-c
P=1.01325E5, T=1373, B=500, W(C)=3.5E-2
DEGREES OF FREEDOM 0
POLY_3: c-e
Using global minimization procedure
Calculated 138 grid points in 0 s
Found the set of lowest grid points in 0 s
Calculated POLY solution 0 s, total time 0 s
POLY_3:

*) You may like to store the resulting H value by entering a **symbol**, which will contain the value of a **variable**. The name could be **H1100** and it should contain the current value of **H**. You could just as well store other properties that you may like to use or inspect later on, e.g. Bp(fcc) being the mass (B) for the phase fcc or w(fcc,C) being the the mass fraction (w) of C in the fcc phase.

POLY_3: ent-sym var H1100=H;
POLY_3: ent-sym var Bpfcc=Bp(fcc);
POLY_3: ent-sym var Bpcem=Bp(cem);
POLY_3: ent-sym var wCfcc=w(fcc,C);
POLY_3: ent-sym var wCcem=w(cem,C);
POLY_3:

*) You like to know H of a non-equilibrium state with the same constitution but at 800°C. In principle, that state could be obtained by cooling the alloy to 800°C without the two phases interacting with each other. You should thus consider what happens to each one of the phases when cooled to 800°C. Start with the fcc phase by changing the status of the phase cementite to **dormant**. When giving the conditions you should give the size and composition of the system as the values for fcc at 1100°C. Those values are available directly.

POLY_3: ch-st p cem=dor
POLY_3: s-c T=1073 B=Bpfcc w(C)=wCfcc
POLY_3: c-e
Using global minimization procedure
Calculated 137 grid points in 0 s
Found the set of lowest grid points in 0 s
Calculated POLY solution 0 s, total time 0 s
POLY_3: ent-sym var Hfcc=H;
POLY_3:

*) It should be realized that this value of H is not for 1 mol but for the actual amount of fcc in the frozen-in alloy. Now, consider cementite.

POLY_3: ch-st p cem=ent 1
POLY_3: ch-st p fcc=dor
POLY_3: s-c B=Bpcem w(C)=wCcem
POLY_3: c-e
Using global minimization procedure
Calculated 1 grid points in 0 s
Global minimization failed, error code 2011
Fewer grid points than components
. Using normal POLY minimization.
Testing POLY result by global minimization procedure
Using already calculated grid
8 ITS, CPU TIME USED 0 SECONDS
POLY_3:

*) You can directly evaluate the sum of enthalpies for the two phases. The value for cementite is the current value available under the symbol H.

POLY_3: ent-sym var Hfr=H+Hfcc;

```

POLY_3:
  *) Finally, compute the equilibrium at 800°C.
POLY_3: ch-st p fcc=ent 1
POLY_3: s-c B=500 w(C)=.035
POLY_3: c-e
  Using global minimization procedure
  Calculated          138  grid points in          0  s
  Found the set of lowest grid points in          0  s
  Calculated POLY solution          0 s, total time  0  s
POLY_3: ent-sym var H800=H;
POLY_3: ent-sym var HA=H1100-Hfr;
POLY_3: ent-sym var HB=H1100-H800;
POLY_3: ent-sym var HC=Hfr-H800;
POLY_3: sh HA HB HC
  HA=97398.071
  HB=107518.39
  HC=10120.32
POLY_3: exit
  CPU time          0  seconds

```

Comments

- 1) Properties of frozen-in states can be obtained from POLY but only by considering the phases separately and adding the results.
- 2) The liberated heat for the piece of cast iron is 97 kJ under freezing-in conditions and otherwise 107 kJ.

1.5. Reversible and irreversible processes

Consider a cylinder that can be in contact with any of two heat reservoirs of 20 and 50°C. There is a piston by which the volume can be changed. The cylinder contains pure N₂ gas and is initially at a pressure of 1 atm.

- a) Using the first heat reservoir one compresses the gas slowly and isothermally at 20°C to a pressure of 10 atm.
- b) One continues by compressing adiabatically (i.e., with no heat exchange) until a temperature of 50°C has been reached.
- c) Using the second heat reservoir one releases the pressure to a value P_3 slowly and isothermally at 50°C.
- d) One continues releasing the pressure to 1 atm adiabatically. The pressure P_3 was chosen in such a way that the final temperature was 20°C. It is thus possible to repeat this cycle any number of times.

Evaluate the heat and work received by the system for each one of the four steps. Then add up the net work, W , done by the system on the surroundings and calculate the ratio of that work and the heat drawn from the warm reservoir, Q_3 . Assume that all the four processes are carried out in a reversible fashion.

Hint

Let conditions of the initial state be T_0, P_0 and after the first, second and third step T_0, P_1 , T_2, P_2 and T_2, P_3 , respectively. After the fourth step it is again T_0, P_0 . P_2 and P_3 are not known but may be evaluated because the entropy is not changed by an adiabatic process. For the second step you thus have $S_2 = S_1$, i.e. $S(T_2, P_2) = S(T_0, P_1)$, which yields P_2 . For the fourth step you have $S_3 = S_0$, i.e. $S(T_2, P_3) = S(T_0, P_0)$, which yields P_3 . Denote the heat and work received by the system during the first step by Q_1 and W_1 etc. The heats received during the isothermal steps, i.e. the first and third steps, are according to the definition of entropy for a reversible and isothermal process

$$Q_1 = \int T_0 dS = T_0 (S_1 - S_0) = T_0 (S(T_0, P_1) - S(T_0, P_0)) \text{ and}$$

$Q_3 = \int T_2 dS = T_2 (S_3 - S_2) = T_2 (S(T_2, P_3) - S(T_2, P_2))$. The change of internal energy for the first step is equal to the work plus the heat, which yields $W_1 = U_1 - U_0 - Q_1 = U(T_0, P_1) - U(T_0, P_0) - Q_1$

and for the third step $W_3 = U_3 - U_2 - Q_3 = U(T_2, P_3) - U(T_2, P_2) - Q_3$. For the adiabatic steps

$$Q_2 = 0 \text{ and } Q_4 = 0 \text{ and } W_2 = U_2 - U_1 = U(T_2, P_2) - U(T_0, P_1) \text{ and}$$

$$W_4 = U_0 - U_3 = U(T_0, P_0) - U(T_2, P_3). \text{ You can finally evaluate } W/Q_3 = -\Sigma W_i / Q_3.$$

Instructions for using T-C

In POLY you can use the value of any state variable, e.g. S, as a condition for the equilibrium.

Prompts, commands and responses

```
SYS: go da
THERMODYNAMIC DATABASE module running on PC/WINDOWS NT
Current database: TCS Demo Al-Mg-Si Alloys TDB v1
```

```
VA DEFINED
TDB_DALMGSI: sw psub
Current database: TCS Public Pure Substances TDB v1
```

```
VA DEFINED
TDB_PSUB: def-el N
N DEFINED
TDB_PSUB: rej p *
GAS:G REJECTED
TDB_PSUB: rest p gas
GAS:G RESTORED
TDB_PSUB: get
REINITIATING GES5 .....
ELEMENTS .....
SPECIES .....
PHASES .....
PARAMETERS ...
FUNCTIONS ....
```

List of references for assessed data

```
'TCS public data set for gaseous species, stoichiometric solids and
liquids in the Cu-Fe-H-N-O-S system.'
```

The list of references can be obtained in the Gibbs Energy System also by the command LIST_DATA and option R

```
-OK-
TDB_PSUB: go pol
```

POLY version 3.32, Aug 2001

*) You know the state after the first step, which has $P_1 = 10$ atm. In order to make the properties of that state available, you should compute it.

```
POLY_3: s-c T=293 P=1013250 N=1
```

```
POLY_3: c-e
```

Using global minimization procedure

```
Calculated      8409  grid points in          0  s
```

```
POLY_3: ent-sym var S1=S;
```

```
POLY_3: ent-sym var U1=U;
```

*) You don't know P of the state after the second step but you know $T=323$. The new P can be obtained because S must be unchanged by the adiabatic compression. You should thus replace the condition on P by this condition on S. By just typing S, the current value of S will be used as a condition.

```
POLY_3: s-c T=323 P=none S=
```

```
Value /87.34372665/:
```

```
POLY_3: c-e
```

Normal POLY minimization, not global

Testing POLY result by global minimization procedure

```
Calculated      8409  grid points in          0  s
```

```
6 ITS, CPU TIME USED  0 SECONDS
```

*) Save the new P value.

```
POLY_3: ent-sym var P2=P;
```

```
POLY_3: ent-sym var S2=S;
```

```
POLY_3: ent-sym var U2=U;
```

```
POLY_3: ent-sym var W2=U2-U1;
```

*) P after the third step is obtained in the same way by considering the fourth step, which is also adiabatic. You should thus start by computing the final state, which is equal to the initial state, and then compute the state after the third step using the S value from the final state.

```
POLY_3: s-c S=none T=293 P=101325
```

```
POLY_3: c-e
```

Using global minimization procedure

```
Calculated      8409  grid points in          0  s
```

```
POLY_3: ent-sym var S0=S;
```

```
POLY_3: ent-sym var U0=U;
```

```
POLY_3: ent-sym var Q1=293*(S1-S0);
```

```
POLY_3: ent-sym var W1=U1-U0-Q1;
```

```
POLY_3: s-c T=323 P=none S=
```

```
Value /96.91616004/:
```

```
POLY_3: c-e
```

Normal POLY minimization, not global

Testing POLY result by global minimization procedure

```
Calculated      8409  grid points in          0  s
```

```
6 ITS, CPU TIME USED  0 SECONDS
```

```
POLY_3: ent-sym var P3=P;
```

```
POLY_3: ent-sym var S3=S;
```

```
POLY_3: ent-sym var U3=U;
```

```
POLY_3: ent-sym var Q3=323*(S3-S2);
```

```
POLY_3: ent-sym var W3=U3-U2-Q3;
```

```
POLY_3: ent-sym var W4=U0-U3;
```

```
POLY_3: ent-sym var Ratio=-(W1+W2+W3+W4)/Q3;
```

*) You may inspect all the values by the command **evaluate**.

```
POLY_3: eval
```

```
Name(s):
```

```
S1=85.924577
```

```
U1=-1293.0812
```

```
P2=1013250.
```

```
S2=87.343727
```

```

U2=-981.04929
W2=312.03187
S0=95.49701
U0=-1293.0812
Q1=-2804.723
W1=2804.723
P3=101325.
S3=96.91616
U3=-981.04929
Q3=3091.896
W3=-3091.896
W4=-312.03187
RATIO=9.2879257E-2
POLY_3: exit
CPU time          0 seconds

```

Comments

- 1) The efficiency, i.e., the part of the heat, drawn from the warm reservoir, that is recovered as work was evaluated as is 0.09288.
- 2) This is actually the Carnot cycle and in Section 1.6 it will be shown that the maximum efficiency can be calculated from $(T_{\text{high}}-T_{\text{low}})/T_{\text{high}}=30/(50+273.15)=0.09284$. The difference is caused by approximating 273.15 by 273. The result does not at all depend on the gas being ideal but that simplified your calculations.

1.6. The second law of thermodynamics

When you in problem 1.3B calculated the final temperature when a gas mixture of 2 mole of H_2 and 0.1 mole of O_2 was reacting adiabatically after being ignited, you relied on an algorithm hidden inside the program. Check that the final state was really the state expected from the second law. To make a decisive test you may now use 2 mole of H_2 and 1 mole of O_2 but again under a constant pressure of 1 atm and with an initial temperature of 25°C . The adiabatic temperature would then be very high and, although all of H_2 and O_2 could in principle form H_2O , some would be dissociated into H_2 and O_2 and one could easily introduce some deviation from the equilibrium constitution at one temperature by first computing the equilibrium at a different temperature.

Hint

- 1) Equilibria are usually computed by minimizing a function called Gibbs energy. However, it applies only under constant T and P and in the present case T varies during the reaction.
- 2) For a closed system the second law gives $dS=dQ/T+d_{\text{ip}}S$ and for adiabatic conditions $dS=d_{\text{ip}}S$. In Problem 1.5 you considered reversible, adiabatic processes, for which $d_{\text{ip}}S=0$ and the entropy does not change. The present process is adiabatic but not reversible because the formation of H_2O molecules occurs spontaneously. For each H_2O molecule formed, the temperature will rise and that should continue until $d_{\text{ip}}S/dN_{\text{H}_2\text{O}}=0$, i.e., until S reaches a maximum where $dS/dN_{\text{H}_2\text{O}}=0$. The problem is thus to examine if the amount of H_2O , formed when the final temperature is reached, gives a higher S value than any other amount of H_2O would do if evaluated at the temperature reached when that amount has formed. You should test a slightly higher amount and a slightly lower.

- 3) The SER reference is based on the enthalpy at 298 K and entropy at 0 K (= zero) for the pure elements in their stable states, e.g. H₂ and O₂, and at 25°C and 1 bar. Furthermore, in ideal gases there is no heat of mixing. Thus H=0 for our initial gas mixture because no H₂O has yet formed. The final state under adiabatic conditions and constant pressure can thus be obtained from the condition H=0.

Instructions for using T-C

- 1) A convenient way to introduce a deviation from the state of equilibrium is to compute the equilibrium at a slightly different temperature. The problem is then to find the temperature where that constitution would give the prescribed H value. You could not use POLY because it cannot handle non-equilibrium states. Instead, go to the tabulation module, tabulate H for a range of temperatures and find the temperature where H has the initial value. At the same temperature you can read the S value and compare with the stored S value.
- 2) POLY can give the value of a property per mole of units of the components, X_m, and for the whole system, taking the size into account, X. The two quantities will be identical if the system is defined with N=1 because N is often the number of atoms. POLY can also give the properties for other measures of the size, e.g. per mass or volume. A further alternative is to get the properties per mole of the formula unit used in the model as it is stored in the database, X_f. That is sometimes useful when working with both POLY and the tabulation module because the tabulation module always gives properties per mole of formula unit. However, it must be remembered that the content in one formula unit may change if the composition changes, e.g. for an interstitial solution phase, or when a molecular reaction occurs inside the gas phase and results in a change in the number of molecules, i.e. species. Sometimes one would thus have to transform the value of a property from formula unit to atom. The present case gives an example.

Prompts, commands and responses

```

SYS: go da
THERMODYNAMIC DATABASE module running on PC/WINDOWS NT
Current database: TCS Demo Al-Mg-Si Alloys TDB v1

VA DEFINED
TDB_DALMGSI: sw psub
Current database: TCS Public Pure Substances TDB v1

VA DEFINED
TDB_PSUB: def-el H O
H O DEFINED
TDB_PSUB: rej p *
GAS:G H2O_L H2O2_L
REJECTED
TDB_PSUB: rest p gas
GAS:G RESTORED
TDB_PSUB:
*) For the gas one must also decide what species to include.
TDB_PSUB: rej sp *
VA H O
H1O1 H1O2 H2
H2O1 H2O2 O2
O3 REJECTED
TDB_PSUB: rest sp H2 O2 H2O1
H2 O2 H2O1
RESTORED

```

```
TDB_PSUB: get
REINITIATING GES5 .....
ELEMENTS .....
SPECIES .....
PHASES .....
PARAMETERS ...
FUNCTIONS ....
```

List of references for assessed data

```
'TCS public data set for gaseous species, stoichiometric solids and
liquids in the Cu-Fe-H-N-O-S system.'
```

The list of references can be obtained in the Gibbs Energy System also by the command LIST_DATA and option R

-OK-

```
TDB_PSUB: go pol
```

```
POLY version 3.32, Aug 2001
```

```
POLY_3:
```

*) Instead of the usual condition on T you should use the condition on H, which should have the value 0 if the initial temperature is 25°C and the gas only contains the stable species of the elements.

```
POLY_3: s-c P=101325 H=0
```

*) It may be tempting to use N(H2)=2 as a condition but that kind of condition can be used only if H₂ has been chosen as a component. As a default, the monatomic elements are used as components and N(H)=4 would be a proper condition if a different choice is not made. If the initial content is given through selected species, one can use a special command “set-initial amount”, applied to each one of the species. For each one POLY will dissociate the species in the defined components and add the new amount of each component (element) to whatever amount has already been entered. In the present, simple case one could thus give the commands s-i-a N(H2)=2 and s-i-a N(O2)=1.

```
POLY_3: s-i-a N(H2)=2
```

```
POLY_3: s-i-a N(O2)=1
```

```
POLY_3: c-e
```

Normal POLY minimization, not global

```
*** ERROR 1614 IN QTHISS
```

```
*** CONDITIONS CAN NOT BE FULLFILLED
```

Give the command INFO TROUBLE for help

*) The first way to try to overcome this difficulty is to try again.

```
POLY_3: c-e
```

Normal POLY minimization, not global

Testing POLY result by global minimization procedure

```
Calculated      8409  grid points in          0  s
```

```
6 ITS, CPU TIME USED  0 SECONDS
```

```
POLY_3: sh T y(gas,H2)
```

```
T=3510.6826
```

```
Y(GAS,H2)=0.28710262
```

*) Save the molar entropy of this state of equilibrium, to be compared with later on. Then you should introduce an equilibrium constitution from a slightly different temperature, e.g 3400 K.

```
POLY_3: ent-sym var Smeq=Sm;
```

```
POLY_3: s-c T=3400 H=none
```

```
POLY_3: c-e
```

Using global minimization procedure

```
Calculated      8409  grid points in          0  s
```

Found the set of lowest grid points in 0 s
 Calculated POLY solution 0 s, total time 0 s
 POLY_3: **sh y(gas,H2)**
 Y(GAS,H2)=0.25240009

*) Go to TAB to evaluate the adiabatic temperature for a gas with this constitution.

POLY_3: **go tab**
 TAB: **tab-sub gas**
 FRACTION OF CONSTITUENT (RETURN FOR PROMPT):
 H2 /.2524000862/:

*) Here you can see that the H₂ content from the state of equilibrium is introduced.

H2O1 /.6213998707/:
 Pressure /101325/:
 Low temperature limit /298.15/: **3400**
 High temperature limit /2000/: **3800**
 Step in temperature /100/: **20**
 Output file /SCREEN/:

O U T P U T F R O M T H E R M O - C A L C
 2007. 4.21 19.51.49

Phase : GAS Pressure : 101325.00
 Specie: *

```
*****
  T      Cp      H      S      G
  (K)    (Joule/K) (Joule) (Joule/K) (Joule)
*****
3400.00  5.11140E+01 -1.58737E+04  2.79248E+02 -9.65318E+05
3420.00  5.11729E+01 -1.48509E+04  2.79548E+02 -9.70906E+05
3440.00  5.12314E+01 -1.38268E+04  2.79847E+02 -9.76500E+05
3460.00  5.12896E+01 -1.28016E+04  2.80144E+02 -9.82100E+05
3480.00  5.13475E+01 -1.17753E+04  2.80440E+02 -9.87706E+05
3500.00  5.14051E+01 -1.07477E+04  2.80734E+02 -9.93317E+05
3520.00  5.14624E+01 -9.71905E+03  2.81027E+02 -9.98935E+05
3540.00  5.15193E+01 -8.68923E+03  2.81319E+02 -1.00456E+06
3560.00  5.15760E+01 -7.65828E+03  2.81609E+02 -1.01019E+06
3580.00  5.16323E+01 -6.62620E+03  2.81898E+02 -1.01582E+06
3600.00  5.16883E+01 -5.59299E+03  2.82186E+02 -1.02146E+06
3620.00  5.17441E+01 -4.55866E+03  2.82473E+02 -1.02711E+06
3640.00  5.17995E+01 -3.52323E+03  2.82758E+02 -1.03276E+06
3660.00  5.18546E+01 -2.48669E+03  2.83042E+02 -1.03842E+06
3680.00  5.19095E+01 -1.44904E+03  2.83325E+02 -1.04408E+06
3700.00  5.19640E+01 -4.10309E+02  2.83606E+02 -1.04975E+06
3720.00  5.20173E+01  6.29503E+02  2.83887E+02 -1.05543E+06
3740.00  5.20703E+01  1.67038E+03  2.84166E+02 -1.06111E+06
3760.00  5.21231E+01  2.71231E+03  2.84443E+02 -1.06680E+06
3780.00  5.21756E+01  3.75530E+03  2.84720E+02 -1.07249E+06
3800.00  5.22279E+01  4.79934E+03  2.84996E+02 -1.07818E+06
*****
```

*) You are looking for T where H=0. It is closer to 3700 than 3720 K. Try again.

TAB: **tab-sub gas**
 FRACTION OF CONSTITUENT (RETURN FOR PROMPT):
 H2 /.2524000862/:
 H2O1 /.6213998707/:
 Pressure /101325/:
 Low temperature limit /3400/: **3700**
 High temperature limit /3800/: **3710**
 Step in temperature /20/: **1**
 Output file /SCREEN/:

OUTPUT FROM THERMO - CALC
2007. 4.21 19.51.49

Phase : GAS Pressure : 101325.00
Specie: *

```
*****
T      Cp      H      S      G
(K)    (Joule/K) (Joule) (Joule/K) (Joule)
*****
3700.00 5.19640E+01 -4.10309E+02 2.83606E+02 -1.04975E+06
3701.00 5.19666E+01 -3.58344E+02 2.83620E+02 -1.05004E+06
3702.00 5.19693E+01 -3.06376E+02 2.83634E+02 -1.05032E+06
3703.00 5.19720E+01 -2.54406E+02 2.83648E+02 -1.05060E+06
3704.00 5.19746E+01 -2.02432E+02 2.83662E+02 -1.05089E+06
3705.00 5.19773E+01 -1.50456E+02 2.83676E+02 -1.05117E+06
3706.00 5.19800E+01 -9.84776E+01 2.83690E+02 -1.05146E+06
3707.00 5.19826E+01 -4.64963E+01 2.83704E+02 -1.05174E+06
3708.00 5.19853E+01 5.48767E+00 2.83719E+02 -1.05202E+06
3709.00 5.19880E+01 5.74743E+01 2.83733E+02 -1.05231E+06
3710.00 5.19906E+01 1.09464E+02 2.83747E+02 -1.05259E+06
*****
```

*) Now you can see that T should be closer to 3708 than 3707 K

TAB: **tab-sub gas**

FRACTION OF CONSTITUENT (RETURN FOR PROMPT):

H2 /.2524000862/:

H2O1 /.6213998707/:

Pressure /101325/:

Low temperature limit /3700/: **3707**

High temperature limit /3750/: **3708**

Step in temperature /1/: **.1**

Output file /SCREEN/:

OUTPUT FROM THERMO - CALC
2007. 4.21 19.51.49

Phase : GAS Pressure : 101325.00
Specie: *

```
*****
T      Cp      H      S      G
(K)    (Joule/K) (Joule) (Joule/K) (Joule)
*****
3707.00 5.19826E+01 -4.64963E+01 2.83704E+02 -1.05174E+06
3707.10 5.19829E+01 -4.12980E+01 2.83706E+02 -1.05177E+06
3707.20 5.19832E+01 -3.60997E+01 2.83707E+02 -1.05180E+06
3707.30 5.19834E+01 -3.09014E+01 2.83709E+02 -1.05182E+06
3707.40 5.19837E+01 -2.57030E+01 2.83710E+02 -1.05185E+06
3707.50 5.19840E+01 -2.05047E+01 2.83712E+02 -1.05188E+06
3707.60 5.19842E+01 -1.53062E+01 2.83713E+02 -1.05191E+06
3707.70 5.19845E+01 -1.01078E+01 2.83714E+02 -1.05194E+06
3707.80 5.19848E+01 -4.90934E+00 2.83716E+02 -1.05197E+06
3707.90 5.19850E+01 2.89153E-01 2.83717E+02 -1.05199E+06
3708.00 5.19853E+01 5.48767E+00 2.83719E+02 -1.05202E+06
*****
```

*) The temperature should be close to 3707.9 K and the entropy at that temperature is about 283.716. However, this is for 1 formula unit. Go to POLY and evaluate the entropy for 1 mole of atoms. First you must evaluate the **number of atoms per formula unit**, Naperf. Soon you will see that it is an advantage that it was entered as a function.

TAB: **b**

POLY_3: **ent-sym fun Naperf=2*y(gas,H2)+2*y(gas,O2)+3*y(gas,H2O1);**

POLY_3: **ent-sym var Sml=2.83717E+02/Naperf;**

*) Now, examine the effect of the equilibrium constitution from a slightly higher temperature.

POLY_3: **s-c T=3600**

POLY_3: **c-e**

Using global minimization procedure

Calculated 8409 grid points in 0 s

Found the set of lowest grid points in 0 s

Calculated POLY solution 0 s, total time 0 s

POLY_3: **sh y(gas,H2)**

Y(GAS,H2)=0.31480022

POLY_3: **go tab**

TAB: **tab-sub gas**

FRACTION OF CONSTITUENT (RETURN FOR PROMPT):

H2 /.3148002234/:

H2O1 /.527799665/:

Pressure /101325/:

Low temperature limit /3707/: **3200**

High temperature limit /3708/: **3500**

Step in temperature /.1/: **20**

Output file /SCREEN/:

O U T P U T F R O M T H E R M O - C A L C
2007. 4.21 19.51.50

Phase : GAS Pressure : 101325.00

Specie: *

T (K)	Cp (Joule/K)	H (Joule)	S (Joule/K)	G (Joule)
3200.00	4.86922E+01	-7.12688E+03	2.71444E+02	-8.75749E+05
3220.00	4.87512E+01	-6.15244E+03	2.71748E+02	-8.81181E+05
3240.00	4.88099E+01	-5.17683E+03	2.72050E+02	-8.86619E+05
3260.00	4.88683E+01	-4.20005E+03	2.72350E+02	-8.92063E+05
3280.00	4.89263E+01	-3.22210E+03	2.72650E+02	-8.97513E+05
3300.00	4.89840E+01	-2.24300E+03	2.72947E+02	-9.02969E+05
3320.00	4.90415E+01	-1.26274E+03	2.73243E+02	-9.08430E+05
3340.00	4.90986E+01	-2.81341E+02	2.73538E+02	-9.13898E+05
3360.00	4.91554E+01	7.01199E+02	2.73831E+02	-9.19372E+05
3380.00	4.92119E+01	1.68487E+03	2.74123E+02	-9.24852E+05
3400.00	4.92681E+01	2.66967E+03	2.74414E+02	-9.30337E+05
3420.00	4.93241E+01	3.65560E+03	2.74703E+02	-9.35828E+05
3440.00	4.93797E+01	4.64263E+03	2.74991E+02	-9.41325E+05
3460.00	4.94350E+01	5.63078E+03	2.75277E+02	-9.46828E+05
3480.00	4.94901E+01	6.62003E+03	2.75562E+02	-9.52336E+05
3500.00	4.95448E+01	7.61038E+03	2.75846E+02	-9.57850E+05

*) You can see that H=0 for a temperature closer to 3340 than 3360 K.

TAB: **tab-sub gas**

FRACTION OF CONSTITUENT (RETURN FOR PROMPT):

H2 /.3148002234/:

```
H2O1 /.527799665/:
Pressure /101325/:
Low temperature limit /3200/: 3345
High temperature limit /3500/: 3350
Step in temperature /20/: .5
Output file /SCREEN/:
```

```
      O U T P U T   F R O M   T H E R M O - C A L C
      2007. 4.21                               19.51.50
```

```
Phase : GAS                Pressure :      101325.00
Specie: *
```

```
*****
      T          Cp          H          S          G
      (K)        (Joule/K)   (Joule)   (Joule/K)   (Joule)
*****
3345.00  4.91128E+01  -3.58130E+01  2.73611E+02  -9.15266E+05
3345.50  4.91142E+01  -1.12562E+01  2.73619E+02  -9.15403E+05
3346.00  4.91157E+01  1.33013E+01  2.73626E+02  -9.15540E+05
3346.50  4.91171E+01  3.78594E+01  2.73633E+02  -9.15677E+05
3347.00  4.91185E+01  6.24183E+01  2.73641E+02  -9.15813E+05
3347.50  4.91199E+01  8.69779E+01  2.73648E+02  -9.15950E+05
3348.00  4.91213E+01  1.11538E+02  2.73655E+02  -9.16087E+05
3348.50  4.91228E+01  1.36099E+02  2.73663E+02  -9.16224E+05
3349.00  4.91242E+01  1.60661E+02  2.73670E+02  -9.16361E+05
3349.50  4.91256E+01  1.85223E+02  2.73677E+02  -9.16498E+05
```

*) You may estimate the temperature for $H=0$ to 3345.7 K and there the entropy is 273.622.

When now evaluating the molar entropy it is not necessary again to write the expression for Naperf because being a function it will always be evaluated for the current equilibrium.

```
TAB: b
POLY_3: ent-sym var Sm2=2.73622E+02/Naperf;
POLY_3: eval
Name(s):
  SMEQ=108.27891
  NAPERF=2.5277997
  SM1=108.2311
  SM2=108.24513
POLY_3:
POLY_3: exit
CPU time          0 seconds
```

Comments

- 1) Both Sm_1 and Sm_2 are lower than S_{meq} , supporting the accuracy of the optimisation procedure hidden inside POLY even for a case where T is not constant.
- 2) There may be numerical difficulties to find the equilibrium constitution if a condition requires that a state variable should be zero. Sometimes you get an error message after the compute-equilibrium command.
- 3) Numerical difficulties in POLY may be overcome by using better start values for the constitution. POLY can provide you with such values as default values in response to the command set-start-constitution. The quickest way is to try again. It works sometimes.

- 4) The amount of species can be used as conditions using the command "set-initial-amount". A mistake (e.g. an incorrect value of N(H₂O)) can be corrected either by setting the contents of all the components equal to zero, e.g. N(H)=0 and N(O)=0 or by entering the negative of the incorrect value (e.g. "s-i-a N(H₂O)=-2"), and then entering the correct value.
- 5) You should remember that TAB gives values per mole of formula units.
- 6) The present problem concerned non-equilibrium states and so did Problem 1.4. In that case it was possible to apply POLY because the deviation from equilibrium could be represented by the fraction of each phase and the internal conditions of each phase were not a sign of non-equilibrium for that phase alone. POLY could thus be applied to each phase separately. In the present case the non-equilibrium concerned the internal constitution of a single phase and POLY could not be applied because it requires that the state of equilibrium be computed before it can yield any thermodynamic information.

1.7. Condition of internal equilibrium

Most programs for the computation of equilibria work by minimizing a function G , called Gibbs energy, under constant T and P . So far you have only learned that equilibrium is found where $d_{ip}S$ has a maximum. Use that criterion in order to check that the state obtained from an available program is actually a state where $d_{ip}S$ has a maximum. Make the test with an Al alloy with 50 mass% Si at 500°C and 1 atm.

Hint

The constitution of the alloy is varied when the program searches for the minimum of G . You should use the program to evaluate the equilibrium and examine how the equilibrium constitution is described. Then you can vary the constitution around the equilibrium constitution and check that $d_{ip}S$ has a maximum there. You know that $d_{ip}S = dS - dQ/T$. From the first law you know $dU = dQ - PdV$ and thus $d_{ip}S = dS - dQ/T = dS - (dU + PdV)/T$. For constant P you get $d_{ip}S = d[S - (U + PV)/T]$. You should thus define a function $FUNC = S - (U + PV)/T$ and test if the equilibrium state obtained from the program actually is a maximum of $FUNC$ when the constitution is varied around the equilibrium constitution. Since you are going to compare situations at the same temperature, 500°C, you could just as well multiply by T and define the function as $FUNC = ST - U - PV$.

Instructions for using T-C

You are going to consider situations of non-equilibrium and sometimes one has to use the tabulation module for such cases. However, in the present case, as in Problem 1.4, it is possible to use POLY by considering one phase at a time.

Prompts, commands and responses

```
SYS: go da
THERMODYNAMIC DATABASE module running on PC/WINDOWS NT
Current database: TCS Demo Al-Mg-Si Alloys TDB v1

VA DEFINED
TDB_DALMGSI: sw pbin
Current database: TCS Public Binary Alloys TDB v1
```

```

VA                /-  DEFINED
IONIC_LIQ:Y       L12_FCC                B2_BCC
BCC_B2 REJECTED
TDB_PBIN: def-el Al Si
AL                SI  DEFINED
TDB_PBIN: l-sys
ELEMENTS, SPECIES, PHASES OR CONSTITUENTS: /CONSTITUENT/:  CONSTITUENT
LIQUID:L         :SI AL:
  > This is metallic liquid solution phase, with C-N species
FCC_A1           :AL:VA:
BCC_A2           :AL:VA:
HCP_A3           :AL:VA:
DIAMOND_FCC_A4  :SI:
  > This is the Diamond phase for Si-C
ALCU_THETA      :AL:AL:
AL3NI2          :AL:AL:VA:
TIAL            :AL:AL:
TI3AL           :AL:AL:
TDB_PBIN: rej p *
LIQUID:L         FCC_A1                BCC_A2
HCP_A3           DIAMOND_FCC_A4        ALCU_THETA
AL3NI2           TIAL                 TI3AL
REJECTED
TDB_PBIN: rest p liq diam
*** DIAM INPUT IGNORED
LIQUID:L RESTORED
TDB_PBIN:

```

*) It is evident that "diam" was not accepted for the Si phase. From the list you can see its full name. Try a longer abbreviation.

```

TDB_PBIN: rest p diam fcc
DIAMOND_FCC_A4  RESTORED
TDB_PBIN: get
REINITIATING GES5 .....
ELEMENTS .....
SPECIES .....
PHASES .....
PARAMETERS ...
Rewind to read functions          2
FUNCTIONS ....

```

List of references for assessed data

```

90Din 'Alan Dinsdale, SGTE Data for Pure Elements, NPL Report
      DMA(A)195, Rev. August 1990'
91Din 'A.T. Dinsdale, SGTE Data for Pure Elements, CALPHAD, Vol.15,
      No.4, pp.317-425, (1991)'

```

The list of references can be obtained in the Gibbs Energy System also by the command LIST_DATA and option R

-OK-

```
TDB_PBIN: go pol
```

```

POLY version 3.32, Aug 2001
POLY_3: s-c T=773 P=101325 N=1 x(Si)=.5
POLY_3: c-e
Using global minimization procedure
Calculated          138 grid points in          0 s
Found the set of lowest grid points in          0 s
Calculated POLY solution          0 s, total time  0 s

```

POLY_3:

*) The ordinary thermodynamic quantities for the state of equilibrium are now available but the expression derived in the Hint must be calculated from those available. That is done by entering a symbol for a variable or function with the value of the expression. The value of such a variable will never change unless it is entered again. A function will be evaluated from the current equilibrium every time it is called upon. It cannot be entered again unless it is first deleted.

POLY_3: **ent-sym fun FUNC=S*T-U-P*V;**

POLY_3: **ent-sym var FUNCEq=FUNC;**

POLY_3: **sh FUNCEq**

FUNCEQ=23650.72

POLY_3: **l-e**

OUTPUT TO SCREEN OR FILE /SCREEN/:

Options /VWCS/:

*) The option C will give you the composition of phases. Option N will in addition give you the constitution of phases, i.e., the amounts of species for a gas and site-fraction for phases with sublattices.

Options /VWCS/: **XN**

Output from POLY-3, equilibrium = 1, label A0 , database: PBIN

Conditions:

T=773, P=1.01325E5, N=1, X(SI)=0.5

DEGREES OF FREEDOM 0

Temperature 773.00 K (499.85 C), Pressure 1.013250E+05

Number of moles of components 1.00000E+00, Mass 2.75340E+01

Total Gibbs energy -2.36507E+04, Enthalpy 1.78647E+04, Volume 0.00000E+00

Component	Moles	M-Fraction	Activity	Potential	Ref.stat
AL	5.0000E-01	5.0000E-01	1.5148E-02	-2.6929E+04	SER
SI	5.0000E-01	5.0000E-01	4.2013E-02	-2.0372E+04	SER

LIQUID Status ENTERED Driving force 0.0000E+00

Number of moles 5.0711E-01, Mass 1.3691E+01 Mole fractions:

AL 9.85988E-01 SI 1.40122E-02

Constitution:

AL 9.85988E-01 SI 1.40122E-02

DIAMOND_FCC_A4 Status ENTERED Driving force 0.0000E+00

Number of moles 4.9289E-01, Mass 1.3843E+01 Mole fractions:

SI 1.00000E+00 AL 0.00000E+00

Constitution:

SI 1.00000E+00

POLY_3:

*) The state of equilibrium consists of two phases, liq and Si with the diamond structure. There are several internal variables in this list. However, only one can be chosen as independent. It is not necessary to decide which one if the deviation from equilibrium is simply defined by using the state of equilibrium from different temperatures, e.g. 770 and 776 K.

POLY_3: **s-c T=770**

POLY_3: **c-e**

Using global minimization procedure

Calculated 138 grid points in 0 s

Found the set of lowest grid points in 0 s

Calculated POLY solution 0 s, total time 0 s

POLY_3:

*) You will soon need the constitution of this state. Store it as variables.

POLY_3: **ent-sym var Npliq=Np(liq);**

```
POLY_3: ent-sym var Npdia=Np(dia);
POLY_3: ent-sym var xSiliq=x(liq,Si);
POLY_3: ent-sym var xSidia=x(dia,Si);
POLY_3:
```

*) At 773 K you should now evaluate the function for the constitution obtained by a computation at 770 K. That can be done with the tabulation module but also with POLY. In both cases one must consider each phase separately and taking into account its amount. This time you can use POLY as in Problem 1.4. See the last Comment to Problem 1.6 for which it was not possible to use POLY. Start by suspending the Si phase and give the liquid the composition characteristic of 770 K.

```
POLY_3: ch-st p dia=sus
POLY_3: s-c T=773 N=Npliq x(Si)=xSiliq
POLY_3: c-e
Using global minimization procedure
Calculated          137  grid points in          0  s
Found the set of lowest grid points in          0  s
Calculated POLY solution          0 s, total time  0  s
POLY_3: ent-sym var FUNCliq=FUNC;
POLY_3:
```

*) Do the same for the Si phase.

```
POLY_3: ch-st p dia=ent 1
POLY_3: ch-st p liq=sus
POLY_3: s-c N=Npdia x(Si)=xSidia
POLY_3: c-e
Using global minimization procedure
Calculated          1  grid points in          0  s
Global minimization failed, error code          2011
Fewer grid points than components
. Using normal POLY minimization.
Testing POLY result by global minimization procedure
Using already calculated grid
6 ITS, CPU TIME USED  0 SECONDS
POLY_3: ent-sym var FUNClow=FUNC+FUNCliq;
POLY_3:
```

*) Now, go to a constitution characteristic of a higher temperature.

```
POLY_3: s-c T=776 N=1 x(Si)=.5
POLY_3: ch-st p liq=ent 1
POLY_3: c-e
Using global minimization procedure
Calculated          138  grid points in          0  s
Found the set of lowest grid points in          0  s
Calculated POLY solution          0 s, total time  0  s
POLY_3: ent-sym var Npliq=Np(liq);
POLY_3: ent-sym var Npdia=Np(dia);
POLY_3: ent-sym var xSiliq=x(liq,Si);
POLY_3: ent-sym var xSidia=x(dia,Si);
POLY_3: ch-st p dia=sus
POLY_3: s-c T=773 N=Npliq x(Si)=xSiliq
POLY_3: c-e
Using global minimization procedure
Calculated          137  grid points in          0  s
Found the set of lowest grid points in          0  s
Calculated POLY solution          0 s, total time  0  s
POLY_3: ent-sym var FUNCliq=FUNC;
POLY_3: ch-st p dia=ent 1
POLY_3: ch-st p liq=sus
POLY_3: s-c N=Npdia x(Si)=xSidia
POLY_3: c-e
Using global minimization procedure
```

```

Calculated          1 grid points in          0 s
Global minimization failed, error code          2011
Fewer grid points than components
. Using normal POLY minimization.
Testing POLY result by global minimization procedure
Using already calculated grid
  6 ITS, CPU TIME USED  0 SECONDS
POLY_3: ent-sym var FUNChi=FUNC+FUNCliq;
POLY_3: sh FUNClow FUNCeq FUNCi
  FUNCLOW=23650.698
  FUNCEQ=23650.72
  FUNCHI=23650.698
POLY_3: exit
CPU time          0 seconds

```

Comments

You find that FUNC is higher for the state of equilibrium than for the non-equilibrium states at the same temperature, which actually means that $d_{ip}S=0$ for that constitution if T, P and the composition are constant. In fact, this function is identical to Gibbs energy, G.

1.8. Driving force

Consider an Fe alloy with 30 mol% Cr at 650°C and 1 atm. It is in a state of homogeneous bcc and is supersaturated with respect to the sigma phase of an approximate 50/50 composition. Evaluate the driving force for the precipitation of sigma, using the basic definition $D = T \cdot d_{ip}S / d\xi$.

Hint

From Problem 1.7 you know that $d_{ip}S=d[S-(U+PV)/T]$. You may thus evaluate the function $S-(U+PV)/T$ for the initial state and for a state with a minute amount, ΔN , of sigma, measured per mole. Take the difference, multiply by T and divide by ΔN because the amount of sigma expressed in mole can be used to represent the progress of the process, ξ .

Instructions for using T-C

The initial homogeneous bcc state is not a state of equilibrium but may be treated as such if sigma is dormant or suspended. The same goes for the bcc matrix after some sigma has precipitated but then one may have to consider that the amount of bcc is less than in the first case. Similarly, the properties of the minute amount of sigma can be treated by POLY if bcc is made dormant.

Prompts, commands and responses

```

SYS: go da
THERMODYNAMIC DATABASE module running on PC/WINDOWS NT
Current database: TCS Demo Al-Mg-Si Alloys TDB v1

```

```

VA  DEFINED
TDB_DALMGSI:

```

*) This time you should **switch** to the database **tern** that has a good description of a few ternary systems, among them Fe-Cr-C.

```

TDB_DALMGSI: sw DFeCrC
Current database: TCS Demo Fe-Cr-C Alloys TDB v1

```

```

VA DEFINED
TDB_DALMGSI: def-el Fe Cr
FE CR DEFINED
TDB_DALMGSI:

```

*) You may expect that the database will contain more phases than those you are interested in. You may thus **reject** all(*) phases and then **restore** the phases **bcc** and **sigma** and finally **get data**.

```

TDB_DALMGSI: rej p *
LIQUID:L BCC_A2 FCC_A1
HCP_A3 SIGMA REJECTED
TDB_DALMGSI: rest p bcc sigma
BCC_A2 SIGMA RESTORED
TDB_DALMGSI: get
REINITIATING GES5 .....
ELEMENTS .....
SPECIES .....
PHASES .....
PARAMETERS ...
Rewind to read functions 25
FUNCTIONS ....

```

List of references for assessed data

The list of references can be obtained in the Gibbs Energy System also by the command LIST_DATA and option R

-OK-

```
TDB_PTERN: go pol
```

```

POLY version 3.32, Aug 2001
POLY_3: s-c T=923 P=101325 N(Fe)=0.7 N(Cr)=0.3
POLY_3:

```

*) Sigma should not be present in the initial state. You should thus **change** the status of that phase to **dormant** before computing the equilibrium, which will be a restricted equilibrium.

```

POLY_3: ch-st p sigma=dor
POLY_3: c-e
Using global minimization procedure
Calculated 137 grid points in 0 s
Found the set of lowest grid points in 0 s
Calculated POLY solution 0 s, total time 0 s
POLY_3:

```

*) Now you like to define the special function discussed in the hint. However, it may be advantageous to multiply by T because $D = T \cdot d_{ip} S / d\xi$. You should thus **enter** a **symbol** for a **function** you may call **FUNC** and store the present value of ST-U-PV in it.

```

POLY_3: ent-sym fun FUNC=S*T-U-P*V;
POLY_3:

```

*) This function will later be available to you when considering other restricted equilibria. In order to retain the present value you should store it in a variable that may be called FUinit. The value of FUNC for other states you may later store as other variables.

```

POLY_3: ent-sym var FUini=FUNC;
POLY_3:

```

*) It may be interesting to list the **status** of the **phases**.

```

POLY_3:l-st p
*** STATUS FOR ALL PHASES
PHASE STATUS DRIVING FORCE moles
BCC_A2 ENTERED 0.00000000E+00 1.00000000E+00

```

```
SIGMA          DORMANT      1.58194990E-02
```

```
POLY_3:
```

*) For the dormant phase, sigma, POLY gives a quantity called driving force. It may be interesting to save it for later comparisons. POLY recognizes it by the symbol DGm(phase name).

```
POLY_3: ent-sym var Drforc=DGm(sigma);
```

```
POLY_3:
```

*) Next, allow 0.001 mole of sigma to precipitate and consider two subsystems. The first one should be the bcc phase and sigma is already dormant but you also need to know the composition of bcc remaining after some sigma has precipitated. You don't know the composition of sigma exactly but it is well known that in the Fe-Cr system it is close to equiatomic. Accepting that composition you know that 0.001 mole of sigma contains $N(\text{Fe})=0.0005=N(\text{Cr})$. Thus, you know the content of bcc, $N(\text{Fe})=0.6995$ and $N(\text{Cr})=0.2995$.

```
POLY_3: s-c N(Fe)=.6995 N(Cr)=.2995
```

```
POLY_3: c-e
```

```
Using global minimization procedure
```

```
Using already calculated grid
```

```
Found the set of lowest grid points in          0 s
```

```
Calculated POLY solution          0 s, total time    0 s
```

```
POLY_3:
```

*) The value of FUNC in this state may be stored in a variable FUbcc.

```
POLY_3: ent-sym var FUbcc=FUNC;
```

```
POLY_3:
```

*) Finally you turn to the sigma phase by entering it and making bcc dormant.

```
POLY_3: ch-st p sigma=ent 0
```

```
POLY_3: ch-st p bcc=dor
```

```
POLY_3: s-c N(Fe)=.0005 N(Cr)=.0005
```

```
POLY_3: c-e
```

```
Using global minimization procedure
```

```
Calculated          137 grid points in          0 s
```

```
Found the set of lowest grid points in          0 s
```

```
Calculated POLY solution          0 s, total time    0 s
```

```
POLY_3:
```

*) The value of FUNC in this state you may store in a variable FUsig.

```
POLY_3: ent-sym var FUsig=FUNC;
```

*) You may like to inspect the results so far. Just print evaluate-functions.

```
POLY_3: eval
```

```
Name(s):
```

```
FUNC=37.04877
```

```
FUINIT=37744.672
```

```
DRFORC=0.015819499
```

```
FUBCC=37707.715
```

```
FUSIG=37.04877
```

```
POLY_3:
```

*) The three values of FUNC are shown. You like to know how much it has changed by the precipitation of the small quantity of sigma. The change will be $\Delta \text{FUNC} = \text{FUbcc} + \text{FUsig} - \text{FUinit}$. It was caused by the change in the amount of sigma, $\Delta \xi = 0.001$ moles of atoms in sigma. The driving force for the precipitation of sigma can thus be approximated by the ratio.

```
POLY_3: ent-sym var Dappr=(FUbcc+FUsig-FUinit)/0.001;
```

*) Now you may compare with the quantity called driving force. Multiply it by RT.

```
POLY_3: ent-sym fun Dtrue=Drforc*8.3145*T;
```

```
POLY_3: sh Dappr Dtrue
```

```
DAPPR=91.992213
```

```
DTRUE=121.40347
```

```
POLY_3: exit
```

```
CPU time          0 seconds
```

Comments

- 1) Of course, POLY has an efficient method of evaluating the driving force for formation of a phase that does not take part in an equilibrium but for some reason it is presented as a dimensionless quantity by dividing it with RT .
- 2) POLY's method will yield the most favourable composition for the new phase, i.e. the composition that has the highest driving force. In the present problem you were asked to derive the value for a sigma phase with a 50/50 composition. If both values, D_{true} and D_{appr} , were trusted, then the conclusion would be that 50/50, which was used in the present calculations, is not the most favourable composition.

1.9. The combined first and second law

Evaluate the quantity $(\partial S / \partial N)_{U,V}$ from a tabulation module that can list H and S for a series of temperatures. Choose Al_2O_3 at 1 atm.

Hint

- 1) The combined law can be written as $dS = (1/T)dU + (P/T)dV - (G_m/T)dN + (D/T)d\xi$ and $(\partial S / \partial N)_{U,V} = -G_m / T$. Thermodynamic tables usually give values under a given P and for a series of T and, as a consequence, they usually list values of H and S . It would have been nice to have a table of U and S but if that is not available one could usually approximate U with H for condensed materials. Use that approximation here.
- 2) Primarily one should expect both S and H to increase if the size, N , is increased because they are both extensive quantities. In order to vary S but keep H constant it seems necessary to allow T to decrease to a level where the tabulated value, which may be given for 1 mole of formula units, has decreased by a factor equal to the one by which N has increased. Denoting the two temperatures by 1 and 2, we get $H=N_1H_1=N_2H_2$ where $N_1=1$. On the other hand, for the entropy we have $\Delta S=N_1S_1-N_2S_2=N_1(S_1-S_2N_2/N_1)=N_1(H_2S_1-H_1S_2)/H_2$. However, $\Delta N=N_2-N_1=N_1(N_2/N_1-1)=N_1(H_1/H_2-1)=N_1(H_1-H_2)/H_2$. We find $\Delta S / \Delta N=(H_2S_1-H_1S_2)/(H_1-H_2)$. From the table one should thus examine the H and S values for two neighbouring temperatures and evaluate $(\partial S / \partial N)_{U,V} \cong \Delta S / \Delta N = (H_2S_1 - H_1S_2) / (H_1 - H_2)$. That should be a reasonably correct value in the middle of the range of T . Make the test at 1000 and 1100 K.
- 3) Sometimes a table also gives the values of G but all values given in a table of properties are molar quantities. G should thus be identical to G_m as given under (1), provided that N is expressed in the same kind of mole, in the present case mole of formula units. You may thus check your result by comparing with G_m/T at the two temperatures.

Instructions for using T-C

Indeed, the tabulation module in T-C also lists G .

Prompts, commands and responses

```

SYS: go da
THERMODYNAMIC DATABASE module running on PC/WINDOWS NT
Current database: TCS Demo Al-Mg-Si Alloys TDB v1

VA DEFINED
TDB_DALMGSI: sw pgeo
Current database: Saxena Pure Minerals Database v1

O VA DEFINED
STEAM OXYGEN HYDROGEN
REJECTED
CARBON_MONOXIDE CARBON_DIOXIDE METHANE
REJECTED
TDB_PGEO:
*) This is a database for mineral oxides and the element O is automatically defined.
TDB_PGEO: def-el Al O
AL DEFINED
TDB_PGEO: 1-sys
ELEMENTS, SPECIES, PHASES OR CONSTITUENTS: /CONSTITUENT/: CONSTITUENT
GAS:G :O2:
> Gaseous Mixture with C-H-O species, using ideal gas model
CORUNDUM :AL2O3:
TDB_PGEO: rej p gas
GAS:G REJECTED
TDB_PGEO: get
REINITIATING GES5 .....
ELEMENTS .....
SPECIES .....
PHASES .....
PARAMETERS ...
Rewind to read functions 1
FUNCTIONS ....
-OK-
TDB_PGEO: go tab
TAB:
*) Print ? and press return to see what options TAB offers. You like to tabulate a substance.
TAB: tab-sub
Substance (phase): CORUND
FRACTION OF CONSTITUENT (RETURN FOR PROMPT):
Pressure /100000/:
Low temperature limit /298.15/: 1000
High temperature limit /2000/: 1500
Step in temperature /100/:
Output file /SCREEN/:

O U T P U T F R O M T H E R M O - C A L C
2007. 4.21 20.49.58

Phase : CORUNDUM Pressure : 100000.00
Specie: *

*****
T Cp H S G
(K) (Joule/K) (Joule) (Joule/K) (Joule)
*****
1000.00 1.23205E+02 -1.59839E+06 1.79265E+02 -1.77765E+06

```

1100.00	1.24863E+02	-1.58598E+06	1.91087E+02	-1.79618E+06
1200.00	1.26312E+02	-1.57342E+06	2.02015E+02	-1.81584E+06
1300.00	1.27609E+02	-1.56072E+06	2.12177E+02	-1.83656E+06
1400.00	1.28793E+02	-1.54790E+06	2.21678E+02	-1.85825E+06
1500.00	1.29892E+02	-1.53497E+06	2.30602E+02	-1.88087E+06

TAB: go pol

POLY version 3.32, Aug 2001

POLY_3:

*) You like to evaluate $(H_2S_1-H_1S_2)/(H_1-H_2)$ and could start with the denominator. You could take numbers from the rows for 1000 and 1100 K in the Table.

POLY_3: ent-sym var deltaH=-1.59839E+06-(-1.58598E+06);

POLY_3: ent-sym var deriv=(-1.58598E+06*1.79265E+02-(-1.59839E+06*1.91087E+02))/deltaH;

POLY_3: ent-sym var G00T1000=-1.77765E+06/1000;

POLY_3: ent-sym var G00T1100=-1.79618E+06/1100;

POLY_3: eval

Name(s):

DELTAH=-12410

DERIV=-1701.9215

G00T1000=-1777.65

G00T1100=-1632.8909

POLY_3: exit

CPU time 0 seconds

Comments

- 1) DERIV is very close to the average of G00 for 1000 and 1100 K. Accepting the approximation of U as H, you have thus confirmed that $(\partial S / \partial N)_{U,V} = -G_m / T$.
- 2) In the problems for Chapter 3 you will learn that this kind of partial derivative can be evaluated directly from POLY.
- 3) In this case you did not need to worry about being consistent with the unit, mole or formula, because all the values came from the tabulation module.

1.10. General conditions of equilibrium

At the melting point of a pure element the liquid and solid phases are in equilibrium with each other and each potential should thus have the same value in both phases. It should be possible to evaluate the melting point by comparing tables of properties of the two phases. Do that for molybdenum.

Hint

The combined law expressed in terms of dU shows that G_m is a potential and is conjugate to N, the amount of matter. There would thus be a driving force for the transfer of matter between the two phases if G_m did not have the same value in both phases. One of them would grow at the expense of the other. You should thus examine at what temperature the two phases have the same G_m value.

Instructions for using T-C

- 1) The tabulation module in T-C lists a function G and for pure elements its value is given per mole of atoms because the formula unit is there one atom. It is then identical to G_m . You should thus make the tabulation module produce tables for both phases in a temperature range and locate the temperature where the G_m values are most similar. Then you can repeat the procedure with a more and more narrow range of T.
- 2) As an alternative, you may apply the tabulation module to a reaction liquid \rightarrow solid and check where G for the reaction is zero.

Prompts, commands and responses

```

SYS: go da
THERMODYNAMIC DATABASE module running on PC/WINDOWS NT
Current database: TCS Demo Al-Mg-Si Alloys TDB v1

VA DEFINED
TDB_DALMGSI: sw PURE
Current database: SGTE Unary (Pure Elements) TDB v4

VA                               /- DEFINED
TDB_PURE4: def-el Mo
MO DEFINED
TDB_PURE4: rej p *
BCC_A2                            FCC_A1                            HCP_A3
LIQUID:L REJECTED
TDB_PURE4: rest p bcc liq
BCC_A2                            LIQUID:L RESTORED
TDB_PURE4: get
REINITIATING GES5 .....
ELEMENTS .....
SPECIES .....
PHASES .....
PARAMETERS ...
Rewind to read functions           2
FUNCTIONS ....

List of references for assessed data

'PURE4 - SGTE Pure Elements (Unary) Database (Version 4.4), developed by
SGTE (Scientific Group Thermodata Europe), 1991-2003, and provided by
TCSAB (Aug. 2003). '

The list of references can be obtained in the Gibbs Energy System also
by the command LIST_DATA and option R

-OK-
TDB_PURE4: go tab
TAB: tab-sub bcc
SPECIFY SUBLATTICE (0 FOR ALL) /0/:
Pressure /100000/:
Low temperature limit /298.15/: 2500
High temperature limit /2000/: 3500
Step in temperature /100/:
Output file /SCREEN/:

O U T P U T   F R O M   T H E R M O - C A L C
2007. 4.22                                     14. 5.24

```



```

DALMGSI = TCS Demo Al-Mg-Si Alloys TDB v1
DFECRC  = TCS Demo Fe-Cr-C Alloys TDB v1
PURE4   = SGTE Unary (Pure Elements) TDB v4
PSUB    = TCS Public Pure Substances TDB v1
PBIN    = TCS Public Binary Alloys TDB v1
PKP     = Kaufman Binary Alloys TDB v1
PCHAT   = Chatenay-Malabry Binary Alloys TDB v1
PTERN   = TCS Public Ternary Alloys TDB v1
PG35    = G35 Binary Semi-Conductors TDB v1
PION    = TCS Public Ionic Solutions TDB v2
PAQ2    = TCS Public Aqueous Solution TDB v2
PGEO    = Saxena Pure Minerals Database v1
PFRIB   = Fridberg Dilute Fe-Alloys MDB v1
USER    = User defined Database

```

DATABASE NAME /PURE4/:

*) With this option you are again asked what database to use. You could thus have gone directly here and never gone to the database module. Now you accept the defaults by pressing return.

DATABASE NAME /PURE4/:

```

VA                /-  DEFINED
REINITIATING GES5 .....
MO  DEFINED
ELEMENTS .....
SPECIES .....
PHASES .....
PARAMETERS ...
Rewind to read functions          2
FUNCTIONS ....

```

List of references for assessed data

```

'PURE4 - SGTE Pure Elements (Unary) Database (Version 4.4), developed by
  SGTE (Scientific Group Thermodata Europe), 1991-2003, and provided by
  TCSAB (Aug. 2003). '

```

The list of references can be obtained in the Gibbs Energy System also by the command LIST_DATA and option R

-OK-

```

Pressure /100000/:
Low temperature limit /2500/:  2500
High temperature limit /3500/:  3500
Step in temperature /100/:

```

```

Output file /SCREEN/:  file
Graphical output? /Y/:
Plot column? /2/:  5

```

```

O U T P U T   F R O M   T H E R M O - C A L C
2007. 4.22                                14. 5.25

```

```

Reaction:  MO<C>=MO<L>
MO<BCC_A2>
MO<LIQUID>

```

```

T          Delta-Cp          Delta-H          Delta-S          Delta-G

```

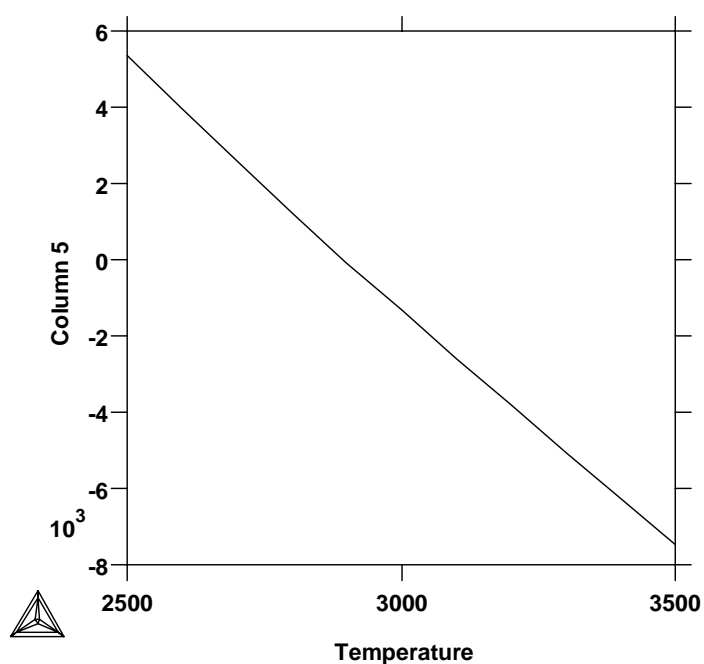
(K)	(Joule/K)	(Joule)	(Joule/K)	(Joule)
2500.00	-4.35298E+00	4.02767E+04	1.39694E+01	5.35317E+03
2600.00	-5.50791E+00	3.97856E+04	1.37769E+01	3.96554E+03
2700.00	-6.90763E+00	3.91670E+04	1.35436E+01	2.59915E+03
2800.00	-8.59201E+00	3.83945E+04	1.32629E+01	1.25839E+03
2900.00	-1.03739E+01	3.74381E+04	1.29274E+01	-5.14896E+01
3000.00	-7.39112E+00	3.65591E+04	1.26292E+01	-1.32839E+03
3100.00	-5.32483E+00	3.59295E+04	1.24225E+01	-2.58036E+03
3200.00	-3.87635E+00	3.54736E+04	1.22777E+01	-3.81495E+03
3300.00	-2.84960E+00	3.51402E+04	1.21750E+01	-5.03729E+03
3400.00	-2.11414E+00	3.48940E+04	1.21015E+01	-6.25092E+03
3500.00	-1.58213E+00	3.47106E+04	1.20483E+01	-7.45826E+03

POSTPROCESSOR VERSION 3.2 , last update 2002-12-01

POST: **exit**

TAB: CPU time 0 seconds

THERMO-CALC (2007.04.22:14.04) : REACTION TABULATION



Comments

- 1) With this option you get a diagram. From the last Table you can confirm that the melting point is very close to 2900 K and the diagram shows the same result.
- 2) You could have obtained a more accurate value by interpolation or by simply running the tabulation from 2890 to 2900 K.
- 3) Here you have learned how to list the properties of a reaction.