THERMOCALC Scripting

back

Scripting allows simple customisation of the running of THERMOCALC. This is a good idea because each time you run the program you would otherwise be met with a barrage of questions many of which you might know in advance are not relevant or that can be set ahead of time. It is usually possible to run THERMOCALC without using scripts, but using them can make your life much easier. In addition some facilities are only available via scripts.

There are two places that scripts are read by THERMOCALC :

1. in a separate file called TH PREFS (Win: THPREFS.TXT): these scripts control global aspects of running THERMOCALC—things that will affect all runs of THERMOCALC with different datafiles,

2. in the second section of a datafile (see datafile layout): these local scripts control the way THERMOCALC runs that particular datafile.

Each of the controls which can be set by scripts are set to a default in THERMOCALC. All that happens with the provision of scripts in TH PREFS or in a datafile is that the default is overridden. Each time it is run, THERMOCALC reads these scripts, and these override the defaults; THERMOCALC does not store or remember previously read scripts.

The basic script format involves a single line of file, starting with a “keyword”. If you don’t get the keyword right (ie if THERMOCALC doesn’t recognise it), then the line is ignored and (generally) there will be no indication that this has happened (but THERMOCALC won’t bomb).

The simplest form of script involves a “keyword” followed by one of “yes”, “no” or “ask”. “Yes” corresponds to some action which will definitely be undertaken, and “no” corresponds to the action definitely not being undertaken. With “ask”, you will be prompted by THERMOCALC, while the program is running, to determine if the action should be undertaken. In a few cases “no” gets substituted for “ask”. If one of “yes”, “no” or “ask” is omitted, then “yes” is assumed.

More complex scripts allow additional information to be provided to THERMOCALC. Remember that if THERMOCALC doesn’t understand what you have tried to do with a script, there is no problem, but that the intended action or inaction will not occur. This will normally be obvious in the way THERMOCALC runs! As each script is a single line of datafile, one script which doesn’t work will not affect succeeding scripts. (In the following, the forms of scripts are in quotes—don’t include the quotes when you use the scripts)

Scripting in the preferences file

The most useful scripts which can be used in TH PREFS are

- calcmode - specifying a calculation mode for all runs
- pdatasuffix - specifying a suffix on TH PDATA1 and TH PDATA2 (Win: TH.PD)
- datasuffix - specifying a datafile suffix
- filecreator - specifying which editor will be used

Other scripts are:

- makelog - whether to make a log file
- logfilename - overriding the log filename root, TH LOG (Win THLOG.TXT)
- datafilename - overriding the data filename root, TH D (Win THD.TXT)
- outfilename - overriding the output filename root, TH O (Win THO.TXT)
- datadialog - whether to always use a standard dialog box for datafile choice, rather than prompting for a filename (Mac)
- autooverwrite - whether automatic overwriting of output files is to occur
- nooutsuffix - whether no suffix on output files will be used
- dontwrap - whether line wrap occurs in screen output
- autoexit - whether to automatically exit THERMOCALC at the end of a run
- dontaskadjust - ask user to adjust window size at run start

In the following, the default given is what THERMOCALC will do if you don't use a script. Of course, there is no problem if you just repeat the default in a script. (Few of the scripts here result in prompts if you use "ask").

calcmode If you are only doing one type of calculation, say only average *PT* calculations, then you can specify this with this script (and save yourself always replying in this way to a prompt). Form:

- "calcmode -1" → calculation mode prompted for (this is the equivalent of the script not occuring in TH PREFS)
- "calcmode 0" → data table mode (this would not be a usual setting
- "calcmode 1" → phase diagram calculations involving phases which are (solid) solutions
- "calcmode 2" → rock calculations (ie average *PT* calculations)
- "calcmode 3" → calculations involving all the reactions between a set of end-members (for phase diagrams not involving (solid) solutions and debugging average *PT* calculations)

Default: calculation mode prompted for

pdatasuffix The pdata files contain the thermodynamic data which THERMOCALC uses for all its calculations. As the pdata files are updated, the results of calculations change (if usually only slightly). Sometimes it is useful to keep older pdata files, so that original calculations can be reproduced. This script allows several generations of pdata files to be kept together, using different suffices on the pdata filenames, and the choice of which ones are used being controlled by a TH PREFS script. Forms:

- "pdatasuffix blah" → pdata files will be expected to have names TH PDATA1BLAH and TH PDATA2BLAH

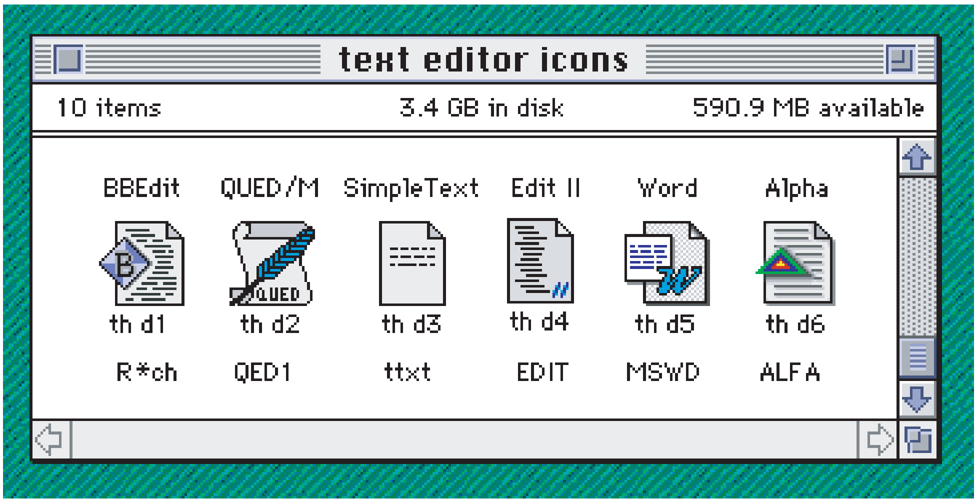
Default: files are named TH PDATA1 and TH PDATA2

datasuffix This script is available if you are spending a lot of time working with just one datafile; it allows you to specify this file in TH PREFS, via the suffix to the root of the datafile name. Having done this, it means that you can only work with this file (until you change the script, or remove it). This root is either the default, 'th d', or whatever is specified in the script datafilename. If "datasuffix" is not present in TH PREFS, then the suffix to 'th d' (or whatever is specified in the script datafilename) is prompted for when you run THERMOCALC. Form:

- "datasuffix blah" → datafile used is TH DBLAH (Win: THBLAH.TXT)

Default: datafile name suffix will be prompted for

filecreator filecreator allows you to specify the text editor application you want to use for data, output and logfiles. You can identify the creator - a 4 character "signature" of a text file, which associates it with a particular application, by using Resorceror or ResEdit, (eg, the creator for the text editor BBEdit is R*ch). Remember that case *does* matter in creators. The benefit of specifying the creator using this script is that output and log files created by THERMOCALC will have this creator, and therefore can be double-clicked to open in that editor. (Of course this assumes that you have got the editor on your machine!) Having specified the creator, the file should appear with the appropriate icon - again only if you have that editor. Below, the name of the editor is given above the example datafile, and the required creator below it:



text editor icons

10 items 3.4 GB in disk 590.9 MB available

BBEdit	QUED/M	SimpleText	Edit II	Word	Alpha
th d1	th d2	th d3	th d4	th d5	th d6
R*ch	QED1	ttxt	EDIT	MSWD	ALFA

In addition, having specified a particular editor, the creator (and therefore the icon) of datafiles will be set to the specified one (even if they were created in, and had the icon of, another editor). Forms:

- "filecreator creator", eg "filecreator QED1"

Default: "filecreator R*ch" (ie BBEdit)

makelog This script controls whether a log file is created. The log file differs from the usual output file in including the prompts put up on the screen, as well as your responses to them, as well as additional output in some situations. Even if you always overwrite the log file with each new run, it is there if you need it if something goes wrong in the current one. You are then able to check if you entered something unintended. With complicated work, it may be worthwhile saving the log files so that you can reproduce the results later. Forms:

- "makelog" 'or "makelog yes" → make a logfile
- "makelog no" → do not make a logfile ("makelog ask" is taken to mean "makelog no")

Default: "makelog yes"

logfilename As with out files (see last), THERMOCALC log files are expected to have the same root, in other words, to start with the same word or words. The standard for log files is the root, TH LOG. (Then there is an optional additional suffix, as outlined in the section output files). To give some flexibility, you can choose your own root for log file names using this script. Forms:

- "logfilename word1", eg "logfilename log"
- "logfilename word1 word2", eg "logfilename thermo log"

Default: "logfilename th log" (Win: logfilename thlog.txt)

datafilename To ease the recognition and organisation of your datafiles, all THERMOCALC datafiles are expected to have the same root, in other words, to start with the same word or words. The standard is the root, 'th d', so that different datafiles might be TH DST, THDRP13, and so on. To give some flexibility, you can choose your own root for datafile names using this script. Forms:

- "datafilename word1", eg "datafilename thdata"
- "datafilename word1 word2", eg "datafilename thermo d"

Default: "datafilename th d" (Win: "datafilename thd.txt")

outfilename In the same way as with datafiles (see last card), all THERMOCALC output files are expected to have the same root, in other words, to start with the same word or words. The standard is the root, TH O. (Then the suffix of the datafile name is added, before an optional additional suffix, as outlined in the section output files). To give some flexibility, you can choose your own root for output file names using this script. Forms:

- "outfilename word1", eg "outfilename thout"
- "outfilename word1 word2", eg "outfilename thermo out"

Default: "outfilename th o" (Win: "outfilename tho.txt")

datadialog This script is available if you would prefer to be prompted for your choice of datafile by the Standard File dialog box (Mac only). Forms:

- "datadialog yes" or "datadialog" → datafile choice is via a dialog box
- "datadialog no" → datafile name is entered at a THERMOCALC prompt

Default: "datadialog no"

nooutsuffix As outlined in the script section for outfilename, output files may have a suffix which is prompted for when you run THERMOCALC . For example, with the default roots, a filename might be TH ORP13Q, in which 'th o' is the root,'RP13' is the datafile suffix (in TH DRP13), and 'q' is the optional suffix. The same applies in Win with THRP13Q.OUT, with the root being 'th.out'. If you never want to make use of the suffix, and thus avoid the appropriate prompt, use this script. Forms:

- "nooutsuffix" or "nooutsuffix yes" → no output suffix will be used
- "nooutsuffix no" → you will be prompted for the optional output suffix (There is no prompt with "nooutsuffix ask": "nooutsuffix no" is assumed)

Default: "nooutsuffix yes".

autooverwrite When THERMOCALC is running, if the log or output filename already exist, then you are normally prompted to determine if you want the file overwritten or not. If it is overwritten, the previous contents of the file are thrown away, and the file is used for the current run. (Otherwise, if you do not want overwriting, THERMOCALC adds a letter suffix to the file and uses that, preserving the orginal file. Thus, if TH oRP13 already exists, and you do not want to overwrite it, the output will be put in TH oRP13A, if that doesn't already exist. If it does, suffices, 'b', 'c', etc are tried til an OK one is found.) If you want automatic overwriting of log and output files, then autooverwrite allows this. See the section output files for details of output file naming. Forms:

- "autooverwrite" or "autooverwrite yes" → automatic overwriting will occur
- "autooverwrite no" → you will be prompted concerning overwriting (There is no prompt with "autooverwrite ask": "autooverwrite no" is assumed)

Default: "autooverwrite yes".

dontwrap Screen output on smaller screens (or longer lines of output) can look messy if a line is wrapped onto the following one. This script allows user control over this. With "outdialog yes", a # is placed at the end of a line which would otherwise wrap onto the following line. The full line of course does appear in the output file. Use in conjunction with "dontaskadjust", to control the width of window on which the wrapping is not done. Forms:

- "dontwrap yes" or "dontwrap" → don't wrap output on screen
- "dontwrap no" → allow wrap of output on screen

Default: "dontwrap yes".

dontaskadjust THERMOCALC may not make an appropriate width window for your screen (Mac only). You can adjust the screen width at any time that THERMOCALC is running, but if you are using "dontwrap yes", the # will appear at the original window width. In this case, "dontaskadjust yes" causes a prompt for you to adjust the window width then, and the wrapping will occur to that width in the run. Forms:

- "dontaskadjust yes" or "dontaskadjust" → prompt for window adjustment
- "dontaskadjust no" → don't prompt for window adjustment

Default: "dontaskadjust yes".

autoexit THERMOCALC can exit automatically when all calculations have finished (ie when ther is the "all done?" prompt) with this script (Mac only). Otherwise you must use command-q in the usual way. Forms:

- "autoexit yes" or "autoexit" → automatically exit after all done
- "autoexit no" → manually exit after all done

Default: "autoexit yes".

top **back**

Scripting in mode 1 datafiles

Scripts for phase diagram calculations that may appear in datafiles.
Related to calculation PT ‘window’

- setdefPwindow - defining the default P window for calculations
- setdefTwindow - defining the default T window for calculations
- setPwindow - defining the P range for calculations
- setTwindow - defining the T range for calculations
- calctatp - whether to calculate T at given P for PT-lines
- moreprec - specifying more output precision
- calcsdnle - whether to perform uncertainty propagation on calculations

Related to phases involved in calculations

- omit *or* ignore - specifying phases to ignore this run
- fluidpresent - whether H_2O or CO_2 in the datafile means fluid present
- fluidexcess - whether fluid is “in excess”
- setexcess - defining other “in excess” phases
- exbuff - allows external buffering calculations involving H_20-CO_2 fluids
- seta - whether to set activities in calculations
- setmu - whether to set chemical potentials in calculations

Related to calculating pseudosections

- pseudosection - whether to generate pseudosection information
- setbulk - defining bulk composition(s) for pseudosections
- setmodeiso - allows the setting of mode isopleths
- zeromodeiso - allows the setting of zero mode isopleths
- setiso - whether to calculate composition isopleths in calculations
- calcg - Gibbs energy of equilibrium calculated
- dogmin - allows Gibbs energy minimisation

Output for DRAWPD

- drawpd - calculated coordinates into 'other' file (default TH DR)

Related to calculating projections

- project - whether to project phases onto a compatibility diagram
- setproject - defining the compatibility diagram (projection plane)
- projcomp - whether to project bulk comps onto the compatibility diagram

Output for Mathematica (eventually to be superceded by DRAWPD)

- smath - calculated PT coordinates into 'other' file
- xinsmath - calculated PT and x coordinates into 'other' file
- tabsmath - calculated PT coordinates into 'other' file in tab-separated form
- cmath - compatibility diagram coordinates into 'other' file
- txmath - Tx/Px pseudosection coordinates into 'other' file

Optional output in the log file

- incthermo - whether to include tables of thermo data in the logfile
- incax - whether to include the a-x relationships in the logfile
- incnler - whether to include the set of reactions in the logfile

setdefPwindow This script sets a default P window to be used if the user hits return (ie inputs nothing) at a prompt for a P (or *PT*) range. The 2 numbers "defplow" and "defphigh" defining the window must be given. Form:

- "setdefPwindow defplow defphigh" or "setdefPwindow yes defplow defphigh" → the given numbers define the default P window;
- "setdefPwindow ask defplow defphigh" → you are prompted to determine if the specified window is to be used;
- "setdefPwindow no defplow defphigh" → the specified window is ignored

Default: 0.1 to 20 kbar

setdefTwindow This script sets a default T window to be used if the user hits return (ie inputs nothing) at a prompt for a T (or *PT*) range. The 2 numbers "deftlow" and "defthigh" defining the window must be given. Form:

- "setdefTwindow deftlow defthigh" or "setdefTwindow yes deftlow defthigh" → the given numbers define the default T window;
- "setdefTwindow ask deftlow defthigh" → you are prompted to determine if the specified window is to be used;
- "setdefTwindow no deftlow defthigh" → the specified window is ignored

Default: 200° to 1000°

setPwindow This script allows the P window within which equilibria are considered to be specified in the datafile. The 2 numbers "plow" and "phigh" defining the window must be given. With these "phigh" $\geq$ "plow", with, for example, equality not being allowed if the calculated equilibria are invariant. If equality is specified, and is not allowed, THERMOCALC prompts for a proper range. Form:

- "setPwindow plow phigh" or "setwindow yes plow phigh" → the given numbers define the P window;
- "setPwindow ask plow phigh" → you are prompted to determine if the specified window is to be used;
- "setPwindow no plow phigh" → the specified window is ignored

Default: prompt for P window.

setTwindow This script allows the T window within which equilibria are considered to be specified in the datafile. The 2 numbers "tlow" and "thigh" defining the window must be given. With these "thigh" $\geq$ "tlow", with, for example, equality not being allowed if the calculated equilibria are invariant. If equality is specified, and is not allowed, THERMOCALC prompts for a proper range. Form:

- "setTwindow tlow thigh" or "setwindow yes tlow thigh" → the given numbers define the T window;
- "setTwindow ask tlow thigh" → you are prompted to determine if the specified window is to be used;

- "setTwindow no tlow thigh" → the specified window is ignored

Default: prompt for T window.

calctatp With a PT window (see setwindow), and considering univariant equilibria (ie reaction lines) in a PT window, there is a choice of whether T should be calculated at a series of P, or P calculated at a series of T. For example, for reactions with a very small $\mathrm{d}P/\mathrm{d}T$, it is easier to calculate P at T. This script allows this choice to be made. (If for the window plow=phigh or tlow=thigh, then the choice does not arise). Form:

- "calctatp" or "calctatp yes" → always calculate T at P;
- "calctatp ask" → you are prompted for the choice;
- "calctatp no" → always calculate P at T

Default: you are prompted for the choice

omit or **ignore** Allows phases for which information is provided in the first part of a datafile to be ignored in using the datafile. This circumvents the need for lengthy lists of phases to be omitted at the 'which phases' prompt when running THERMOCALC or alternatively the cutting and pasting of phase info from the first part to the last (storage) part of datafiles. Form:

- "ignore phase1 phase2 etc" or "ignore yes phase1 phase2 etc" → phase1, phase2, etc, which are present in the first part of the datafile, are to be ignored in the calculations
- "ignore ask phase1 phase2 etc" → you are prompted whether these phases are to be ignored;
- "ignore no anything" → no phases ignored

Default: nothing ignored

fluidpresent In phase diagram calculations involving H_2O or CO_2 (but not H_2O and CO_2, see below), these end-members might be included in a system to be used in fluid-absent equilibria (say with a(H_2O) set at particular values). This script is used to communicate that a system may have H_2O or CO_2 present as phases. If a fluid phase with H_2O and CO_2 is in the system, it is coded in the datafile as a solution, much as solid solutions are, and mode 1 should be used. In this case fluidpresent is automatically yes, and does not need to be scripted. Form:

- "fluidpresent" or "fluidpresent yes" → H_2O or CO_2 is present as a phase;
- "fluidpresent ask" → prompt - this is THERMOCALC normal behaviour in the absence of the script;
- "fluidpresent no" → H_2O or CO_2 cannot be present as a phase

Default: "fluidpresent ask"

fluidexcess In calculations, it is often convenient to consider one or more phases to be "in excess", to reduce the total number of equilibria to be considered, and also to reduce the effective dimension of the system so that compatibility diagrams can be drawn (eg the 6-component system KFMASH is reduced to the effective ternary, AFM, by having muscovite, quartz and H_2O in excess). The fluidexcess script allows fluid in excess to be specified. Form:

- “fluidexcess” or “fluidexcess yes“ → fluid is in excess;
- “fluidexcess ask” → prompt - this is THERMOCALC normal behaviour in the absence of the script;
- “fluidexcess no” → fluid not in excess

Default: “fluidexcess ask”

setexcess In calculations, it is often convenient to consider one or more phases to be “in excess“, to reduce the total number of equilibria to be considered, and also to reduce the effective dimension of the system so that compatibility diagrams can be drawn (eg the 6-component system KFMASH is reduced to the effective ternary, AFM, by having muscovite, quartz and H_2O in excess). The setexcess script allows phases other than fluid to be considered to be in excess (see fluidexcess). Form:

- “setexcess ph1 ph2 ph3 ...” or “setexcess yes ph1 ph2 ph3 ...” → the phases ph1, ph2, and so on are taken to be in excess (above eg “setexcess mu q“);
- “setexcess ask ph1 ph2 ph3 ...” → prompt to determine if the phases ph1, ph2, and so on are taken to be in excess;
- “setexcess no ph1 ph2 ph3 ...” → nothing in excess

Default: prompt for in excess phases.

exbuff Allows external buffering calculations involving H_20-CO_2 fluids to be done. For example in calculating a T-$x(CO_2)$, incombination with “setiso xCO2”, it allows pseudosections to be drawn. Form:

- “exbuff” or “exbuff yes” → external buffering is activated;
- “exbuff no” → not external buffering.

Default: “calcsdnle no”

seta In phase diagram calculations, it is sometimes important to see what the consequence is of varying the activity of an end-member, for example H_2O (under granulite facies conditions). This script allows this to be done. Form:

- “seta” or “seta yes” → you will be prompted for the name of the end-member whose activity will be fixed;
- “seta em” or “seta yes em” → the activity of end-member, em, will be fixed (eg “seta H_2O“);
- “seta ask” → you will be prompted to determine if you wish to fix an activity;
- “seta ask em” → you will be prompted to determine if you wish to fix the activity of end-member, em (eg “seta ask H_2O“);
- “seta no anything” → you do not wish to fix an activity.

Default: “seta no”

setmu In phase diagram calculations, it is sometimes important to see what the consequence is of varying the chemical potential of a component, for example K2O (in orthoamphibole assemblages). This script allows this to be done. Form:

- “setmu” or “setmu yes” → you will be prompted for the name of a component whose chemical potential will be fixed;
- “setmu comp” or “setmu yes comp“ → the chemical potential of component, comp, will be fixed (eg “setmu Na2O”);
- “setmu ask” → you will be prompted to determine if you wish to fix a chemical potential;
- “setmu ask comp” → you will be prompted to determine if you wish to fix the chemical potential of comp (eg “seta ask Na2O”);
- “setmu no anything” → you do not wish to fix a chemical potential.

Default: “setmu no”

calcsdnle Allows uncertainty propagation calculations to be switched on or off. Form:

- “calcsdnle” or “calcsdnle yes” → uncertainty propagation calculations will be performed;
- “calcsdnle ask” → you are prompted whether uncertainty propagation calculations will be performed;
- “calcsdnle no” → no uncertainty propagation

Default: “calcsdnle no”

moreprec Allows an extra decimal place in the output of *PT* and x values. This is useful if equilibria are closely spaced - note, however, that there is no implication that the numbers are somehow more accurate when this switch is on. Form:

- “moreprec” or “moreprec yes” → an extra decimal place is used in the output;
- “moreprec ask” → you are prompted whether an extra decimal place should be used in the output;
- “moreprec no” → no extra decimal place in the output.

Default: “moreprec no”

pseudosection In phase diagram calculations, it is very informative to know if a particular equilibrium is “seen“ by a particular bulk composition. Phase diagrams, for example *PT* diagrams, drawn to show just those equilibria “seen” by a particular bulk composition are referred to as pseudosections. This script allows pseudosection information to be collected. Form:

- “pseudosection” or “pseudosection yes“ → you are prompted for the bulk composition(s) you want to collect pseudosection information for;
- “pseudosection ask” → you are prompted concerning whether you want to collect pseudosection information; if you reply yes, you are prompted for the bulk composition(s) you want to collect pseudosection information for; if you reply no, pseudosection information is not collected;

- “pseudo no” → pseudosection information is not collected.

Default: “pseudosection ask”

setbulk If pseudo is yes, then setbulk allows one bulk composition to be scripted. The simplest way to get this script correct is to run THERMOCALC with the datafile of interest, with “pseudo yes” and without the setbulk script. As the specification of the bulk composition relates directly to the chosen “in excess” phases, they can be scripted for (see also setproject). Forms:

- “setbulk yes 31 20 41 8” or “setbulk 31 20 41 8” → use this bulk composition
- “setbulk ask 31 20 41 8” → prompt to ask the user whether this bulk composition is to be used
- “setbulk no 31 20 41 8” → this bulk composition is not to be used, and the user will be prompted for the bulk composition (useful for temporarily “switching off” the script)

setmodeiso In pseudosection calculations, a useful facility is to be able to fix the value of modes of phases, and calculate all the other composition and mode variables of the phases in an equilibrium. This facility allows the position of mode isopleths on a *PT* pseudosection to be calculated. Regardless of the variance of the field in a *PT* pseudosection, with one mode fixed, the equilibrium is effectively univariant (ie defines a *PT* line), and with two modes fixed, the equilibrium is effectively invariant (ie defines a *PT* point). Form:

- “setmodeiso” or “setmodeiso yes“ → you will be prompted for the names of the composition variables to be fixed;
- “setmodeiso ph” or “setmodeiso yes ph“ → the mode of the phase, ph, is to be fixed, and the values are prompted for (eg “setmodeiso bi“);
- “setmodeiso ph isolow isohigh isoinc” or “setmodeiso yes ph isolow isohigh isoinc” → the mode of the phase, ph, is to have values running from isolow through to isohigh with an interval, isoinc (eg “setiso bi 0.1 0.6 0.1“);
- “setmodeiso ask” → you will be prompted to determine if you wish to fix any composition variables, then prompted for the details;
- “setmodeiso ask ph” → you will be prompted to determine if you wish to fix the mode of the phase, ph (eg “setmodeiso ask bi“);
- “setmodeiso ask ph isolow isohigh isoinc” → you will be prompted to determine if you wish to fix the composition variable, ph, and to have values running from isolow through to isohigh with an interval, isoinc;
- “setmodeiso no anything” → you do not wish to fix any modes.

Default: “setmodeiso ask”

zeromodeiso This qualifies “setmodeiso yes“, making the setting of modes always involve zero values. This facility is central to calculating the lines and points that define *PT* and Tx pseudosections. Form:

- “zeromodeiso” or “zeromodeiso yes“ → allows zero mode isopleths to be set;
- “zeromodeiso ask” → you are prompted whether you want zero mode isopleths to be set;

- "zeromodeiso no" → zero mode isopleths are not set

Default: "zeromodeiso ask"

setiso In phase diagram calculations, an important facility is to be able to fix the value of any compositional variable, and calculate all the other composition variables of the phases in an equilibrium. This facility allows the position of isopleths on a *PT* diagram to be calculated, as well as the edges of trivariant fields on compatibility diagrams. For example, in AFM (in KFMASH), the compositions of, say, coexisting biotite and chlorite can be calculated via fixing x(Fe) in the biotite (or chlorite). A major application concerns T-x(CO_2) (and P-x(CO_2)) diagrams: by fixing x(CO_2) in the fluid and calculating equilibria such diagrams can be calculated. Care is needed in the specification of variance for such calculations. Remember that univariant lines on T-x(CO_2) diagrams are actually divariant on a *PT* diagram. Thus with x(CO_2) to be specified, the required variance when prompted to calculate univariant lines on a T-x(CO_2) is trivariant. In general, for each composition variable fixed, the (effective) variance is decreased by one. Form:

- "setiso" or "setiso yes" → you will be prompted for the names of the composition variables to be fixed;
- "setiso var" or "setiso yes var" → the composition variable, var, is to be fixed, and the values are prompted for (eg "setiso x(bi)");
- "setiso var isolow isohigh isoinc" or "setiso yes var isolow isohigh isoinc" → the composition variable, var, is to have values running from isolow through to isohigh with an interval, isoinc (eg "setiso x(bi) 0.1 0.6 0.1");
- "setiso ask" → you will be prompted to determine if you wish to fix any composition variables;
- "setiso ask var" → you will be prompted to determine if you wish to fix the compositional variable, var (eg "setiso ask x(bi)");
- "setiso ask var isolow isohigh isoinc" → you will be prompted to determine if you wish to fix the composition variable, var, to have values running from isolow through to isohigh with an interval, isoinc;
- "setiso no anything" → you do not wish to fix any composition variables.

Default: "setiso ask"

calcg The Gibbs energy of each equilibrium is calculated. Form:

- "calcg" or "calcg yes" → calculate Gibbs energy;
- "calcg ask" → prompt for calculation of Gibbs energy;
- "calcg no" → do not calculate Gibbs energy

Default: "calcg no"

dogmin Allows Gibbs energy minimisation calculations: for a specified bulk composition all of the equilibria from divariant up to a specified variance are calculated and the one, at each *PT*, that has the lowest Gibbs energy is the stable one. This can be useful for kick-starting the calculation of a pseudosection. Form:

- “dogmin” or “dogmin yes” → do Gibbs energy minimisation;
- “dogmin ask” → prompt for Gibbs energy minimisation;
- “dogmin no” → Gibbs energy minimisation not done

Default: “dogmin no“

drawpd If “makeo yes” is in the preferences file, indicating that a ‘other’ file is to be made (whose default name root is ‘th dr’ - Win: ‘thdr.txt’), this causes coordinate information from the calculations to be put in it in a format that is readable by the DRAWPD software. Form:

- “drawpd” or “drawpd yes” → put coordinate info in ‘other’ file;
- “drawpd no” → nothing put

Default: “smath no“

project In representing the results of phase diagram calculations, it is often convenient to be able to project the calculated compositions of the phases into a compatibility diagram. This script allows this. Form:

- “project” or “project yes” → you are prompted for the definition of the compatibility diagram you want to use;
- “project ask” → you are prompted concerning whether you want to project the phase diagram information; if you reply yes, you are prompted for the definition of the compatibility diagram you want to use; if you reply no then projection is not performed;
- “project no” → projection is not performed.

Default: “project ask”

setproject If project is yes, then setproject allows the definition of the compatibility diagram (or projection plane) to be scripted, at least for compatibility diagrams which have oxides as their corners. So for example, for KFMASH projected from muscovite, quartz and H_2O (ie having muscovite, quartz and H_2O in excess), Al2O3, FeO and MgO (ie AFM) define the corners of the usually used compatibility diagram. The oxides are specified in terms of their position in the list of oxides given by THERMOCALC when this same information is prompted for. The simplest way to get this script correct is to run THERMOCALC with the datafile of interest, with “project yes” and without the setproject script. Forms:

- “setproject yes 1 3 2” or “setproject 1 3 2” → use 1st, 3rd and 2nd component
- “setproject ask 1 3 2” → prompt to ask the user whether this is to be used
- “setproject no 1 3 2” → this is not to be used, and the user will be prompted for the projection plane (useful for temporarily “switching off” the script)

projcomp In phase diagram calculations involving projection and the collection of pseudosection information, particularly when there is projection from variable composition phases, it can be useful to see where the bulk composition(s) plot on the compatibility diagram (projection plane). This script allows this, as long as project is yes and pseudo is yes. A bulk composition is only projected if the calculated equilibrium is “seen” by the bulk composition. Form:

- "projcomp" or "projcomp yes" → do project bulk composition(s) onto the compatibility diagram;
- "projcomp ask" → you are prompted whether you want bulk composition(s) projected onto the compatibility diagram;
- "projcomp no" → bulk composition(s) are not projected

Default: "projcomp ask"

smath If "makeo yes" is in the preferences file, indicating that a 'other' file is to be made (whose default name root is 'th dr' - Win: 'thdr.txt'), this causes coordinate information from the calculations to be put in it. The default form of the coordinate information is Mathematica-readable (ie info being comma-separated, structured with curly brackets). This is the standard information used with the Mathematica "drawline" code for generating various diagrams. This can be varied, or changed via, "xinsmath yes" or "tabsmath yes". Form:

- "smath" or "smath yes" → put coordinate info in "other" file;
- "smath no" → nothing put

Default: "smath no"

xinsmath This qualifies "smath yes". Generally the coordinates output are just the PT. For some calculations, eg for T-$x(\mathrm{CO_2})$ diagrams, composition info is also needed. This script causes it to be output in addition to the PT info. Form:

- "xinsmath" or "xinsmath yes" → put extra coordinate info in "other" file;
- "xinsmath no" → nothing put

Default: "xinsmath no"

tabsmath This qualifies "smath yes". Rather than the coordinate information being in Mathematica-readable format, it is output in simple tab-separated form (as understood by various spreadsheet programs). Form:

- "tabsmath" or "tabsmath yes" → do tab-separated output;
- "tabsmath ask" → ask whether to produce tab-separated output;
- "tabsmath no" → output not tab-separated

Default: "tabsmath no"

cmath If "makeo yes" is in the preferences file, indicating that a 'other' file is to be made (whose default name root is 'th dr' - Win: 'thdr.txt'), this causes compatibility diagram coordinate information from the calculations to be put in it. The form of the coordinate information is Mathematica-readable (ie info being comma-separated, structured with curly brackets), and is intended for use with the Mathematica functions for drawing compatibility diagrams. Form:

- "cmath" or "cmath yes" → put coordinate info in "other" file;

- “cmath no” → nothing put

Default: “cmath no“

txmath If “makeo yes” is in the preferences file, indicating that a ‘other’ file is to be made (whose default name root is ‘th dr’ - Win: ‘thdr.txt’), this causes Tx/Px pseudosection coordinate information from the calculations to be put in it. The form of the coordinate information is Mathematica-readable (ie info being comma-separated, structured with curly brackets), and is intended for use with the Mathematica functions for drawing these pseudosections. Form:

- “txmath” or “txmath yes” → put coordinate info in “other” file;
- “txmath no” → nothing put

Default: “txmath no“

incthermo In phase diagram calculations, there is the option of a table of the thermodynamic data being included in the logfile. This script allows it to happen. Usually there is not much point in including the table, given that it takes a little time to print. One situation it might be useful, though, is when the DQF facility or the dependent end-member facility is used, in which case the data for the constructed end-member may be of interest. Form:

- “incthermo” or “incthermo yes” → include the table;
- “incthermo no” → table not included

Default: table not included

incax In phase diagram calculations, there is the option of a summary of the activity-composition relationships in the datafile being included in the logfile. This script controls this. Given that coding the activity-composition relationships provides lots of opportunities for making mistakes, it is a good idea to include this information at least until it has been thoroughly checked. Form:

- “incax” or “incax yes” → include the a-x info;
- “incax no” → a-x info not included

Default: info included

incnler In phase diagram calculations, there is the option of the set of reactions between the end-members of the minerals used in setting up the set of non-linear equations (equilibrium relationships) being included in the logfile. This script allows it to happen. Usually there is not much point in including this information. Form:

- “incnler” or “incnler yes” → include the reactions;
- “incnler no” → reactions not included

Default: reactions not included

top back