The thermodynamics package∗

Karl D. Hammond
hammondkd@missouri.edu

2022/04/29

Abstract

A package, thermodynamics, is defined that makes typesetting quantities found in thermodynamics texts relatively simple. The commands are flexible and intended to be relatively intuitive. It handles several sets of notation for total, specific, and molar quantities; allows changes between symbols (e.g., $A$ vs. $F$ for Helmholtz free energy); and greatly simplifies the typesetting of symbols and partial derivatives commonly encountered in mixture thermodynamics. Changes of one’s notes from one textbook to another can be achieved relatively easily by changing package options.

1 Introduction

The purpose of this package is to simplify the typesetting of equations in thermodynamics, specifically chemical engineering thermodynamics, which are often cumbersome to enter. For example, consider the following equation:

$$dU = \left(\frac{\partial U}{\partial S}\right)_{V,\vec{n}} dS + \left(\frac{\partial U}{\partial V}\right)_{S,\vec{n}} dV + \sum_{i=1}^C \left(\frac{\partial U}{\partial n_i}\right)_{S,V,n_{j\neq i}} dn_i.$$  \hspace{1cm} (1)

This equation is pretty basic, and equations like it occur all the time in thermodynamics. Without this package, you might typeset it like this:

\begin{verbatim}
d\underline{U} = \left(\frac{\partial \underline{U}}{\partial \underline{S}}\right)_{\underline{V},\vec{n}} d\underline{S} + \left(\frac{\partial \underline{U}}{\partial \underline{V}}\right)_{S,\vec{n}} d\underline{V} + \sum_{i=1}^C \left(\frac{\partial \underline{U}}{\partial n_i}\right)_{S,V,n_{j\neq i}} dn_i.
\end{verbatim}

This is a lot of code, and even then the output is slightly clunky:

$$d\underline{U} = \left(\frac{\partial \underline{U}}{\partial \underline{S}}\right)_{\underline{V},\vec{n}} d\underline{S} + \left(\frac{\partial \underline{U}}{\partial \underline{V}}\right)_{S,\vec{n}} d\underline{V} + \sum_{i=1}^C \left(\frac{\partial \underline{U}}{\partial n_i}\right)_{S,V,n_{j\neq i}} dn_i.$$  \hspace{1cm} (1)

∗This document corresponds to thermodynamics v1.00, dated 2022/04/29.
It is also frustratingly difficult to change one’s notes or handouts from one textbook that uses, say, \( n_1 \) to denotes moles of component 1 to another textbook that uses \( N_1 \) for the same quantity, or perhaps denotes the total internal energy as \( U \) or \( U' \) rather than \( U \). For example, if you wanted it to be

\[
dU = \left( \frac{\partial U}{\partial S} \right)_{V,N_1,...,N_n} dS + \left( \frac{\partial U}{\partial V} \right)_{S,N_1,...,N_n} dV + \sum_{i=1}^{n} \left( \frac{\partial U}{\partial N_i} \right)_{S,V,N_1,...,N_n} dN_i
\]

without changing any of your code—to update it across all handouts, exams, and homework sets after changing textbooks, say—you would be out of luck (or in for a lot of work).

With this package, you could reduce the code to typeset this equation to

\[
\begin{equation}
\frac{dU}{dSt} = \frac{\partial U}{\partial S} \frac{dS}{dSt} + \frac{\partial U}{\partial V} \frac{dV}{dVt} + \sum_{i=1}^{n} \frac{\partial U}{\partial N_i} \frac{dN_i}{dNt_i}
\end{equation}
\]

and it will render similarly to Equation (1), including the shortened underscores and negative kerning. If you later decide to change the notation such that extensive properties are not underlined, you can do that without changing any of your code (just change a package option). Similarly, if you want \( \vec{n} \) replaced by \( n_1,...,n_C \), you can do that with a package option, too.

The package handles second derivatives, too. For example,

\[
\begin{align*}
\frac{\partial H}{\partial T} &= T \frac{\partial S}{\partial T} = -T \frac{\partial^2 G}{\partial T^2} = C_P
\end{align*}
\]

renders (using the default options)

\[
\frac{\partial G_i}{\partial P}_{T,\vec{n}} = \frac{\partial^2 G_i}{\partial P \partial n_i}_{T,\vec{n},\vec{n}_i} = \frac{\partial^2 G_i}{\partial n_i \partial P}_{T,\vec{n}_i} = \frac{\partial V}{\partial n_i}_{T,P,\vec{n}_j,\vec{n}_i} = V_{n_i}
\]

using the defaults.

2 Using the Package

There are three categories of macros defined in this package: macros that produce symbols (or groups of them), macros that typeset derivatives, and macros that are used internally that the user need not know about. There are also several environments that allow the user to switch locally between different delimiters on partial derivatives.
2.1 Predefined Symbols

The macros used to produce symbols fall into five categories: extensive properties, molar properties, specific properties (i.e., per unit mass), partial molar properties, and shortcut macros (e.g., macros for the heat capacities, saturation pressure, and so forth). The macros corresponding to extensive, molar, and specific properties are shown in Table 1. Examples using the enthalpy are

\[ \Ht \quad \Hm \quad \Hs \quad \Hpm_i. \]

Using the default package options, the above renders as

\[ H \quad H \hat{H} \overline{H}_i. \]

In addition, the properties in Table 2 are defined for convenience. How these symbols are rendered can be customized by package options. As long as the user consistently uses \Ut to render the total internal energy, \Um to render the molar internal energy, and so forth, switching notation from, say, \U to \Ut for extensive properties is trivial.

The heat capacities (see Table 2) are generally assumed to be molar (e.g., \cP is interpreted to be the molar heat capacity). To get the specific heat capacities, \cPs and \cVs are provided, which by default render as \overline{C}_P and \overline{C}_V, respectively. There are also extensive versions, so \cPt and \cVt will render as \overline{C}_P and \overline{C}_V, respectively. Note that \cP and friends require you to surround \text with brackets if the portion with \text in it is not the last argument. For example, \cP_i\text{A} will work as expected, but \cP\text{A}_i will not: you need to use \cP^\text{A}_i. Using \cP^\IG_i will work as expected.

You can also get partial molar heat capacities via \cPpm and \cVpm, though the latter’s mathematical definition is a bit hard to wrap one’s head around:

\[ \cPpm_i = \partial \overline{C}_P \partial n_i ]_{T,P,n_{j\neq i}} \]

yields

\[ \overline{C}_{V,i} = \left( \frac{\partial C_V}{\partial n_i} \right)_{T,P,n_{j\neq i}} = \frac{\partial}{\partial n_i} \left[ -T \left( \frac{\partial^2 A}{\partial T^2} \right) V,\overline{\rho} \right]_{T,P,n_{j\neq i}} \]

2.2 Partial Molar Properties

Partial molar quantities are typically defined with the suffix pm. For example, \Upm refers to the partial molar internal energy. There are two options for how to enter partial molar quantities: as commands or as super/subscripts. For example,

\[ \Upm[i] \quad \Upm[\IG][i] \quad \Hpm_i \quad \Hpm[\IG][i] \]

will typeset as

\[ \overline{U}_i \quad \overline{U^\IG}_i \quad \overline{H}_i \quad \overline{H^\IG}_i \]

There are also partial molar heat capacities available via \cPpm and \cVpm. Important: The \text command defined by the amstext package is usually robust enough
Table 1. Commands defined in this package to represent extensive thermodynamic quantities and their molar and specific analogs. These macros should be used even if the symbol the user wishes to use does not match the command used (e.g., $\Ft$ for total Helmholtz free energy even if it ends up being set as $A$).

<table>
<thead>
<tr>
<th>Property</th>
<th>Total</th>
<th>Molar</th>
<th>Specific</th>
<th>Partial Molar</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heat</td>
<td>$\Qt$</td>
<td>$\Qm$</td>
<td>$\Qs$</td>
<td>N/A</td>
</tr>
<tr>
<td>Work</td>
<td>$\Wt$</td>
<td>$\Wm$</td>
<td>$\Ws$</td>
<td>N/A</td>
</tr>
<tr>
<td>Total energy</td>
<td>$\Et$</td>
<td>$\Em$</td>
<td>$\Es$</td>
<td>$\Epm$</td>
</tr>
<tr>
<td>Internal energy</td>
<td>$\Ut$</td>
<td>$\Um$</td>
<td>$\Us$</td>
<td>$\Upm$</td>
</tr>
<tr>
<td>Enthalpy</td>
<td>$\Ht$</td>
<td>$\Hm$</td>
<td>$\Hs$</td>
<td>$\Hpm$</td>
</tr>
<tr>
<td>Entropy</td>
<td>$\St$</td>
<td>$\Sm$</td>
<td>$\Ss$</td>
<td>$\Spm$</td>
</tr>
<tr>
<td>Volume</td>
<td>$\Vt$</td>
<td>$\Vm$</td>
<td>$\Vs$</td>
<td>$\Vpm$</td>
</tr>
<tr>
<td>Helmholtz free energy</td>
<td>$\Ft$</td>
<td>$\Fm$</td>
<td>$\Fs$</td>
<td>$\Fpm$</td>
</tr>
<tr>
<td>Gibbs free energy</td>
<td>$\Gt$</td>
<td>$\Gm$</td>
<td>$\Gs$</td>
<td>$\Gpm$</td>
</tr>
<tr>
<td>Surface area</td>
<td>$\At$</td>
<td>$\Am$</td>
<td>$\As$</td>
<td>$\Apm$</td>
</tr>
<tr>
<td>Grand potential*</td>
<td>$\Lt$</td>
<td>$\Lm$</td>
<td>$\Ls$</td>
<td>$\Lpm$</td>
</tr>
<tr>
<td>Moles</td>
<td>$\Nt$</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>$B$ (generic property)</td>
<td>$\Bt$</td>
<td>$\Bm$</td>
<td>$\Bs$</td>
<td>$\Bpm$</td>
</tr>
<tr>
<td>$M$ (generic property)</td>
<td>$\Mt$</td>
<td>$\Mm$</td>
<td>$\Ms$</td>
<td>$\Mpm$</td>
</tr>
</tbody>
</table>

*The grand potential, $\Omega(T, V, \vec{\mu}) = U - TS - \sum_i \mu_i n_i$, is also called the Landau free energy by some authors.

that something like $\Um^{\text{L}}$ will work as expected, without additional braces. This does not work for partial molar properties; for example, $\Hpm^{\text{L}}_i$ will produce an error, as will $\Hpm_i^{\text{L}}$. The expression $\Hpm_i^{\text{L}}$ will work as expected.

New partial molar properties can be defined for any “simple” symbol using the \partialmolar macro. “Simple” means it has no subscripts or superscripts. For example, the macro for the partial molar Gibbs free energy is defined via the macro

\newcommand*{\Gpm}{\partialmolar{\Gibbs@symbol}}

A list of pre-defined macros for total, molar, specific, and partial molar quantities commonly used in thermodynamics is included in Table 1.

2.3 Other Predefined Symbols and Modifiers

There are a number of predefined symbols and modifiers. While these symbols could be defined or used without these macros, such use is not recommended: changing package options will result in inconsistencies if these macros are not used.

2.3.1 Heat Capacities, Compressibilities, and Expansivities

The isobaric and isochoric heat capacities are produced with \cP and \cV, respectively. Four other measurable quantities are defined: the isothermal and isentropic compressibilities, $\kappa_T$ and $\kappa_S$, respectively; and the isobaric and isentropic volume expansivities, $\alpha_P$ and $\alpha_S$, respectively. Some textbooks use $\beta$ instead of $\alpha$ for the volume expansivity to differentiate it from the linear expansivity;
Table 2. Convenience macros and their default symbols. These are generally “smart”: for example, \( \text{cP}_i \) renders as \( C_P \), as expected, and \( \text{cP} \circ_i \) renders as \( C \circ P_i \), also as expected. You can also reverse it: \( \text{cP} \circ_i \) becomes \( C \circ P_i \).

<table>
<thead>
<tr>
<th>Name</th>
<th>Macro</th>
<th>Sym.</th>
<th>Definition</th>
<th>Base Symbol Macro</th>
</tr>
</thead>
<tbody>
<tr>
<td>Isobaric heat capacity</td>
<td>( \text{cP}^a )</td>
<td>( C_P )</td>
<td>( T \left( \frac{\partial S}{\partial T} \right)_p )</td>
<td>\text{heatcapacitysymbol}</td>
</tr>
<tr>
<td>Isochoric heat capacity</td>
<td>( \text{cV}^a )</td>
<td>( C_V )</td>
<td>( T \left( \frac{\partial V}{\partial T} \right)_V )</td>
<td>\text{heatcapacitysymbol}</td>
</tr>
<tr>
<td>Isothermal compressibility</td>
<td>( \kappa_T )</td>
<td>( \kappa_T )</td>
<td>( -\frac{1}{V} \left( \frac{\partial V}{\partial P} \right)_T )</td>
<td>\text{compressibilitysymbol}</td>
</tr>
<tr>
<td>Isentropic compressibility</td>
<td>( \kappa_S )</td>
<td>( \kappa_S )</td>
<td>( -\frac{1}{V} \left( \frac{\partial V}{\partial S} \right)_S )</td>
<td>\text{compressibilitysymbol}</td>
</tr>
<tr>
<td>Isobaric expansivity</td>
<td>( \alpha_P )</td>
<td>( \alpha_P )</td>
<td>( \frac{1}{V} \left( \frac{\partial V}{\partial T} \right)_p )</td>
<td>\text{expansivitysymbol}</td>
</tr>
<tr>
<td>Isentropic expansivity</td>
<td>( \alpha_S )</td>
<td>( \alpha_S )</td>
<td>( \frac{1}{V} \left( \frac{\partial V}{\partial T} \right)_S )</td>
<td>\text{expansivitysymbol}</td>
</tr>
<tr>
<td>Pure fugacity</td>
<td>( \text{fpure} )</td>
<td>( f )</td>
<td>( \phi_P )</td>
<td></td>
</tr>
<tr>
<td>Mixture fugacity</td>
<td>( \text{fmix} )</td>
<td>( f )</td>
<td>( f_i = x_i \phi_P )</td>
<td></td>
</tr>
<tr>
<td>Saturation fugacity</td>
<td>( \text{fsat} )</td>
<td>( f_{\text{sat}} )</td>
<td>( \phi_{\text{sat} P_{\text{sat}}} )</td>
<td>\text{sat}</td>
</tr>
<tr>
<td>Pure fugacity coeff.</td>
<td>( \text{phipure} )</td>
<td>( \phi )</td>
<td>( \phi_i = \exp \left( \frac{1}{RT} \int_0^P V_i(T, p) - \frac{RT}{p} dp \right) )</td>
<td></td>
</tr>
<tr>
<td>Mixture fugacity coeff.</td>
<td>( \text{phimix} )</td>
<td>( \phi )</td>
<td>( \phi_i = \exp \left( \frac{1}{RT} \int_0^P V_i(T, p, \vec{x}) - \frac{RT}{p} dp \right) )</td>
<td></td>
</tr>
<tr>
<td>Henry’s constant (rational)</td>
<td>( \text{Henryrat} )</td>
<td>( h )</td>
<td>( \gamma_i^\infty f_i )</td>
<td></td>
</tr>
<tr>
<td>Henry’s constant (molal)</td>
<td>( \text{Henrymol} )</td>
<td>( \mathcal{H} )</td>
<td>( M_s \gamma_i^\infty f_i )</td>
<td></td>
</tr>
<tr>
<td>Rational activity coeff.</td>
<td>( \text{gammarat} )</td>
<td>( \gamma^* )</td>
<td>( \gamma/\gamma^\infty )</td>
<td></td>
</tr>
<tr>
<td>Molal activity coeff.</td>
<td>( \text{gammamol} )</td>
<td>( \gamma^\infty )</td>
<td>( x_s \gamma/\gamma^\infty )</td>
<td></td>
</tr>
<tr>
<td>Saturation fugacity coeff.</td>
<td>( \text{phisat} )</td>
<td>( \phi_{\text{sat}} )</td>
<td>( \phi(T, P_{\text{sat}}) )</td>
<td>\text{sat}</td>
</tr>
<tr>
<td>Saturation pressure</td>
<td>( \text{Psat} )</td>
<td>( P_{\text{sat}} )</td>
<td></td>
<td>\text{sat}</td>
</tr>
<tr>
<td>Vapor pressure</td>
<td>( \text{Pvap} )</td>
<td>( \text{Pvap} )</td>
<td>Currently a synonym for ( \text{Psat} )(^b)</td>
<td></td>
</tr>
<tr>
<td>Standard state</td>
<td>( \text{std} )</td>
<td>( \circ )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standard pressure</td>
<td>( \text{Pstd} )</td>
<td>( P^\circ )</td>
<td>( f(T, P^\circ) )</td>
<td>\text{std}</td>
</tr>
<tr>
<td>Standard fugacity</td>
<td>( \text{std} )</td>
<td>( f^\circ )</td>
<td>( f(T, P^\circ) )</td>
<td>\text{std}</td>
</tr>
<tr>
<td>Change on mixing</td>
<td>( \text{Deltamix}^d )</td>
<td>( \Delta M_{\text{mix}} )</td>
<td>( \sum_i x_i M_i )</td>
<td>\text{mixing}</td>
</tr>
<tr>
<td>Change on reaction</td>
<td>( \text{Deltarxn}^d )</td>
<td>( \Delta M_{\text{rxn}} )</td>
<td>( \sum_i \nu_i M_i )</td>
<td>\text{reaction}</td>
</tr>
<tr>
<td>Change on melting</td>
<td>( \text{Deltafus}^d )</td>
<td>( \Delta M_{\text{fus}} )</td>
<td>( M^L - M^S )</td>
<td>\text{fusion}</td>
</tr>
<tr>
<td>Change on boiling</td>
<td>( \text{Deltavap}^d )</td>
<td>( \Delta M_{\text{vap}} )</td>
<td>( M^V - M^L )</td>
<td>\text{vaporization}</td>
</tr>
<tr>
<td>Change on subliming</td>
<td>( \text{Deltasub}^d )</td>
<td>( \Delta M_{\text{sub}} )</td>
<td>( M^V - M^S )</td>
<td>\text{sublimation}</td>
</tr>
</tbody>
</table>

\(^a\)Extensive and specific (per-unit-mass) versions are available as \( \text{cP}^t \) and \( \text{cPs} \), respectively, with similar macros for the isochoric heat capacity.

\(^b\)If you want \( \text{Pvap} \) to produce \( P_{\text{vap}} \) instead of \( P_{\text{sat}} \), you should redefine the \( \text{sat} \) macro.

\(^c\)Typical usage would be $\mu_i = \mu_i^{\text{std}} + RT \log a_i$, yielding $\mu_i = \mu_i^{\circ} + RT \log a_i$.

\(^d\)The usual usage would be something like $\$\text{Deltamix}\Ve\text{IGM} = 0$. 

5
this can be changed by redefining \expansivitysymbol, which is done automatically by some of the package options that create notation specific to a particular textbook.

### 2.3.2 Fugacities and Fugacity Coefficients

\fpure, \fmix
\phipure, \phimix

Different textbooks use different modifications of the symbol \textit{f} for fugacity, so it is recommended to use the macro \fpure to denote the pure-component fugacity and \fmix to denote the mixture fugacity. Similarly, the pure-component fugacity coefficient should be generated with \phipure, and that in the mixture should be \phimix.

For example, the following markup is an example of a common equation in mixture thermodynamics:

\[ \fmix_j = x_j \phimix_j P = x_j \gamma_j \fpure_j. \]

With the default package options, this produces

\[ \hat{f}_j = x_j \gamma_j \hat{f}_j. \]

With the Thompson package option, however, the same markup produces

\[ \hat{f}_j = x_j \gamma_j \hat{f}_j^*. \]

Similarly, the Prausnitz package option causes it to generate

\[ f_j = x_j \phi_j P = x_j \gamma_j f_{\text{pure},j}, \]

and the Sandler option causes it to generate

\[ \bar{f}_j = x_j \bar{\phi}_j P = x_j \gamma_j f_j. \]

### 2.3.3 Activity Coefficients and Henry’s Constants

\gammarat, \gammamol
\Henryrat, \Henrymol

The activity coefficient can be generated with \textit{\gamma}, as usual. The Henry’s Law activity coefficients should be produced with \gammarat (rational basis) and \gammamol (molal basis). There are also macros to generate the Henry’s law constants for both the rational basis (\Henryrat) and the molal basis (\Henrymol). These are interrelated:

\[ \fmix_i = x_i \gamma_i \fpure_i = x_i \gamma_i h_i = C_i \gamma_i \gamma_i h_i \]

using the default options.

### 2.3.4 Saturation Properties

\Psat, \Pvap, \fsat, \phisat

The saturation pressure is generated with \Psat. The macro \Pvap is an alias for \Psat. The fugacity and fugacity coefficient at saturation are accessed via \fsat and \phisat, respectively. Package options can be used to change some of these to match the notation of specific textbooks.

\sat

The \textit{\sat} macro is used “behind the scenes” as part of \Psat, \fsat, and \phisat, which produce \textit{P} sat, \textit{f} sat, and \textit{\phi} sat, respectively. If you wanted to redefine them to be \textit{P} vap, \textit{f} vap, and \textit{\phi} vap, you could simply redefine \textit{\sat} with \renewcommand*{\sat}{\text{vap}}. This is done automatically using package option Sandler.
2.3.5 Standard States

The symbol ° (\circ) is used by default for standard states. This is intended to be easy to change should the user want to replace $P^o$ with $P^\circ$, say. This is accessed via the $P^\circ$ macro. The macro $P^\circ$ is defined as $P^\circ$ for convenience to denote standard pressures, and $f^\circ$ is defined for standard fugacities so as to ease implementation across textbooks.

The usual usage would be something like this:

$$\Delta_{\text{G}}_{\text{rxn}} = \sum_{i=1}^C \nu_i \mu_i = \sum_{i=1}^C \left[ \nu_i \mu_i^\circ + \nu_i RT \log \left( \frac{f_i^\circ}{f_i^\circ} \right) \right] = \Delta_{\text{G}}_{\text{rxn}}^\circ + RT \log \left[ \prod_{i=1}^C \alpha_i^{\nu_i} \right]$$

which produces

$$\Delta_{\text{G}}_{\text{rxn}} = \sum_{i=1}^C \nu_i \mu_i = \sum_{i=1}^C \left[ \nu_i \mu_i^\circ + \nu_i RT \log \left( \frac{f_i^\circ}{f_i^\circ} \right) \right] = \Delta_{\text{G}}_{\text{rxn}}^\circ + RT \log \left[ \prod_{i=1}^C \alpha_i^{\nu_i} \right]$$

with the default settings.

2.3.6 Changes on Mixing, Reaction, Fusion, Vaporization, and Sublimation

Mixing properties are handled via the $\Delta_{\text{mix}}$ macro, and are used as in the following example:

$$\Delta_{\text{mix}} \Delta_{\text{G}} = \Delta_{\text{G}} - \sum_{i=1}^C x_i \Delta_{\text{G}}_i = \Delta_{\text{mix}} \Delta_{\text{H}} = T \Delta_{\text{S}}$$

which yields

$$\Delta_{\text{G}}_{\text{mix}} = G - \sum_{i=1}^C x_i G_i = \Delta_{\text{H}}_{\text{mix}} - T \Delta_{\text{S}}_{\text{mix}}.$$
results in

\[ \Delta H_{\text{rxn}}^\circ = \sum_{i=1}^{C} v_i \Delta H_{f,i}^\circ. \]

It is not anticipated that this command will be combined with something like a heat capacity, which already has a (potentially double) subscript, but as there is no "formation" heat capacity, that should not present a problem.

### 2.4 Residual and Excess Properties

Additional macros are defined that make it easy to typeset the residual (also called "departure") and excess total, molar, specific, and partial molar properties. These macros follow the same pattern: \UR, \URt, \URs, and \URpm typeset the molar, total, specific, and partial molar residual internal energies, respectively, and by default expand to \( U_R^R \), \( U_R^T \), \( \hat{U}_R^R \), and \( U_R^P \) (the last is called as \URpm{i} or \URpm{-i}). Similarly, \UE, \UEt, \UES, and \UEpm typeset the corresponding excess properties. The first character of the macros for other properties follow the same pattern as in Table 1.

The \( R \) and \( E \) characters are generated by the macros \residual and \excess, respectively. These macros can be redefined; for example, if you want \SE, which normally produces \( S^E \), to give you \( S^{EX} \)—and let’s be honest, who doesn’t want that?\(^1\)—then you can redefine it with

\[
\renewcommand*{\excess}{{EX}}
\]

or possibly

\[
\renewcommand*{\excess}{{\mathrm{EX}}}
\]

or even

\[
\renewcommand*{\excess}{\text{EX}}
\]

which cause \SE to expand to \( S^{EX} \), \( S^{EX} \), and \( S^{EX} \), respectively.

It is generally not possible to use superscripts with the excess or residual properties; in the event the user needs this, the \excess and \residual macros can be used directly, viz.,

\[
\begin{gather*}
\begin{align*}
\Hm^\excess,\std &= \HE(T, P\std) \\
\renewcommand*{\excess}{{EX}}
\Hm^\excess,\std &= \HE(T, P\std)
\end{align*}
\end{gather*}
\]

which yields

\[
H^{E,\circ} = \HE(T, P^\circ)
\]

\[
H^{EX,\circ} = \HEX(T, P^\circ)
\]

using the default options.

---

\(^1\) You knew that joke was coming.


2.5 Partial Derivatives

\Partial \Partial*

Partial derivatives are easily rendered using the \Partial command. There is a starred form (\Partial*) that additionally adjusts the spacing after the closing symbol to remove some of the space, anticipating that the following binary operator will overhang the subscripts. Compare the following:

\begin{gather*}
\Partial{H}{T}{P} = C_P \quad \Partial*{H}{T}{P} = C_P
\end{gather*}

which yields

\[ \left( \frac{\partial H}{\partial T} \right)_p = C_P \quad \left( \frac{\partial H}{\partial T} \right)_p = C_P \]

\PartialSecond \PartialSecond* \PartialMixSecond \PartialMixSecond*

Second partial derivatives and mixed-second partial derivatives are typeset with the \PartialSecond and \PartialMixSecond commands, respectively. Like the first-order variety, these also have starred versions that remove the space immediately following the closing symbols, anticipating that the equals sign or other binary operator following the derivative will overhang the elements held constant. For example,

\[ \Vpm_i = \Partial*{V_t}{N_t_i}{T,P,\allNsbut{i}} = \PartialMixSecond*{G_t}{N_t_i}{P}{T,\allNsbut{i}} = \Partial*{G_{pm_i}}{P}{T,\allNs} \]

looks like

\[ \V_i = \left( \frac{\partial V}{\partial n_i} \right)_{T,P,n_{j \neq i}} = \left( \frac{\partial^2 G}{\partial n_i \partial P} \right)_{T,n_{j \neq i}} = \left( \frac{\partial^2 G}{\partial P \partial n_i} \right)_{T,n_{j \neq i}} = \left( \frac{\partial G_i}{\partial P} \right)_{T,n_{j \neq i}}. \]

\PartialBigg \PartialSecondBigg \PartialMixSecondBigg \PartialBigg \PartialSecondBigg \PartialMixSecondBigg

There are instances (such as the line above) when \Partial causes parentheses that are slightly too tall but do not need to be—particularly when partial molar properties, specific quantities, or fugacities are involved. The macro \PartialBigg uses ansmath’s \Biggl and \Biggr macros in place of \left and \right to size the parentheses accordingly; \PartialBigg uses \biggl and \biggr in a similar fashion. For example, compare the following:

\[ \left[ \Vpm_i = \Partial*{V_t}{N_t_i}{T,P,\allNsbut{i}} = \Partial*{Gpm_i}{P}{T,\allNs} = \PartialBigg*{Gpm_i}{P}{T,\allNs} = \RT \PartialBigg*{\log \hat f_i}{P}{T,\allNs} = \RT \PartialBigg*{\log \hat f_i}{P}{T,\allNs} \right] \]

which typesets as

\[ \V_i = \left( \frac{\partial V}{\partial n_i} \right)_{T,P,n_{j \neq i}} = \left( \frac{\partial G_i}{\partial P} \right)_{T,n_{j \neq i}} = \RT \left( \frac{\partial \log \hat f_i}{\partial P} \right)_{T,n_{j \neq i}} = \RT \left( \frac{\partial \log \hat f_i}{\partial P} \right)_{T,n_{j \neq i}} = \RT \left( \frac{\partial \log \hat f_i}{\partial P} \right)_{T,n_{j \neq i}} \]

Note that a similar effect—possibly with other side effects—can be achieved with ansmath’s \smash command, which has the effect of removing all vertical space associated with a particular character. Observe:
\[
\Vpm_i = \frac{\partial \V_t}{\partial \Nt_i} \{T,P,\allNsbut{i}\}
= \frac{\partial \Gpm_i}{\partial \P} \{T,\allNs\}
\]

produces

\[
\V_i = \left( \frac{\partial \V}{\partial \n_i} \right)_{T,P,n_j,i} = \left( \frac{\partial \G_i}{\partial \P} \right)_{T,n_j,i}.
\]

Note that the vertical spacing is not quite as good here as it was above. This can be fixed by using the optional argument to \texttt{\smash}, viz.,

\[
\Vpm_i = \frac{\partial \V_t}{\partial \Nt_i} \{T,P,\allNsbut{i}\}
= \frac{\partial \smash[t]{\Gpm_i}}{\partial \P} \{T,\allNs\}
\]

\[
\V_i = \left( \frac{\partial \V}{\partial \n_i} \right)_{T,P,n_j,i} = \left( \frac{\partial \G_i}{\partial \P} \right)_{T,n_j,i}.
\]

It is possible to “fake” higher-order derivatives via some trickery. For example,

\[
\cPpm_i = T\frac{\partial \Spm_i}{\partial T} \{P,\allXs\}
= T\frac{\partial^2 \St}{\partial^2 \Nt_i} \{P,\allNsbut{i}\}
= -T\frac{\partial^3 \Gt}{\partial^2 T \partial \Nt_i} \{P,\allNsbut{i}\}
\]

gives

\[
\overline{C}_{P,i} = T \left( \frac{\partial \S_i}{\partial T} \right)_{P,X} = T \left( \frac{\partial^2 \S}{\partial T^2} \right)_{P,n_{i}} = -T \left( \frac{\partial^3 \G}{\partial T^2 \partial \Nt_i} \right)_{P,n_{j,i}},
\]

which is probably pretty close to what you wanted. Using this trickery with the package option \texttt{nosubscripts} will not work as well, and the use of third- and higher-order derivatives with this package should generally be considered unsupported.

### 2.6 Holding Constant the Number of Moles of Several Species

It is common in thermodynamics to use notation such as

\[
\V_k = \left( \frac{\partial \V}{\partial \n_k} \right)_{T,P,n_{j\neq k}}
\]

or perhaps

\[
\V_k = \left( \frac{\partial \V}{\partial \n_k} \right)_{T,P,n_1,...,n_l,...,n_C}
\]

to mean partial derivatives that hold the number of moles of each species constant except the one being changed. Similarly, a property determined with all mole fractions held constant might be written

\[
\overline{C}_P = \left( \frac{\partial \H}{\partial T} \right)_{P,X}
\]

or perhaps

\[
\overline{C}_P = \left( \frac{\partial \H}{\partial T} \right)_{P,X_1,...,X_C}.
\]
There are several macros that standardize such constructs. The \texttt{allNs} macro expands to something meaning the number of moles of all species; by default, this is $\vec{n}$ (package option moles-index), but can be changed to $n_1, \ldots, n_C$ using the package option moles-range. Similarly, the macros \texttt{allmus} and \texttt{allmusbut} do the same but with $n$ replaced by $\mu$, and \texttt{allMsbut} is the same with $m$ instead of $n$. There are analogous macros for mole fractions, namely \texttt{allXs} and \texttt{allXsbut} for $x$ and \texttt{allYs} and \texttt{allYsbut} for $y$, as well as \texttt{allWs} and \texttt{allWsbuts} for mass fractions—these implicitly assume that all mole or mass fractions except the last are used as variables. The macros \texttt{allNsbut}, \texttt{allMsbut}, \texttt{allmusbut}, \texttt{allXsbut}, \texttt{allYsbut}, and \texttt{allWsbut} take an optional argument; for example,

\begin{verbatim}
\Partial{\Ht}{\Nt_1}{T,P,\allNsbut{1}} = \Partial{\Ht}{\Nt_1}{T,P,\allNsbut[m]{1}} = \Hpm_1
\end{verbatim}

typesets as

$$\left( \frac{\partial H}{\partial n_1} \right)_{T,P,n_{i\neq 1}} = \left( \frac{\partial H}{\partial n_1} \right)_{T,P,n_{m\neq 1}} = \Hpm_1$$

Similarly,

\begin{verbatim}
\Partial{\Hm}{x_i}{T,P,\allXsbut{i}} = \Partial{\Hm}{x_i}{T,P,\allXsbut[m]{i}} = \Hpm_i - \Hpm_{\text{ncomponents}}
\end{verbatim}

becomes

$$\left( \frac{\partial H}{\partial x_i} \right)_{T,P,x_{j\neq i,C}} = \left( \frac{\partial H}{\partial x_i} \right)_{T,P,x_{m\neq i,C}} = \Hpm_i - \Hpm_C$$

The user must supply their own redefinition if they wish to hold something other than \texttt{ncomponents} constant in addition to the argument. Using the moles-range package option, for which \texttt{allXsbut{k}} expands to $x_1, \ldots, [x_k], \ldots, x_{C-1}$ rather than $x_{j\neq k,C}$, the optional argument is ignored.

Users can define new “all but” macros using the \texttt{allbut} and \texttt{allbutlastand} commands. For example,

\begin{verbatim}
\newcommand*{\allNsbut}{[2][j]{\allbut[#1]{#2}{\Nt}}}
\newcommand*{\allXsbut}{[2][j]{\allbutlastand[#1]{#2}{x}}}
\end{verbatim}

are the definitions of \texttt{allNsbut} and \texttt{allXsbut}, respectively.

### 2.7 Sums and Products

It is common to require sums and products such as

$$\sum_{i=1}^{C} x_i = 1 \quad \text{or} \quad x_C = 1 - \sum_{i=1}^{C-1} x_i \quad \text{or} \quad G = \sum_{j=1}^{C} \mu_j n_j \quad \text{and} \quad K = \prod_{k=1}^{C} a_k^{\nu_k}.$$ 

This package defines shortcuts to typeset such terms thus:

\begin{verbatim}
\[ \sumall_i x_i = 1 \quad \text{quad} \text{or} \quad \sumallbutlast_i x_i \quad \text{quad} \text{or} \quad \sumall \quad \text{quad} \text{or} \quad \prodall_k a_k^{\nu_k} \quad \text{quad} \text{or} \quad \prodall_{k=1}^{C} a_k^{\nu_k}. \]
\end{verbatim}

The symbol $C$ can be changed by redefining \texttt{ncomponents}. This is done automatically by some package options (e.g., \texttt{TesterModell} changes it to $n$; \texttt{Sandler} changes it to $C$; \texttt{Thompson} changes it to $c$).
Table 3. Options controlling which symbols to use by default. The macros \Et, \Ut, \Ft, \Gt, \Ht, \At, and \Nt represent the total energy, internal energy, Helmholtz free energy, Gibbs free energy, enthalpy, surface area, and number of moles, respectively. Symbols are shown as they would appear with the (default) option intensive-plain.

<table>
<thead>
<tr>
<th>Option</th>
<th>\Et</th>
<th>\Ut</th>
<th>\Ft</th>
<th>\Gt</th>
<th>\Ht</th>
<th>\At</th>
<th>\Nt</th>
</tr>
</thead>
<tbody>
<tr>
<td>EUAGHan</td>
<td>E</td>
<td>U</td>
<td>A</td>
<td>G</td>
<td>H</td>
<td>a</td>
<td>n</td>
</tr>
<tr>
<td>EUAGHAn</td>
<td>E</td>
<td>U</td>
<td>A</td>
<td>G</td>
<td>H</td>
<td>a</td>
<td>N</td>
</tr>
<tr>
<td>EUHAGan</td>
<td>(synonym for EUAGHan)</td>
<td>EUHAGaN</td>
<td>(synonym for EUAGHAn)</td>
<td>EUFGHan</td>
<td>E</td>
<td>U</td>
<td>F</td>
</tr>
<tr>
<td>EUFGHAN</td>
<td>E</td>
<td>U</td>
<td>F</td>
<td>G</td>
<td>H</td>
<td>A</td>
<td>N</td>
</tr>
<tr>
<td>EEFGHan</td>
<td>E</td>
<td>E</td>
<td>F</td>
<td>G</td>
<td>H</td>
<td>A</td>
<td>N</td>
</tr>
<tr>
<td>EEFGHAN</td>
<td>E</td>
<td>E</td>
<td>F</td>
<td>G</td>
<td>H</td>
<td>A</td>
<td>N</td>
</tr>
<tr>
<td>EEAGHan</td>
<td>E</td>
<td>E</td>
<td>A</td>
<td>G</td>
<td>H</td>
<td>a</td>
<td>N</td>
</tr>
<tr>
<td>EUAGHan</td>
<td>E</td>
<td>U</td>
<td>A</td>
<td>G</td>
<td>H</td>
<td>A</td>
<td>n</td>
</tr>
<tr>
<td>EUAGHAN</td>
<td>E</td>
<td>U</td>
<td>A</td>
<td>G</td>
<td>H</td>
<td>A</td>
<td>N</td>
</tr>
<tr>
<td>EUFGHan</td>
<td>E</td>
<td>U</td>
<td>F</td>
<td>G</td>
<td>H</td>
<td>a</td>
<td>N</td>
</tr>
<tr>
<td>EUFGHAN</td>
<td>E</td>
<td>U</td>
<td>F</td>
<td>G</td>
<td>H</td>
<td>a</td>
<td>N</td>
</tr>
</tbody>
</table>

3 Loading the Package

To load the package with the defaults enabled, load it the usual way:

\usepackage{thermodynamics}

The package options loaded by default are EUAGHan, subscripts, parentheses, intensive-plain, and moles-index. These define, respectively, the default symbols to use for total energy, internal energy, Helmholtz free energy, and so forth; the manner of writing partial derivatives; the delimiters around partial derivatives; and the manner of denoting extensive, molar, and specific properties. The default behavior can be altered by options in the following section.

3.1 Package Options

There are three categories of options: options that affect which symbols are used, options that affect how symbols are decorated, and options that affect how partial derivatives are displayed. These are presented in turn.

3.1.1 Options that Change Symbol Sets

There are several options that choose the set of symbols to use for total energy, internal energy, Helmholtz free energy, and so forth. These are summarized in Table 3. The default is EUAGHan.

Using EUAGHan (the default), we might use the following markup:

\[ \Ft = \Ut - TVt + \sum_i \mu_i Nt_i + \sigma d\At \quad qquad \Hm = Um + PVm \quad qquad Et = Ut + \frac{1}{2} mv^2 \]
which would look like

\[ A = U - TS = -PV + \sum_i \mu_i n_i + \sigma \delta \quad H = U + PV \quad E = U + \frac{1}{2}mv^2. \]

Using the `EUFGHAN` option, the same markup would yield

\[ F = U - TS = -PV + \sum_i \mu_i N_i + \sigma \delta \quad H = U + PV \quad E = U + \frac{1}{2}mv^2. \]

### 3.1.2 Options for Extensive vs. Molar Properties

There are four sets of notation that define how extensive properties are represented, as shown in Table 4. The default is intensive-plain, which (using the volume as an example) represents the total, molar, specific, and partial molar volumes, respectively, as \( V, V, \hat{V}, \text{ and } V \) respectively.

For example, the definition of the partial molar enthalpy would be different depending on which set of notation is used. The markup

\[ \begin{align*}
\text{H}_{pm,i} &= \text{Partial}\{H\}\{Nt\_i\}\{T,P,\text{allNbsBut}\{i\}} \\
&= \text{Partial}\{H\}\{Nt\_i\}\{T,P,\text{allNbsBut}\{i\}}
\end{align*} \]

yields the following, depending on the package option loaded:

\[ \begin{align*}
\overline{H}_i &= \left( \frac{\partial H}{\partial n_i} \right)_{T,P,n_{ji}} = \left( \frac{\partial nH}{\partial n_i} \right)_{T,P,n_{ji}} \quad \text{intensive-plain} \\
\overline{H}_i &= \left( \frac{\partial h}{\partial n_i} \right)_{T,P,n_{ji}} = \left( \frac{\partial nH}{\partial n_i} \right)_{T,P,n_{ji}} \quad \text{extensive-plain} \\
\overline{H}_i &= \left( \frac{\partial h}{\partial n_i} \right)_{T,P,n_{ji}} = \left( \frac{\partial nH}{\partial n_i} \right)_{T,P,n_{ji}} \quad \text{extensive-superscript} \\
\overline{H}_i &= \left( \frac{\partial h}{\partial n_i} \right)_{T,P,n_{ji}} = \left( \frac{\partial nH}{\partial n_i} \right)_{T,P,n_{ji}} \quad \text{intensive-lowercase}
\end{align*} \]

The use of intensive-lowercase is strongly discouraged.

Note that the number of moles can be changed from \( n \) to \( N \) via the options in the previous section.

### 3.1.3 Options Affecting Partial Derivatives

There are several options that change how partial derivatives are rendered. First are the options that affect the delimiters. We will use the following code as an example:

\[ \begin{align*}
\{ \text{Partial}\{V\}\{T\}\{P\} \} &= \text{PartialMixSecond}\{G\}\{T\}\{P\}\{\} \\
&= \text{PartialMixSecond}\{G\}\{P\}\{T\}\{\} \\
&= -\text{Partial}\{S\}\{P\}\{T\}. \quad \]

Using the parentheses option (the default), this gives

\[ \left( \frac{\partial V}{\partial T} \right)_p = \left( \frac{\partial^2 G}{\partial T \partial P} \right)_p = \left( \frac{\partial^2 G}{\partial P \partial T} \right)_p = -\left( \frac{\partial S}{\partial P} \right)_T. \]
Table 4. Notation sets that can be set using the options intensive-plain (the default), extensive-plain, extensive-superscript, and intensive-lowercase, respectively. Note that specific quantities are generally assumed to look like the molar symbol with a caret on top.

<table>
<thead>
<tr>
<th>Option</th>
<th>$V_t$</th>
<th>$V_m$</th>
<th>$V_s$</th>
<th>$V_{pm_i}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>intensive-plain</td>
<td>$V$</td>
<td>$V$</td>
<td>$\hat{V}$</td>
<td>$\bar{V}_i$</td>
</tr>
<tr>
<td>extensive-plain</td>
<td>$V$</td>
<td>$\check{V}$</td>
<td>$\bar{V}_i$</td>
<td></td>
</tr>
<tr>
<td>extensive-superscript</td>
<td>$V^s$</td>
<td>$V$</td>
<td>$\hat{V}$</td>
<td>$\bar{V}_i$</td>
</tr>
<tr>
<td>intensive-lowercase</td>
<td>$V$</td>
<td>$v$</td>
<td>$\check{v}$</td>
<td>$\bar{v}_i$</td>
</tr>
</tbody>
</table>

The option **brackets** changes the output to

$$\left[ \frac{\partial V}{\partial T} \right]_P = \left[ \frac{\partial^2 G}{\partial T \partial P} \right] = \left[ \frac{\partial^2 G}{\partial P \partial T} \right] = - \left[ \frac{\partial S}{\partial P} \right]_T.$$

The option **bar** changes the output to

$$\frac{\partial V}{\partial T} \bigg|_P = \frac{\partial^2 G}{\partial T \partial P} = \frac{\partial^2 G}{\partial P \partial T} = - \frac{\partial S}{\partial P} \bigg|_T.$$

The option **plain-derivatives** eliminates all delimiters; this forces the **nosubscripts** option. The output in this case is

$$\frac{\partial V(T, P)}{\partial T} = \frac{\partial^2 G(T, P)}{\partial T \partial P} = \frac{\partial^2 G(P, T)}{\partial P \partial T} = - \frac{\partial S(P, T)}{\partial P}.$$

Accompanying the plain-derivatives option is the **nosubscripts** option, which overrides the default option **subscripts**. This option makes partial derivatives such as

$$\left( \frac{\partial V}{\partial P} \right)_T \quad \text{(subscripts option)},$$

and instead renders them

$$\left( \frac{\partial V(P, T)}{\partial P} \right) \quad \text{(nosubscripts option)}.$$

Combined with plain-derivatives, this would give

$$\frac{\partial V(P, T)}{\partial P} \quad \text{(nosubscripts and plain-derivatives options)}.$$

The order of the variables is determined by the arguments given: it is always written with the variable that is changing first, and the other variables in the order given in the final argument to \(\texttt{\textbackslash Partial} \) and friends. This will result in things like the following:

\[
\texttt{\textbackslash Partial}\{\texttt{\textbackslash Vm}\}\{T\}\{P\} = \texttt{\textbackslash PartialMixSecond}\{\texttt{\textbackslash Gm}\}\{T\}\{P\}\}
= \texttt{\textbackslash PartialMixSecond}\{\texttt{\textbackslash Gm}\}\{P\}\{T\}\} = -\texttt{\textbackslash Partial}\{\texttt{\textbackslash Sm}\}\{P\}\{T\}, \]

14
which would produce (using plain-derivatives)

\[
\frac{\partial V(T,P)}{\partial T} = \frac{\partial^2 G(T,P)}{\partial T \partial P} = \frac{\partial^2 G(P,T)}{\partial P \partial T} = -\frac{\partial S(P,T)}{\partial P}
\]

This is not unclear, but the variables appear in a different order.

thermoparentheses If you want to use parentheses locally, even though your overall document uses another delimiter, the thermoparentheses environment will do that. Similarly, thermobrackets will temporarily switch to brackets, thermobar will temporarily switch to a tailing vertical bar, and thermoplain will remove delimiters altogether.

thermosubscripts The environments thermosubscripts and thermoNOSubscripts force the use or disuse of subscripts, respectively.

3.1.4 Options Regarding the Number of Moles

\[\allNs\] The default option moles-index defines the macro \allNs to expand to \(\vec{n}\) and the macro \allNsbut{i} to expand to \(n_j\). You can change the dummy index: \allNsbut[k]{i} expands to \(n_k\) by default. This is typically not necessary, however: if you type \allNsbut{j}, the package will figure out that you want \(n_k\) rather than \(n_j\). The time to use the optional argument is in situations such as

\[
\left( \frac{\partial \mu_j}{\partial n_k} \right)_{n_{\neq k}},
\]

which is incorrect if the dummy index \(j\) is used in place of the \(i\).

You can change these to expand to ranges using the moles-range option, which renders \allNs as \(n_1, \ldots, n_C\) and \allNsbut{i} as \(n_1, \ldots, [n_i], \ldots, n_C\). The optional argument is ignored in this set of notation. Examples of these options are shown in Table 5.

\[\ncomponents\] You can change the symbol for the number of components (default: \(C\)) by redefining the macro \ncomponents.

3.1.5 Other Options

\[\dbar\] The default for path-dependent one-forms (often called “inexact differentials”) is \dbar, which looks like \(\bar{\partial}\). This can be changed, if desired, to a delta (\(\delta\)) with the delta option to the package.

It should be noted that the \dbar macro is not very stable: changing the typeface to something not supported will probably ruin it, as the kerning is very font-specific. This package currently supports Computer Modern, Times, Palatino, Bitstream Charter, Garamond, and Utopia, but other typefaces may require a manual redefinition.

3.1.6 Options for Specific Textbooks

There are several options that load package options and/or redefine particular commands to match the notation in a particular textbook. So far the following textbooks are supported:
Table 5. Illustration of the moles-index and moles-range options and their effects on \allNs and \allNsbut.

<table>
<thead>
<tr>
<th>Macro\a</th>
<th>moles-index</th>
<th>moles-range</th>
</tr>
</thead>
<tbody>
<tr>
<td>\allNs</td>
<td>$n_i^j,...,n_C^i$</td>
<td>$n_{11}^j,...,n_{i1}^j,...,n_C^i$</td>
</tr>
<tr>
<td>\allNsbut{i}</td>
<td>$n_{i1}^j,...,n_{i2}^j,...,n_C^i$</td>
<td>$n_{i1}^j,...,[n_{i1}]^j,...,n_C^i$</td>
</tr>
<tr>
<td>\allNsbut{j}</td>
<td>$n_{i1}^j,...,[n_{i1}]^j,...,n_C^i$</td>
<td>$n_{i1}^j,...,[n_{i1}]^j,...,n_C^i$</td>
</tr>
<tr>
<td>\allNsbut{\ncomponents}</td>
<td>$n_{i1}^C,...,n_{i2}^C,...,n_{i1}^C,...,n_C^i$</td>
<td>$n_{i1}^C,...,n_{i1}^C,...,n_{i1}^C,...,n_C^i$</td>
</tr>
<tr>
<td>\allNsbut[k]{i}</td>
<td>$n_{i1}^j,...,n_{i1}^j,...,n_C^j,...,n_{i1}^j,...,n_C^j,...,n_C^j$</td>
<td>$n_{i1}^j,...,[n_{i1}]^j,...,n_C^j,...,n_{i1}^j,...,n_C^j,...,n_C^j$</td>
</tr>
<tr>
<td>\allXs</td>
<td>$x_i^j,...,x_C^i$</td>
<td>$x_{11}^j,...,x_{12}^j,...,x_{11}^j,...,x_C^i$</td>
</tr>
<tr>
<td>\allXsbut{\ncomponents-1}</td>
<td>$x_{11}^j,...,x_{12}^j,...,x_{11}^j,...,x_{12}^j,...,x_{11}^j,...,x_{12}^j,...,x_{12}^j,...,x_C^i$</td>
<td>$x_{11}^j,...,x_{11}^j,...,x_{11}^j,...,x_{12}^j,...,x_{12}^j,...,x_C^i$</td>
</tr>
<tr>
<td>\allXsbut[k]{\ncomponents-1}</td>
<td>$x_{11}^j,...,x_{11}^j,...,x_{11}^j,...,x_{11}^j,...,x_{11}^j,...,x_{12}^j,...,x_{12}^j,...,x_{12}^j,...,x_C^i$</td>
<td>$x_{11}^j,...,x_{11}^j,...,x_{11}^j,...,x_{11}^j,...,x_{11}^j,...,x_{12}^j,...,x_{12}^j,...,x_{12}^j,...,x_C^i$</td>
</tr>
</tbody>
</table>

\aYou may use C directly instead of \ncomponents here, but then it will not change C-1 to C-2 (or, say, M-1 to M-2) if you want to redefine \ncomponents later.

\bThis would typically be used to denote something like $G(T,P,n,x_1,...,x_{C-1})$ rather than in a subscript, but it looks silly if we don’t handle this case this way.

Bejan Notation used by Bejan, *Advanced Engineering Thermodynamics*, Third Edition. Wiley: Hoboken, 2006. Loads non-default package options EUFGHAN, intensive-lowercase, and delta. Also swaps the definitions of \@specific and \@intensive and redefines \cV, \cVs, \cVt, \expansivitysymbol, and \ncomponents to match his notation.

CBK Notation used by Çengel, Boles, and Kanoğlu, *Thermodynamics: An Engineering Approach*, Ninth Edition. McGraw Hill: Singapore, 2020. Loads the non-default package options EUAGHAN and intensive-lowercase; also redefines partial molar, specific, and molar properties’ notation to fit theirs, and redefines \pressure@symbol, \Deltarxn, \compressibilitysymbol, and \expansivitysymbol to fit their usage. Their prodigious symbols for specific and total volume are not supported.


Koretsky Notation used by Koretsky, *Engineering and Chemical Thermodynamics*, Second Edition, Wiley: New Caledonia, 2013. Loads the non-default package options EUAGHAN, brackets, intensive-lowercase, and delta; modifies the intensive-lowercase defaults to make uppercase partial molar properties to match Koretsky’s notation and redefines \expansivitysymbol, \IS, \residual, \IG, \IGM, \Deltafus, \Deltasub, \Deltavap, \Henryrat, \gammarat, \Deltafus, \Deltavap, and \Deltasub to match his use. Also redefines the fugacity coefficients to use $\varphi$ instead of $\phi$. 
Notation used by Moran, Shapiro, Boettner, and Bailey, *Fundamentals of Engineering Thermodynamics*, Eighth Edition. Wiley: Kendallville, 2014. Loads the non-default package options EUFGHAn, intensive-lowercase, and delta; removes the left parenthesis in partial derivatives and redefines \IGM, \IG, \expansivitysymbol, \allcomponents, \allbut, \Ft, \Helmholtz@symbol, \@intensive, \@specific, \fmix, \phimix, and \pressure@symbol to fit their somewhat ill-advised notation. Also resets \partialmolar to match their use.

Notation used by Prausnitz, Lichtenthaler, and de Azevedo, *Molecular Thermodynamics of Fluid-Phase Equilibria*, Third Edition, Pearson, 1998. Loads the non-default package option intensive-lowercase and redefines \fmix, \phimix, \fpure, \phipure, \residual, \allcomponents, and \allbut to fit their notation.

Notation used by Sandler, *Chemical, Biochemical, and Engineering Thermodynamics*, Fifth Edition. Wiley: Hoboken, 2017. Loads non-default package options EUAGHaN and extensive-plain; also redefines \sat, \excess, \residual, \ncomponents, \fmix, \fstd, \phimix, \allcomponents, \IS, \Deltamix, \Deltarxn, \Deltasub, \Deltamus, \Deltavap, and \Henryrat. Also redefines \cV, \cP, \cVt, and \cPt.

Notation used by Smith, Van Ness, Abbott, and Swihart, *Introduction to Chemical Engineering Thermodynamics*, Ninth Edition. McGraw-Hill: Boston, 2021. Loads the non-default package option extensive-superscript and redefines \allcomponents, \allbut, \IG, \IGM, \IS, \expansivitysymbol, and \Deltarxn to fit their notation. The extensive heat capacities are also redefined, as they do not use such an entity.

Notation used by Tester and Modell, *Thermodynamics and Its Applications*, Third Edition, Prentice Hall: Upper Saddle River, 1997. Loads the non-default package options EUAGHAn and delta; also redefines \ncomponents to be \( n \) and redefines \allcomponents, \allbut, and \allbutlastand to fit their (somewhat inconsistent) notation. Also redefines \IG, \IGM, \IS, \excess, \reaction, \Henryrat, \Henrymol, \gammarat, and \gammamol to fit their style.

Notation used by Thompson, *A Unified Introduction to Chemical Engineering Thermodynamics*, Stillwater Press: Orono, 2000. Loads the non-default package options EUAGHAn and delta; also redefines \excess, \residual, \allcomponents, \ncomponents, \IS, \IG, \IGM, \fpure, \phipure, \mix, \Deltamix, \Deltarxn, \@intensive, and \allbut to match his notation. Note that he uses \( c, n_C \), and \( n \) for the number of components in various places in the book; I chose \( c \) for the definition of \ncomponents, but it is impossible to be completely consistent with his notation.

There may well be some inconsistencies between the notation in these books and the symbols used here. I will fix such inconsistencies as I become aware of them.
4 Implementation

This package requires the \texttt{amstext} package, as \texttt{text} is used to handle \texttt{sat}, \texttt{IS}, \texttt{IG}, \texttt{IGM}, \texttt{Henrymol}, \texttt{fusion}, \texttt{reaction}, \texttt{vaporization}, and \texttt{sublimation} by default, as well as several other macros defined by package options.

\begin{verbatim}
\RequirePackage{amstext}
\end{verbatim}

4.1 Symbols Controlled by Package Options

We set some symbols prior to declaring the package options. The default symbols follow package option \texttt{EUAGHan}, even though the macros follow the option \texttt{EUFGHAN}.

\begin{verbatim}
\dbar \text\ The way \texttt{dbar} is defined depends on the typeface you are using. We try to determine, at \texttt{\begin{document}}, which typeface you chose based on the packages that are loaded and some of their internal definitions. The \texttt{thermodynamics} package currently supports Computer Modern (the default or through \texttt{1modern}), Palatino (through \texttt{pxfonts} or \texttt{newpxmath}), Times (through \texttt{txfonts}, \texttt{mathptmx}, or \texttt{newtxmath}), Utopia (through \texttt{mathdesign}), Bitstream Charter (through \texttt{mathdesign}), and Garamond (through \texttt{mathdesign}). Definitions of \texttt{dbar} (with \texttt{newcommand*} or \texttt{providecommand*}) in the preamble will override the ones here.
\end{verbatim}

\begin{verbatim}
\AtBeginDocument{
\@ifpackageloaded{pxfonts}{% 
  \providecommand*{\dbar}{\{\mkern5mu\mathchar'26\mkern-10mu d\}}% 
}\@ifpackageloaded{newpxmath}{% 
  \providecommand*{\dbar}{\{\mkern5mu\mathchar'26\mkern-10mu d\}}% 
}\@ifpackageloaded{txfonts}{% 
  \providecommand*{\dbar}{\{\mkern5mu\mathchar'26\mkern-11mu d\}}% 
}\@ifpackageloaded{mathptmx}{% 
  \providecommand*{\dbar}{\{\mkern5mu\mathchar'26\mkern-10mu d\}}% 
}\@ifpackageloaded{newtxmath}{% 
  \providecommand*{\dbar}{\{\mkern5mu\mathchar'26\mkern-12mu d\}}% 
}\@ifpackageloaded{mathdesign}{% 
  \def@charter{mdbch}% 
  \def@utopia{mdput}% 
  \def@garamond{mdugm}% 
  \ifx\MD@default@family@utopia 
    \providecommand*{\dbar}{\{\mkern8mu\mathchar'26\mkern-20mu d\}}% 
  \fi 
  \ifx\MD@default@family@charter 
    \providecommand*{\dbar}{\{\mkern5mu\mathchar'26\mkern-15mu d\}}% 
  \fi 
  \ifx\MD@default@family@garamond 
    \providecommand*{\dbar}{\{\mkern5mu\mathchar'26\mkern-17mu d\}}% 
  \fi 
}\@ifpackageloaded{mathdesign}{% 
  \def@charter{mdbch}% 
  \def@utopia{mdput}% 
  \def@garamond{mdugm}% 
  \ifx\MD@default@family@utopia 
    \providecommand*{\dbar}{\{\mkern8mu\mathchar'26\mkern-20mu d\}}% 
  \fi 
  \ifx\MD@default@family@charter 
    \providecommand*{\dbar}{\{\mkern5mu\mathchar'26\mkern-15mu d\}}% 
  \fi 
  \ifx\MD@default@family@garamond 
    \providecommand*{\dbar}{\{\mkern5mu\mathchar'26\mkern-17mu d\}}% 
  \fi 
}\@ifpackageloaded{mathdesign}{% 
  % Defaults to Computer Modern 
  \providecommand*{\dbar}{\{\mkern3mu\mathchar'26\mkern-12mu d\}}
}\end{verbatim}
Symbols are defined for the total energy, internal energy, Helmholtz free energy, Gibbs free energy, enthalpy, entropy, surface area, volume, number of moles, heat, and work; these are $E$, $U$, $A$, $G$, $H$, $S$, $a$, $V$, $n$, $Q$, and $W$, respectively. These commands should not be used on their own, but rather accessed through the macros \totalenergy, \internalenergy, and \Helmholtz (using the internal energy as an example).

The default symbols are not intended to be easy to change—the intended mechanism is through package options. If you want to use a non-standard symbol that is not available through one of the package options, you can redefine these macros inside \makeatletter...\makeatother. For example,

\begin{verbatim}
\makeatletter
\renewcommand*{\Helmholtz@symbol}{H}
\renewcommand*{\enthalpy@symbol}{h}
\makeatother
\end{verbatim}

would define the ill-advised notation that I have nonetheless heard of that uses $H$ for Helmholtz free energy and $h$ for enthalpy. Note that the macros for temperature and pressure are only used inside the definitions of the compressibilities, expansivitites, and heat capacities; there is no user-level macro for the temperature or the pressure, so it is up to the user to use consistent symbols for those properties.

We then define two macros and several lengths that we shall use when drawing rules above or below a symbol. The default is to use underlined symbols for extensive quantities, plain symbols for molar quantities, and carets for specific quantities, but this can be changed using package options.

\begin{verbatim}
\newcommand*{\thermo@underline}[1]{\mkern1mu\underline{\mkern-1mu #1\mkern-4mu}\mkern4mu}
\newcommand*{\thermo@overline}[1]{\mkern2mu\overline{\mkern-2mu #1\mkern-1mu}\mkern1mu}
\end{verbatim}

We define three commands to use to denote the beginning and end of partial derivatives. These symbols can be customized by package options. Default is parentheses,
meaning that \[ \frac{\partial f}{\partial x} \] renders as
\[
\frac{\partial f}{\partial x} y
\]
with the defaults. The macro \texttt{\PartialEmptyClose} is used when the last argument to \texttt{\Partial} is empty, which is important for the bar option to the \texttt{documentclass} or inside the \texttt{thermobar} environment.

\begin{verbatim}
55 \newcommand*{\PartialOpen}{(}
56 \newcommand*{\PartialClose}{)}
57 \newcommand*{\PartialEmptyClose}{)}
\end{verbatim}

### 4.2 Package Options

We declare a bunch of options for which sets of symbols to use. These are summarized in Table 3.

\begin{verbatim}
58 \DeclareOption{EUAGHan}{}% the default
59 \DeclareOption{EUAGHaN}{\renewcommand*{\mole@symbol}{N}}%
60 \DeclareOption{EUHAGan}{\ExecuteOptions{EUAGHan}}
61 \DeclareOption{EUHAGaN}{\ExecuteOptions{EUAGHaN}}
62 \DeclareOption{EUFGHAn}{% this is my favorite set of symbols
63 \renewcommand*{\Helmholtz@symbol}{F}%
64 \renewcommand*{\area@symbol}{A}%
65 }
66 \DeclareOption{EUFGHAN}{% this is my second favorite set of symbols
67 \renewcommand*{\Helmholtz@symbol}{F}%
68 \renewcommand*{\area@symbol}{A}%
69 \renewcommand*{\mole@symbol}{N}%
70 }
71 \DeclareOption{EEFGHAn}{% this treats all energy the same way(?)
72 \renewcommand*{\internalenergy@symbol}{E}%
73 \renewcommand*{\Helmholtz@symbol}{F}%
74 \renewcommand*{\area@symbol}{A}%
75 }
76 \DeclareOption{EEFGHAN}{% this treats all energy the same way(?)
77 \renewcommand*{\internalenergy@symbol}{E}%
78 \renewcommand*{\Helmholtz@symbol}{F}%
79 \renewcommand*{\area@symbol}{A}%
80 \renewcommand*{\mole@symbol}{N}%
81 }
82 \DeclareOption{EEAGHan}{%
83 \renewcommand*{\internalenergy@symbol}{E}%
84 \renewcommand*{\Helmholtz@symbol}{F}
85 }
86 \DeclareOption{EEAGHaN}{%
87 \renewcommand*{\internalenergy@symbol}{E}%
88 \renewcommand*{\Helmholtz@symbol}{F}
89 \renewcommand*{\mole@symbol}{N}%
90 }
91 \DeclareOption{EEAGHan}{%
92 \renewcommand*{\internalenergy@symbol}{E}%
93 }
94 \DeclareOption{EEAGHaN}{%
\end{verbatim}
The delta option redefines \(\dbar\) to produce the symbol \(\delta\). The default is to use a \(d\) with a slash through it (\(\bar{d}\)) for inexact differentials unless the user overrides it with this option. The macro could also be redefined manually, of course.

Next, we define options for the set of notation. The default is intensive-plain, which produces things like \(V\) for molar volume, \(\bar{V}\) for total volume, and \(\hat{V}\) for specific volume. These are defined via the internal macros \@extensive, \@intensive, and \@specific.

\begin{verbatim}
\let\@extensive\relax
\let\@intensive\relax
\let\@specific\hat
\DeclareOption{extensive-plain}{\let\@extensive\relax\let\@intensive\thermo@underline}
\DeclareOption{intensive-plain}{\let\@extensive\thermo@underline\let\@intensive\relax}
\DeclareOption{intensive-lowercase}{\let\@extensive\MakeUppercase\let\@intensive\MakeLowercase\def\@specific#1{\MakeLowercase\hat #1}}
\AtEndOfPackage{\renewcommand*{\URt}{\internalenergy@symbol^{\residual,t}}}
\end{verbatim}
The next two options choose whether variables held constant are subscripted (the default) or placed next to the function. The \texttt{nosubscripts} option currently requires that the variable being changed is the first one in the argument list. The difference is

\[
\left( \frac{\partial U}{\partial S} \right)_V \text{ versus } \left( \frac{\partial U(S,V)}{\partial S} \right)
\]

for subscripts and \texttt{nosubscripts}, respectively.

These options change how \texttt{\Partial} and friends render derivatives. The default is parentheses, but other options include brackets, braces, a vertical bar on the right side, or plain (undecorated) derivatives.
4.3 The Number of Moles Macros

\ncomponents

We define the number of components, default $C$, for use in the “all moles” and related macros.

178 \newcommand*{\ncomponents}{C}

\allNs

Several macros define a shorthand for “moles of all species” (\allNs) and “moles of all species except” (\allNsbut), as well as similar quantities for masses (\allMs, \allMsbut) and chemical potentials (\allmus, \allmusbut), which occur frequently in mixture thermodynamics. The default is for \allNs to become $\vec{n}$ and \allNsbut{i} to become $n_{j\neq i}$. The optional argument changes which index (default: $j$) to use in the left side of the inequality.\footnote{The index $j$ is automatically replaced with $k$ if the user issues \allNsbut{j}.} Essentially identical commands are defined for chemical potentials and masses: \allmus and \allmusbut and \allMs and \allMsbut, respectively.

179 \newcommand*{\allNs}{\allcomponents{\Nt}}

180 \newcommand*{\allXs}{\allcomponents{x}}

181 \newcommand*{\allYs}{\allcomponents{y}}

182 \newcommand*{\allmus}{\allcomponents{\mu}}

183 \newcommand*{\allMs}{\allcomponents{m}}

184 \newcommand*{\allWs}{\allcomponents{w}}

\allXs
\allYs
\allWs
\allWsbut
\allXsbut
\allYsbut
\allmus
\allMs
\allmusbut
\allMsbut

Similar commands are defined for mole fractions (\allXs, \allYs, etc.), but these assume the last mole fraction is not one of the variables—that is, \allXsbut and \allYsbut assume the argument and \ncomponents are held constant. For example,

\allXsbut
\allYsbut
\allXsbut\[2\]\[j\]{\allbutlastand\[#1\]{#2}{x}}

\allYsbut\[2\]\[j\]{\allbut\[#1\]{#2}{y}}

\allmusbut\[2\]\[j\]{\allbut\[#1\]{#2}{\mu}}

\allMsbut\[2\]\[j\]{\allbut\[#1\]{#2}{m}}

\allWsbut\[2\]\[j\]{\allbutlastand\[#1\]{#2}{w}}

\allbutlastand
\allbut
\allcomponents

\allbut
\allcomponents

The \allcomponents, \allbut, and \allbutlastand macros can be used to define new entities; say, if you want to use $z_i$ as a mole fraction, then use

\newcommand*{\allZsbut}{\allcomponents{z}}

Similarly, something meaning the concentrations of every species could be defined via

\newcommand*{\allCs}{\allcomponents{C}}

191 \newcommand*{\allcomponents}{\vec{#1}}

192 \newcommand*{\allbut}{\vec{#1}}

193 \def\tmp@arg{#2}\

194 \def\tmp@arg{#1}%
We then define two package options that change how to render \allNs and friends.

\DeclareOption{moles-index}{
\DeclareOption{moles-range}{
\renewcommand*{\allcomponents}[1]{{#1}_1,\ldots,{#1}_{\ncomponents}}
\renewcommand*{\allbut}[3][j]{{#3}_{#1\neq #2}}
\renewcommand*{\allbutlastand}[3][j]{{#3}_{#1\neq #2}}
}\edef\tmp@arg{#2}\edef\tmp@@arg{#1\neq #2}
\ifx\tmp@@arg\components{#3}_{#1\neq #2}\else{#3}_{#1\neq #2,\ncomponents}\fi
\fi
\fi
}\edef\tmp@arg{#1}\edef\tmp@@arg{#2}\edef\@one{1}
\ifx\tmp@arg\@one{#3}_{#2},\ldots,{#3}_{\ncomponents}\else{#3}_{1,\ldots,\{#3}_{#2},\ldots,{#3}_{\ncomponents}}\fi
\fi
The remaining options define textbook-specific notation.
\DeclareOption{Bejan}{
\ExecuteOptions{EUFGHAN,intensive-lowercase,delta}
\let@specific\MakeLowercase
\def@intensive#1{\MakeLowercase{\bar{#1}}}
\AtEndOfPackage{
\DeclareSubscrSymbol{cV}{\bar\heatcapacitysymbol}{v}
\DeclareSubscrSymbol{cVs}{\heatcapacitysymbol}{v}
\DeclareSubscrSymbol{cVt}{\Nt\heatcapacitysymbol}{v}
\renewcommand*{\expansivitysymbol}{\beta}
\renewcommand*{\ncomponents}{n}
}
}\DeclareOption{CBK}{
\ExecuteOptions{EUAGHAN,intensive-lowercase}
\AtEndOfPackage{
\let@specific\MakeLowercase
\renewcommand*{\@intensive}{\MakeLowercase{\bar{#1}}}
\renewcommand*{\pressure@symbol}{p}
\renewcommand*{\Deltarxn}{#1_R}
\renewcommand*{\compressibilitysymbol}{\beta}
\renewcommand*{\expansivitysymbol}{\alpha}
}
}\DeclareOption{ElliottLira}{
\AtEndOfPackage{
\renewcommand*{\allcomponents}{#1}
\renewcommand*{\Deltarxn}{\Delta #1}
}
}\DeclareOption{Koretsky}{
\ExecuteOptions{EUAGHAn,brackets,intensive-lowercase,delta}
\AtEndOfPackage{
\renewcommand*{\partialmolar}{#1}{\gdef\pm@symbol{#1}\generic@pm}
\renewcommand*{\expansivitysymbol}{\beta}
\renewcommand*{\IS}{\text{ideal}}
\renewcommand*{\residual}{\text{dep}}
\renewcommand*{\IG}{\text{ideal}}
\renewcommand*{\IGM}{\text{ideal}}
\renewcommand*{\Deltafus}{\Delta #1_{\text{fusion}}}
\renewcommand*{\Deltavap}{\Delta #1_{\text{vaporization}}}
\renewcommand*{\Henryrat}{\gamma^{\text{Henry's}}}
\let\Delta@fus@sym\relax
\DeclareSubscrSymbol{@Deltafus}{\Delta@fus@sym}{\fusion}
\renewcommand*{\Deltafus}{\def\Delta@fus@sym{\Delta #1}\@Deltafus}

\let\Delta@vap@sym\relax
\DeclareSubscrSymbol{@Deltavap}{\Delta@vap@sym}{\vaporization}
\renewcommand*{\Deltasub}[1]{\def\Delta@sub@sym{\Delta #1}\@Deltasub}
\let\Delta@sub@sym\relax
\DeclareSubscrSymbol{@Deltasub}{\Delta@sub@sym}{\sublimation}
\renewcommand*{\Deltasub}[1]{\def\Delta@sub@sym{\Delta #1}\@Deltasub}
\DeclareOption{MSBB}{
\ExecuteOptions{EUFGHAn,intensive-lowercase,delta}
\AtEndOfPackage{
\renewcommand*{\IGM}{\ast}
\renewcommand*{\IG}{\ast}
\renewcommand*{\expansivitysymbol}{\beta}
\renewcommand*{\allcomponents}{#1}
\renewcommand*{\allbut}[3][j]{%\def\tmp@arg{#2}\def\tmp@@arg{#1}\ifx\tmp@arg\tmp@@arg
{#3}_{#1}\else
{#3}_{#1}\fi}
\Helmholtz@symbol\psi
\Ft{\Psi}
\def@intensive#1{\MakeLowercase{\thermo@overline #1}}
\let\@specific\MakeLowercase
\renewcommand*{\partialmolar}[1]{\gdef\pm@symbol{#1}\generic@pm}
\fmix{\bar f}
\phimix{\bar\phi}
\pressure@symbol{p}
}}
\DeclareOption{Prausnitz}{
\ExecuteOptions{intensive-lowercase}
\AtEndOfPackage{
\fmix{f}
\phimix{\phi}
\fsat{f^\text{pure}^\text{sat}}
\DeclareSubscrSymbol{fpure}{f}{\text{pure}}
\DeclareSubscrSymbol{phipure}{\phi}{\text{pure}}
\residual{\mathcal{R}}
\allcomponents{{#1}_i}
\allbut{{#3}[i]}{#1}\def\tmp@arg{#2}\def\tmp@@arg{#1}\ifx\tmp@arg\tmp@@arg
{#3}_{#1}\else
{#3}_{#1}\fi}
}}
\DeclareOption{Sandler}{
  \ExecuteOptions{EUAGHaN,extensive-plain}
  \AtEndOfPackage{
    \renewcommand*{\sat}{\text{vap}}
    \renewcommand*{\excess}{\text{ex}}
    \renewcommand*{\residual}{\text{r}}
    \renewcommand*{\ncomponents}{\mathcal{C}}
    \renewcommand*{\fmix}{\bar f}
    \renewcommand*{\fstd}{\bar f^\text{circ}}
    \renewcommand*{\phimix}{\bar \phi}
    \renewcommand*{\allcomponents}{\text{thermo@underline{#1}}}
    \renewcommand*{\IG}{\text{IG}}
    \renewcommand*{\IS}{\text{IM}}
    \renewcommand*{\Deltamix}{\Delta^\text{mixing} #1}
    \renewcommand*{\Deltarxn}{\Delta^\text{reaction} #1}
    \renewcommand*{\Deltasub}{\Delta^\text{sublimation} #1}
    \renewcommand*{\Deltafus}{\Delta^\text{fusion} #1}
    \renewcommand*{\Deltavap}{\Delta^\text{vaporization} #1}
    \renewcommand*{\Henryrat}{H}
    \DeclareSubscrSymbol{cV}{\heatcapacitysymbol}{\volume@symbol}
    \DeclareSubscrSymbol{cP}{\heatcapacitysymbol}{\pressure@symbol}
    \DeclareSubscrSymbol{cVt}{\Nt\heatcapacitysymbol}{\volume@symbol}
    \DeclareSubscrSymbol{cPt}{\Nt\heatcapacitysymbol}{\pressure@symbol}
  }
}\DeclareOption{SVNAS}{
  \ExecuteOptions{extensive-superscript}
  \AtEndOfPackage{
    \renewcommand*{\allcomponents}[1]{#1}
    \renewcommand*{\allbut}[3][j]{
      \def\tmp@arg{#2}
      \def\tmp@@arg{#1}
      \ifx\tmp@arg\tmp@@arg
        {#3}_{k}
      \else
        {#3}_{#1}
      \fi
    }
    \renewcommand*{\IG}{{ig}}
    \renewcommand*{\IGM}{{ig}}
    \renewcommand*{\IS}{{id}}
    \renewcommand*{\expansivitysymbol}{\beta}
    \DeclareSubscrSymbol{cVt}{\Nt\heatcapacitysymbol}{\volume@symbol}
    \DeclareSubscrSymbol{cPt}{\Nt\heatcapacitysymbol}{\pressure@symbol}
}
}\DeclareOption{TesterModell}{
  \ExecuteOptions{EUAGHaN,delta}
  \AtEndOfPackage{
    \renewcommand*{\ncomponents}{n}
    \renewcommand*{\allcomponents}[1][1]{#1_i}
  }
}
\renewcommand*{\allbut}[3][j]{{% 
  \def\tmp@arg{#2}% 
  \def\tmp@@arg{#1}% 
  \ifx\tmp@arg\tmp@@arg 
    \{#3\}_{#1}\relax[#2] 
  \else 
    \{#3\}_{#1}\relax[#2] 
  \fi 
} 
\renewcommand*{\allbutlastand}[3][j]{{% 
  \edef\tmp@arg{#1}% 
  \edef\tmp@@arg{#2}% 
  \ifx\tmp@@arg\components 
    \{#3\}_{#1}\relax[#2] \% 
  \else 
    \ifx\tmp@arg\tmp@@arg 
      \{#3\}elax[#2,\components] \% 
    \else 
      \{#3\}elax[#2,\components] \% 
    \fi 
  \fi 
} 
\renewcommand*{\IG}{{ig}}% Tester & Modell never use "igm" anywhere, and I only found ig once 
\renewcommand*{\IGM}{{igm}} 
\renewcommand*{\IS}{{ID}}% 
\renewcommand*{\excess}{{EX}}% 
\renewcommand*{\reaction}{{rx}}% 
\renewcommand*{\Henryrat}{f^\ast\ast} 
\renewcommand*{\Henrymol}{f^\ast} 
\renewcommand*{\gammarat}{\gamma^\ast\ast} 
\renewcommand*{\gammamol}{\gamma^\ast} 
} 
\DeclareOption{Thompson}{{% 
  \ExecuteOptions{EUAGHAu,delta} 
  \AtEndOfPackage{% 
    \renewcommand*{\excess}{{EX}} 
    \renewcommand*{\residual}{{R}} 
    \renewcommand*{\allcomponents}[1]{{#1}_j} 
    \renewcommand*{\ncomponents}{c} 
    \renewcommand*{\IS}{{IS}} 
    \renewcommand*{\IG}{{IG}} 
    \let\IGM\IG 
    \def\@fpure_#1{f_{#1}\@ifnextchar^{}{^\bullet}} 
    \renewcommand*{\fpure}{\@ifnextchar_{\@fpure}{f}} 
    \def\@phi@pure_#1{\phi_{#1}\@ifnextchar^{}{^\bullet}} 
    \renewcommand*{\phipure}{\@ifnextchar_{\@phi@pure}{\phi}} 
    \renewcommand*{\mixing}{{MIX}} 
    \renewcommand*{\Deltamix}[1]{\Delta_{\mixing #1}} 
    \renewcommand*{\Deltarxn}[1]{\Delta #1} 
    \def\@@intensive_#1{_{#1}\@ifnextchar^{}{^\bullet}} 
    \newcommand*{\@intensive}[1]{#1\@ifnextchar_{\@@intensive}{} } 
    \renewcommand*{\allbut}[3][j]{{%
We execute the default options below.

\ExecuteOptions{EUAGHan,subscripts,parentheses,intensive-plain,moles-index}
\ProcessOptions

4.4 Commands for Partial Derivatives

The \Partial command and its second-order siblings are defined as below. They typeset partial derivatives of the first argument with respect to the second (and third, in the case of mixed second partial derivatives) arguments, holding the last argument constant.

The starred forms adjust the spacing after the partial derivative so the trailing binary operator (assumed to be the same width as an equals sign) overhangs the variables held constant. We thus set operator@width to be just greater than the width of an equals sign.

\newlength{\Partial@const@width}
\newlength{\operator@width}
\settowidth{\operator@width}{=}
\newlength{\adjust@width}
\setlength{\adjust@width}{0.1\operator@width}
\addtolength{\operator@width}{\adjust@width}

\Partial* The command \Partial and its friends drastically simplify the creation of partial derivatives. The command \Partial* is the same as \Partial except that it adjusts the spacing so the (presumably) binary operator that follows it slightly overlaps the subscripts.

\def\Partial@start{\left\PartialOpen}
\def\Partial@end{\right\PartialClose}
\def\Partial@empty@end{\right\PartialEmptyClose}
\def\Partial{\@ifstar\Partial@star\Partial@nostar}
%^^A Several ifs to use later (breaks it if these definitions are inside)
\newif\iftwo@has@Nt
\two@has@Ntfalse
\newif\iftwo@has@xory
\two@has@xoryfalsetrue
\newif\ifthree@has@Nt
\three@has@Ntfalse
\newif\ifthree@has@xory
\three@has@xoryfalse
\AtEndOfPackage{%
\edef\thermo@Nt{\Nt}%
\edef\thermo@x{x}%
\edef\thermo@y{y}\
\edef\thermo@parse@two#1\relax\
\@for\@i:=#1\do{\edef\@@i{\@i}\
\ifx\@@i\thermo@Nt\two@has@Nttrue\@break@tfor\fi\
\ifx\@@i\thermo@x\two@has@xorytrue\@break@tfor\fi\
\ifx\@@i\thermo@y\two@has@xorytrue\@break@tfor\fi\}
\newcommand*{\Partial@nostar}[3]{{%\if@subscripted
% Handles situation of empty variables held constant
\ifx\tmp@arg\@empty
\ensuremath{\Partial@start\frac{\partial #1}{\partial #2}\Partial@empty@end}\else
\ensuremath{\Partial@start\frac{\partial #1}{\partial #2}\Partial@end_{#3}}\fi
\else
% If not subscripted, we put the arguments (if any) in the derivative
\ifx\tmp@arg\@empty
\ensuremath{\Partial@start\frac{\partial #1}{\partial #2}\Partial@empty@end}\else
\thermo@parse@two#2\relax\iftwo@has@Nt
\renewcommand*{\allbut}[3][j]{\allcomponents{##3}}\%\ensuremath{\Partial@start\frac{\partial #1(#3)}{\partial #2}\Partial@end}\else
\iftwo@has@xory
\renewcommand*{\allbutlastand}[3][j]{\allcomponents}{##3}}\%\ensuremath{\Partial@start\frac{\partial #1(#3)}{\partial #2}\Partial@end}\else
\ensuremath{\Partial@start\frac{\partial #1(#2,#3)}{\partial #2}\Partial@end}\fi\fi\fi\fi\fi\fi
The \PartialBigg macro (and its starred form) replace the \texttt{\left} and \texttt{\right} commands in \Partial with amsmath's \Biggl and \Biggr variants. The starred form is inherited from \Partial without modification.

\def\PartialBigg{%
  \def\Partial@start{\Biggl\PartialOpen}%
  \def\Partial@end{\Biggr\PartialClose}%
  \def\Partial@empty@end{\Biggr\PartialClose}%
  \Partial%
}%

The \Partialbigg macro does the same thing as \PartialBigg, except using amsmath's \biggl/\biggr variants.

\def\Partialbigg{%
  \def\Partial@start{\biggl\PartialOpen}%
  \def\Partial@end{\biggr\PartialClose}%
  \def\Partial@empty@end{\biggr\PartialClose}%
  \Partial%
}%

The second partial derivatives are defined similarly to \Partial.

\def\PartialSecond{%
  \ifstar\PartialSecond@star\PartialSecond@nostar\fi
  \newcommand*{\PartialSecond@nostar}[3]{%
    \def\tmp@arg{#3}%
    \if@subscripted
      % Handles situation of empty variables held constant
      \ifx\tmp@arg\@empty
        \ensuremath{\Partial@start\frac{\partial^2 #1}{\partial #2^2}\Partial@empty@end}%
      \else
        \ensuremath{\Partial@start\frac{\partial^2 #1}{\partial #2^2}\Partial@end_{#3}}%
      \fi
    \else
      \thermo@parse@two#2\relax
    \fi
  }%
\iftwo@has@Nt
  \renewcommand*{\allbut}[3][j]{\allcomponents{##3}}%
  \ensuremath{\Partial@start\frac{\partial^2 #1(#3)}{\partial #2^2}\Partial@end}\
\else
  \iftwo@has@xory
    \renewcommand*{\allbutlastand}[3][j]{\allbut{\ncomponents}{##3}}%
    \ensuremath{\Partial@start\frac{\partial^2 #1(#3)}{\partial #2^2}\Partial@end}\
  \else
    \ensuremath{\Partial@start\frac{\partial^2 #1(#2,#3)}{\partial {#2}^2}\Partial@end}\
  \fi
\fi
\fi
\fi
\newcommand*{\PartialSecond@star}[3]{
  \settowidth{\Partial@const@width}{\ensuremath{#3}}%
  \addtolength{\Partial@const@width}{-0.15\Partial@const@width}%
  \PartialSecond@nostar{#1}{#2}{#3}\
  \if@subscripted
    \ifdim\operator@width<\Partial@const@width
      \kern -\operator@width
    \else
      \kern -\Partial@const@width
    \fi
  \fi
  \fi
}\PartialSecondBigg
\PartialSecondBigg*
The \PartialSecondBigg macro and its starred variant replace \left and \right with amsmath’s \Biggl and \Biggr.
\def\PartialSecondBigg{%
  \def\Partial@start{\expandafter\Biggl\PartialOpen}%
  \def\Partial@end{\expandafter\Biggr\PartialClose}%
  \PartialSecond%
}\PartialSecondbigg
\PartialSecondbigg*
The \PartialSecondbigg macro and its starred variant replace \left and \right with amsmath’s \biggl and \biggr.
\def\PartialSecondbigg{%
  \def\Partial@start{\expandafter\biggl\PartialOpen}%
  \def\Partial@end{\expandafter\biggr\PartialClose}%
  \PartialSecond%
}\PartialMixSecond
\PartialMixSecond*
The macro \PartialMixSecond takes an extra argument, but is otherwise the same as its same-variable cousin.
\def\PartialMixSecond{\@ifstar\PartialMixSecond@star\PartialMixSecond@nostar}\
\newcommand*{\PartialMixSecond@nostar}[4]{
  \def\tmp@arg{#4}\
  \if@subscripted
    % Handles situation of empty variables held constant
  \fi
}
The macro \PartialMixSecondBigg and its starred form are analogous to \PartialBigg and \PartialSecondBigg. \PartialMixSecondbigg and its starred form are similarly analogous to \Partialbigg and \PartialSecondbigg.

If the user does not load the amsmath package, we will not have access to \Biggl, \Biggr, \biggl, and \biggr, so we revert them back to the ordinary \left and \right versions and warn the user.

4.5 Local Override of Delimiters

We define seven environments that locally override the delimiters on partial derivatives generated with \Partial and friends and/or the subscript notation for partial derivatives.

```latex
\newenvironment*{thermoparentheses}{\renewcommand*{\PartialOpen}{(}}{\renewcommand*{\PartialClose}{)}}
\newenvironment*{thermobrackets}{\renewcommand*{\PartialOpen}{[}}{\renewcommand*{\PartialClose}{]}}
\newenvironment*{thermobraces}{\renewcommand*{\PartialOpen}{\{}%\renewcommand{\PartialClose}{\}}}{}
\newenvironment*{thermobar}{\renewcommand*{\PartialOpen}{\bar}}{\renewcommand*{\PartialClose}{}}
\newenvironment*{thermoplain}{\renewcommand*{\PartialOpen}{\null}}{\renewcommand*{\PartialClose}{\null}}
```

34
4.6 User-Interface Macros to Define Symbols

First, we define a (messy!) command that serves to create “subscripted” symbols; for example, typing $c_P_i$ should yield $C_P_i$ rather than $C_{Pi}$, $C_{PI}$, or $C_{P}$. Superscripts are also handled properly and can be in either order.

\newcommand*[\DeclareSubscrSymbol][3]{\expandafter\def\csname #1\endcsname{\@ifnextchar^\csname @#1\endcsname{\@ifnextchar_{\csname @@#1\endcsname{#2}_{#3}}}}}
\DeclareSubscrSymbol{cV}{\@intensive\heatcapacitysymbol}{\volume@symbol}
\DeclareSubscrSymbol{cP}{\@intensive\heatcapacitysymbol}{\pressure@symbol}

We also introduce extensive (rather than molar) heat capacity macros.
\cPs and \cVs are the specific heat capacities.
\kappaT The isothermal and adiabatic compressibilities are defined similarly, but those do not have extensive versions for obvious reasons.
\alphaP The macro \\alphaP is intended to refer to the isobaric volume expansivity, while \alphaS is the isentropic volume expansivity.
\Psat The \Psat macro (and its clone, the \Pvap macro) should be used for the saturation pressure.
\phisat Similarly, a \phisat macro typesets the fugacity coefficient at saturation. The \fsat macro similarly renders the fugacity at saturation. Other saturation properties should use \M^{\sat} or similar, preferably by defining another macro such as \Gmsat.

\Psat std The \Psat macro denotes standard properties. \Pstd and \fstd are defined for convenience and for consistency across textbooks.

\DeltaGmix Mixing properties, such as $\Delta G_{\text{mix}}$, should be accessed using \DeltaGmix\Gm and similar constructions—this construct will typeset as $\Delta_{\text{MIX}}G$ using the Thompson package option, for example, and as $\Delta_{\text{mix}}G$ using the Sandler package option.

\DeltaGfusion Similar entities for property changes on fusion, reaction, sublimation, and vaporization are defined. The heat of reaction is handled differently if \cP is its argument: one must typeset $\Delta C_P^{\text{rxn}}$ and $\Delta C_P^{\circ}$, rather than $\Delta C_{P\text{rxn}}$ and $\Delta C_{P\circ}$. 

\DeltaGvaporization
It would be nice not to have to handle \cP in a special way....
\let\Delta@rxn@sym\relax
\DeclareSubscrSymbol{@DeltarxncP}{\Delta@rxn@sym}{\pressure@symbol,\reaction}
\DeclareSubscrSymbol{@Deltarxn}{\Delta@rxn@sym}{\reaction}
\newcommand*{\Deltarxn}[1]{% 
  \def\tmp@arg{#1}\
  \def\tmp@@arg{\cP}\
  \ifx\tmp@arg\tmp@@arg\relax
  \def\@command{\def\Delta@rxn@sym{\Delta\@intensive\heatcapacitysymbol}\
    \@DeltarxncP}
  \else
  \def\@command{\def\Delta@rxn@sym{\Delta{#1}}\@Deltarxn}
  \fi
  \@command
}
\let\Delta@f@sym\relax
\DeclareSubscrSymbol{@Deltaf}{\Delta@f@sym}{f}
\newcommand*{\Deltaf}[1]{\def\Delta@f@sym{\Delta{#1}}\@Deltaf}
\fmix
The \fmix command is intended to describe fugacities in mixtures. It renders as \hat{f} by default, and would be used as \fmix_i or the like, producing \hat{f}_i; some authors like to use \hat{f}_i or just \hat{f}_i, and this command creates a consistent way to change between such options.
\newcommand*{\fmix}{\hat{f}}
\phimix
A similar command, \phimix, renders \hat{\phi} by default to represent the fugacity coefficient in the mixture.
\newcommand*{\phimix}{\hat{\phi}}
\fpure
The \fpure command is intended to describe fugacities in pure substances. It renders as f by default, and would be used as \fpure or \fpure_i or the like, producing f_i; some authors like to use f_i, and others like to use \hat{f}_i; this command creates a consistent way to change between the these options. A similar command for \phi is given, \phipure, for fugacity coefficients.
\newcommand*{\fpure}{f}
\let\phipure\phi

4.7 Partial Molar Quantities

\partialmolar
Partial molar quantities appear as \overline{G}_i or, for residual properties, \overline{G}_i^R, rather than something like \overline{G}_i or \overline{G}_i^R, which looks better but is harder to implement for obvious reasons. Their definitions allow them to be used as symbols, something like \Gpm_i, \Gpm^\{\IGM\}_i, \Gpm^\{\IGM\}_i, and even \Gpm^\{\IGM\}; they can also be treated as commands: \Gpm{i} is equivalent to \Gpm_i and \Gpm{\{\IGM\}}{i} is equivalent to \Gpm^{\{\IGM\}}_i. The macro \partialmolar can be used to create an arbitrary partial molar symbol.
4.8 Symbol Definitions

These macros define the user interface to the symbols for energy, volume, and so forth. We define the “extra” symbols \( \text{Bt} \) and \( \text{Mt} \) to represent arbitrary properties.

\[ \text{\texttt{\textbackslash Nt}} \] First, the extensive properties.
\[ \text{\texttt{\textbackslash Et}} \]
\[ \text{\texttt{\textbackslash Ut}} \]
\[ \text{\texttt{\textbackslash Ft}} \]
\[ \text{\texttt{\textbackslash Gt}} \]
\[ \text{\texttt{\textbackslash Lt}} \]
\[ \text{\texttt{\textbackslash At}} \]
\[ \text{\texttt{\textbackslash St}} \]
\[ \text{\texttt{\textbackslash Vt}} \]
\[ \text{\texttt{\textbackslash Qt}} \]
\[ \text{\texttt{\textbackslash Wt}} \]
\[ \text{\texttt{\textbackslash Mt}} \]
\[ \text{\texttt{\textbackslash Bt}} \]

\[ \text{\texttt{\textbackslash Em}} \]
\[ \text{\texttt{\textbackslash Fm}} \]
\[ \text{\texttt{\textbackslash Gm}} \]
\[ \text{\texttt{\textbackslash Hm}} \]
\[ \text{\texttt{\textbackslash Lm}} \]
\[ \text{\texttt{\textbackslash Am}} \]
\[ \text{\texttt{\textbackslash Sm}} \]
\[ \text{\texttt{\textbackslash Vm}} \]
\[ \text{\texttt{\textbackslash Qm}} \]

\[ \text{\texttt{\textbackslash Wm}} \] More molar properties.
\[ \text{\texttt{\textbackslash Mm}} \]
\[ \text{\texttt{\textbackslash Bm}} \]
\newcommand*{\Mm}{\@intensive M}
\newcommand*{\Bm}{\@intensive B}

\Es Now the specific (per unit mass) properties.
\Us \newcommand*{\Us}{\@specific\internalenergy@symbol}
\Fs \newcommand*{\Fs}{\@specific\Helmholtz@symbol}
\Gs \newcommand*{\Gs}{\@specific\Gibbs@symbol}
\Hs \newcommand*{\Hs}{\@specific\enthalpy@symbol}
\Ls \newcommand*{\Ls}{\@specific\Landau@symbol}
\As \newcommand*{\As}{\@specific\area@symbol}
\Ss \newcommand*{\Ss}{\@specific\entropy@symbol}
\Vs \newcommand*{\Vs}{\@specific\volume@symbol}
\Qs \newcommand*{\Qs}{\@specific\heat@symbol}
\Ws \newcommand*{\Ws}{\@specific\work@symbol}
\Ms \newcommand*{\Ms}{\@specific M}
\Bs \newcommand*{\Bs}{\@specific B}

\Epm Now the partial molar properties.
\Upm \newcommand*{\Upm}{\partialmolar{\totalenergy@symbol}}
\Hpm \newcommand*{\Hpm}{\partialmolar{\internalenergy@symbol}}
\Fpm \newcommand*{\Fpm}{\partialmolar{\enthalpy@symbol}}
\Gpm \newcommand*{\Gpm}{\partialmolar{\Helmholtz@symbol}}
\Apm \newcommand*{\Apm}{\partialmolar{\area@symbol}}
\Spm \newcommand*{\Spm}{\partialmolar{\entropy@symbol}}
\Vpm \newcommand*{\Vpm}{\partialmolar{\volume@symbol}}
\Lpm \newcommand*{\Lpm}{\partialmolar{\Omega}}
\Bpm \newcommand*{\Bpm}{\partialmolar{B}}

\cPpm Partial molar heat capacities are \textit{hard}, but the following implementation seems to work flawlessly...so far.
\cVpm \newcommand*{\cPpm}{% 
  \def@@@generic@pm##1##2{% 
    \thermo@overline{\pm@symbol^{##2}_{\pressure@symbol,\pm@arg}}% 
  } 
\newcommand*{\cVpm}{% 
  \def@@@generic@pm##1##2{% 
    \thermo@overline{\pm@symbol^{##2}_{\volume@symbol,\pm@arg}}% 
  }
4.9 Residual and Excess Properties

Macros are defined for residual properties (departure from non-ideal gases) and excess properties (departure from ideal solutions). We begin with two macros to use for defining generic residual and excess properties that are not already defined.

\newcommand*{\residual}{R}
\newcommand*{\excess}{E}

The ordinary residual properties are molar.
\newcommand*{\UR}{\Um^{\residual}}
\newcommand*{\HR}{\Hm^{\residual}}
\newcommand*{\FR}{\Fm^{\residual}}
\newcommand*{\GR}{\Gm^{\residual}}
\newcommand*{\VR}{\Vm^{\residual}}
\newcommand*{\SR}{\Sm^{\residual}}

We define similar macros for the extensive residual properties. These are appended by a \textit{t} (for “total”).
\newcommand*{\URt}{\Ut^{\residual}}
\newcommand*{\HRt}{\Ht^{\residual}}
\newcommand*{\FRt}{\Ft^{\residual}}
\newcommand*{\GRt}{\Gt^{\residual}}
\newcommand*{\VRt}{\Vt^{\residual}}
\newcommand*{\SRt}{\St^{\residual}}

Similar macros are declared for the specific residual properties. These are appended by an \textit{s}.
\newcommand*{\URs}{\Us^{\residual}}
\newcommand*{\HRs}{\Hs^{\residual}}
\newcommand*{\FRs}{\Fs^{\residual}}
\newcommand*{\GRs}{\Gs^{\residual}}
\newcommand*{\VRs}{\Vs^{\residual}}
\newcommand*{\SRs}{\Ss^{\residual}}

Now for the partial molar residual properties. These are appended by \textit{pm}.
\newcommand*{\URpm}{\partialmolar{\Um^{\residual}}}
\newcommand*{\HRpm}{\partialmolar{\Hm^{\residual}}}
\newcommand*{\FRpm}{\partialmolar{\Fm^{\residual}}}
\newcommand*{\GRpm}{\partialmolar{\Gm^{\residual}}}
\newcommand*{\VRpm}{\partialmolar{\Vm^{\residual}}}
\newcommand*{\SRpm}{\partialmolar{\Sm^{\residual}}}

Now for the partial molar excess properties. These are appended by \textit{pm}.
\newcommand*{\HE}{\Um^{\excess}}
\newcommand*{\FE}{\Fm^{\excess}}
\newcommand*{\GE}{\Gm^{\excess}}
\newcommand*{\VE}{\Vm^{\excess}}
\newcommand*{\SE}{\Sm^{\excess}}
\newcommand*{\UE}{\Um^{\excess}}
Similarly, the extensive excess properties.
\newcommand*{\UEt}{\Ut^\excess}
\newcommand*{\FEt}{\Ft^\excess}
\newcommand*{\GET}{\Gt^\excess}
\newcommand*{\VEt}{\Vt^\excess}
\newcommand*{\SET}{\St^\excess}

Now for the specific excess properties.
\newcommand*{\UES}{\Us^\excess}
\newcommand*{\FEs}{\Fs^\excess}
\newcommand*{\HEs}{\Hs^\excess}
\newcommand*{\GES}{\Gs^\excess}
\newcommand*{\SES}{\Ss^\excess}
\newcommand*{\VEs}{\Vs^\excess}

Finally, the excess partial molar quantities.
\newcommand*{\UEpm}{\partialmolar{\internalenergy@symbol^\excess}}
\newcommand*{\HEpm}{\partialmolar{\enthalpy@symbol^\excess}}
\newcommand*{\FEpm}{\partialmolar{\Helmholtz@symbol^\excess}}
\newcommand*{\GEpm}{\partialmolar{\Gibbs@symbol^\excess}}
\newcommand*{\VEpm}{\partialmolar{\volume@symbol^\excess}}
\newcommand*{\SEpm}{\partialmolar{\entropy@symbol^\excess}}

The \sumall command and its cousin, \sumallbutlast, simplify the typesetting of commonly-used sums; the command \prodall does the same thing for products, viz.,
\begin{align*}
\sumall_{i=1}^C x_i = 1 \quad \sumallbutlast_{i=1}^{C-1} x_i = 1 - x_C
\end{align*}
\[ K = \exp\left(\frac{-\Delta G^\circ_{\text{rxn}}}{RT}\right) = \prod_{i=1}^C a_i^{\nu_i} \]

The \IG, \IGM, and \IS macros (meaning “ideal gas,” “ideal gas mixture,” and “ideal solution,” respectively) should be used to make clean transitions between textbooks—some use “IM” rather than “IS” for example.
The Henry’s Law constants for the rational basis \( y_iP = x_ih_i \) and the molal basis \( y_iP = C_iH_i \) are given by the macros \Henryrat and \Henrymol, respectively. Using them this way consistently allows for easy switching back and forth.

\[ \newcommand*{\Henryrat}{h} \]
\[ \newcommand*{\Henrymol}{\mathcal{H}} \]

The ordinary activity coefficient is universally denoted \( \gamma \), so I have not defined a special macro for that. However, the Henry’s Law activity coefficients are far from universal, so I have defined macros to make their use consistent. The defaults render \gammarat as \( \gamma^* \) and \gammamol as \( \gamma^\square \). \gammamol will use \textquote{square} from packages if it is defined; if not, it “fakes it” with the definition below (based on the amsthm package).

\[ \begin{align*}
\text\small\text\AtBeginDocument{\%}
\text\small\providecommand*{\square}{\text{\hfil\vrule\vbox to.6em{\hrule width.6em\vfil\hrule}\vrule\hfil}}\% \\
\text\small\newcommand*{\gammarat}{\gamma^*} \\
\text\small\newcommand*{\gammamol}{\gamma^\square}
\end{align*} \]

\textquote{Jacobian} The \( \text\small\Jacobian \) command typesets the Jacobian, viz.,

\[ \begin{vmatrix}
\frac{\partial (K,L)}{\partial (X,Y)} \\
\frac{\partial (K)}{\partial X}_Y \left( \frac{\partial K}{\partial Y} \right)_X \\
\frac{\partial (L)}{\partial X}_Y \left( \frac{\partial L}{\partial Y} \right)_X
\end{vmatrix} \]

This macro is defined purely for convenience.

\[ \text\small\newcommand*{\Jacobian}{\frac{\partial (#1,#2)}{\partial (#3,#4)}}\% \]

\textbf{Change History}

\textbf{v1.00}

General: Initial public release ........ 1
Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols
\@extensive ........ 21
\@intensive ........ 21
\@specific ......... 21
\A
\allbut ........... 11, 23
\allbutlaststand ... 11, 23
\allcomponents .... 23
\allMs ............ 10, 23
\allMsbut .......... 10, 23
\allmus ........... 10, 23
\allmusbut ......... 10, 23
\allNs .......... 10, 15, 16, 23
\allNsbut .......... 10, 15, 16, 23
\allWs ........... 10, 23
\allWsbut .......... 10, 23
\allXs ........... 10, 16, 23
\allXsbut .......... 10, 16, 23
\allYs ........... 10, 23
\allYsbut .......... 10, 23
\alphaP ........... 4, 5, 36
\alphaS ........... 4, 5, 36
\Am ................ 4, 39
\Apm ................ 4, 39
\area@symbol ........ 19
\As ................ 4, 39
\At ................ 4, 38
\B
\Bm ................ 4, 38
\Bpm ................ 4, 39
\Bs ................ 4, 39
\Bt ................ 4, 38
\C
\compressibility@symbol .......... 5, 35
\cP ................ 4, 5, 35
\cPpm .............. 3, 39
\cPs ................ 3, 36
\cPt ................ 3, 35
\CV ................ 4, 5, 35
\CVpm ................ 3, 39
\CVs ................ 3, 36
\Ct ................ 3, 35
\D
\dbar .............. 15, 18
\G
\gammamol .......... 5, 6, 42
\gammarat .......... 5, 6, 42
\GE ................ 40
\GEpm .............. 41
\GEs .............. 41
\GEt .............. 41
\Gibbs@symbol .... 19
\Gm ................ 4, 38
\Gpm .............. 4, 39
\HR .............. 40
\HRpm .............. 40
\HRs .............. 40
\HRt .............. 40
\Gs ................ 4, 39
\Gt ................ 4, 38
\H
\HE ................ 40
\heat@symbol ........ 19
\heatcapacit@symbol ........ 5, 35
\Helmholtz@symbol ........ 19
\Henrymol .......... 5, 6, 42
\Henryrat .......... 5, 6, 42
\HEpm .............. 41
\HEs .............. 41
\HEt .............. 41
\Hm ................ 3, 4, 38
\Hp ................ 3, 4, 39
\HR .............. 40
\HRpm .............. 40
\HRs .............. 40
\HRt .............. 40
\Bs .............. 3, 4, 39
\Ht ................ 3, 4, 38
\I
\IG .............. 41
\IGM .............. 41
\internal@energy@symbol ........ 19
\IS .............. 41
\J
\Jacobian .......... 42
\K
\kappaS ........... 4, 5, 36