

Thermodynamik-Schnittstelle

für

CAPE-Anwendungen

Projekt der IK-CAPE

User's Guide

Version vom 09.10.98

Gliederung

1 EINLEITUNG	1-1
2 ALLGEMEINES	2-1
2.1 Stoffsysteme und Stoffsystemversionen	2-1
2.2 Tabelle der Stoffkonstanten	2-2
2.3 Tabelle der temperaturabhängigen Stoffdaten (Stoff-Funktionen)	2-3
2.4 Tabelle der Stoffdatenmatrizen	2-4
2.5 Tabelle der Stoffdaten-Parameter	2-5
2.6 Tabelle der binären temperaturabhängigen Stoffdaten	2-5
2.7 Tabelle der Inkrementtypen	2-5
2.8 Stoffdatentypen, die nur zur Berechnung benutzt werden	2-6
2.9 Gültige Funktionen zur Beschreibung der Temperaturabhängigkeit	2-6
2.10 Gültige Methoden zur Mittelwertbildung	2-7
2.10.1 Reinstoffdaten	2-7
2.10.2 Gasdichte	2-7
2.11 Gültige Methoden zur Berechnung von Aktivitätskoeffizienten L-V	2-8
2.12 Gültige Methoden zur Berechnung von Aktivitätskoeffizienten L-L	2-8
2.13 Gültige Methoden zur Berechnung des Realverhaltens in der Gasphase	2-8
2.14 Gültige Methoden zur Berechnung von Zustandsgleichungen	2-9
2.15 Gültige Methoden zur Berechnung von Exzeßenthalpien	2-9
2.16 Gültige Methoden zur Berechnung der isothermen Druckabhängigkeit der Enthalpie im Gas	2-9
2.17 Chemische Reaktionen	2-10
3 NEUTRALES STOFFDATENFILE	3-1
3.1 Beispielfile	3-1
3.2 Der Steuersatz	3-3
3.3 Der Systemsteuersatz	3-3
3.4 Daten	3-4
3.4.1 Komponenten-Kurznamen	3-4
3.4.2 Komponenten-Langnamen	3-4
3.4.3 Komponenten-CAS-Nummern	3-4
3.4.4 Stoffkonstanten	3-4
3.4.5 Parametersätze	3-5
3.4.6 Koeffizientensätze für Stoff-Funktionen	3-5
3.4.7 Matrizen	3-5

3.4.8 Koeffizientensätze für binäre Stoff-Funktionen	3-6
3.5 Berechnungsbeschreibungen für Phasengleichgewichte	3-6
3.5.1 Flüssig-Dampf-Gleichgewichte	3-6
3.5.2 Flüssig-Flüssig-Gleichgewichte	3-6
3.6 Berechnungsbeschreibungen für Enthalpien	3-7
3.7 Chemische Reaktionen	3-8
3.7.1 Gleichgewichtsreaktionen	3-8
3.7.2 Kinetisch kontrollierte Reaktionen	3-9
3.7.3 Reaktionen mit vorgegebenem Umsatz	3-10
3.7.4 Reaktionen mit vorgegebener Reaktionskoordinate	3-10
3.8 Mittelwerte	3-10
3.9 Inkremente	3-11
3.10 Phasenschlüssel	3-12
3.11 Kommentare	3-12
4 BESCHREIBUNG DER SCHNITTSTELLENPROGRAMME	4-1
4.1 Beschreibung der Darstellung	4-1
4.2 Einspeichern der Basisdaten	4-1
T_LOAD	4-2
4.3 Auslesen von Daten aus der internen Struktur	4-3
T_G_AVER	4-4
T_G_BINA_COEFF	4-5
T_G_CHEMICS	4-7
T_G_CH_ALLG	4-8
T_G_CH_EQ	4-9
T_G_CH_KI	4-10
T_G_CH_PHASE	4-12
T_G_CH_TYPE	4-13
T_G_COMP_NUMBER	4-14
T_G_ENTH	4-15
T_G_IDEN_COMP	4-17
T_G_INFO_COMP	4-19
T_G_INKR	4-20
T_G_LABEL_ENTH	4-21
T_G_LLEQ	4-22
T_G_LVEQ	4-23
T_G_MATRIX	4-25
T_G_NAME_LENGTH	4-26
T_G_NO_REAC	4-27
T_G_PARAM	4-28
T_G_PURE_COEFF	4-29
T_G_PURE_EXTR	4-31
T_G_REACTION	4-32
T_G_SINGLE_COMP	4-34
T_G_SYSTEM_NAME	4-35
4.4 Methoden in der internen Datenstruktur setzen	4-36
T_S_CHEMICS	4-37
T_S_ENTH	4-38

T_S_LLEQ	4-39
T_S_LVEQ	4-40
T_S_SYSTEM	4-41
T_S_VERSION	4-42

4.5 Berechnungsprogramme	4-43
T_AVER	4-44
T_AVER_DERIVATIVE	4-46
T_CHEM	4-47
T_CHEM_DERIVATIVE	4-50
T_CH	4-51
T_CH_DERIVATIVE	4-52
T_COMPR	4-54
T_COMPR_DERIVATIVE	4-56
T_ENTH	4-57
T_ENTH_DERIVATIVE	4-59
T_FUGA	4-60
T_FUGA_DERIVATIVE	4-62
T_GAMMA	4-63
T_GAMMA_DERIVATIVE	4-65
T_H_EX	4-66
T_H_EX_DERIVATIVE	4-67
T_H_ISO	4-68
T_H_ISO_DERIVATIVE	4-70
T_LLEQ	4-71
T_LLEQ_DERIVATIVE	4-73
T_LVEQ	4-74
T_LVEQ_DERIVATIVE	4-76
T_PURE	4-78
T_PURE_DERIVATIVE	4-80

1 Einleitung

Die im folgenden beschriebenen Schnittstellen-Komponenten setzen auf dem Konzept eines Software-Pakets auf, das - unabhängig von Umfang und innerer Komplexität - über eine begrenzte Anzahl von Unterprogrammaufrufen sämtliche thermodynamische Zustandsgrößen und Stoffdaten liefert, die in einem übergeordneten CAPE-Anwendungsprogramm benötigt werden.

Die Unterprogramme sind in Standard-Fortran 77 geschrieben, mit den in den "IK-CAPE Programmierrichtlinien für Fortran" festgelegten Ausnahmen.

Die verschiedenen Unterprogramme erledigen ihre Aufgabe durch Zugriff auf interne Strukturen, die jedoch für den Nutzer diese Pakets unsichtbar bleiben. Sie sind ausschließlich wichtig für Weiterentwicklungen innerhalb des Pakets selbst.

Die internen Strukturen werden in einem konkreten Anwendungsfall durch Einlesen einer Datei mit den Stoffdaten und thermodynamischen Anweisungen des aktuellen Projektes spezifisch aufgebaut und gefüllt. Das Einlesen wird wieder durch Aufruf eines Unterprogramms erledigt, das in dem Paket enthalten ist.

Das Format der oben genannten Datei wird ebenfalls beschrieben. Es ist so konzipiert, daß es für spätere Erweiterungen in den Daten oder Methoden offen bleibt.

2 Allgemeines

2.1 Stoffsysteme und Stoffsystemversionen

Das Thermodynamik-Paket ist in der Lage, mehrere Stoffsysteme gleichzeitig zu verwalten.

Diese Fähigkeit ist z. B. wichtig, um die Stoffdaten auf beiden Seiten eines Wärmetauschers unabhängig voneinander zu beschreiben. Ein zweiter wichtiger Anwendungsfall sind große Anlagenfließbilder. Hier kann die Aufteilung in mehrere Stoffsysteme und das Zuordnen von Anlagenteilen zu diesen Stoffsystemen zu einer erheblichen Reduzierung der Gleichungen und damit auch der Rechenzeit führen.

Verschiedene Stoffsysteme werden durch verschiedene Stoffdaten-Label und Versionsnummern unterschieden.

Die durch verschiedene Label unterschiedenen Stoffsysteme haben keinerlei Beziehungen untereinander. Sie können sich nicht nur in den Stoffdaten, sondern auch in der Zahl und Art der Komponenten unterscheiden.

Im Unterschied dazu stellen die verschiedenen Versionsnummern zum selben Label Varianten der Stoffdatenwerte zur Version 1 (= Basisversion) dar. Eine Version mit höherer Versionsnummer kann erst angegeben werden, wenn die Basisversion schon existiert. Gewisse Daten der Basisversion sind für höhere Versionen unveränderbar:

- Anzahl Komponenten
- Komponentennamen
- Molekulargewichte
- Enthalpiebeschreibungen

2.2 Tabelle der Stoffkonstanten

Kürzel	Nummer	Stoffdatentyp	Einheit
MOLW	1	Molmasse	kg/kmol
TC	2	kritische Temperatur	Kelvin
PC	3	kritischer Druck	Pascal
RHOC	4	kritische Dichte	kmol/m ³
AC	5	azentrischer Faktor	
MP	6	Schmelzpunkt	Kelvin
MH	7	Schmelzwärme	J/kmol
VO25	8	spez. Volumen bei 25 Grad Celsius	m ³ /kmol
BETS	9	Bildungsenthalpie des Feststoffs	J/kmol
BETL	10	Bildungsenthalpie der Flüssigkeit	J/kmol
BETG	11	Bildungsenthalpie des Gases	J/kmol
FETS	12	freie Bildungsenthalpie des Feststoffs	J/kmol
FETL	13	freie Bildungsenthalpie der Flüssigkeit	J/kmol
FETG	14	freie Bildungsenthalpie des Gases	J/kmol
ETRS	15	Standardentropie des Feststoffs	J/kmol/K
ETRL	16	Standardentropie der Flüssigkeit	J/kmol/K
ETRG	17	Standardentropie des Gases	J/kmol/K
PARA	18	Parachor	g ^{1/4} cm ³ /s ^{1/2} /mol
DIVO	19	Diffusionsvolumen nach Fuller	

2.3 Tabelle der temperaturabhängigen Stoffdaten (Stoff-Funktionen)

Kürzel	Nummer	Stoffdatentyp	Einheit
CS	1	spezifische Wärmekapazität des Feststoffes	J/kmol/K
CL	2	spezifische Wärmekapazität der Flüssigkeit	J/kmol/K
CPID	3	spezifische Wärmekapazität des idealen Gases	J/kmol/K
HSUB	4	Sublimationsenthalpie	J/kmol
HVAP	5	Verdampfungsenthalpie	J/kmol
DENS	6	Dichte des Feststoffes	kmol/m ³
DENL	7	Dichte der Flüssigkeit	kmol/m ³
VISL	8	dyn. Viskosität der Flüssigkeit	Pa s
VISV	9	dyn. Viskosität des Gases	Pa s
KLIQ	10	Wärmeleitfähigkeit der Flüssigkeit	W/m/Kelv
KVAP	11	Wärmeleitfähigkeit des Gases	W/m/Kelv
VPS	12	Sublimationsdruck	Pa
VP	13	Dampfdruck	Pa
ST	14	Oberflächenspannung	N/m
VIR	15	2. Virialkoeffizient	m ³ /kmol

2.4 Tabelle der Stoffdatenmatrizen

Kürzel	Nummer	Stoffdatentyp
NRTL A	1	NRTL-Matrix A LV
NRTL B	1	NRTL-Matrix B LV
NRTL C	1	NRTL-Matrix Alpha LV
NRTL D	1	NRTL-Matrix Beta LV
NRTL E	1	NRTL-Matrix E LV
NRTL F	1	NRTL-Matrix F LV
NRLL ALPH	2	NRTL-Matrix Alpha LL
NRLL BETA	2	NRTL-Matrix Beta LL
NRLL A	2	NRTL-Matrix A LL
NRLL B	2	NRTL-Matrix B LL
QUAV A	3	UNIQUAC-Matrix A LV
QUAV B	3	UNIQUAC-Matrix B LV
QUAV C	3	UNIQUAC-Matrix C LV
QUAV D	3	UNIQUAC-Matrix D LV
QUAL	4	UNIQUAC-Matrix LL
SRK	5	Redlich-Kwong-Soave-Matrix
WILS A	6	Wilson-Matrix A
WILS B	6	Wilson-Matrix B
WILS C	6	Wilson-Matrix C
WILS D	6	Wilson-Matrix D
HNRY A	7	Henry-Matrix A
HNRY B	7	Henry-Matrix B
HNRY C	7	Henry-Matrix C
HNRY D	7	Henry-Matrix D
PR	8	Peng-Robinson-Matrix
DIFV	9	Diffusionskoeffizienten des Gases

2.5 Tabelle der Stoffdaten-Parameter

Kürzel	Nummer	Stoffdatentyp	Einheit
UNIQ	1	UNIQUAC-Parameter r, q	
RKS	2	Redlich-Kwong-Soave Parameter a, b	Pa*m ⁶ *Kelv ^{0.5} /kmol ² m ³ /kmol
PRAB	3	Peng-Robinson-Parameter a, b	Pa*m ⁶ *Kelv ^{0.5} /kmol ² m ³ /kmol
FLOR	4	Flory-Huggins Parameter r, chi0, chi1	
DMER	5	Dimerisationsparameter a, b	Pa, Kelv
TMER	6	Tetramerisationsparameter a, b	Pa, Kelv
HMER	7	Hexamerisationsparameter a, b	Pa, Kelv

2.6 Tabelle der binären temperaturabhängigen Stoffdaten

Kürzel	Nummer	Stoffdatentyp	Einheit
CROS	1	Kreuzvirialkoeffizienten	m ³ /kmol
HMIA	2	Mischungswärme, Temperaturfunktion des A Koeffizienten	
HMIB	3	Mischungswärme, Temperaturfunktion des B Koeffizienten	
HMIC	4	Mischungswärme, Temperaturfunktion des C Koeffizienten	
HMID	5	Mischungswärme, Temperaturfunktion des D Koeffizienten	
HMIE	6	Mischungswärme, Temperaturfunktion des E Koeffizienten	
HMIF	7	Mischungswärme, Temperaturfunktion des F Koeffizienten	

2.7 Tabelle der Inkrementtypen

Kürzel	Nummer	Stoffdatentyp
UFGR	1	Unifac-Inkmente

2.8 Stoffdatentypen, die nur zur Berechnung benutzt werden

Kürzel	Nummer	Stoffdatentyp
DENV	1	Dichte des Gases

2.9 Gültige Funktionen zur Beschreibung der Temperaturabhängigkeit

Methoden-Kennung	Erläuterung
POLY	Polynom
EPOL	Polynom als Exponent
ANTO	Antoine-Gleichung
ANT1	erweiterte Antoine-Gleichung
WATS	Watson-Gleichung
WAGN	Wagner-Gleichung
SUTH	Sutherland-Gleichung
KIRC	Kirchhoff-Gleichung
CPL	HOECHST: Gleichung für die spez. Wärmekapazität der Flüssigkeit
ICPL	HOECHST: Gleichung für die spez. Wärmekapazität der Flüssigkeit
VISC	HOECHST: Gleichung für die dynamische Viskosität
RACK	Rackett-Gleichung nach DIPPR
ALYL	Aly-Lee-Gleichung
KIR1	erweiterte Kirchhoff-Gleichung nach DIPPR
DIP4	DIPPR-Gleichung für Verdampfungsenthalpie und Oberflächenspannung
DIP5	DIPPR-Gleichung für Wärmeleitfähigkeit und Viskosität des Gases

2.10 Gültige Methoden zur Mittelwertbildung

2.10.1 Reinstoffdaten

Methoden-Kennung	Erläuterung
MOLA	Molanteilige Mittelung
MASS	Gewichtsanteilige Mittelung
MOLG	Molanteilig-Logarithmische Mittelung
MALG	Gewichtsanteilig-logarithmische Mittelung
VOLU	Mittelwert der Dichte über das Volumen
WLFG	Spezielle Art der Mittelung für Wärmeleitfähigkeiten des Gases
VISG	Spezielle Art der Mittelung für Viskositäten des Gases
WILK	Mittelung nach Wilke für die Viskosität des Gases
WAMA	Mittelung nach Wassiljewa, Mason, Saxena für die Wärmeleitfähigkeit des Gases
DIST	Spezielle Art der Mittelung für Oberflächenspannungen nach DIPPR
DIKL	Spezielle Art der Mittelung für Wärmeleitfähigkeiten flüssig nach DIPPR

2.10.2 Gasdichte

Methoden-Kennung	Erläuterung
NONE	$Z = 1$
VLEQ	Z nach Phasengleichgewichtsmethode
RKS	Z nach Redlich-Kwong-Soave
PR	Z nach Peng-Robinson
PMER	Z unter Berücksichtigung der Assoziation
VIRI	Z aus der Virialgleichung

2.11 Gültige Methoden zur Berechnung von Aktivitätskoeffizienten L-V

Methoden-Kennung	Erläuterung
IDEA	Aktivitätskoeffizienten = 1
NRTL	Aktivitätskoeffizienten nach dem NRTL-Ansatz
UNIQU	Aktivitätskoeffizienten nach dem UNIQUAC-Ansatz
WILS	Aktivitätskoeffizienten nach dem Wilson-Ansatz
FLOR	Aktivitätskoeffizienten nach Flory-Huggins
UNIF	Aktivitätskoeffizienten nach UNIFAC

2.12 Gültige Methoden zur Berechnung von Aktivitätskoeffizienten L-L

Methoden-Kennung	Erläuterung
NONE	Aktivitätskoeffizienten = 1
NRTL	Aktivitätskoeffizienten nach dem NRTL-Ansatz
UNIQU	Aktivitätskoeffizienten nach dem UNIQUAC-Ansatz
UNIF	Aktivitätskoeffizienten nach UNIFAC

2.13 Gültige Methoden zur Berechnung des Realverhaltens in der Gasphase

Methoden-Kennung	Erläuterung
NONE	Fugazitätskoeffizienten = 1
PR	Fugazitätskoeffizienten nach dem Ansatz von Peng-Robinson
RKS	Fugazitätskoeffizienten nach dem Ansatz von Redlich-Kwong-Soave
VIRI	Fugazitätskoeffizienten nach der Virialgleichung
PMER	Assoziation in der Gasphase

2.14 Gültige Methoden zur Berechnung von Zustandsgleichungen

Methoden-Kennung	Erläuterung
NONE	Aktivitätskoeffizienten = 1
RKS	Zustandsgleichung nach dem Ansatz von Redlich-Kwong-Soave
PR	Zustandsgleichung nach dem Ansatz von Peng-Robinson

2.15 Gültige Methoden zur Berechnung von Exzeßenthalpien

Methoden-Kennung	Erläuterung
NONE	Exzeßenthalpie = 0
NRTL	Exzeßenthalpie nach NRTL
UNIQU	Exzeßenthalpie nach UNIQUAC
WILS	Exzeßenthalpie nach Wilson
FLOR	Exzeßenthalpie nach Flory-Huggins
HMIX	Mischungswärme nach dem Ansatz von Redlich-Kister

2.16 Gültige Methoden zur Berechnung der isothermen Druckabhängigkeit der Enthalpie im Gas

Methoden-Kennung	Erläuterung
NONE	Druckabhängigkeit = 0
RKS	Druckabhängigkeit nach dem Ansatz von Redlich-Kwong-Soave
PR	Druckabhängigkeit nach dem Ansatz von Peng-Robinson
PMER	Assoziation in der Gasphase

2.17 Chemische Reaktionen

Methoden-Kennung	Erläuterung
COOR	Vorgegebene Reaktionskoordinate
CONV	Vorgegebener Umsatz bezogen auf den Eingangszustand
STAT	Vorgegebener Umsatz bezogen auf den Ausgangszustand
EQLM	Gleichgewichtsreaktion in der Flüssigphase als Funktion der Molenbrüche
EQVM	Gleichgewichtsreaktion in der Dampfphase als Funktion der Molenbrüche
EQLC	Gleichgewichtsreaktion in der Flüssigphase als Funktion der Konzentrationen
EQVC	Gleichgewichtsreaktion in der Dampfphase als Funktion der Konzentrationen
EQLA	Gleichgewichtsreaktion in der Flüssigphase als Funktion der Aktivitätskoeffizienten
EQVP	Gleichgewichtsreaktion in der Dampfphase als Funktion der Partialdrücke
EQLF	Gleichgewichtsreaktion in der Flüssigphase als Funktion der Fugazität
EQVF	Gleichgewichtsreaktion in der Dampfphase als Funktion der Fugazität
KILM	kinetisch kontrollierte Reaktion in der Flüssigphase als Funktion der Molenbrüche
KIVM	kinetisch kontrollierte Reaktion in der Dampfphase als Funktion der Molenbrüche
KILC	kinetisch kontrollierte Reaktion in der Flüssigphase als Funktion der Konzentrationen
KIVC	kinetisch kontrollierte Reaktion in der Dampfphase als Funktion der Konzentrationen
KILW	kinetisch kontrollierte Reaktion in der Flüssigphase als Funktion der Massen
KIVW	kinetisch kontrollierte Reaktion in der Dampfphase als Funktion der Massen

3 Neutrales Stoffdatenfile

Das Einlesen von Stoffdateninformationen und Methodenkennzeichnungen geschieht über eine Ascii-Datei mit einer Satzlänge von 80 Bytes. Sie werden durch einen Aufruf des Unterprogramms T_LOAD in die internen Speicherstrukturen des Thermodynamik-Paktes eingelesen. Dabei werden die Daten weder manipuliert noch geprüft.

Der Aufbau des neutralen Stoffdatenfiles ist wie folgt festgelegt:

- Steuersatz mit Beschreibungsgrößen des Gesamtproblems
- für jedes Stoff-System
 - ein Steuersatz mit den Beschreibungsgrößen und
 - den Daten des Stoffsystems in internen Einheiten.

3.1 Beispielfile

Beschreibung des Systems HCL-Wasser unter Berücksichtigung der Elektrolytbildung durch Reaktion:

```

THER      4 40 8 0 1
SYST DATA  1 4
NAME      1 WASSER
SHOR      1 H2O
MOLW      1 EXPL 1.80000000D+01
CPID      1 EXPL POLY  3 2.72149902D+02 4.83150085D+02
3.16838941D+04 4.54897143D+00 6.28571429D-03
CL        1 EXPL POLY  3 2.72149902D+02 4.83150085D+02
1.02428103D+05 -1.67978314D+02 2.59142857D-01
HVAP      1 EXPL WATS  4 2.72149902D+02 4.83150085D+02
6.39728596D+06 3.29398380D-01 6.47130005D+02 0.00000000D+00
VP        1 EXPL ANTO  3 2.72149902D+02 4.83150085D+02
2.33009463D+01 3.89239727D+03 -4.26586796D+01
NAME      2 CHLORWASSERSTOFF
SHOR      2 HCL
MOLW      2 EXPL 3.60000000D+01
CPID      2 EXPL POLY  3 2.72149902D+02 4.83150085D+02
3.18058134D+04 -1.46595429D+01 1.94285714D-02
CL        2 EXPL POLY  1 2.72149902D+02 4.83150085D+02
2.50000000D+05
HVAP      2 EXPL WATS  4 2.72149902D+02 4.83150085D+02
2.61636984D+06 3.80000000D-01 3.24649994D+02 0.00000000D+00
VP        2 EXPL ANTO  3 2.72149902D+02 4.83150085D+02
3.51640422D+01 1.81798213D+04 6.18714299D+02
NAME      3 (H2O)H
SHOR      3 (H2O)H
MOLW      3 EXPL 1.90000000D+01
CPID      3 EXPL POLY  1 2.72149902D+02 4.83150085D+02
1.00000000D+03
CL        3 EXPL POLY  3 2.72149902D+02 4.83150085D+02
2.78614783D+05 -2.51830857D+02 3.88571429D-01
HVAP      3 EXPL POLY  1 2.72149902D+02 4.83150085D+02
1.00000000D+03
VP        3 EXPL ANTO  3 2.72149902D+02 4.83150085D+02
-4.26073594D+00 7.46152845D+02 8.73416793D+00
NAME      4 2(H2O)CL
SHOR      4 2(H2O)CL
MOLW      4 EXPL 7.10000000D+01
CPID      4 EXPL POLY  1 2.72149902D+02 4.83150085D+02
1.00000000D+03
```

CL 4 EXPL POLY 3 2.72149902D+02 4.83150085D+02
 2.78614783D+05 -2.51830857D+02 3.88571429D-01
 HVAP 4 EXPL POLY 1 2.72149902D+02 4.83150085D+02
 1.00000000D+03
 VP 4 EXPL ANTO 3 2.72149902D+02 4.83150085D+02
 4.26073594D+00 7.46152845D+02 8.73416793D+00
 AVER MOLA MOLW
 AVER MOLA CPID
 AVER MOLA CL
 AVER MOLA HVAP
 ENTH SP00 NRTL NONE
 PHAS 1 LIQU
 PHAS 2 LIQU
 PHAS 3 LIQU
 PHAS 4 LIQU
 TREF 1 2.73150000D+02
 TREF 2 2.73150000D+02
 TREF 3 2.73150000D+02
 TREF 4 2.73150000D+02
 T_PH 1 2.73150000D+02
 T_PH 2 2.73150000D+02
 HREF 1 0.00000000D+00
 HREF 2 0.00000000D+00
 HREF 3 -2.27388000D+07
 HREF 4 -2.27388000D+07
 NRTL C 1 2 -5.39800000D-02 -5.39800000D-02
 NRTL C 1 3 -1.07500000D-01 -1.07500000D-01
 NRTL C 1 4 -1.07500000D-01 -1.07500000D-01
 NRTL C 2 3 3.09800000D-01 3.09800000D-01
 NRTL C 2 4 3.09800000D-01 3.09800000D-01
 NRTL C 3 4 2.50000000D-01 2.50000000D-01
 NRTL A 1 2 -1.12300000D+01 1.32100000D+01
 NRTL A 1 3 -2.85000000D+00 1.99700000D+00
 NRTL A 1 4 -2.85000000D+00 1.99700000D+00
 NRTL A 2 3 -4.89600000D+00 4.88400000D+00
 NRTL A 2 4 -4.89600000D+00 4.88400000D+00
 NRTL B 1 2 -3.76978400D+03 -1.38291961D+03
 NRTL B 1 3 -1.99987973D+02 -1.97188146D+03
 NRTL B 1 4 -1.99987973D+02 -1.97188146D+03
 NRTL B 2 3 -5.68965784D+02 -3.17980829D+02
 NRTL B 2 4 -5.68965784D+02 -3.17980829D+02
 LVEQ SP00 NRTL NONE NONE NONE NONE
 CHEM REA1 1
 EQLA 1 0.00000000D+00
 F(T) 2 -1.88400000D+01 5.47000000D+03
 STOE 1 -3.
 STOE 2 -1.
 STOE 3 1.
 STOE 4 1.

3.2 Der Steuersatz

Argumente	Format
Satzkennung = THER	A4, 1X
Maximale Komponentenzahl	I5
Anzahl Zeichen des Komponenten-Longnamens	I5
Anzahl Zeichen des Komponenten-Kurznamens	I5
Anzahl Zeichen der Cas-Nummer	I5
maximale Anzahl Reaktionen in einem Reaktionssystem	I5

3.3 Der Systemsteuersatz

Argumente	Format
Satzkennung = SYST	A4, 1X
Stoffsystemname	A4, 1X
Versionsnummer	I5
Komponentenzahl	I5

3.4 Daten

Die Datensätze beginnen mit einer Kennung, die den Datentyp beschreibt. Die Reihenfolge der Daten ist beliebig.

Die Formate der Daten sind abhängig vom Datentyp. Soweit möglich wurden die Daten in Datenklassen mit einheitlichen Formaten zusammengefaßt. Alle Einheiten sind interne Einheiten.

Die Datentypen und ihre Einheiten sind im Kapitel Allgemeines festgelegt.

3.4.1 Komponenten-Kurznamen

Argumente	Format
Satzkennung = SHOR	A4, 1X
Komponentennummer	I4, 1X
Komponenten-Kurzname	A

3.4.2 Komponenten-Langnamen

Argumente	Format
Satzkennung = NAME	A4, 1X
Komponentennummer	I4, 1X
Komponenten-Langname	A

3.4.3 Komponenten-CAS-Nummern

Argumente	Format
Satzkennung = CASN	A4, 1X
Komponentennummer	I4, 1X
Komponenten-CAS-Nummer	A

3.4.4 Stoffkonstanten

Die gültigen Stoffkonstanten sind mit ihren Typkürzeln im Kapitel Allgemeines beschrieben. Diese Typkürzel sind jeweils als Kennung dem Eingabesatz voranzustellen.

Argumente	Format
Typkürzel	A4, 1X
Komponentennummer	I4, 1X
Quelle=Hinweis auf Herkunft der Daten	A4,1X
Wert	1D16.8

3.4.5 Parametersätze

Die gültigen Parametersätze sind mit ihren Typkürzeln im Kapitel Allgemeines beschrieben. Diese Typkürzel sind jeweils als Kennung dem Eingabesatz voranzustellen.

Argumente	Format
Typkürzel	A4, 1X
Komponentennummer	I4, 1X
Wert 1 bis Wert 4	4D16.8

3.4.6 Koeffizientensätze für Stoff-Funktionen

Die gültigen Datentypen sind mit ihren Typkürzeln im Kapitel Allgemeines beschrieben. Diese Typkürzel sind jeweils als Kennung dem ersten Eingabesatz voranzustellen. Stoffdaten-Funktionen werden in mehreren Sätzen eingegeben. Der erste Eingabesatz erhält hierbei die Funktion eines Steuersatzes. Im 2. und eventuell 3. Satz werden die Koeffizienten angegeben.

Argumente	Format
Typkürzel	A4, 1X
Komponentennummer	I4, 1X
Quelle=Hinweis auf Herkunft der Daten	A4,1X
Ansatzkennung	A4, 1X
Koeffizientenzahl	I5
untere Temperaturgrenze des Gültigkeitsbereichs	D16.8
obere Temperaturgrenze des Gültigkeitsbereichs	D16.8
Extrapolationskennung (=0 oder =1 für spez. Extrapolation)	I5

Fortsetzungssatz:

Argumente	Format
Koeffizienten	5D16.8

3.4.7 Matrizen

Die gültigen Datentypen sind mit ihren Typkürzeln im Kapitel Allgemeines beschrieben. Diese Typkürzel sind jeweils als Kennung dem ersten Eingabesatz voranzustellen. Die Elemente werden paarweise, jeweils 1 Paar je Satz, eingelesen.

Argumente	Format
Typkürzel	A10
Komponentennummer_i	I4, 1X
Komponentennummer_j	I4, 1X
$W(i,j)$	1D16.8
$W(j,i)$	1D16.8

3.4.8 Koeffizientensätze für binäre Stoff-Funktionen

Die gültigen Datentypen sind mit ihren Typkürzeln im Kapitel Allgemeines beschrieben. Diese Typkürzel sind jeweils als Kennung dem ersten Eingabesatz voranzustellen. Stoffdaten-Funktionen werden in mehreren Sätzen eingegeben. Der erste Eingabesatz erhält hierbei die Funktion eines Steuersatzes. Im 2. und eventuell 3. Satz werden die Koeffizienten angegeben.

Argumente	Format
Typkürzel	A4, 1X
Komponentennummer_i	I4, 1X
Komponentennummer_j	I4, 1X
Quelle=Hinweis auf Herkunft der Daten	A4,1X
Ansatzkennung	A4, 1X
Koeffizientenzahl	I4, 1X

Fortsetzungssatz:

Argumente	Format
Koeffizienten	5D16.8

3.5 Berechnungsbeschreibungen für Phasengleichgewichte

Alle Beschreibungen der Berechnungsmethoden bzw. Berechnungswege für Phasengleichgewichte werden in Labels gespeichert.

3.5.1 Flüssig-Dampf-Gleichgewichte

Argumente	Format
Satzkennung = LVEQ	A4, 1X
Label	A4, 1X
Methode für Aktivitätskoeffizienten	A4, 1X
Methode für Fugazitätskoeffizienten	A4, 1X
Poynting-Korrektur	A4, 1X
Inertbehandlung	A4, 1X
Zustandsgleichungen	A4, 1X

3.5.2 Flüssig-Flüssig-Gleichgewichte

Argumente	Format
Satzkennung = LLEQ	A4, 1X
Label	A4, 1X
Methode für Aktivitätskoeffizienten	A4, 1X

3.6 Berechnungsbeschreibungen für Enthalpien

Alle Beschreibungen der Berechnungsmethoden bzw. Berechnungswege für Enthalpien werden in Labels gespeichert.

Die Festlegung, wie die Enthalpie eines vorgegebenen Gemisches in einem vorgegebenen Zustand zu berechnen ist, umfaßt Aussagen über

- den Referenzpunkt mit Angabe der Temperatur und Phase
- den Wert der Bildungsenthalpie am Referenzpunkt
- den Integrationsweg zum gewünschten Zustandspunkt
- die Methode zur Berechnung der Druckabhängigkeit
- die Methode zur Berechnung der Exzeßenthalpie

Diese Festlegung wird aus einem Block von Sätzen aufgebaut. Dieser Block wird mit einem ENTH- Satz eingeleitet, der die allgemeinen Daten der Enthalpieberechnung enthält. Die Folgesätze mit den Kennungen PHAS, TREF, HREF, T-PH spezifizieren für jede Komponente den Enthalpienullpunkt sowie den additiven Term der Bildungsenthalpie und den Phasenumwandlungspunkt zwischen Flüssigkeit und Gas.

Argumente	Format
Satzkennung = ENTH	A4, 1X
Label	A4,1X
Methode für Exzeßenthalpie	A4,1X
Methode für Druckabhängigkeit im Gas	A4,1X

Folgesatz : Phasenkennungen

Argumente	Format
Satzkennung = PHAS	A4, 1X
Komponentennummer	I4,1X
Referenzphase	A4,1X

Als Phasenkennungen sind zugelassen: LIQU , GAS, SOLI .

Für alle nicht angegebenen Komponenten wird als Referenzphase standardmäßig LIQUID gewählt.

Folgesatz: Referenztemperatur

Argumente	Format
Satzkennung = TREF	A4, 1X
Komponentennummer	I4,1X
Referenztemperatur	D16.8

Für alle nicht angegebenen Komponenten wird als Referenztemperatur standardmäßig 273.15 Kelvin gewählt.

Folgesatz: Referenzenthalpie

Argumente	Format
Satzkennung = HREF	A4, 1X
Komponentennummer	I4,1X
Referenzenthalpie	D16.8

Für alle nicht angegebenen Komponenten wird als Referenzenthalpie standardmäßig 0 J/kmol gewählt.

Folgesatz: Phasenumwandlungstemperatur

Argumente	Format
Satzkennung = T-PH	A4, 1X
Komponentennummer	I4,1X
Phasenumwandlungstemperatur Flüssig-Gas	D16.8

Für alle nicht angegebenen Komponenten wird als Umwandlungstemperatur standardmäßig Systemtemperatur gewählt.

3.7 Chemische Reaktionen

Die Daten eines Reaktionssystems werden in einer Folge von Sätzen beschrieben. Der erste Satz enthält den Namen des Reaktionssystems (Chemieblocks) und die Anzahl der simultan ablaufenden Reaktionen.

Argumente	Format
Satzkennung = CHEM	A4, 1X
Label	A4,1X
Anzahl Reaktionen	I5

Für jede Reaktion folgen nun Datensätze, die die Reaktion beschreiben. Der Aufbau der Datensätze ist abhängig vom Reaktionstyp.

3.7.1 Gleichgewichtsreaktionen

Gleichgewichtsreaktionen werden durch den Reaktionstyp, die Reaktionsenthalpie, die stöchiometrischen Koeffizienten und durch Koeffizienten zur Bestimmung der Temperaturabhängigkeit der Gleichgewichtsfunktion beschrieben.
Die Kennungen für Gleichgewichtsreaktionen sind im Kapitel Allgemeines beschrieben.

Argumente	Format
Satzkennung = Reaktionstyp	A4, 1X
Reaktionsnummer	I4,1X
Reaktionsenthalpie	D16.8

Folgesätze zur Angabe der stöchiometrischen Koeffizienten:

Argumente	Format
Satzkennung = STOE	A4, 1X
Komponentennummer	I4,1X
Stöchiometrischer Koeffizient	D16.8

Folgesatz zur Beschreibung der Temperaturabhängigkeit der Gleichgewichtsfunktion:

Argumente	Format
Satzkennung = F(T)	A4, 1X
Anzahl Koeffizienten	I4,1X
Koeffizienten (eventuell 2. Zeile)	4D16.8

3.7.2 Kinetisch kontrollierte Reaktionen

Kinetisch kontrollierte Reaktionen werden durch den Reaktionstyp, die Reaktionsenthalpie, die stöchiometrischen Koeffizienten, durch Koeffizienten zur Bestimmung der Temperaturabhängigkeit und die dazugehörigen Alpha- bzw. Gamma-Faktoren beschrieben. Die Kennungen für kinetisch kontrollierte Reaktionen sind im Kapitel Allgemeines beschrieben.

Argumente	Format
Satzkennung = Reaktionstyp	A4, 1X
Reaktionsnummer	I4,1X
Reaktionsenthalpie	D16.8

Folgesätze zur Angabe der stöchiometrischen Koeffizienten:

Argumente	Format
Satzkennung = STOE	A4, 1X
Komponentennummer	I4,1X
Stöchiometrischer Koeffizient	D16.8

Folgesätze zur Beschreibung der Temperaturabhängigkeit mit den Kennungen:

F(T)

-F(T)

PHI

Es sind maximal 6 PHI- und 6 F(T)-Funktionen vorgesehen.

Argumente	Format
Satzkennung	A4, 1X
Label	A4,1X
Koeffizienten	4D16.8

Folgesätze zur Eingabe der zu den Funktionen gehörenden Alpha- bzw. Gamma-Werte:

Argumente	Format
Satzkennung	A4, 1X
Label	A4,1X
Komponentennummer	I4, 1X
Wert	D16.8

3.7.3 Reaktionen mit vorgegebenem Umsatz

Die Kennungen für den Reaktionstyp sind CONV bzw. STAT.

Argumente	Format
Satzkennung = Reaktionstyp	A4, 1X
Reaktionsnummer	I4,1X
Reaktionsenthalpie	D16.8
Umsatz	D16.8
Referenzkomponente	I4

3.7.4 Reaktionen mit vorgegebener Reaktionskoordinate

Argumente	Format
Satzkennung = Reaktionstyp	A4, 1X
Reaktionsnummer	I4,1X
Reaktionsenthalpie	D16.8
Zeta	D16.8

3.8 Mittelwerte

Die Kennungen für die Mittelwertbildung sind im Kapitel Allgemeines beschrieben.

Hier ist auch angegeben, welche Standardwerte für welchen Stoffdatentyp gültig sind.

Argumente	Format
Satzkennung = AVER	A4, 1X
Mittelwertkennung	A4,1X
Stoffdatentyp	A4,1X

3.9 Inkremente

Z. Zt. steht nur UNIFAC als Inkrementenmethode zur Verfügung. Die Eingabe hierfür gliedert sich in 2 Teile:

1. Reinstoffinformationen
2. Gemischinformationen

Die Daten der Reinstoffinformationen werden in einer Folge von Sätzen beschrieben. Der erste Satz ist wie folgt aufgebaut:

Argumente	Format
Satzkennung = INKR	A4, 1X
Komponentennummer	I4,1X
Inkrementtyp	A4, 1X
Anzahl Inkremente	I4, 1X

Es folgen nun Sätze, die jedes Inkrement dieser Komponente beschreiben:

Argumente	Format
Inkrementnummer	I4, 1X
Häufigkeit dieses Inkrements	I4,1X
Hauptgruppennummer	I4, 1X
R-Parameter dieses Inkrements	D16.8
Q-Parameter dieses Inkrements	D16.8

Die Daten der Gemischinformation sind wie folgt aufgebaut:

Argumente	Format
Satzkennung = MIXT	A4, 1X
Inkrementtyp	A4,1X
Anzahl Wechselwirkungsparameterpaare	I4

Für jedes Binärpaar an Wechselwirkungsparametern folgt nun eine Zeile:

Argumente	Format
Hauptgruppennummer i	I4, 1X
Hauptgruppennummer j	I4,1X
$W(i, j)$	D16.8
$W(j, i)$	D16.8

3.10 Phasenschlüssel

Die Phasenzugehörigkeit der Komponenten wird in mehreren Datensätzen je Phase eingegeben. Reicht die Eingabebreite nicht aus, kann für die entsprechende Phase eine oder mehrere weitere Datenzeilen eingegeben werden.

Argumente	Format
Satzkennung = PH_K	A4, 1X
Phasenkennung	A4, 1X
Komponentennummern	14(I4,1X)

3.11 Kommentare

Um die Übersichtlichkeit der Übertragungsdatei zu erhöhen, wurden Kommentarzeilen zugelassen.

Argumente	Format
Satzkennung = COMM	A4, 1X
Text	A75

4 Beschreibung der Schnittstellenprogramme

4.1 Beschreibung der Darstellung

Alle Programme der Eingabeschnittstelle sind von Fortran aus als Subroutine aufzurufen.

Im folgenden Teil werden die Aufrufe und der Zweck aller Schnittstellenprogramme im einzelnen erklärt. Die Übersicht zu jedem Programm ist wie folgt zu lesen:

- Der 1. Abschnitt zeigt den Programmnamen und eine kurze Übersicht über die Programmfunktion.
- Unter FORMAT folgt eine Darstellung des Aufrufformates mit der Erklärung der Variablen und des Zugriffes im Unterprogramm. Zugriff Lesen bedeutet, daß die Variable nur gelesen wird; Schreiben bedeutet, daß die Variable geschrieben wird; Ändern bedeutet, daß der Wert der Variablen verändert wird; Schreiben/Lesen bedeutet, daß je nach Aufgabenstellung die Variable entweder gelesen oder geschrieben wird.
- Im Abschnitt ARGUMENTE werden Funktion und mögliche Inhalte der einzelnen Parameter erklärt.
- Im Teil FUNKTION wird die Arbeitsweise des Programms näher erläutert und Hinweise zur Benutzung gegeben.

4.2 Einspeichern der Basisdaten

T_LOAD

Speichern der Daten aus dem Stoffdaten-File in die interne Struktur

FORMAT T_LOAD(CHANNEL, RVAL)

Name	Typ	Zugriff
CHANNEL	I4	Lesen
RVAL	I4	Schreiben

ARGUMENTE **CHANNEL**

Kanalnummer, unter der die einzulesende Datei geöffnet ist

RVAL

Fehlerkennung

= 0 : alles ok.

= 1: Speicherplatz erschöpft, Datei nicht komplett eingelesen

= 10: Reihenfolge der Steuersätze nicht ok.

= 20: Stoffsystemreihenfolge nicht möglich

= 30: nicht identifizierbarer Datentyp

= 40: maximale Inkrementzahl überschritten

< 0 : Dateilesefehler, RVAL = negativer IOSTAT-Wert

FUNKTION

Das Stoffdaten-File muß bei Aufruf des Unterprogramms T_LOAD bereits geöffnet sein und wird von diesem auch nicht geschlossen.

Alle Daten der Übertragungsdatei werden mit Fortran-Readbefehlen eingelesen und in die internen Strukturen für die Schnittstellenprogramme gespeichert.

Das zuerst eingelesene Stoffsystem wird automatisch als aktives Stoffsystem gespeichert, so daß es den Berechnungsprogrammen ohne explizite Aktivierung zur Verfügung steht.

4.3 Auslesen von Daten aus der internen Struktur

Die Schnittstellenprogramme, die mit dem Prefix "T_G_" beginnen, liefern Informationen aus der internen Datenstruktur- z.B. Anzahl der Komponenten, Namen von Komponenten, Koeffizientensätze. Diese Informationen beziehen sich immer auf das jeweils aktive Stoffsystem.

Diese Zugriffstechnik macht den Anwender unabhängig von der internen Datenstruktur.

T_G_AVER

Holen der Methode der Mittelwertbildung

FORMAT

T_G_AVER(PROPTYP, AVERTYP, RVAL)

Name	Typ	Zugriff
PROPTYP	C	Lesen
AVERTYP	C	Schreiben
RVAL	I4	Schreiben

ARGUMENTE***PROPTYP***

Stoffdatentyp

AVERTYP

Typ der Mittelwertbildung

RVAL

Fehlerkennung

= 0 : alles ok.

=-1 : ungültiger Stoffdatentyp

=-2 : Stoffdatentyp nicht gespeichert

FUNKTION

Der Typ der Mittelwertbildung ist als Kenngröße für jeden Reinstoffdatentyp gespeichert. Die Länge des Characterstrings AVERTYP muß mindestens 4 sein, um alle Kennungen eindeutig umspeichern zu können.

T_G_BINA_COEFF

Koeffizienten für binäre Stoff-Funktionen holen

FORMAT T_G_BINA_COEFF(PROPTYP, COMP_I, COMP_J, EQUATION,
COEFF_NO, SOURCE, COEFFICIENTS, RVAL)

Name	Typ	Zugriff
PROPTYP	C	Lesen
COMP_I	I4	Lesen
COMP_J	I4	Lesen
EQUATION	C	Schreiben
COEFF_NO	I4	Schreiben
SOURCE	C	Schreiben
COEFFICIENTS	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE***PROPTYP***

Stoffdatentyp

COMP_I

Nummer der gewünschten 1. Komponente

COMP_J

Nummer der gewünschten 2. Komponente

EQUATION

Kennung des Gleichungstyps

COEFF_NO

Anzahl der gespeicherten Koeffizienten

SOURCE

Quelleninformation

COEFFICIENTS

Vektor der gespeicherten Koeffizienten; gefüllt von 1 bis COEFF_NO

RVAL

Fehlerkennung

= 0 : alles ok.

=-1 : Stoffdatum nicht gespeichert

=-2 : ungültige Komponentenummer

T_G_BINA_COEFF

=-3 : Stoffdatentyp nicht gültig

> 0 : Stoffdatum für diese Kombination nicht gespeichert

FUNKTION

Die Länge der Characterstrings EQUATION und SOURCE muß mindestens 4 sein, um alle Kennungen eindeutig speichern zu können.

T_G_BINA_COEFF arbeitet z.Zt. nur für symmetrische Matrizen, so daß erwartet wird, daß COMP_I < COMP_J ist.

T_G_CHEMICS

Holen der Namen und der Reaktionszahl von gespeicherten Chemieblöcken

FORMAT T_G_CHEMICS(TASK, LABEL, NO_REAC, RVAL)

Name	Typ	Zugriff
TASK	I4	Lesen
LABEL	C	Lesen/Schreiben
NO_REAC	I4	Schreiben
RVAL	I4	Schreiben

ARGUMENTE***TASK***

Aufgabenstellung

=1: Hole 1. gespeicherten Labelnamen mit Reaktionszahl

=2: Hole nächsten gespeicherten Labelnamen mit Reaktionszahl

=3: Hole Reaktionszahl zu vorgegebenem Label

LABEL

Label des Chemieblockes

Output für Task = 1, 2

Input für TASK = 3

NO_REAC

Anzahl Reaktionen dieses Chemieblockes

RVAL

Fehlerkennung

= 0 : alles ok.

= 1 : kein Chemielabel

=-1 : bei TASK = 2: kein weiterer Label gefunden

bei TASK = 3 : Label nicht gefunden

=-2 : ungültige TASK

FUNKTION

Mit TASK=2 kann das Programm erst nach einem erfolgreichen Aufruf mit TASK=1 angesprochen werden.

T_G_CH_ALLG

Holen der Zusatzinformationen für allgemeine Reaktionstypen für eine Reaktion in einem Chemieblock

FORMAT T_G_CH_ALLG(LABEL, NUMBER, VALUE, COMP_NR, RVAL)

Name	Typ	Zugriff
LABEL	C	Lesen
NUMBER	I4	Lesen
VALUE	R8	Schreiben
COMP_NR	I4	Schreiben
RVAL	I4	Schreiben

ARGUMENTE***LABEL***

Label des Chemieblockes

NUMBER

Nummer der Reaktion

VALUE

gewünschter Wert

COMP_NR

Komponentennummer

RVAL

Fehlerkennung

= 0 : alles ok.

= 1 : Label nicht gefunden

=-1 : ungültiger Reaktionstyp

FUNKTION

Für Reaktionen vom Typ COOR wird in VALUE die gewünschte Reaktionskoordinate zurückgegeben. Der Parameter COMP_NR wird in diesem Fall nicht benutzt.

Für Reaktionen vom Typ CONV bzw. STATE wird in VALUE der gewünschte Umsatz und in COMP_NR die Nummer der Referenzkomponente zurückgegeben.

T_G_CH_EQ

Holen der Zusatzinformationen für EQ-Reaktionen: Koeffizienten und Anzahl Koeffizienten der F(T)-Funktion

FORMAT T_G_CH_EQ(LABEL, NUMBER, COEFF_NO, COEFFICIENTS, RVAL)

Name	Typ	Zugriff
LABEL	C	Lesen
NUMBER	I4	Schreiben
COEFF_NO	I4	Schreiben
COEFFICIENTS	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE***LABEL***

Label des Chemieblockes

Output für Task = 1, 2

Input für TASK = 3

NUMBER

Nummer der Reaktion

COEFF_NO

Anzahl Koeffizienten der F(T)-Funktion

COEFFICIENTS

Koeffizienten der F(T)-Funktion; gefüllt von 1 bis COEFF_NO

RVAL

Fehlerkennung

= 0 : alles ok.

= 1 : Label nicht gefunden

=-1 : ungültiger Reaktionstyp

FUNKTION

Je Reaktion vom Typ EQ ist eine F(T)-Funktion zur Beschreibung der Gleichgewichtskonstanten gespeichert.

T_G_CH_KI

Holen der Zusatzinformationen für KI-Reaktionen für eine Reaktion in einem Chemieblock

FORMAT

T_G_CH_KI(TASK, LABEL, NUMBER, FUNC_LABEL, FAK,
FUNCTION, VEKTOR, RVAL)

Name	Typ	Zugriff
TASK	I4	Lesen
LABEL	C	Lesen
NUMBER	I4	Lesen
FUNC_LABEL	C	Schreiben
FAK	R8	Schreiben
FUNCTION	R8(*)	Schreiben
VEKTOR	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**TASK**

Aufgabenstellung

=1: Hole 1. gespeicherte F(T)-Funktion mit dazugehörigen Alpha-Werten

=2: Hole nächste gespeicherte F(T)-Funktion mit dazugehörigen Alpha-Werten

=3: Hole 1. gespeicherte Phi-Funktion mit dazugehörigen Gamma-Werten

=4: Hole nächste gespeicherte Phi-Funktion mit dazugehörigen Gamma-Werten

LABEL

Label des Chemieblockes

NUMBER

Nummer der Reaktion dieses Chemieblockes

FUNC_LABEL

Label der gespeicherten Funktion

FAK

bei TASK=1, 2: Vorzeichenfaktor (1.D0 bei F(T), -1.D0 bei -F(T))

FUNCTION

Koeffizienten der Funktion; Länge des Vektors = 4

VEKTOR

Vektor der Alpha- bzw. Gamma-Werte ; Länge = Anzahl Komponenten des aktuellen Stoffsystems

RVAL

Fehlerkennung

= 0 : alles ok.

= 1 : Label nicht gefunden

=-1 : ungültiger Reaktionstyp

=-2 : ungültige Aufrufreihenfolge

=-3 : keine weitere Funktion gefunden

FUNKTION

Je KI-Reaktion können bis zu 6 F(T)- bzw. PHI-Funktionen mit den dazugehörigen Alpha- bzw. Gamma-Werten gespeichert sein.

T_G_CH_PHASE

Holen der Phasenkenennung für eine Reaktion im gerade aktiven Chemieblock

FORMAT

T_G_CH_PHASE(NUMBER, PHASE, RVAL)

Name	Typ	Zugriff
NUMBER	I4	Lesen
PHASE	C	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**NUMBER**

Nummer der Reaktion dieses Chemieblockes

PHASE

Reaktionphase

RVAL

Fehlerkennung

= 0 : alles ok.

= 1 : Label nicht gefunden

FUNKTION

Die für die Reaktion mit der angegebenen Nummer im gerade aktiven Chemieblock gespeicherte Reaktionsphase ('LIQU' oder 'VAPO') wird zurückgegeben.

T_G_CH_TYPE

Holen des Reaktionstyps und der Reaktionsphase für eine Reaktion in einem Chemieblock

FORMAT

T_G_CH_TYPE(LABEL, NUMBER, TYPE, PHASE, RVAL)

Name	Typ	Zugriff
LABEL	C	Lesen/Schreiben
NUMBER	I4	Lesen
TYPE	C	Schreiben
PHASE	C	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**LABEL**

Label des Chemieblockes

NUMBER

Nummer der Reaktion dieses Chemieblockes

TYPE

Reaktionstyp

PHASE

Reaktionsphase

RVAL

Fehlerkennung

= 0 : alles ok.

= 1 : Label nicht gefunden

FUNKTION

Für jede Reaktion sind der Reaktionstyp und die Reaktionsphase ('LIQU' oder 'VAPO') gespeichert.

Die Reaktionstypen gliedern sich in 3 Gruppen:

- 1.: Gleichgewichtsreaktionen mit der Typanfängskennung 'EQ'
- 2.: kinetische Reaktionen mit der Typanfängskennung 'KI'
- 3.: allgemeine Reaktionstypen wie 'CONV', 'STAT', 'COOR'

T_G_COMP_NUMBER

Holen der maximalen bzw. aktuellen Komponentenzahl

FORMAT T_G_COMP_NUMBER(TASK, NUMBER, RVAL)

Name	Typ	Zugriff
TASK	I4	Lesen
NUMBER	I4	Schreiben
RVAL	I4	Schreiben

ARGUMENTE***TASK***

Aufgabenstellung

= 1 : holen der maximalen Komponentenzahl

= 2 : holen der aktuellen Komponentenzahl

NUMBER

gewünschte Komponentenzahl

RVAL

Fehlerkennung

= 0 : alles ok.

= 1 : maximale Komponentenzahl nicht gesetzt

= 2 : aktuelle Komponentenzahl nicht gesetzt

=-1 : ungültige TASK

FUNKTION

Die Komponentenzahlen werden aus den internen Speicherfeldern geholt und dem aufrufenden Programm zur Verfügung gestellt.

T_G_ENTH

Holen der Daten für Enthalpieberechnungen

FORMAT T_G_ENTH(TASK, LABEL, EXCESS, ISOT, PHASE, TREF, HREF, TPHASE, RVAL)

Name	Typ	Zugriff
TASK	I4	Lesen
LABEL	C	Lesen/Schreiben
EXCESS	C	Schreiben
ISOT	C	Schreiben
PHASE	C(*)	Schreiben
TREF	R8(*)	Schreiben
HREF	R8(*)	Schreiben
TPHASE	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**TASK**

Aufgabenstellung

= 1 : Hole 1. Label mit Aufschlüsselung

= 2 : Hole nächsten Label mit Aufschlüsselung

= 3 : Hole Aufschlüsselung zu vorgegebenem Label

= 4 : Hole Aufschlüsselung zu vorgegebenem Label einschließlich Hilfsvektor für Phasenumwandlung

LABEL

Labelname

Output bei TASK = 1, 2

Input bei TASK = 3, 4

EXCESS

Methodenkennung zur Berechnung der Exzeßenthalpie

ISOT

Methodenkennung zur Berechnung der isothermen Druckabhängigkeit im Gas

PHASE

Kennzeichnungen der Phase am Referenzpunkt

TREF

Temperaturen am Referenzpunkt

HREF

Enthalpien am Referenzpunkt

TPHASE

Phasenumwandlungstemperaturen

Ist keine Phasenumwandlungstemperatur angegeben, wird das entsprechende Vektorelement auf -1.D30 gesetzt.

Für TASK=3 ist die Vektorlänge = Anzahl Komponenten; für TASK=4 ist die Vektorlänge = 2* Anzahl Komponenten

RVAL

Fehlerkennung

= 0 : alles ok.

= 1 : kein Enthalpielabel

= 2 : keine Phasenumwandlungstemperaturen gegeben

=-1 : bei TASK=3, 4 : Label nicht gefunden

bei TASK=2: kein weiterer Label vorhanden

=-2 : fehlerhafte TASK

FUNKTION

Mit TASK=2 kann das Programm erst nach einem erfolgreichen Aufruf mit TASK=1 angesprochen werden.

T_G_IDEN_COMP

Holen der Stoffidentifikationsdaten

- Name
- Langname
- Cas-Nummer

FORMAT T_G_IDEN_COMP(TASK, COMP_NR, NAME, RVAL)

Name	Typ	Zugriff
TASK	I4	Lesen
COMP_NR	I4	Lesen
NAME	C	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**TASK**

Aufgabenstellung

- = 1 : holen des Namens
- = 2 : holen des Langnamens
- = 3 : holen der Cas-Nummer

COMP_NR

Komponentennummer

- > 0 : Nummer der gewünschten Komponente
- = 0 : alle Komponenten

NAME

je nach TASK gefüllte Ausgabegröße

RVAL

Fehlererkennung

- = 0 : alles ok.
- = 1 : Daten nicht gespeichert
- = 2 : ungültige Komponentennummer
- = 4 : Länge des Ausgabefeldes zu klein
- =-1 : ungültige TASK

FUNKTION

Die Stoffidentifikationsdaten (Name, Langname, CAS-Nummer) werden aus den internen Speicherfeldern geholt und dem aufrufenden Programm zur Verfügung gestellt.

Für den Fall COMP_NR = 0 werden die gewünschten Identifikationsdaten für alle Komponenten im Vektor NAME zurückgegeben. Die Dimension von NAME muß also mindestens der aktuellen Komponentenzahl entsprechen. Für den Fall COMP_NR > 0 wird eine einzelne Date zurückgegeben.

Ist der Characterstring zu kurz für die gespeicherten Daten, wird RVAL=4 gesetzt; der gespeicherte String wird beim Umspeichern abgeschnitten.

T_G_IDEN_COMP

Ist der Characterstring länger als die gespeicherten Daten, wird mit Blanks aufgefüllt.

T_G_INFO_COMP

Holen der Praesenzinformationen der Komponenten in den verschiedenen Phasen

FORMAT T_G_INFO_COMP(PHASE, ANZAHL, VECTOR, RVAL)

Name	Typ	Zugriff
PHASE	C	Lesen
ANZAHL	I4	Schreiben
VECTOR	I4	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**PHASE**

Kennung der Phase, für die die Information abgefragt werden soll

- = LIQU : Flüssigkeit bzw. 1. flüssige Phase
- = LIQ2 : 2. flüssige Phase
- = VAPO: Dampf
- = SOLI : Feststoff

ANZAHL

Anzahl Komponenten, die in dieser Phase vorkommen

VECTOR

Vektor mit Komponentennummern, gefüllt von 1 bis Anzahl

RVAL

Fehlerkennung

- = 0 : alles ok.
- = 1 : Daten nicht gespeichert
- = 2 : keine Komponente in dieser Phase
- =-1 : Phase nicht gültig

FUNKTION

Die Praesenzinformationen werden aus den internen Speicherfeldern geholt und dem aufrufenden Programm zur Verfügung gestellt.

Ist der entsprechende Zeiger nicht gesetzt, wird das Programm mit RVAL=1 verlassen.

Ist keine Komponente in der entsprechenden Phase vorgesehen, ist die zurückgegebene ANZAHL = 0.

T_G_INKR

Holen von Inkrementinformationen

FORMAT T_G_INKR(INKTYP, COMP_NR, ANZINKR, INKR_NO, NUMBER, RVAL)

Name	Typ	Zugriff
INKTYP	C	Lesen
COMP_NR	I4	Lesen
ANZINKR	I4	Schreiben
INKR_NO	I4(*)	Schreiben
NUMBER	I4(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**INKTYP**

Inkrementtyp

COMP_NR

Komponentennummer

ANZINKR

Anzahl der für diese Komponente gespeicherten Inkremente

INKR_NO

Vektor mit den Inkrementnummern; gefüllt von 1 bis ANZINKR

NUMBER

Vektor mit der Häufigkeitsangabe für jedes Inkrement; gefüllt von 1 bis ANZINKR

RVAL

Fehlerkennung

= 0 : alles ok.

= 1 : Daten nicht gespeichert

=-1 : ungültiger Inkrementtyp

=-2 : ungültige Komponentennummer

FUNKTION

T_G_INKR holt für den angegebenen Inkrementtyp und die angegebene Komponente die Anzahl der für diese Komponente gespeicherten Inkremente, die gespeicherten Inkrementnummern und die Häufigkeitsangabe aus den internen Speicherfeldern.

Die Übersetzung der Inkrementnummern in Characterstrings, die die Inkremente identifizieren, muß im Anwendungsprogramm anhand der aktuell für diesen Inkrementtyp hinterlegten Datei erfolgen.

T_G_LABEL_ENTH

Holen der Namen von gespeicherten Enthalpieberechnungslabeln

FORMAT T_G_LABEL_ENTH(TASK, LABEL, RVAL)

Name	Typ	Zugriff
TASK	I4	Lesen
LABEL	C	Schreiben
RVAL	I4	Schreiben

ARGUMENTE

TASK

Aufgabenstellung

= 1 : Hole 1. gespeicherten Labelnamen

= 2 : Hole nächsten gespeicherten Labelnamen

LABEL

Label der Enthalpiebeschreibung

RVAL

Fehlererkennung

= 0 : alles ok.

= 1 : kein Enthalpielabel

=-1 : Bei TASK=2: kein weiterer Label gefunden

=-2 : ungültige TASK

FUNKTION

Mit TASK=2 kann das Programm erst nach einem erfolgreichen Aufruf mit TASK=1 angesprochen werden.

T_G_LLEQ

Methoden für Flüssig-flüssig-Gleichgewichtsberechnungen holen

FORMAT

T_G_LLEQ(TASK, LABEL, METHODEN, RVAL)

Name	Typ	Zugriff
TASK	I4	Lesen
LABEL	C	Lesen/Schreiben
METHODEN	C(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**TASK**

Aufgabenstellung

= 1 : Hole 1. Label mit Aufschlüsselung

= 2 : Hole nächsten Label mit Aufschlüsselung

= 3 : Hole Aufschlüsselung zu vorgegebenem Label

LABEL

Label der Phasengleichgewichtsbeschreibung

Output für Task = 1, 2

Input für Task = 3

METHODEN

Methodenkennungen für Flüssig-flüssig-Phasengleichgewichtsbeschreibungen

(1) = Aktivitätskoeffizienten

RVAL

Fehlerkennung

= 0 : alles ok.

= 1 : kein Label gespeichert

= -1: bei TASK = 3: Label nicht gefunden

bei TASK = 2: kein weiterer Label vorhanden

= -2: fehlerhafte TASK

FUNKTION

Mit TASK = 2 kann das Unterprogramm erst nach einem erfolgreichen Aufruf mit TASK = 1 aufgerufen werden.

Die Fehlermarke RVAL= 1 wird nicht nur für den Fall gesetzt, daß kein Phasengleichgewichtslabel gespeichert ist, sondern auch für den Fall, daß alle Methodenkennungen auf "NONE" gesetzt sind.

T_G_LVEQ

Methoden für Flüssig-Dampf-Gleichgewichte holen

FORMAT T_G_LVEQ(TASK, LABEL, METHODEN, RVAL)

Name	Typ	Zugriff
TASK	I4	Lesen
LABEL	C	Lesen/Schreiben
METHODEN	C(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**TASK**

Aufgabenstellung

= 1 : Hole 1. Label mit Aufschlüsselung

= 2 : Hole nächsten Label mit Aufschlüsselung

= 3 : Hole Aufschlüsselung zu vorgegebenem Label

LABEL

Label der Phasengleichgewichtsbeschreibung

Output für Task = 1, 2

Input für Task = 3

METHODEN

Methodenkennungen für Flüssig-Dampf-Phasengleichgewichtsbeschreibungen

(1) = Aktivitätskoeffizienten

(2) = Fugazitätskoeffizienten

(3) = Kennung für Poynting-Korrektur

(4) = Kennung für Inert-Behandlung

(5) = Zustandsgleichungen

RVAL

Fehlerkennung

= 0 : alles ok.

= 1 : kein Label gespeichert

= -1: bei TASK = 3: Label nicht gefunden

bei TASK = 2: kein weiterer Label vorhanden

= -2: fehlerhafte TASK

FUNKTION

Mit TASK = 2 kann das Unterprogramm erst nach einem erfolgreichen Aufruf mit TASK = 1 aufgerufen werden.

Die Fehlermarke RVAL= 1 wird nicht nur für den Fall gesetzt, daß kein Phasengleichgewichtslabel gespeichert ist, sondern auch für den Fall, daß alle Methodenkennungen auf "NONE" gesetzt sind.

T_G_MATRIX

Holen von Matrixelementen

FORMAT T_G_MATRIX(PROPTYP, PROPSPEZ, COMP_I, COMP_J, ELEMENTS, RVAL)

Name	Typ	Zugriff
PROPTYP	C	Lesen
PROPSPEZ	C	Lesen
COMP_I	I4	Lesen
COMP_J	I4	Lesen
ELEMENTS	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE **PROPTYP**

Matrixtyp

PROPSPEZ

spezieller Matrixtyp; muß angegeben werden, wenn es zu einem Typ mehrere Matrizen gibt

COMP_I

Nummer der ersten Komponente

COMP_J

Nummer der zweiten Komponente

ELEMENTS

Vektor mit den gewünschten Matrixelementen

RVAL

Fehlerkennung

= 0 : alles ok.

= 1 : Datentyp nicht gespeichert

= -1: angegebener Matrixtyp fehlerhaft

= -2: angegebene Komponentennummer fehlerhaft

FUNKTION

Sind COMP_I und COMP_J explizit mit einer Komponentennummer belegt, werden die Matrixelemente $W(i,j)$ und $W(j,i)$ zurückgegeben. Die Dimension des Vektors ELEMENTS muß also ≥ 2 sein.

Wird COMP_I oder COMP_J gleich 0 eingegeben, wird die entsprechende Spalte bzw. Zeile der Matrix zurückgegeben. Die Dimension des Vektors ELEMENTS muß also mindestens Anzahl Komponenten sein.

T_G_NAME_LENGTH

Holen der Anzahl Zeichen der Komponentenidentifikationsdaten

FORMAT T_G_NAME_LENGTH(TASK, NUMBER, RVAL)

Name	Typ	Zugriff
TASK	I4	Lesen
NUMBER	I4	Schreiben
RVAL	I4	Schreiben

ARGUMENTE

TASK

Aufgabenstellung

= 1: holen der Anzahl Zeichen des Namens

= 2: holen der Anzahl Zeichen des Langnamens

= 3: holen der Anzahl Zeichen der Cas-Nummer

NUMBER

Anzahl Zeichen je nach TASK

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Zeichenzahl = 0

=-1: ungültige TASK

FUNKTION

Die Anzahl Zeichen der gewünschten Identifikationsdaten wird aus dem internen Speicher geholt und in die Ausgabegröße gespeichert.

T_G_NO_REAC

Holen der maximalen Anzahl Reaktionen, die in einem Chemieblock definiert sein dürfen.

FORMAT T_G_NO_REAC (NUMBER, RVAL)

Name	Typ	Zugriff
NUMBER	I4	Schreiben
RVAL	I4	Schreiben

ARGUMENTE **NUMBER**

maximale Anzahl Reaktionen

RVAL

Fehlerkennung

= 0 : alles ok.

FUNKTION Die maximale Anzahl Reaktionen wird aus den internen Speicherfeldern geholt und dem aufrufenden Programm zur Verfügung gestellt.

T_G_PARAM

Holen von Stoffdaten-Parametern

FORMAT T_G_PARAM(PROPTYP, COMP_NR, ELEMENTS, NELEM, RVAL)

Name	Typ	Zugriff
PROPTYP	C	Lesen
COMP_NR	I4	Lesen
ELEMENTS	R8(*)	Schreiben
NELEM	I4	Schreiben
RVAL	I4	Schreiben

ARGUMENTE **PROPTYP**

Stoffdatentyp

COMP_NR

Komponentennummer

ELEMENTS

Vektor der gespeicherten Elemente

NELEM

Anzahl der Elemente

RVAL

Fehlerkennung

= 0 : alles ok.

=-1 : Stoffdatentyp nicht gültig

=-2 : Stoffdatum für alle Komponenten nicht gespeichert

=-3 : ungültige Komponentennummer

FUNKTION Für die Komponente COMP_NR werden aus dem internen Speicher die Elemente aller Parametervektoren zurückgegeben.

T_G_PURE_COEFF

Koeffizienten für Reinstoff-Funktionen holen

FORMAT T_G_PURE_COEFF(PROPTYP, COMP_NR, EQUATION,
COEFF_NO, SOURCE, COEFFICIENTS,
TMIN, TMAX, RVAL)

Name	Typ	Zugriff
PROPTYP	C	Lesen
COMP_NR	I4	Lesen
EQUATION	C	Schreiben
COEFF_NO	I4	Schreiben
SOURCE	C	Schreiben
COEFFICIENTS	R8(*)	Schreiben
TMIN	R8	Schreiben
TMAX	R8	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**PROPTYP**

Stoffdatentyp

COMP_NR

Komponentennummer

EQUATION

Kennung des Gleichungstyps

COEFF_NO

Anzahl der gespeicherten Koeffizienten

SOURCE

Quelleninformation

COEFFICIENTS

Vektor der gespeicherten Koeffizienten; gefüllt von 1 bis COEFF_NO

TMIN

untere Temperaturgrenze des Gültigkeitsbereichs der Koeffizienten

TMAX

obere Temperaturgrenze des Gültigkeitsbereichs der Koeffizienten

RVAL

Fehlerkennung

= 0 : alles ok.

> 0 : Stoffdatum für diese Komponente nicht gespeichert

=-1 : Stoffdatum für alle Komponenten nicht gespeichert

=-2 : ungültige Komponentenummer

=-3 : Stoffdatentyp nicht gültig

FUNKTION

Die Länge der Characterstrings EQUATION und SOURCE muß mindestens 4 sein, um alle Kennungen eindeutig speichern zu können.

T_G_PURE_EXTR

Holen der Extrapolationskennung für Stoff-Funktionen

FORMAT T_G_PURE_EXTR(PROPTYP, COMP_NR, EXTKENN, RVAL)

Name	Typ	Zugriff
PROPTYP	C	Lesen
COMP_NR	I4	Lesen
EXTKENN	I	Schreiben
RVAL	I4	Schreiben

ARGUMENTE **PROPTYP**

Stoffkonstantenkennung

COMP_NR

Komponentennummer

EXTKENN

Extrapolationskennung

=0: keine spezielle Extrapolationsfunktion

=1: spezielle Extrapolationsfunktion

RVAL

Fehlerkennung

= 0 : alles ok.

=-1 : Stoffdatum nicht gespeichert

=-2 : ungültige Komponentennummer

=-3 : Stoffdatentyp nicht gültig

> 0 : Stoffdatum für diese Komponente nicht gespeichert

FUNKTION

T_G_PURE_EXTR ruft das UP T_ASK_PURE auf, das die Zeigernummer des Stoffdatentyps im Common /PROP_ZEIGER/ zurückgibt. Die Position der zurückzugebenden Date wird bestimmt und die Extrapolationskennung umgespeichert.

T_G_REACTION

Holen der allgemeinen, vom Reaktionstyp unabhängigen, Daten für eine Reaktion eines Chemieblockes

FORMAT T_G_REACTION(LABEL, NUMBER, TYPE, PHASE, DHR, STOE, RVAL)

Name	Typ	Zugriff
LABEL	C	Lesen/Schreiben
NUMBER	I4	Lesen
TYPE	C	Schreiben
PHASE	C	Schreiben
DHR	R8	Schreiben
STOE	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**LABEL**

Labelname

NUMBER

Nummer der Reaktion

TYPE

Kennzeichnung des Reaktionstyps

PHASE

Phase, in der die Reaktion abläuft

'LIQU' : Reaktion in der Flüssigkeit

'VAPO' : Reaktion in der Dampfphase

DHR

Reaktionswärme

STOE

Stoichiometrische Koeffizienten

RVAL

Fehlerkennung

= 0 : alles ok.

= 1 : Label nicht gefunden

= 2 : Reaktionsnummer fehlerhaft

FUNKTION

Der Vektor STOE muß mindestens die Länge Anzahl Komponenten des gerade aktiven Stoffsystems haben. Die stöchiometrischen Koeffizienten wegreakierender Koeffizienten werden negativ, die entstehender Komponenten positiv angegeben. Nicht beteiligte Komponenten haben den Wert 0.

T_G_SINGLE_COMP

Holen von Stoffkonstanten

FORMAT T_G_SINGLE_COMP(PROPTYP, COMP_NR, SOURCE, VALUE, RVAL)

Name	Typ	Zugriff
PROPTYP	C	Lesen
COMP_NR	I4	Lesen
SOURCE	C	Schreiben
VALUE	R8	Schreiben
RVAL	I4	Schreiben

ARGUMENTE **PROPTYP**

Stoffdatentyp

COMP_NR

Komponentennummer

= 0: für alle Komponenten

> 0: nur für die angegebene Komponente

SOURCE

Quelleninformation(en)

VALUE

Stoffkonstante(n)

RVAL

Fehlerkennung

= 0: alles ok.

> 0: Stoffdatum für diese Komponente nicht gespeichert (bei COMP_NR = 0)

= 1: Stoffdatum für diese Komponente nicht gespeichert (bei COMP_NR > 0)

=-1: Stofftyp nicht gespeichert

=-2: ungültige Komponentennummer

=-3: ungültiger Stoffdatentyp

FUNKTION

Wird der Parameter COMP_NR > 0 eingegeben, so werden in SOURCE und in VALUE die Daten für diese Komponente zurückgegeben

Wird der Parameter COMP_NR = 0 eingegeben, so werden in SOURCE und in VALUE die Daten für alle Komponenten zurückgegeben. Die Dimension der beiden Parameter muß in diesem Fall mindestens gleich der aktuellen Komponentenzahl sein.

T_G_SYSTEM_NAME

Namen und Versionen von Stoffsystemen holen

FORMAT T_G_SYSTEM_NAME(TASK, LABEL, VERS, RVAL)

Name	Typ	Zugriff
TASK	I4	Lesen
LABEL	C	Schreiben
VERS	I4	Schreiben
RVAL	I4	Schreiben

ARGUMENTE***TASK***

Aufgabenstellung

= 1: Name und Version des gerade aktiven Stoffsystems holen

= 2: Name und Version des ersten gespeicherten Stoffsystems holen

= 3: Name und Version des nächsten Stoffsystems holen

LABEL

Name des Stoffsystems

VERS

Versionsnummer des Stoffsystems

RVAL

Fehlerkennung

= 0: alles ok.

=-1: ungültige TASK

= 1: bei TASK=3: kein weiteres System gespeichert

bei TASK=2: kein System gespeichert

= 2: bei TASK=3: noch kein Aufruf mit TASK = 2

FUNKTION

Die Länge von LABEL muß mindestens 4 sein, um die gespeicherten Namen eindeutig aufnehmen zu können.

4.4 Methoden in der internen Datenstruktur setzen

Die Schnittstellenprogramme, die mit dem Prefix "T_S_" beginnen, aktivieren Methoden und Datensätze, auf die die interne Datenstruktur durch das Einlesen des Datenfiles vorbereitet worden ist - z.B. Wechsel zu einem anderen Stoffsystem, Aktivieren einer anderen Methode des Phasengleichgewichts.

T_S_CHEMICS

Aktivieren eines Chemieblockes

FORMAT T_S_CHEMICS(LABEL, RVAL)

Name	Typ	Zugriff
LABEL	C	Lesen
RVAL	I4	Schreiben

ARGUMENTE **LABEL**

Label des Chemieblockes

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Label nicht gefunden

FUNKTION

T_S_CHEMICS kennzeichnet für die Programme, die Reaktionen berechnen, den Chemieblock, in dem die beim nächsten Aufruf zu berechnende Reaktion gespeichert ist.

T_S_ENTH

Setzen der Methoden zur Berechnung der Enthalpie

FORMAT T_S_ENTH(LABEL, RVAL)

Name	Typ	Zugriff
LABEL	C	Lesen
RVAL	I4	Schreiben

ARGUMENTE **LABEL**

Label der Berechnungsvorschrift

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Label nicht gefunden

FUNKTION T_S_ENTH kennzeichnet für die Programme, die Enthalpien berechnen, die Methode, nach der beim nächsten Aufruf gerechnet werden soll.

T_S_LLEQ

Setzen der Methoden zur Berechnung von Flüssig-flüssig-Gleichgewichten

FORMAT T_S_LLEQ(LABEL, RVAL)

Name	Typ	Zugriff
LABEL	C	Lesen
RVAL	I4	Schreiben

ARGUMENTE **LABEL**

Label der Berechnungsvorschrift

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Label nicht gefunden

FUNKTION

T_S_LLEQ kennzeichnet für die Programme, die Flüssig-flüssig-Gleichgewichte berechnen, die Methode, nach der beim nächsten Aufruf gerechnet werden soll.

T_S_LVEQ

Setzen der Methoden zur Berechnung von Flüssig-Dampf-Gleichgewichten

FORMAT T_S_LVEQ(LABEL, RVAL)

Name	Typ	Zugriff
LABEL	C	Lesen
RVAL	I4	Schreiben

ARGUMENTE ***LABEL***

Label der Berechnungsvorschrift

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Label nicht gefunden

FUNKTION T_S_LVEQ kennzeichnet für die Programme, die Flüssig-Dampf-Gleichgewichte berechnen, die Methode, nach der beim nächsten Aufruf gerechnet werden soll.

T_S_SYSTEM

Aktivieren eines Stoffsystems

FORMAT T_S_SYSTEM(LABEL, VERS, RVAL)

Name	Typ	Zugriff
LABEL	C	Lesen
VERS	I4	Lesen
RVAL	I4	Schreiben

ARGUMENTE **LABEL**

Name des zu aktivierenden Stoffsystems

VERS

Versionsnummer des zu aktivierenden Stoffsystems

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Stoffsystem nicht gefunden

FUNKTION

Die Berechnungsprogramme und T_G-Programme der Schnittstelle arbeiten mit den Daten des gerade aktiven Stoffsystems.

Der benötigte Stoffdatenblock muß deshalb vor Aufruf dieser Programme durch einen Aufruf von T_S_SYSTEM aktiviert werden.

Beim Arbeiten mit nur einem Stoffdatenblock kann dieser Aufruf entfallen, da im Programm T_LOAD automatisch der erste eingelesene Stoffdatenblock aktiviert wird.

T_S_VERSION

Aktivieren der Daten, die ausschliesslich zu einem Stoffsystem mit Versionsnummer > 1 gehören

FORMAT

T_S_VERSION(LABEL, VERS, RVAL)

Name	Typ	Zugriff
LABEL	C	Lesen
VERS	I4	Lesen
RVAL	I4	Schreiben

ARGUMENTE

LABEL

Name des zu aktivierenden Stoffsystems

VERS

Versionsnummer des zu aktivierenden Stoffsystems

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Stoffsystem nicht gefunden

=-1: Versionsnummer = 1

FUNKTION

Mit T_S_VERSION werden nur die Daten aktiviert, die zu diesem Stoffsystem mit Versionsnummer > 1 gehören. Die T_G-Programme der Schnittstelle können dann diese Daten liefern.

Für die Berechnung ist es notwendig nach einem Aufruf von T_S_VERSION durch einen Aufruf von T_S_SYSTEM das Gesamtsystem wieder zu aktivieren.

4.5 Berechnungsprogramme

Die Berechnungsprogramme sind so aufgebaut, daß jeweils ein Unterprogramm die Berechnung der Werte und eventuell der Ableitungen übernimmt. Ein zweites Unterprogramm mit dem Postfix `_DERIVATIVE` liefert die Werte der Ableitungen, die intern gespeichert worden sind. Hier wird also nicht mehr gerechnet.

Der Parameter `WHAT` in der Parameterliste der Berechnungsprogramme steuert die Berechnung der Ableitungen. Die Kennungen, die in den Unterprogrammen gültig sind, werden je Unterprogramm in der Parameterlistenbeschreibung angegeben. Es ist die Angabe jeder Kombination dieser Kennungen möglich. Sollen keine Ableitungen gebildet werden, ist der Parameter `WHAT = Blank` zu setzen.

In den Programmen, die die Ableitungen zurückgeben, hat der Parameter `WHAT` die Länge 1. Hier führt jede Eingabe eines nicht definierten Zeichen zum Setzen der `RVAL`-Marke.

T_AVER

Berechnung von Mittelwerten

FORMAT T_AVER(PROPTYP, WHAT, T, P, MOLES, VALUE, RVAL)

Name	Typ	Zugriff
PROPTYP	C	Lesen
WHAT	C	Lesen
T	R8	Lesen
P	R8	Lesen
MOLES	R8(*)	Lesen
VALUE	R8	Schreiben
RVAL	R8	Schreiben

ARGUMENTE **PROPTYP**

Stoffdatentyp, entweder Reinstoffwert oder spez. Wert wie die Gasdichte

WHATSteuerparameter, der die zu bildenden partiellen Ableitungen kennzeichnet
gültige Kennzeichnungen sind:

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

'M' : Ableitung nach den Molanteilen

T

Temperatur in Kelvin

P

Druck in Pascal

MOLES

Vektor mit den Molanteilen der Komponenten in der dem Stoffdatentyp entsprechenden Phase

VALUE

berechneter Mittelwert

RVAL

Fehlerkennung

= 0: alles ok.

=-1: ungültiger Stoffdatentyp oder Stoffdatum nicht gespeichert

= 1: Basisstoffdaten fehlen für die Berechnung

FUNKTION

Die Berechnung des mittleren Stoffwertes erfolgt nach der im Stoffdatenspeicher eingetragenen Methode.

Ist im Parameter WHAT gekennzeichnet, daß die partiellen Ableitungen gebildet werden sollen, so werden diese direkt mitberechnet und zur Abfrage mit dem Programm T_AVER_DERIVATIVE intern gespeichert. Ein erneuter Aufruf von T_AVER überschreibt die Ableitungen.

T_AVER_DERIVATIVE

Rückgabe von Ableitungen von Reinstoffmittelwerten

FORMAT T_AVER_DERIVATIVE(WHAT, DERIVATIVE, RVAL)

Name	Typ	Zugriff
WHAT	C1	Lesen
DERIVATIVE	R8	Schreiben
RVAL	I4	Schreiben

ARGUMENTE

WHAT

Steuerparameter, welche Ableitung zurückgegeben werden soll

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

'M' : Ableitungen nach den Molanteilen

DERIVATIVE

enthält die gewünschte Ableitung

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Ableitung wurde nicht gebildet

=-1: ungültige Ableitungskennung

FUNKTION

T_AVER_DERIVATIVE kann erst aufgerufen werden, wenn durch einen vorangegangenen Aufruf von T_AVER sichergestellt ist, daß die Ableitungen nach den Zustandsgrößen im internen Speicher zur Verfügung stehen.

Wird T_AVER_DERIVATIVE mit der Kennung "T" bzw. "P" aufgerufen, wird der Wert der Ableitung in DERIVATIVE(1) zurückgegeben.

Wird T_AVER_DERIVATIVE mit der Kennung "M" aufgerufen, wird in DERIVATIVE der Vektor der Ableitungen für alle Komponenten zurückgegeben. Die Dimension des Vektors muß in diesem Fall mindestens gleich der aktuellen Komponentenzahl sein.

T_CHEM

Berechnung von chemischen Reaktionen

FORMAT T_CHEM(NUMBER, WHAT, T, P, ZETA, VOLUME, MOLES, VALUE, RVAL)

Name	Typ	Zugriff
NUMBER	I4	Lesen
WHAT	C	Lesen
T	R8	Lesen
P	R8	Lesen
ZETA	R8	Lesen
VOLUME	R8	Lesen
MOLES	R8(*)	Lesen
VALUE	R8	Schreiben
RVAL	R8	Schreiben

ARGUMENTE**NUMBER**

Nummer der Reaktion des gerade aktiven Chemieblockes, die berechnet werden soll

WHAT

Steuerparameter, der die zu bildenden partiellen Ableitungen kennzeichnet
gültige Kennzeichnungen sind:

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

'M' : Ableitung nach den Molanteilen

'Z' : Ableitung nach der Reaktionskoordinate

'V' : Ableitung nach dem Reaktionsvolumen

T

Temperatur in Kelvin

P

Druck in Pascal

ZETA

Reaktionskoordinate

VOLUME

Reaktionsvolumen

MOLES

Vektor mit den Molanteilen der Komponenten in der dem Reaktionstyp entsprechenden Phase

VALUE

Funktionswert

RVAL

Fehlerkennung

= 0: alles ok.

=-1: ungültige Reaktionsnummer

FUNKTION

Es werden Daten zum aktivierten Reaktionssystem (s. T_S_CHEMICS) und zwar für die Reaktion mit der angegebenen Nummer berechnet.

Ist im Parameter WHAT gekennzeichnet, daß die partiellen Ableitungen gebildet werden sollen, so werden diese direkt mitberechnet und zur Abfrage mit dem Programm T_CHEM_DERIVATIVE intern gespeichert. Ein erneuter Aufruf von T_CHEM überschreibt die Ableitungen.

T_CHEM_DERIVATIVE

Rückgabe von Ableitungen von einer Reaktion

FORMAT T_CHEM_DERIVATIVE(WHAT, DERIVATIVE, RVAL)

Name	Typ	Zugriff
WHAT	C1	Lesen
DERIVATIVE	R8	Schreiben
RVAL	I4	Schreiben

ARGUMENTE

WHAT

Steuerparameter, welche Ableitung zurückgegeben werden soll

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

'Z' : Ableitung nach Zeta

'V' : Ableitung nach dem Volumen

'M' : Ableitungen nach den Molanteilen

DERIVATIVE

enthält die gewünschte Ableitung

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Ableitung wurde nicht gebildet

=-1: ungültige Ableitungskennung

FUNKTION

T_CHEM_DERIVATIVE kann erst aufgerufen werden, wenn durch einen vorangegangenen Aufruf von T_CHEM sichergestellt ist, daß die Ableitungen nach den Zustandsgrößen im internen Speicher zur Verfügung stehen.

Wird T_CHEM_DERIVATIVE mit der Kennung "T", "V", "Z" bzw. "P" aufgerufen, wird der Wert der Ableitung in DERIVATIVE(1) zurückgegeben.

Wird T_CHEM_DERIVATIVE mit der Kennung "M" aufgerufen, wird in DERIVATIVE der Vektor der Ableitungen für alle Komponenten zurückgegeben. Die Dimension des Vektors muß in diesem Fall mindestens gleich der aktuellen Komponentenzahl sein.

T_CH

Berechnung der Massen- und Enthalpiebilanzterme für Reaktionen

FORMAT

T_CH(TASK, WHAT, ZETA, VALUE, RVAL)

Name	Typ	Zugriff
TASK	I4	Lesen
WHAT	C	Lesen
ZETA	R8	Lesen
VALUE	R8(*)	Schreiben
RVAL	R8	Schreiben

ARGUMENTE**TASK**

Aufgabenstellung

= 1 : Massenbilanz

= 2 : Enthalpiebilanz

WHAT

Steuerparameter, der die zu bildenden partiellen Ableitungen kennzeichnet
gültige Kennzeichnungen sind:

'Z' : Ableitung nach der Reaktionskoordinate

ZETA

Reaktionskoordinate

VALUE

gewünschte Werte

RVAL

Fehlerkennung

= 0: alles ok.

FUNKTION

Es werden Werte zum aktivierten Reaktionssystem (s. T_S_CHEMICS) berechnet.

Für TASK = 1 enthält VALUE als Vektor der Länge Anzahl Komponenten die Massenbilanzsummen je Komponente.

Für TASK = 2 enthält VALUE die Summe der Reaktionswärmen.

Ist im Parameter WHAT gekennzeichnet, daß die partiellen Ableitungen gebildet werden sollen, so werden diese direkt mitberechnet und zur Abfrage mit dem Programm T_CH_DERIVATIVE intern gespeichert. Ein erneuter Aufruf von T_CH überschreibt die Ableitungen.

T_CH_DERIVATIVE

Rückgabe von Ableitungen der Reaktionsterme für Massen- bzw. Enthalpiebilanz.

FORMAT T_CH_DERIVATIVE(TASK, WHAT, COMP_NR, DERIVATIVE, RVAL)

Name	Typ	Zugriff
TASK	I4	Lesen
WHAT	C1	Lesen
COMP_NR	I4	Lesen
DERIVATIVE	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**TASK**

=1: Massenbilanzableitungen

=2: Enthalpiebilanzableitungen

WHAT

Steuerparameter, welche Ableitung zurückgegeben werden soll

'Z' : Ableitung nach Zeta

COMP_NR

Komponentennummer für TASK=1

DERIVATIVE

enthält die gewünschte Ableitung

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Ableitung wurde nicht gebildet

=-1: ungültige Ableitungskennung

FUNKTION

T_CH_DERIVATIVE kann erst aufgerufen werden, wenn durch einen vorangegangenen Aufruf von T_CH sichergestellt ist, daß die Ableitungen nach den Zustandsgrößen im internen Speicher zur Verfügung stehen.

Die Dimension des Vektors DERIVATIVE muß mindestens gleich der Reaktionszahl des gerade aktiven Chemiesystems sein.

Für TASK = 1 werden im Ausgabevektor die Ableitungen der j-ten Komponente für alle Reaktionen geliefert.

Für Task =2 enthält der Ausgabevektor die Ableitungen des Enthalpiebilanzterms je Reaktion.

T_COMPR

Berechnung der Kompressibilitäten in der Gasphase

FORMAT T_COMPR(METHODE, WHAT, T, P, VAPOR, Z, RVAL)

Name	Typ	Zugriff
METHODE	C	Lesen
WHAT	C	Lesen
T	R8	Lesen
P	R8	Lesen
VAPOR	R8(*)	Lesen
Z	R8	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**METHODE**

Kennung der Methode, nach der die Kompressibilität berechnet werden soll.

= 'RKS' : Redlich-Kwong-Soave

= 'VLEQ' : Berechnung nach der im Flüssig-Dampf-Phasengleichgewichtslabel angegebenen Methode für Fugazitäten oder Zustandsgleichungen

WHAT

Steuerparameter, der die zu bildenden partiellen Ableitungen kennzeichnet gültige Kennzeichnungen sind:

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

'Y' : Ableitung nach den Molanteilen des Dampfes

T

Systemtemperatur in Kelvin

P

Systemdruck in Pascal

VAPOR

Molanteile im Dampf

Z

Kompressibilitätsfaktor

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Z = 1; alle Ableitungen = 0

=-1: Methodenkennzeichnung nicht gültig

FUNKTION

Die Berechnung der Kompressibilität erfolgt nach der in der Parameterliste angegebenen Methode bzw. bei METHOD = 'VLEQ' nach der für die Fugazitäten oder Zustandsgleichungen angegebenen Methode.

Ist im Parameter WHAT gekennzeichnet, daß die partiellen Ableitungen gebildet werden sollen, so werden diese direkt mitberechnet und zur Abfrage mit dem Programm T_COMPR_DERIVATIVE intern gespeichert.

T_COMPR_DERIVATIVE

Rückgabe der Ableitungen der Kompressibilitäten

FORMAT T_COMPR_DERIVATIVE(WHAT, DERIVATIVE, RVAL)

Name	Typ	Zugriff
WHAT	C1	Lesen
DERIVATIVE	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE

WHAT

Steuerparameter, welche Ableitung zurückgegeben werden soll

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

'Y' : Ableitungen nach den Molanteilen des Dampfes

DERIVATIVE

enthält die gewünschte Ableitung

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Ableitung wurde nicht gebildet

=-1: ungültige Ableitungskennung

FUNKTION

Die Dimension des Vektors DERIVATIVE muß mindestens gleich der aktuellen Komponentenzahl sein. Für die Ableitungen nach T,P wird in DERIVATIVE(1) ein Wert zurückgegeben; die Ableitung nach Y ist ein Vektor mit Länge gleich der aktuellen Komponentenzahl.

T_COMPR_DERIVATIVE kann erst aufgerufen werden, wenn durch einen vorangegangenen Aufruf von T_COMPR sichergestellt ist, daß die Ableitungen nach den Zustandsgrößen im internen Speicher zur Verfügung stehen.

T_ENTH

Berechnung von Enthalpien und deren Ableitungen

FORMAT

T_ENTH(WHAT, PHASE, T, P, MOLES, ENTHALPY, RVAL)

Name	Typ	Zugriff
WHAT	C	Lesen
PHASE	C	Lesen
T	R8	Lesen
P	R8	Lesen
MOLES	R8(*)	Lesen
ENTHALPY	R8	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**WHAT**

Steuerparameter, der die zu bildenden partiellen Ableitungen kennzeichnet
gültige Kennzeichnungen sind:

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

'M' : Ableitung nach den Molanteilen

PHASE

Kennung der Phase, für die die Enthalpie berechnet werden soll

'LIQU' : Flüssigkeit

'VAPO' : Dampf

'SOLI' : Feststoff

T

Systemtemperatur in Kelvin

P

Systemdruck in Pascal

MOLES

Molanteile der Komponenten in der Phase PHASE

ENTHALPY

berechnete Enthalpie

RVAL

Fehlerkennung

= 0: alles ok.

- = 1: Rechnung ohne Enthalpie
- = 2: bei isothermer Druckabhängigkeit keine reelle Lösung für die Kompressibilität gefunden
- =-1: Methodenkennzeichnung nicht gültig für Exzeßenthalpie oder isotherme Druckabhängigkeit

FUNKTION

Die Berechnung der Enthalpie erfolgt nach der Berechnungsvorschrift, die in dem durch einen Aufruf von T_S_ENTH gesetzten Label gespeichert ist.

Ist im Parameter WHAT gekennzeichnet, daß die partiellen Ableitungen gebildet werden sollen, so werden diese direkt mitberechnet und zur Abfrage mit dem Programm T_ENTH_DERIVATIVE intern gespeichert.

T_ENTH_DERIVATIVE

Rückgabe der Ableitungen der Enthalpien

FORMAT T_ENTH_DERIVATIVE(WHAT, PHASE, DERIVATIVE, RVAL)

Name	Typ	Zugriff
WHAT	C1	Lesen
PHASE	C	Lesen
DERIVATIVE	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE **WHAT**

Steuerparameter, welche Ableitung zurückgegeben werden soll

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

'M' : Ableitungen nach den Molanteilen

PHASE

Nummer der Phase, für die Ableitungen zurückgegeben werden sollen

'LIQU' : Flüssigkeit

'VAPO' : Dampf

'SOLI' : Feststoff

DERIVATIVE

enthält die gewünschte Ableitung

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Ableitung wurde nicht gebildet

=-1: ungültige Ableitungskennung

FUNKTION

Die Dimension des Vektors DERIVATIVE muß mindestens gleich der aktuellen Komponentenzahl sein. Für die Ableitungen nach T,P wird in DERIVATIVE(1) ein Wert zurückgegeben; die Ableitung nach M ist ein Vektor mit Länge gleich der aktuellen Komponentenzahl.

T_ENTH_DERIVATIVE kann erst aufgerufen werden, wenn durch einen vorangegangenen Aufruf von T_ENTH sichergestellt ist, daß die Ableitungen nach den Zustandsgrößen für die entsprechende Phase im internen Speicher zur Verfügung stehen.

T_FUGA

Berechnung von Fugazitätskoeffizienten

FORMAT T_FUGA(TASK, METHODE, WHAT, T, P, LIQUID, VAPOR, PHI, RVAL)

Name	Typ	Zugriff
TASK	I4	Lesen
METHODE	C	Lesen
WHAT	C	Lesen
T	R8	Lesen
P	R8	Lesen
LIQUID	R8(*)	Lesen
VAPOR	R8(*)	Lesen
PHI	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**TASK**

Aufgabenstellung

= 1 : berechne $\frac{\varphi_i^0}{\varphi_i^v}$ = 2 : berechne φ_i^v = 3 : berechne φ_i^l = 4 : berechne φ_i^0 = 5 : berechne $\frac{\varphi_i^l}{\varphi_i^v}$ **METHODE**

Kennung der Methode, nach der die Fugazitätskoeffizienten berechnet werden sollen.

WHATSteuerparameter, der die zu bildenden partiellen Ableitungen kennzeichnet
gültige Kennzeichnungen sind:

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

'X' : Ableitung nach den Molanteilen der Flüssigkeit

'Y' : Ableitung nach den Molanteilen des Dampfes

T

Systemtemperatur in Kelvin

P

Systemdruck in Pascal

LIQUID

Molanteile in der Flüssigkeit

VAPOR

Molanteile im Dampf

PHI

berechnete Fugazitätskoeffizienten

RVAL

Fehlerkennung

= 0: alles ok.

= 1: alle PHI-Werte = 1; alle Ableitungen = 0

=-1: Methodenkennzeichnung nicht gültig

FUNKTION

Die Berechnung der Fugazitätskoeffizienten erfolgt nach der in der Parameterliste angegebenen Methode.

Ist im Parameter WHAT gekennzeichnet, daß die partiellen Ableitungen gebildet werden sollen, so werden diese direkt mitberechnet und zur Abfrage mit dem Programm T_FUGA_DERIVATIVE intern gespeichert.

T_FUGA_DERIVATIVE

Rückgabe der Ableitungen der Fugazitätskoeffizienten

FORMAT T_FUGA_DERIVATIVE(WHAT, COMP_NR, DERIVATIVE, RVAL)

Name	Typ	Zugriff
WHAT	C1	Lesen
COMP_NR	I4	Lesen
DERIVATIVE	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**WHAT**

Steuerparameter, welche Ableitung zurückgegeben werden soll

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

'X' : Ableitungen nach den Molanteilen in der Flüssigkeit

'Y' : Ableitungen nach den Molanteilen im Dampf

COMP_NR

Komponentennummer, für die die Ableitung nach den Molanteilen als Vektor zurückgegeben werden soll

DERIVATIVE

enthält die gewünschte Ableitung

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Ableitung wurde nicht gebildet

=-1: ungültige Ableitungskennung

FUNKTION

Die Dimension des Vektors DERIVATIVE muß mindestens gleich der aktuellen Komponentenzahl sein.

T_FUGA_DERIVATIVE kann erst aufgerufen werden, wenn durch einen vorangegangenen Aufruf von T_FUGA sichergestellt ist, daß die Ableitungen nach den Zustandsgrößen im internen Speicher zur Verfügung stehen.

T_GAMMA

Berechnung von Aktivitätskoeffizienten

FORMAT T_GAMMA(METHODE, WHAT, T, LIQUID, VALUE, RVAL)

Name	Typ	Zugriff
METHODE	C	Lesen
WHAT	C	Lesen
T	R8	Lesen
LIQUID	R8(*)	Lesen
VALUE	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE **METHODE**

Kennung der Methode, nach der die Aktivitätskoeffizienten berechnet werden sollen.

WHATSteuerparameter, der die zu bildenden partiellen Ableitungen kennzeichnet
gültige Kennzeichnungen sind:

'T' : Ableitung nach der Temperatur

'X' : Ableitung nach den Molanteilen der Flüssigkeit

T

Systemtemperatur in Kelvin

LIQUID

Molanteile in der Flüssigkeit

VALUE

berechnete Aktivitätskoeffizienten

RVAL

Fehlerkennung

= 0: alles ok.

= 1: alle Gamma-Werte = 1; alle Ableitungen = 0

=-1: Methodenkennzeichnung nicht gültig

FUNKTION

Die Berechnung der Aktivitätskoeffizienten erfolgt nach der in der Parameterliste angegebenen Methode.

Ist im Parameter WHAT gekennzeichnet, daß die partiellen Ableitungen gebildet werden sollen, so werden diese direkt mitberechnet und zur Abfrage mit dem Programm T_GAMMA_DERIVATIVE intern gespeichert. Ein erneuter Aufruf von

T_GAMMA, auch implizit über das Programm T_LVEQ, überschreibt die Ableitungen.

T_GAMMA_DERIVATIVE

Rückgabe der Ableitungen der Aktivitätskoeffizienten

FORMAT T_GAMMA_DERIVATIVE(WHAT, COMP_NR, DERIVATIVE, RVAL)

Name	Typ	Zugriff
WHAT	C1	Lesen
COMP_NR	I4	Lesen
DERIVATIVE	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE

WHAT

Steuerparameter, welche Ableitung zurückgegeben werden soll

'T' : Ableitung nach der Temperatur

'X' : Ableitungen nach den Molanteilen in der Flüssigkeit

COMP_NR

Komponentennummer, für die die x-Ableitung als Vektor zurückgegeben werden soll

DERIVATIVE

enthält die gewünschte Ableitung

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Ableitung wurde nicht gebildet

=-1: ungültige Ableitungskennung

FUNKTION

Die Dimension des Vektors DERIVATIVE muß mindestens gleich der aktuellen Komponentenzahl sein.

T_GAMMA_DERIVATIVE kann erst aufgerufen werden, wenn durch einen vorangegangenen Aufruf von T_GAMMA sichergestellt ist, daß die Ableitungen nach den Zustandsgrößen im internen Speicher zur Verfügung stehen.

T_H_EX

Berechnung von Exzeßenthalpien und deren Ableitungen

FORMAT T_H_EX(METHODE, WHAT, T, MOLES, H_E, RVAL)

Name	Typ	Zugriff
METHODE	C	Lesen
WHAT	C	Lesen
T	R8	Lesen
MOLES	R8(*)	Lesen
H_E	R8	Schreiben
RVAL	I4	Schreiben

ARGUMENTE **METHODE**

Methodenkennung zur Berechnung der Exzeßenthalpie

WHAT

Steuerparameter, der die zu bildenden partiellen Ableitungen kennzeichnet
gültige Kennzeichnungen sind:

'T' : Ableitung nach der Temperatur

'M' : Ableitung nach den Molanteilen

T

Systemtemperatur in Kelvin

MOLES

Molanteile der Komponenten

H_E

berechnete Exzeßenthalpie

RVAL

Fehlerkennung

= 0: alles ok.

=-1: ungültige Methodenkennzeichnung

FUNKTION

Die Berechnung der Exzeßenthalpie erfolgt nach der in der Parameterliste angegebenen Methode.

Ist im Parameter WHAT gekennzeichnet, daß die partiellen Ableitungen gebildet werden sollen, so werden diese direkt mitberechnet und zur Abfrage mit dem Programm T_H_EX_DERIVATIVE intern gespeichert.

T_H_EX_DERIVATIVE

Rückgabe der Ableitungen der Exzeßenthalpien

FORMAT T_H_EX_DERIVATIVE(WHAT, DERIVATIVE, RVAL)

Name	Typ	Zugriff
WHAT	C1	Lesen
DERIVATIVE	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE *WHAT*

Steuerparameter, welche Ableitung zurückgegeben werden soll

'T' : Ableitung nach der Temperatur

'M' : Ableitungen nach den Molanteilen

DERIVATIVE

enthält die gewünschte Ableitung

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Ableitung wurde nicht gebildet

=-1: ungültige Ableitungskennung

FUNKTION

Die Dimension des Vektors DERIVATIVE muß mindestens gleich der aktuellen Komponentenzahl sein. Für die Ableitung nach T wird in DERIVATIVE(1) ein Wert zurückgegeben; die Ableitung nach M ist ein Vektor mit Länge gleich der aktuellen Komponentenzahl.

T_H_EX_DERIVATIVE kann erst aufgerufen werden, wenn durch einen vorangegangenen Aufruf von T_H_EX sichergestellt ist, daß die Ableitungen nach den Zustandsgrößen für die entsprechende Phase im internen Speicher zur Verfügung stehen.

T_H_ISO

Berechnung der isothermen Druckabhängigkeit der Enthalpie im Gas und deren Ableitungen

FORMAT T_H_ISO(METHODE, WHAT, JKOMP, T, P, MOLES, H_P, RVAL)

Name	Typ	Zugriff
METHODE	C	Lesen
WHAT	C	Lesen
JKOMP	I4	Lesen
T	R8	Lesen
P	R8	Lesen
MOLES	R8(*)	Lesen
H_P	R8	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**METHODE**

Methodenkennzeichnung zur Berechnung der isothermen Druckabhängigkeit

WHAT

Steuerparameter, der die zu bildenden partiellen Ableitungen kennzeichnet gültige Kennzeichnungen sind:

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

'M' : Ableitung nach den Molanteilen

JKOMP

Komponentennummer

=0 : Berechnung von ΔH für das Gemisch

>0 : Berechnung von ΔH der Komponente JKOMP

T

Systemtemperatur in Kelvin

P

Systemdruck in Pascal

MOLES

Molanteile der Komponenten

H_P

ΔH

RVAL

Fehlerkennung

= 0: alles ok.

= 1: bei isothermer Druckabhängigkeit keine Lösung für die Kompressibilität gefunden

=-1: Methodenkennzeichnung nicht gültig für isotherme Druckabhängigkeit

FUNKTION

Die Berechnung von ΔH erfolgt nach der in der Parameterliste angegebenen Methode.

Ist im Parameter WHAT gekennzeichnet, daß die partiellen Ableitungen gebildet werden sollen, so werden diese direkt mitberechnet und zur Abfrage mit dem Programm T_H_ISO_DERIVATIVE intern gespeichert.

T_H_ISO_DERIVATIVE

Rückgabe der Ableitungen der isothermen Druckabhängigkeit

FORMAT T_H_ISO_DERIVATIVE(WHAT, DERIVATIVE, RVAL)

Name	Typ	Zugriff
WHAT	C1	Lesen
DERIVATIVE	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE

WHAT

Steuerparameter, welche Ableitung zurückgegeben werden soll

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

'M' : Ableitungen nach den Molanteilen

DERIVATIVE

enthält die gewünschte Ableitung

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Ableitung wurde nicht gebildet

=-1: ungültige Ableitungskennung

FUNKTION

Die Dimension des Vektors DERIVATIVE muß mindestens gleich der aktuellen Komponentenzahl sein. Für die Ableitungen nach T,P wird in DERIVATIVE(1) ein Wert zurückgegeben; die Ableitung nach M ist ein Vektor mit Länge gleich der aktuellen Komponentenzahl.

T_H_ISO_DERIVATIVE kann erst aufgerufen werden, wenn durch einen vorangegangenen Aufruf von T_H_ISO sichergestellt ist, daß die Ableitungen nach den Zustandsgrößen für die entsprechende Phase im internen Speicher zur Verfügung stehen.

T_LLEQ

Berechnung von Flüssig-flüssig-Gleichgewichtsdaten

FORMAT T_LLEQ(WHAT, T, LIQUID1, LIQUID2, VALUE1, VALUE2, RVAL)

Name	Typ	Zugriff
WHAT	C	Lesen
T	R8	Lesen
LIQUID1	R8(*)	Lesen
LIQUID2	R8(*)	Lesen
VALUE1	R8(*)	Schreiben
VALUE2	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**WHAT**

Steuerparameter, der die zu bildenden partiellen Ableitungen kennzeichnet
gültige Kennzeichnungen sind:

'T' : Ableitung nach der Temperatur

'X' : Ableitung nach den Molanteilen der beiden flüssigen Phasen

T

Systemtemperatur in Kelvin

LIQUID1

Molanteile der Komponenten in der 1. flüssigen Phase

LIQUID2

Molanteile der Komponenten in der 2. flüssigen Phase

VALUE1

Gleichgewichtsdaten der Komponenten in der 1. flüssigen Phase

VALUE2

Gleichgewichtsdaten der Komponenten in der 2. flüssigen Phase

RVAL

Fehlerkennung

= 0: alles ok.

= 1: alle Werte = 1; alle Ableitungen = 0

FUNKTION

Das Unterprogramm T_LLEQ berechnet die Gleichgewichtsdaten der einzelnen Komponenten zu vorgegebener Temperatur und Molanteilen der beiden flüssigen Phasen.

Die Berechnung erfolgt nach der durch T_S_LLEQ vorgegebenen Methode. Ist keine Berechnungsmethode angegeben, so erhalten sämtliche Ausgabewerte den Wert 1.

Ist mit dem Parameter WHAT gekennzeichnet, daß Ableitungen gebildet werden sollen, so werden diese direkt mitberechnet und zur Abfrage mit dem Programm T_LLEQ_DERIVATIVE gespeichert.

T_LLEQ_DERIVATIVE

Rückgabe der Ableitungen von flüssig-flüssig-Gleichgewichtsdaten

FORMAT T_LLEQ_DERIVATIVE(WHAT, PHASE_NR, COMP_NR,
DERIVATIVE, RVAL)

Name	Typ	Zugriff
WHAT	C1	Lesen
PHASE_NR	I4	Lesen
COMP_NR	I4	Lesen
DERIVATIVE	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**WHAT**

Steuerparameter, welche Ableitung zurückgegeben werden soll

'T' : Ableitung nach der Temperatur

'X' : Ableitungen nach den Molanteilen in der angegebenen Phase

PHASE_NR

Nummer der Phase, für die die x-Ableitung als Vektor zurückgegeben werden soll

COMP_NR

Komponentennummer, für die die x-Ableitung als Vektor zurückgegeben werden soll

DERIVATIVE

enthält die gewünschte Ableitung

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Ableitung wurde nicht gebildet

=-1: ungültige Ableitungskennung

FUNKTION

Die Dimension des Vektors DERIVATIVE muß mindestens gleich der aktuellen Komponentenzahl sein.

T_LLEQ_DERIVATIVE kann erst aufgerufen werden, wenn durch einen vorangegangenen Aufruf von T_LLEQ sichergestellt ist, daß die Ableitungen nach den Zustandsgrößen im internen Speicher zur Verfügung stehen.

T_LVEQ

Berechnung von Flüssig-Dampf-Gleichgewichtsdaten nach Aufgabenstellung

FORMAT T_LVEQ(TASK, WHAT, T, P, LIQUID, VAPOR, VALUE, RVAL)

Name	Typ	Zugriff
TASK	I4	Lesen
WHAT	C	Lesen
T	R8	Lesen
P	R8	Lesen
LIQUID	R8(*)	Lesen
VAPOR	R8(*)	Lesen
VALUE	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**TASK**

Aufgabenstellung

= 1: Die k-Werte sollen berechnet werden

= 2: nur die Aktivitätskoeffizienten sollen berechnet werden

= 3: nur die Fugazitätskoeffizienten sollen berechnet werden

WHAT

Steuerparameter, der die zu bildenden partiellen Ableitungen kennzeichnet
gültige Kennzeichnungen sind:

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

'X' : Ableitung nach den Molanteilen der Flüssigkeit

'Y' : Ableitung nach den Molanteilen des Dampfes

T

Systemtemperatur in Kelvin

P

Systemdruck in Pascal

LIQUID

Molanteile der Komponenten in der Flüssigkeit

VAPOR

Molanteile der Komponenten im Dampf

VALUE

enthält die berechneten Werte (je nach TASK)

RVAL

Fehlerkennung

= 0: alles ok.

= 1: alle Werte = 1; alle Ableitungen = 0

FUNKTION

Das Unterprogramm T_LVEQ berechnet die k-Werte der einzelnen Komponenten zu vorgegebenem Druck, Temperatur und Molanteilen.

Die Berechnung erfolgt nach der durch T_S_LVEQ vorgegebenen Methode. Ist keine Berechnungsmethode angegeben, so erhalten sämtliche Ausgabewerte den Wert 1.

Ist mit dem Parameter WHAT gekennzeichnet, daß Ableitungen gebildet werden sollen, so werden diese direkt mitberechnet und zur Abfrage mit dem Programm T_LVEQ_DERIVATIVE gespeichert.

Wird T_LVEQ mit TASK = 2, 3 aufgerufen, so werden die Aktivitäts- und Fugazitäts- Koeffizienten nach der im Label für die Phasengleichgewichtsberechnung angegebenen Methode berechnet.

Die berechneten k-Werte werden im Bereich 1.D-15 bis 1.D15 eingeschränkt. Dies erspart den übergeordneten Programmen die Overflow-Abfrage beim Quadrieren bei Maschinen mit eingeschränkter Darstellungsgröße.

T_LVEQ_DERIVATIVE

Rückgabe der Ableitungen von k-Werten, Aktivitätskoeffizienten, Fugazitätskoeffizienten

FORMAT T_LVEQ_DERIVATIVE(TASK, WHAT, COMP_NR, DERIVATIVE, RVAL)

Name	Typ	Zugriff
TASK	I4	Lesen
WHAT	C1	Lesen
COMP_NR	I4	Lesen
DERIVATIVE	R8(*)	Schreiben
RVAL	I4	Schreiben

ARGUMENTE**TASK**

- = 1: Holen der Ableitung der k-Werte
- = 2: Holen der Ableitung der Aktivitätskoeffizienten
- = 3: Holen der Ableitung der Fugazitätskoeffizienten

WHAT

Steuerparameter, welche Ableitung zurückgegeben werden soll

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

'X' : Ableitungen nach den Molanteilen in der Flüssigkeit

'Y' : Ableitung nach den Molanteilen im Dampf

COMP_NR

Komponentennummer, für die die x-Ableitung als Vektor zurückgegeben werden soll

DERIVATIVE

enthält die gewünschte Ableitung

RVAL

Fehlerkennung

- = 0: alles ok.
- = 1: Ableitung wurde nicht gebildet
- =-1: ungültige Ableitungskennung

FUNKTION

Die Dimension des Vektors DERIVATIVE muß mindestens gleich der aktuellen Komponentenzahl sein.

T_LVEQ_DERIVATIVE

T_LVEQ_DERIVATIVE kann erst aufgerufen werden, wenn durch einen vorangegangenen Aufruf von T_LVEQ sichergestellt ist, daß die Ableitungen nach den Zustandsgrößen im internen Speicher zur Verfügung stehen.

T_PURE

Berechnung von Reinstoffdaten als Funktion von T und/oder P

FORMAT T_PURE(PROPTYP, WHAT, COMP_NR, T, P, VALUE, RVAL)

Name	Typ	Zugriff
PROPTYP	C	Lesen
WHAT	C	Lesen
COMP_NR	I4	Lesen
T	R8	Lesen
P	R8	Lesen
VALUE	R8	Schreiben
RVAL	I4	Schreiben

ARGUMENTE **PROPTYP**

Stoffdatentyp

WHAT

Steuerparameter, der die zu bildenden partiellen Ableitungen kennzeichnet
gültige Kennzeichnungen sind:

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

COMP_NR

Komponentennummer

= 0 : Werte für alle Komponenten berechnen

> 0 : Werte für eine bestimmte Komponente berechnen

T

Temperatur in Kelvin

P

Druck in Pascal

VALUE

berechnete Stoffdate(n)

RVAL

Fehlerkennung

= 0 : alles ok.

> 0 : Stoffdatum für diese Komponente nicht gespeichert

=-1 : Stoffdatum nicht gespeichert

- =-2 : ungültige Komponentenummer
- =-3 : Stoffdatentyp nicht gültig
- =-4 : Ansatz nicht identifizierbar im Berechnungsprogramm
- >1000: gespeicherter Gültigkeitsbereich für die Temperatur verletzt

FUNKTION

T_PURE liefert zu vorgegebener Temperatur den Wert der Reinstoffdatenfunktion. (Druckabhängige Funktionen sind bisher nicht vorhanden)

Wird der Parameter COMP_NR > 0 eingegeben, wird das Stoffdatum und, falls gewünscht, die Ableitung für eine einzelne Komponente geliefert.

Wird COMP_NR = 0 eingegeben, wird das Stoffdatum und die Ableitungen für alle Komponenten bestimmt. In diesem Fall muß VALUE mindestens die Dimension der aktuellen Komponentenzahl haben. Die Größe RVAL>0 gibt in diesem Fall die Nummer der Komponente an, für die zuletzt festgestellt wurde, daß das Stoffdatum fehlt. RVAL > 1000 bedeutet in diesem Fall, daß mindestens eine Komponente die Temperaturgrenzen verletzt hat.

Ist im Parameter WHAT gekennzeichnet, daß die partiellen Ableitungen gebildet werden sollen, so werden diese direkt mitberechnet und zur Abfrage mit dem Programm T_PURE_DERIVATIVE intern gespeichert. Ein erneuter Aufruf von T_PURE überschreibt die Ableitungen.

T_PURE_DERIVATIVE

Rückgabe der Ableitungen von Reinstoffdaten

FORMAT T_PURE_DERIVATIVE(WHAT, DERIVATIVE, RVAL)

Name	Typ	Zugriff
WHAT	C1	Lesen
DERIVATIVE	R8	Schreiben
RVAL	I4	Schreiben

ARGUMENTE ***WHAT***

Steuerparameter, welche Ableitung zurückgegeben werden soll

'T' : Ableitung nach der Temperatur

'P' : Ableitung nach dem Druck

DERIVATIVE

enthält die gewünschte Ableitung

RVAL

Fehlerkennung

= 0: alles ok.

= 1: Ableitung wurde nicht gebildet

=-1: ungültige Ableitungskennung

FUNKTION

T_PURE_DERIVATIVE kann erst aufgerufen werden, wenn durch einen vorhergegangenen Aufruf von T_PURE sichergestellt ist, daß die Ableitungen nach den Zustandsgrößen im internen Speicher zur Verfügung stehen.

Wurde T_PURE für eine einzelne Komponente (COMP_NR > 0) aufgerufen, wird in DERIVATIVE(1) die Ableitung für diese Komponente zurückgegeben.

Wurde T_PURE für alle Komponenten aufgerufen (COMP_NR = 0), wird in DERIVATIVE der Vektor der Ableitungen für alle Komponenten zurückgegeben. Die Dimension des Vektors muß in diesem Fall mindestens gleich der aktuellen Komponentenzahl sein.

