Radian

Radian Corporation

8409 RESEARCH BLVD. • AUSTIN, TEXAS 78758 • TELEPHONE 512 - 454-9535

FINAL REPORT

VOLUME III

APCO Contract No. CPA 70-45

A STUDY OF THE LIMESTONE INJECTION WET SCRUBBING PROCESS

Radian Corporation
8409 RESEARCH BLVD. • AUSTIN, TEXAS 78758 • TELEPHONE 512 - 454-9535

FINAL REPORT
VOLUME III

APCO Contract No. CPA 70-45

A STUDY OF THE LIMESTONE INJECTION WET SCRUBBING PROCESS

Presented to:

AIR POLLUTION CONTROL OFFICE
DEPARTMENT OF HEALTH, EDUCATION AND WELFARE
411 West Chapel Hill Street
Durham, North Carolina 27701

1 November 1971

Radian Corporation 8409 RESEARCH BLVD. • P.O. BOX 9948 • AUSTIN, TEXAS 78758 • TELEPHONE 512 - 454-9535

TABLE OF CONTENTS

1.0 SYSTEM ORGANIZATION

The group of computer programs written and used to conduct the prototype process simulations are functionally divided into three parts. Those routines in the executive section provide control and information on the progress of a simulation run. The equipment routines perform the calculations necessary to model the unit process operations. The auxiliary section is a collection of routines that provides special calculations for equipment routines, interface other programs, and summarize the results of a run. This functional division logically follows from the block oriented design of the system.

The blocks are the equipment routines, which are connected together by streams. A stream, or stream vector, represents the flow of material from one unit operation to another. Thus, the simulations very much resemble the engineer's flow sheet for the system being simulated. This type of simulation system is quite similar to many used in the petrochemical industry.

The following pages contain a description of the system, operating instructions, a technical note describing the equilibrium program used by the system, a sample data input, and program listings.

2.0 SYSTEM DESCRIPTION

The simulation system was written almost entirely in FORTRAN V as defined for a Univac 1108. The deviations from this language are slight and were done to use particular features of the machine available to us. Conceptually the system could be converted to run on any manufacturer's computer. The labor required to accomplish this depends on the machine and configuration to which it is being adapted. The description that follows is of the major elements of the system. In some instances it will be necessary to use some of the FORTRAN language.

2.1 Executive Section

The executive program, or more correctly, the driving program PROSIM first calls the card input routine READIN which does the following:

1) reads and prints one card image containing the run identification

2) reads a variable number of cards giving the external equipment number, name, and stream array. For a card having equipment No. I ($I \leq 25$), the six character name is stored in the array location IDEQP(I), the input streams are stored in the first five locations of ISTM(I,j), j=1,...,10. The output streams are stored in the last five locations of this array. The stream information

can be input in any order; the output streams are denoted by a preceding minus sign. The information on the card is then printed.

3) Reads a card image containing the order of process calculations. This is a sequence of external equipment numbers ordered from left to right indicating the sequence of subroutine calls (equipment calculations). The numbers are separated by commas, or a subset may be enclosed in parenthesis (1,4,7,8), to indicate a recycle loop. Up to three such loops can be used. They can be nested or separate. The list is terminated by an asterisk.

4) Reads a variable number of card images that give the equipment number and up to eight parameter values. For equipment I these values are stored in PA(I,j), j = 1,8. Upon furnishing these inputs the next call is to the routine PRELD which utilizes each equipment routine. This is where the source equipment routines read their card input if any. The next call is to PROCND where all equipment parameters and inputs for the simulation are printed in a stylized fashion.

The routine PROLOG calls each equipment routine specified in the calculation sequence until all recycle loops are converged or satisfied, which occurs when the flag L3, L2, L1 are all zero, and the sequence is completed. After all calculations are complete, the routine PTSUM is called to print the contents of the stream vector for all streams used. The last subroutine called, ARTWRK, prints outputs generated by equipment subroutines.

2.2 Process Equipment Routines

The technical description of the equipment routines have been given earlier in this report. The description that follows concerns itself with how they are used in the system.

2.2.1 Function

These routines perform the calculations of the process simulation. Normally they implement equations and algorithms to model the particular process unit operations by converting one or more input streams into one or more output streams in a specified fashion. This function is more simply described by the reaction $\overline{y} = f(\overline{x})$ where $\overline{x}$ is an input stream vector, $\overline{y}$ the output stream vector, and $f(\cdot)$ the functional description. In multiple inputs and outputs $\overline{x}$ and $\overline{y}$ are matrices. The location and identity of $\overline{x}$ and $\overline{y}$ are supplied by the executive program. In some instances, parameters that particularize the operation of an equipment are necessary. These are read in by the executive program and supplied to the routines when they are needed. Thus the preceding relation is more formally given by $\overline{y} = f(\overline{x};\overline{p})$, $\overline{p}$ being the parameter vector.

2.2.2 Coding Procedures

The calling sequence for each routine is CALL NAME(P,$) where NAME is six characters long, and P is the parameter array.

The following lines of code are required in the routine to interface with the executive program.

```
INCLUDE CMMN, LIST
   .
   .
   .
LOCSL = IDEQ(NL)
LIS = ISTM(LOCSL,1)
LOS = ISTM(LOCSL,6)
   .
   .
   .
coding for process calculations
   .
   .
   .
RETURN 2
END
```

LIS is the location of the first input stream.

LOS is the location of the first output stream.

Additional input streams if required are found in ISTM(LOCSL,2-5), and additional outputs are in ISTM(LOCSL,7-10).

These routines are re-entrant which means that the same routine may be used to compute another unit operation in the process having different parameters.

2.3 Stream Vector

The stream vector is not a program, but a specified array of storage that contains the information transferred between equipment routines. The locations in the vector are the same for all of them. In the following table these locations and what they designate are listed.

TABLE OF STREAM VECTOR LOCATIONS

Number of Element	Item	Units
1	Name	——
2	Flag	——
3	Flag	——
4	Flag	——
5	Temperature	°K
6	Pressure	Atm
7	Heat Content	cal/sec
8-9	not used	——
Gas Phase		
10	Gas Flow Rate	g-moles/sec
11	SO_2 Content	g-moles/sec

TABLE OF STREAM VECTOR LOCATIONS (Cont'd.)

Number of Element	Item	Units
12	CO_2	g-moles/sec
13	NO	g-moles/sec
14	NO_2	g-moles/sec
15	O_2	g-moles/sec
16	CO	g-moles/sec
17	N_2	g-moles/sec
18	HCl	g-moles/sec
19	H_2O	g-moles/sec
20-30	not used	—
Equilibrium Species		
31	Moles total SO_2	g-moles/sec
32	Moles total CO_2	g-moles/sec
33	Moles total SO_3	g-moles/sec
34	Moles total N_2O_5	g-moles/sec
35	Moles total CaO	g-moles/sec
36	Moles total MgO	g-moles/sec
37	Moles total Na_2O	g-moles/sec
38	Moles total HCl	g-moles/sec
39	Moles total H_2O	g-moles/sec
40-47	Not used	—
Liquid Phase		
48	Ionic strength	—
49	PH	—

TABLE OF STREAM VECTOR LOCATIONS (Cont'd.)

Number of Element	Item	Units
50	Liquid flow rate	g-liquid H_2O/sec
51	Concentration of H^+	g-moles/kg H_2O
52	Concentration of OH^-	g-moles/kg H_2O
53	Concentration of HSO_3^-	g-moles/kg H_2O
54	Concentration of $SO_3^=$	g-moles/kg H_2O
55	Concentration of $SO_4^=$	g-moles/kg H_2O
56	Concentration of HCO_3^-	g-moles/kg H_2O
57	Concentration of $CO_3^=$	g-moles/kg H_2O
58	Concentration of NO_3^-	g-moles/kg H_2O
59	Concentration of HSO_4^-	g-moles/kg H_2O
60	Concentration of $H_2SO_3(\ell)$	g-moles/kg H_2O
61	Concentration of $H_2CO_3(\ell)$	g-moles/kg H_2O
62	Concentration of Ca^{++}	g-moles/kg H_2O
63	Concentration of $CaOH^+$	g-moles/kg H_2O
64	Concentration of $CaSO_3(\ell)$	g-moles/kg H_2O
65	Concentration of $CaCO_3(\ell)$	g-moles/kg H_2O
66	Concentration of $CaHCO_3^+$	g-moles/kg H_2O
67	Concentration of $CaSO_4(\ell)$	g-moles/kg H_2O
68	Concentration of $CaNO_3^+$	g-moles/kg H_2O
69	Concentration of Mg^{++}	g-moles/kg H_2O
70	Concentration $MgOH^+$	g-moles/kg H_2O
71	Concentration $MgSO_3(\ell)$	g-moles/kg H_2O
72	Concentration of $MgHCO_3^+$	g-moles/kg H_2O

TABLE OF STREAM VECTOR LOCATIONS (Cont'd.)

Number of Element	Item	Units
73	Concentration of $MgSO_4(\ell)$	g-moles/kg H_2O
74	Concentration of $MgCO_3(\ell)$	g-moles/kg H_2O
75	Concentration of Na^+	g-moles/kg H_2O
76	Concentration of $NaOH(\ell)$	g-moles/kg H_2O
77	Concentration of $NaCO_3^-$	g-moles/kg H_2O
78	Concentration of $NaHCO_3(\ell)$	g-moles/kg H_2O
79	Concentration of $NaSO_4^-$	g-moles/kg H_2O
80	Concentration $NaNO_3(\ell)$	g-moles/kg H_2O
81	Concentration of Cl^-	g-moles/kg H_2O
82-99	not used	——
Solid Phase		
100	Weight fraction solids	g-solid/g total
101	Amount of CaO(s)	g-moles/sec
102	Amount of $Ca(OH)_2(s)$	g-moles/sec
103	Amount of $CaCO_3(s)$	g-moles/sec
104	Amount of $CaSO_3(s)$	g-moles/sec
105	Amount of $CaSO_3 \cdot \frac{1}{2}H_2O(s)$	g-moles/sec
106	Amount of $CaSO_4(s)$	g-moles/sec
107	Amount of $CaSO_4 \cdot 2H_2O(s)$	g-moles/sec
108	Amount of MgO(s)	g-moles/sec
109	Amount of $Mg(OH)_2(s)$	g-moles/sec
110	Amount of $MgCO_3(s)$	g-moles/sec
111	Amount of $MgCO_3 \cdot 3H_2O(s)$	g-moles/sec

TABLE OF STREAM VECTOR LOCATIONS (Cont'd.)

Number of Element	Item	Units
112	Amount of $MgCO_3 \cdot 5H_2O(s)$	g-moles/sec
113	Amount of $MgSO_3(s)$	g-moles/sec
114	Amount of $MgSO_3 \cdot 3H_2O(s)$	g-moles/sec
115	Amount of $MgSO_3 \cdot 6H_2O(s)$	g-moles/sec
116	Amount of $MgSO_4(s)$	g-moles/sec
117	Amount of Soluble Na_2O	g-moles/sec
118	Amount of Insoluble Fly Ash	g-moles/sec
119	Amount of NaCl	g-moles/sec
120	not used	——

3.0 OPERATING INSTRUCTIONS

Each simulation run is determined by data card entry. The data sheets used for generating entries for the prototype simulations are given in the following table titled PROSIM CARD INPUT SEQUENCE.

First, an array describing the process matrix is given. The number of the equipment subroutine is given in columns 1-2. The name of the equipment subroutine is given in columns 5-10. The numbers of the input and output streams are given in columns 11-60. A plus sign is used to denote an input stream and a minus is used for output streams. These entries were fixed for the prototype simulations since the process flow arrangement was not altered.

PROSIM CARD INPUT SEQUENCE

Column Number	0	2	4	10	15	20	25	30	35 ... 60	FORMAT
Run Identification	PROTOTYPE SIMULATION NO. ____									(13A6)
	Eq. No.		Equip. Name	Stream Numbers						
Process Matrix	1		FLUGAS	-1						(I2,2X,A6,10I5)
	2		WTRMKP	-14						
	3		CLRHTR	1	-2					
	4		SCRUBR	-3	-6	2	15			
	5		CLRHTR	3	-4					
	6		DIVDER	7	-8	-9				
	7		CLRFYR	9	-10	-11				
	8		FILTER	11	-12	-13				
	9		EQMIXR	8	10	12	14	-15		
	10		OVALMB	2	14	-3	-13			
	11		FLTRBM	13						
	12		STKGAS	5						
	13		PMPFAN	4	-5					
	@	EOF								
Process Sequence	1, 2, 3 (10, 8 (7, 6, 9, 4)) 5, 11, 13, 12*									(80R1)

NOTE: The number of streams to be specified by each equipment subroutine is fixed by that subroutine.

Radian

PROSIM CARD INPUT SEQUENCE (Continued)

Column Number 0	2	10	20	30	40	50	60	70	80	FORMAT
Equipment Parameters	No.	P1	P2	P3	P4	P5	P6	P7	P8	(I2, E8.3, 7E10.3)
(CLRHTR)	3*	TIGS	ΔPI							
(SCRUBR)	4*	LSR 8.**	LISF	NSP	XLH(CAO)	XLH(MGO)				
(CLRHTR)	5*	TISG	ΔPI							
(DIVDER)	6*	LISR								
(CLRFYR)	7*	LSF 15.**	LSR 8.**	LFB 13.**	LWM 14.**	XSOC				
(FILTER)	8*	XWSFB	XWSCB	XSOF						
(OVALMB)	10*	XA(SO_2)	XA(CO_2)	XA(NO)	XA(NO_2)	XO	ΔPI	XLSU(CAO)	XLSU(MGO)	
(PMPFAN)	13*	POSG	XUW							
	@	EOF								

* Fixed entries for the Prototype Simulation. These numbers designate the equipment subroutine that requires these input data.

** Fixed entries for the Prototype Simulation. These stream numbers are required inputs for the SCRUBR and CLRFYR subroutines.

PROSIM CARD INPUT SEQUENCE (Continued)

Column Number	Card	0–10	10–20	20–30	30–40	40–50	50–60	60–70	70–80	Format
Input Stream Quantities										
FLUGAS	1	VIFG	TIFG	PIFG						(8E10.3)
	2	YFG(SO_2)	(CO_2)	(NO)	(NO_2)	(O_2)	(CO)	(N_2)	(HCL)	
	3	YFG(H_2O)								
	4	WILS								
	5	XWLS(CAO)	(MGO)	($CACO_3$)	($MGCO_3$)	($CASO_3$)	($MGSO_3$)	($CASO_4$)	($MGSO_4$)	
	6	WIFA								
	7	XWFA(NA_2O)	(INS. FLY ASH)	(NACL)						
	8	XLD($CA(OH)_2$)	XLD($CACO_3$)	XLD($MG(OH)_2$)	XLD($MGCO_3$)	XLD($MGSO_4$)	XLD(NA_2O)	XLD(NACL)		
	9	XSD($CA(OH)_2$)	XSD($CACO_3$)	XSD($MG(OH)_2$)	XSD($MGCO_3$)	XSD($MGSO_4$)	XSD(NA_2O)	XSD(NACL)		
WTRMKP	1	MIWM($SO_3^=$)	($CO_3^=$)	($SO_4^=$)	(NO_3^-)	(CA^{++})	(MG^{++})	(NA^+)	(CL^-)	

Next, the sequence of process calculations is given. This is an ordered list of equipment numbers denoting the desired calculational sequence. The rationale for the order of calculations is given in Volume II, T.N. 200-014-19.

Last, an array of process equipment parameters is listed in the PROSIM CARD INPUT SEQUENCE. Some of these entries were held constant for the prototype simulations. That is, the equipment parameters LSR, LSF, LFB and LWM were fixed entries for the prototype simulations. Here, LSR, LSF, LFB and LWM are the stream numbers (8, 15, 13, 14) for the slurry recycle, scrubber feed, filter bottoms, and water makeup streams, respectively. The SCRUBR subroutine is programmed so that it requires specification of LSR, whereas the CLRFYR subroutine requires specification of LSF, LSR, LSB and LWM. Definitions for the remaining program entry variables are given below:

TIGS	-	temperature of the flue gas stream (°F)
TISG	-	temperature of the stack gas stream (°F)
ΔPI	-	pressure drop (atm)
LISF	-	scrubber feed rate (gal/min)
XLH(J)	-	fraction J hydrating in scrubber
LISR	-	initial value for slurry recycle rate (gH_2O/sec)
XSOC	-	clarifier efficiency
XSOF	-	filter efficiency
XWSFB	-	weight fraction solids in filter bottoms
XWSCB	-	weight fraction solids in clarifier bottoms
XA(J)	-	fraction of J absorbed
XO	-	fraction of SO_2 oxidized
XLSU(J)	-	fraction of J utilized in the system
POSG	-	pressure of the stack gas stream at the I.D. fan outlet (psia)
XUW	-	fan efficiency

VIFG	-	velocity of the flue gas stream (acfm)	
TIFG	-	temperature of the flue gas stream (°F)	
PIFG	-	pressure of the flue gas stream (psia)	
YFG(J)	-	mole fraction of J	
WILS	-	weight rate of limestone	(lbm/min)
XWLS(J)	-	weight fraction of J	
WIFA	-	weight rate of fly ash	(lbm/min)
XWFA	-	weight fraction of J	
XLD(J)	-	fraction of J from LS that dissolves	
XSD(J)	-	fraction of J from SF that dissolves	
MIWM(J)	-	concentration of J in water makeup (mg/ℓ)	

A sample input deck listing (for PSN #2B) is given in the first part of Section 5.0. The equipment numbers, equipment names, stream numbers, calculation sequence, and fixed parameters correspond to those shown in the PROSIM CARD INPUT SEQUENCE table. It should be noted that not all equipment subroutines require input data. Thus, no equipment parameters for these subroutines are indicated in the PROSIM CARD INPUT SEQUENCE table. Specification of input stream quantities for the inlet gas-solid stream (FLUGAS) and for the water makeup (WTRMKP) is indicated on the last page of the PROSIM CARD INPUT SEQUENCE table.

4.0 TECHNICAL NOTE 200-004-17 - A DESCRIPTION OF THE RADIAN CHEMICAL EQUILIBRIUM PROGRAM USED IN THE PROCESS MODEL SIMULATION SYSTEM

1.0 INTRODUCTION

The Prototype Process Model Simulation Program required certain modifications be made to the Radian Chemical Equilibrium program. These changes consisted of redefining the inputs, computed quantities, and the addition of a few new variables. The basic structure and operating characteristics of the program was not markedly changed.

The following is a technical description of the modified program.

2.0 PROGRAM INPUTS

The normal inputs to the program are the solution temperature, the key species, and some option flags. The solution temperature, in degrees Kelvin, is used to compute equilibrium constants, number of moles of hydrated water per mole, and constants associated with activity coefficients.

The key species are total moles of the following neutral species: SO_2, CO_2, SO_3, N_2O_5, CaO, MgO, Na_2O, HCl, and H_2O. These are the inputs for normal operation; the optional inputs will be discussed in a later section.

3.0 PROGRAM FORMULATION

The program essentially solves sets of nonlinear equations. The possible equations solved and variables solved for are listed below:

$$K_W = a_{H^+} a_{OH^-} / \gamma_{H_2O} \tag{1}$$

$$K_2 a_{H_2SO_3} = a_{H^+} a_{HSO_3^-} \tag{2}$$

$$K_3 a_{HSO_3^-} = a_{H^+} a_{SO_3^=} \quad (3)$$

$$K_4 a_{HSO_4^-} = a_{H^+} a_{SO_4^=} \quad (4)$$

$$K_5 a_{H_2CO_3} = a_{H^+} a_{HCO_3^-} \quad (5)$$

$$K_6 a_{HCO_3^-} = a_{H^+} a_{CO_3^=} \quad (6)$$

$$K_7 a_{CaOH^+} = a_{Ca^{++}} a_{OH^-} \quad (7)$$

$$K_8 a_{CaSO_3} = a_{Ca^{++}} a_{SO_3^=} \quad (8)$$

$$K_9 a_{CaCO_3} = a_{Ca^{++}} a_{CO_3^=} \quad (9)$$

$$K_{10} a_{CaHCO_3^+} = a_{Ca^{++}} a_{HCO_3^-} \quad (10)$$

$$K_{11} a_{CaSO_4} = a_{Ca^{++}} a_{SO_4^=} \quad (11)$$

$$K_{12} a_{CaNO_3^+} = a_{Ca^{++}} a_{NO_3^-} \quad (12)$$

$$K_{sp}[CaCO_3(S)] \geq a_{Ca^{++}} a_{CO_3^=} \quad (13)$$

$$K_{sp}[CaSO_4(S)] \geq a_{Ca^{++}} a_{SO_4^=} \gamma_{H_2O}^{h_1} \quad (14)$$

$$K_{sp}[CaSO_3(S)] \geq a_{Ca^{++}} a_{SO_3^=} \gamma_{H_2O}^{\frac{1}{2}} \quad (15)$$

$$K_{sp}[Ca(OH)_2(S)] \geq a_{Ca^{++}} a_{OH^-}^2 \quad (16)$$

$$K_{17} a_{MgOH^+} = a_{Mg^{++}} a_{OH^-} \quad (17)$$

$$K_{18} a_{MgSO_3} = a_{Mg^{++}} a_{SO_3^=} \quad (18)$$

$$K_{19} a_{MgHCO_3^+} = a_{Mg^{++}} a_{HCO_3^-} \quad (19)$$

$$K_{20} a_{MgSO_4} = a_{Mg^{++}} a_{SO_4^=} \quad (20)$$

$$K_{21} a_{MgCO_3} = a_{Mg^{++}} a_{CO_3^=} \quad (21)$$

$$K_{sp}[Mg(OH)_2(S)] \geq a_{Mg^{++}} a^2_{OH^-} \quad (22)$$

$$K_{sp}[MgSO_3(S)] \geq a_{Mg^{++}} a_{SO_3^=} \gamma^{h_2}_{H_2O} \quad (23)$$

$$K_{sp}[MgCO_3(S)] \geq a_{Mg^{++}} a_{CO_3^=} \gamma^{h_3}_{H_2O} \quad (24)$$

$$K_{24} a_{NaOH} = a_{Na^+} a_{OH^-} \quad (25)$$

$$K_{25} a_{NaCO_3^-} = a_{Na^+} a_{CO_3^=} \quad (26)$$

$$K_{26} a_{NaHCO_3} = a_{Na^+} a_{HCO_3^-} \quad (27)$$

$$K_{27} a_{NaSO_4^-} = a_{Na^+} a_{SO_4^=} \quad (28)$$

$$K_{28} a_{NaNO_3} = a_{Na^+} a_{NO_3^-} \quad (29)$$

Where: $a_i = \gamma_i m_i$ = activity of component i
γ_i = activity coefficient of component i
m_i = molality (moles/KgmH_2O) of component i
h_i = number of moles hydrated water per mole of component i

Letting L_K be the amount of molecular water in kgms, then the material balances are written

$$Total_{SO_2}/L_K = m_{HSO_3^-} + m_{SO_3^=} + m_{H_2SO_3} + m_{CaSO_3}$$

$$+ m_{CaSO_3(S)} + m_{MgSO_3} + m_{MgSO_3(S)} \quad (30)$$

$$\mathrm{Total}_{CO_2/L_K} = m_{HCO_3^-} + m_{CO_3^=} + m_{H_2CO_3} + m_{CaCO_3} + m_{CaHCO_3^+} + m_{CaCO_3(S)} + m_{MgHCO_3^+} + m_{MgCO_3} + m_{MgCO_3(S)} + m_{NaCO_3^-} + m_{NaHCO_3} \quad (31)$$

$$\mathrm{Total}_{SO_3/L_K} = m_{SO_4^=} + m_{HSO_4^-} + m_{CaSO_4} + m_{CaSO_4(S)} + m_{MgSO_4} + m_{NaSO_4^-} \quad (32)$$

$$\mathrm{Total}_{N_2O_5/L_K} = \left(m_{NO_3^-} + m_{CaNO_3^+} + m_{NaNO_3}\right)/2 \quad (33)$$

$$\mathrm{Total}_{CaO/L_K} = m_{Ca^{++}} + m_{CaOH^+} + m_{CaSO_3} + m_{CaCO_3} + m_{CaHCO_3^+} + m_{CaSO_4} + m_{CaNO_3^+} + m_{CaCO_3(S)} + m_{CaSO_4(S)} + m_{CaSO_3(S)} + m_{Ca(OH)_2(S)} \quad (34)$$

$$\mathrm{Total}_{MgO/L_K} = m_{Mg^{++}} + m_{MgOH^+} + m_{MgSO_3} + m_{MgHCO_3^+} + m_{MgCO_3} + m_{MgSO_4} + m_{Mg(OH)_2(S)} + m_{MgCO_3(S)} + m_{MgSO_3(S)} \quad (35)$$

$$Total_{Na_2O/L_K} = \left(m_{Na^+} + m_{NaOH} + m_{NaCO_3^-} + m_{NaHCO_3} + m_{NaSO_4^-} + m_{NaNO_3}\right)/2 \qquad (36)$$

$$Total_{HCl/L_K} = m_{Cl^-} \qquad (37)$$

the electroneutrality equation

$$m_{H^+} + 2m_{Ca^{++}} + m_{CaOH^+} + m_{CaHCO_3^+} m_{CaNO_3^+} + m_{Mg^{++}}$$

$$+ m_{MgHCO_3^+} + m_{Na^+} - m_{OH^-} - m_{HSO_3^-} - 2m_{SO_3^=} - 2m_{SO_4^-}$$

$$- m_{HCO_3^-} - 2m_{CO_3^=} - m_{NO_3^-} - m_{HSO_4^-} - m_{NaCO_3^-} - m_{NaSO_4^-}$$

$$- m_{Cl^-} = 0 \qquad (38)$$

and finally the water balance equation:

$$Total_{H_2O/L_K} = 55.50622 + \tfrac{1}{2}\left(m_{H^+} + m_{OH^-} + m_{HSO_3^-} + m_{HSO_4^-} + m_{HCO_3^-} + 2m_{H_2SO_3} + 2m_{H_2CO_3} + m_{CaOH^-}\right.$$

$$m_{CaHCO_3^+} + m_{MgOH^+} + m_{MgHCO_3^+} + m_{NaOH}$$

$$\left. + m_{NaHCO_3} - m_{Cl^-}\right) + m_{Ca(OH)_2(S)}$$

$$+ m_{Mg(OH)_2(S)} + \tfrac{1}{2}m_{CaSO_3(S)} + h_1 m_{CaSO_4(S)}$$

$$+ h_2 m_{MgSO_3(S)} + h_3 m_{MgCO_2(S)} \qquad (39)$$

The number of equations and unknowns that are solved with this program depends upon the number of key species input. If for example, SO_3 is zero, then all equations and unknowns involving this input are omitted from the calculations. If all of the input species are zero then only the trivial case involving m_{OH^-} and m_{H^+} is solved.

4.0 PROGRAM OPTIONS

A number of options have been incorporated into the program that modify either the specifications of inputs or unknown variables. These options are utilized in the process simulation model and in the analysis of experimental data.

A. Suppression of Solids Precipitation

This option simply deletes the seven solid species and precipitation equations from the calculations.

B. Partial Pressure of SO_2

In the basic mode of operation the partial pressure of SO_2 is computed after convergence by

$$P_{SO_2} = a_{H_2SO_3} / \gamma_{H_2O} K_{P\,SO_2}$$

where $K_{P\,SO_2}$ is the partial pressure constant

However, there is an option where P_{SO_2} may be specified as an input and the program computes $m_{H_2SO_3}$ and $N_{T\,SO_2}$

C. Partial Pressure of CO_2

The same option as is discussed in (B) is available for CO_2. These two may be used separately or together.

D. Weight Percent Solids

The fraction of solids in an aqueous solution is given by

$$X_s = \frac{W_{ES} + W_{NES}}{W_{NT} + W_{NES}}$$

where

W_{ES} = weight equilibrium solids
W_{NES} = weight nonequilibrium solids
W_{NT} = weight total input species

For this option, the weight fraction of solids, X_s, replaces $N_{T\,H_2O}$ as an input. The constant W_{NES} is also input. Since W_{ES} and the unknown $N_{T\,H_2O}$ are functions of the molecular water L_K, this option merely replaces equation (39) with the one given above. After convergence the total moles of water is printed.

E. Specified pH

Normally the pH of the solution is computed and printed after convergence; however, an option is provided to specify the pH of the solution. Since this eliminates one unknown, the electroneutrality equation (38) is not used in the calculations. It is computed after convergence to give an error estimate of the size of unknown species that might be present in the solutions. This option is used primarily in the analysis of test data.

5.0 OPERATING INSTRUCTIONS

A. Calling Sequence

CALL SOLNEQ (X, NS, CM, PP, TK, IOPT)

B. Arguments

X = An array containing the initial guesses for the molalities, when the routine returns X contains the solution.

NS = 1, No solids are considered in the solution.
= 0, Solids are computed.

CM = An array containing the input moles.
CM(1) = Moles of SO_2
CM(2) = Moles of CO_2
CM(3) = Moles of SO_3
CM(4) = Moles of N_2O_5
CM(5) = Moles of CaO
CM(6) = Moles of MgO
CM(7) = Moles of Na_2O
CM(8) = Moles of HCl
CM(9) = Moles of H_2O or weight fraction of solids (X_s)
CM(10) = Moles of W_{NES}, or pH

if CM(I) = 0, then variables and equations are deleted from the solutions.

PP	=	An array containing input partial pressure
PP(1)	=	pSO_2 (atm)
PP(2)	=	pCO_2 (atm)
TK	=	Temperature (Kelvin)
IOPT	=	0 Use CM(1) and CM(2) as inputs, compute PP(1) and PP(2).
	=	1 Use PP(1) and CM(2) as inputs, compute CM(1) and PP(2).
	=	2 Use CM(1) and PP(2) as inputs, compute PP(1) and CM(2).
	=	3 Use PP(1) and PP(2) as inputs, compute CM(1) and CM(2).
	=	4 Use CM(9) = X_s and CM(10) as inputs, compute H_2O.
	=	8 Use CM(10) = pH as an input, compute the error in the electroneutrality equation.

C. Diagnostic Messages and Error Stops

If the number of equations does not match the number of variables, the routine exits via CALL EXIT.

D. External References

There are two routines subordinated to SOLNEQ:

CTEMP
NOLIN

6.0 REFERENCES

The description of the equilibrium program may be found in Volume I, Section V of Radian Corporation Final Report for Contract CPA 22-69-138, "A Theoretical Description of the Limestone Wet Scrubbing Process", 9 June 1970. The equilibrium constant and activity coefficient formulation is also given there. Radian Technical Note 200-004-02 discusses the activity of water.

5.0 SAMPLE INPUT DECK AND PROGRAM LISTINGS

SAMPLE INPUT DECK LISTING

```
@N XQT PROSIM
      PROTOTYPE SIMULATION NO. 2-B
 1   FLUGAS    -1
 2   WTRMKP   -14
 3   CLRHTR     1    -2
 4   SCRUBR    -3    -6     2     15
 5   CLRHTR     3    -4
 6   DIVDER     7    -8    -9
 7   CLRFYR     9   -10   -11
 8   FILTER    11   -12   -13
 9   EQMIXR     8    10    12    14   -15
10   OVALMB     2    14    -3   -13
11   FLTRBM    13
12   STKGAS     5
13   PMPFAN     4    -5
@ EOF
     1,2,3(10,8(7,6,9,4))5,11,13,12*
 3  225.       .144
 4   8.        1400.     1.        .2          .2
 5  250.       .144
 6 62118.5
 7  15.     8.        13.       14.          .01
 8   .6        .3        .01
10  .9    5.24013-04  .1        1.           .5          .288        .75        .75
13  14.696         .85
@ EOF
100000.        275.      14.696
  .002      .145          4.5E-4      5.E-5    .03           0.    .7425       0.
  .08
  31.3624
 1.
  40.03
  1.121E-3.991423      7.456E-3
   .4          .4          .4        .4          .4        .4         .4
   .35         .35         .35       .35         .35       .35        .35

@@ *************************      REMOTE STOP      ****************************
```

```
@I FOR ARTWRK
      SUBROUTINE ARTWRK
      INCLUDE CMMN.LIST
      DIMENSION LB(4)
      DATA LB(1)/'REHEAT'/LB(2)/'ER    '/LB(3)/'COOLER'/LB(4)/'      '/
C
      WRITE(6,100)
      CALL DATIME
      WRITE(6,101) LABEL
      WRITE(6,200)
      WRITE(6,300)
      WRITE(6,400) PRD(99),PRD(100)
      WRITE(6,500)
      IS=1
      IRD=97
      DO 1 I=1,2
      WRITE(6,600) LB(IS),LB(IS+1),PRD(IRD)
      IS=IS+2
      IRD=IRD+1
    1 CONTINUE
  100 FORMAT(1H1)
  101 FORMAT(1H+,45X,13A6)
  200 FORMAT(45X,' EQUIPMENT OUTPUTS'/)
  300 FORMAT(10X,'I.D. FAN POWER REQUIEMENTS;')
  400 FORMAT(10X,'HORSEPOWER =',1PE10.4,4X,'KILOWATTS =',1PE10.4)
  500 FORMAT(10X,  'COOLER/REHEATER;')
  600 FORMAT(1H ,25X,A6,A2,' HEAT DUTY =',F10.2,'CAL/SEC')
      RETURN
      END
```

```
      SUBROUTINE ASTAR(X,PARTL)
C **********************************************************************
C *    THIS ROUTINE  MULTIPLIES THE DERIVATIVE  MATRIX BY ITS          *
C *    TRANSPOSE AND THEN SCALES THE PRODUCT                           *
C **********************************************************************
      COMMON /GRAD/PD(50,50),AS(50,50),SQ(50)
      COMMON /LIMS/NF
      DOUBLE PRECISION SUM
      DOUBLE PRECISION PD,AS,SQ,X
      DIMENSION X(1)
C
      CALL PARTL(X)
      DO 1 I=1,NF
      SUM=0.
      DO 2 K=1,NF
      SUM=SUM+PD(K,I)*PD(K,I)
    2 CONTINUE
      SQ(I)=SQRT(SUM)
      IF(SQ(I).LT.1.E-4) SQ(I)=1.E-4
      AS(I,I)=1.
    1 CONTINUE
C
      NF1=NF-1
      DO 3 I=1,NF1
      JS=I+1
      DO 4 J=JS,NF
      SUM=0.
      DO 5 K=1,NF
      SUM=SUM+PD(K,I)*PD(K,J)
    5 CONTINUE
      AS(I,J)=SUM/(SQ(I)*SQ(J))
      AS(J,I)=AS(I,J)
    4 CONTINUE
    3 CONTINUE
      RETURN
      END
```

```
@I FOR CLRFYR
      SUBROUTINE CLRFYR(P,S)
C ****************************
C *       THIS ROUTINE ACTS    *
C *       A SOLID-LIQUID       *
C *       SEPERATOR            *
C ****************************
      INCLUDE CMMN,LIST
      LOCSL=ISEQ(NL)
      LIS=ISTM(LOCSL,1)
      LOS1=ISTM(LOCSL,6)
      LOS2=ISTM(LOCSL,7)
      LSF=P(1)
      LSR=P(2)
      LFB=P(3)
      LWM=P(4)
      KEY1=SV(LOS1,2)
      IF(KEY1.NE.0) GO TO 1
      PRD(28)=(1.-P(5))*100.
      RETURN 2
    1 CONTINUE
      SV(LOS1,5)=SV(LOS2,5)
      SV(LIS ,5)=SV(LOS2,5)
      SV(LOS1,49)=SV(LOS2,49)
      SV(LIS,49)=SV(LOS2,49)
      SV(LIS,48)=SV(LOS2,48)
      SV(LOS1,48)=SV(LOS2,48)
C
      DO 2 I=51,81
      SV(LOS1,I)=SV(LOS2,I)
      SV(LIS ,I)=SV(LOS2,I)
    2 CONTINUE
C
      SV(LIS,50)=SV(LSF,50)+SV(LFB,50)-SV(LWM,50)-SV(LSR,50)
      SV(LOS1,50)=SV(LIS ,50)-SV(LOS2,50)
C
      XWS=SV(LOS2,100)
      R3=XWS/(1-XWS)
      R1=(R3/(1.-P(5)))*(SV(LOS2,50)/SV(LIS,50))
      R2=P(5)*R1*(SV(LIS,50)/SV(LOS1,50))
      XWSF=R1/(1+R1)
      SV(LIS,100)=XWSF
```

```
      XWSL=R2/(1+R2)
      SV(LOS1,100)=XWSL
C
      DO5 I=101,119
      SV(LOS1,I)=((XWS -XWSF)/(XWSF-XWSL))*(XWSL/XWS )*SV(LOS2,I)
      SV(LIS,I)=SV(LOS1,I)+SV(LOS2,I)
    5 CONTINUE
C
      CALL TOEQSP(LIS)
      CALL TOEQSP(LOS1)
      RETURN 2
      END
```

```
@I FOR CLRHTR
      SUBROUTINE CLRHTR(P,$)
C ***********************************
C *      THIS ROUTINE SIMULATES A   *
C *      HEAT EXCHANGER             *
C ***********************************
      INCLUDE CMMN,LIST
C
      LOCSV=ISEQ(NL)
      LIS=ISTM(LOCSV,1)
      LOS=ISTM(LOCSV,6)
      KEY1=SV(LOS,2)
      IF(KEY1.NE.0) GO TO 1
      IRD=97
      IUSE=IUSE+1
      IF(IUSE.NE.1) GO TO 3
      PRD(30)=P(1)
      PRD(62)=P(2)
      GO TO 4
    3 PRD(31)=P(1)
      PRD(64)=P(2)
    4 CONTINUE
      RETURN 2
    1 CONTINUE
      SV(LOS,5)=(P(1)+459.688)/1.8
      SV(LOS,6)=SV(LIS,6)-P(2)/14.696
      DO 2 I=10,120
      SV(LOS,I)=SV(LIS,I)
    2 CONTINUE
      CALL DHGS(LIS,H1)
      CALL DHGS(LOS,H2)
      SV(LOS,7)=H2
      SV(LIS,7)=H1
      Q=H2-H1
      PRD(IRD)=Q
      IRD=IRD+1
      RETURN 2
      END
```

```
@I FOR CONVCO
      SUBROUTINE CONVCO(X,Y,YA,S,IL)
C
      IF(IL.NE.0) GO TO  1
      X0=X
      Y0=Y
      X=X0+(YA-Y)/S
      IF(X.LT.0.) X=0.
      RETURN
    1 CONTINUE
      SL=(Y-Y0)/(X-X0)
      X1=X
      X=X0+(YA-Y0)/SL
      Y0=Y
      X0=X1
      RETURN
      END
```

```
@I FOR CONVSO
      SUBROUTINE CONVSO(X,XMX,XMN,Y,YMX,YMN,YO,IC)
C
      DY=Y-YO
      IF(IC.EQ.0) XO=X
      IF(ABS(1-X/XO).LT.1.E-3) GO TO 7
      XO=X
      IF(DY)1,2,3
    1 IF(Y.LT.YMN) GO TO 4
      YMN=Y
      XMN=X
      IN=IN+1
      IP=0
      GO TO 5
    4 X=.9*X
      GO TO 2
    3 IF(Y.GT.YMX) GO TO 6
      YMX=Y
      XMX=X
      IP=IP+1
      IN=0
      GO TO 5
    6 X=1.1*X
      GO TO 2
    7 CONTINUE
      IF(IC.EQ.0) GO TO 8
      SL=(Y1-Y)/(X1-XO)
    8 CONTINUE
      IC=1
      X=XO-DY/SL
      IF(ABS(1.-X/XO).GT.1.E-2)  X=(X+XO)/2.
      Y1=Y
      X1=XO
      XO=X
      GO TO 2
    5 SL=(YMX-YMN)/(XMX-XMN)
      X=(YO-YMN)/SL+XMN
      DX=X-XO
      SL=(Y-YO)/(XO-X)
      LC=LC+1
      IF(LC.EQ.4) IC=0
      X1=X
```

```
2 RETURN
  END
```

```
      SUBROUTINE CTEMP(TK)
      COMMON /PPRESS/PK(2),L1,L2
      COMMON/FUNC/F(50),CK(50),CT(10)
      COMMON/TEMPER/AC,BC,CPK(2),NHY(5),WS(10)
      DIMENSION A(31),B(31),C(31),D(31),LB(62),EK25(31)
      DOUBLE PRECISION F,CK,CT
      DATA (LB(I),I=1,60)/'H20    ','       ','H2SO3 ','       ','HSO3  ','
     1       ','HSO4  ','       ','H2CO3 ','       ','HCO3  ','       ','CAO
     2H+ ','       ','CASO3 ','       ','CACO3 ','       ','CAHCO3','+
     3','CASO4 ','       ','CANO3+','       ','CACO3(','S)    ','       '
     4CASO4(','S)      ','CASO3(','S)      ','CA(OH)','2(S)  ','MGOH+ ','
     5  ','MGSO3 ','       ','MGHCO3','+       ','MGSO4 ','       ','MGCO3
     6','       ','MG(OH)','2(S)  ','MGCO3(','S)      ','NAOH  ','       ','
     7NACO3-','       ','NAHCO3','       ','NASO4-','       ','NANO3 ','
     8  ','PP(SO2',')      ','PP(CO2',')       '/
      DATA (EK25(I),I=1,30)/  .10129E-13,  .12957E-01,  .62207E-07,
     *  .10396E-01,  .44513E-06,  .46817E-10,  .42229E-01,  .39999E-03,
     *  .62996E-03,  .55002E-01,  .48969E-02,  .484350   ,  .48695E-08,
     *  .24000E-04,  .84004E-07,  .56287E-05,  .26001E-02,  .11997E-02,
     *  .110010   ,  .56636E-02,  .39999E-03,  .12552E-10,  .21500E-04,
     *  3.71540   ,  .54002E-01,  1.77830   ,  .190240   ,  2.51190   ,
     *  1.21580   ,  .34215E-01/
      DATA (A(I),I=1,30)  /   4471.0   ,  -843.67   ,  -633.84   ,
     *   475.14   ,   3404.7   ,   2902.4   ,  -273.00   ,  -504.80   ,
     *  -475.48   ,  -301.85   ,   2572.1   ,  -1245.0   ,  -1560.0   ,
     *   4944.0   ,   .00000   ,   765.60   ,  -517.99   ,  -432.50   ,
     *  -235.08   ,  -1057.9   ,  -504.80   ,  -480.85   ,  -955.14   ,
     *   .00000   ,  -303.41   ,   .00000   ,  -241.00   ,   .00000   ,
     *  -1370.0   ,  -1015.0   /
      DATA (B(I),I=1,30)  /   .00000   ,   .00000   ,   .00000   ,
     *   .00000   ,   .00000   ,   .00000   ,   .00000   ,   .00000   ,
     *   .00000   ,   .00000   ,   23.150   ,   .00000   ,   .00000   ,
     *   37.745   ,   .00000   ,   4.470E-7 ,   .00000   ,   .00000   ,
     *   .00000   ,   .00000   ,   .00000   ,   .00000   ,   .00000   ,
     *   .00000   ,   .00000   ,   .00000   ,   .00000   ,   .00000   ,
     *   .00000   ,   .00000   /
      DATA (C(I),I=1,30)  /   .1706E-01,   .00000   ,   .00000   ,
     *  .18222E-01,  .32786E-01,  .23795E-01,   .00000   ,   .00000   ,
     *   .00000   ,   .00000   ,   .00000   ,   .00000   ,   .00000   ,
     *   .00000   ,   .00000   ,  .0171245  ,   .00000   ,   .00000   ,
     *   .00000   ,   .00000   ,   .00000   ,   .00000   ,   .00000   ,
     *   .00000   ,   .00000   ,   .00000   ,   .00000   ,   .00000   ,
```

```
     *    .00000    ,    .00000    /
      DATA (D(I),I=1,30)   /   6.0875   ,  -4.7171   ,  -3.3320   ,
     *   5.0435   ,   14.843   ,    6.4980   ,  -2.2900   ,  -5.0910   ,
     *  -4.7954   ,  -2.2720   ,    63.600   ,  -4.4900   ,  -13.880   ,
     *   105.36   ,  -7.0757   ,  2.424      ,  -4.3223   ,  -4.3715   ,
     *  -1.7470   ,  -5.7950   ,  -5.0910    ,  -12.514   ,  -7.8710   ,
     *   .57000   ,  -2.2852   ,   .25000    ,  -1.5290   ,    .40000  ,
     *  -4.5100   ,  -4.8700   /
      DATA WS(1)/74.10/WS(2)/100.09/WS(3)/120.14/WS(4)/129.14/WS(6)
     */58.34/WS(7)/84.33/
      DATA A(31)/802.2/D(31)/-1.583/LB(61)/'MGSO3( '/LB(62)/'S)     '/
      DATA EK25(31)/5.3272E-5/
C
  100 FORMAT(1H1)
  101 FORMAT(9X,A6,A4,6(1PE10.4,5X))
  102 FORMAT(9X,'REACTANT     K(25 C)',8X,'K(TEMP)', 10X,'A',14X,'B',14X,
     *'C',14X,'D'/)
  103 FORMAT(44X,  'EQUILIBRIUM CONSTANTS')
  104 FORMAT(//,21X,'THIS PROGRAM DEVELOPED BY RADIAN CORP. UNDER NAPCA/
     *HEW SPONSORSHIP')
  107 FORMAT(35X,  ' TEMPERATURE',F10.3,'   DEGREES  C'//)
  108 FORMAT(/17X, 'HYDRATES IF PRECIPITATED ARE MGCO3*',I1,'H2O,MGSO3*'
     *,I1,'H2O,CASO3*(1/2)H2O')
  109 FORMAT(1H+,81X,',CASO4*2H2O')
C
      IF(TOLD.EQ.TK) RETURN
      TC=TK-273.16
      WRITE(6,100)
      WRITE(6,107) TC
      WRITE(6,103)
      WRITE(6,102)
      TC2=TC*TC
      DC=87.740-0.40008*TC+9.398E-4*TC2-1.410E-6*TC2*TC
      AC=1.8246E6/(DC*TK*SQRT(DC*TK))
      AC=2.3026*AC
      BC=50.29/SQRT(DC*TK)
C
      DM=2.3025851
      DO 1 I=1,28
      CK(I)=DM*(-A(I)/TK-B(I)*ALOG10(TK)-C(I)*TK+D(I))
      IF(I.EQ.14.AND.TC.GT.40.) CK(14)=DM*(-701.81/TK-1.7288E-6*ALOG10(T
     *K)-.0195566*TK+3.7175)
      IF(I.NE.23  GO  O 3
```

```
      CK(29)=DM*(-A(31)/TK+D(31))
      IF(TC.GT.38.) CK(29)=DM*(955.2/TK-7.195)
      EK=EXP(CK(29))
      WRITE(6,101) LB(61),LB(62),EK25(31),EK,A(31),B(31),C(31),D(31)
    3 CONTINUE
      EK=EXP(CK(I))
      IS=2*I-1
      WRITE(6,101) LB(IS),LB(IS+1),EK25(I),EK,A(I),B(I),C(I),D(I)
    1 CONTINUE
C
      DO 2 I=29,30
      TEMP =DM*(-A(I)/TK-B(I)*ALOG10(TK)-C(I)*TK+D(I))
      EK=EXP(TEMP)

      J=I-28
      CPK(J)=EK
      IS=2*I-1
      WRITE(6,101) LB(IS),LB(IS+1),EK25(I),EK,A(I),B(I),C(I),D(I)
    2 CONTINUE
      NHY(1)=2
      IF(TC.GT.40.) NHY(1)=0
      NHY(2)=5
      IF(TC.GT.25.) NHY(2)=3
      NHY(3)=6
      IF(TC.GT.38.) NHY(3)=3
      WS(5)=136.14+NHY(1)*18.016
      WS(8)=84.33+NHY(2)*18.016
      WS(9)=104.38+NHY(3)*18.016
      WRITE(6,108) NHY(2),NHY(3)
      IF(NHY(1).NE.0) WRITE(6,109)
      WRITE(6,104)
      TOLD=TK
      RETURN
      END
```

```
@N FOR DELTA
      SUBROUTINE DELTA(DELT,XL)
C ********************************************************************
C *    THIS ROUTINE FINDS THE CORRECTION VECTOR                        *
C ********************************************************************
      COMMON /GRAD/PD(50,50),AS(50,50),SQ(50)
      COMMON /RESID/ GS(50)
      COMMON/FUNC/F(50),CK(50),CT(10)
      COMMON /LIMS/NF
      DOUBLE PRECISION A,B,DETRM
      DOUBLE PRECISION PD,AS,SQ,GS,DELT
      DOUBLE PRECISION F,CK,CT
      DIMENSION A(50,50),B(50,1),DELT(1)
C
      AM=10.
      DO 1 I=1,NF
      B(I,1)=0.
      B(I,1)=GS(I)
      DO 2 J=1,NF
      A(I,J)=0.
      A(I,J)=AS(I,J)
    2 CONTINUE
    1 CONTINUE
C
      DO 3 I=1,NF
      A(I,I)=A(I,I)+XL
    3 CONTINUE
C
      CALL MATIN(A,NF,B,1,DETRM)
C
      AMAG=0.
      DO 4 I=1,NF
      DELT(I)=B(I,1)/SQ(I)
      AMAG=AMAG+DELT(I)*DELT(I)
    4 CONTINUE
      AMAG=SQRT(AMAG)
      IF(AMAG.LT.AM) RETURN
      DO 7 I=1,NF
      DELT(I)=AM*DELT(I)/AMAG
    7 CONTINUE
      RETURN
      END
```

```
@I FOR DHTOT/S,DHTOT/S,DHTOT/R
      SUBROUTINE  DHTOT(L,DH,IFLAG)
C
C
C
C
C
C
C
C     ******************************************************************
C     *                            ENTHALPY                            *
C     *   THIS SUBROUTINE CALCULATES THE ENTHALPY OF A GAS-SOLID-LIQUID*
C     *   STREAM RELATIVE TO SOME REFERENCE STATE. THE REFERENCE       *
C     *   TEMPERATURE WILL BE TAKEN AS  25 C (298.16), AND THE         *
C     *   REFERENCE OF ELEMENTS AND H+ HAVING ZERO ENTHALPY WILL       *
C     *   BE TAKEN. THE COMPOSITIONS AND AMOUNTS OF EACH PHASE ARE     *
C     *   TAKEN AS KNOWN. THE NECESSARY THERMOCHEMICAL DATA ARE TAKEN  *
C     *   FROM THE LIWS DATA BASE. THE SUBROUTINE IS WRITTEN IN TERMS  *
C     *   OF THREE PARTS-ONE WHICH CALCULATES THE ENTHALPY OF THE GAS  *
C     *   PHASE   , ONE WHICH CALCULATES THE ENTHALPY OF THE SOLID-    *
C     *   PHASE ,AND ONE WHICH CALCULATES THE ENTHALPY OF LIQUID PHASE.*
C     ******************************************************************
C
C     ENTRY  DHG(L,DH)       -THIS ENTRY COMPUTES GAS ENTHALPY PHASE(DH)
C                             ONLY,FOR L TH STREAM (IFLAG=1  )
C
C     ENTRY  DHL(L,DH)       -THIS ENTRY COMPUTES LIQUID ENTHALPY PHASE
C                            (DH)ONLY,FOR L TH STREAM (IFLAG=2 )
C
C     ENTRY  DHS(L,DH)       -THIS ENTRY COMPUTES SOLID ENTHALPY  PHASE
C                             (DH) ONLY,FOR L TH STREAM (IFLAG=3  )
C
C     ENTRY  DHGS(L,DH)      -THIS ENTRY COMPUTES GAS-SOLID ENTHALPY
C                             PHASE (DH) FOR L TH STREAM (IFLAG=4   )
C
C     ENTRY DHSL(L,DH)       -THIS ENTRY COMPUTES SOLID-LIQUID ENTHALPY
C     .                       PHASE (DH)FOR L TH STREAM (IFLAG=5  )
C
C     ENTRY  DHSLG(L,DH)     -THIS ENTRY COMPUTES SOLID-LIQUID-GAS
C                             ENTHALPY(DH)FOR L TH STREAM (IFLAG=6 )
C
```

```
C     IFLAG                     -IF ENTERED THROUGH STANDARD ENTRY,VALUE
C                                OF IFLAG DETERMINES COMPUTE OPTION AS
C                                INDICATED ABOVE.
C
      INCLUDE LCMN,LIST
      INCLUDE CMMN,LIST
C
C
      REAL NGS,NW,MW,MI
C
C
     ODEFINE          D(T)        = 87.740 -.40008*(T-273.16)
     1                              +9.398E-4*(T-273.16)**2
     2                              -1.410E-6*(T-273.16)**3
     ODEFINE          DDT(T)      = -.40008 + 1.8796E-3*(T-273.16)
     1                               -4.23E-6 * (T-273.16)**2
      DEFINE          TAU(T)      = DDT(T)/D(T)    + 1.0/T
     ODEFINE          DL(T)       = +.99995282
     1                               +.46726616E-4 *  (T-273.16)
     2                               -.74105074E-5 *  (T-273.16)**2
     3                               +.41079583E-7 *  (T-273.16)**3
     4                               -.13370708E-9 *  (T-273.16)**4
     ODEFINE          ALPHA(T)   =-((+.46726616E-4)
     1                            +2.*(-.74105074E-5)*(T-273.16)
     2                            +3.*(+.41079583E-7)*(T-273.16)**2
     3                            +4.*(-.13370708E-9)*(T-273.16)**3)/DL(T)
     ODEFINE          A(T)        = (1.8246E+06)* SQRT(DL(T))/
     1                                SQRT(D(T)*T )**3
     ODEFINE          B(T)        = (50.29) * SQRT (DL(T))/
     1                                SQRT( D(T)*T)
      DEFINE          BETA(T,I) = ALIT(I)*B(T)*SQRT(CAPI)
C
C     ******************************************************************
C
C
      IF (IFLAG.LT.1.OR.IFLAG.GT.6)  GO TO 3
      GO TO 4
    3 CONTINUE
      PRINT 110,IFLAG
  110 FORMAT(25X,  'ERROR STOP IN ENTHALPY ROUTINE-ENTRY DH-IFLAG=',I5/)
      STOP
C
C
```

```
      ENTRY  DHG(L,D
C
      IFLAG=1
      GO TO 4
C
C
      ENTRY  DHL(L,DH)
C
      IFLAG=2
      GO TO 4
C
C
      ENTRY DHS(L,DH)
C
      IFLAG=3
      GO TO 4
C
C
      ENTRY DHGS(L,DH)
C
      IFLAG =4
      GO TO 4
C
C
      ENTRY DHSL(L,DH)
C
      IFLAG =5
      GO TO 4
C
C
      ENTRY DHSLG(L,DH)
      IFLAG =6
    4 CONTINUE
      DHSOL=0.0
      DHGAS=0.0
      DHLIQ=0.0
      IF(IFLAG.EQ.1.OR.IFLAG.EQ.4.OR.IFLAG.EQ.6 ) GO TO 5
      GO TO 6
    5 CONTINUE
C
C
C     *******************************************************************
C     *                                                                 *
```

```
C      *               PART THAT COMPUTES THE ENTHALPY OF THE GAS           *
C      *                              STREAM                                *
C      **********************************************************************
C
C
       TI>298.16
       SUM=0.0
       TGS=SV(L,5)
       DO 1 JG=1,9
       I=10 + JG
       TERM1= DHF(JG)
       NGS= SV(L,I)
       IF(NGS.EQ.0) GO TO 1
      0TERM2=(TGS-TI)*(ACP(JG) + (TGS+TI)*.5*BCP(JG)
      1              +((TGS*TGS  +TGS*TI + TI*TI)/3.0)* CCP(JG))
       SUM=SUM  + NGS*(TERM1 + TERM2)
     1 CONTINUE
       DHGAS=SUM
C
C
     6 CONTINUE
       IF(IFLAG.GT.2)  GO TO 7
       GO TO 8
     7 CONTINUE
C
C
C      **********************************************************************
C      *                                                                    *
C      *               PART THAT COMPUTES THE ENTHALPY OF THE SOLIDS        *
C      *                           IN THE STREAM                            *
C      *                                                                    *
C      **********************************************************************
C
C
       TI>298.16
       TGS>SV¢L,5I
       SUM=0.0
       DO 2 JS=21,39
       I=80 + JS
       NGS=SV(L,I)
       TERM1= DHF(JS)
       IF(NGS.EQ.0) GO TO 2
      0TERM2=(TGS-TI)*(ACP JS   + TGS+TI)*.5*BC> JS
```

```
                 - ..C/(''GS*T    * DCP(JS
      SUM=SUM + NGS*(TERM1 + TERM2)
    2 CONTINUE
      DHSOL=SUM
C
C
C
C
    8 CONTINUE
      IF(IFLAG.EQ.2.OR.IFLAG.GT.4) GO TO 9
      GO TO 10
    9 CONTINUE
C
C
C     ******************************************************************
C     *                                                                *
C     *              PART  THAT COMPUTES THE ENTHALPY OF THE           *
C     *                    LIQUIDS IN THE STREAM                       *
C     *
C     *
C     ******************************************************************
C
C
C
      TI=298.16
      MW= 55.50622
      NW= SV(L,50)/18.016
      SUM=0.0
      T= SV(L,5)
      R=  1.9872
      DO 11 JL=50,81
      TERM1=DHF(JL)
      MI>SV¢L,JLI
      IF(JL.EQ.50) MI=MW
      IF( MI.EQ.0 )  GO TO 11
     OTERM2=(T-TI)*(ACP(JL)+ (  T+TI)*.5 *BCP(JL)
     1             +((T * T + T *TI + TI*TI)/3.0)* CCP(JL))
      SUM =SUM  + MI *( TERM1 + TERM2)
   11 CONTINUE
      SUM= SUM * ( SV(L,50)/1000.0)
      DHLIQ= SUM
C
C
```

```
      CAPI=0.0
      DO 12  I=51,81
      IF( ZE(I).EQ.0)    GO TO 12
      CAPI = CAPI + ZE(I)*ZE(I)*SV(L,I)
   12 CONTINUE
      CAPI>CAPI/2.0
      SUM1=0.0
      SUM2=0.0
      DO 13  I=51,81
      SUM1=SUM1 + SV(L,I)
      IF(ZE(I).EQ.0) GO TO 14
      BET =BETA(T,I)
      X1=BET - 2.0*ALOG(1.0+BET) + BET/(1.0+BET)
      X2= A(T)*ZE(I)*ZE(I)*BLIT(I) +UCAP(I)
     0TERM= SV(L,I)*((-A(T)*ZE(I)*ZE(I)*SQRT(CAPI)/BET**3)*X1
     1                + (CAPI/2.0)*X2)
      GO TO 15
   14 CONTINUE
      TERM= UCAP(I) *CAPI * SV(L,I)/2.0
   15 CONTINUE
      SUM2 = SUM2 + TERM
   13 CONTINUE
      ALOGAW>-¢2.3025851*SUM2 < SUM1I / MW
      SUM =0.0
      DO  20 I=51,81
      DUM>TAU¢TI*3.0<ALPHA¢TI
      BET=BETA(T,I)
      X1> .5 *R * T * T *A¢TI* ZE¢II*ZE¢II*¢2.3025851I
      X2> DUM*BLIT¢II*CAPI
     0X3> -SQRT¢CAPII*¢2.0*BET*TAU¢TI <DUMI/
     1              ((1.0 +BET)**2)
      TERM = X1 * ( X2 + X3 )
      SUM = SUM + TERM * SV(L,I)
   20 CONTINUE
      SUM = SUM / MW
      SUM1=0.0
      SUM2=0.0
      DO 21 I>51,81
      SUM1=SUM1 + SV(L,I)*(UCAP(I)*CAPI/2.0 + .43429448 )
      BET=BETA(T,I)
      X1= SV(L,I)*ZE(I)*ZE(I)/(BET*BET)
      X2 > -6.0* ALOG¢1.0< BETI/BET
      X3= (6.0 +9.0*BET +2.0*BET*BET)/    .0 +BET)**2)
```

```
      SUM2 =SUM2 + X1 * (X2 +X3)
   21 CONTINUE
      X1 = R*T*T*.5
      X2> 3.0*TAU¢T) < ALPHA¢T)
      X3= ALOGAW + (2.3025851 * SUM1 )/ MW
      X4=(A(T)*(TAU(T)+ALPHA(T))*SQRT(CAPI) * (2.3025851) * SUM2)/MW
      TERM = X1*( X2 * X3 - X4 )
      DHLIQ>DHLIQ < ¢TERM < SUM )* NW
C
C
   10 CONTINUE
C
C     *****************************************************************
C     *                                                               *
C     *   FOLLOWING PORTION TAKES INDIVIDUAL ENTHALPY COMPONENTS      *
C     *             CHANGES AND COMPUTES TOTAL ENTHALPY CHANGE        *
C     *                                                               *
C     *****************************************************************
C
C
      DH> DHGAS< DHSOL <DHLIQ
      P(1)=P(1)
      END
```

```
@I FOR DIVDER
      SUBROUTINE DIVDER(P,S)
C *********************************
C *       THIS ROUTINE SERVES     *
C *       AS A STREAM SPLITTER    *
C *********************************
      INCLUDE CMMN,LIST
      LOCSL=ISEQ(NL)
      LIS=ISTM(LOCSL,1)
      LOS1=ISTM(LOCSL,6)
      LOS2=ISTM(LOCSL,7)
      KEY1=SV(LOS1,2)
      IF(KEY1.NE.0) GO TO 1
      SV(LOS1,2)=1
      SV(LOS1,50)=P(1)
      RETURN 2
    1 CONTINUE
      IF(SV(LOS1,50).NE.0.) SV(LOS1,   5)=SV(LOS2,   5)
      SV(LIS ,5)=SV(LOS2,5)
      SV(LIS,49)=SV(LOS2,49)
      SV(LOS1,49)=SV(LOS2,49)
      SV(LIS,48)=SV(LOS2,48)
      SV(LOS1,48)=SV(LOS2,48)
      SV(LIS,50)=SV(LOS1,50)+SV(LOS2,50)
C
      DO 2 I=51,81
      SV(LIS ,I)=SV(LOS2,I)
      IF(SV(LOS1,50).EQ.0.) GO TO 2
      SV(LOS1,I)=SV(LOS2,I)
    2 CONTINUE
C
      DO 3 I=101,119
      SV(LIS ,I)=(SV(LIS ,50)/SV(LOS2,50))*SV(LOS2,I)
      SV(LOS1,I)=(SV(LOS1,50)/SV(LOS2,50))*SV(LOS2,I)
    3 CONTINUE
C
      DO 4I=31,39
      SV(LOS1,I)=(SV(LOS1,50)/SV(LOS2,50))*SV(LOS2,I)
      SV(LIS ,I)=(SV(LIS ,50)/SV(LOS2,50))*SV(LOS2,I)
    4 CONTINUE
      IF(SV(LOS1,50).NE.0.) SV(LOS1,100)=SV(LOS2,100)
      SV(LIS,100)=SV .OS2, 00
```

```
RETURN 2
END
```

```
@I FOR DNTLA
      SUBROUTINE DNTLA(L,XLH,XDLA,DNLA,WLA,TWLA)
      INCLUDE CMMN,LIST
      DIMENSION XLH(1),XDLA(1),DNLA(1),WLA(1),TWLA(1),TLA(9)
C
      DO 2 J=1,19
      I=100+J
      WLA(J)=SV(L,I)
    2 CONTINUE
      CALL TOSOSP(WLA,TLA)
C
      DO 1 J=1,19
      I=100+J
      WLA(J)=(1-XDLA(J))*SV(L,I)
    1 CONTINUE
      WLA(1)=SV(L,101)*(1-XLSU(1))+SV(L,101)*XLSU(1)*(1-XLH(1))
      WLA(8)=SV(L,108)*(1-XLSU(2))+SV(L,108)*XLSU(2)*(1-XLH(2))
      WLA(2)=SV(L,101)*XLSU(1)*XLH(1)*(1-XDLA(2))
      WLA(9)=SV(L,108)*XLSU(2)*XLH(2)*(1-XDLA(9))
C
      CALL TOSOSP(WLA,TWLA)
      DO 4 J=1,9
      DNLA(J)=TLA(J)-TWLA(J)
    4 CONTINUE
      RETURN
      END
```

```
@I FOR DNTLAO
      SUBROUTINE DNTLAO(L,X1,X2)
      INCLUDE CMMN,LIST
C
      P(1)=P(1)
      SV(L,31)=SV(L,104)+SV(L,113)
      SV(L,32)=SV(L,103)+SV(L,110)
      SV(L,33)=SV(L,106)+SV(L,116)
      SV(L,34)=0.
      SV(L,35)=X1*SV(L,101)+SV(L,103)+SV(L,104)+SV(L,106)
      SV(L,36)=X2*SV(L,108)+SV(L,110)+SV(L,113)+SV(L,116)
      SV(L,37)=SV(L,117)+.5*SV(L,119)
      SV(L,38)=SV(L,119)
      SV(L,39)=-.5*SV(L,119)
      RETURN
      END
```

```
      SUBROUTINE DNTSF(L,XDSF,DNS,WSF,TWSF)
      INCLUDE CMMN,LIST
      DIMENSION         DNS(1),WSF(1),TWSF(1),XDSF(1)
C
      DO 1 J=1,19
      I=100+J
      WSF(J)=(1-XDSF(J))*SV(L,I)
    1 CONTINUE
      CALL TOSOSP(WSF,TWSF)
      DO 2 J=1,9
      I=30+J
      DNS(J)=SV(L,I)-TWSF(J)
    2 CONTINUE
      RETURN
      END
```

```
@I FOR EFHOLD
      SUBROUTINE EFHOLD(P,$)
      RETURN 2
      END
```

```
@I FOR EQMIXR
      SUBROUTINE EQMIXR(P,$)
      INCLUDE CMMN.LIST
      DIMENSION PP(2)
C ****************************
C *      THIS ROUTINE MODELS
C *      THE PROCESS WATER
C *      HOLD TANK
C ****************************
      LOCSL=ISEQ(NL)
      LIS1=ISTM(LOCSL,1)
      LIS2=ISTM(LOCSL,2)
      LIS3=ISTM(LOCSL,3)
      LIS4=ISTM(LOCSL,4)
      LOS=ISTM(LOCSL,6)
      KEY1=SV(LOS,2)
      IF(KEY1.NE.0) GO TO 1
      IOPT=0
      NS=0
      WNES=0
      RETURN 2
    1 CONTINUE
C
      SV(LOS,5)=SV(LIS4,5)
      T=SV(LOS,5)
      DO 2 I=31,39
      SV(LOS,I)=SV(LIS1,I)+SV(LIS2,I)+SV(LIS3,I)+SV(LIS4,I)
    2 CONTINUE
      I=50
      SV(LOS,I)=SV(LIS1,I)+SV(LIS2,I)+SV(LIS3,I)+SV(LIS4,I)
      SV(LOS,51)=SV(LIS3,51)
      SV(LOS,52)=SV(LIS3,52)
      DO 4 J=1,29
      I=52+J
      SV(LOS,I)=(SV(LIS1,I)*SV(LIS1,50)+SV(LIS2,I)*SV(LIS2,50)+SV(LIS3,
     *I)*SV(LIS3,50)+SV(LIS4,I)*SV(LIS4,50))/SV(LOS,50)
    4 CONTINUE
      DO 5 J=1,19
      I=100+J
      SV(LOS,I)=SV(LIS1,I)+SV(LIS2,I)+SV(LIS3,I)+SV(LIS4,I)
    5 CONTINUE
      NS=0
```

```
      SUM=SV(LIS1,100)+SV(LIS2,100)+SV(LIS3,100)
      IF(SUM.EQ.0.) NS=1
C
    7 CONTINUE
      SV(LOS,35)=SV(LOS,35)-SV(LOS,101)
      SV(LOS,36)=SV(LOS,36)-SV(LOS,108)
      CALL EQUILB(LOS,IOPT,T,PP,WNES,NS)
      WRITE(6,200)
  200 FORMAT(//42X,'PROCESS WATER HOLD TANK')
      SV(LOS,50)=SV(LOS,50)*1000.
      SV(LOS,35)=SV(LOS,35)+SV(LOS,101)
      SV(LOS,36)=SV(LOS,36)+SV(LOS,108)
      RETURN 2
      END
```

```
@I FOR EQS
      SUBROUTINE EQS(W)
C ***************************************************************
C *    N FUNCTIONS OF THE  N-VECTOR X  ARE SUPPLIED HERE        *
C ***************************************************************
      COMMON/FUNC/F(50),CK(50),CT(10)
      COMMON /ACOEF/GLN(50),DGLN(50),IZ(50)
      COMMON /LIMS/NF
      COMMON/TEMPER/AC,BC,CPK(2),NHY(5),WS(10)
      COMMON /PPRESS/PK(2),L1,L2,L3,L4,REN
      COMMON /NEW/ IV(50), IE(50), CL,TH(11)
      COMMON /TPAR/ATP(7)
      DIMENSION W(1),X(50),EQ(50)
      DOUBLE PRECISION F,CK,CT,W,X,EQ
      DOUBLE PRECISION FNEQ,TH
      DOUBLE PRECISION ATP
      NAMELIST/TEST/WS,CT,TH,X/
C
      DEFINE Y(I)=EXP(X(I))
C
      DO 3 I=1,36
      X(I)=-85.
    3 CONTINUE
      X(38)=-85.
      X(39)=0
      DO 1 I=1,NF
      NV=IV(I)
      X(NV)=W(I)
    1 CONTINUE
      X(37)=LOG(CL/Y(39))
C
      TH(8)=Y(1)+2*Y(12)+Y(13)+Y(16)+Y(18)+2*Y(23)+Y(24)+Y(26)+Y(31)
      TH(9)=Y(2)+Y(3)+2*Y(4)+2*Y(5)+Y(6)+2*Y(7)+Y(8)+Y(9)+Y(33)+Y(35)
     *+Y(37)
      EQ(36)=LOG(TH(8))-LOG(TH(9))
      REN=EQ(36)
      IF(L4.NE.8) GO TO 10
      REN=TH(8)-TH(9)
      EQ(36)=X(1)+GLN(1)+CT(10)
   10 CONTINUE
C
      CALL GAMMA(X)
```

```
C
      EQ( 1)=GLN( 1)+GLN( 2)+X( 1)+X( 2)-CK( 1)-GLN(39)
      IF(L4.EQ.8) EQ(1)=X(2)+GLN(2)-CT(10)-CK(1)-GLN(39)
      IF(L1.EQ.1) GO TO 4
      EQ( 2)=GLN( 1)+GLN( 3)+X( 1)+X( 3)-GLN(10)-X(10)-CK( 2)
      TH(1)=Y(3)+Y(4)+Y(10)+Y(14)+Y(21)+Y(25)
      EQ(29)=CT(1)-LOG(TH(1))-X(39)
      GO TO 5
    4 EQ(2)=GLN(1)+GLN(3)+X(1)+X(3)-PK(1)-CK(2)-GLN(39)
      TH(1)=Y(3)+Y(4)+Y(14)+Y(21)+Y(25)+EXP(PK(1)-GLN(10)+GLN(39))
      EQ(29)=X(10)-LOG(TH(1))-X(39)
    5 CONTINUE
      EQ( 3)=GLN( 1)+GLN( 4)+X( 1)+X( 4)-GLN( 3)-X( 3)-CK( 3)
      EQ( 4)=GLN( 1)+GLN( 5)+X( 1)+X( 5)-GLN( 9)-X( 9)-CK( 4)
      IF(L2.EQ.2) GO TO 6
      EQ( 5)=GLN( 1)+GLN( 6)+X( 1)+X( 6)-GLN(11)-X(11)-CK( 5)
      TH(2)=Y(6)+Y(7)+Y(11)+Y(15)+Y(16)+Y(19)+Y(26)+Y(28)+Y(30)+Y(33)
     *+Y(34)
      EQ(30)=CT(2)-LOG(TH(2))-X(39)
      GO TO 7
    6 EQ(5)=GLN(1)+GLN(6)+X(1)+X(6)-PK(2)-CK(5)-GLN(39)
      TH(2)=Y(6)+Y(7)+Y(15)+Y(16)+Y(19)+Y(26)+Y(28)+Y(30)+Y(33)+Y(34)
     *+EXP(PK(2)-GLN(11)+GLN(39))
      EQ(30)=X(11)-LOG(TH(2))-X(39)
    7 CONTINUE
      EQ( 6)=GLN( 1)+GLN( 7)+X( 1)+X( 7)-GLN( 6)-X( 6)-CK( 6)
      EQ( 7)=GLN(12)+GLN( 2)+X(12)+X( 2)-GLN(13)-X(13)-CK( 7)
      EQ( 8)=GLN(12)+GLN( 4)+X(12)+X( 4)-GLN(14)-X(14)-CK( 8)
      EQ( 9)=GLN(12)+GLN( 7)+X(12)+X( 7)-GLN(15)-X(15)-CK( 9)
      EQ(10)=GLN(12)+GLN( 6)+X(12)+X( 6)-GLN(16)-X(16)-CK(10)
      EQ(11)=GLN(12)+GLN( 5)+X(12)+X( 5)-GLN(17)-X(17)-CK(11)
      EQ(12)=GLN(12)+GLN( 8)+X(12)+X( 8)-GLN(18)-X(18)-CK(12)
      EQ(13)=GLN(12)+GLN(7)+X(12)+X(7)-CK(13)
      ATP(1)=EQ(13)
      EQ(13)=FNEQ(EQ(13),X(19))
      EQ(14)=GLN(12)+GLN(5)+X(12)+X(5)-CK(14)+NHY(1)*GLN(39)
      ATP(2)=EQ(14)
      EQ(14)=FNEQ(EQ(14),X(20))
      EQ(15)=GLN(12)+GLN(4)+X(12)+X(4)-CK(15)+.5*GLN(39)
      ATP(3)=EQ(15)
      EQ(15)=FNEQ(EQ(15),X(21))
      EQ(16)=GLN(12)+2*GLN(2)+X(12)+2*X(2)-CK(16)
      ATP(4)=EQ(16)
```

```
      EQ(16)=FNEQ(EQ(16),X(22))
      EQ(17)=GLN(23)+GLN( 2)+X(23)+X( 2)-GLN(24)-X(24)-CK(17)
      EQ(18)=GLN(23)+GLN( 4)+X(23)+X( 4)-GLN(25)-X(25)-CK(18)
      EQ(19)=GLN(23)+GLN( 6)+X(23)+X( 6)-GLN(26)-X(26)-CK(19)
      EQ(20)=GLN(23)+GLN( 5)+X(23)+X( 5)-GLN(27)-X(27)-CK(20)
      EQ(21)=GLN(23)+GLN( 7)+X(23)+X( 7)-GLN(28)-X(28)-CK(21)
      EQ(22)=GLN(23)+2*GLN(2)+X(23)+2*X(2)-CK(22)
      ATP(5)=EQ(22)
      EQ(22)=FNEQ(EQ(22),X(29))
      EQ(38)=GLN(23)+GLN(4)+X(23)+X(4)-CK(29)+NHY(3)*GLN(39)
      ATP(7)=EQ(38)
      EQ(38)=FNEQ(EQ(38),X(38))
      EQ(23)=GLN(23)+GLN(7)+X(23)+X(7)-CK(23)+NHY(2)*GLN(39)
      ATP(6)=EQ(23)
      EQ(23)=FNEQ(EQ(23),X(30))
      EQ(24)=GLN(31)+GLN( 2)+X(31)+X( 2)-GLN(32)-X(32)-CK(24)
      EQ(25)=GLN(31)+GLN( 7)+X(31)+X( 7)-GLN(33)-X(33)-CK(25)
      EQ(26)=GLN(31)+GLN( 6)+X(31)+X( 6)-GLN(34)-X(34)-CK(26)
      EQ(27)=GLN(31)+GLN( 5)+X(31)+X( 5)-GLN(35)-X(35)-CK(27)
      EQ(28)=GLN(31)+GLN( 8)+X(31)+X( 8)-GLN(36)-X(36)-CK(28)
      TH(3)=Y(5)+Y(9)+Y(17)+Y(20)+Y(27)+Y(35)
      EQ(31)=CT(3)-LOG(TH(3))-X(39)
      TH(4)=Y(8)+Y(18)+Y(36)
      EQ(32)=CT(4)-LOG(TH(4))-X(39)+.69315
      TH(5)=Y(12)+Y(13)+Y(14)+Y(15)+Y(16)+Y(17)+Y(18)+Y(19)+Y(20)+Y(21)
     *+Y(22)
      EQ(33)=CT(5)-LOG(TH(5))-X(39)
      TH(6)=Y(23)+Y(24)+Y(25)+Y(26)+Y(27)+Y(28)+Y(29)+Y(30)+Y(33)
      EQ(34)=CT(6)-LOG(TH(6))-X(39)
      TH(7)=Y(31)+Y(32)+Y(33)+Y(34)+Y(35)+Y(36)
      EQ(35)=CT(7)-LOG(TH(7))-X(39)+.69315
      H2SO3=Y(10)
      H2CO3=Y(11)
      IF(L2.EQ.2) H2CO3=EXP(PK(2)-GLN(11)+GLN(39))
      IF(L1.EQ.1) H2SO3=EXP(PK(1)-GLN(10)+GLN(39))
      TH(10)=55.50622+.5*(Y(1)+Y(2)+Y(3)+Y(6)+Y(9)+2*H2SO3+2*H2CO3+Y(13)
     *+Y(16)+Y(24)+Y(26)+Y(32)+Y(34)-Y(37))
      TH(10)=TH(10)+Y(22)+Y(29)+.5*Y(21)+NHY(1)*Y(20)+NHY(2)*Y(30)+NHY(3
     *)*Y(38)
      IF(L3.EQ.4) GO TO 8
      EQ(39)=CT(9)-LOG(TH(10))-X(39)
      GO TO 9
    8 CONTINUE
```

```
      TH(10)=18.016*TH(10)
      TH(11)=CT(10)+Y(39)*(WS(1)*Y(22)+WS(2)*Y(19)+WS(4)*Y(21)+WS(5)*
     *Y(20)+WS(6)*Y(29)+WS(8)*Y(30)+WS(9)*Y(38))
      IF(TH(11).LT.0.) TH(11)=TH(11)-CT(10)
      EQ(39)=LOG(TH(11))-CT(9)-LOG(TH(10))-X(39)
    9 CONTINUE
C
      NF=NF
      DO 2 I=1,NF
      NE=IE(I)
      NV=IV(I)
      W(I)=X(NV)
      F(I)=EQ(NE)
    2 CONTINUE
      RETURN
      END
```

```
@I FOR EQUILB
      SUBROUTINE EQUILB(LSV,IOPT,T,PP,WNES,NS)
C *************************************
C *      THIS ROUTINE INTERFACES       *
C *      SOLNEQ WITH EQUIPMENT         *
C *      ROUTINES                      *
C *************************************
      INCLUDE CMMN.LIST
      COMMON /TRFR/TMW,WACT,PH,AZ
      DIMENSION X(50),CM(11),PP(2)
C
      P(1)=P(1)
      DO 1 I=31,39
      IX=I-30
      CM(IX)=SV(LSV,I)
    1 CONTINUE
C
      L4=AND(IOPT,4)
      IF(L4.EQ.4) CM(9)=SV(LSV,100)
      IF(L4.EQ.4) CM(10)=WNES
      IF(L4.EQ.4) CM(11)=SV(LSV,119)
C
      X(39)=SV(LSV,50)/1000.
C
      DO 2 I=51,68
      IX=I-50
      X(IX)=SV(LSV,I)
    2 CONTINUE
C
      DO 3 I=69,74
      IX=I-46
      X(IX)=SV(LSV,I)
    3 CONTINUE
C
      DO 4 I=75,81
      IX=I-44
      X(IX)=SV(LSV,I)
    4 CONTINUE
C
      X(19)=SV(LSV,103)/X(39)
      IF(T.GE.313.16) X(20)=SV(LSV,106)/X(39)
      IF(T.LT.313.16) X(20)=SV(LSV,107)/X(39)
```

```
      X(21)=SV(LSV,105)/X(39)
      X(22)=SV(LSV,102)/X(39)
      X(29)=SV(LSV,109)/X(39)
      IF(T.GE.373.16) X(30)=SV(LSV,110)/X(39)
      IF(T.LT.373.16.AND.T.GT.298.16) X(30)=SV(LSV,111)/X(39)
      IF(T.LE.298.16) X(30)=SV(LSV,112)/X(39)
      IF(T.GE.311.16) X(38)=SV(LSV,114)/X(39)
      IF(T.LT.311.16) X(38)=SV(LSV,115)/X(39)
C
      CALL SOLNEQ(X,NS,CM,PP,T,IOPT)
C
      AH2O=WACT
      SV(LSV,48)=AZ
      SV(LSV,49)=PH
      SV(LSV,31)=CM(1)
      SV(LSV,32)=CM(2)
      IF(L4.EQ.4) SV(LSV,39)=TMW
      DO 8 I=1,50
      IF(X(I).EQ.1.E-10) X(I)=0.
    8 CONTINUE
      DO 5 I=1,18
      IX=I+50
      SV(LSV,IX)=X(I)
    5 CONTINUE
C
      DO 6 I=23,28
      IX=I+46
      SV(LSV,IX)=X(I)
    6 CONTINUE
C
      DO 7 I=31,37
      IX=I+44
      SV(LSV,IX)=X(I)
    7 CONTINUE
      SV(LSV,81)=CM(8)/X(39)
C
      SV(LSV,50)=X(39)
      SV(LSV,103)=X(19)*X(39)
      IF(T.GE.313.16) SV(LSV,106)=X(20)*X(39)
      IF(T.LT.313.16) SV(LSV,107)=X(20)*X(39)
      SV(LSV,105)=X(21)*X(39)
      SV(LSV,102)=X(22)*X(39)
      SV(LSV,109)=X(29)*X(39)
```

```
      IF(T.GE.373.16) SV(LSV,110)=X(30)*X(39)
      IF(T.LT.373.16.AND.T.GT.298.16) SV(LSV,111)=X(30)*X(39)
      IF(T.LE.298.16) SV(LSV,112)=X(30)*X(39)
      IF(T.LT.311.16) SV(LSV,115)=X(38)*X(39)
      IF(T.GE.311.16) SV(LSV,114)=X(38)*X(39)
C
      RETURN
      END
```

```
@I FOR FILTER
      SUBROUTINE FILTER(P,$)
C **********************************
C *        THIS ROUTINE ACTS AS     *
C *        A SOLID -LIQUID          *
C *        SEPERATOR ONLY           *
C **********************************
      INCLUDE CMMN,LIST
      LOCSL=ISEQ(NL)
      LIS=ISTM(LOCSL,1)
      LOS1=ISTM(LOCSL,6)
      LOS2=ISTM(LOCSL,7)
      KEY1=SV(LOS1,2)
      IF(KEY1.NE.0) GO TO 1
      PRD(29)=100*P(1)
      PRD(27)=100*P(2)
      PRD(67)=(1.-P(3))*100.
      SV(LOS2,100)=P(1)
      SV(LIS,100) =P(2)
      RETURN 2
    1 CONTINUE
      SV(LOS1,5)=SV(LOS2,5)
      SV(LIS ,5)=SV(LOS2,5)
      SV(LOS1,48)=SV(LOS2,48)
      SV(LIS,48)=SV(LOS2,48)
      SV(LIS,49)=SV(LOS2,49)
      SV(LOS1,49)=SV(LOS2,49)
C
      DO 2 I=51,81
      SV(LOS1,I)=SV(LOS2,I)
      SV(LIS ,I)=SV(LOS2,I)
    2 CONTINUE
C
      R1=P(2)/(1-P(2))
      R3=P(1)/(1-P(1))
      SV(LIS,50)=(SV(LOS2,50)/(1-P(3)))*R3/R1
      SV(LOS1,50)=SV(LIS ,50)-SV(LOS2,50)
      R2=P(3)*R1*(SV(LIS,50)/SV(LOS1,50))
      XWS=R2/(1+R2)
      SV(LOS1,100)=XWS
C
      DO5 I=101,119
```

```
      SV(LOS1,I)=((P(1)-P(2))/(P(2)-XWS))*(XWS/P(1))*SV(LOS2,I)
      SV(LIS,I)=SV(LOS1,I)+SV(LOS2,I)
    5 CONTINUE
C
      CALL TOEQSP(LIS)
      CALL TOEQSP(LOS1)
      RETURN 2
      END
```

```
@I FOR FLTRBM
      SUBROUTINE FLTRBM(P,$)
      RETURN 2
      END
```

```
@I FOR FLUGAS
      SUBROUTINE FLUGAS(P,$)
      INCLUDE CMMN,LIST
      INCLUDE LCMN,LIST
      DIMENSION YFG(9),XWLS(8),XWFA(3)
      LOCSV=ISEQ(NL)
      LOS=ISTM(LOCSV,6)
      KEY=SV(LOS,2)
      IF(KEY.NE.0) RETURN 2
      READ(5,100) VIFG,TIFG,PIFG
      PRD(1)=VIFG
      PRD(2)=TIFG
      PRD(3)=PIFG
  100 FORMAT(8E10.3)
      VFG=(28.32/60.)*VIFG
      TFG=(TIFG+459.688)/1.8
      PFG=PIFG/14.696
      SV(LOS,5)=TFG
      SV(LOS,6)=PFG
      READ(5,100) YFG
      DO 1 J=1,9
      I=10+J
      K=33+J
      PRD(K)=100*YFG(J)
      SV(LOS,I)=12.187158*PFG*VFG*YFG(J)/TFG
    1 CONTINUE
      SV(LOS,10)=12.187158*PFG*VFG/TFG
      READ(5,100) WILS
      PRD(8)=WILS
      WLS=(453.6/60.)*WILS
      READ(5,100) XWLS
      DO 3 J=1,8
      K=42+J
      PRD(K)=100*XWLS(J)
    3 CONTINUE
      SV(LOS,101)=XWLS(1)*WLS/XMW(21)
      SV(LOS,108)=XWLS(2)*WLS/XMW(28)
      SV(LOS,103)=XWLS(3)*WLS/XMW(23)
      SV(LOS,110)=XWLS(4)*WLS/XMW(30)
      SV(LOS,104)=XWLS(5)*WLS/XMW(24)
      SV(LOS,113)=XWLS(6)*WLS/XMW(33)
      SV(LOS,106 =XWLS(7)*W.S/XMW(26)
```

```
      SV(LOS,116)=XWLS(8)*WLS/XMW(36)
      READ(5,100) WIFA
      PRD(25)=WIFA
      WFA=(453.6/60.)*WIFA
      READ(5,100) XWFA
      PRD(51)=100*XWFA(1)
      PRD(52)=100*XWFA(2)
      PRD(69)=100*XWFA(3)
      SV(LOS,117)=XWFA(1)*WFA/XMW(37)
      SV(LOS,118)=XWFA(2)*WFA/XMW(38)
      SV(LOS,119)=XWFA(3)*WFA/XMW(39)
      READ(5,100) XLD
      READ(5,100) XSD
C
      DO 2 J=1,6
      K=12+J
      L=18+J
      PRD(K)=XLD(J)*100.
      PRD(L)=XSD(J)*100.
    2 CONTINUE
      PRD(70)=XLD(7)*100.
      PRD(71)=XSD(7)*100.
      TMS =SV(LOS,11)+SV(LOS,104)+SV(LOS,113)+SV(LOS,106)+SV(LOS,116)
      TMCA=SV(LOS,101)+SV(LOS,103)+SV(LOS,104)+SV(LOS,106)
      TMMG=SV(LOS,108)+SV(LOS,110)+SV(LOS,113)+SV(LOS,116)
      PRD(9)=100*(TMCA+TMMG)/TMS
      PRD(10)=TMMG/(TMMG+TMCA)
      PRD(26)=WFA/(32.066*TMS)
      PRD(68)=100*(TMCA+TMMG-SV(LOS,101)-SV(LOS,108))/(TMMG+TMCA)
      RETURN 2
      END
```

```
@I FOR FNEQ
      FUNCTION FNEQ(F,X)
      DOUBLE PRESCISION FNEQ,F,X
      IF(F.LT.0..AND.X.LT.-50) GO TO 1
      FNEQ=F
      RETURN
    1 CONTINUE
      X=X-1
      IF(X.LT.-85.) X=-85.
      FNEQ=0.
      RETURN
      END
```

```
@I FOR GABSO
      SUBROUTINE GABSO(LI,LO,XO,DNA,ABN)
      INCLUDE CMMN,LIST
      DIMENSION DNA(1)
      P(1)=P(1)
      DO 1 J=1,4
      I=J+10
      SV(LO,I)=(1-XA(J))*SV(LI,I)
    1 CONTINUE
      SV(LO,15)=SV(LI,15)-.5*XO*(SV(LI,11)-SV(LO,11))-.75*(SV(LI,13)-SV(
     *LO,13))-.25*(SV(LI,14)-SV(LO,14))
      SV(LO,18)=0.
C
      ABN=0.
      DO 2 J=1,5
      I=10+J
      DNA(J)=SV(LI,I)-SV(LO,I)
      ABN=ABN+DNA(J)
    2 CONTINUE
      DNA(8)=SV(LI,18)
      DNA(4)=.5*(DNA(3)+DNA(4))
      DNA(3)=DNA(1)*XO
      DNA(5)=0.
      DNA(1)=DNA(1)*(1-XO)
      DNA(6)=0.
      DNA(7)=0.
      RETURN
      END
```

```
@N FOR GAMMA
      SUBROUTINE GAMMA(X)
C ********************************************************************
C *    THIS ROUTINE COMPUTES LN(G(I)) AS A FUNCTION OF EXP(X(I))      *
C ********************************************************************
      COMMON/NEW/IV(50),IE(50),CL
      COMMON /TEMPER/AC,BC
      COMMON /LIMS/NF,NT
      COMMON /PPPESS/PK(2),L1,L2,L3,L4,REN
      COMMON /ACOEF/GLN(50),DGLN(50),IZ(50),FI
      DOUBLE PRECISION X
      DIMENSION X(1),A(50),B(50),U(50),NS(50)
      DATA (IZ(I),I=1,38)/1,1,1,4,4,1,4,1,1,0,0,4,1,0,0,1,0,1,0,0,0,0,4,
     11,0,1,0,0,0,0,1,0,1,0,1,0,1,0/
      DATA U(10)/.076/U(11)/.076/U(14)/.076/U(15)/.076/U(17)/.076/U(25)/
     1.076/U(27)/.076/U(28)/.076/U(32)/.076/U(34)/.076/U(36)/.076/
      DATA (A(I),I=1,37)    /   6.00000    ,    3.00000    ,    4.50000    ,
     *   4.50000    ,   3.00000    ,   4.50000    ,   4.50000    ,   2.00000    ,
     *   3.00000    ,   3.00000    ,   3.00000    ,   4.50000    ,   3.00000    ,
     *   3.00000    ,   3.00000    ,   3.00000    ,   3.00000    ,   3.00000    ,
     *   3.00000    ,   3.00000    ,   3.00000    ,   3.00000    ,   3.00000    ,
     *   3.00000    ,   3.00000    ,   3.00000    ,   3.00000    ,   3.00000    ,
     *   3.00000    ,   3.00000    ,   5.00000    ,   3.00000    ,   3.00000    ,
     *   3.00000    ,   3.00000    ,   3.00000    ,   4.00000    /
      DATA (B(I),I=1,37)    /   .400000    ,    .300000    ,    .000000    ,
     *   .000000    ,   .000000    ,   .000000    ,   .000000    ,  -.200000    ,
     *   .300000    ,   .300000    ,   .300000    ,   .100000    ,   .300000    ,
     *   .300000    ,   .300000    ,   .300000    ,   .300000    ,   .300000    ,
     *   .300000    ,   .300000    ,   .300000    ,   .300000    ,   .300000    ,
     *   .300000    ,   .300000    ,   .300000    ,   .300000    ,   .300000    ,
     *   .300000    ,   .300000    ,   .100000    ,   .300000    ,   .300000    ,
     *   .300000    ,   .300000    ,   .300000    ,   .000000    /
      DATA (NS(I),I=1,38)/13*1,4*0,5*1,2*0,7*1,0/
C
      DEFINE Y(I)=EXP(X(I))
C
      FI=0.
      DO 1 I=1,36
      FI=FI+IZ(I)*Y(I)
    1 CONTINUE
      X(37)=LOG(1./Y 39))
      FI=FI+Y(37)
```

```
      FI=FI/2.
      IF(L4.EQ.8) FI=FI+ABS(REN)
      SFI=SQRT(FI)
C
      GLN(39)=0.
      DO 2 I=1,37
      DNM=1.+BC*A(I)*SFI
      GLN(I)=AC*IZ(I)*(-SFI/DNM+B(I)*FI)+2.3026*U(I)*FI
      TEMP=Y(I)*(-AC*IZ(I)*SFI*((1+DNM)/DNM-2*LOG(DNM)/(DNM-1))/(DNM-1)*
     **2+2.3026*U(I)*FI/2+AC*IZ(I)*B(I)*FI/2+NS(I))
      GLN(39)=GLN(39)+TEMP
    2 CONTINUE
      GLN(39)=-.018016*GLN(39)
      RETURN
      END
```

```
@I FOR GPARS
      SUBROUTINE GPARS(GD,A,IC,IA,IS,NE)
      DOUBLE PRECISION GD,A,EPF
      DIMENSION GD(50,50)
      DATA ALFA/10./
C
      GD(NE,IC)=-1.
      GD(NE,IA)=-1.
      IF(ABS(A).GT.500.) GO TO 1
      EPF=1.-EXP(-ABS(A)/ALFA)
      GD(NE,IS)=EPF
      RETURN
    1 EPF=1.
      RETURN
      END
```

```
@I FOR GSTAR
      SUBROUTINE GSTAR(X,EQS)
C * **********************************
C *            *     T                *
C *          G (X)=B (X)*F(X)         *
C * **********************************
      COMMON /RESID/ GS(50)
      COMMON /GRAD/PD(50,50),AS(50,50),SQ(50)
      COMMON/FUNC/F(50),CK(50),CT(10)
      COMMON /LIMS/NF
      DOUBLE PRECISION SUM
      DOUBLE PRECISION PD,AS,SQ,GS
      DOUBLE PRECISION F,CK,CT,X
      DIMENSION X(1)
C
C
C       ASSUME THE PARTIALS HAVE BEEN EVALUATED
C
      DO 1 I=1,NF
      SUM=0.
      DO 2 K=1,NF
      SUM=SUM+PD(K,I)*F(K)
    2 CONTINUE
      GS(I)=SUM/SQ(I)
    1 CONTINUE
      RETURN
      END
```

```
@I FOR INTIT
      SUBROUTINE INTIT(L,IOPT,XWS,WNES)
      INCLUDE CMMN,LIST
      INCLUDE LCMN,LIST
      DIMENSION WTS(10)
      DATA (WTS(I),I=1,9)/64.066,44.011,80.066,106.016,56.08,40.32,61.98
     *2,36.465,18.016/
C
      DEFINE X(I)=SV(L,I)
C
      IF(IOPT.NE.4) GO TO 1
      XSCAO=X(35)-(X(31)+X(32)+X(33))
      IF(XSCAO.LT.0.) GO TO 2
      X(51)=4.E-12
      X(105)=X(31)
      X(106)=X(33)
      X(103)=X(32)
      X(109)=X(36)
      X(102)=XSCAO
    3 WES=XMW(25)*X(105)+XMW(26)*X(106)+XMW(23)*X(103)+XMW(22)*X(102)
     *+XMW(29)*X(109)
      WTM=0.
      DO 10 J=1,8
      I=30+J
      WTM=WTM+WTS(J)*X(I)
   10 CONTINUE
      X(50)=(WES+WNES)/XWS-WNES-WTM
    5 FLK=X(50)/1000.
      X(52)=5.3E-14/X(51)
      IF(X(35).NE.0.) X(62)=3.2E-6/X(52)**2
      IF(XSCAO.LT.0.) X(62)=10.*X(52)
      IF(X(31).NE.0.) X(54)=8.4E-8/X(62)
      IF(X(32).NE.0.) X(57)=1.8E-9/X(62)
      IF(X(36).NE.0.) X(69)=9.5E-12/X(52)**2
    9 IF(X(33).NE.0.) X(55)=3.4E-5/X(62)
      IF(X(5).GT.313.16) GO TO 20
      X(107)=X(106)
      X(106)=0.
   20 CONTINUE
      X(58)=2*X(34)/FLK
      X(75)=2*X(37)/FLK
C
```

```
      X(53)=X(5 . *X(54)/4.3E-5
      X(60)=X(51)*X(53)/7.9E-3
      X(59)=X(51)*X(55)/4.9E-3
      X(56)=X(51)*X(57)/6.7E-11
      X(61)=X(51)*X(56)/5.2E-7
      X(63)=X(62)*X(52)/3.6E-2
      X(64)=X(62)*X(54)/3.0E-4
      X(65)=X(62)*X(57)/4.8E-4
      X(66)=X(62)*X(56)/4.6E-2
      X(67)=X(62)*X(55)/3.6E-3
      X(68)=X(62)*X(58)/2.3E-1
      X(70)=X(69)*X(52)/1.9E-3
      X(71)=X(69)*X(54)/9.3E-4
      X(72)=X(69)*X(56)/9.6E-2
      X(73)=X(69)*X(55)/3.1E-3
      X(74)=X(69)*X(57)/3.0E-4
      X(76)=X(75)*X(52)/3.7
      X(77)=X(75)*X(57)/4.5E-2
      X(78)=X(75)*X(56)/1.8
      X(79)=X(75)*X(55)/1.7E-1
      X(80)=X(75)*X(58)/2.5
      RETURN
    2 D1=X(35)-X(31)
      X(51)=5.E-9
      IF(D1.LT.0.) GO TO 4
      X(105)=X(31)
      GO TO 6
    4 X(105)=X(31)
      GO TO 7
    6 D2=D1-X(33)
      IF(D2.LT.0.) GO TO 8
      X(106)=X(33)
      X(103)=D2
      GO TO 7
    8 X(106)=D1
      IF(X(36).EQ.0.) CALL EXIT
      X(109)=2.*X(36)
    7 GO TO 3
    1 X(51)=3.E-4
      X(52)=5.3E-14/X(51)
      FLK=X(50)/1000.
      IF(X(36).LT.X(33)) GO TO 11
      X(73)=.7*X(33)/FLK
```

```
      GO TO 12
   11 X(73)=.7*X(36)/FLK
   12 IF(X(36).NE.0) X(69)=X(36)/FLK-X(73)
      IF(X(35).NE.0) X(62)=.5*X(35)/FLK
      IF(X(33).NE.0) X(54)=.7*4.3E-8*X(31)/FLK/X(51)
      PCO2=XWS
      IF(X(32).NE.0) X(57)=6.34E-19*PCO2/X(51)**2
      IF(X(33).NE.0) GO TO 13
      IF(X(35).NE.0) GO TO 13
      GO TO 9
   13 CONTINUE
      DCAO=X(35)/FLK-X(62)
      DSO3=X(33)/FLK-X(73)
      IF(DSO3.GT.DCAO) GO TO 14
      X(106)=DSO3*FLK
      GO TO 15
   14 X(106)=DCAO*FLK
   15 IF(X(62)*X(54).GT.8.4E-8)X(105)=1.E-4*FLK
      GO TO 9
      END
```

```
@I FOR MATCH
      FUNCTION MATCH(NAME)
      DIMENSION ILAB(15)
      DATA (ILAB(I),I=1,13)/'FLUGAS','STKGAS','WTRMKP','FLTRBM','SCRUBR'
     *,'CLRHTR','FILTER','DIVDER','EQMIXR','EFHOLD','PMPFAN','CLRFYR',
     *'OVALMB'/
C
      DO 1 I=1,13
      IF(ILAB(I).EQ.NAME) GO TO 2
    1 CONTINUE
C
    2 MATCH=I
      RETURN
      END
```

```
@N FOR MATIN
      SUBROUTINE MATIN (A,N,B,M,DETERM)
      DOUBLE PRECISION A,B,DETERM,PIVOT,AMAX,SWAP,T
      DIMENSION IPIVOT(50),A(50,50),B(50, 1),INDEX(50,2),PIVOT(50)
      EQUIVALENCE (IROW,JROW), (ICOLUM,JCOLUM), (AMAX, T, SWAP)            F402
C
C     INITIALIZATION
C
   10 DETERM=1.0
   15 DO 20 J=1,N
   20 IPIVOT(J)=0
   30 DO 550 I=1,N
C
C     SEARCH FOR PIVOT ELEMENT
C
   40 AMAX=0.0
   45 DO 105 J=1,N
   50 IF (IPIVOT(J)-1) 60,105,60
   60 DO 100 K=1,N
   70 IF (IPIVOT(K)-1) 80,100,740
   80 IF (ABS(AMAX)-ABS(A(J,K))) 85,100,100
   85 IROW=J
   90 ICOLUM=K
   95 AMAX=A(J,K)
  100 CONTINUE
  105 CONTINUE
  106 IF (ABS(AMAX).LE.0.0) GO TO 739
  110 IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1
C
C     INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL
C
  130 IF (IROW-ICOLUM) 140,260,140
  140 DETERM = -DETERM
  150 DO 200 L=1,N
  160 SWAP=A(IROW,L)
  170 A(IROW,L)=A(ICOLUM,L)
  200 A(ICOLUM,L)=SWAP
  205 IF(M)260,260,210
  210 DO 250 L=1,M
  220 SWAP=B(IROW,L)
  230 B(IROW,L)=B(ICOLUM,L)
  250 B ICO.JM, . =SWAP
```

```
  260 INDEX(I,1)=IROW
  270 INDEX(I,2)=ICOLUM
  310 PIVOT(I)=A(ICOLUM,ICOLUM)
  320 DETERM=DETERM*PIVOT(I)
C
C     DIVIDE PIVOT ROW BY PIVOT ELEMENT
C
  330 A(ICOLUM,ICOLUM)=1.0
  340 DO 350 L=1,N
  350 A(ICOLUM,L)=A(ICOLUM,L)/PIVOT(I)
  355 IF(M)380,380,360
  360 DO 370 L=1,M
  370 B(ICOLUM,L)=B(ICOLUM,L)/PIVOT(I)
C
C     REDUCE NON-PIVOT ROWS
C
  380 DO 550 L1=1,N
  390 IF(L1-ICOLUM) 400,550,400
  400 T=A(L1,ICOLUM)
  420 A(L1,ICOLUM)=0.0
  430 DO 450 L=1,N
  450 A(L1,L)=A(L1,L)-A(ICOLUM,L)*T
  455 IF(M) 550,550,460
  460 DO 500 L=1,M
  500 B(L1,L)=B(L1,L)-B(ICOLUM,L)*T
  550 CONTINUE
C
C     INTERCHANGE COLUMNS
C
  600 DO 710 I=1,N
  610 L=N+1-I
  620 IF (INDEX(L,1)-INDEX(L,2)) 630,710,630
  630 JROW=INDEX(L,1)
  640 JCOLUM=INDEX(L,2)
  650 DO 705 K=1,N
  660 SWAP=A(K,JROW)
  670 A(K,JROW)=A(K,JCOLUM)
  700 A(K,JCOLUM)=SWAP
  705 CONTINUE
  710 CONTINUE
      GO TO 740
  739 DETERM=0.0
  740 RETURN
```

```
@N FOR NOLIN
      SUBROUTINE NOLIN(X,N,EPS,EQS,PARTL)
      COMMON/LIMS/NF
      COMMON/FUNC/F(50),CK(50),CT(10)
      COMMON /GRAD/PD(50,50),AS(50,50),SQ(50)/RESID/GS(50)
      COMMØN /NEW/IV(50),IE(50),CL,TH(11)
      DIMENSION X(1),DELT(50)
      DOUBLE PRECISION F,CK,CT,PD,AS,SQ,GS
      DOUBLE PRECISION X,DELT,PHI,PH1,PH2
      NF=N
      EPS=1.E-5
      DL=1.E-8
      PH1=PHI(X,EQS)
C
   10 CONTINUE
      PH2=PH1
      CALL ASTAR(X,PARTL)
      CALL GSTAR(X,EQS)
      CALL DELTA(DELT,DL)
      IC=1
      DO9I=1,N
      TE=ABS(DELT(I))/(.001+ABS(X(I)))
      IF(TE.GT.EPS) IC=0
      X(I)=X(I)+DELT(I)
    9 CONTINUE
      IF(IC.EQ.1) RETURN
      PH1=PHI(X,EQS)
      IF(PH1.LT.1.D-2) DL=0.
      GO TO 10
      END
```

```
@I FOR ORDER
      SUBROUTINE ORDER(NS,NI,NO)
      DIMENSION NS(10),IDUM(10)
      NO=0
      NI=0
      DO 1 I=1,10
      IF(NS(I))2,1,3
    2 NO=NO+1
      NX=5+NO
      IDUM(NX)=-NS(I)
      GO TO 1
    3 NI=NI+1
      IDUM(NI)=NS(I)
    1 CONTINUE
      DO 4 I=1,10
      NS(I)=IDUM(I)
    4 IDUM(I)=0
      RETURN
      END
```

```
@I FOR OVALMB
      SUBROUTINE OVALMB(P,$)
      INCLUDE CMMN,LIST
      DIMENSION PP(2)
C ****************************************************
C *       THIS ROUTINE PERFORMS AN OVERALL         *
C *       MATERIAL BALANCE ABOUT THE SYSTEM        *
C ****************************************************
      LOCSV=ISEQ(NL)
      LIS1=ISTM(LOCSV,1)
      LIS2=ISTM(LOCSV,2)
      LOS1=ISTM(LOCSV,6)
      LOS2=ISTM(LOCSV,7)
      KEY1=SV(LOS2,2)
      IF(KEY1.NE.0) GO TO 1
      INIT=0
      ISET=0
      AH2O=1.
      PRD(4)=P(1)*100.
      PRD(5)=P(3)*100.
      PRD(7)=P(4)*100.
      PRD(6)=P(5)*100.
      PRD(63)=P(6)
      PRD(11)=P(7)*100.
      PRD(12)=P(8)*100.
      XA(2)=P(2)
      RETURN 2
    1 CONTINUE
      XA(1)=P(1)
      XA(3)=P(3)
      XA(4)=P(4)
      SV(LOS1,6)=SV(LIS1,6)-P(6)/14.696
      PGO =SV(LOS1,6)
      XO=P(5)
      XLSU(1)=P(7)
      XLSU(2)=P(8)
      XF8=SV(LOS2,100)
      IF(ISET.EQ.1) GO TO 5
      CALL GABSO(LIS1,LOS1,XO,    ONAT,ABN)
      CALL WAVAP(LIS1,ABN,PGO,TAS,WV)
      IF(P(2).EQ.0.) XA(2)=3.E-4
    5 CONTINUE
```

```
      VPLOG=5.22684-1750.286/(TAS-38.2)
      YGS=(AH2O/SV(LOS1,6))*10.**VPLOG
      CALL GABSO(LIS1,LOS1,XO,    DNAT,ABN)
      WV=(SV(LIS1,10)-SV(LIS1,19)-ABN)*YGS/(1.-YGS)-SV(LIS1,19)
      SV(LOS2,101)=(1-P(7))*SV(LIS1,101)
      SV(LOS2,108)=(1-P(8))*SV(LIS1,108)
      SV(LOS2,118)=SV(LIS1,118)
      CALL DNTLAO(LIS1,P(7),P(8))
      IF(ISET.EQ.1) GO TO 3
      ISET=1
      SW=SV(LIS1,101)+SV(LIS1,108)
      WFB=5.50622*SW*(1-XFB)/XFB
      WM=(WV+WFB+SW)*18.016
      SV(LIS2,50)=WM
      CALL TOEQSP(LIS2)
    3 CONTINUE
      DO 4 J=1,8
      I=J+30
      SV(LOS2,I)=DNAT(J)+SV(LIS1,I)+SV(LIS2,I)
    4 CONTINUE
      SV(LOS1,5)=TAS
      SV(LOS2,5)=TAS
      SV(LIS2,5)=TAS
      WNES=56.08*SV(LOS2,101)+40.32*SV(LOS2,108)+75.80*SV(LOS2,118)
      IF(INIT.NE.0) GO TO 8
      IOPT=4
      CALL INTIT(LOS2,IOPT,XFB,WNES)
      INIT=1
    8 CONTINUE
C
      CALL EQUILB(LOS2,4,TAS,PP,WNES,0)
C
      WRITE(6,200)
  200 FORMAT(  45X,'FILTER BOTTOMS')
      SV(LOS2,50)=1000.*SV(LOS2,50)
      SV(LIS2,39)=SV(LOS2,39)+WV
      SV(LIS2,50)=18.016*SV(LIS2,39)
      IF(ABS(SV(LIS2,50)-WM)/WM.LT..05) GO TO 6
      WM=SV(LIS2,50)
      GO TO 3
    6 CALL TOEGSP(LIS2)
      PP(1)=0.
      PP(2)=0.
```

```
      CALL EQUILB(LIS2,0,TAS,PP,WNES,1)
      WRITE(6,300)
  300 FORMAT(45X,'MAKEUP WATER')
      SV(LIS2,50)=1000.*SV(LIS2,50)
      SV(LOS1,19)=SV(LIS1,19)+WV
      SV(LOS1,17)=SV(LIS1,17)
      SV(LOS1,16)=SV(LIS1,16)
      SV(LIS1,35)=SV(LIS1,35)+SV(LOS2,101)
      SV(LIS1,36)=SV(LIS1,36)+SV(LOS2,108)
      SV(LOS2,35)=SV(LOS2,35)+SV(LOS2,101)
      SV(LOS2,36)=SV(LOS2,36)+SV(LOS2,108)
      SUM=0.
      DO 7 J=1,9
      I=10+J
      SUM=SUM+SV(LOS1,I)
    7 CONTINUE
      SV(LOS1,10)=SUM
      RETURN 2
      END
```

```
@I FOR PARTL
C
C
C ***************************************************************
C *    THE PARTIAL DERIVATIVES @F/@X ARE PLACED HERE            *
C *    PD(I,J)= -@F(I)/@X(J)                                    *
C ***************************************************************
      SUBROUTINE PARTL(W)
      COMMON/FUNC/F(50),CK(50),CT(10)
      COMMON /GRAD/PD(50,50),AS(50,50),SQ(50)/RESID/GS(50)
      COMMON /ACOEF/GLN(50),DGLN(50),IZ(50)
      COMMON /LIMS/NF,NT
      COMMON /PPRESS/PK(2),L1,L2,L3,L4,REN
      COMMON /NEW/ IV(50),IE(50),CL,TH(11)
      COMMON/TEMPER/AC,BC,CPK(2),NHY(5),WS(10)
      DIMENSION W(1),X(50),GD(50,50)
      COMMON /TPAR/ATP(7)
      DOUBLE PRECISION BR,TH
      DOUBLE PRECISION ATP
      DOUBLE PRECISION F,CK,CT,W,X,GD
      DOUBLE PRECISION PD,AS,SQ,GS
C
      DEFINE Y(I)=EXP(X(I))
C
C ***************************************************************
C          ZERO UNUSED MOLARITIES AND CONVERT                   *
C ***************************************************************
      DO 5 I=1,40
      DO 6 J=1,40
      GD(I,J)=0.
    6 CONTINUE
    5 CONTINUE
      DO 1 I=1,36
      X(I)=-90.
    1 CONTINUE
      X(38)=-90.
      X(39)=0
C
      DO 2 I=1,NF
      NV=IV(I)
      X(NV)=W(I)
      X 37 =.0G C./Y(39))
```

```
    2 CONTINUE
C
      GD( 1, 1)=-1.0
      IF(L4.EQ.8) GD(1,1)=0.
      GD( 1, 2)=-1.0
      GD( 2, 1)=-1.0
      GD( 2, 3)=-1.0
      GD( 2,10)=+1.0
      IF(L1.EQ.1) GD(2,10)=0.
      GD( 3, 1)=-1.0
      GD( 3, 4)=-1.0
      GD( 3, 3)=+1.0
      GD( 4, 1)=-1.0
      GD( 4, 5)=-1.0
      GD( 4, 9)=+1.0
      GD( 5, 1)=-1.0
      GD( 5, 6)=-1.0
      GD( 5,11)=+1.0
      IF(L2.EQ.2) GD(5,11)=0.
      GD( 6, 1)=-1.0
      GD( 6, 7)=-1.0
      GD( 6, 6)=+1.0
      GD( 7,12)=-1.0
      GD( 7, 2)=-1.0
      GD( 7,13)=+1.0
      GD( 8,12)=-1.0
      GD( 8, 4)=-1.0
      GD( 8,14)=+1.0
      GD( 9,12)=-1.0
      GD( 9, 7)=-1.0
      GD( 9,15)=+1.0
      GD(10,12)=-1.0
      GD(10, 6)=-1.0
      GD(10,16)=+1.0
      GD(11,12)=-1.0
      GD(11, 5)=-1.0
      GD(11,17)=+1.0
      GD(12,12)=-1.0
      GD(12, 8)=-1.0
      GD(12,18)=+1.0
      CALL GPARS(GD,ATP(1),12, 7,19,13)
      CALL GPARS(GD,ATP(2),12, 5,20,14)
      CALL GPARS(GD,ATP(3),12, 4,21,15)
```

```
      CALL GPARS(GD,ATP(4),12, 2,22,16)
      GD(16, 2)=2*GD(16, 2)
      GD(17,23)=-1.0
      GD(17, 2)=-1.0
      GD(17,24)=+1.0
      GD(18,23)=-1.0
      GD(18, 4)=-1.0
      GD(18,25)=+1.0
      GD(19,23)=-1.0
      GD(19, 6)=-1.0
      GD(19,26)=+1.0
      GD(20,23)=-1.0
      GD(20, 5)=-1.0
      GD(20,27)=+1.0
      GD(21,23)=-1.0
      GD(21, 7)=-1.0
      GD(21,28)=+1.0
      CALL GPARS(GD,ATP(5),23, 2,29,22)
      CALL GPARS(GD,ATP(6),23, 7,30,23)
      CALL GPARS(GD,ATP(7),23, 4,38,38)
      GD(22, 2)=2*GD(22, 2)
      GD(24, 2)=-1.0
      GD(24,32)=+1.0
      GD(25,31)=-1.0
      GD(25, 7)=-1.0
      GD(25,33)=+1.0
      GD(26,31)=-1.0
      GD(26, 6)=-1.0
      GD(26,34)=+1.0
      GD(27,31)=-1.0
      GD(27, 5)=-1.0
      GD(27,35)=+1.0
      GD(28,31)=-1.0
      GD(28, 8)=-1.0
      GD(28,36)=+1.0
      BR=TH(1)
      GD(29, 3)=Y( 3)/BR
      GD(29, 4)=Y( 4)/BR
      GD(29,10)=Y(10)/BR
      IF(L1.EQ.1) GD(29,10)=-1.
      GD(29,14)=Y(14)/BR
      GD(29,21)=Y(21)/BR
      GD(29,25)=Y(25 /BR
```

```
GD(29,39)=1.
BR=TH(2)
GD(30, 6)=Y( 6)/BR
GD(30, 7)=Y( 7)/BR
GD(30,11)=Y(11)/BR
IF(L2.EQ.2) GD(30,11)=-1.
GD(30,15)=Y(15)/BR
GD(30,16)=Y(16)/BR
GD(30,19)=Y(19)/BR
GD(30,26)=Y(26)/BR
GD(30,28)=Y(28)/BR
GD(30,30)=Y(30)/BR
GD(30,33)=Y(33)/BR
GD(30,34)=Y(34)/BR
GD(30,39)=1.
BR=TH(3)
GD(31, 5)=Y( 5)/BR
GD(31, 9)=Y( 9)/BR
GD(31,17)=Y(17)/BR
GD(31,20)=Y(20)/BR
GD(31,27)=Y(27)/BR
GD(31,35)=Y(35)/BR
GD(31,39)=1.
BR=TH(4)
GD(32,36)=Y(36)/BR
GD(32, 8)=Y( 8)/BR
GD(32,18)=Y(18)/BR
GD(32,39)=1.
BR=TH(5)
GD(33,12)=Y(12)/BR
GD(33,13)=Y(13)/BR
GD(33,14)=Y(14)/BR
GD(33,15)=Y(15)/BR
GD(33,16)=Y(16)/BR
GD(33,17)=Y(17)/BR
GD(33,18)=Y(18)/BR
GD(33,19)=Y(19)/BR
GD(33,20)=Y(20)/BR
GD(33,21)=Y(21)/BR
GD(33,22)=Y(22)/BR
GD(33,39)=1.
BR=TH(6)
GD(34,23)=Y(23)/BR
```

```
      GD(34,24)=Y(24)/BR
      GD(34,25)=Y(25)/BR
      GD(34,26)=Y(26)/BR
      GD(34,27)=Y(27)/BR
      GD(34,28)=Y(28)/BR
      GD(34,29)=Y(29)/BR
      GD(34,30)=Y(30)/BR
      GD(34,38)=Y(38)/BR
      GD(34,39)=1.
      BR=TH(7)
      GD(35,31)=Y(31)/BR
      GD(35,32)=Y(32)/BR
      GD(35,33)=Y(33)/BR
      GD(35,34)=Y(34)/BR
      GD(35,35)=Y(35)/BR
      GD(35,36)=Y(36)/BR
      GD(35,39)=1.
      IF(L4.EQ.8) GO TO 9
      BR=TH(8)
      GD(36, 1)=-  Y( 1)/BR
      GD(36,12)=-2*Y(12)/BR
      GD(36,13)=-  Y(13)/BR
      GD(36,16)=-  Y(16)/BR
      GD(36,18)=-  Y(18)/BR
      GD(36,23)=-2*Y(23)/BR
      GD(36,24)=-  Y(24)/BR
      GD(36,26)=-  Y(26)/BR
      GD(36,31)=-  Y(31)/BR
      BR=TH(9)
      GD(36, 2)=  Y( 2)/BR
      GD(36, 3)=  Y( 3)/BR
      GD(36, 4)=2*Y( 4)/BR
      GD(36, 5)=2*Y( 5)/BR
      GD(36, 6)=  Y( 6)/BR
      GD(36, 7)=2*Y( 7)/BR
      GD(36, 8)=  Y( 8)/BR
      GD(36, 9)=  Y( 9)/BR
      GD(36,33)=  Y(33)/BR
      GD(36,35)=  Y(35)/BR
      GD(36,39)=-Y(37)/BR
      GO TO 10
    9 GD(36, 1)=-1.
   10 CONTINUE
```

```
      BR=TH(10)
      GD(39, 1)=.5*Y( 1)/BR
      GD(39, 2)=.5*Y( 2)/BR
      GD(39, 6)=.5*Y( 6)/BR
      GD(39, 9)=.5*Y( 9)/BR
      GD(39,10)=Y(10)/BR
      IF(L1.EQ.1) GD(39,10)=0.
      GD(39,11)=Y(11)/BR
      IF(L2.EQ.2) GD(39,11)=0.
      GD(39,13)=.5*Y(13)/BR
      GD(39,16)=.5*Y(16)/BR
      GD(39,24)=.5*Y(24)/BR
      GD(39,26)=.5*Y(26)/BR
      GD(39,32)=.5*Y(32)/BR
      GD(39,34)=.5*Y(34)/BR
      GD(39,20)=NHY(1)*Y(20)/BR
      GD(39,21)=.5*Y(21)/BR
      GD(39,22)=Y(22)/BR
      GD(39,30)=NHY(2)*Y(30)/BR
      GD(39,29)=Y(29)/BR
      GD(39,38)=NHY(3)*Y(38)/BR
      GD(39,39)=1.+.5*Y(37)/BR
      IF(L3.NE.4) GO TO 8
      BR=TH(11)
      GD(39,19)=          -WS(2)*Y(19)*Y(39)/BR
      GD(39,20)=GD(39,20)-WS(5)*Y(20)*Y(39)/BR
      GD(39,21)=GD(39,21)-WS(4)*Y(21)*Y(39)/BR
      GD(39,22)=GD(39,22)-WS(1)*Y(22)*Y(39)/BR
      GD(39,29)=GD(39,29)-WS(6)*Y(29)*Y(39)/BR
      GD(39,30)=GD(39,30)-WS(8)*Y(30)*Y(39)/BR
      GD(39,38)=GD(39,38)-WS(9)*Y(38)*Y(39)/BR
      GD(39,39)=GD(39,39)+CT(10)/BR-1
    8 CONTINUE
C
      DO 3 I=1,NF
      NE=IE(I)
      DO 4 J=1,NF
      NV=IV(J)
      PD(I,J)=GD(NE,NV)
    4 CONTINUE
    3 CONTINUE
      RETURN
      END
```

```
@IN PDP PDPELT,PDPELT
CMMN* FCOPY
       COMMON                        @ COMMON DATA FOR ALL SUBROUTINES
      *SV(50,120),                   @ STREAM DATA
      *ISTM(25,10),                  @ STREAM CONNECTION ARRAY
      *IDEQP(25),                    @ EQUIPMENT NAME
      *ISEQ(30),                     @ ORDER OF PROCESS CAL.
      *PA(25,8),                     @ EQUIP. PARAMETERS
      *L1,L2,L3,                     @ FLAGS FOR THE RECYCLE LOOPS
      *NL,                           @ CURRENT INDEX
      *LSRL(3),LERL(3),              @ START AND END OF RECYCLE LOOPS
      *WV,XLSU(2),XSD(7),XLD(7),DNAT(9),XA(4),PRD(100),LABEL(13), AH20
       DIMENSION P(8)
       END
LCMN* FCOPY
       COMMON /DARRY/XMW(100) ,DHF(100) ,ENTO(100),ACP(100) ,BCP(100) ,
      1              CCP(100) ,DCP(100) ,TMIN(100),TMAX(100),ZE(100)  ,
      2              ALIT(100),BLIT(100),UCAP(100)
       END
PHYSD* FCOPY
      DDATA  XMW( 1)/   64.066/, DHF( 1)/  -70930./,ENTO( 1)/  59.270/,  SO2
      1      XMW( 2)/   44.011/, DHF( 2)/  -94010./,ENTO( 2)/  51.030/,  CO2
      2      XMW( 3)/   30.008/, DHF( 3)/   21590./,ENTO( 3)/  50.310/,  NO
      3      XMW( 4)/   46.008/, DHF( 4)/    7960./,ENTO( 4)/  57.290/,  NO2
      4      XMW( 5)/   32.000/, DHF( 5)/       0./,ENTO( 5)/  48.970/,  O2
      5      XMW( 6)/   28.011/, DHF( 6)/  -26416./,ENTO( 6)/  47.210/,  CO
      6      XMW( 7)/   28.016/, DHF( 7)/       0./,ENTO( 7)/  45.770/,  N2
      7      XMW( 8)/   36.465/, DHF( 8)/  -22063./,ENTO( 8)/  44.645/,  HCL
      8      XMW( 9)/   18.016/, DHF( 9)/  -57770./,ENTO( 9)/  45.070/,  H2O
      9      XMW(21)/   56.080/, DHF(21)/ -151730./,ENTO(21)/   9.480/   CAO
      DDATA  XMW(22)/   74.100/, DHF(22)/ -235600./,ENTO(22)/  19.920/,  CA(OH)2
      1      XMW(23)/  100.090/, DHF(23)/ -288110./,ENTO(23)/  22.190/,  CACO3
      2      XMW(24)/  120.140/, DHF(24)/ -275880./,ENTO(24)/  24.200/,  CASO3
      3      XMW(25)/  129.140/, DHF(25)/ -316200./,ENTO(25)/  29.?  /,  CASO3* W
      4      XMW(26)/  136.140/, DHF(26)/ -340190./,ENTO(26)/  25.490/,  CASO4
      5      XMW(27)/  172.170/, DHF(27)/ -482400./,ENTO(27)/  45.3  /,  CASO4*2W
      6      XMW(28)/   40.320/, DHF(28)/ -143630./,ENTO(28)/   6.40 /,  MGO
      7      XMW(29)/   58.340/, DHF(29)/ -221000./,ENTO(29)/  15.10 /,  MG(OH)2
      8      XMW(30)/   84.330/, DHF(30)/ -263980./,ENTO(30)/  15.70 /,  MGCO3
      9      XMW(31)/  138.380/, DHF(31)/ -473100./,ENTO(31)/  45.3  /   MGCO3*3W
      DDATA  XMW(32)/  174.410/, DHF(32)/       0./,ENTO(32)/   0.0  /,  MGCO3*5W
             XMW 33 /   )4.380/, )H  33 / -24 000./,ENTO(33 /  22.500/,  MGSO3
```

```
2      XMW(34)/   58.430/, DHF(34)/ -463300./,ENTO(34)/  52.1  /,  MGSO3*3W
3      XMW(35)/  212.480/, DHF(35)/ -676300./,ENTO(35)/  81.7  /,  MGSO3*6W
4      XMW(36)/  120.380/, DHF(36)/ -305500./,ENTO(36)/  21.900/,  MGSO4
5      XMW(37)/   61.980/, DHF(37)/ -102970./,ENTO(37)/  16.990/,  NA2O
6      XMW(38)/   75.800/, DHF(38)/ -242547./,ENTO(38)/  11.240/,  FLY ASH
7      XMW(39)/   58.448/, DHF(39)/ - 98260./,ENTO(39)/  17.240/   NACL
0DATA TMIN( 1)/      298./,TMAX( 1)/     1500./,ACP( 1)/    6.945/,  SO2
1      TMIN( 2)/      298./,TMAX( 2)/     1500./,ACP( 2)/    6.739/,  CO2
2      TMIN( 3)/      298./,TMAX( 3)/     1500./,ACP( 3)/    6.440/,  NO
3      TMIN( 4)/      298./,TMAX( 4)/     1300./,ACP( 4)/    6.569/,  NO2
4      TMIN( 5)/      298./,TMAX( 5)/     1500./,ACP( 5)/    6.117/,  O2
5      TMIN( 6)/      298./,TMAX( 6)/     1500./,ACP( 6)/    6.750/,  CO
6      TMIN( 7)/      298./,TMAX( 7)/     1500./,ACP( 7)/    6.457/,  N2
7      TMIN( 8)/      298./,TMAX( 8)/     1500./,ACP( 8)/    6.734/,  HCL
8      TMIN( 9)/      298./,TMAX( 9)/     1500./,ACP( 9)/    7.136/,  H2O
9      TMIN(21)/      273./,TMAX(21)/     1173./,ACP(21)/   10.000/   CAO
0DATA TMIN(22)/      298./,TMAX(22)/      973./,ACP(22)/   25.130/,  CA(OH)2
1      TMIN(23)/      298./,TMAX(23)/      973./,ACP(23)/   24.970/,  CACO3
2      TMIN(24)/      298./,TMAX(24)/      873./,ACP(24)/   13.770/,  CASO3
3      TMIN(25)/      298./,TMAX(25)/      373./,ACP(25)/   26.2  /,  CASO3* W
4      TMIN(26)/      298./,TMAX(26)/      973./,ACP(26)/   17.220/,  CASO4
5      TMIN(27)/      298./,TMAX(27)/      373./,ACP(27)/   44.5  /,  CASO4*2W
6      TMIN(28)/      298./,TMAX(28)/     2000./,ACP(28)/   10.170/,  MGO
7      TMIN(29)/      298./,TMAX(29)/     1000./,ACP(29)/   17.526/,  MG(OH)2
8      TMIN(30)/      298./,TMAX(30)/      700./,ACP(30)/   18.610/,  MGCO3
9      TMIN(31)/      298./,TMAX(31)/      373./,ACP(31)/   51.5  /   MGCO3*3W
0DATA TMIN(32)/        0./,TMAX(32)/        0./,ACP(32)/    0.0  /,  MGCO3*5W
1      TMIN(33)/      298./,TMAX(33)/      973./,ACP(33)/   17.090/,  MGSO3
2      TMIN(34)/      298./,TMAX(34)/      373./,ACP(34)/   50.1  /,  MGSO3*3W
3      TMIN(35)/      298./,TMAX(35)/      373./,ACP(35)/   83.0  /,  MGSO3*6W
4      TMIN(36)/      298./,TMAX(36)/     1400./,ACP(36)/   16.530/,  MGSO4
5      TMIN(37)/      298./,TMAX(37)/      973./,ACP(37)/   15.590/,  NA2O
6      TMIN(38)/      298./,TMAX(38)/      800./,ACP(38)/   14.920/,  FLY ASH
7      TMIN¢39)/      298./,TMAX¢39)/     1000./,ACP¢39)/    9.8796/  NACL
0DATA  BCP( 1)/10.010E-3/,CCP( 1)/ -3.794E-6/,DCP( 1)/    0.0  /,  SO2
1       BCP( 2)/10.140E-3/,CCP( 2)/ -3.415E-6/,DCP( 2)/    0.0  /,  CO2
2       BCP( 3)/ 2.069E-3/,CCP( 3)/ -.4206E-6/,DCP( 3)/    0.0  /,  NO
3       BCP( 4)/ 1.570E-3/,CCP( 4)/   .708E-6/,DCP( 4)/    0.0  /,  NO2
4       BCP( 5)/ 3.167E-3/,CCP( 5)/ -1.005E-6/,DCP( 5)/    0.0  /,  O2
5       BCP( 6)/ 1.811E-3/,CCP( 6)/ -.2675E-6/,DCP( 6)/    0.0  /,  CO
6       BCP( 7)/ 1.389E-3/,CCP( 7)/ -0.069E-6/,DCP( 7)/    0.0  /,  N2
7       BCP( 8)/ 0.431E-3/,CCP( 8)/  .3613E-6/,DCP( 8)/    0.0  /,  HCL
8       BCP( 9)/ 2.640E-3/,CCP( 9)/  .0459E-6/,DCP( 9)/    0.0  /,  H2O
```

```
9      BCP(21)/ 4.840E-3/,CCP(21)/        0./,DCP(21)/  1.08E+5/   CAO
ODATA  BCP(22)/ 2.880E-3/,CCP(22)/        0./,DCP(22)/  4.51E+5/,  CA(OH)2
1      BCP(23)/ 5.240E-3/,CCP(23)/        0./,DCP(23)/  6.20E+5/,  CACO3
2      BCP(24)/15.180E-3/,CCP(24)/        0./,DCP(24)/  1.66E+5/,  CASO3
3      BCP(25)/        0./,CCP(25)/       0./,DCP(25)/    0.0  /,  CASO3* W
4      BCP(26)/23.370E-3/,CCP(26)/        0./,DCP(26)/  0.33E+5/,  CASO4
5      BCP(27)/        0./,CCP(27)/       0./,DCP(27)/    0.0  /,  CASO4*2W
6      BCP(28)/ 1.740E-3/,CCP(28)/        0./,DCP(28)/  1.48E+5/,  MGO
7      BCP(29)/14.522E-3/,CCP(29)/        0./,DCP(29)/ 3.076E<5/,  MG(OH)2
8      BCP(30)/13.790E-3/,CCP(30)/        0./,DCP(30)/  4.16E+5/,  MGCO3
9      BCP(31)/        0./,CCP(31)/       0./,DCP(31)/    0.0  /   MGCO3*3W
ODATA  BCP(32)/        0./,CCP(32)/       0./,DCP(32)/    0.0  /,  MGCO3*5W
1      BCP(33)/16.840E-3/,CCP(33)/        0./,DCP(33)/  1.48E+5/,  MGSO3
2      BCP(34)/        0./,CCP(34)/       0./,DCP(34)/    0.0  /,  MGSO3*3W
3      BCP(35)/        0./,CCP(35)/       0./,DCP(35)/    0.0  /,  MGSO3*6W
4      BCP(36)/21.800E-3/,CCP(36)/        0./,DCP(36)/  0.02E+5/,  MGSO4
5      BCP(37)/ 5.400E-3/,CCP(37)/        0./,DCP(37)/  0.00   /,  NA2O
6      BCP(38)/ 7.900E-3/,CCP(38)/        0./,DCP(38)/  3.61E+5/,  FLY ASH
7      BCP(39)/5.4183E-3/,CCP(39)/        0./,DCP(39)/-.54487E<5/  NACL
ODATA  ALIT(50)/0.0      /,BLIT(50)/0.0      /,ZE(50)/ 0.     /,  H2O(LIQ)
1      ALIT¢51I/6.0      /,BLIT¢51I/0.4      /,ZE¢51I/<1.     /,  H<
2      ALIT(52)/3.0      /,BLIT(52)/0.3      /,ZE(52)/-1.     /,  OH-
3      ALIT(53)/4.5      /,BLIT(53)/0.0      /,ZE(53)/-1.     /,  HSO3-
4      ALIT(54)/4.5      /,BLIT(54)/0.0      /,ZE(54)/-2.     /,  SO3--
5      ALIT(55)/3.0      /,BLIT(55)/0.0      /,ZE(55)/-2.     /,  SO4--
6      ALIT(56)/4.5      /,BLIT(56)/0.0      /,ZE(56)/-1.     /,  HCO3-
7      ALIT(57)/4.5      /,BLIT(57)/0.0      /,ZE(57)/-2.     /,  CO3--
8      ALIT(58)/2.0      /,BLIT(58)/-.2      /,ZE(58)/-1.     /,  NO3-
9      ALIT(59)/3.0      /,BLIT(59)/0.3      /,ZE(59)/-1.     /   HSO4-
ODATA  ALIT(60)/0.0      /,BLIT(60)/0.0      /,ZE(60)/ 0.     /,  H2SO3(L)
1      ALIT(61)/0.0      /,BLIT(61)/0.0      /,ZE(61)/ 0.     /,  H2CO3(L)
2      ALIT(62)/4.5      /,BLIT(62)/0.1      /,ZE(62)/+2.     /,  CA++
3      ALIT(63)/3.0      /,BLIT(63)/0.3      /,ZE(63)/+1.     /,  CAOH+
4      ALIT(64)/0.0      /,BLIT(64)/0.0      /,ZE(64)/ 0.     /,  CASO3(L)
5      ALIT(65)/0.0      /,BLIT(65)/0.0      /,ZE(65)/ 0.     /   CACO3(L)
ODATA  ALIT(66)/3.0      /,BLIT(66)/0.3      /,ZE(66)/+1.     /,  CAHCO3+
1      ALIT(67)/0.0      /,BLIT(67)/0.0      /,ZE(67)/ 0.     /,  CASO4(L)
2      ALIT(68)/3.0      /,BLIT(68)/0.3      /,ZE(68)/+1.     /,  CANO3+
3      ALIT(69)/3.0      /,BLIT(69)/0.3      /,ZE(69)/+2.     /,  MG++
4      ALIT(70)/3.0      /,BLIT(70)/0.3      /,ZE(70)/+1.     /,  MGOH+
5      ALIT(71)/0.0      /,BLIT(71)/0.0      /,ZE(71)/ 0.     /,  MGSO3(L)
6      ALIT(72)/3.0      /,BLIT(72)/0.3      /,ZE(72)/+1.     /,  MGHCO3+
7      ALIT(73)/0.0      /,BLIT(73)/0.0      /,ZE(73 / 0.     /,  MGSO4(.
```

```
8      ALIT(74)/0.0        /,BLIT(74)/0.0        /,ZE(74)/ 0.      /,  MGCO3(L)
9      ALIT(75)/5.0        /,BLIT(75)/0.1        /,ZE(75)/+1.      /   NA+
0DATA  ALIT(76)/0.0        /,BLIT(76)/0.0        /,ZE(76)/ 0.      /,  NAOH(L)
1      ALIT(77)/3.0        /,BLIT(77)/0.3        /,ZE(77)/-1.      /,  NACO3-
2      ALIT(78)/0.0        /,BLIT(78)/0.0        /,ZE(78)/ 0.      /,  NAHCO3(L
3      ALIT(79)/3.0        /,BLIT(79)/0.3        /,ZE(79)/+1.      /,  NASO4-
4      ALIT(80)/0.0        /,BLIT(80)/0.0        /,ZE(80)/ 0.      /,  NANO3(L)
5      ALIT(81)/4.0        /,BLIT(81)/0.0        /,ZE(81)/-1.      /   CL-
0DATA  UCAP(50)/0.0        /,DHF(50)/-68317.   /,XMW(50)/ 18.016   /,  H2O(LIQ)
1      UCAP(51)/0.0        /,DHF(51)/    0.0   /,XMW(51)/  1.008   /,  H+
2      UCAP(52)/0.0        /,DHF(52)/-54799.   /,XMW(52)/ 17.008   /,  OH-
3      UCAP(53)/0.0        /,DHF(53)/-149376.  /,XMW(53)/ 81.07    /,  HSO3-
4      UCAP(54)/0.0        /,DHF(54)/-152276.  /,XMW(54)/ 80.06    /,  SO3--
5      UCAP(55)/0.0        /,DHF(55)/-216900.  /,XMW(55)/ 96.06    /,  SO4--
6      UCAP(56)/0.0        /,DHF(56)/-164729.  /,XMW(56)/ 61.02    /,  HCO3-
7      UCAP(57)/0.0        /,DHF(57)/-162126.  /,XMW(57)/ 60.01    /,  CO3--
8      UCAP(58)/0.0        /,DHF(58)/- 49372.  /,XMW(58)/ 62.01    /,  NO3-
9      UCAP(59)/0.0        /,DHF(59)/-211660.  /,XMW(59)/ 97.07    /   HSO4-
0DATA  UCAP(60)/0.076      /,DHF(60)/-145516.  /,XMW(60)/ 82.08    /,  H2SO3(L)
1      UCAP(61)/0.076      /,DHF(61)/-166971.  /,XMW(61)/ 62.03    /,  H2CO3(L)
2      UCAP(62)/0.0        /,DHF(62)/-129770.  /,XMW(62)/ 40.08    /,  CA++
3      UCAP(63)/0.0        /,DHF(63)/-183320.  /,XMW(63)/ 57.09    /,  CAOH+
4      UCAP(64)/0.076      /,DHF(64)/-278736.  /,XMW(64)/120.14    /,  CASO3(L)
5      UCAP(65)/0.076      /,DHF(65)/-289720.  /,XMW(65)/100.08    /,  CACO3(L)
6      UCAP(66)/0.0        /,DHF(66)/-293118.  /,XMW(66)/101.10    /,  CAHCO3+
7      UCAP(67)/0.076      /,DHF(67)/-344033.  /,XMW(67)/136.14    /,  CASO4(L)
8      UCAP(68)/0.0        /,DHF(68)/-173445.  /,XMW(68)/102.09    /,  CANO3+
9      UCAP(69)/0.0        /,DHF(69)/-110410.  /,XMW(69)/ 24.32    /   MG++
0DATA  UCAP(70)/0.0        /,DHF(70)/-162839.  /,XMW(70)/ 41.33    /,  MGOH+
1      UCAP(71)/0.076      /,DHF(71)/-260707.  /,XMW(71)/104.38    /,  MGSO3(L)
2      UCAP(72)/0.0        /,DHF(72)/-274083.  /,XMW(72)/ 85.34    /,  MGHCO3+
3      UCAP(73)/0.076      /,DHF(73)/-322469.  /,XMW(73)/120.38    /,  MGSO4(L)
4      UCAP(74)/0.076      /,DHF(74)/-270226.  /,XMW(74)/ 84.33    /,  MGCO3(L)
5      UCAP(75)/0.0        /,DHF(75)/- 57279.  /,XMW(75)/ 22.99    /,  NA+
6      UCAP(76)/0.076      /,DHF(76)/-112078.  /,XMW(76)/ 40.00    /,  NAOH(L)
7      UCAP(77)/0.0        /,DHF(77)/-218017.  /,XMW(77)/ 83.01    /,  NACO3-
8      UCAP(78)/0.076      /,DHF(78)/-222008.  /,XMW(78)/ 84.02    /,  NAHCO3(L
9      UCAP(79)/0.0        /,DHF(79)/-273076.  /,XMW(79)/119.06    /   NASO4-
0DATA  UCAP(80)/0.076      /,DHF(80)/-106651.  /,XMW(80)/ 85.01    /,  NANO3(L)
1      UCAP(81)/0.0        /,DHF(81)/- 40023.  /,XMW(81)/ 35.46    /   CL-
0DATA  TMIN¢50)/      273./,TMAX¢50)/     373./,ACP¢50)/      18.04/,  H2O¢LIQ)
1      TMIN¢51)/      298./,TMAX¢51)/     333./,ACP¢51)/      23.00/,  H<
2      TMIN¢52)/      298./,TMAX¢52)/     333./,ACP¢52)/    - 47.00/,  OH-
```

```
3       TMIN¢53I/       298./,TMAX¢53I/     333./,ACP¢53I/    - 16.00/, HSO3-
4       TMIN¢54I/       298./,TMAX¢54I/     333./,ACP¢54I/    -121.00/, SO3--
5       TMIN¢55I/       298./,TMAX¢55I/     333./,ACP¢55I/    - 99.00/, SO4--
6       TMIN¢56I/       298./,TMAX¢56I/     333./,ACP¢56I/    - 27.00/, HCO3-
7       TMIN¢57I/       298./,TMAX¢57I/     333./,ACP¢57I/    -132.00/, CO3--
8       TMIN¢58I/       298./,TMAX¢58I/     333./,ACP¢58I/    - 49.00/, NO3-
9       TMIN¢59I/       298./,TMAX¢59I/     333./,ACP¢59I/    - 13.00/  HSO4-
ODATA   TMIN¢60I/       298./,TMAX¢60I/     333./,ACP¢60I/       7.00/, H2SO3¢LI
1       TMIN¢61I/       298./,TMAX¢61I/     333./,ACP¢61I/      91.00/, H2CO3¢LI
2       TMIN¢62I/       298./,TMAX¢62I/     333./,ACP¢62I/      45.00/, CA<<
3       TMIN¢63I/       298./,TMAX¢63I/     333./,ACP¢63I/    -  2.00/, CAOH<
4       TMIN¢64I/       298./,TMAX¢64I/     333./,ACP¢64I/    - 76.00/, CASO3¢LI
5       TMIN¢65I/       298./,TMAX¢65I/     333./,ACP¢65I/    - 87.00/, CACO3¢LI
6       TMIN¢66I/       298./,TMAX¢66I/     333./,ACP¢66I/      18.00/, CAHCO3<
7       TMIN¢67I/       298./,TMAX¢67I/     333./,ACP¢67I/    -  8.00/, CASO4¢LI
8       TMIN¢68I/       298./,TMAX¢68I/     333./,ACP¢68I/    -  4.00/, CANO3<
9       TMIN¢69I/       298./,TMAX¢69I/     333./,ACP¢69I/      51.00/  MG<<
ODATA   TMIN¢70I/       298./,TMAX¢70I/     333./,ACP¢70I/       4.00/, MGOH<
1       TMIN¢71I/       298./,TMAX¢71I/     333./,ACP¢71I/    - 70.00/, MGSO3¢LI
2       TMIN¢72I/       298./,TMAX¢72I/     333./,ACP¢72I/      24.00/, MGHCO3<
3       TMIN¢73I/       298./,TMAX¢73I/     333./,ACP¢73I/    - 48.00/, MGSO4¢LI
4       TMIN¢74I/       298./,TMAX¢74I/     333./,ACP¢74I/    - 81.00/, MGCO3¢LI
5       TMIN¢75I/       298./,TMAX¢75I/     333./,ACP¢75I/      35.00/, NA<
6       TMIN¢76I/       298./,TMAX¢76I/     333./,ACP¢76I/    - 12.00/, NAOH¢LI
7       TMIN¢77I/       298./,TMAX¢77I/     333./,ACP¢77I/    - 97.00/, NACO3-
8       TMIN¢78I/       298./,TMAX¢78I/     333./,ACP¢78I/       3.00/, NAHCO3¢L
9       TMIN¢79I/       298./,TMAX¢79I/     333./,ACP¢79I/    - 64.00/  NASO4-
ODATA   TMIN¢80I/       298./,TMAX¢80I/     333./,ACP¢80I/    - 14.00/, NANO3¢LI
1       TMIN¢81I/       298./,TMAX¢81I/     333./,ACP¢81I/    - 51.00/  CL-
 END
```

```
@I FOR PHI
      FUNCTION PHI(X,EQS)
      DOUBLE PRECISION PHI,TEMP
      DOUBLE PRECISION F,CK,CT,X
C ***************************************************************
C *    FORM THE SUM OF THE SQARES OF THE N FUNC.
C ***************************************************************
      COMMON/FUNC/F(50),CK(50),CT(10)
      COMMON /LIMS/NF
      DIMENSION X(1)
      CALL EQS(X)
      TEMP=0.
      DO 1 I=1,NF
      TEMP=TEMP+F(I)*F(I)
    1 CONTINUE
      PHI=TEMP
      RETURN
      END
```

```
@I FOR PMPFAN
      SUBROUTINE PMPFAN(P,$)
      INCLUDE CMMN,LIST
      LOCSV=ISEQ(NL)
      LIS=ISTM(LOCSV,1)
      LOS=ISTM(LOCSV,6)
      KEY1=SV(LOS,2)
      IF(KEY1.NE.0) GO TO 1
      PRD(61) =P(1)
      RETURN 2
    1 CONTINUE
      P(1)=P(1)/14.696
      SV(LOS,5)=SV(LIS,5)
      SV(LOS,6)=P(1)
      Q=.1738423*SV(LIS,10)*SV(LIS,5)/SV(LIS,6)
      DP=2116.22*(P(1)-SV(LIS,6))
      DHP=(Q*DP/P(2))/33000.
      DO 3 I=10,119
      SV(LOS,I)=SV(LIS,I)
    3 CONTINUE
      DKW=DHP/1.341
      PRD(99)=DHP
      PRD(100)=DKW
      RETURN 2
      END
```

```
@I FOR PRELD
      SUBROUTINE PRELD
      INCLUDE CMMN,LIST
      DO 1 J=1,50
      I=ISEQ(J)
      NL=J
      IF(I.EQ.0) GO TO 2
      NAME=IDEOP(I)
      IN=MATCH(NAME)
      CALL PSULD(IN,I)
    1 CONTINUE
    2 DO 3 J=1,50
      SV(J,2)=1
    3 CONTINUE
      RETURN
      END
```

```
@I FOR PROCND
      SUBROUTINE PROCND
      INCLUDE CMMN.LIST
      WRITE(6,100)
      CALL DATIME
      WRITE(6, 1) LABEL
      WRITE(6, 2)
      WRITE(6, 3) PRD(1),PRD(2),PRD(3)
      WRITE(6, 4) PRD(4),PRD(5)
      WRITE(6, 5) PRD(6),PRD(7)
      WRITE(6, 7) PRD(8),PRD(9)
      WRITE(6, 8) PRD(10)
      WRITE(6,81) PRD(68)
      WRITE(6,99)
      IF(PRD(75).NE.0.) WRITE(6,98)
      WRITE(6, 9)
      WRITE(6,10)
      WRITE(6,11) PRD(11),PRD(13),PRD(19)
      WRITE(6,12) PRD(12),PRD(14),PRD(20)
      WRITE(6,13) PRD(15),PRD(21)
      WRITE(6,14) PRD(65),PRD(16),PRD(22)
      WRITE(6,15) PRD(66),PRD(17),PRD(23)
      WRITE(6,16) PRD(18),PRD(24)
      WRITE(6,161) PRD(70),PRD(71)
      WRITE(6,17) PRD(25),PRD(26)
      WRITE(6,18)
      WRITE(6,19) PRD(27),PRD(28)
      WRITE(6,20)PRD(29),PRD(67)
      WRITE(6,21) PRD(30),PRD(31)
      WRITE(6,40) PRD(61)
      WRITE(6,42)
      WRITE(6,43) PRD(62),PRD(63),PRD(64)
      WRITE(6,23)
      WRITE(6,24) PRD(32),PRD(33)
      WRITE(6,26)
      WRITE(6,27)
      WRITE(6,28)
      WRITE(6,29) PRD(34),PRD(43),PRD(51),PRD(53)
      WRITE(6,30) PRD(35),PRD(44),PRD(69),PRD(54)
      WRITE(6,31) PRD(36),PRD(45),PRD(55)
      WRITE(6,32) PRD(37),PRD(46),PRD(52),PRD(56)
      I RITE(6,33  PRD(38),PR) 47 , ?RD 57
```

```
      WRITE(6,34) PRD(39),PRD(48),PRD(58)
      WRITE(6,35) PRD(40),PRD(49),PRD(59)
      WRITE(6,36) PRD(41),PRD(50),PRD(60)
      WRITE(6,37) PRD(42)
    1 FORMAT(1H+,T36,13A6)
    2 FORMAT(35X,'PROCESS CONDITIONS')
    3 FORMAT(5X,'FLUE GAS RATE ='F10.2,' ACFM AT TFG =' G10.3,'DEG. F AND
     *PFG =' G10.3,'PSIA')
    4 FORMAT(10X,'SO2 ABSORBED =',G10.3,'%',T45,'NO  ABSORBED =',G10.3,'
     *%')
    5 FORMAT(10X,'SO2 OXIDIZED =',G10.3,'%',T45,'NO2 ABSORBED =',G10.3,'
     *%')
    7 FORMAT(/,T14,'LIMESTONE RATE =',G10.3,' LBS/MIN =',G10.3,' % THEOR
     *ETICAL')
    8 FORMAT(13X,'LIMESTONE COMPOSITION =',G10.3,'MOLES (MG)/MOLES(CA+MG
     *)')
   81 FORMAT(13X,'LIME REACTED IN BOILER =',G10.3,'%')
    9 FORMAT(10X,'LIME HYDRATED',T45,'SOLIDS DISSOLVED IN SCRUBBER')
   10 FORMAT(13X,'IN SYSTEM',T58,'LA',T71,'SF')
   11 FORMAT(20X,'CAO =',G10.3,'%',T45,'CA(OH)2 =',G10.3,'%',2X,G10.3,'%
     *')
   12 FORMAT(20X,'MGO =',G10.3,'%',T45,'MG(OH)2 =',G10.3,'%',2X,G10.3,'%
     *')
   13 FORMAT(13X,'IN SCRUBBER',T45,'CACO3   =',G10.3,'%',2X,G10.3,'%')
   14 FORMAT(20X,'CAO =',G10.3,'%',T45,'MGCO3   =',G10.3,'%',2X,G10.3,'%
     *')
   15 FORMAT(20X,'MGO =',G10.3,'%',T45,'MGSO4   =',G10.3,'%',2X,G10.3,'%
     *')
   16 FORMAT(44X,'NA2O    =',G10.3,'%',2X,G10.3,'%')
  161 FORMAT(44X,'NACL    =',G10.3,'%',2X,G10.3,'%')
   17 FORMAT(13X,'FLY ASH RATE =',G10.3,'LBS/MIN =',G10.3,' LBS FA/LB SU
     *LFUR IN COAL')
   18 FORMAT( 3X, 'EQUIPMENT PARAMETERS;')
   19 FORMAT(5X,'SOLIDS IN CLARIFIER BOTTOMS =' ,G10.3,' WT.%     CLARIF
     *IER EFFICIENCY =',G10.3,'%')
   20 FORMAT(5X,'SOLIDS IN FILTER    BOTTOMS =',G10.3,' WT.%     FILTER
     *   EFFICIENCY =',G10.3,'%')
   21 FORMAT(5X,'TEMP. OF COOLED G-S STREAM  =',G10.3,'DEG F TEMP. OF RE
     *HEATED STACK GAS =',G10.3,'DEG. F')
   23 FORMAT( 3X,'INITIAL VALUES ;')
   24 FORMAT(10X,'SCRUBBER FEED RATE =',G10.3,'GPM H2O SF/1000 ACFM FG
     *=',F10.2,' GPM H2O')
   26 FORMAT(/3X,'INPUT STREAM COMPOSITIONS ;')
```

```
 27 FORMAT(7X,'FLUE GAS',T29,'LIMESTONE',T53,'FLY ASH',T74,,'WATER MAK
   *EUP')
 28 FORMAT(4X,'COMP.  MOLE %'T26,'COMP.      WT.%',T48,'COMP.          W
   *T.%',T74,'COMP.    MG/L')
 29 FORMAT(6X,'SO2 =',G10.3,T26,'CAO   =',G10.3,T48,'NA2O      =',G10.3
   *,T75,'SO3 =',G10.3)
 30 FORMAT(6X,'CO2 =',G10.3,T26,'MGO   =',G10.3,T48,'NACL      =',G10.3
   *,T75,'CO3 ='G10.3)
 31 FORMAT(6X,'NO  =',G10.3,T26,'CACO3 =',G10.3,T47,'INSOLUBLE',T75,'S
   *O4 =',G10.3)
 32 FORMAT(6X,'NO2 =',G10.3,T26,'MGCO3 =',G10.3,T48,'FLY ASH  =',G10.3
   *,T75,'NO3 =',G10.3)
 33 FORMAT(6X,'O2  =',G10.3,T26,'CASO3 =',G10.3,T75,'CA  =',G10.3)
 34 FORMAT(6X,'CO  =',G10.3,T26,'MGSO3 =',G10.3,T75,'MG  =',G10.3)
 35 FORMAT(6X,'N2  =',G10.3,T26,'CASO4 =',G10.3,T75,'NA  =',G10.3)
 36 FORMAT(6X,'HCL =',G10.3,T26,'MGSO4 =',G10.3,T75,'CL  =',G10.3)
 37 FORMAT(6X,'H2O =',G10.3,)
 40 FORMAT(5X,'I.D. FAN EXIT PRESSURE =',G10.3,' PSIA')
 42 FORMAT(5X,'UNIT PRESSURE LOSSES,PSIA')
 43 FORMAT(13X,'COOLER =',G10.3,T37,'SCRUBBER =',G10.3,T62,'REHEATER =
   *',G10.3)
 98 FORMAT(1H+,9X,'NO')
 99 FORMAT(13X,'PRECIPITATION ALLOWED IN SCRUBBER')
100 FORMAT(1H1)
    RETURN
    END
```

```
@I FOR PROLOG
      SUBROUTINE PROLOG
      INCLUDE CMMN,LIST
C *      THIS ROUTINE DECODES
C *      THE PROCESS LOGIC AND
C *      EXECUTES
      P(1)=P(1)
      L1=0
      I=0
    1 I=I+1
      IF(ISEQ(I).EQ.0) RETURN
      IF(I.EQ.LSRL(1)) GO TO 3
      GO TO 4
    3 L2=0
      L3=0
    4 IF(I.EQ.LSRL(2)) L3=0
      NEXN=ISEQ(I)
      NAME=IDEQP(NEXN)
      NEIN=MATCH(NAME)
      NL=I
      CALL PSULD(NEIN,NEXN)
      IF(I.EQ.LERL(3).AND.L3.EQ.0) I=LSRL(3)-1
      IF(I.EQ.LERL(2).AND.L2.EQ.0) I=LSRL(2)-1
      IF(I.EQ.LERL(1).AND.L1.EQ.0) I=LSRL(1)-1
      GO TO 1
      END
```

```
@I FOR PROSIM
C ************************************
C *      THIS IS THE DRIVING PROGRAM   *
C ************************************
      INCLUDE LCMN,LIST
      INCLUDE PHYSD,LIST
      CALL READIN
      CALL PRELD
      CALL PROCND
      CALL PROLOG
      CALL PTSUM
      CALL ARTWRK
      END
```

```
@I FOR PSULD
      SUBROUTINE PSULD(IN,IX)
      INCLUDE CMMN.LIST
C
      DO 200 I=1,8
      P(I)=PA(IX,I)
  200 CONTINUE
C
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13), IN
    1 CALL FLUGAS(P,$100)
    2 CALL STKGAS(P,$100)
    3 CALL WTRMKP(P,$100)
    4 CALL FLTRBM(P,$100)
    5 CALL SCRUBR(P,$100)
    6 CALL CLPHTR(P,$100)
    7 CALL FILTER(P,$100)
    8 CALL DIVDER(P,$100)
    9 CALL EQMIXR(P,$100)
   10 CALL EFHOLD(P,$100)
   11 CALL PMPFAN(P,$100)
   12 CALL CLRFYR(P,$100)
   13 CALL OVALMB(P,$100)
  100 RETURN
      END
```

```
@I FOR PTSUM
      SUBROUTINE PTSUM
      INCLUDE CMMN,LIST
      INCLUDE LCMN,LIST
      DIMENSION TM(6),TMG(6),TML(6),TMS(6),WT(6),WG(6),WL(6),WS(6),D(6),
     *HT(6),HG(6),HL(6),HS(6),IPSV(6)
C
      DIMENSION IFRMT(606),IRRAY(9)
      DATA (IFRMT(I),I=  1,  6)/36H( 4X,'   FLOW RATE (G/SEC)          ',6(/
      DATA (IFRMT(I),I=  7, 12)/36H( 4X,'             (G-MOLE/SEC) ',6(/
      DATA (IFRMT(I),I= 13, 18)/36H( 4X,'   TEMPERATURE (DEG. K)       ',6(/
      DATA (IFRMT(I),I= 19, 24)/36H( 4X,'   PRESSURE   (ATM)           ',6(/
      DATA (IFRMT(I),I= 25, 30)/36H( 4X,'   ENTHALPY   (CAL/SEC)       ',6(/
      DATA (IFRMT(I),I= 31, 36)/36H( 4X,'   SOLIDS     (WT %)          ',6(/
      DATA (IFRMT(I),I= 37, 42)/36H( 4X,'   FLOW RATE (G/SEC)          ',6(/
      DATA (IFRMT(I),I= 43, 48)/36H( 4X,'             (G-MOLE/SEC) ',6(/
      DATA (IFRMT(I),I= 49, 54)/36H( 4X,'   ENTHALPY   (CAL/SEC)       ',6(/
      DATA (IFRMT(I),I= 55, 60)/36H( 4X,'   COMP       (G-MOLES/SEC)',6(/
      DATA (IFRMT(I),I= 61, 66)/36H( 4X,'     SO2                      ',6(/
      DATA (IFRMT(I),I= 67, 72)/36H( 4X,'     CO2                      ',6(/
      DATA (IFRMT(I),I= 73, 78)/36H( 4X,'     NO                       ',6(/
      DATA (IFRMT(I),I= 79, 84)/36H( 4X,'     NO2                      ',6(/
      DATA (IFRMT(I),I= 85, 90)/36H( 4X,'     O2                       ',6(/
      DATA (IFRMT(I),I=595,600)/36H( 4X,'     CO                       ',6(/
      DATA (IFRMT(I),I= 91, 96)/36H( 4X,'     N2                       ',6(/
      DATA (IFRMT(I),I= 97,102)/36H( 4X,'     HCL                      ',6(/
      DATA (IFRMT(I),I=103,108)/36H( 4X,'     H2O                      ',6(/
      DATA (IFRMT(I),I=109,114)/36H( 4X,'   FLOW RATE (G/SEC)          ',6(/
      DATA (IFRMT(I),I=115,120)/36H( 4X,'             (G-MOLES/SEC)',6(/
      DATA (IFRMT(I),I=121,126)/36H( 4X,'   ENTHALPY   (CAL/SEC)       ',6(/
      DATA (IFRMT(I),I=127,132)/36H( 4X,'   COMP       (G-MOLES/SEC)',6(/
      DATA (IFRMT(I),I=133,138)/36H( 4X,'     SO2                      ',6(/
      DATA (IFRMT(I),I=139,144)/36H( 4X,'     CO2                      ',6(/
      DATA (IFRMT(I),I=145,150)/36H( 4X,'     SO3                      ',6(/
      DATA (IFRMT(I),I=151,156)/36H( 4X,'     N2O5                     ',6(/
      DATA (IFRMT(I),I=157,162)/36H( 4X,'     CAO                      ',6(/
      DATA (IFRMT(I),I=163,168)/36H( 4X,'     MGO                      ',6(/
      DATA (IFRMT(I),I=169,174)/36H( 4X,'     NA2O                     ',6(/
      DATA (IFRMT(I),I=175,180)/36H( 4X,'     HCL                      ',6(/
      DATA (IFRMT(I),I=181,186)/36H( 4X,'     H2O                      ',6(/
      DATA (IFRMT(I),I=187,192)/36H( 4X,'   DENSITY    (G/ML AT T)     ',6(/
      DATA (IFRMT(I),I=193,198)/36H( 4X,'   IONIC STRENGTH             ',6(/
```

```
DATA (IFRMT(I),I=199,204)/36H( 4X,'   PH                        ',6(/
DATA (IFRMT(I),I=205,210)/36H( 4X,'   LIQ. H2O RATE (KG/SEC)    ',6(/
DATA (IFRMT(I),I=211,216)/36H( 4X,'     H+                      ',6(/
DATA (IFRMT(I),I=217,222)/36H( 4X,'     OH-                     ',6(/
DATA (IFRMT(I),I=223,228)/36H( 4X,'     HSO3-                   ',6(/
DATA (IFRMT(I),I=229,234)/36H( 4X,'     SO3=                    ',6(/
DATA (IFRMT(I),I=235,240)/36H( 4X,'     SO4=                    ',6(/
DATA (IFRMT(I),I=241,246)/36H( 4X,'     HCO3-                   ',6(/
DATA (IFRMT(I),I=247,252)/36H( 4X,'     CO3=                    ',6(/
DATA (IFRMT(I),I=253,258)/36H( 4X,'     NO3-                    ',6(/
DATA (IFRMT(I),I=259,264)/36H( 4X,'     HSO4-                   ',6(/
DATA (IFRMT(I),I=265,270)/36H( 4X,'     H2SO3(L)                ',6(/
DATA (IFRMT(I),I=271,276)/36H( 4X,'     H2CO3(L)                ',6(/
DATA (IFRMT(I),I=277,282)/36H( 4X,'     CA++                    ',6(/
DATA (IFRMT(I),I=283,288)/36H( 4X,'     CAOH+                   ',6(/
DATA (IFRMT(I),I=289,294)/36H( 4X,'     CASO3(L)                ',6(/
DATA (IFRMT(I),I=295,300)/36H( 4X,'     CACO3(L)                ',6(/
DATA (IFRMT(I),I=301,306)/36H( 4X,'     CAHCO3+)                ',6(/
DATA (IFRMT(I),I=307,312)/36H( 4X,'     CASO4(L)                ',6(/
DATA (IFRMT(I),I=313,318)/36H( 4X,'     CANO3+                  ',6(/
DATA (IFRMT(I),I=319,324)/36H( 4X,'     MG++                    ',6(/
DATA (IFRMT(I),I=325,330)/36H( 4X,'     MGOH+                   ',6(/
DATA (IFRMT(I),I=331,336)/36H( 4X,'     MGSO3(L)                ',6(/
DATA (IFRMT(I),I=337,342)/36H( 4X,'     MGHCO3+                 ',6(/
DATA (IFRMT(I),I=343,348)/36H( 4X,'     MGSO4(L)                ',6(/
DATA (IFRMT(I),I=349,354)/36H( 4X,'     MGCO3(L)                ',6(/
DATA (IFRMT(I),I=355,360)/36H( 4X,'     NA+                     ',6(/
DATA (IFRMT(I),I=361,366)/36H( 4X,'     NAOH(L)                 ',6(/
DATA (IFRMT(I),I=367,372)/36H( 4X,'     NACO3-                  ',6(/
DATA (IFRMT(I),I=373,378)/36H( 4X,'     NAHCO3(L)               ',6(/
DATA (IFRMT(I),I=379,384)/36H( 4X,'     NASO4-                  ',6(/
DATA (IFRMT(I),I=385,390)/36H( 4X,'     NANO3-                  ',6(/
DATA (IFRMT(I),I=391,396)/36H( 4X,'     CL-                     ',6(/
DATA (IFRMT(I),I=397,402)/36H( 4X,'   FLOW RATE (G/SEC)         ',6(/
DATA (IFRMT(I),I=403,408)/36H( 4X,'              (G-MOLES/SEC)',6(/
DATA (IFRMT(I),I=409,414)/36H( 4X,'   ENTHALPY (CAL/SEC)        ',6(/
DATA (IFRMT(I),I=415,420)/36H( 4X,'     CAO                     ',6(/
DATA (IFRMT(I),I=421,426)/36H( 4X,'     CA(OH)2                 ',6(/
DATA (IFRMT(I),I=427,432)/36H( 4X,'     CACO3                   ',6(/
DATA (IFRMT(I),I=433,438)/36H( 4X,'     CASO3                   ',6(/
DATA (IFRMT(I),I=439,444)/36H( 4X,'     CASO3*1/2H2O            ',6(/
DATA (IFRMT(I),I=445,450)/36H( 4X,'     CASO4                   ',6(/
DATA (IFRMT(I),I=451,456)/36H( 4X,'     CASO4*2H2O              ',6(/
```

```
      DATA (IFRMT(I),I=457,462)/36H( 4X,'      MGO                   ',6(/
      DATA (IFRMT(I),I=463,468)/36H( 4X,'      MG(OH)2               ',6(/
      DATA (IFRMT(I),I=469,474)/36H( 4X,'      MGCO3                 ',6(/
      DATA (IFRMT(I),I=475,480)/36H( 4X,'      MGCO3*3H2O            ',6(/
      DATA (IFRMT(I),I=481,486)/36H( 4X,'      MGCO3*5H2O            ',6(/
      DATA (IFRMT(I),I=487,492)/36H( 4X,'      MGSO3                 ',6(/
      DATA (IFRMT(I),I=493,498)/36H( 4X,'      MGSO3*3H2O            ',6(/
      DATA (IFRMT(I),I=499,504)/36H( 4X,'      MGSO3*6H2O            ',6(/
      DATA (IFRMT(I),I=505,510)/36H( 4X,'      MGSO4                 ',6(/
      DATA (IFRMT(I),I=511,516)/36H( 4X,'      NA2O                  ',6(/
      DATA (IFRMT(I),I=517,522)/36H( 4X,'      INSOLUBLE FLY ASH     ',6(/
      DATA (IFRMT(I),I=523,528)/36H( 4X,'      NACL                  ',5(/
      DATA (IFRMT(I),I=529,534)/36H( 4X,'    TOT FLOW RATE (LBS/MIN)',6(/
      DATA (IFRMT(I),I=535,540)/36H( 4X,'              (LB-MOLES/MIN)',6(/
      DATA (IFRMT(I),I=541,546)/36H( 4X,'    GAS FLOW RATE (LBS/MIN)',6(/
      DATA (IFRMT(I),I=547,552)/36H( 4X,'              (LB-MOLES/MIN)',6(/
      DATA (IFRMT(I),I=553,558)/36H( 4X,'              (ACFM)       ',6(/
      DATA (IFRMT(I),I=559,564)/36H( 4X,'    LIQ FLOW RATE (LBS/MIN)',6(/
      DATA (IFRMT(I),I=565,570)/36H( 4X,'              (LB-MOLES/MIN)',6(/
      DATA (IFRMT(I),I=571,576)/36H( 4X,'           (1000 LBS H2O/MIN)',6(/
      DATA (IFRMT(I),I=577,582)/36H( 4X,'    LIQ DENSITY (LBS/GALQT)',6(/
      DATA (IFRMT(I),I=583,588)/36H( 4X,'    SOL FLOW RATE (LBS/MIN)',6(/
      DATA (IFRMT(I),I=589,594)/36H( 4X,'              (LB-MOLES/MIN)',6(/
      DATA (IFRMT(I),I=601,606)/36H( 4X,'                   (GAL/MIN)',6(/
      DATA (IRRAY(I),I=7,8)/12H2X,G10.5)) /
C
     ODEFINE        DL(T)       = +.99995282
     1                            +.46726616E-4 *  (T-273.16)
     2                            -.74105074E-5 *  (T-273.16)**2
     3                            +.41079583E-7 *  (T-273.16)**3
     4                            -.13370708E-9 *  (T-273.16)**4
      CEU=60./453.6
      IS=1
    2 IC=0
      DO 1 IJ=IS,25
      IF(SV(IJ,5).EQ.0.) GO TO 1
      IC=IC+1
      IPSV(IC)=IJ
      IF(IC.EQ.6) GO TO 3
    1 CONTINUE
    3 CONTINUE
      IS=IJ+1
C
```

```
C
      DO 4 I=1,IC
      SUM=0.
      LSV=IPSV(I)
      DO 5 J=1,9
      JG=10+J
    5 SUM=SUM+SV(LSV,JG)*XMW(J)
      WG(I)=SUM
      CALL DHG(LSV,DH)
      HG(I)=DH
      TMG(I)=SV(LSV,10)
      CALL TOLISP(LSV)
      SUM=0.
      DO 15 JT=31,39
   15 SUM=SUM+SV(LSV,JT)
      TML(I)=SUM
      WL(I)=SV(LSV,31)*XMW(1)+SV(LSV,32)*XMW(2)+SV(LSV,33)*XMW(54)+SV(LS
     *V,34)*(XMW(58)+XMW(4))+SV(LSV,35)*XMW(21)+SV(LSV,36)*XMW(28)+SV(LS
     *V,37)*XMW(37)+SV(LSV,38)*XMW(38)+SV(LSV,39)*XMW(50)
      CALL DHL(LSV,DH)
      HL(I)=DH
C
C
C
      SUM=0
      DO 7 J=51,81
    7 SUM=SUM+SV(LSV,J)*XMW(J)
      D(I)=DL(SV(LSV,5))*(1.+SUM/1000.)
      IF(SV(LSV,50).EQ.0.) D(I)=0.
C
      SUM=0.
      SUM1=0.
      DO 8 J=1,19
      JS=100+J
      JW=20+J
      SUM=SUM+XMW(JW)*SV(LSV,JS)
      SUM1=SUM1+SV(LSV,JS)
    8 CONTINUE
      WS(I)=SUM
      TMS(I)=SUM1
      WT(I)=WG(I)+WL(I)+WS(I)
      SV(LSV,100)=100.*WS(I)/WT(I)
      TM(I)=TMG(I)+TML(I)+TMS(I)
```

```
      CALL DHS(LSV,DH)
      HS(I)=DH
      HT(I)=HG(I)+HL(I)+HS(I)
    4 CONTINUE
C
C
C
      WRITE(6,100)
  100 FORMAT(1H1)
      CALL DATIME
      WRITE(6,101) LABEL
  101 FORMAT(1H+,45X,13A6)
      WRITE(6,110) (IPSV(I),I=1,IC)
  110 FORMAT(4X,'STREAM NUMBER',18X,6(I2,10X))
      WRITE(6,102)
  102 FORMAT(4X,'TOTAL STREAM')
      CALL MOVE(1)
      WRITE(6,IRRAY) (WT(I),I=1,IC)
      CALL MOVE(7)
      WRITE(6,IRRAY) (TM(I),I=1,IC)
      CALL PLINE(5,13)
      CALL PLINE(6,19)
      CALL MOVE(25)
      WRITE(6,IRRAY) (HT(I),I=1,IC)
      CALL PLINE(100,31)
      WRITE(6,103)
  103 FORMAT(/4X,'GAS PHASE')
      CALL MOVE(37)
      WRITE(6,IRRAY) (WG(I),I=1,IC)
      CALL MOVE(43)
      WRITE(6,IRRAY) (TMG(I),I=1,IC)
      CALL MOVE(49)
      WRITE(6,IRRAY) (HG(I),I=1,IC)
      CALL MOVE(55)
      WRITE(6,IRRAY)
      LES=61
      DO 9 I=11,15
      CALL PLINE(I,LES)
    9 LES=LES+6
      CALL PLINE(16,595)
      DO 18 I=17,19
      CALL PLINE(I,LES)
    8 .ES=.ES+6
```

```
      WRITE(6,104)
  104 FORMAT(/4X,'LIQUID PHASE')
      CALL MOVE(109)
      WRITE(6,IRRAY) (WL(I),I=1,IC)
      CALL MOVE(115)
      WRITE(6,IRRAY) (TML(I),I=1,IC)
      CALL MOVE(121)
      WRITE(6,IRRAY) (HL(I),I=1,IC)
      CALL MOVE(127)
      WRITE(6,IRRAY)
      LES=133
      DO 10 I=31,39
      CALL PLINE(I,LES)
   10 LES=LES+6
      CALL MOVE(187)
      WRITE(6,IRRAY) (D(I),I=1,IC)
      CALL PLINE(48,193)
      CALL PLINE(49,199)
      CALL DATIME
      WRITE(6,101) LABEL
      WRITE(6,110) (IPSV(I),I=1,IC)
      DO 11 I=1,6
      LSV=IPSV(I)
   11 SV(LSV,50)=SV(LSV,50)/1000.
      CALL PLINE(50,205)
      WRITE(6,105)
  105 FORMAT(4X,'   COMP (G-MOLES/KG H2O)')
      LES=211
      DO 12 I=51,81
      CALL PLINE(I,LES)
   12 LES=LES+6
      WRITE(6,106)
  106 FORMAT(/4X,'SOLID PHASE')
      CALL MOVE(397)
      WRITE(6,IRRAY) (WS(I),I=1,IC)
      CALL MOVE(403)
      WRITE(6,IRRAY) (TMS(I),I=1,IC)
      CALL MOVE(409)
      WRITE(6,IRRAY) (HS(I),I=1,IC)
      WRITE(6,100)
      CALL DATIME
      WRITE(6,101) LABEL
      WRITE(6,110) (IPSV(I),I=1,IC)
```

```
      WRITE(6,107)
  107 FORMAT(4X,'   COMP (G-MOLES/SEC)')
      LES=415
      DO 13 I=101,119
      CALL PLINE(I,LES)
   13 LES=LES+6
      WRITE(6,108)
  108 FORMAT(/4X,'SUMMARY IN ENGINEERING UNITS')
      DO 14 I=1,6
      WT(I)=CEU*WT(I)
      WG(I)=CEU*WG(I)
      WL(I)=CEU*WL(I)
      WS(I)=CEU*WS(I)
      TM(I)=CEU*TM(I)
      TMG(I)=CEU*TMG(I)
      TML(I)=CEU*TML(I)
      TMS(I)=CEU*TMS(I)
      D(I)=8.345*D(I)
      LSV=IPSV(I)
   14 SV(LSV,50)=CEU*SV(LSV,50)
      CALL MOVE(529)
      WRITE(6,IRRAY) (WT(I),I=1,IC)
      CALL MOVE(535)
      WRITE(6,IRRAY) (TM(I),I=1,IC)
      CALL MOVE(541)
      WRITE(6,IRRAY) (WG(I),I=1,IC)
      CALL MOVE(547)
      WRITE(6,IRRAY) (TMG(I),I=1,IC)
      DO 16 I=1,IC
   16 TMG(I)=(TMG(I)/273.16)*(SV(I,5)/SV(I,6))*359.0
      CALL MOVE(553)
      WRITE(6,IRRAY) (TMG(I),I=1,IC)
      CALL MOVE(559)
      WRITE(6,IRRAY) (WL(I),I=1,IC)
      CALL MOVE(565)
      WRITE(6,IRRAY) (TML(I),I=1,IC)
      CALL PLINE(50,571)
      DO 17 I=1,IC
      LSV=IPSV(I)
   17 SV(LSV,50)=1000.*SV(LSV,50)/(8.345*DL(SV(LSV,5)))
      CALL PLINE(50,601)
      CALL MOVE(577)
      WRI E 5,IRRAY) (D(I),I= ,IC)
```

```
      CALL MOVE(583)
      WRITE(6,IRRAY) (WS(I),I=1,TC)
      CALL MOVE(589)
      WRITE(6,IRRAY) (TMS(I),I=1,IC)
      IF(IS.GE.26) RETURN
      GO TO 2
C
      SUBROUTINE PLINE(LOC,IFR)
C
      DIMENSION TP(6)
      L1=IPSV(1)
      L2=IPSV(2)
      L3=IPSV(3)
      L4=IPSV(4)
      L5=IPSV(5)
      L6=IPSV(6)
      TP(1)=SV(L1,LOC)
      TP(2)=SV(L2,LOC)
      TP(3)=SV(L3,LOC)
      TP(4)=SV(L4,LOC)
      TP(5)=SV(L5,LOC)
      TP(6)=SV(L6,LOC)
      IBGN=IFR+1
      IEND=IFR+5
      WRITE(6,200) (IFRMT(I),I=IBGN,IEND),(TP(I),I=1,TC)
  200 FORMAT(4X,4A6,A2,6(2X,G10.5))
      RETURN
C
      SUBROUTINE MOVE(IFR)
C
      DO 30 N=1,6
      LNE=IFR+(N-1)
   30 IRRAY(N)=IFRMT(LNE)
      RETURN
C
      END
```

```
C
C ****************************
C *       READS INPUT DATA     *
C ****************************
      SUBROUTINE READIN
      INCLUDE CMMN,LIST
      DIMENSION NS(10)
      DIMENSION IDO(80)
      DIMENSION PTEMP(8)
C
      P(1)=P(1)
      IORD=1
   10 GO TO (1,2,3,4,5), IORD
C*******************
C *       RUN I.D.  *
C*******************
    1 READ(5,100) LABEL
      WRITE(6,200)
      WRITE(6,201) LABEL
      WRITE(6,202)
      IORD=2
      GO TO 10
C ****************************
C *        PROCESS ARRAY      *
C ****************************
    2 READ(5,101,ERR=15,END=11) NUM,NAME,NS
      IDEQP(NUM)=NAME
      CALL ORDER(NS,NI,NO)
      DO 7 I=1,10
    7 ISTM(NUM,I)=NS(I)
      I=NUM
      WRITE(6,203) I,IDEQP(I),(ISTM(I,J),J=1,NI)
      IF(NO.EQ.0) GO TO 10
      NOP=5+NO
      WRITE(6,204) (ISTM(I,J),J=6,NOP)
      GO TO 10
   11 IORD=IORD+1
      GO TO 10
C ***************************************
C *        ORDER OF PROCESS CALCULATIONS *
C ***************************************
```

```
 3 READ 5,102  IDO
   IC=1
   LW=0
   LE=0
   NLP=1
   NRP=1
   LER=0
   NSRLP=0
   DO 8 I=1,80
   IF(IDO(I).LE.37B) GO TO 8    @ NON NUMERIC
   IF(IDO(I).EQ.51B) GO TO 9    @ (
   IF(IDO(I).EQ.40B) GO TO 16   @ )
   IF(IDO(I).EQ.56B) GO TO 12   @ .
   IF(IDO(I).EQ.50B) GO TO 13   @ *
   NSRLP=0
   LW=LW+1
   NC=NC+(IDO(I)-60B)*10**(2-LW)
   GO TO 8
12 NC=NC/10**(2-LW)
   ISEQ(IC)=NC
   NC=0
   LW=0
   IC=IC+1
   IF(LE.EQ.1) GO TO 14
   IF(LER.EQ.1) GO TO 17
   GO TO 8
13 LE=0
   GO TO 12
16 IF(NSLRP.EQ.1) GO TO 17
   LER=1
   NSLRP=1
   GO TO 12
 9 LSRL(NLP)=IC+1
   NLP=NLP+1
   NSRP=0
   GO TO 12
17 NRP=NLP-NRP-NSRP
   LERL(NRP)=IC-1
   NSRP=NSRP+1
   NRP=1
   LER=0
 8 CONTINUE
14 CONTINUE
```

```
      WRITE(6,205) IDO
      DO 18 I=1,3
      IB=LSRL(I)
      IE=LERL(I)
      IF(IB.GT.0) WRITE(6,206) ISEQ(IB),ISEQ(IE)
   18 CONTINUE
      IORD=IORD+1
      GO TO 10
    4 CONTINUE
C *********************************
C *      EQUIPMENT PARAMETERS     *
C *********************************
      READ (5,103,ERR=15,END=19) NUM,PTEMP
      DO 20 I=1,8
      PA(NUM,I)=PTEMP(I)
   20 CONTINUE
      GO TO 10
   19 IORD=IORD+1
      GO TO 10
    5 CONTINUE
C
      RETURN
C ***************************
C $      FORMATS            *
C ***************************
  100 FORMAT(13A6)
  101 FORMAT(I2,2X,A6,10I5)
  102 FORMAT(80R1)
  103 FORMAT(I2,F8.3,7F10.3)
  200 FORMAT(1H1)
  201 FORMAT(15X,13A6)
  202 FORMAT(45X,  'PROCESS DESCRIPTION'// 1X,   'EQUP. NO.',10X,'EQUIP.
     *NAME',10X,'INPUT STREAMS          OUTPUT STREAMS'/)
  203 FORMAT(4X,I5,17X,A6,2X,5I5)
  204 FORMAT(1H+,57X,5I5)
  205 FORMAT(//40X,'ORDER OF PROCESS CALCULATIONS',/15X,80R1)
  206 FORMAT(/20X, 'RECYCLE LOOP FROM',5X,I5,5X,'TO',5X,I5)
   15 RETURN
      END
```

```
@I FOR SCRUBR
      SUBROUTINE SCRUBR(P,$)
      INCLUDE CMMN,LIST
      DIMENSION XDSF(19),XDLA(19),WSF(19),TWSF(9),PP(2)  ,DNLA(9)
     *,DNSF(9),WLA(19),TWLA(9),XLH(2)
      DIMENSION TEMP(19)
     ODEFINE         DL(T)       = +.99995282
     1                              +.46726616E-4 *  (T-273.15)
     2                              -.74105074E-5 *  (T-273.16)**2
     3                              +.41079583E-7 *  (T-273.16)**3
     4                              -.13370708E-9 *  (T-273.16)**4
      LOCSV=ISEQ(NL)
      LIS1=ISTM(LOCSV,1)
      LIS2=ISTM(LOCSV,2)
      LOS1=ISTM(LOCSV,6)
      LOS2=ISTM(LOCSV,7)
      KEY1=SV(LOS2,2)
      IF(KEY1.NE.0) GO TO 1
      LSR=P(1)
      SF=P(2)
      PRD(33)=SF
      PRD(32)=1000*SF/PRD(1)
      NS=P(3)
      PRD(75)=NS
      XLH(1)=P(4)
      XLH(2)=P(5)
      PRD(65)=100*XLH(1)
      PRD(66)=100*XLH(2)
      SF=(453.6*8.27/60.)*SF
      SV(LIS2,50)=SF
      SV(LOS2,50)=SF
      IOPT=2
      WNES=0
      INIT=0
      YMX=-2.302
      YMN=-20.
      XMX=0.
      XMN=SF
      IC=1
      IL=0
      ISF=0
      RETURN 2
```

```
    1 CONTINUE
      TAS=SV(LOS1,5)
      SV(LOS2,5)=TAS
      IF(ISF.EQ.1) GO TO 14
      SF=(453.6*8.345*DL(TAS)/60.)*P(2)
   14 CONTINUE
      TE=2*SF-SV(LIS2,50)
      TE=TE/SF
      IF(ABS(1-TE).LT..005) GO TO 2
      SV(LIS2,50)=SF*TE
      RETURN 2
    2 CONTINUE
C
      CALL XSODIS(NS,XSD,XDSF)
      CALL DNTSF(LIS2,XDSF,DNSF ,WSF,TWSF)
      CALL XSODIS(NS,XLD,XDLA)
      CALL DNTLA(LIS1,XLH,XDLA,DNLA,WLA,TWLA)
C
      DO 3 J=1,9
      I=30+J
      SV(LOS2,I)=DNSF(J)+DNLA(J)+DNAT(J)
    3 CONTINUE
      SV(LOS2,39)=SV(LOS2,39)-WV
      TCO2=SV(LOS2,32)
      PP(2)=(SV(LOS1,12)/SV(LOS1,10))*SV(LOS1,6)
      PSO  =(SV(LOS1,11)/SV(LOS1,10))*SV(LOS1,6)
      PSOL=LOG(PSO)
      DO 12 JT=1,19
      JS=JT+100
      SV(LOS2,JS)=TEMP(JT)
   12 CONTINUE
      IF(INIT.NE.0) GO TO 10
      XWS=PP(2)
      CALL INTIT(LOS2,IOPT,XWS,WNES)
      INIT=1
   10 CONTINUE
C
      CALL EQUILB(LOS2,IOPT,TAS,PP,WNES,NS)
C
      WRITE(6,200)
  200 FORMAT(  46X,'SCRUBBER BOTTOMS')
      TEP=ABS((PSO-PP(1))/PSO)
      WRITE(6,  )
```

```
  400 FORMAT(/43X, 'CONVERGENCE PARAMETERS'//
      WRITE(6,401) PSO,XSR
  401 FORMAT(20X,  'DESIRED PSO2 =',1PE12.5,10X,'SR/SF =',1PE12.5/)
      WRITE(6,402) TCO2,XA(2)
  402 FORMAT(20X,  'DESIRED TOT. CO2 =',1PE12.5,10X,'XA(CO2) =',1PE12.5)
      IF(TEP.LT.1.E-3) GO TO 5
      XSR=SV(LSR,50)
      IF(XSR.EQ.0..AND.PP(1).LT.PSO) ISF=1
      IF(ISF.EQ.1) XSR=SF
      PY=LOG(PP(1))
      CALL CONVSO(XSR,XMX,XMN,PY,YMX,YMN,PSOL,IC)
      IF(ISF.NE.1) SV(LSR,50)=XSR
      IF(ISF.NE.1) GO TO 11
      SF=XSR
      SV(LIS2,50)=SF
   11 CONTINUE
      GO TO 8
    5 L2=1
      IC=0
      TEC=(TCO2-SV(LOS2,32))/(SV(LIS1,101)+SV(LIS1,108))
      IF(ABS(TEC).LT.5.E-4) GO TO 9
      CALL CONVCO(XA(2),TCO2,SV(LOS2,32),SV(LIS1,12),IL)
      IL=1
      GO TO 8
    9 L1=1
    8 SV(LOS2,50)=1000*SV(LOS2,50)
      DO 13 IT=1,19
      JS=IT+100
      TEMP(IT)=SV(LOS2,JS)
   13 CONTINUE
      SV(LOS2,101)=0.
      SV(LOS2,108)=0.
      SV(LOS2,116)=0.
      SV(LOS2,117)=0.
      SV(LOS2,119)=0.
      IF(TAS.LT.373.16) SV(LOS2,110)=0.
C
      DO 6 J=1,19
      I=100+J
      I=I
      IF(NS.NE.0) SV(LOS2,I)=0.
      SV(LOS2,I)=SV(LOS2,I)+WSF(J)+WLA(J)
    6 CONTINUE
```

```
C
      SV(LOS2,118)=SV(LIS1,118)+SV(LIS2,118)
      DO 7 J=1,9
      I=30+J
      SV(LOS2,I)=SV(LOS2,I)+TWSF(J)+TWLA(J)
    7 CONTINUE
      CALL SVDMP
      CALL EXIT
      RETURN 2
      END
```

```
@I FOR SOLNEQ
      SUBROUTINE SOLNEQ(X,NS,CM,PP,TK,IOPT)
      COMMON /LIMS/NF,NT
      COMMON /TRFR/TMW,WACT,PH,AZ
      COMMON/TEMPER/AC,BC,CPK(2),NHY(5),WS(10)
      COMMON /ACOEF/GLN(50),DGLN(50),IZ(50),FI
      COMMON/FUNC/F(50),CK(50),CT(10)
      COMMON /PPRESS/PK(2),L1,L2,L3,L4,REN
      COMMON /NEW/IV(50),IE(50),CL,TH(11)
      EXTERNAL EQS,PARTL
      DOUBLE PRECISION F,CK,CT,XLN,TH
      DIMENSION X(1),CM(11),LB(100),PP(2),XLN(50),WTS(10)
C
      DATA ICME1/037677557674/ICMV1/037773576 7371/ICME2/037562 5373147/
     1ICMV2/031527731 6717/ICME3/0373576766773/ICMV3/027757667756 7/ICME4/
     2036737777577 7/ICMV4/017777757767 7/ICME5/0357777700037/ICMV5/037777
     30001777/ICME6/033776007777 7/ICMV6/0374007777777/ICME7/027701777777
     47/ICMV7/000377777777 7/
      DATA (LB(I),I=1,74)/'H+    ','      ','OH-   ','      ','HSO3- ','
     1      ','SO3-- ','      ','SO4-- ','      ','HCO3- ','      ','CO3
     2-- ','      ','NO3-  ','      ','HSO4- ','      ','H2SO3 ','
     3','H2CO3 ','      ',                    'CA++  ','      ','CAOH+ ','
     B     '                                                          ,'
     4CASO3 ','      ','CACO3 ','      ','CAHCO3','+     ','CASO4 ','
     5   ','CANO3+','      ','CACO3(','S)    ','CASO4(','S)    ','CASO3(
     C','S)    ','CA(OH)','2(S)  ',                    'MG++  ','      ','MGOH+
     6','      ','MGSO3 ','      ','MGHCO3','+     ','MGSO4 ','      ','
     7MGCO3 ','      ','MG(OH)','2(S)  ','MGCO3(','S)    ',
     D                    'NA+   ','      ','NAOH  ','      ','NACO3-','
     8   ','NAHCO3','      ','NASO4-','      ','NANO3 ','      ','CL-
     9','      '/
      DATA (WTS(I),I=1,9)/64.066,44.011,80.066,108.016,56.08,40.32,61.98
     *2,36.465,18.016/
      DATA LB(75)/:MGSO3('/LB(76)/'S)    :/
C
      DEFINE  G(I)=EXP(GLN(I))
C
  100 FORMAT(1H1)
  101 FORMAT(//38X,'AQUEOUS SOLUTION EQUILIBRIA '/)
  102 FORMAT(15X, 'COMPONENT          MOLALITY              ACTIVITY
     1        , 'ACTIVITY COEFFICIENT')
  103 FORMAT(15X,A6,A4,1PE15.3,4X,1PE15.3,1PE18 3)
```

```
  104 FORMAT(1X,/15X,'  PH = ',F10.3,10X,'IONIC STRENGTH = ',1PE10.5,
     110X,' RES. E.N. = ',1PE10.3)
  105 FORMAT(42X,  'INPUT MOLES'/)
  106 FORMAT(4X,'SO3  =',1PE10.5,4X,'N2O5 =',1PE10.5,4X,'CAO  =',1PE10.5
     1,4X,                    'MGO  =',1PE10.5/4X,'NA2O =',1PE10.5,4X,
     2                    'HCL  =',1PE10.5,4X,'H2O  =',1PE10.5)
  107 FORMAT(15X,'H2O           ',34X,1PE18.3)
  108 FORMAT(4X,'SO3  =',1PE10.5,4X,'N2O5 =',1PE10.5,4X,'CAO  =',1PE10.5
     1,4X,                    'MGO  =',1PE10.5/4X,'NA2O =',1PE10.5,4X,
     2                    'HCL  =',1PE10.5)
  109 FORMAT(1H+,70X,'TEMPERATURE',F10.3,'  DEG. C')
  111 FORMAT(4X,'SO2  =',1PE10.5)
  112 FORMAT(4X,'CO2  =',1PE10.5)
  113 FORMAT(4X,'PSO2 =',1PE10.5,2X,'ATM.')
  114 FORMAT(4X,'PCO2 =',1PE10.5,2X,'ATM.')
  115 FORMAT(/ 38X,'MOLECULAR WATER = ',1PE10.5,' KGS.')
  116 FORMAT(1H+,44X,'% WT. SOLIDS = ',1PE10.5,4X,'WT. NEQ SOLIDS = ',
  116*1PE10.5)
  117 FORMAT(40X,'MOLES TOTAL WATER = ',1PE10.5)
  118 FORMAT(1H+,T8,'SPECIFIED')
  200 FORMAT(1X,'NO. FUNCTIONS ',I5,10X,'NO. VARIABLES ',I5)
C ***************************************
C *    INITIALIZATION OF ROUTINE        *
C ***************************************
      EPS=1.E-4
      L1=AND(IOPT,1)
      L2=AND(IOPT,2)
      L3=AND(IOPT,4)
      L4=AND(IOPT,8)
      IF(NS.EQ.0) GO TO 20
      IEQ=3777637037778
      IVN=3747703777778
      GO TO 21
   20 CONTINUE
      IEQ=3777777777778
      IVN=3777777777778
   21 CONTINUE
      CL=CM(8)
      IXSO=1
      IXMG=1
C ***************************************************************
C *       DETERMINE  WHICH FUNCTIONS AND VARIABLES ARE NEEDED   *
C ***************************************************************
```

```
C
      IF(CM(1).NE.0.0.OR .PP(1).NE.0.0)  GO TO 51
      IEQ=AND(IEQ,ICME1)
      IVN=AND(IVN,ICMV1)
      IXS0=0
   51 IF(CM(2).NE.0.0.OR .PP(2).NE.0.0)  GO TO 52
      IEQ=AND(IEQ,ICME2)
      IVN=AND(IVN,ICMV2)
   52 IF(CM(3).NE.0.) GO TO 53
      IEQ=AND(IEQ,ICME3)
      IVN=AND(IVN,ICMV3)
   53 IF(CM(4).NE.0.) GO TO 54
      IEQ=AND(IEQ,ICME4)
      IVN=AND(IVN,ICMV4)
   54 IF(CM(5).NE.0.) GO TO 55
      IEQ=AND(IEQ,ICME5)
      IVN=AND(IVN,ICMV5)
   55 IF(CM(6).NE.0.) GO TO 56
      IEQ=AND(IEQ,ICME6)
      IVN=AND(IVN,ICMV6)
      IXMG=0
   56 IF(CM(7).NE.0.) GO TO 57
      IEQ=AND(IEQ,ICME7)
      IVN=AND(IVN,ICMV7)
   57 IF(CM(8).NE.0.) GO TO 58
      CL=1.E-36
   58 CONTINUE
C
      DO 6 I=3,7
      IF(CM(I).NE.0.) CT(I)=LOG(CM(I))
    6 CONTINUE
      NF=1
      NV=1
      ICK=1
      DO 59 I=2,36
   45 ICE=AND(ICK,IEQ)
      IF(ICE.EQ.0) GO TO 60
      NF=NF+1
      IE(NF)=I
   60 ICV=AND(ICK,IVN)
      IF(ICV.EQ.0) GO TO 59
      NV=NV+1
      IV(NV)=I
```

```
   59 ICK=2*ICK
      IF(NF.EQ.NV) GO TO 61
      WRITE(6,200) NF,NV
      CALL EXIT
   61 IE(1)=1
      IV(1)=1
      NF=NF+1
      NV=NV+1
      IE(NF)=39
      IV(NV)=39
C
      IPMS=0
      IF(IXSO.EQ.0.OR.IXMG.EQ.0) GO TO 62
      IF(NS.NE.0) GO TO 62
      NF=NF+1
      NV=NV+1
      IE(NF)=38
      IV(NV)=38
      IPMS=1
   62 CONTINUE
C
C ***********************************************
C *     COMPUTE TEMPERATURE DEPENDENT VARIABLES   *
C ***********************************************
C
      CALL CTEMP(TK)
C
      WRITE(6,100)
      CALL DATIME
      TC=TK-273.16
      WRITE(6,109) TC
      WRITE(6,105)
      IF(L1.EQ.1) GO TO 8
      WRITE(6,111) CM(1)
      IF(CM(1).NE.0.) CT(1)=LOG(CM(1))
      GO TO 9
    8 WRITE(6,113) PP(1)
      X(10)=PP(1)*CPK(1)  +X(3)+X(4)+X(14)+X(21)+X(25)
      PK(1)=LOG(PP(1)*CPK(1))
    9 IF(L2.EQ.2) GO TO 10
      WRITE(6,112) CM(2)
      IF(CM(2).NE.0.) CT(2)=LOG(CM(2))
      GO TO
```

```
.0 WRITE(6,[illegible]4) PP(2
   X(11)=PP(2)*CPK(2) +X(6)+X(7)+X(15)+X(16)+X(19)+X(26)+X(28)+X(30)
  1+X(33)+X(34)
   PK(2)=LOG(PP(2)*CPK(2))
11 CONTINUE
   IF(L3.NE.4) WRITE(6,106) (CM(I),I=3,9)
   IF(CM(9).NE.0.) CT(9)=LOG(CM(9))
   IF(L3.NE.4) GO TO 30
   WRITE(6,108) (CM(I),I=3,8)
   PCSL=100.*CM(9)
   WRITE(6,116) PCSL,CM(10)
   SUM=0.
   DO 31 I=3,8
31 SUM=SUM+WTS(I)*CM(I)
   SUM=SUM+WTS(9)*.5*(CM(8)-CM(11))
   CT(9)=LOG(CM(9)*WTS(9))
   CT(10)=(1-CM(9))*CM(10)-CM(9)*SUM
30 CONTINUE
   IF(L4.EQ.8) CT(10)=CM(10)*LOG(10.)
C
   DO 1 I=1,NF
   NV=IV(I)
   IF(X(NV).EQ.0.) X(NV)=1.E-11
   XLN(I)=LOG(X(NV))
 1 CONTINUE
C
   CALL NOLIN(XLN,NF,EPS,EQS,PARTL)
C
   DO 2 I=1,NF
   NV=IV(I)
   X(NV)=EXP(XLN(I))
 2 CONTINUE
C
   IF(L1.EQ.1) CM(1)=X(10)
   IF(L2.EQ.2) CM(2)=X(11)
   IF(L1.EQ.1) X(10) =G(39)*PP(1)*CPK(1)/G(10)
   IF(L2.EQ.2) X(11) =G(39)*PP(2)*CPK(2)/G(11)
C
   IS=1
   WRITE(6,101)
   WRITE(6,102)
   A=G(39)
   WACT=A
```

```
      WRITE(6,107) A
      NFM1=NF-1
      IF(IPMS.EQ.1) NFM1=NFM1-1
      DO 3 I=1,NFM1
      NV=IV(I)
      PG=G(NV)
      A=X(NV)*PG
      IS=2*NV-1
      WRITE(6,103) LB(IS),LB(IS+1),X(NV),A,PG
      IF(NV.EQ.29.AND.IPMS.EQ.1) WRITE(6,103) LB(75),LB(76),X(38),X(38),
     *PG
    3 CONTINUE
      IF(CM(8).EQ.0.) GO TO 4
      PG=G(37)
      CL=CL/X(39)
      A=CL*PG
      IS=73
      WRITE(6,103) LB(IS),LB(IS+1),CL,A,PG
    4 CONTINUE
C
      IF(L1.EQ.1) GO TO 12
      IF(CM(1).EQ.0.) GO TO 13
      PP(1)=X(10)*G(10)/(CPK(1)*G(39))
      WRITE(6,113) PP(1)
      GO TO 13
   12 WRITE(6,111) CM(1)
   13 IF(L2.EQ.2) GO TO 14
      IF(CM(2).EQ.0.) GO TO 15
      PP(2)=X(11)*G(11)/(CPK(2)*G(39))
      WRITE(6,114) PP(2)
      GO TO 15
   14 WRITE(6,112) CM(2)
   15 CONTINUE
      WRITE(6,115) X(39)
      IF(L3.EQ.4) TMW=X(39)*TH(10)
      IF(L3.EQ.4) WRITE(6,117) TMW
C
      PH=-ALOG10(X(1)*G(1))
      WRITE(6,104) PH,FI,REN
      IF(L4.EQ.8) WRITE(6,118)
      AZ=FI
C
C
```

```
RETURN
END
```

```
@I FOR STKGAS
      SUBROUTINE STKGAS(P,S)
      RETURN 2
      END
      END
```

```
@I FOR TOEQSP
      SUBROUTINE TOEQSP(L)
      INCLUDE CMMN,LIST
C
      DEFINE X(I)=SV(L,I)
C
      P(1)=P(1)
    1 CONTINUE
      X(31)=X(53)+X(54)+X(60)+X(64)+X(71)
      X(32)=X(56)+X(57)+X(61)+X(65)+X(66)+X(72)+X(74)+X(77)+X(78)
      X(33)=X(55)+X(59)+X(67)+X(73)+X(79)
      X(34)=.5*(X(58)+X(69)+X(80))
      X(35)=X(62)+X(63)+X(64)+X(65)+X(66)+X(67)+X(68)
      X(36)=X(69)+X(70)+X(71)+X(72)+X(73)+X(74)
      X(37)=.5*(X(75)+X(76)+X(77)+X(78)+X(79)+X(80))
      X(38)=X(81)
      X(39)=55.50622+.5*(X(51)+X(52)+X(53)+X(56)+X(59)+2*X(60)+2*X(61)
     *+X(63)+X(66)+X(70)+X(72)+X(76)+X(78)-X(81))
C
      IF(IE.EQ.1) GO TO 2
C
      X(31)=X(31)*(X(50)/1000.)+(X(104)+X(105)+X(113)+X(114)+X(115))
      X(32)=X(32)*(X(50)/1000.)+(X(103)+X(110)+X(111)+X(112))
      X(33)=X(33)*(X(50)/1000.)+(X(106)+X(107)+X(116))
      X(34)=X(34)*(X(50)/1000.)
      X(35)=X(35)*(X(50)/1000.)+(X(102)+X(103)+X(104)+X(105)+X(106)
     *+X(107)+X(101))
      X(36)=X(36)*(X(50)/1000.)+(X(109)+X(110)+X(111)+X(112)+X(113)+X(11
     *4)+X(115)+X(116)+X(108))
      X(37)=X(37)*(X(50)/1000.)+(X(117)+.5*X(119))
      X(38)=X(38)*(X(50)/1000.)+X(119)
      X(39)=X(39)*(X(50)/1000.)+(X(102)+.5*X(105)+2*X(107)+X(109)+3*X(11
     *1)+5*X(112)+3*X(114)+6*X(115)-.5*X(119))
C
      RETURN
C
      ENTRY TOLISP(L)
C
      IE=1
      GO TO 1
    2 CONTINUE
      DO 3 I=31,39
```

```
      X(I)=X(I)*X(50)/1000.
    3 CONTINUE
      IE=0
      RETURN
      END
```

```
@I FOR TOSOSP
      SUBROUTINE TOSOSP(S,TKS)
      DIMENSION S(1),TKS(1)
C
      TKS(1)=S(4)+S(5)+S(13)+S(14)+S(15)
      TKS(2)=S(3)+S(10)+S(11)+S(12)
      TKS(3)=S(6)+S(7)+S(16)
      TKS(4)=0.
      TKS(5)=S(1)+S(2)+S(3)+S(4)+S(5)+S(6)+S(7)
      TKS(6)=S(8)+S(9)+S(10)+S(11)+S(12)+S(13)+S(14)+S(15)+S(16)
      TKS(7)=S(17)+.5*S(19)
      TKS(8)=S(19)
      TKS(9)=S(2)+.5*S(5)+2*S(7)+S(9)+3*S(11)+5*S(12)+3*S(14)+6*S(15)
     *-.5*S(19)
      RETURN
      END
```

```
@I FOR WAVAP
      SUBROUTINE WAVAP(L,ABN,P,T,WV)
      INCLUDE CMMN,LIST
C
      T=328.
      TN=SV(L,5)
      CALL DHGS(L,H1)
    2 SV(L,5)=T
      CALL DHGS(L,H2)
      CPM=(H1-H2)/(TN-T)
      DH=13630.6-10.464*T
      PEB=CPM*(TN-T)/DH
      DPEB=-CPM/DH+10.464*PEB/DH
      VPLOG=5.22684-1750.286/(T-38.2)
      DLOGVP=1750.286/ (T-38.2)**2
      YGS=(10.**VPLOG)/P
      DYGS=2.303*YGS*DLOGVP
      PMB=(SV(L,10)-SV(L,19)-ABN)*YGS/(1.-YGS)-SV(L,19)
      DPMB=(SV(L,10)-SV(L,19)-ABN)*DYGS/(1.-YGS)**2
      TO=T
      T=T-(PMB-PEB)/(DPMB-DPEB)
      IF(ABS(1.-PEB/PMB).GT..005) GO TO 2
      SV(L,5)=TN
      WV=CPM*(TN-T)/DH
      RETURN
      END
```

```
@I FOR WTRMKP
      SUBROUTINE WTRMKP(P,$)
      INCLUDE CMMN.LIST
      INCLUDE LCMN.LIST
      DIMENSION WM(8)
C
     DDEFINE         DL(T)      = +.99995282
     1                            +.46726616E-4  *  (T-273.16)
     2                            -.74105074E-5  *  (T-273.16)**2
     3                            +.41079583E-7  *  (T-273.16)**3
     4                            -.13370708E-9  *  (T-273.16)**4
C
      P(1)=P(1)
      LOCSV=ISEQ(NL)
      LOS=ISTM(LOCSV,6)
      KEY=SV(LOS,2)
      IF(KEY.NE.0) RETURN 2
      READ(5,100) WM
  100 FORMAT(8E10.3)
C
      DO 1 J=1,8
      I=52+J
      PRD(I)=WM(J)
    1 CONTINUE
C
      T=323.16
      DW=1000.*DL(T)
      SV(LOS,54)=WM(1)/(DW*XMW(54))
      SV(LOS,57)=WM(2)/(DW*XMW(57))
      SV(LOS,55)=WM(3)/(DW*XMW(55))
      SV(LOS,58)=WM(4)/(DW*XMW(58))
      SV(LOS,62)=WM(5)/(DW*XMW(62))
      SV(LOS,69)=WM(6)/(DW*XMW(69))
      SV(LOS,75)=WM(7)/(DW*XMW(75))
      SV(LOS,81)=WM(8)/(DW*XMW(81))
C
      RETURN 2
      END
```

```
@I FOR XSODIS
      SUBROUTINE XSODIS(N,XDS,XD)
      DIMENSION XD(1),XDS(1)
C
      XD( 1)=0.
      XD( 2)=XDS(1)
      XD( 3)=XDS(2)
      XD( 4)=1-N
      XD( 5)=1-N
      XD( 6)=1-N
      XD( 7)=1-N
      XD( 8)=0.
      XD( 9)=XDS(3)
      XD(10)=XDS(4)
      XD(11)=XDS(4)
      XD(12)=XDS(4)
      XD(13)=1-N
      XD(14)=1-N
      XD(15)=1-N
      XD(16)=XDS(5)
      XD(17)=XDS(6)
      XD(18)=0.
      XD(19)=XDS(7)
      RETURN
      END

@I FOR READIN
```