

A CSMP/360 Precompiler for Kinetic Chemical Equations

R. Lawrence Clark
Gabriel F. Groner

March 1973

P-4812-1

Any views expressed in this paper are those of the authors. They should not be interpreted as reflecting the views of The Rand Corporation or the official opinion or policy of any of its governmental or private research sponsors. Papers are reproduced by The Rand Corporation as a courtesy to members of its staff.

P-4812
2

A CSMP/360 PRECOMPILER FOR KINETIC CHEMICAL EQUATIONS*

R. Lawrence Clark
Gabriel F. Groner

The Rand Corporation, Santa Monica, California

INTRODUCTION

CHEMCSMP is designed to bring both a convenient problem-oriented language and the power of CSMP/360 [1] to those who solve problems in kinetic chemical equations, where the primary interest is in the transient states of chemical systems. Its language and computational techniques are based on those of BIOMOD [2], an interactive model construction and simulation system that was especially designed for convenience of use. Whereas BIOMOD requires a sophisticated graphics terminal and a large amount of computer storage, CHEMCSMP can be used at more conventional facilities.

Chemical equations are written in CHEMCSMP in a format resembling conventional chemical notation. They may be combined with other CSMP statements, thus utilizing the powerful functions and simplified output specifications of CSMP. CHEMCSMP is implemented via a precompiler, which translates the chemical equations into pure CSMP. An augmented function library is used to perform some of the internal computations. Thus, CSMP itself need not be altered to run CHEMCSMP.

One drawback of using numerical integration for solving systems of chemical and other dynamic equations is that some of the equations may react relatively slowly while

* This paper was prepared for publication in *Simulation*, the monthly journal of Simulation Councils, Inc.

This research was supported by the National Institutes of Health under Grant No. CA-12369.

P-48-2

3

others are simultaneously very fast. This results in a set of so-called "stiff" equations. In this situation, the time step that can be transversed in one iteration of the numerical integration scheme will be determined by the fastest reaction. This means that a large number of integration steps may have no significant effect on the slower equations, which may be the only ones of interest. Also, using numerous small integration steps is likely to lead to a large error buildup.

To overcome this difficulty, CHEMCSMP provides for two simultaneous types of chemical reactions, slow and fast. A slow reaction is dynamically solved via differential equations, and generally approaches equilibrium over a number of simulation time steps. A fast reaction is solved via matrix operations so that it is forced to equilibrium at every time step. For a slow reaction, the user provides two rate coefficients, indicating the rates at which the reaction proceeds in the forward and backward directions; these coefficients may be variable. For a fast reaction, a single equilibrium constant indicates the balance to be maintained between the two sides of the equation. The fast reactions must be in equilibrium initially. The user may either provide initial values that are in equilibrium and have the equilibrium constants computed, or he may provide equilibrium constants and estimates of the initial values and have the initial values computed.

A CHEMCSMP program may contain several distinct chemical compartments. Each compartment is chemically isolated from every other compartment. However, the user may explicitly indicate flow of material into or out of a compartment. A compartment description may be part of a CSMP macro definition and therefore may be used repeatedly.

USING CHEMCSMP

To specify a chemical compartment, the user first writes a statement consisting of the word COMPARTMENT; as in CSMP, only the first four letters need be coded. The COMPARTMENT card may appear either in the DYNAMIC segment or within the definition of a macro used only in the DYNAMIC segment. Next, the statements comprising the compartment are coded as described below. Finally, the compartment description is terminated by an ENDCOMPARTMENT (or ENDC) card.

Five different statement types may be used in describing a compartment. Each statement must begin with a letter indicating its type. Blanks may be used freely except within names and constants.

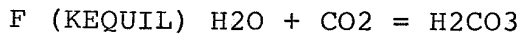
Slow reactions are written with an S followed by two rate coefficients, followed in turn by a chemical equation. The rate coefficients may be either CSMP variable names or real constants. They must be separated by a comma and enclosed in parentheses. The first rate coefficient indicates the rate at which the reaction proceeds from left to right (the forward rate). The second rate coefficient indicates the right-to-left or backward rate. The chemical equation is written as two expressions separated by an equals sign. Each expression is written as a sum of species names, optionally preceded by a coefficient of one or two digits. Species names follow the CSMP naming conventions, but are restricted to at most five characters. Thus, acceptable slow reaction statements are

S (KFWD,KBACK) 2 H2 + O2 = 2 H2O

and

S(3.2E-3,.00615)H2+CL2=2HCL

Fast reactions are written in the same way except that the letter F is used instead of the S and a single equilibrium constant replaces the rate coefficients. The equilibrium constant may be either a real constant or the name of a variable whose value is constant. An example of a fast reaction statement is



If a model contains fast reactions, the simulation-time routines check that these reactions are initially at equilibrium, i.e., the initial values of the species agree with the equilibrium constant. If not, the initial values are adjusted to force agreement with the equilibrium constant. However, if the value of an equilibrium constant is zero, CHEMCSMP assumes that the reaction is initially in equilibrium, and so the equilibrium constant is computed as the ratio of products of initial concentrations.

Gain terms are used to indicate changes in the amount of a species resulting from other than chemical interaction. This situation may arise, for example, as a result of flows into or out of the compartment, or from changes in the chemical phase of a species. ~~They are represented by the~~ letter G, followed by the parenthesized name of the species to which the gain applies, followed in turn by a CSMP assignment statement. The value of the expression on the right-hand side of the assignment statement is added to the derivative of the species. Losses may be represented as negative gains. A legal gain statement is shown below.



~~Non-reacting species may be used to affect the total~~ mass of a compartment without causing the species to enter into any reaction. These species are indicated by the letter N, followed by the name of the species. For example, the

statement

N CO2

indicates that CO2 is a non-reacting species in the compartment currently being described. Because a non-reacting species contributes to compartment mass, it affects the concentrations of the species in that compartment.

Arbitrary assignments provide a mechanism for entering CSMP assignment statements within a compartment. This enables the user to provide the definition of a variable, e.g., a rate coefficient, within the compartment in which it is used. To write an arbitrary assignment, the user merely codes the letter A, followed by a normal CSMP assignment statement. An example statement is shown below.

A KFWD = 1.4*SIN(GAMMA)/DELTA

In addition to the basic equations describing a compartment, it is necessary to specify the initial amounts of the species. By convention, the name of the initial amount of a species is the name of the species, followed by the digit 0. Thus, CO20 represents the initial amount of CO2. Therefore, care should be used in naming variables to avoid conflicting usages. An initial species name is used for every species appearing in a statement of type S, F or N. The values may be assigned to these variables via PARAMETER cards. Those whose values are not assigned will be assumed to be zero by CSMP.

Two special functions are provided for use with chemical species. The function CONC(x) computes the concentration of species x, i.e., the ratio of the number of moles of x to the total number of moles in the compartment. The function DERV(x) computes the derivative of the species x. This could be calculated by using the CSMP function DERIV(ic,x);

P-4812
7

however, DERV uses information already calculated for use by the chemistry solution routines and is therefore more efficient and more accurate. If these functions are applied to variables other than chemical species, an error message results.

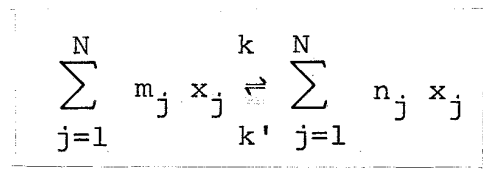
All amounts of species are assumed to be in units of moles. Thus, the concentrations are in units of mole-fractions (i.e., moles per mole), and the rate coefficients should be specified in units of 1/TIME.

COMPUTATIONAL TECHNIQUES

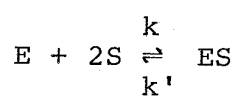
Dynamic Computations

The derivative of a chemical species is the sum of three terms: its gain term, the sum of contributions from slow reactions in its compartment, and the sum of contributions from fast reactions in its compartment. A gain term is simply the value of the expression to the right of an equals sign in a type G statement. The slow and fast contributions are computed from the chemical reactions [3].

Consider the reaction



where k and k' are the forward and backward rate coefficients and where there are N participating species such that m₁ molecules of x₁ react with m₂ of x₂, etc. to form n₁ molecules of x₁, n₂ of x₂, etc. Thus, for the reaction



if x₁ represents the species E, x₂ represents S, and x₃ represents ES, we would have m₁=1, m₂=2, m₃=0, n₁=0, n₂=0, and n₃=1. If this is a slow reaction, its contribution to the derivative of x_j is given by

$$dx_j/dt = \left(\sum_{i=1}^N x_i \right) \left(k \prod_{i=1}^N [x_i]^{m_i} - k' \prod_{i=1}^N [x_i]^{n_i} \right) \left(n_j - m_j \right)$$

P-4812
9

where $[x_i]$ represents the mole-fraction concentration of x_i ,
i.e.,

$$[x_j] = x_j / \sum_{i=1}^N x_i$$

Using this formula, the above reaction results in

$$dE/dt = \text{sum} \cdot (k \cdot [E] \cdot [S]^2 - k' \cdot [ES]) \cdot (-1)$$

$$dS/dt = \text{sum} \cdot (k \cdot [E] \cdot [S]^2 - k' \cdot [ES]) \cdot (-2)$$

and

$$dES/dt = \text{sum} \cdot (k \cdot [E] \cdot [S]^2 - k' \cdot [ES]) \cdot (1)$$

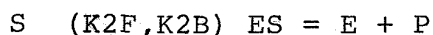
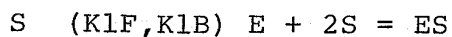
where

$$\text{sum} = E + S + ES$$

and, e.g.,

$$[E] = E/\text{sum}$$

If there is more than one reaction in a compartment, a similar formula applies, where each reaction has its individual k , k' , m_i , and n_i . However, N is the number of species in the entire compartment. Thus, if the only reactions in a compartment are specified by



the resulting derivatives are

$$dE/dt = \text{sum} \cdot (K1F \cdot [E] \cdot [S]^2 - K1B \cdot [ES]) \cdot (-1) \\ + \text{sum} \cdot (K2F \cdot [ES] - K2B \cdot [E] \cdot [P]) \cdot (1)$$

$$dS/dt = \text{sum} \cdot (K1F \cdot [E] \cdot [S]^2 - K1B \cdot [ES]) \cdot (-2)$$

$$dES/dt = \text{sum} \cdot (K1F \cdot [E] \cdot [S]^2 - K1B \cdot [ES]) \cdot (1) \\ + \text{sum} \cdot (K2F \cdot [ES] - K2B \cdot [E] \cdot [P]) \cdot (-1)$$

$$dP/dt = \text{sum} \cdot (K2F \cdot [ES] - K2B \cdot [E] \cdot [P]) \cdot (1)$$

P-4812
10

where

$$\text{sum} = E + S + ES + P$$

and where, e.g.,

$$[E] = E/\text{sum}$$

The fast reactions result in a similar set of equations except that these reactions are always in equilibrium. Gains and slow reactions change the amounts of the species, and so the fast reactions must react just enough to remain in equilibrium. Computationally, this requires computing the gain and slow-reaction terms, solving a set of linear algebraic equations (one for each fast reaction), then using the solutions to compute the fast reactions' contributions to the derivatives. The technique assumes that the fast reactions are initially in equilibrium. The computations are performed one compartment at a time.

Initial Computation

The fast reactions are initially forced to equilibrium by computing initial values that minimize the total thermodynamic free energy of the fast-reaction system while conserving mass. The computation [4] proceeds in three phases. The first phase employs linear programming to obtain a feasible estimate of the solution. The second uses a steepest descent method to minimize the free energy function. The final phase, which improves the accuracy of the second, is based on Newton's method.

If a model does not contain fast reactions or if the user does not wish to have the fast reactions initialized, the storage requirements of the execution phase can be appreciably reduced. This is accomplished by substituting a dummy subroutine for the one that performs the initialization

P. 4812

computations. The user simply inserts the following cards after the STOP statement of the model:

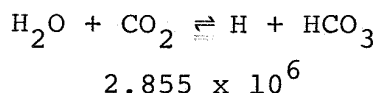
```
SUBROUTINE ZZCHEQ  
RETURN  
END
```

P-4812
12

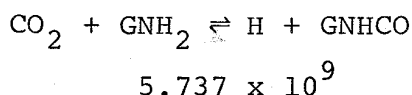
ILLUSTRATIVE EXAMPLE

To illustrate the use of CHEMCSMP, consider a hypothetical two-phase system, comprising a gas phase and a liquid phase. The gas phase consists of two substances, CO₂ and H₂O. The liquid phase consists of eight substances or species; viz., H₂O, CO₂, H, OH, HCO₃, GNH₃, GNH₂, and GNHCO. (The letter G denotes glycyl-glycine.) The system has two slow reactions, both of which are in the liquid phase:

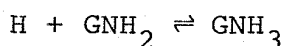
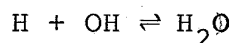
.02



88868.8



The system also has two fast reactions in the liquid phase:



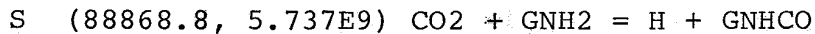
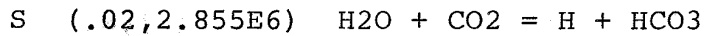
The equilibrium constants for the fast reactions are:

$$\frac{[\text{H}][\text{OH}]}{[\text{H}_2\text{O}]} = 2.184 \times 10^{-18}$$

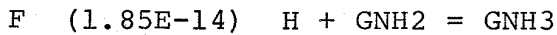
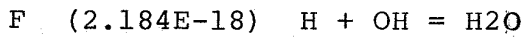
$$\frac{[\text{H}][\text{GNH}_2]}{[\text{GNH}_3]} = 1.85 \times 10^{-14}$$

where [x] denotes the mole-fraction concentration of x. We assume that CO₂ changes from a liquid to a gas at a rate proportional to the difference in its concentrations in the two phases.

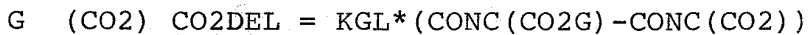
To transform this system description into a CHEMCSMP program, we create two compartments, one for the liquid phase and one for the gas phase. Each compartment is delimited by a COMPARTMENT card and an ENDCOMPARTMENT card; COMP and ENDC would work as well. In the liquid compartment, we rewrite the slow equations as



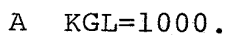
and the fast reactions as



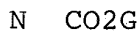
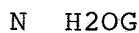
To indicate that the mass-transfer rate of CO_2 between phases is proportional to its concentration difference, we write



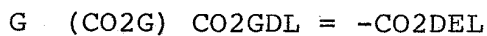
The name CO_2DEL is arbitrary; CO_2G is the name of CO_2 in the gas phase. The proportionality coefficient, KGL , is defined by



Because the gas phase comprises H_2O and CO_2 , neither of which participates in a chemical reaction in that phase, we define them as non-reacting species in the gas compartment by writing



However, CO_2G leaves the gas compartment at the same rate that it enters the liquid compartment. We indicate this by writing



Note that (1) each chemical species must be named explicitly as either a participant in a chemical reaction or as a non-reacting species; (2) unique species names must be provided for each compartment; (3) all mass gains and losses must be explicitly stated.

Outside the compartments we compute the number of moles in the liquid compartment with the CSMP statement

```
LIQUID = H2O/CONC(H2O)
```

because the concentration of H_2O is equal to H_2O divided by the number of moles in its compartment. We compute the number of moles in the gas compartment with

```
GAS=H2OG+CO2G
```

since only these two species appear in the compartment. These statements appear outside the compartments to emphasize that chemical compartments may be embedded in an otherwise standard CSMP program.

After adding a PARAMETER statement to specify the initial amounts of the species (e.g., H2OG0, CO2G0, H0, etc), and entering translation and simulation control statements, the complete CHEMCSMP program appears as shown in Fig. 1.

P-4912
15

CSMP CHEMICAL PRECOMPILER
INPUT LISTING

STATNO	STATEMENT	
1	TITLE CHEMCSMP EXAMPLE	8.
2	INITIAL	9.
3	PARAMETER H2O0=.222172,GNH20=6.E-5,CO20=0.,HO=1.521E-11, ...	10.
4	HCO30=0.,GNHCO0=0.,OH0=7.088E-9,GNH30=6.E-5, ...	11.
5	CO2G0=4.3616E-5, H2OG0=3.6658	12.
6	DYNAMIC	13.
7	COMPARTMENT	14.
8	S (.02,2.855E6) H2O + CO2 = H + HCO3	15.
9	S (88868.8, 5.737E9) CO2 + GNH2 = H + GNHCO	16.
10	F (2.18E-18) H + OH = H2O	17.
11	F (1.85E-14) H + GNH2 = GNH3	18.
12	G (CO2) CO2DEL=KGL*(CONC(CO2G)-CONC(CO2))	19.
13	A KGL=1000.	20.
14	ENDCOMPARTMENT	21.
15	COMPARTMENT	22.
16	N H2OG	23.
17	N CO2G	24.
18	G (CO2G) CO2GDL=-CO2DEL	25.
19	ENDCOMPARTMENT	26.
20	LIQUID=H2O/CONC(H2O)	27.
21	GAS=H2OG+CO2G	28.
22	TERMINAL	29.
23	METHOD ADAMS	30.
24	TIMER DELT=1.E-12,FINTIM=1.E-10	31.
25	PRINT CO2,H2O,H,HCO3,GNH2,GNHCO,GNH3,OH,CO2G,H2OG,LIQUID,GAS	32.
26	END	33.
27	STOP	34.
28	ENDJOB	35.

Fig. 1--User's Program for the Illustrative Example

P-4512
16

THE PRECOMPILER

CHEMCSMP is designed to appear to the user as a more powerful version of CSMP. In reality, it operates as two job steps. The first, the precompiler, reads the user's input statements and writes an intermediate data set consisting only of CSMP statements. The second job step, standard CSMP, treats the intermediate data set as if it were user input. Job Control Language to accomplish this task can be provided in a Catalogued Procedure, making communication with CHEMCSMP as easy as with CSMP.

The precompiler, a Fortran-coded program, examines each input statement to determine whether it is a control card of interest. When a COMPARTMENT card is encountered, it begins translating subsequent input until the matching ENDCOMPARTMENT card appears.

The CSMP code resulting from a compartment consists of two parts. The first is a PROCEDURE that sets up tables used by the solution routines. This PROCEDURE is executed only on the first iteration of the DYNAMIC segment (when TIME equals zero). The second part of the code consists of assignment and integral statements used to calculate the values of the species. The integrals utilize function DERV to calculate the derivatives of the species based on information supplied by the PROCEDURE section. In addition, each model contains initialization calls in the INITIAL section, and calls to update the derivatives of the species at the end of the DYNAMIC section.

Fig. 2 illustrates the precompiler output for the program shown in Fig. 1. ZZFLAG is initially set to 0. to indicate that chemical equilibrium calculations must be performed before computing derivatives of chemical species,

```
*****CONTINUOUS SYSTEM MODELING PROGRAM*****  
***PROBLEM INPUT STATEMENTS***  
  
TITLE CHEMCSMP EXAMPLE  
INITIAL  
PARAMETER H2O0=.222172,GNH20=6.E-5,CO20=0.,HO=1.521E-11, ...  
HCO30=0.,GNHCO0=0.,OHO=7.088E-9,GNH30=6.E-5, ...  
CO2G0=4.3616E-5, H2OG0=3.6658  
NOSORT  
ZZFLAG=0.  
CALL ZZCHMO  
DYNAMIC  
CO2DEL=KGL*(CONC(CO2G)-CONC(CO2))  
KGL=1000.  
PROCEDURE ZZDM01=ZZFF01(TIME)  
IF (ZZFLAG.NE.0.) GO TO 92301  
CALL ZZCHM1  
CALL ZZCHM2(0,.02,2.855E6)  
CALL ZZCHM3(1,H2O,H2O0,0.)  
CALL ZZCHM3(1,CO2,CO20,CO2DEL)  
CALL ZZCHM3(-1,H,HO,0.)  
CALL ZZCHM3(-1,HCO3,HCO30,0.)  
CALL ZZCHM2(0,88868.8,5.737E9)  
CALL ZZCHM3(1,CO2,CO20,CO2DEL)  
CALL ZZCHM3(1,GNH2,GNH20,0.)  
CALL ZZCHM3(-1,H,HO,0.)  
CALL ZZCHM3(-1,GNHCO,GNHCO0,0.)  
CALL ZZCHM2(1,2.18E-18,0.)  
CALL ZZCHM3(1,H,HO,0.)  
CALL ZZCHM3(1,OH,OH0,0.)  
CALL ZZCHM3(-1,H2O,H2O0,0.)  
CALL ZZCHM2(1,1.85E-14,0.)  
CALL ZZCHM3(1,H,HO,0.)  
CALL ZZCHM3(1,GNH2,GNH20,0.)  
CALL ZZCHM3(-1,GNH3,GNH30,0.)  
CALL ZZCHM4  
92301 ZZDM01=0.  
ENDPROC
```

Fig. 2--Precompiler Output for
the Illustrative Example

```
CO2 = INTGRL(CO2 ,DERV(CO2 )+ZZDM01)
H2O = INTGRL(H2O ,DERV(H2O )+ZZDM01)
H = INTGRL(HO ,DERV(H )+ZZDM01)
HCO3 = INTGRL(HCO3O ,DERV(HCO3 )+ZZDM01)
GNH2 = INTGRL(GNH2O ,DERV(GNH2 )+ZZDM01)
GNHCO = INTGRL(GNHCOO,DERV(GNHCO)+ZZDM01)
OH = INTGRL(OH ,DERV(OH )+ZZDM01)
GNH3 = INTGRL(GNH3O ,DERV(GNH3 )+ZZDM01)
CO2GDL=-CO2DEL
PROCEDURE ZZDM02=ZZFFO2(TIME)
IF (ZZFLAG.NL.O.) GO TO 92302
CALL ZZCHM1
CALL ZZCHM2(O,O. ,O. ,O. )
CALL ZZCHM3( O ,H2OG ,H2OGO ,O. )
CALL ZZCHM2(O,O. ,O. ,O. )
CALL ZZCHM3( O ,CO2G ,CO2GO ,CO2GDL)
CALL ZZCHM4
92302 ZZDM02=C.
ENDPRO
CO2G = INTGRL(CO2GO ,DERV(CO2G )+ZZDM02)
H2OG = INTGRL(H2OGO ,DERV(H2OG )+ZZDM02)
LIQUID=H2OG/CONC(H2O)
GAS=H2OG+CO2G
NJSORT
IF (ZZFLAG.EQ.O.) CALL ZZCHEQ
ZZFLAG=1.
CALL ZZDAUZ
TERMINAL
METHOD ADAMS
TIMER DELT=1.E-12,FINTIM=1.E-10
PRINT CO2,H2O,H,HCO3,GNH2,GNHCO,GNH3,OH,CO2G,H2OG,LIQUID,GAS
END
STOP
```

27.
28.
29.
30.
31.
32.
33.
34.

OUTPUT VARIABLE SEQUENCE
ZZFLAG ZZ0001 ZZDM01 ZZ0003 CO2 ZZ0005 H2O ZZ0007 H ZZ0009
HCO3 ZZ0011 GNH2 ZZ0013 GNHCO ZZ0015 OH ZZ0017 GNH3 ZZDM02
ZZ0019 CO2G ZZ0021 H2OG KGL CO2DEL CO2GDL LIQUID GAS ZZ0022
ZZFLAG

OUTPUTS INPUTS PARAMS INTEGERS + MEM BLKS FORTRAN DATA CDS
35(500) 69(1400) 12(400) 10+ 0= 10(300) 61(600) 8

ENDJOB 35.

Fig. 2 (cont'd)

P-4812
19

and that the data structure that stores the description of the chemical system must be initialized. ZZCHM0 sets the chemical data structure to its null state. ZZCHM1 initializes a compartment. ZZCHM2 stores the slow/fast designation and rate coefficients for a reaction. ZZCHM3 stores a species coefficient and the addresses of the species value, the initial value, and the gain term, if any. ZZCHM4 computes the concentrations of all species in a compartment. ZZCHEQ computes the equilibrium state for chemical reactions during the first iteration. ZZDAUZ computes the derivatives of the chemical species during each iteration.

P-4812
20

RESTRICTIONS

In order to facilitate precompiler efficiency, several minor restrictions have been placed on the CHEMCSMP program structure.

1. Every CHEMCSMP program must contain an INITIAL statement, a DYNAMIC statement, and a TERMINAL statement. The precompiler inserts additional structure statements in the vicinity of these cards. If one or more of these sections would not otherwise be used, the user need only insert the required card in the standard position.
2. Those control cards that the precompiler examines must begin in column 1. These consist of the statements INITIAL, DYNAMIC, TERMINAL, MACRO, ENDMACRO, COMPARTMENT, ENDCOMPARTMENT, and ENDJOB.
3. Non-control statements whose first four characters are the same as those of the control cards listed above must not begin in column 1. In general, we recommend that all control statements begin in column 1 and that all non-control statements begin in some other column.
4. A compartment may not contain more than 25 reactions (S, F, or N statements), nor may it refer to more than 50 distinct species.
5. An entire model may contain at most 25 compartments, 200 reactions, 250 distinct species, and 800 references to the species.
6. Each statement within a compartment must be contained on a single card. No continuation mechanism is provided.

7. Blank cards and comments cards (asterisk in column 1) may not appear within a compartment.
8. At most, 100 diagnostics are produced by the pre-compiler. If a greater number of errors is detected, only the first 100 are printed.

1-4812
22

PRINTED OUTPUT

The CHEMCSMP precompiler lists the user's input statements, and, if precompilation is successful, the CSMP translator in turn prints its input (i.e., the precompiler output). Thus, the user receives listings of both the input and output of the precompiler.

If the precompiler detects errors, diagnostics are printed after the input listing, and CSMP processing is bypassed. Each diagnostic consists of a statement number corresponding to those appearing in the input listing and a message. The following messages may be produced:

MISPLACED CONTROL CARD

The precompiler expects to encounter control cards in the order INITIAL, DYNAMIC, TERMINAL, ENDJOB. Macro definitions (MACRO ... ENDMACRO) must appear before the INITIAL card. Compartments (COMPARTMENT ... ENDCOMPARTMENT) must appear either in the DYNAMIC section or within a macro definition.

INVALID STATEMENT TYPE

The first character of a statement within a compartment must be one of the letters S, F, G, N, or A.

MODEL CONTAINS NO COMPARTMENTS

CHEMCSMP should not be used for models that do not require its special features.

COMPARTMENT CONTAINS NO REACTIONS

Every compartment must contain at least one statement of type S, F, or N.

MORE THAN 50 SPECIES IN COMPARTMENT

A compartment may not refer to more than 50 species.

MORE THAN 25 REACTIONS IN COMPARTMENT

A compartment may not contain more than 25 statements of types S, F, and N.

INVALID SYNTAX

A statement within a compartment must conform to the syntax specifications described under "Using CHEMCSMP." This message indicates that the statement does begin with a valid statement type code, but that it is otherwise invalid. Possible elusive causes include species names of more than 5 characters, constants of more than 12 characters, species coefficients of more than 2 digits, or blankseembedded within a name or constant.

Standard CSMP output statements may be used to print, print-plot, or range selected variables. These options are described in the CSMP User's Manual [5].

In addition to the foregoing messages, which are produced by the precompiler, there are a number of messages that may arise during simulation. The following indicate incorrect formulation of the problem or poor choice of integration technique.

ERROR: CHEMICAL VALUE < 0

The dynamic computations have caused the amount of some chemical species to become negative. This condition may result from using integration step sizes that are too large.

ERROR: CAN'T FIND VALUE

The CONC or DERV function has been applied to a variable that is not a chemical species.

ERROR: EMPTY COMPARTMENT

The sum of the amounts of the species in a compartment is zero.

ERROR: SINGULAR MATRIX

The fast reactions cannot be solved because the associated matrix has become singular.

The next set of messages indicates errors in the calculation of the equilibrium state for fast reactions.

JM nn IS 0

THIS PROBLEM IS INFEASIBLE. THE FOLLOWING LINEAR
COMBINATION OF ROWS

...
LEADS TO THE FOLLOWING INFEASIBLE EQUATION
...

SIMPLEX METHOD HAS FAILED DUE TO EXCESSIVE ROUND-OFF
ERROR

ITERATION LIMIT EXCEEDED

MATRIX HAS NULLITY ON ROW nn

MATRIX HAS NULLITY ON ROW FOR COMPARTMENT nn

COMPARTMENT nn TOO SMALL

Each of these messages indicates that the model, although syntactically valid, cannot be solved by the numerical methods of CHEMCSMP. The system follows these messages by the message

SIMULATION ENDED DUE TO ERRORS
and terminates the run.

The following messages should never appear. They indicate that system limitations on the size of the model have been exceeded, but that the violation has not been detected by the precompiler. If any of these does appear, it reflects an inconsistency in the code.

ERROR: MANY COMPARTMENTS

ERROR: TOO MANY REACTIONS

ERROR: TOO MANY REACTANTS

ERROR: TOO MANY CHEMICALS

P-4/6/12
24

RESOURCE REQUIREMENTS

CHEMCSMP will operate at any installation that can use CSMP/360, provided that sufficient core storage is available. As with CSMP, the exact amount of core required to run a simulation depends on the problem. The example in Figs. 1 and 2 requires approximately 158,000 (decimal) bytes. If automatic adjustment of initial conditions for fast reactions is not desired, this requirement is reduced to about 110,000 bytes. The precompiler utilizes only 37,000 bytes.

Execution times are also problem-dependent. The previous example was precompiled in 1.79 processor seconds on Rand's 360/65. Its translation and execution by CSMP required 188.62 processor seconds.

AVAILABILITY

The Rand Corporation will distribute CHEMCSMP to any installation that desires it. A nominal charge* will be made for copying and shipping the distribution tape.

The necessary routines for CHEMCSMP are distributed on a 9-track, 800 bpi tape with standard label. The tape is an OS/360 Unloaded Partitioned Data Set (PDS) named C5075.NIH.CHEMLIB. The PDS contains the precompiler (member name PRECOMP) and the subroutines that are accessed by the automatic call facility during CSMP's linkage-editing phase (member names ZZCHM0, ZZCHM1, ZZCHM2, ZZCHM3, ZZCHM4, ZZCHEQ, ZZDAUZ, CONC and DERV).

The unloaded PDS should be loaded and catalogued on a direct-access volume by OS Utility IEHMOVE [6]. A good space parameter for the disk is (3072,(80,40,4)). (In the following JCL, we shall assume that the data set is renamed CHEMLIB.)

CHEMCSMP may then be executed by providing the following Job Control Language statements.

*These charges currently average about \$15 if the user supplies a tape. An additional \$25 is charged if Rand provides the tape. The preferable way to order CHEMCSMP is by sending a purchase order for an amount "not to exceed \$50" to:

The Rand Corporation
RCC Production Control
1700 Main Street
Santa Monica, Calif. 90406

```
// jobcard
//STEPA EXEC PGM=PRECOMP
//STEPLIB DD DSN=CHEMLIB,DISP=SHR
//FT06F001 DD SYSOUT=A
//FT01F001 DD UNIT=SYSDA,SPACE=(80,(200,100,1)),
//      DISP=(NEW,PASS),DCB=(RECFM=F,BLKSIZE=80)
//FT05F001 DD *
.
. (Insert CHEMCSMP program here)
.
/*
//STEPB EXEC CSMP,COND=(5,LT,STEPA)
//CSMP.SYSLIB DD DSN=CSMP.LOADM,DISP=SHR
//      DD DSN=SYS1.FORTLIB,DISP=SHR
//      DD DSN=CHEMLIB,DISP=SHR
//CSMP.FT01F001 DD DSN=*.STEPA.FT01F001,DISP=(OLD,DELETE)
```

The above JCL assumes that the Catalogued Procedure for CSMP has been installed exactly as specified in the CSMP Operator's Manual [7]. Any deviation from those specifications must be reflected in the JCL.

A more convenient way of using CHEMCSMP would be through use of its own Catalogued Procedure. Such a procedure would have the same form as the above Job Control Language statements, normally replacing the FT01F001 DD statement in STEPA by a forward reference such as

```
//FT01F001 DD DDNAME=SYSIN
```

Then, if the procedure were named CHEMCSMP, the user would simply provide the following statements.

```
// jobcard
//      EXEC CHEMCSMP
//SYSIN DD *
.
. (Insert CHEMCSMP program here)
.
/*
```

P-4812
27

LAST PAGE

REFERENCES

1. *System/360 Continuous System Modeling Program (360A-CX-16X) Application Description*, International Business Machines Corporation, Form No. C28-6586, 1969.
2. Groner, G. F., R. L. Clark, R. A. Berman, and E. C. DeLand, "BIOMOD: An Interactive Computer Graphics System for Modeling," *AFIPS Conference Proceedings (1971 FJCC)*, Vol. 39, AFIPS Press, Montvale, New Jersey, 1971, pp. 369-378. (Also, The Rand Corporation, R-6117-NIH, July 1971).
3. Clasen, R. J., *The Numerical Integration of Kinetic Equations for Chemical Systems Having Both Slow and Fast Reactions*, The Rand Corporation, P-3547, September 1967.
4. -----, *The Numerical Solution of the Chemical Equilibrium Problem*, The Rand Corporation, RM-4345-PR, January 1965.
5. *System/360 Continuous System Modeling Program User's Manual*, International Business Machines Corporation, Form No. H20-0367, 1969.
6. *IBM System/360 Operating System Utilities*, International Business Machines Corporation, Form No. C28-6586, 1969.
7. *System/360 Continuous System Modeling Program Operator's Manual*, International Business Machines Corporation, Form No. H20-0368, 1969.

Program 114"
for ...
one ...
with ...