418

A COMBINED MODULAR AND SIMULTANEOUS LINEAR EQUATION EXECUTIVE SYSTEM FOR PROCESS SIMULATION

A COMBINED MODULAR AND SIMULTANEOUS LINEAR EQUATION EXECUTIVE SYSTEM FOR PROCESS SIMULATION

BY

JOSEPH PAUL GEORGES HEBERT LISLOIS, B. ENG.

A Project Report

Submitted to the School of Graduate Studies

in Partial Fulfilment of the Requirements

for the Degree

Master of Engineering

McMaster University

December, 1976

MASTER OF ENGINEERING (1976) McMaster University
(Chemical Engineering) Hamilton, Ontario

TITLE: A Combined Modular and Simultaneous Linear Equation Executive System for Process Simulation

AUTHOR: Joseph Paul Georges Hebert Lislois, B. Eng. (Royal Military College of Canada)

SUPERVISOR: Professor T.W. Hoffman

NUMBER OF PAGES: xii, 339

ABSTRACT:

A new computer executive system for the steady state simulation of chemical processes has been developed which combines the modular (GEMCS) approach with the simultaneous linear equation (SYMBØL) approach to simulation. In the combined system, a GEMCS simulation, using non-linear models, is used to generate the coefficients for the set of linear equations describing the process. This linear system of equations may also include the constraints on the process which dictate the operating conditions for the actual process. The solution of the linear equations then provide new operating conditions (feed flowrates together with the component flowrates in the recycle streams) for the modular simulation, which in turn provides new coefficients; etc. This iterative procedure is automatically continued until the system is converged to the desired point.

A modular simulation for an actual Naphtha Reforming Plant has also been achieved and it was used as a test case to demonstrate the use and effectiveness of this new executive system. In the course

of developing this simulation, the application of a method for correcting plant data was demonstrated. This is the first real application of this method to be reported in the current literature.

ACKNOWLEDGEMENTS

The study reported herein was supervised by Dr. T.W. Hoffman, professor of Chemical Engineering at McMaster University. The author is specially grateful to him for his assistance and encouragement throughout the progress of this study.

Considerable help was obtained from computer personnel; particularly the discussions with and help from Mr. P. Dunmore and Mrs. M. Bayley were most useful.

The Reformer Plant used as a test case in this simulation project is a real process; however, the location of it has been purposely witheld to protect some proprietary aspects of it. At the same time, the availability of an actual process and plant data has made this project a realistic one and is very much appreciated by the author.

The author wishes to acknowledge the financial assistance obtained through McMaster University.

Special thanks are also due to Messers. Jules Thibault, Luis Mirammontes-Vidal and Luis Rubio Garcia, fellow classmates of the 1975 simulation course for their help in building the Naphtha Reformer Plant Simulation. The careful typing of the thesis by Mrs. Hazel Coxall is appreciated.

TABLE OF CONTENTS

		Page
I.	INTRODUCTION	
1.1	General Definitions	2
1.2	Motivation for This Simulation Project	4
	PART I	
	THE DEVELOPMENT OF THE COMBINED GEMCS-SYMBØL ROUTINE	
2.	A REVIEW OF GEMCS AND SYMBØL	8
2.1	The Modular GEMCS Simulation	8
2.2	The Global or Simultaneous Equation Approach to Process Simulation	9
3.	DEVELOPMENT OF THE COMBINED GEMCS-SYMBØL EXECUTIVE	
3.1	General	18
3.2	GEMCS Data Set - Stream Equivalence Information	19
3.3	Data Set for SYMBØL	20
3.4	The Calculation of the Unit Coefficients in SYMBØL	21
3.5	Description of the Subroutines Written for the Combined Executive	24
3.6	Hardware Problem - Lack of CORE Storage	61
4.	USING THE COMBINED GEMCS-SYMBØL SYSTEM	
4.1	Introduction	74
4.2	SYMBØL Information Flow Diagram	74
4.3	Equivalent Streams	75
4.4	Changes in the GEMCS Data Set	77
4.5	Preparing the SYMBØL Data Set	79
4.6	Additional Remarks	80

5. TESTING THE COMBINED GEMCS-SYMBØL EXECUTIVE

Page

5.1 Introduction 84

5.2 SYMBØL Information Flow Diagram 84

5.3 Equivalent Stream and Stream Codes 89

5.4 Data Set for GEMCS 93

5.5 SYMBØL Data Set 94

5.6 Comparison Between the Combined Executive with the Self Standing GEMCS Simulation 101

PART II

THE STEADY STATE SIMULATION OF A NAPHTHA REFORMER PLANT USING THE GEMCS MODULAR SIMULATION ALONE

6. THE STEADY-SIMULATION OF A NAPHTHA REFORMER PLANT

6.1 Introduction 104

6.2 General 105

6.3 Process Description 107

6.4 Questions to be Answered by the Simulation 112

6.5 Stream Information Vector 13

7. THERMODYNAMICS AND PHYSICAL PROPERTY DATA

7.1 Equilibrium Constants and Non-Ideality 116

7.2 Heat Capacity 120

7.3 Density 120

7.4 Heat of Vaporization of Water 120

7.5 Vapour Pressure of Water 121

7.6 Heat of Vaporization of Naphtha 121

7.7 Vapour Pressure of Naphtha 122

7.8 Heat of Reaction of Naphtha 122

Page

7.9 Heat of Combustion of Fuel 124

8. PROCESS INFORMATION FLOW DIAGRAM AND ASSOCIATE MODULES

8.1 Process Information Flow Diagram 125

8.2 The Naphtha Reformer Reactor (REACO1) 125

8.3 The Water Gas Shift Reactor (REACO2) 138

8.4 The Methanator Reactor (REACO3) 139

8.5 Heat Exchanger Models 141

8.6 Miscellaneous Models 157

8.7 Controller Models 160

9. ESTIMATING PARAMETERS IN MODELS FROM PLANT BASE CASE DATA

9.1 Background 163

9.2 Plant Data and Observations 163

9.3 General Background-Parameters Estimation in Models 168

9.4 Correction of Measured Plant Data According to Ripps 171

9.5 Adjustment of Temperature Measurements in the Flue Gas Heat Exchanger Train 178

9.6 Parameter Estimation in Reactor Models 188

9.7 Parameter Estimation in the Heat Exchanger Models 194

9.8 Parameter Estimation for Pressure Drop Calculation in Each Model 197

10. CONCLUSION AND RECOMMENDATIONS 200

REFERENCES 202

APPENDIX A. PRØGRAM LISTINGS OF THE COMBINED GEMCS-SYMBØL EXECUTIVE SYSTEM AND NOTES 204

APPENDIX B. LISTING OF SUBROUTINE EQK 243

APPENDIX C. PHYSICAL PROPERTY PACKAGE 245

APPENDIX D. NAPHTHA FEEDSTOCK COMPOSITION 253

Page

APPENDIX E. FLOWCHARTS AND LISTINGS OF THE MODELS OF THE NAPHTHA REFORMER PLANT SIMULATION 254

APPENDIX F. GEMCS DATA SET AND CONVERGED SOLUTION OF THE COMBINED EXECUTIVE 313

INDEX OF FIGURES

		Page
3.1	Schematic Picture of a Program Divided in Overlays	63
3.2	Example of a Source Deck Program divided in Overlays	66
3.3	Combined GEMCS-SYMBØL Executive Program Divided in Overlays	68
3.4	Summary of the Combined GEMCS-SYMBØL Program	70
5.1	SYMBØL Information Flow Diagram of the Naphtha Reformer Plant for use with the Combined GEMCS-SYMBØL Executive	85
6.1	Overall Strategy for Solving Simulation projects	106
6.2	Process Flowsheet of the Naphtha Reformer Plant	109
7.1	Naphtha Vapour Pressure Variation against Temperature - Initial and Final Distillate	123
8.1	GEMCS Information Flow Diagram of the Naphtha Reformer Plant	26
8.2	Equilibrium Constants Variation with Temperature	130
8.3	Temperature Variation of Naphtha in the Naphtha Vaporizer/Superheater	151
8.4	Auxillary Burner, Water Injector and Cocurrent-Countercurrent Exchanger	151
8.5	Input/Output Information of the Auxillary Boiler	155
9.1	Plant Data of the Flue Gas Heat-Exchanger Train	179
9.2	Corrected Plant Data from Ripps' Analysis of the Flue Heat Exchanger Train	187
9.3	Program Listing of Newton Convergence Method	196

INDEX OF TABLES

		Page
2.1	SYMBØL General Data Set Format	16
3.1	Description of the KK(-) Vector	29
4.1	SYMBØL General Data Set Format Used in the Combined GEMCS-SYMBØL system	81
5.1	Equivalent Streams	91
5.2	SYMBØL Data Set of the Naphtha Reformer Plant Simulation for the Combined Executive	95
5.3	Comparison Between the Combined Executive with the Self-Standing GEMCS Simulation	102
8.1	Summary of Models for the Reformer Plant Simulation	127
8.2	Moles of Each Component Remaining per Mole of Naphtha Fed	132
8.3	Heat Exchangers in the Reformer Process	142
9.1	Composition of the Key streams of the Naphtha Reformer Plant	164
9.2	Flue Gas Analysis	167
9.3	Parameter Estimation in the Reformer Reactor	190
9.4	Parameter Estimation in the Water Gas Shift Reactors	192
9.5	Calculated and Measured Flows out of the Methanator Reactor	193
9.6	Overall Heat Transfer Coefficient-Area Product (UA) in Each Model	195
9.7	Constant Factor K for Pressure Drop Calculation in Models	199

Table and Appendices

A.1	Preparation of the GEMCS Data Set	207
A.2	Equipment list (EN vector) for CVGP1	220
A.3	Equipment list for CØNTL2 (or CØNTL3)	223
C.1	Data Set for READIN	247
D.1	Composition of the Naphtha Feed	253

LIST OF ABBREVIATIONS

A. For the SYMBØL Simulation

an.	another stream anywhere in the plant
CFLØW*	flow of a component
CFRAC*	fraction of one component in a stream which goes to the second stream specified
comp.	component
id.	identifier
i/p	input
NCASE	case number
NCP	number of components in the stream list
NCSD	total number of cascades on the information flow diagram
NMIX	total number of mixers on the information flow diagram
NPROB	problem number (identifier)
NRTR	total number of reactors on the information flow diagram
NSEP	total number of separators on the information flow diagram
NST	total number of streams on the information flow diagram
o/p	output
RATIØ*	ratio of one component in a stream which goes to the second stream specified
SCØMP*	composition of a stream (any number in the correct ratio)
SFRAC*	fraction of each component of a stream which goes to the second stfeam specified
st.	stream
STOICF*	stoichiometric coefficients of the reaction

* These abreviations are also the name of the subroutines called by SYMBØL to calculate these coefficients.

B. For the GEMCS simulation

EEN(-)	vector where all equipment parameter lists are stored
EN(-)	equipment parameter list
III	maximum number of streams that may be stored in the SN table
IS	position of a stream stored in the SN table
LLST(-)	vector containing the calculation order list
LØØP	loop (iteration) number
MSN	largest stream number on information flow diagram
NCALC	total number of equipments in LLST
NE	current equipment number (=EN(1))
NIN	number of input streams to an equipment (=EN(6))
NØCØMP	total number of flow components in the stream list
NØUT	number of output streams from an equipment (=EN(11))
NPØINT(-,-)	matrix indicating the start of the EN vector of equipment K (NPOINT(K,1)) in the EEN vector and the length of this EN vector (NPOINT(K,2))
NS(-)	vector of stream codes (NS(I) is the stream codes of stream I)
NTYPE	equipment type number (=EN(2)) (used in subroutine MØDULE)
SI(I,J)	input stream information of the I th. input stream, J the element
SN(-,-)	stream matrix (also called SN table) where the values of the streams to be saved (for further use in calculation of equipment) are stored
SØ (I,J)	output stream information of the I th. output stream, J th. element

I. INTRODUCTION

There are two general approaches to obtaining a steady-state simulation of a chemical process:

(i) The modular approach, where each equipment in the process is modeled and calculated in turn in a specified calculation order.

(ii) The global or simultaneous equation approach, where all the equations describing all physical/chemical changes occurring in the system are solved simultaneously.

Each of these approaches requires a computer executive system to facilitate the organization and manipulation of the subroutine modules or the equations. An example of the modular system is GEMCS, which has been developed and used for a number of years at McMaster. The simultaneous equation executive system, known as SYMBØL, was developed at Cambridge University; it uses a linearized set of equations. It has been found (H1), in some instances, that the two systems can be used together to provide a simulation that otherwise could not be obtained using each alone.

The object of this project was to combine the two executive systems so as to allow the two simulation approaches to be used at the same time without manual data manipulation when going from one system to the other. A secondary objective was to test the use of the combined system with a real plant simulation.

This report is divided into two parts: Part I describes how the two systems were combined while part II describes the modular GEMCS

simulation of a naphtha reformer plant which was used to test and evaluate the combined system.

Before proceeding with the details of the two parts, it is worthwhile reviewing some basic definitions in this field to orient the uninformed reader.

1.1 General Definitions

1.1.1 The Modular Approach

In the early stage of his education, the chemical engineer learns the basic theory that enables him to understand the unit operations and processes from which result the different equipment forming an actual chemical plant. (A unit operation involves purely physical reaction(s) such as distillation, solvent extraction, heat and mass transfer; a unit process involves chemical change(s) or reaction(s)).

A chemical plant is built with a great variety of such units. Each of these can be described mathematically (i.e. "modeled") by a set of equations based on laws of physics and chemistry; this set of equations, peculiar to one unit, enables one to transform the input information to the unit to the output information in much the same way as the corresponding equipment modifies the input flows entering it. These equations and the techniques to solve them can be grouped into a computer subroutine program called the module of the unit and defined as a unit computation.

In a simulation, the input information will flow and be transformed through a network of unit computations or modules, in much the same way as the incoming flows of material to a chemical plant are transformed through a network of equipments to produce a certain material production and some by-products.

An approach to simulation using flow of information through a network of unit computations is called modular.

1.1.2 Simultaneous Linear Equation Approach

Another approach involves the use of simultaneous linear equations. These equations are not based on the laws of physics and chemistry underlying a specific unit or equipment; it requires only to know the coefficients which linearly transform the input information to the output information, such as the fraction of a particular incoming component which reacts in a chemical reactor, or which goes to the overhead product of a distillation column, or many other similar coefficients.

Solving these equations simultaneously (by matrix inversion or equivalent) gives directly the flows to and from the different equipments. The advantage of this approach is its simplicity and the speed with which it can solve any given problem (in general, it is much more rapid than the modular approach). However, the matrix inversion or equivalent may be very difficult for a very large matrix since in general the larger the matrix the more expensive it is to invert by computer. The main problem in using the simultaneous linear equation approach which is often encountered, is getting the coefficients for the linear algebraic equation.

1.1.3 Questions Posed for any Simulation

As stressed in reference (C3), it is extremely important at the beginning of any simulation project to determine the questions for which the simulation is seeking answers. In other words, it is very desirable to know what one is actually looking for at the end of the simulation project;

for instance, one may want to bring some changes in the normal operating conditions of a plant in order to improve the production or to reduce the unit cost of the product.

These questions posed at the beginning of a simulation study will help in building up an appropriate stream list which will include key components which relate to the questions posed; trace components that do not affect the answers to these questions should not be included. Similarly, these questions will determine the level of sophistication required for any unit computation; for instance, the module of an equipment whose precision would not affect the answers of these questions is made very simple and only indicates the overall general behaviour of the unit. On the other hand, a key module whose precision would affect directly the answers is made very accurate (-ie. with a high level of sophistication). This level of sophistication which is required in some modules is a very important point since the higher the level of sophistication, usually the more complex and the more expensive it is to run on a computer.

1.2 Motivation for This Simulation Project

Most of the steady state simulations of chemical plants that have been carried out in the Department of Chemical Engineering at McMaster University have been accomplished through the use of a modular approach to simulation executive such as PACER, Process Assembly Case Evaluation Routine (C3), MACSIM, McMaster SIMulator System (J2), and, finally, GEMCS, General Engineering and Management Computation System (J1) which is an updated version of PACER and MACSIM. The GEMCS executive is now mainly used in the steady state simulation of existing chemical plants and in the design of new plants.

However, the very powerful GEMCS modular approach failed, when used alone, to answer some of the questions that were posed at the outset of the steady-state simulation (- a heat and material balance) of the Bayer process for extracting alumina from bauxite ore, operated by Alcan at Arvida, Quebec. This simulation of the Bayer process was carried out by the Department of Chemical Engineering at McMaster University under the supervision of T.W. Hoffman.

The simulation of the Bayer process was to be used to answer questions related to process changes. These process changes would reflect the economic importance of the chemicals used in it and consequently would affect the unit cost of alumina.

The Bayer process simulation is very complex because of the large number of recycle streams and the large fraction of the plant flow which is recycled in the main recycle stream. Indeed, the experiences with this plant have motivated many interesting general studies in the modular simulation technology (C4) (C5), among which is the one described below (H1):

In the Bayer process, there are a number of constraints that have to be met: for example, the concentration of sodium ions (as sodium hydroxide and sodium aluminate) and alumina (as sodium aluminate) and the concentration of sodium carbonate after the digester, the production of alumina trihydrate from the process and the ratio of aluminate ions to sodium ions after the precipitators. These constraints are maintained by adjusting the feedrates of bauxite, make-up caustic, condensate wash water at two locations and the purge flowrate to a calciner. This control system is highly interactive. The normal

method of handling such control/constraint problems in the modular approach is through control modules which are usually based on a linearization of the system; in addition, usually only one control variable relates to any given constraint (for example one feed stream controls the production). In a highly interactive system such as in the Bayer process, all control variables affect all constraints. In principle, the system can be linearized and the interactions can be accounted for through the use of the Jacobian in the Newton-Raphson convergence technique. As reported in reference (H1), such a scheme was tried without success. This lack of success was related to the difficulty in obtaining an accurate Jacobian and this arises because the derivatives in the Jacobian must be evaluated numerically (that is making a small perturbation in the feed rate and obtaining a converged solution of the entire plant simulation to within a given tolerance). Note that since the system has a large fraction recycle, a very large number of iterations are required to achieve a converged solution even with a modest tolerance ($(\underline{x}_{n+1} - \underline{x}_n)/x_{n+1} < 0.0001$). Although this solution with this tolerance was adequate for the simulation, the Jacobian was shown to be highly inaccurate. To achieve a Jacobian of sufficient accuracy would require an inordinate amount of computation time even for one perturbation; hence this method was proved to be unsuitable and an alternate solution was necessary.

It was shown (H1) that a satisfactory simulation which met all the constraints could be achieved if the GEMCS modular approach was combined with the simultaneous equation approach. In this case, a simulation was achieved at any given operating condition using the modular system; this solution was used to generate the coefficients in

SYMBØL linearized model equations which in turn could be solved to obtain the feedrates to meet the given constraints; and these new feedrates were used in the modular system to provide new linearized coefficients, and so forth. The system was shown to converge. In this example, the two executive systems were self standing and the information was calculated and/or transferred manually from one system to the other. There is an obvious need to have this part of the operation carried out automatically; hence the motivation for the project reported herein.

PART I

THE DEVELOPMENT OF THE COMBINED

GEMCS-SYMBØL ROUTINE

2. A REVIEW OF GEMCS AND SYMBØL

2.1 The Modular GEMCS Simulation

The modular GEMCS system is completely described in reference (J1) and as a result, only a brief summary of the system will be given here.

Basically, the process flow sheet is translated into an information flow diagram in which the information concerning the flow of material and the thermodynamic state of any stream (called the SN list) is transferred from one unit computation or module to another. All incoming information is assumed known and the incoming information is modified in the unit computation; the outgoing information reflects these changes and becomes the incoming information for the next module. Each unit computation must describe mathematically the physical and/or the chemical processes occurring in the actual physical units. Other unit computations will normally be included on the information flow diagram to describe control processes within the plant and also to aid in mathematical operations in effecting the overall mathematical simulation such as convergence promotion on an unknown recycle stream.

To each unit computation, a list of parameter must be included in a vector of information (called the EN vector) describing the streams in and out of the simulated equipment, the numbers identifying the equipment and the unit computation and some other parameters peculiar to each unit computation such as the overall heat transfer coefficient-area product in heat exchangers, the assumed values of outlet temperatures, etc. The EN vector is often referred to as the equipment vector, since it is specific to a particular equipment in the information flow diagram.

2.2 The Global or Simultaneous Equation Approach to Process Simulation

The earliest attempt to formulate a general method to obtain a steady-state mass balance for a chemical process can be traced to Nagiev (N1,N2). The method used in this work is based on Nagiev's concepts and was developed by Bending as a Ph.D. thesis. The executive system, named SYMBØL, is a proprietary program marketed by the Computer-Aided Design Centre in Cambridge, U.K.

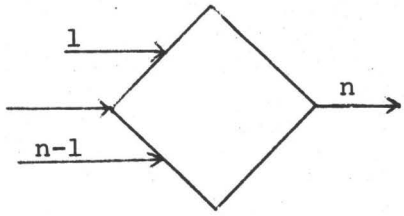
The method involves transforming the process flow sheet into an information flow diagram involving reactors, separators, mixers and/or cascades. The mass balance is formulated in terms of linear algebraic equations, the coefficients of which indicate how the input material to any of these units is transformed to the output. For example, in a chemical reactor, the coefficient may be the fraction of a given component which reacts, or in a distillation column (a separator), this coefficient may be the fraction of a given input component which goes overhead. The difficulty is that these coefficients must be known a priori.

For each unit in the system and for each output stream from it, there is an equation for each component. Hence, if there are N streams in the system and C components, plus the total flowrate, then N(C+1) equations result for the N(C+1) component flowrates, some of which are known. In any event, in principle, the system of equations may be solved simultaneously by matrix inversion techniques for the unknown component flowrates.

To set up the large number of linear equations is too tedious for manual manipulation. The SYMBØL executive system does this automatically by recognizing equations and their coefficients for the specific units (reactors, separators, mixers, cascades) and for material sources

(feed streams) or constraints among the components. Remembering that in simulation, the input information is assumed known, then these equations are formulated as follows:

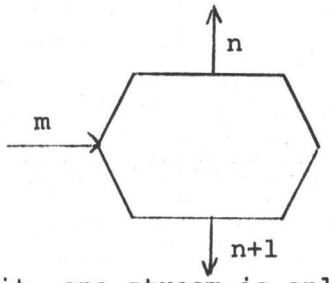
2.2.1 Mixer



At least two and a maximum of seven streams are completely mixed to give one output stream. Hence, if streams 1 to (n-1) are mixed to give stream n, then for each component (including the total flow):

$$F_{n,c} - \sum_{m=1}^{n-1} F_{m,c} = 0 \qquad c = 1,2,....C+1 \qquad (2.1)$$

2.2.2 Separator



In this unit, one stream is split into two output streams. Let stream m be split into stream n and n+1. Define $b_{m,n,c}$ as the fraction of material C in stream m which goes to stream n and then from this definition

$$F_{n,c} - b_{m,n,c} F_{m,c} = 0 \qquad c = 1,2,...C \qquad (2.2)$$

The material balance on any component yields:

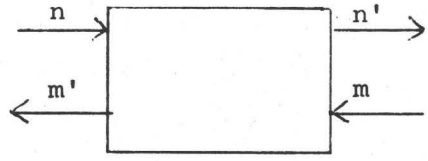
$$F_{n,c} + F_{n+1,c} - F_{m,c} = 0 \qquad c = 1,2,....C+1 \qquad (2.3)$$

For any stream

$$F_{n,C+1} - \sum_{c=1}^{C} F_{n,c} = 0 \qquad (2.4)$$

Hence there are 2(C+1) equations for this unit.

2.2.3 Cascade



Let stream m contact stream n with m' and n' becoming the effluent streams, respectively. Let the fraction $b_{m,m',c}$ be the fraction of component c in stream m which remains in stream m'. Then by definition:

$$F_{m',c} - b_{m,m',c} F_{m,c} = 0 \qquad c = 1,2,.....C \qquad (2.5)$$

By a material balance on component c

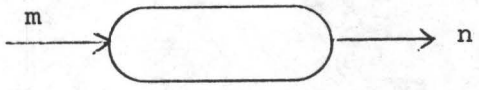
$$F_{m,c} + F_{n,c} - F_{m',c} - F_{n',c} = 0 \qquad c = 1,2,.....C \qquad (2.6)$$

The two additional equations are obtained for the total flow, viz:

$$F_{m',C+1} - \sum_{c=1}^{C} F_{m',c} = 0 \qquad (2.7)$$

and $$F_{n',C+1} - \sum_{c=1}^{C} F_{n',c} = 0 \qquad (2.8)$$

2.2.4 Reactor



Let the input stream be stream m, and the output stream, n. Define a stoichiometric coefficient for the reaction, based on mass of c reacting (not moles), be defined by d_c. Let g_c be the extent of reaction (conversion) of any particular reactant, c. Then by definition of the stoichiometric coefficient:

$$\frac{F_{n,c'} - F_{m,c'}}{F_{n,c} - F_{m,c}} = \frac{d_{c'}}{d_c} \qquad (2.9)$$

Note any consistent convention of letting the stoichiometric coefficient be negative for a reactant and positive for a product (or vice versa) can be adopted. Hence:

$$d_{c'}(F_{n,c} - F_{m,c}) - d_c(F_{n,c'} - F_{m,c'}) = 0 \qquad \begin{matrix} c=1,2\ldots C \\ c \neq c' \end{matrix} \qquad (2.10)$$

By definition of g_c:

$$(1 - g_c)F_{m,c} - F_{n,c} = 0 \qquad c = c' \text{ only} \qquad (2.11)$$

For the total flow

$$F_{n,C+1} - \sum_{c=1}^{C} F_{n,c} = 0 \qquad (2.12)$$

Hence C+1 equations are defined for this unit. Additional equations may be written for sources and constraints.

2.2.5 Sources

By defining the composition, x, of any feed stream and the flowrate of any one component or the total flowrate , the feed stream is defined by the following (C+1) equations:

$$F_{n,c} = \text{constant} \qquad c=1,2,....\text{or } C+1 \qquad (2.13)$$

$$F_{n,C+1} - \frac{F_{n,c}}{x_c} = 0 \qquad c=1,2,...C \qquad (2.14)$$

2.2.6 Additional Constraints

The power of SYMBØL resides in its ability to handle constraints in the system. These constraints may relate to the concentration of any given stream, the flowrate of any component (or the total) in another stream or the total flowrate being specified for a given stream (including a product stream). The latter constraint is simply

$$F_{n,c} = \text{constant} \qquad c=1,2,.....\text{or } C+1 \qquad (2.15)$$

The former is expressed through the definition of a factor $f_{m,c,m',c'}$ which is the relationship between the flowrate of component c in stream m and the flowrate of component c' in stream m'. This definition yields the equation

$$F_{m,c} - f_{m,c,m',c'} F_{m',c'} = 0 \qquad (2.16)$$

Note m and m' may refer to the same stream.

2.2.7 Use of Additional Constraint Equations

It is a function of the executive system to recognize when additional constraint equations are necessary. This situation occurs if any of the unit or source equations are missing. This means that for every coefficient in a reactor or a separator unit or for every flowrate of a source that is not defined, there must be an additional constraint equation. The executive system recognizes this through the variable MØDE. Whenever MØDE=1 for a reactor, separator or source, then an additional constraint equation is required.

2.2.8 Filling in the Data for SYMBØL

Two separate programs, DATAPREP1 and DATAPREP2, are provided with the SYMBØL executive. For very complicated and long information flow diagrams it is strongly susggested to use these programs to prepare the data for the SYMBØL executive; their use helps to avoid errors in filling in the data.

However, one can use the SYMBØL executive directly. The data

set format will look as the one summarized in the next table. The general order is to give the basic overall system information followed by the information required for all sources, then for all mixers, separators, cascades, reactors, and finally the extra constraints. It is important to mention again that for each mode=1 encountered for a source, a separator or a reactor, an additional constraints must be specified.

Table 2.1 SYMBØL General Data Set Format[1]

1-Overall System Data	2 TITLE CARDS defining the problem. *NPROB* *NCASE* *NST* *NCP* *NSCE* *NMIX* *NSEP* *NCSD* *NRTR*	
Equipment	**Mode**	**Information to be supplied**
2-SOURCES	1,2 and 3	*mode* *st.id* <SCØMP,NCP values>
	2	*comp.id.(may be total flow)* <CFLØW, one value>
	3	*comp.id**an.st.id**comp.id* <RATIØ, one value>
3-MIXERS	number of streams to be mixed (maximum=7)	*mode* *o/p st.id.* *i/p st.id.*...*c/p st.id.*
4-SEPARATORS	all modes	*mode* *i/p st.id.* *o/p st.id.* *o/p st.id*
	1	*o/p st.id.* <SCØMP,NCP values>
	2	*o/p st.id.* <SCØMP,NCP values> *comp.id.in it* <CFLØW, one value>
	3	*o/p st.id.* <SCØMP,NCP values> *comp.id.in it* *an.st.id.* *comp.id. in it* <RATIØ, one value>
	4	*o/p st.id.* <SFRAC,NCP values>
	5	*o/p st.id.* <CFRAC[2](total flow), one value>
	6[3]	*st.id* *comp.id.in it* *an.st.id* *comp.id. in it* <RATIØ one value>
5-CASCADES	1 and 2	*mode* *i/p st.id.* *o/p st.id.* *i/p st.id.**o/p st.id* *o/p st.id.in(a)* *i/p st.id.in(a)*<SFRAC,NCP values>
	2	*o/p st.id. in(b)* *i/p st.id.in(b)* <SFRAC,NCP values>

6-REACTORS 1,2 and 3		*mode* *i/p st.id.* *o/p st.id.*
	1	<STØICF,NCP values>
	2	*comp.id.* <fractional conversion,(1-CFRAC), one value>
	3	*comp.id. in o/p st.* *an.st.id.* *comp.id. in it* <RATIØ one value>
7-EXTRA CONSTRAINTS[4]	(a)	*st.id* Icomp.id.in it* *0* *0* <CFLØW,one value>
	(b)	*st.id* *comp.id.in it* *an.st.id.* *comp.id.in it* <RATIØ one value>

1- In this table, the number between the marks'* *' is an integer and the number between '< >' is a real number. Each line in the table represents a computer punched card.

2- Fraction of the whole input stream which goes to the specified output stream (composition unchanged).

3- There should be NCP such declarations, each indicating directly or indirectly the split across the separator.

4- For each mode 1 of any SOURCE, SEPARATOR or REACTOR, and extra constraint must be supplied.

3. DEVELOPMENT OF THE COMBINED GEMCS-SYMBØL EXECUTIVE

3.1 General

It is desired to build a combined GEMCS-SYMBØL executive system in which the modular simulation system (GEMCS), with its associated non-linear models provides the linear coefficients for the linear algebraic equations of the SYMBØL system. The solution from the SYMBØL system, with its ability to handle constraints is then used to generate the required feed flowrates to meet the required constraints and these flowrates are used, in turn, in the GEMCS system. Thus, an iterative procedure is used to achieve a simulation to a desired tolerance. The SYMBØL system may be thought of as acting as a control module within the GEMCS system. Any number of iterations may be used within the GEMCS system (that is, the GEMCS simulation may or may not be a converged solution) before a SYMBØL solution is affected. Experience will dictate the best procedure to obtain a simulation with the least computer time.

This procedure suggests a number of requirements:

1. The stream information vector must be the same for the two systems.

2. A separate information flow diagram is required for each system and both executive must be able to recognize the equivalence of streams on each information flow diagram. Note that with the restrictions SYMBØL puts on the types of units that can be used, there will not necessarily be a one-to-one correspondence of streams in the information flow diagrams.

Since the solution of the GEMCS simulation provides all the stream information, it is possible to calculate the coefficients for the linear algebraic equations directly from this information. Although it may be possible to obtain these coefficients directly from the equipment vector (EN vector), it was felt that such a procedure would lack generality. Moreover, when transferring information from SYMBØL back to GEMCS, only selected stream information is required such as selected feed streams and recycle streams.

Given these requirements and the suggested modus operandi for the combined system, the initial information to be supplied by the user for each system is defined.

3.2 GEMCS Data Set - Stream Equivalence Information

Since the GEMCS system is to be run first, the data set for it must be essentially the same as it would be if it were self-standing. However, there is some additional information required, namely the stream equivalence information.

In the GEMCS executive system, the second position in the stream information vector (SN(-,2)) is not used in general. It was used originally in the PACER/MACSIM system for stream code information but this function has been made redundant in the GEMCS system. This second position can be used here to indicate the stream number in the SYMBØL information flow diagram, corresponding to stream number SN(-,1) in GEMCS and thus the stream equivalence is set in a combined executive. It is therefore a simple matter to provide a subroutine within the SYMBØL system (called STR(S)) to search for the second position in the SN table to find the equivalent stream.

This procedure creates an additional problem, namely that such a stream must be stored in the SN table. When the GEMCS system is used alone, whether a stream is permanently stored in the SN table or not, is determined by the calculation order. Only recycle streams (that is, those for which initial values of the information flow vector must be specified) are required to be stored. Streams having a stream code equal to 7 are stored until they are used once more as input stream in the calculation order. In the present context, however, some of these streams will be required by SYMBØL for calculation of the coefficients in the algebraic equations for the SYMBØL units. On the other hand, the stream information for these streams will not have to be transferred from SYMBØL back to GEMCS after the SYMBØL calculation is completed. This suggests a requirement for a new stream code and therefore a new stream code (=4) was introduced to indicate streams which need to be saved for purposes of the SYMBØL coefficient calculations. The stream codes 1 and 2 are used exclusively for feed and recycle streams respectively, that is, streams whose information is definitely required to effect a GEMCS solution and hence must be transferred from SYMBØL to GEMCS.

3.3 Data Set for SYMBØL

After the GEMCS information flow diagram has been translated into a SYMBØL information flow diagram in terms of the available SYMBØL units, the user must supply that information which relates to the SYMBØL information flow diagram, viz:

(i) The number of streams in the SYMBØL information flow diagram and the number of components.

(ii) The number of sources, mixers, separators, cascades and reactors.

(iii) The equipment type (MØDE) for each of the units in (ii) and the stream and/or the key component identifiers for the input and output streams for each of the units.

(iv) The additional constraint information (the number of constraints being equal to the number of sources, separators and reactors which have MØDE = 1).

In some instances, the user may be required or it may be necessary to read in the information that cannot be calculated or determined from the GEMCS stream list. This information is read in a specified order in an AX(-)vector by DATAIN. Examples of when the AX vector is used are given in section 5.5.4.

3.4 The Calculation of the Unit coefficients in SYMBØL

As indicated earlier, the coefficients for the linear equations for the SYMBØL units are to be calculated from the stream information as calculated by the GEMCS simulation. It is important to recognize the types of coefficients that are required; these are summarized below as suggested from the model equations which were derived in section 2.2.

3.4.1 For Source Streams

(i) The composition of a stream as any number as long as they are in the correct ratio (NCP values required)

(ii) The flowrate of a component or the total flowrate (one value required)

(iii) The ratio of the flowrate of component A in stream B to the flowrate of component C in stream D (one value required)

3.4.2 For Separators

For separators, coefficients are required to describe the composition of a stream, the flowrate of a component or total flow, and the ratio of a component flowrate to another component flowrate (including total flow). These are the same calculations as described in section 3.4.1. In addition, the following are required:

(i) The fraction of each component of the input stream which goes to a specified output stream from the separator (NCP values required)

(ii) The fraction of the total input stream which goes to a specified output stream with composition unchanged (one value required)

(iii) as per (iii) in 3.4.1, except NCP such declarations must be made, remembering that each ratio must specify in some way the split across the separator.

3.4.3 For Reactors

(i) The stoichiometric coefficients (based on mass units) for one reaction (which may be a hypothetical reaction made up of a number of true chemical reactions). If a hypothetical reaction is used, care must be exercised in expressing all stoichiometric coefficients relative to this reactant, that is the mass of a component which reacts with a unit mass of this key component.

(ii) The extent of the reaction (fractional conversion) of a specified key component in the reaction described in (i) (one value required).

3.4.4 For Cascades

(i) The fraction of a component in a stream which proceeds through the cascade, the remaining fraction being transferred to the other stream with which it is contacted in the cascade unit (NCP values required)

Note that the type of calculation required for one equipment is essentially the same as that required for another equipment. For instance, the type of calculation required for the separator 3.4.2 (i) is essentially the same as that required for determining the fractional conversion in a reactor 3.4.3(ii) except that the latter is one minus the fraction calculated for the separator. Recognizing these requirements, six subroutines have been written to calculate these coefficients (CFLØW, CFRAC, RATIØ, SFLØW, SFRAC, STØICF), each being called from the main program of SYMBØL as they are required to complete the data set for it.

Given these general requirements of the combined system, the following sections give the details of the subroutines to calculate the coefficients for the SYMBØL system.

3.5 Description of the Subroutines Written for the Combined Executive

3.5.1 Subroutine STR(S)

The subroutine searches the stream list in the SN table to find the location of the stream (IS position in the SN table) with identifier, S, in the second position of the stream information vector. In this way, it is very similar to the subroutine STREAM(S) in the GEMCS executive system. It is used to find the stream information associated with the SYMBØL information flow diagram during the calculation of the linear coefficients for the SYMBØL calculation.

```
      SUBROUTINE STR(S)
C
C     SAME AS SUBROUTINE STREAM(S) EXCEPT IT SEARCHES FOR SN(IS,2)
C     INSTEAD OF SN(IS,1) ...
C
      COMMON NS(12),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,MIT,SN(72,16)
      COMMON/SPEC/LLL,MMM
      DO 1 IS=1,III
      KLK=IS
      IF(ABS(SN(IS,2)-ABS(S))-0.01) 4,1,1
    1 CONTINUE
      IS=KLK+1
      IF(S) 4,4,2
    2 WRITE(6,3) S
    3 FORMAT(/,1X,'STREAM',F5.0,' IN SYMBOL HAS NO SPECIFIED EQUIVALENT ST
     1REAM IN GEMCS',/,' WHICH IS SAVED IN THE SN MATRIX',/)
    4 IF(KPRNT(1)) 7,7,5
    5 WRITE(6,6) IS
    6 FORMAT(1H ,' ** IS =',I4)
    7 RETURN
      END
```

3.5.2 Subroutine STINFØ(L,N)

The purpose of this subroutine is to collect the correct stream information associated with any SYMBØL unit and to place it in the SI or SØ vector so that the coefficients in the SYMBØL units may be calculated.

This subroutine searches for stream number L on the SYMBØL information flow diagram in the SN matrix by setting S equal to L and by calling subroutine STR(S). If no stream number L is saved in the SN matrix it sets MMM equal to 1 and increases the value of LLL by 1 meaning that SYMBØL cannot get any information about stream number L because there is actually no equivalence between this SYMBØL stream and any GEMCS stream. This situation should only arise on two occasions. The first occasion is encountered whenever the user wants to set the coefficients a priori; in this case, one of the key stream required to calculate these coefficients is simply not made equivalent to any stream in GEMCS. The other occasion occurs whenever a SYMBØL stream has absolutely no equivalent stream in GEMCS. In each of these cases, the user must supply these coefficients. The system recognizes these coefficients by the variable MMM set equal to 1 and stores or reads them in the AX(-) vector where LLL indicates the position of the coefficients in it. More details on this feature are contained in Chapter 4 and an example is shown in section 5.5.4.

If an equivalent stream is found, this subroutine will store it in the SI vector for N in the range 1 to 4 or in the SØ vector for N in the range 5 to 8. It will store these streams in the SI or SØ vector in weight units if the modular simulation uses mole units. If the variable N is equal to 0, no transfer to the SI or SØ vector is required; N=0 is used in subroutine DATAIN to determine if the coefficients can be calculated from the GEMCS stream information or if they must be supplied by the user.

```
      SUBROUTINE STINFO(L,N)
C     THIS SUBROUTINE FINDS THE INFORMATION ABOUT SYMBOL STREAM ID L
C     IN THE SN TABLE OF GEMCS.  IF AN EQUIVALENT STREAM IS FOUND IN
C     GEMCS, THE RESULTING INFORMATION IS TRANSFERED TO THE SI(N,-) OR
C     SO(N-4,-) MATRIX.
C     HOWEVER, IF NO SUCH STREAM EQUIVALENCE BETWEEN SYMBOL AND GEMCS
C     SIMULATIONS IS FOUND, MMM IS SET EQUAL TO  1  AND LLL IS INCREASED
C     BY  1 .
C     N.LE.0 THE TRANSFER TO THE SI(-,-) OR SO(-,-) IS NOT REQUIRED
      COMMON NS(12),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(4,16)
      COMMON /SBL/ KSC(10,5),KM(7,9),KSP(6,8),KSP6(1,1),KC(1,1),KR(12,6),
     1             KE(5,4),KK(15),AX(100),AMW(11),TOLERX
      COMMON /SPEC/LLL,MMM
      NCP=NOCOMP
      MMM=0
      S=L
      CALL STR(S)
      IF(IS-III) 2,2,1
    1 MMM=1
      LLL=LLL+1
      RETURN
    2 CONTINUE
      IF(N.GT.0) GO TO 3
      RETURN
    3 IF(N.GT.4) GO TO 7
      DO 4 I=1,JJ
      SI(N,I)=SN(IS,I)
    4 CONTINUE
      IF(KK(11).EQ.1) GO TO 5
      RETURN
    5 SUM=0.
      DO 6 I=1,NCP
      J=I+5
      SI(N,J)=SI(N,J)*AMW(I)
      SUM=SUM+SI(N,J)
    6 CONTINUE
      IF(SI(N,3).LT.0.000001) RETURN
      SI(N,3)=SUM
      RETURN
    7 M=N-4
      DO 8 I=1,JJ
      SO(M,I)=SN(IS,I)
    8 CONTINUE
      IF(KK(11).EQ.1) GO TO 9
      RETURN
    9 SUM=0.
      DO 10 I=1,NCP
      J=I+5
      SO(M,J)=SO(M,J)*AMW(I)
      SUM=SUM+SO(M,J)
   10 CONTINUE
      IF(SO(M,3).LT.0.000001) RETURN
      SO(M,3)=SUM
      RETURN
      END
```

3.5.3 Subroutine DATAIN

The policy adopted at the outset of this project was to try to keep the two self-standing programs essentially the same in the combined system so that any changes could be incorporated easily into the self-standing or combined systems. In the SYMBØL executive system, information is read as it is actually needed in setting up the matrix to be solved. In the present case, some information must be read in initially such as MØDE and associated streams with each unit (essentially that which comes from DATAPREP 2, in the data preparation subroutines supplied with SYMBØL). The information associated with the equation coefficients and stream composition is not normally read in but, in most cases, calculated from the GEMCS stream information ; hence the need for reading and storing only part of the information which SYMBØL normally expects in its data set.

The initial information is read in with the initial data set and placed in a labelled COMMON/SBL/ is as follows:

KK(M) vector of basic system data as summarized in Table 3.1. Its length is 12 + the value of KK(12).

Table 3.1 Description of the KK(-) Vector

I	meaning of KK(I)
1	problem number (NPROB)
2	case number (NCASE)
3	total number of streams on SYMBØL information flow diagram (NST)
4	number of components in the stream information vector (does not include the total flow) (NCP)
5	number of source streams on SYMBØL information flow diagram (NSCE)
6	number of mixers on SYMBØL information flow diagram (NMIX)
7	number of separators on SYMBØL information flow diagram (NSEP)
8	number of cascades on SYMBØL information flow diagram (NCSD)
9	number of reactors on SYMBØL information flow diagram (NRTR)
10	number of extra constraints that must be specified (this value is calculated by subroutine DATAIN) (NEXTRA)
11	switch, = 1 if GEMCS uses mole units = 0 if GEMCS uses mass units
12	number of source streams to be tested in the convergence tester subroutine. (If set equal to 0, the system will test all sources stream)
13 ⋮ 12+KK(12)	source stream identifier whose stream will be tested for convergence in subroutine CØNTL3

KSC(I,J) : matrix of information for all NSCE sources in the SYMBØL information flow diagram, with dimensions KSC(NSCE,5)

mode of the source	J	meaning of KSC(-,J)	Basic required information to get
all modes	1	mode	-the composition of the source stream
	2	source stream id.	
2	3	identity of one component in the source	-the flow of that component
3	3	-identity of one component in the source	-the ratio of the flow of the specified component in the source to the flow of the other component in the other stream
	4	-another stream id.	
	5	-identity of one component in it	

KM(I,J) : matrix of information required for all NMIX mixers in the SYMBØL information flow diagram, with dimensions KM(NMIX,9)

J	KM(-,J)
1	number of stream to be mixed (maximum 7)
2	output stream identifier
3 to 9	input stream identifiers

KSP(I,J) matrix of information for all NSEP separators (including mode 6) in the SYMBØL information flow diagram, with dimensions KSP(NSEP, 8)

mode of the separator	J	meaning of KSP(-,J)
all modes (including) (mode 6)	1 2 3 and 4	MODE input stream identifier output stream identifiers
1	5	output stream identifier
2	5 6	output stream identifier component identifier
3	5 6 7 8	output stream identifier component identifier another stream identifier component identifier
4	5	output stream identifier
5	5	output stream identifier

KSP6(I,J) matrix of information for all separators having a mode 6 (say there are M6 of those) in the SYMBØL information flow diagram, with dimensions KSP6(M6,4×NCP)

J	KSP6(-,J)
1,5,9,.....(4*NCP-3)	stream identifier
2,6,10,....(4*NCP-2)	component identifier
3,7,11,....(4*NCP-1)	stream identifier
4,8,12,....(4*NCP)	component identifier

KC(I,J) matrix of information for all NCSD cascades in the SYMBØL information flow diagram, dimensioned KC(NCSD,9)

mode of the cascade	J	KC(-,J)
1 and 2	1	mode
	2	input stream identifier
	3	output " "
	4	input " "
	5	output " "
	6	output " " in (a)
	7	input " " in (a)
2	8	output stream identifier in (b)
	9	input " " in (b)

KR(I,J) matrix of information for all NRTR reactors in the SYMBØL information flow diagram, with dimensions KR(NRTR,6)

mode of the Reactor	J	KR(-,J)
1,2 and 3	1	mode
	2	input stream identifier
	3	output stream identifier
2	4	component identifier
3	4	component identifier
	5	another stream identifier
	6	component identifier

KE(I,J) matrix of information for all NEXTRA constraints, with dimensions KE(NEXTRA,4). Subroutine DATAIN calculates NEXTRA as it is reading the data. Each time a mode 1 is encountered for a source strea,, a reactor or a separator, the variable KEXTRA is increased by one; when all the information for the reactors has been read, KK(10) is set equal to KEXTRA (or NEXTRA)

J	meaning of KE(-,J)
1	stream identifier
2	component identifier
3	another stream identifier
4	component identifier

AX(I) vector of coefficients that cannot be calculated from GEMCS stream information; therefore this vector must be supplied by the user.

AMW(I) vector of molecular weights according to the order of each component in the stream information vector. The dimensions are AMW(NCP). This vector is used only when KK(11) = 1.

TØLERX variable specified by the user indicating the tolerance on which he wants the SYMBØL source stream flows to be tested against the corresponding GEMCS flows for the convergence criterion in CØNTL4.

As indicated earlier, the SYMBØL system is a mass balancing system and hence mass units are the required working units. On the other hand, it is often more convenient to use mole units in a modular simulation since many physical properties and most chemistry laws are given in moles unit.

The extra constraints are required to be read at the same time as the other matrix of information. For that reason, the main SYMBØL program does not count the actual number of constraints as it use to do in the self standing program. This count of additional constraints has been built up in subroutine DATAIN and removed from the SYMBØL main program so that the required information for the additional constraints could be read in at this point. Once the number of extra constraints is known, as calculated in DATAIN, the subroutine sets KK(10) equal to that calculated number of constraints. Whenever a mode 1 is used for any source, reactor or separator, the user usually will want to set the value of that constraint. For that reason, all coefficients for the extra constraints are not calculated through the use of the GEMCS stream information from the modular simulation; instead, they are specified by the user and stored in the AX(-) vector.

Each time the key streams to calculate the coefficients from the GEMCS stream information are read in, the subroutine DATAIN calls subroutine STINFØ(L,N) (where L is stream number just read and N is set equal to zero) to determine if there is an equivalent GEMCS stream available in the SN table. If there is no specified equivalent stream which is saved in the SN Table, the variable MMM tells DATAIN to read immediately the required coefficients and to store them in the AX(-) vector, starting at position LLL in the vector.

The general order to read in the data is essentially the same as per the self-standing SYMBØL program. It is summarized as follows:

1 - vector of basic system information (KK(I),I=1 to 9)

2 - matrix of information for all a) sources*
b) mixers*

c) separators*

d) cascades*

e) reactors*

f) extra constraints*

3 - KK(11)

4 - if KK(11) = 1, vector of molecular weights

5 - KK(12)

6 - if KK(12)>0, KK(I),I=13 to KK(12) - source streams to be tested in CØNTL4 for convergnece

7 - TØLERX

* Whenever the user must specify the coefficients because they cannot be calculated, these coefficients are read in on a different card, immediately after the key stream (and/or component) identifiers to calculate these coefficients have been specified. (Refer to Chapter 4 for more information).

```
      SUBROUTINE DATAIN
C
C     THIS SUBROUTINE READS IN THE BASIC INFORMATION FOR THE SYMBOL
C     SIMULATION IN THE COMBINED GEMCS-SYMBOL EXECUTIVE
C
      COMMON NS(12 ),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,NIT,SN(32,16)
      COMMON/SBL/ KSC(10,5),KM(7,3),KSP(6,8),KSP6(1,1),KC(1,1),KP(12,6),
     1            KE(5,4),KK(15),AX(100),AMW(11),TOLERX
      COMMON/SPEC/LLL,MMM
      READ(5,*) (KK(I),I=1,9)
      WRITE(6,1)
    1 FORMAT(1H1)
      WRITE(6,*) (KK(I),I=1,9)
      KEY=KK(1)
      KEXTRA=0
      NCP=KK(4)
      LLL=0
C
C     MAIN CYCLE
C
      L=5
   10 IF(KK(L)) 60,30,20
   20 GO TO (60,60,60,50,70,160,190,340,380),L
   30 L=L+1
      IF(L-9) 10,10,40
   40 IF(KEXTRA) 60,50,520
   50 READ(5,*) KK(11)
      IF(KK(11).EQ.1) READ(5,*) (AMW(I),I=1,NCP)
      WRITE(6,*) KK(11)
      IF(KK(11).EQ.1) WRITE(6,*) (AMW(I),I=1,NCP)
      WRITE(6,8) LLL
    8 FORMAT(/,* ****** LLL =*,I5,* *******,/)
      READ(5,*) KK(12)
      IF(KK(12).GE.1) READ(5,*) (KK(I),I=13,KK(12))
      WRITE(6,*) KK(12)
      IF(KK(12).GE.1) WRITE(6,*) (KK(I),I=13,KK(12))
      READ(5,*) TOLERX
      WRITE(6,11) TOLERX
   11 FORMAT(/,* TOLERX =*,1PE12.3)
      WRITE(6,9)
    9 FORMAT(1H1)
      RETURN
   60 WRITE(6,550)
      RETURN
C
C     SOURCES
C
   70 NSCE=KK(5)
      WRITE(6,2) NSCE
    2 FORMAT(/,* NSCE =*,I5,/)
      DO 100 I=1,NSCE
      READ(5,*) (KSC(I,J),J=1,2)
      WRITE(6,*) (KSC(I,J),J=1,2)
      MODE=KSC(I,1)
      NA=KSC(I,2)
      CALL SINFO(NA,0)
      IF(MMM.NE.1) GO TO 90
      K1=LLL
      LLL=LLL+NCP-1
      READ(5,*) (AX(J),J=K1,LLL)
      WRITE(6,*) (AX(J),J=K1,LLL)
   90 GO TO (100,110,140),MODE
  100 KEXTRA=KEXTRA+1
      GO TO 150
  110 READ(5,*) KSC(I,3)
      WRITE(6,*) KSC(I,3)
      IF(MMM.NE.1) GO TO 150
  120 LLL=LLL+1
  130 READ(5,*) AX(LLL)
      WRITE(6,*) AX(LLL)
      GO TO 150
  140 READ(5,*) (KSC(I,J),J=3,5)
      WRITE(6,*) (KSC(I,J),J=3,5)
      IF(MMM.EQ.1) GO TO 120
```

```
      NC=KSP(I,4)
      CALL STINFO(NC,0)
      IF(MMM.EQ.1) GO TO 170
  150 CONTINUE
      GO TO 3
C
C      MIXERS
C
  160 NMIX=KK(6)
      WRITE(6,3) NMIX
    3 FORMAT(/,' NMIX =',I5,/)
      DO 180 I=1,NMIX
      READ(5,*) KMIX,KOUT,(KM(I,J),J=1,KMIX)
      DO 170 J=1,KMIX
      KM(I,KMIX-J+3)=KM(I,KMIX-J+1)
  170 CONTINUE
      KM(I,1)=KMIX
      KM(I,2)=KOUT
      K1=KMIX+2
      WRITE(6,*) (KM(I,J),J=1,K1)
  180 CONTINUE
      GO TO 3
C
C      SEPARATORS
C
  190 NSEP=KK(7)
      WRITE(6,4) NSEP
    4 FORMAT(/,' NSEP =',I5,/)
      K6=4*NCP
      M6=0
      DO 330 I=1,NSEP
      READ(5,*) (KSP(I,J),J=1,4)
      WRITE(6,*)(KSP(I,J),J=1,4)
      MODE=KSP(I,1)
      GO TO (210,220,230,240,250,300),MODE
  200 READ(5,*) KSP(I,5)
      WRITE(6,*) KSP(I,5)
      NA=KSP(I,5)
      CALL STINFO(NA,0)
      GO TO (210,210,210,270,285),MODE
  210 IF(MMM.NE.1) GO TO 230
      K1=LLL
      LLL=LLL+NCP-1
      READ(5,*) (AX(J),J=K1,LLL)
      WRITE(6,*)(AX(J),J=K1,LLL)
  230 GO TO (240,250,260),MODE
  240 KEXTRA=KEXTRA+1
      GO TO 330
  250 READ(5,*) KSP(I,5)
      WRITE(6,*) KSP(I,6)
      IF(MMM.NE.1) GO TO 330
      LLL=LLL+1
      READ(5,*) AX(LLL)
      WRITE(6,*) AX(LLL)
      GO TO 330
  260 READ(5,*) (KSP(I,J),J=6,8)
      WRITE(6,*) (KSP(I,J),J=6,8)
      IF(MMM.EQ.1) GO TO 265
      NC=KSP(I,M)
      CALL STINFO(NC,0)
      IF(MMM.NE.1) GO TO 330
      GO TO 266
  265 LLL=LLL+1
  266 READ(5,*) AX(LLL)
      WRITE(6,*) AX(LLL)
      GO TO 330
  270 IF(MMM.EQ.1) GO TO 280
      KIN=KSP(I,3)
      CALL STINFO(KIN,0)
      IF(MMM.NE.1) GO TO 330
  280 K1=LLL
      LLL=LLL+NCP-1
      READ(5,*) (AX(J),J=K1,LLL)
      WRITE(6,*)(AX(J),J=K1,LLL)
      GO TO 330
```

```
  285 IF(MMM.EQ.1) GO TO 290
      KIN=KSP(I,2)
      CALL STINFO(KIN,I)
      IF(MMM.NE.1) GO TO 370
  290 READ(5,*) AX(LLL)
      WRITE(6,*) AX(LLL)
      GO TO 300
  300 M6=M6+1
      K1=K+2
      READ(5,*) (KSP6(M6,L),L=K,K1)
      WRITE(6,*) (KSP6(M6,L),L=K,K1)
      NA=KSP6(M6,K)
      CALL STINFO(NA,0)
      IF(MMM.EQ.1) GO TO 310
      NC=KSP6(M6,K+2)
      CALL STINFO(NC,0)
      IF(MMM.NE.1) GO TO 315
  310 READ(5,*) AX(LLL)
      WRITE(6,*) AX(LLL)
  315 K=K+4
  320 CONTINUE
  330 CONTINUE
      GO TO 30
C
C       CASCADES
C
  340 NCSD=KK(8)
      WRITE(6,5) NCSD
    5 FORMAT(/,' NCSD =',I5,/)
      DO 350 I=1,NCSD
      READ(5,*) (KC(I,J),J=1,7)
      WRITE(6,*) (KC(I,J),J=1,7)
      NA=KC(I,6)
      NB=KC(I,7)
      M=2
  350 CALL STINFO(NA,0)
      IF(MMM.EQ.1) GO TO 355
      CALL STINFO(NB,0)
      IF(MMM.NE.1) GO TO 360
  355 K1=LLL
      LLL=LLL+NCP-1
      READ(5,*) (AX(J),J=K1,LLL)
      WRITE(6,*) (AX(J),J=K1,LLL)
  360 IF(M.EQ.1) GO TO 370
      MODE=KC(I,1)
      IF(MODE.EQ.1) GO TO 370
      READ(5,*) (KC(I,J),J=8,9)
      WRITE(6,*) (KC(I,J),J=8,9)
      NA=KC(I,8)
      NB=KC(I,9)
      M=1
      GO TO 350
  370 CONTINUE
      GO TO 30
C
C       REACTORS
C
  380 NRTR=KK(9)
      WRITE(6,6) NRTR
    6 FORMAT(/,' NRTR =',I5,/)
      DO 390 I=1,NRTR
      READ(5,*)(KR(I,J),J=1,3)
      WRITE(6,*) (KR(I,J),J=1,3)
      MODE=KR(I,1)
      KIN=KR(I,2)
      KOUT=KR(I,3)
      CALL STINFO(KIN,0)
      IF(MMM.EQ.1) GO TO 395
      CALL STINFO(KOUT,0)
  395 GO TO (396,400,410),MODE
  396 KEXTRA=KEXTRA+1
      IF(MMM.NE.1) GO TO 510
      GO TO 49
  400 READ(5,*) KR(I,4)
      WRITE(6,*) KR(I,4)
```

```
      IF(MMM.NE.1) GO TO 51
      READ(5,*) AX(LLL)
      WRITE(6,*) AX(LLL)
      LLL=LLL+1
      GO TO 49
  41  READ(5,*) (KP(I,J),J=4,6)
      WRITE(6,*) (KP(I,J),J=4,6)
      IF(MMM.EQ.1) GO TO 49
      NC=KP(I,5)
      CALL SETINFO(NC,I)
      IF(MMM.NE.1) GO TO 51
      READ(5,*) AX(LLL)
      WRITE(6,*) AX(LLL)
      LLL=LLL+1
  49  K1=LLL
      LLL=LLL+NCP-1
      READ(5,*) (AX(J),J=K1,LLL)
      WRITE(6,*) (AX(J),J=K1,LLL)
  51  CONTINUE
      GO TO 30
C
C     EXTRA CONSTRAINTS
  52  KK(1)=KEXTRA
      WRITE(6,7) KEXTRA
    7 FORMAT(/,' KEXTRA =',I5,/)
      DO 54 I=1,KEXTRA
      READ(5,*) (KE(I,J),J=1,4)
      WRITE(6,*) (KE(I,J),J=1,4)
      LLL=LLL+1
      READ(5,*) AX(LLL)
      WRITE(6,*) AX(LLL)
  54  CONTINUE
      GO TO 56
  55  FORMAT(/,' ****** ERROR IN DATA SET FOR SYMBOL PART (MB ERROR 22,
     1NEGATIVE NUMBER OF EQUIPMENTS)',/)
      END
```

3.5.4 Subroutine CFLØW(N,M,E)

The purpose of this subroutine is to obtain the flowrate, E, of a component M in SYMBØL stream N. It is called by the SYMBØL main program at any time the flowrate of a component is required (that is for a source mode 2 and for a separator mode 2). It calls STINFØ(N,1) to get the GEMCS equivalent stream and then stores it in the SØ(1,-) vector after it translates that flowrate into mass units, if required.

Component M may be any of the NCP components or it can be the total flow (NCP+1=NVAR).

If the specified equivalent flow is not saved in the SN matrix, the program will take this value from the AX(-) vector which has already been read in by subroutine DATAIN.

```
      SUBROUTINE CFLOW(N,M,E)

       THIS SUBROUTINE FIND THE FLOW OF A COMPONENT IN A STREAM
       TRANSFER OF INFORMATION FROM GEMCS TO SYMBOL

       N = STREAM ID IN SYMBOL
       M = COMPONENT ID
       E = ITS FLOW

      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(72,16)
      COMMON/SBL/ KSO(16,5),KM(7,9),KSP(6,8),KSP6(1,1),KC(1,1),KR(12,5),
     1            KE(5,4),KK(15),AX(100),AMW(11),TOLERX
      COMMON/SYM/NEQ,NVAR,NCP,NEM,IRCW(2925),JCOL(2925),A(3825),NW(7560)
      COMMON/SPEC/LLL,MMM
      NK1=1
      CALL STINFO(N,NK1)
      IF(MMM.NE.1) GO TO 10
      E=AX(LLL)
      IF(KPRNT(10).EQ.1) WRITE(6,*) E
      RETURN
   10 CONTINUE
      IF(KPRNT(10).NE.1) GO TO 15
      WRITE(6,1000)
      WRITE(6,1001) (SI(1,J),J=1,JJ)
      WRITE(6,1000)
   15 IF(M.LE.NCP) GO TO 20
      E=SI(1,3)
      IF(KPRNT(10).EQ.1) WRITE(6,*) E
      RETURN
   20 E=SI(1,M+5)
      IF(KPRNT(10).EQ.1) WRITE(6,*) E
      RETURN
 1000 FORMAT(/,' ----',/)
 1001 FORMAT(5E16.5)
      END
```

3.5.5 Subroutine CFRAC(N2,N1,M,R)

The purpose of this subroutine is to calculate the coefficient for the separator (mode 5) or reactor (mode 2) equation. It calculates the fraction of component M of the input stream N1 which goes to the specified output stream N2 of a separator. This subroutine is also used to determine the fractional conversion X of component M in a reactor where N1 and N2 are the input and output streams of the reactor, respectively. In this case, the program recognizes that

$$X = 1 - R$$

This subroutine calls STINFØ(N1(or N2), 1(or 5)) to find the GEMCS equivalent stream of the SYMBØL stream N1 or N2 and stores these streams in the SI(1,-) (for N=1) or SØ(1,-)(for N=5) vector (in mass units). If one or both of these SYMBØL streams do not have an equivalent GEMCS stream stored in the SN matrix, the value of the ratio, R, is transferred from the AX(-) vector which has been read in via the DATAIN subroutine. Component M may be any of the NCP components or can be the total flow (the NCP+1 component).

```
      SUBROUTINE CFRAC(N2,N1,M,P)
C
C     TRANSFER OF INFORMATION FROM GEMCS TO SYMBOL
C     THIS SUBROUTINE FINDS THE FRACTION OF COMPONENT M OF THE INPUT
C     STREAM N1 WHICH GOES TO THE OUTPUT N2
C
      COMMON AS(12,),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NT,SM(72,16)
      COMMON/SBL/ KSC(10,5),KM(7,9),KSP(5,8),KSP6(1,1),KC(1,1),KP(12,6),
     1           KE(5,4),KK(15),AX(100),AMW(11),TOLERX
      COMMON/SYM/NEQ,NVAR,NCP,NEM,IPOW(2925),JCOL(2925),A(3825),NW(7560)
      COMMON/SPEC/LLL,MMM
      NK1=1
      NK2=2
      CALL STINFO(N1,NK1)
      IF(MMM.EQ.1) GO TO 30
      CALL STINFO(N2,NK2)
      IF(MMM.EQ.1) GO TO 30
      IF(KPRNT(10).NE.1) GO TO 5
      WRITE(6,1000)
      WRITE(6,1001) (SI(1,J),J=1,JJ)
      WRITE(6,1000)
      WRITE(6,1001) (SO(1,J),J=1,JJ)
      WRITE(6,1000)
    5 J=M+5
      IF(M-NCP) 10,10,20
   10 IF(SI(1,J).L.0.000001) GO TO 15
      P=SO(1,J)/SI(1,J)
      IF(KPRNT(10).EQ.1) WRITE(6,*) P
      RETURN
   15 P=1.0
      IF(KPRNT(10).EQ.1) WRITE(6,*) P
      RETURN
   20 P=SO(1,3)/SI(1,3)
      IF(KPRNT(10).EQ.1) WRITE(6,*) P
      RETURN
   30 P=AX(LLL)
      IF(KPRNT(10).EQ.1) WRITE(6,*) P
      RETURN
 1000 FORMAT(/,' ----',/)
 1001 FORMAT(5E16.5)
      END
```

3.5.6 Subroutine RATIØ(N1,M1,N2,M2,R)

This subroutine is used to calculate the ratio of the flow of component M1 in stream N1 to the flow of component M2 in stream N2. It therefore satisfies the requirements for a source with MØDE =3. It calls subroutine STINFØ to find the GEMCS equivalent streams N1 and N2 and stores them (in mass units) in the SI(1,-) or SI(2,-) vectors respectively. M1 and M2 can be any stream components, including the total flow.

If SYMBØL streams N1 and/or N2 have no specified GEMCS equivalent streams saved in the SN matrix, the model reads the value of the ratio from the AX(-) vector.

```
      SUBROUTINE RATIO(N1,M1,N2,M2,P)
C
C     TRANSFER OF INFORMATION FROM GEMCS TO SYMBOL
C     THIS SUBROUTINE CALCULATES THE RATIO OF THE FLOW OF COMPONENT M1
C     IN STREAM N1 TO THE FLOW OF COMPONENT M2 IN STREAM N2
C
      COMMON NS(12,),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(-2,16)
      COMMON/SBL/ KSC(10,5),KM(7,9),KSP(5,5),KSP5(1,1),KC(1,1),KP(12,5),
     1           KE(5,4),KK(15),AX(100),AMW(11),TOLERX
      COMMON/SYM/NEQ,NVAR,NCP,NEM,IPOW(2925),JCOL(2925),A(3825),NW(7560)
      COMMON/SPEC/LLL,MMM
      NK1=1
      NK2=2
      CALL STINFO(N1,NK1)
      IF(MMM.EQ.1) GO TO 70
      CALL STINFO(N2,NK2)
      IF(MMM.EQ.1) GO TO 70
      IF(KPRNT(10).NE.1) GO TO 5
      WRITE(6,1000)
      WRITE(6,1001) (SI(1,J),J=1,JJ)
      WRITE(6,1000)
      WRITE(6,1001) (SI(2,J),J=1,JJ)
      WRITE(6,1000)
    5 IF(M1-NCP) 10,10,20
   10 P=SI(1,M1+5)
      GO TO 30
   20 P=SI(1,3)
   30 IF(M2-NCP) 40,40,50
   40 Q=SI(2,M2+5)
      GO TO 60
   50 Q=SI(2,3)
   60 P=P/Q
      IF(KPRNT(10).EQ.1) WRITE(6,*) P
      RETURN
   70 P=AX(LLL)
      IF(KPRNT(10).EQ.1) WRITE(6,*) P
      RETURN
 1000 FORMAT(/,' ----',/)
 1001 FORMAT(5F16.5)
      END
```

3.5.7 Subroutine SCØMP(N,B)

This subroutine is used to obtain the composition (the B(-) vector) of a SYMBØL source stream N. (It is used also to find the composition of a specified output stream from a separator modes 1, 2 or 3). The composition of a source stream in SYMBØL can be any numbers in the correct ratio and therefore the flowrates of components can be taken directly from the appropriate GEMCS stream information vector. This is done by calling STINFØ to find the equivalent GEMCS stream and to store the stream information in the SI(1,-) vector (in mass units) from where it is transferred to the B(-) vector as required by the SYMBØL system.

If SYMBØL stream N has no specified GEMCS equivalent stream saved in the SN matrix, subroutine VALUES(B) is called and the B(-) vector (NCP values) is transferred from the AX(-) vector which was orginally read in via subroutine DATAIN.

```
      SUBROUTINE SCOMP(N,B)
C
C     THIS SUBROUTINE FINDS THE COMPOSITION OF A STREAM ...
C     TRANSFER OF INFORMATION FROM GEMCS TO SYMBOL
C
C     N = STREAM NUMBER IN SYMBOL
C     B = VECTOR USED TO RECORD THE INFORMATION
C
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(12,16)
      COMMON/SBL/ KSC(10,5),KM(7,9),KSP(8,9),KSP6(1,1),KC(1,1),KR(12,6),
     1            KE(5,4),KK(15),AX(100),AMW(11),TOLERX
      COMMON/SYM/NEQ,NVAR,NCP,NEM,IROW(2925),JCOL(2925),A(3825),NW(7560)
      COMMON/SPEC/LLL,MMM
      DIMENSION B(1)
      NK1=1
      CALL STINFO(N,NK1)
      IF(MMM.NE.1) GO TO 10
      CALL VALUES(B)
      RETURN
   10 CONTINUE
      IF(KPRNT(10).NE.1) GO TO 15
      WRITE(6,1000)
      WRITE(6,1001) (SI(1,J),J=1,JJ)
      WRITE(6,1000)
   15 DO 20 I=1,NCP
      B(I)=SI(1,I+5)
   20 CONTINUE
      IF(KPRNT(10).EQ.1) WRITE(6,*) (B(I),I=1,NCP)
      RETURN
 1000 FORMAT(/,' ----',/)
 1001 FORMAT(5F16.5)
      END
```

3.5.8 Subroutine SFRAC(N,M,B)

This subroutine calculates the fraction of each component of the input SYMBØL stream N which goes to the output stream M in a SEPARATOR (mode 4) or in a CASCADE (mode 1 and 2). It calls subroutine STINFØ which in turns finds the GEMCS equivalent streams and stores the respective stream information in SI(1,-) and SØ(1,-). The required fractions (B(-) vector) are calculated from the information in these SI and SØ vectors.

If no equivalent streams are found, subroutine SFRAC calls subroutine VALUES(B) which reads the required NCP values (coefficients) from the AX(-) vector, which in turn were read in via DATAIN.

```
      SUBROUTINE SFRAC(N,M,B)
C
C     TRANSFER OF INFORMATION FROM GEMCS TO SYMBOL
C     THIS SUBROUTINE CALCULATES THE FRACTION OF EACH COMPONENT OF
C     STREAM N WHICH GOES TO STREAM M
C
      COMMON NS(120),SI(5,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,NIT,SN(72,16)
      COMMON/SBL/ KS(10,5),KM(7,9),KSP(6,6),KSP6(1,1),KC(1,1),KP(12,6),
     1            KE(5,4),KK(15),AX(100),AMW(11),TOLERX
      COMMON/SYM/NEQ,NVAR,NCP,NEM,IPOW(2925),JCOL(2925),A(3825),NW(7560)
      COMMON/SPEC/LLL,MMM
      DIMENSION B(1)
      NK1=1
      NK2=5
      CALL STINFO(N,NK1)
      IF(MMM.EQ.1) GO TO 30
      CALL STINFO(M,NK2)
      IF(MMM.EQ.1) GO TO 30
      IF(KPRNT(10).NE.1) GO TO 5
      WRITE(6,1000)
      WRITE(6,1001) (SI(1,J),J=1,JJ)
      WRITE(6,1000)
      WRITE(6,1001) (SO(1,J),J=1,JJ)
      WRITE(6,1000)
    5 DO 20 I=1,NCP
      J=I+5
      IF(SI(1,J).LE.0.00001) GO TO 10
      B(I)=SO(1,J)/SI(1,J)
      GO TO 20
   10 B(I)=1.0
   20 CONTINUE
      IF(KPRNT(10).EQ.1) WRITE(6,*) (B(I),I=1,NCP)
      RETURN
   30 CALL VALUES(B)
      RETURN
 1000 FORMAT(//,' ----',/)
 1001 FORMAT(5F16.5)
      END
```

3.5.9 Subroutine STØICF(M,N,B)

This subroutine calculates the stoichiometric coefficients used in a SYMBØL reactor. Stream M and stream N are the output and input streams of the reactor respectively. The mass flowrate of all components on these streams are transferred from the equivalent streams in GEMCS by subroutine STINFØ (if they are there) to SØ(1,-) and SI(1,-) respectively. The program searches through these stream information vectors for any component which reacts completely and if found uses it as the key component. If no component reacts completely, the first component in the stream information vector which indicates a decrease in mass flowrate is taken as the key component. The stoichiometric coefficients d_c as defined by

$$d_c = \frac{F_{c,in} - F_{c,out}}{F_{key,in} - F_{key,out}} \qquad c = 1,2 \ldots . NCP \qquad (3.1)$$

are calculated for all components in the stream information vector. These are transferred to the B(-) vector as required by SYMBØL. Although the stoichiometric coefficients for reactants as defined by equation 3.1 may be positive or negative (those for products being of opposite sign, remembering that any sign convention is acceptable to SYMBØL),subroutine STØICF uses positive stoichiometric coefficients for reactants and negative coefficients for products.

If either the input or output stream has no specified equivalent GEMCS stream stored in the SN matrix, this subroutine calls subroutine VALUES(B) which transfers the NCP stoichiometric coefficients from the AX(-) vector, which was read in by DATAIN originally.

```
      SUBROUTINE STOICE(M,N,B)
C
C     TRANSFER OF INFORMATION FROM GEMCO TO SYMBOL
C     THIS SUBROUTINE CALCULATES THE STOICHIOMETRIC COEFFICIENTS
C     USED IN A SYMBOL REACTOR
C
C     M = OUTPUT STREAM ID
C     N = INPUT STREAM ID
C
      COMMON NS(12),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(12,16)
      COMMON/SBL/ KSC(10,5),KM(7,9),KSP(6,8),KSP6(1,1),KC(1,1),KP(12,6),
     1           KE(5,4),KK(15),AX(100),AMW(11),TOLERX
      COMMON/SYM/NEQ,NVAR,NCP,NEM,IROW(2925),JCOL(2925),A(3825),NW(7560)
      COMMON/SPEC/LLL,MMM
      DIMENSION B(1)
      NK1=1
      NK2=5
      CALL STINFO(N,NK1)
      IF(MMM.EQ.1) GO TO 60
      CALL STINFO(M,NK2)
      IF(MMM.EQ.1) GO TO 60
      IF(KPRNT(10).NE.1) GO TO 5
      WRITE(6,1000)
      WRITE(6,1001) (SI(1,J),J=1,JJ)
      WRITE(6,1000)
      WRITE(6,1001) (SO(1,J),J=1,JJ)
      WRITE(6,1000)
C
C     SELECT THE KEY COMPONENT
C
    5 DO 20 I=1,NCP
      J=I+5
      IF(SI(1,J)-0.01) 20,20,10
   10 IF(SO(1,J)) 20,40,20
   20 CONTINUE
      DO 30 I=1,NCP
      J=I+5
      IF(SI(1,J)-SO(1,J)) 30,30,40
   30 CONTINUE
   40 KCMP=I
C
C     CALCULATE NOW THE STOICHIOMETRIC COEFFICIENTS
C
      LCMP=KCMP+5
      ZZ=SI(1,LCMP)-SO(1,LCMP)
      IF(KPRNT(10).EQ.1) WRITE(6,*) KCMP,ZZ
      DO 50 I=1,NCP
      J=I+5
      B(I)=(SI(1,J)-SO(1,J))/ZZ
   50 CONTINUE
      IF(KPRNT(10).EQ.1) WRITE(6,*) (B(I),I=1,NCP)
      RETURN
   60 CALL VALUES(B)
      RETURN
 1000 FORMAT(/,' ----',/)
 1001 FORMAT(5F16.5)
      END
```

3.5.10 Subroutine VALUES(B)

This subroutine is used to transfer NCP coefficients from the AX(-) vector to the B(-) vector as required by SYMBØL. These values must have been supplied by the user and read in by subroutine DATAIN. Since this option is possible for all coefficients, all subroutines which calculate coefficients for SYMBØL may exercise the option of using user-supplied coefficients rather than those that could be calculated from GEMCS stream information.

```
      SUBROUTINE VALUES(B)
C
C     GET NCP VALUES IN THE AX VECTOR READ IN SUBROUTINE DATAIN
C
      COMMON NS(12),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SP(72,16)
      COMMON/SBL/ KSC(10,5),KM(7,9),KSP(6,8),KSP6(1,1),KC(1,1),KR(12,5),
     1           KE(5,4),KK(15),AX(100),AMW(11),TOLERX
      COMMON/SYM/NEQ,NVAR,NCP,NER,IROW(2925),JCOL(2925),A(3825),NW(7560)
      COMMON/SPEC/LLL,MMM
      DIMENSION B(1)
      I=0
      K1=LLL
      LLL=LLL+NCP-1
      DO 1 J=K1,LLL
      I=I+1
      B(I)=AX(J)
    1 CONTINUE
      IF(KPRNT(10).EQ.1) WRITE(6,*) (B(I),I=1,NCP)
      RETURN
      END
```

3.5.11 Subroutine CØNTL4(NX,NSCE)

Subroutine CØNTL4 is called by SYMBØL just before transferring back the specific calculated flows (feed and cut streams) to the SN table; it tests if the certain prespecified streams have converged to within a certain specified tolerance (TØLERX)* as follows:

CØNTL4 will test all SYMBØL source streams if KK(12)* is specified as being zero. However, very often not all source streams need to be tested for convergence since many of them do not directly affect the constraints. For that reason, this convergence tester was built in such a way that the user can specify the streams that should be tested for convergence (which may happen not to be the source streams); thus KK(12)$\geq$1 indicates the number of streams that should be tested for convergence criteria while KK(13 to 12+KK(12))* contains the identifier of these streams.

The convergence test is carried out in the units of the GEMCS modular simulation; for that reason, CØNTL4 calls subroutine STR(S) directly to find the GEMCS equivalent stream and then stores it in the SI(1,-) position (subroutine STINFØ is not used here since it always stores the GEMCS streams in the SI or SØ vectors in mass units). It then transfers from the A matrix the corresponding SYMBØL stream to store it in the SØ vector (in the units used in the modular simulation). It then carries out the convergence test for J = 6 to JJ (component flows) and for J = 3 (total

---

*TOLERX,KK(12), etc. were read in initially in subroutine DATAIN.

flow) as follows:

$$\frac{SØ(1,J) - SI(1,J)}{SØ(1,J)} \leq TØLERX$$ for all NSCE streams (if KK(12)=0)

or for all specified streams (if KK(12)≥1)

If all component flows in each tested stream (including the total flow) meet the above criterion, NIT and LØØP are set equal to 999 and the program will return and exit; if any one of the component flows tested does not meet the convergence criterion, both NIT and LØØP are set equal to 1 and the iterative procedure of using GEMCS and SYMBØL will proceed until the convergence criterion in CØNTL4 is met for all the component flows tested.

```
      SUBROUTINE CONTL-(NX,NSCE)

       TEST IF CALCULATED SYMBOL FLOWS (SOURCE STREAMS) HAVE THE
      SAME VALUE AS THEIR CORRESPONDANT GEMCS STREAM TO WITHIN TOLERX

      COMMON NS(12),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(72,16)
      COMMON/SBL/ KSC(12,5),KM(7,9),KSP(6,8),KSP6(1,1),KC(1,1),KR(12,6),
     1           KE(5,4),KK(15),AX(100),AMW(11),TOLERX
      COMMON/SYM/NEQ,NVAR,NCP,NEM,IPOW(2925),JCOL(2925),A(3825),NW(756)
      COMMON/SPEC/LLL,MMM
      NK1=1
      LL=0
      M=KK(12)
      IF(M.LE.0) M=NSCE


       MAIN CYCLE

      DO 170 N=1,M

       GET GEMCS FLOWS OUT OF THE SN MATRIX AND STORE IN SI(1,-) VECTOR

      NA=KSC(N,2)
      IF(KK(12).GE.1) NA=KK(12+N)
      S=FLOAT(NA)
      CALL STP(S)
      IF(IS-III) 5,5,170
    5 DO 6 J=1,JJ
    6 SI(1,J)=SN(IS,J)
      NB=SN(IS,1)+0.001

       GET SYMBOL FLOWS OUT OF THE A VECTOR
       GET SYMBOL FLOWS OUT OF THE A(-) VECTOR AND STORE IN SO(1,-)
      VECTOR IN THE UNITS USED IN GEMCS

      DO 10 J=1,JJ
   10 SO(1,J)=0.
      SO(1,1)=FLOAT(NB)
      SO(1,2)=FLOAT(NA)
      K1=NX+(NA-1)*NVAR
      SUM=0.
      DO 50 I=1,NVAR
      I=I+5
      IF(I-NCP) 20,20,30
   20 SO(1,J)=A(K1+I)
      IF(KK(11).NE.1) GO TO 50
      SO(1,J)=SO(1,J)/AMW(I)
      SUM=SUM+SO(1,J)
   30 IF(KK(11).EQ.1) GO TO 40
      SO(1,3)=A(K1+I)
      GO TO 50
   40 SO(1,3)=SUM
   50 CONTINUE

       NOW CARRY ON CONVERGENCE TEST

      WRITE(6,160) NA,NB
      LM=0
      DO 150 I=1,NVAR
      IF(I-NCP) 60,60,70
   60 J=I+5
      S=SO(1,J)
      G=SI(1,J)
      GO TO 80
   70 S=SO(1,3)
      G=SI(1,3)
   80 IF(S.LT.0.001.OR.G.LT.0.001) GO TO 150
      R=(S-G)/S
      IF(ABS(R)-TOLERX) 150,150,90
   90 LM=LM+1
      IF(LM-1) 150,100,110
  100 WRITE(6,190)
  110 IF(I-NCP) 120,120,130
  120 WRITE(6,200) I,S,G,R
```

```
      GO TO 14
  17  WRITE(6,210) S,G,P
  14  LL=LL+1
  15  CONTINUE
      IF(LM) 160,160,170
  160 WRITE(6,220)
  170 CONTINUE
      IF(LL.LT.1) GO TO 230
      IF(LOOP.GT.1) LOOP=1
      NIT=1
      KOUNT=1
      JSWMX=1
      GO TO 240
C
  180 FORMAT(/,' TEST ON SYMBOL STREAM',I4,' ( GEMCS STREAM',IL,' )')
  190 FORMAT(/,'  COMPONENT',11X,'SYMBOL',10X,'GEMCS',17X,'FRAC')
  200 FORMAT(7X,I3,6X,F12.5,4X,F12.5,8X,F12.10)
  210 FORMAT(6X,'TOTAL',5X,F12.5,4X,F12.5,8X,F12.10)
  220 FORMAT(' ****** CONVERGED FOR ALL COMPONENT ******')
C
  230 NIT=999
      LOOP=999
  240 RETURN
      END
```

3.5.12 Subroutine FEEDBK(NST,NX)

The purpose of this subroutine is to return to the SN table, in the GEMCS executive system, the updated feed stream information (stream code = 1) to meet the specified constraints and good estimates of the stream information for the recycle streams (stream code = 2) to hasten convergence under the updated feed conditions.

Subroutine FEEDBK considers each of the NST streams which were calculated by SYMBØL. It finds the equivalent GEMCS stream, if it is specified, and determines whether it has a stream code designation of 1 or 2. If it finds such a stream code, this stream information vector is transferred from the A matrix (which contains the stream information vectors for the SYMBØL streams) to the SN table in GEMCS with the appropriate stream number for the modular information flow diagram. In searching the SYMBØL stream table (the A matrix), NX+1 indicates the start of the stream flow-rate information in the A matrix; such information is available in chronological order according to the stream identification number on the SYMBØL information flow diagram.

The stream information vector is in mass units in the SYMBØL system; if the GEMCS system requires mole units, subroutine FEEDBK translates each element (including the total flow) to mole units before storing it in the SN table.

```
      SUBROUTINE FEEDBK(NST,NX)

C     THIS SUBROUTINE FEEDS THE CALCULATED SYMBOL STREAM VALUES BACK
C     TO GEMCS (IN THE SN TABLE) FOR ALL GEMCS STREAMS HAVING A STREAM
C     CODE (NS(-)) OF 1 OR 2 AND HAVING A SPECIFIED EQUIVALENT IN THE
C     SYMBOL SIMULATION ...

      COMMON NS(12 ),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP.
     1       NLP,NIT ,SN(72,16)
      COMMON/SBL/ KS (10,5),KM(7,9),KSP(6,8),KSP6(1,1),KC(1,1),KR(12,6),
     1           KE(5,4),KK(15),AX(100),AMW(11),TOLERX
      COMMON/SYM/NEQ,NVAR,NCP,NEM,IROW(2925),JCOL(2925),A(3825),NW(750)
      COMMON/SPEC/LLL,MMM
      N1=NVAR*NST+NX
      N2=0
      N3=NX+1
      N4=0
      WRITE(6,70)
      DO 60 K1=N3,N1,NVAR
      N2=N2+1
      CALL STINFO(N2,N4)
      IF(MMM.EQ.1) GO TO 60
      K2=SN(IS,1)+0.001
      IF(NS(K2).GE.3) GO TO 60
      SUM=0.
      DO 50 I=1,NVAR
      K3=K1+I-1
      IF(I-NVAR) 10,40,40
10    I=I+5
      IF(KK(11).GE.1) GO TO 20
      SN(IS,J)=A(K3)
      GO TO 30
20    SN(IS,J)=A(K3)/AMW(I)
30    SUM=SUM+SN(IS,J)
      GO TO 50
40    SN(IS,3)=SUM
50    CONTINUE
      WRITE(6,80) (SN(IS,J),J=1,JJ)
      WRITE(6,90)
60    CONTINUE
70    FORMAT(1H1,5X,*CALCULATED SYMBOL STREAMS TRANSFERRED BACK TO GEMCS
     1*,//)
80    FORMAT(5F16.5)
90    FORMAT(/,* ----*,/)
      RETURN
      END
```

3.6 Hardware Problem - Lack of CORE Storage

When the combined GEMCS and SYMBØL executive program was first put together, with the appropriate GEMCS simulation of the test case (to be described in Chapter 5 and in part II), the program could not be operated since there was not enough core storage to run it as a standard job on McMaster's CDC/6400 computer. This meant that the combined system which included the unit computations for the Reformer Plant (a problem of intermediate size) required more than 112,000(octal) core locations; in fact, the combined system could not be accommodated with the ultimate storage available (120,000). It was important that the system be accommodated as a standard job and that it be able to operate with even larger modular simulations. In the present case, some reduction in core requirements could have been affected by reducing the blank and labelled COMMØN declarations. This would not be a general solution to the problem since larger modular simulations (such as the Alcan simulation) would have to be accommodated in some other way.

For this reason, the FØRTRAN ØVERLAY procedures for the CDC system were used to overcome this difficulty. Since each simulation system is used sequentially, the combined system is ideally suited to the use of ØVERLAYS. This means that as the modular simulation is being calculated, the SYMBØL system is not required and can reside on a file and vice versa. That information which is common to the two simulation systems can reside in blank COMMØN in the main program at all times.

Since most computer users are not familiar with the overlay system of the CDC/6400, it is worthwhile reviewing the essential details of it as it is applied to this combined simulation executive system. These details

relate to the function of the ØVERLAYS and their set up and basic rules so that the reader will be able to follow how and why the new combined executive works.

3.6.1 FØRTRAN ØVERLAYS

Whenever a program is divided into overlays, the core storage requirements for the overall program are reduced, thereby making a more efficient use of the available field length. By definition, an overlay is a portion of a program which is written on a file and which is loaded at execution time without relocation. A program divided into overlays consists of FØRTRAN program units which are required at different times during program execution. They are loaded from their files only when they are required. Consequently, many subroutines (or program units) may occupy the same storage location during program execution.

There are three levels of overlays as depicted in Figure 3.1 each level being identified by a pair of integers as follows:

- main or zero overlay (0,0)
- primary overlay (i,0)
- secondary overlay (i,j)

where i and j are integers in the octal range 1-77. No two overlays which reside inside the same program for execution are allowed to have the same integer identifiers. They must be unique to each overlay.

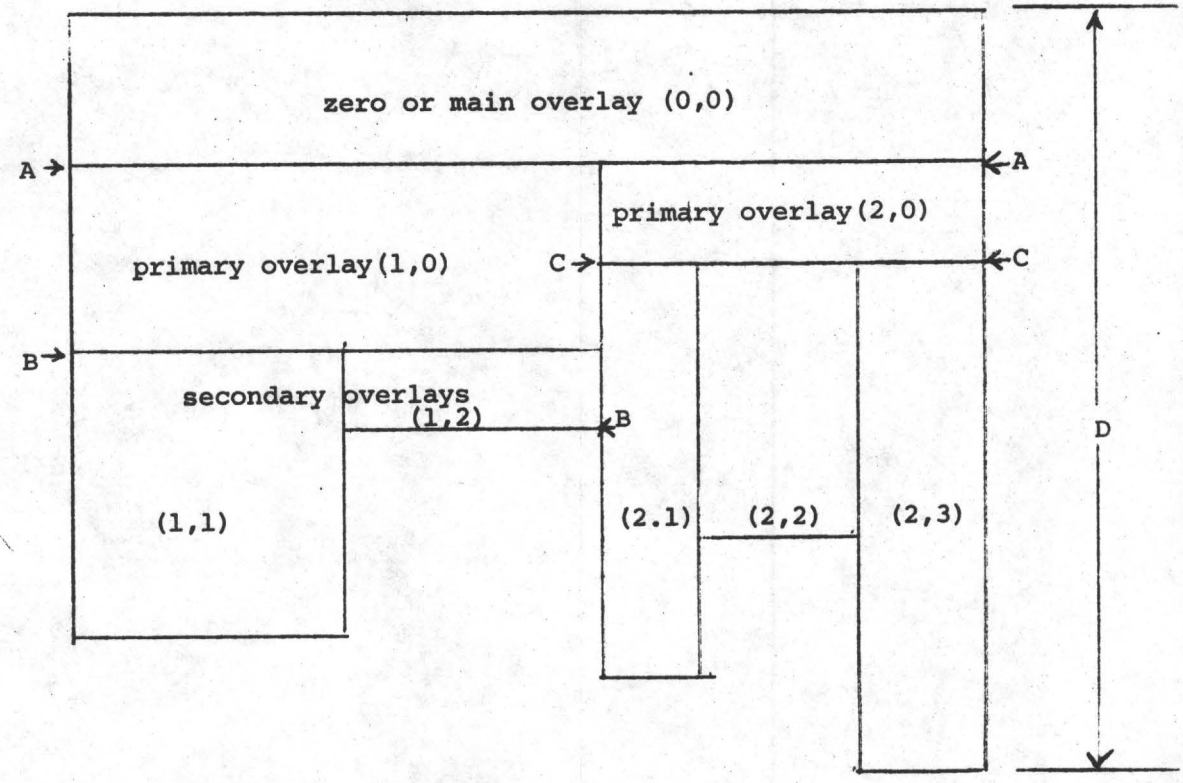


Figure 3.1 Schematic picture of a Program Divided in Overlays.

A - Fixed starting address for all primary overlays of the main overlay (0,0)

B - Fixed starting address for all secondary overlays of the primary overlay (1,0)

C - Fixed starting address for all secondary overlays of the primary overlay (2,0)

D - User field length actually used.

An overlay is established with an ØVERLAY directive which has the following format, starting in column 7 or later (note that each ØVERLAY directive must be contained in only one card):

```
ØVERLAY ( FILENAME, i,j)
```

where FILENAME is the name of the file on which the overlay is written

i and j are the integers identifying the overlay level

(in the octal range 0-77)

The first overlay to be loaded must have the identifiers (i,j)equal to (0,0) and must have a file name declared. In any ØVERLAY directive other than that for the main overlay, the level identifiers (i,j) must be specified but the file name may not be declared; in such cases, the overlay is simply loaded on the same file as the main overlay. However, for long overlays, it is preferable to have on a different file (This is for reason of speed; since the computer does not have to search for the overlay inside a file, it just calls the file named by the ØVERLAY directive). Finally, the main ØVERLAY is loaded whenever its filename is encountered (starting in column one and followed by a period) in the control cards required for the job the primary overlay is loaded whenever it is called by the main overlay; similarly, the secondary overlay is loaded whenever it is called by its primary overlay. Other parameters may be included in the ØVERLAY directive, but since these are not going to be used here, the reader may refer to references (B1), (C1) and (C2) for more details on overlays.

Each overlay consists of a main program and probably several subroutines. In a more practical way, the overlay consists of the program units that appear between its ØVERLAY directive and the next ØVERLAY directive or end of the source deck (The reader should refer to Figure 3.2, where the schematic picture of a program divided in overlays of Figure 3.1 is actually given as example on how the source deck is constructed for a program divided into overlays). As such, a PRØGRAM statement must appear after each ØVERLAY directive. All files that are required in any overlay such as input, output and punch files, must be declared

in the PRØGRAM statement of the main overlay only; no such files are declared in PRØGRAM statements of any overlay other than the main overlay (see example, Figure 3.2). The program name is the entry point for that overlay (it is used as the transfer address).

An ØVERLAY is brought from the file into the core by a CALL statement (which begins in column 7 or later) as follows:

```
CALL  ØVERLAY(8HFILENAME, i.j)
```

where FILENAME is the name of the file (written in H format) on which the overlay is written. The Hollerith count (mH) pertains to the length of FILENAME.

i and j are the level identifiers in octal (range 0 to 77_8) or their decimal equivalent.

Again, other parameters may be included (C1), (C2).

Data are passed between overlays through the use of blank or labelled CØMMØN. Any element of the CØMMØN declarations of the main overlay may be referred to by any primary or secondary overlay; however, the reverse is not true unless it is declared in the main overlay as well. In other words, communication or reference to an element of any common declaration of any overlay must be inside that overlay or be in higher level overlays. Whenever blank common is used to communicate data between overlays, care must be taken so that sufficient storage length is reserved to include the largest loaded overlay and the blank common declarations.

The main overlay must be loaded first. Any primary overlay followed by its associated secondary overlay (s) can follow, then any other primary overlay followed by its associated secondary overlay(s) and so forth.

Figure 3.2 Example of a Source Deck Program Divided in Overlays

```
OVERLAY(MAIN,0,0)
PROGRAM TST (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
...
CALL OVERLAY(4HJOHN,1,0)
CALL OVERLAY(3HBOB,2,0)
...
END
SUBROUTINE MARK
...
OVERLAY(JOHN,1,0)
PROGRAM ONE ZERO
...
CALL BOOK
...
CALL OVERLAY(1,1)
CALL OVERLAY(1,2)
...
END
SUBROUTINE BOOK
OVERLAY(1,1)
PROGRAM ONE ONE
...
CALL LIBRAR
...
END
SUBROUTINE LIBRAR
...
OVERLAY(1,2)
PROGRAM ONE TWO
...
CALL BOOK
...
END
OVERLAY(BOB,2,0)
PROGRAM TWO ZERO
...
CALL OVERLAY(3HAAA,2,1)
...
CALL OVERLAY(3HBBB,2,2)
...
CALL OVERLAY(3HCCC,2,3)
....
END
OVERLAY(AAA,2,1)
PROGRAM TWO ONE
...
END
OVERLAY(BBB,2,2)
PROGRAM TWO TWO
...
CALL MARK
...
END
OVERLAY(CCC,2,3)
PROGRAM TWO THREE
...
END
```

When an END statement is encountered in the main program of an overlay, control returns to the next executable statement after the CALL ØVERLAY directive statement that loaded it, for primary and secondary overlays. For a main overlay, an END statement in its main program returns control to the operating system.

3.6.2 The Combined GEMCS-SYMBØL Executive Divided into Overlays

The new combined GEMCS-SYMBØL executive is typical of a program that must be divided into overlays. Once all data are read in, the system has no further need for subroutines which read the data set into the computer. Similarly, when the modular simulation (GEMCS) is being calculated, the system does not need the simultaneous linear equation simulation (SYMBØL) in core and vice versa. As a result, the new combined executive was divided into one main overlay and three primary overlays as shown on Figure 3.3. The actual source deck configuration is summarized in Figure 3.4.

As seen on these figures, subroutine DISKIØ has been included in two different overlays, instead of using it in the main overlay only. Even if the subroutine (a small one anyway) is duplicated, at least the common declaration of the EEN and EN vectors and of the NPØINT matrix does not need to be saved in the main overlay, thereby reducing the core storage required, especially when the SYMBØL overlay is loaded (remembering that SYMBØL does not require the services of the EEN and EN vectors and of the NPØINT matrix). Similarly, subroutines STINFØ and STR have been duplicated in two primary overlays in order to reduce the core storage required when the GEMCS overlay is loaded since GEMCS does not use the variable placed in the labelled CØMMØN /SBL/ and /SYM/. This is particularly important when large plant simulations are attempted.

Figure 3.3 Combined GEMCS-SYMBØL Executive Program

Divided in Øverlays

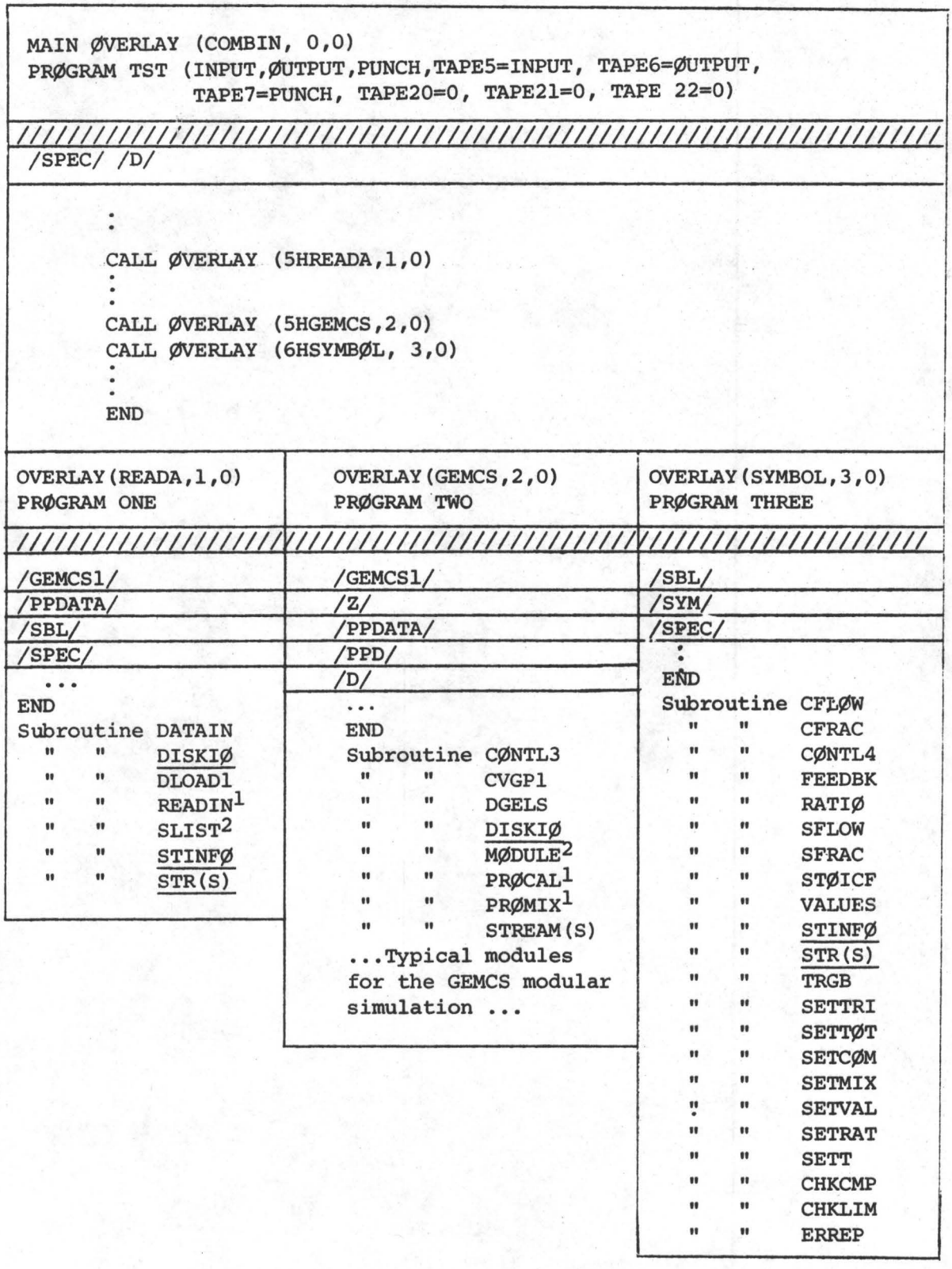


Figure 3.3 (Cont'd)

//////////	blank common	NS(-),SI(4,-),SØ(4,-),KPRNT(-), NØCØMP,JJ,IS,III,LØØP,NLP,NIT, SN(-,-)
/-/	labelled common	
	/PPD/	NØPP,X,XMOLE(-),PPSC(-.-),PPMX(-)
	/D/	KØUNT,JSWMX
	/SPEC/	LLL,MMM
	/PPDATA/	NTPC(-),LOC(-),PROP(-,-,-) (saved on file TAPE 21)
	/GEMCS1)	LLST(-),EEN(-),NPØINT(-,-),EN(-) NE,NIN,NØUT,MSN,MØDE,NPLNT,ISP, NC,NCALC,NSR,NCOUNT (saved on file TAPE 20)
	/Z/	XA(-),XØ(-),DX(-,-),X2(-),SCPD(-),YSP(-)
	/SBL/	KSC(-,5)KM(-,9),KSP(-,8),KSP6(-,-) KC(-,9),KR(-,6),KE(-,4),AX(-),AMW(-), TØLERX (saved on file TAPE 22)
	/SYM/	NEQ,NVAR,NCP,NEM,IRØW(-),JCØL(-),A(-),NW(-)

---

1 - Subroutine READIN,PRØCAL, and PROMIX are not an integral part of the Combined or GEMCS systems. However, these physical property subroutines are very useful in many simulations and, for that reason, they are included here.

2 - Subroutine MØDULE and SLIST are typical of any simulation. For that reason, they must be written for any new plant simulation although the basic format remains the same.

Figure 3.4 Summary of the Combined GEMCS-SYMBØL Program

```
 OVERLAY(COMBIN,0,0)
 PROGRAM TST (INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7=PUN
1CH,TAPE20=0,TAPE21=0,TAPE22=0)
 ...
 COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
1       NLP,NIT,SN(72,16)
 COMMON/SPEC/LLL,MMM
 COMMON/D/KOUNT,JSWMX
 ...
 CALL OVERLAY(5HREADA,1,0)
 ...
 CALL OVERLAY(5HGEMCS,2,0)
 CALL OVERLAY(6HSYMBOL,3,0)
 ...
 STOP
END

 OVERLAY(READA,1,0)
 PROGRAM ONE
 COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
1       NLP,NIT,SN(72,16)
 COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSR,NCOUNT
 COMMON/SBL/ KSC(10,5),KM(7,9),KSP(5,8),KSP6(1,1),KC(1,1),KR(12,6),
1            KE(5,4),KK(15),AX(50),AMW(11),TOLERX
 COMMON/PPDATA/ NTPC(17),LOC(17),PROP(17,16,4)
 COMMON/SPEC/LLL,MMM
 CALL DLOAD2
 CALL READIN
 CALL DATAIN
 BUFFER IN (20,0) (LLST(1),NCOUNT)
 CALL UNIT (20)
 BUFFER OUT (21,0) (NTPC(1),PROP(17,16,4))
 CALL UNIT (21)
 BUFFER OUT (22,0) (KSC(1,1),TOLERX)
 CALL UNIT (22)
 END
 SUBROUTINE DATAIN
 ...
 SUBROUTINE DISKIO(IFNT,MM)
 ...
 SUBROUTINE DLOAD2
 ...
 SUBROUTINE READIN
 ...
 SUBROUTINE SLIST
 ...
 SUBROUTINE STINFO(L,N)
 ...
 SUBROUTINE STR(S)
 ...

 OVERLAY(GEMCS,2,0)
 PROGRAM TWO
 COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
1       NLP,NIT,SN(72,16)
 COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSR,NCOUNT
 COMMON/PPC/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
 COMMON/PPDATA/ NTPC(17),LOC(17),PROP(17,16,4)
 COMMON/D/KOUNT,JSWMX
 REWIND 20
 BUFFER IN (20,0) (LLST(1),NCOUNT)
 IF(UNIT(20).GE.0.0) STOP *TAPE20 ERROR*
 REWIND 21
 BUFFER IN (21,0) (NTPC(1),PROP(17,16,4))
 IF(UNIT(21).GE.0.0) STOP *TAPE21 ERROR*
 ...
 END
```

Figure 3.4 (cont'd)


```
      SUBROUTINE CONTL3
      ...
      SUBROUTINE CVGP1
      ...
      SUBROUTINE DGELS(R,A,M,N,EPS,IER,AUX)
      ...
      SUBROUTINE DISKIO(IPNT,MM)
      ...
      SUBROUTINE MODULE(NT)
      ...
      SUBROUTINE PROCAL
      ...
      SUBROUTINE PROMIX(M)
      ...
      SUBROUTINE STREAM(S)
      ...
C
C     ALL SUBROUTINES OF THE MODULAR GEMCS SPECIFIC SIMULATION
C     ARE INCLUDED HERE
C
      ...

      OVERLAY(SYMBOL,3,0)
      PROGRAM THREE
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,NIT,SN(72,16)
      COMMON/SBL/ KSC(10,5),KM(7,9),KSP(6,8),KSP6(1,1),KC(1,1),KR(12,5),
     1            KE(5,4),KK(15),AX(50),AMW(11),TOLERX
      COMMON/SYM/NEQ,NVAR,NCP,NEM,IROW(2925),JCOL(2925),A(3825),NW(7560)
      COMMON/SPEC/LLL,MMM
      REWIND 22
      BUFFER IN (22,0) (KSC(1,1),TOLERX)
      IF(UNIT(22).GE.0.0) STOP *TAPE23 ERROR*
      ...
      END
      SUBROUTINE CFLOW(N,M,E)
      ...
      SUBROUTINE CFRAC(N2,N1,M,R)
      ...
      SUBROUTINE CONTL4(NX,NSCE)
      ...
      SUBROUTINE FEEDBK(NST,NX)
      ...
      SUBROUTINE RATIO(N1,M1,N2,M2,R)
      ...
      SUBROUTINE SCOMP(N,B)
      ...
      SUBROUTINE SFRAC(N,M,B)
      ...
      SUBROUTINE STINFO(L,N)
      ...
      SUBROUTINE STR(S)
      ...
      SUBROUTINE STOICF(M,N,B)
      ...
      SUBROUTINE VALUES(B)
      ...
      SUBROUTINE TRGB (NTC,NW,VAL,ICL,IRW,NVAR,NEQ,NEM,XMIN)
      ...
      SUBROUTINE SETTRI (NO,NC,N1,R1,N2,R2)
      ...
      SUBROUTINE SETTOT (NS)
      ...
      SUBROUTINE SETCOM (NS,P)
      ...
      SUBROUTINE SETMIX (INS,I,JNO,J)
      ...
      SUBROUTINE SETVAL (NS,NC,VAL)
      ...
      SUBROUTINE SETRAT (NS1,NC1,NS2,NC2,V)
      ...
      SUBROUTINE SETT (NS,NC,V)
      ...
      SUBROUTINE CHKCMP (I,IVAL,NVAL,L)
      ...
      SUBROUTINE CHKLIM (I,J,K,L)
      ...
      SUBROUTINE ERREP (L)
      ...
```

Note 1 The general statement

BUFFER ØUT(u,o)(a,b)

simply writes a list of data or words on file 'u'; 'a' is the first word to be transmitted on file 'i' and 'b' is the last one; arrays are stored in the order in which they appear in the dimension (or common) declarations with larger subscripts at higher addresses.

Each BUFFER IN or BUFFER OUT statement must call function UNIT(u); the function unit(u) checks if the data have been transmitted properly. There are 3 possible values returned by function UNIT(u).

- 1 indicate that the unit 'u' is ready to be used with no error.

0,1 indicate that the unit 'u' may be used, but the previous operation (in this case BUFFER IN or BUFFER OUT) retraced an error (like reading after an end of file, etc.)

Note 2 In the combined executive, when either the primary overlay GEMCS or SYMBØL is brought into core, their associated variable values that has been read in by primary overlay READA and stored on separate files, these variable values are read each time from their respective files. The statement REWIND(u) brings the 'reader' to the first datum (or word) 'a' stored on file 'u' and the statement BUFFER IN (u,o)(a,b) actually 'reads' the data from a to b inclusively.

During the execution of the primary overlay GEMCS some of the values (like the assumed values in the EN vector) of the variables that were read initially from their respective files may be updated. For that reason just before the primary overlay GEMCS returns control to the main overlay CØMBIN, the values of all the variables initially stored on files are overwritten on their respective files. In other words, the system updates these files with any change of values for the next possible use of their content. Although not indicated in this figure, (see Appendix A for complete listing of the combined executive), the following statements are encountered

```
      REWIND (20 )
      BUFFER OUT (20,0) (LLST(1),NCOUNT)
and CALL UNIT (20)
```

which updates the values of the EN vectors.

Labelled common /GEMCS1/, /PPDATA/, and /PPD/ have been written on separate files (TAPE20, TAPE21 and TAPE22 respectively) so that the value of their variables are not lost when a new primary overlay is loaded. Figure 3.4 explains how those files are actually used. The reader is referred to (C1 and C2) for more information on BUFFER IN and OUT and CALL UNIT (nn) statements which are used to store, to get and to change the values of the variables saved by these labelled common declarations on file. The essential information about these statements is described in Figure 3.4.

As a result of saving the variable values of these labelled common declarations on file, the blank common declared in the main overlay has been reduced significantly to the elements that are needed by the three main overlays only. When the primary overlay (SYMBØL,3,0) is loaded, the use of files to store the labelled common allowed a saving of 15,000 CM of core storage (for the intermediate size test case simulation).

The combined GEMCS-SYMBØL executive programs and subroutines (other than the subroutines described in this Chapter) are listed in Appendix A.

4. USING THE COMBINED GEMCS-SYMBØL SYSTEM

4.1 Introduction

It is important that the user of this combined system has available a working modular simulation which represents fairly well the measured data from the actual process. This normally will mean not only that the modules are formulated and shown to calculate reasonable results under a variety of conditions but also that the adjustable parameters in the models are chosen so as to predict accurately the actual output streams from them. This will mean that a GEMCS data set has been prepared. As mentioned previously, the main reason for using the combined system is to meet easily the constraints that are desired on the process through the adjustment of feed streams; thus the SYMBØL system obviates the use of many of the control models within GEMCS simulation.

4.2 SYMBØL Information Flow Diagram

After the GEMCS modular simulation has been completed, the next step in the use of the combined system is to construct an information flow diagram for the SYMBØL system using the SYMBØL units as described in Chapter 2. There are some aspects of this construction which do not arise when constructing a self-standing one. For example, usually each reaction in any reactor is represented by one reactor unit; this makes assigning stoichiometric coefficients to reactants and products an easy task. In the combined system, the stoichiometric coefficients are calculated from the input and output stream information for any reactor and these will reflect all reactions occurring in the reactor; hence only one reactor unit is required in the SYMBØL system for every reactor or series of reactors

in the GEMCS system. Sometimes the units in the SYMBØL system are quite different from the real system. For example, a change of phase which occurs in a cooler-condenser can be accommodated by a reactor model where the stoichiometric coefficients are 1 and -1 for the two phases and the fraction condensed is the extent of the reaction. Hence almost any chemical process can be translated into a SYMBØL information flow diagram through imaginative use of the modeling units.

4.3 Equivalent Streams

To minimize error in preparing the data set for the combined system, it is suggested that a table be constructed showing the equivalent streams and the stream codes for the GEMCS simulation alone. Some streams within the GEMCS system change temperature only; their mass flowrate and composition remain unchanged. Hence any one of these streams may be the equivalent stream in the SYMBØL system. To minimize storage in the SN table, that stream which already may be saved should be chosen as the equivalent stream.

The stream codes in the GEMCS data set must be updated to accommodate the equivalent streams. Generally, all GEMCS streams which are equivalent streams should be saved in the SN table, that is have a stream code less than or equal to 4. All equivalent streams other than recycle cut streams and feed streams should be given a stream code of 4.

In some instances, the user may wish to specify the coefficients required for a SYMBØL unit instead of having them calculated from the GEMCS stream information. In such cases the user does not specify one of the key equivalent streams to calculate these coefficients; however, the

coefficients must be supplied in the data set for the AX(-) vector where they are stored. As indicated in Table 4.1, these coefficients are read in immediately after the key stream (and/or key component) identifiers have been read; as mentioned previously, subroutine DATAIN, which reads the SYMBØL data set in the combined executive, checks automatically if all SYMBØL key streams have their GEMCS equivalent stream which are saved in the SN table immediately after they have been read in; if one of these key streams for the calculation of the SYMBØL coefficients has no equivalent stream which is saved in the SN table, then DATAIN reads the required SYMBØL coefficients for that unit (before it reads the information for the next unit) and stores them in the AX(-) vector. When the main program for SYMBØL prepares the matrix of SYMBØL coefficients the same procedure is applied; it calculates these coefficients using the stream information in the SN table or if not available SYMBØL reads them from the AX(-) vector.

The same situation exists when a SYMBØL stream has actually no equivalent stream in the GEMCS information flow diagram: the coefficients needed for a corresponding SYMBØL unit must be supplied through the AX(-) vector in the data set, immediately after the stream (and sometimes the component) identifiers have been supplied.

Again, to minimize error in the preparation of the data set, any GEMCS equivalent stream which is saved and specified in the SN table should be indicated (in brackets) beside its equivalent SYMBØL stream identifier on the SYMBØL information flow diagram so that the user will recognize readily if the coefficients for a SYMBØL unit must be supplied or can be calculated from GEMCS stream information.

4.4 Changes in the GEMCS data set

In constructing this combined system executive, an attempt was made to keep the data sets for both the GEMCS and SYMBØL parts nearly the same as for the two systems when they are self-standing. This procedure should minimize errors and reduce the work required to construct the data set. As a result, only minor changes need to be made in the original GEMCS data set. These are listed below in order.

4.4.1 KPRNT Vector

The element 1 to 8 in the KPRNT vector are the same as for GEMCS. KPRNT(9) and (10) have been used in the combined system as follows:

(i) KPRNT(10) = 1 causes the printing of all GEMCS streams called by SYMBØL and their position in the SN table (IS)

(ii) KPRNT(9) = 1 causes printing of all information required by SYMBØL along with a table of calculated SYMBØL flows and the stream information transferred back to the SN table for use by GEMCS.

KPRNT(I) = 0 suppresses this printing.

Note that the SYMBØL data set is always printed as it is first read.

4.4.2 Calculation Order List, Stream Codes

If control modules are used in the GEMCS simulation alone to control feed streams, these should be removed. The calculation order list will have to be altered accordingly, including any modification required

to the stream codes resulting from this change. In fact, some streams may disappear and they should be given a stream code of zero. All internal streams which must be saved, other than cut recycle streams, should be given a stream code of 4, as suggested earlier. Note that the storage requirements in the SN table may increase considerably and the user should check to ensure that the SN table dimensions are sufficient to accomodate the new stream storage requirements.

4.4.3 Stream Information

Since the equivalence of streams is provided through the stream numbers in the SN(IS,1) and SN(IS,2) positions of the stream information vector, this information must be read in with the data set for GEMCS. This requirement increases significantly the number of streams to be read; however, with these equivalent streams, only the first two numbers are required to be punched; the remaining information may be zero since it is not required in the GEMCS calculation (Note that the number of cards for each stream has to be appropriate for the number of components in the stream information vector).

Sometimes, a stream in the GEMCS simulation alone is read in showing the composition of the stream in mole or weight percent (depending upon the units used in the GEMCS simulation) with its total flow SN(IS,3) set equal to one. In the combined system, this total flow should be made equal to zero, the reason being that if that stream is used only in very rare occasions (where, for instance, in normal plant operation a water stream is introduced to control the temperature of another stream if it exceeds a certain temperature only), the total flow is the only information that SYMBØL will have to understand if the stream is used (SN(IS,3)>0.) or not SN(IS,3)=0.); however, the composition of that stream may be required by

SYMBØL, even if its total flow is zero.

4.4.4 Equipment Information

Aside from removing the EN vector for the control modules that have been removed from the GEMCS information flow diagram, modification of the equipment vector data is not required. Note that no error will occur if such equipment information is kept even if its modules have been removed from the GEMCS information flow diagram.

4.4.5 Additional Information

At the end of the normal GEMCS data set, the user must specify the number of iterations he wants to be done in the GEMCS simulation before transferring to the SYMBØL system. (FØRMAT F12.5). If the GEMCS system is to converge before transferring, any large number smaller than 999 (say 200) should be read in.

4.4.6 Physical Property Information

If the physical property package (subroutines PRØCAL and PRØMIX) is used in the GEMCS simulation, the coefficients for the polynomial regression equations should be included after the above information.

4.5 Preparing the SYMBØL Data Set

The SYMBØL data set in the combined system is essentially the same as that required by SYMBØL when it is self-standing, except now most coefficients are calculated within the program from the GEMCS stream information. However, whenever the coefficient(s) must be supplied,

these coefficients must appear on a new data card. Note that all extra-constraints must be supplied since they are always set by some requirement usually relating to technical feasibility or desired condition. The general format for data is shown on Table 4.1. In this table, all coefficients indicated between the '< >' marks are normally computed from GEMCS stream information (except for the additional constraints). If these coefficients are to be supplied by the user (i.e. no equivalent stream is supplied), they are read in at the appropriate position in the data set (as shown in Table 4.1).

Except in the case of the simplest information flow diagrams, it is recommended that the DATAPREP1 and DATAPREP2 programs (H3) be used to help in preparing the data set for SMMBØL. The inexperienced user should always use them.

4.6 Additional Remarks

The final step in the use of the combined system before its execution is to adjust the dimensions on the SN table to accommodate the increased number of streams that must be saved, and on the labelled commons /SBL/, and /GEMCS1/ as appropriate.

The user should also update subroutine MØDULE(NT) since it should call subroutine CØNTL3 instead of CØNTL2. Figure 3.4 shows where the GEMCS modules for a specific plant simulation should be inserted in the source deck of the combined system.

Also, due to minor changes in the calculation order list, (some CONTROL modules being removed), the user must specify the number of modules controlled by CØNTL3 (EN(16)).

Table 4.1 SYMBØL General Data Set Format Used in the Combined[1]

GEMCS - SYMBØL system

1 - Overall *NPROB* *NCASE*

System Data *NST* *NCP* *NSCE* *NMIX* *NSEP* *NCSD* *NRTR*

Equipment	Mode	Information to be supplied
2-SOURCES	1,2 and 3	*mode* *st.id* <SCØMP, NCP values>
	2	*comp.id. (may be total flow)* <CFLØW, one value>
	3	*comp.id. in this st* *an.st.id* *comp.id. in it* <RATIØ, one value>
3-MIXERS	no. of streams to be mixed (maximum=7)	*mode* *o/p st.id* *i/p st.id* ... *i/p st.id*
4-SEPARATORS	all modes	*mode* *i/p st.id.* *o/p st.id* *o/p st.id*
	1	*o/p st.id.* <SCØMP, NCP values>
	2	*o/p st.id* <SCØMP,NCP values> *comp. id. in it* <CFLØW, one value>
	3	*o/p st.id.* <SCØMP, NCP values> *comp.id. in it* *an.st.id* *comp.id.in it* <RATIØ, one value>
	4	*o/p st.id* <SFRAC, NCD value>
	5	*o/p st.id* <CFRAC[2], one value>

	6[3]	*st.id* *comp.id.in it* *st.id.* *comp.id.in it* <RATIØ, one value>
5-CASCADES	1 and 2	*mode* *i/p st.id* *o/p st.id.**i/p st.id* *o/p st.id* *o/p st.id.in(a)* *i/p st.id.in(a)* <SFRAC,NCP values>
	2	*o/p st.id.in(b)* *i/p st.id.in(b)* <SFRAC,NCP values>
6-REACTORS	1,2 and 3	*mode* *i/p st.id.* *o/p st.id.*
	1	<STØICF,NCP values>
	2	*comp.id* <fractional conversion (1-CFRAC), one value> <STØICF,NCP values>
	3	*comp.id.in o/p st.* *an.st.id.* *comp.id.in it* <RATIØ, one value> <STØICF,NCP values>
7-EXTRA CONSTRAINTS[4]	(a)	*st.id.**comp.id.in it**an.st.id.**comp.id.in it* <RATIØ, one value>
	(b)	*st.id* *comp.id.* *0* *0* <CFLØW, one value>
8-a)*KK(11)*		if 1, GEMCS uses mole units is 0, GEMCS uses mass units
b)		if KK(11)=1, <MW,NCP values>

9-a)	*N=KK(11)*	- N is the number of sources that the user wants to test in CØNTL4 for convergence criteria - if set=0, CØNTL4 will test all SOURCE streams
b)	if N>0,	*SOURCE st.id.*.....*SOURCE st.id.*....., N integers indicating the stream identifier of each of the N streams that the user wants to have tested in CØNTL4 for convergence criteria
10-	<TOLERX>	tolerance on convergence criteria in ØONTL4

1- In this table, the number between the marks'* *' is an integer and must be specified by the user. The number between the marks'< >' is a real number; only the <EXTRA-CONSTRAINTS> and <TOLERX> must be specified by the user; any other '< >' coefficient(s) is (are) normally calculated from GEMCS stream information; if it (they) can not be calculated from GEMCS stream information (no equivalent stream specified), the user must supply these whenever they are encountered in the table.

2- Fraction of the whole input stream which goes to the specified output stream (composition unchanged).

3- There should be NCP such declarations, each indicating directly or indirectly the split across the separator.

4- For each mode 1 of any SOURCE, SEPARATOR or REACTOR, an extra-constraint must be supplied.

5. TESTING THE COMBINED GEMCS-SYMBØL EXECUTIVE

5.1 Introduction

The Naphtha Reformer Plant, whose GEMCS modular simulation is fully described in part II, was used as an example to test the actual combined GEMCS-SYMBØL executive. This plant was chosen as it is a very simple one and very well known to the writer. However, it should be stressed that the combined executive would be better tested by using a more complicated plant such as Alcan's Bayer Process in which case many constraints and a large number of recycles interact to modify to some extent the desired production. In the case of the Naphtha Reformer Plant, the only specified constraint is the hydrogen production which is not a strong non-linear function of the naphtha feed.

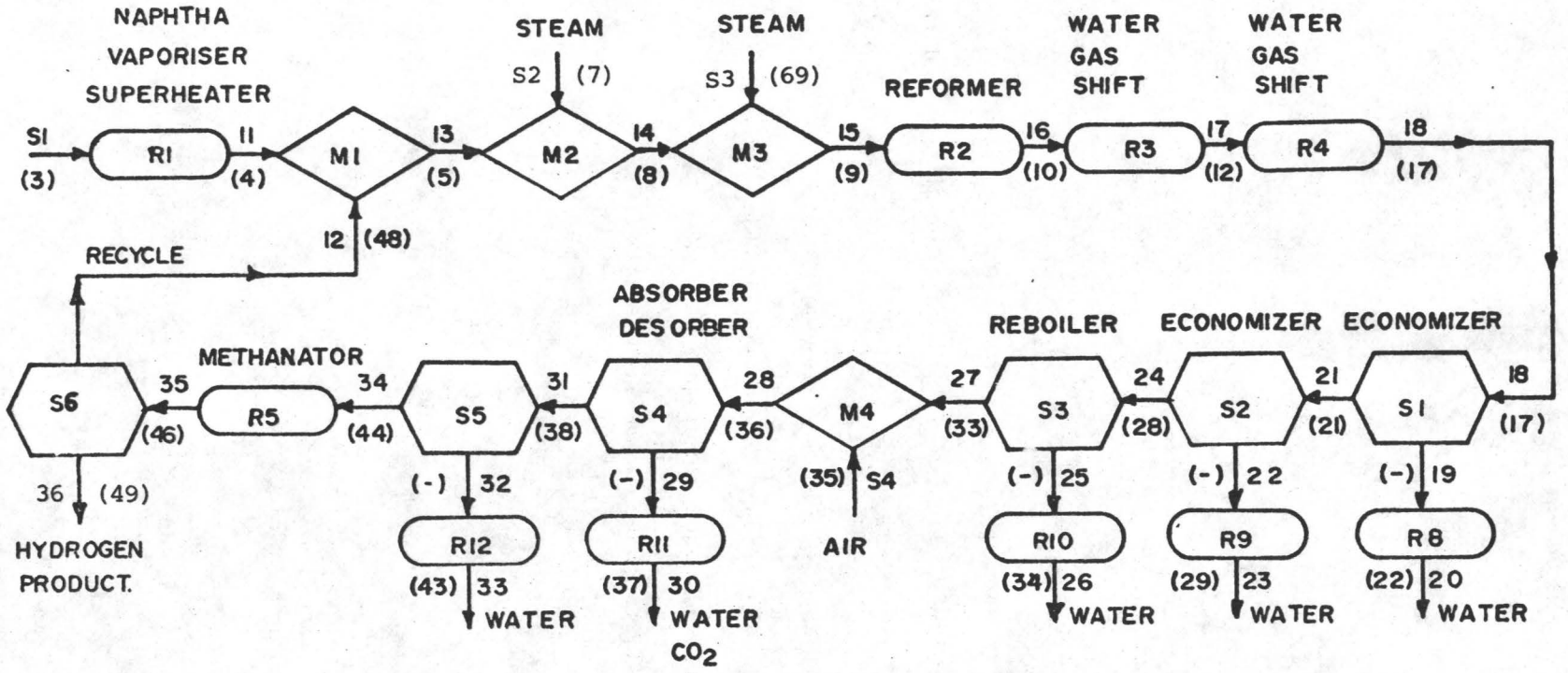
In the following sections, using the specific example, a complete description of the steps required to use the combined executive will be given, along with some useful comments.

5.2 SYMBØL Information Flow Diagram

Using the SYMBØL units described in section 2.2, the SYMBØL information flow diagram was constructed as shown on Figure 5.1 using both the process flowsheet and the GEMCS modular information flow diagram. A total of 41 streams, 10 sources, 7 mixers, 6 separators, and 12 reactors were used.

It should be noted that each independent main stream (in this case, the naphtha/stream/recycle reaction stream and the fuel combustion stream) requires a separate diagram on the overall SYMBØL information flow diagram. The interaction between these two diagrams arises through

MAIN PROCESS STREAM:



FUEL/FLUE GAS SYSTEM:

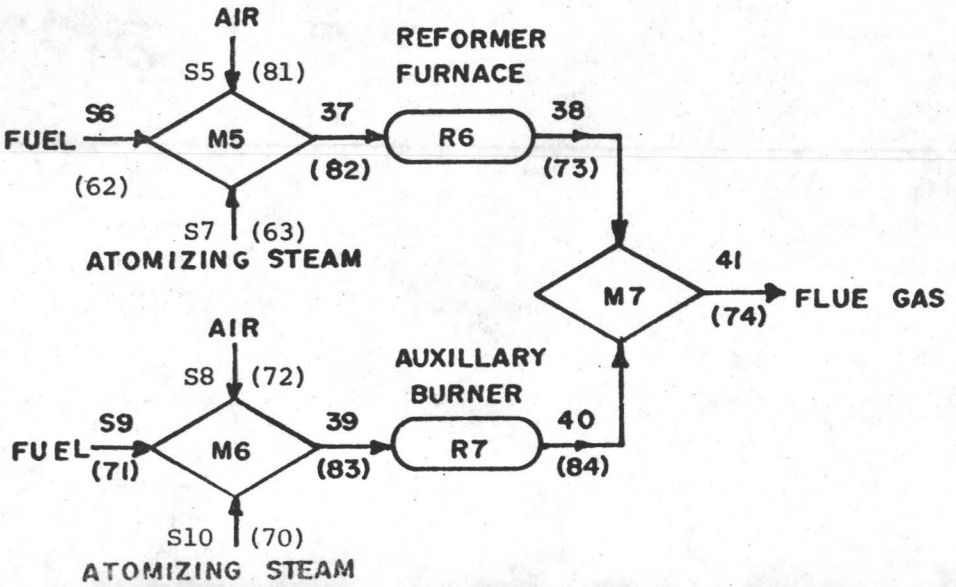


Figure 5.1 SYMBØL Information Flow Diagram of the Naphtha Reformer Plant for use with the Combined GEMCS-SYMBØL Executive

the coefficients relating the flowrates of the main streams.

It is worthwhile noting at this point that reactors R2, R3 and R4 could have been grouped together into only one SYMBØL reactor; similarly separators S1, S2 and S3 could have been grouped into only one separator, since in each of these, only water (which is condensed in the corresponding GEMCS equipment) is separated from the main flow of product gases. However, the SYMBØL information flow diagram was constructed as per Figure 5.1 in order to keep a close correspondance with the GEMCS information flow diagram. For a large and complicated plant however, the above procedure of grouping several reactors (or separators) in series, should be used to reduce the dimensions required for the SYMBØL simulation part (thereby reducing the core storage requirements).

Each equipment on Figure 5.1 has the following meaning (the corresponding GEMCS module is shown in brackets).

R1 : Naphtha vaporizer/super heater (EXCHO3,NE=3).

Since there is a change of phase occurring in EXCHO3, where the incoming naphtha liquid is vaporized (and super heated), it was translated into SYMBØL by a reactor, the underlying reaction being

naphtha(ℓ) -> naphtha(g)

or component number 11 -> component number 8.

The extent of reaction is the fraction vaporized.

R2 : Reformer reactor (REACO1,NE=10)

R3 : High Temperature Water Gas Shift reactor(REACO2,NE=12)

R4 : Low Temperature Water Gas Shift reactor (REACO2,NE=17)

R5 : Methanator reactor (REACO3,NE=29)

R6 and R7 : represent the combustion of fuel in the reformer reactor respectively, which is a reaction that occurs with no absolute contact with the main reforming process stream. However, the amount of fuel required in the auxillary burner is determined by the steam requirements for the process; any make-up of steam must be produced in the auxillary boiler, since the plant is operated to be self-sufficient in steam. Similarly, the amount of fuel required in the reformer furnace is determined by the heat load required in the reformer reactor. In that respect, there is an interaction between the fuel used in both the reformer furnace and the auxillary burner and the main reforming process stream but that interaction relies on the need for heat in the main reforming process but no mix-up of material occurs.

The following SYMBØL units represent the cooler-condensers (EXHO2, models) where part of the incoming water vapour of the product gases is condensed and discarded in a subsequent water separator (SEPAO1 models).

R8 and S1 (EXCHO2,NE=19) and (SEPAO1,NE=20)

R9 and S2 (EXCHO2,NE=21) and (SEPAO1,NE=22)

R10 and S3 (EXCHO2,NE=23) and (SEPAO1,NE=24)

R12 and S5 (EXCHO2,NE=27) and (SEPAO1,NE=28)

In these pairs of SYMBØL units, the water is first separated in the separators (corresponding to SEPAO1 models) and then the

condensation of that water (occurring in the EXCHO2 models) is translated to the correct position in the stream information vector by a reactor. Note that it was not really necessary to actually condense that water with a reactor since it is discarded any way in the actual plant and it is not used anymore in the process so that it does not really matter if the discarded water is condensed or not in the SYMBØL simulation. Note that the input stream to each of these "condensing water" reactors has no equivalent stream in the GEMCS information flow diagram. As a result, the coefficients required for these SYMBØL reactors (stoichiometric coefficients and extent of reaction) cannot be calculated from GEMCS stream information and therefore must be specified. This scheme will demonstrate, however, how these values are read in and stored in the AX(-) vector.

S4 and R11 represent the absorber (ABSO1,NE=26) where once again a change of phase occurs, that is some water is condensed (and then is discarded as above, so that the condensation "reaction" does not need to be translated by a reactor)

S6 Recycle splitter (JUNCO1,NE=31)

M1 and M4 Mixers (JUNCO1,NE=7 and 25)

M2 represent the controller (CØNTO2,NE=9) that adds the water (steam) to the reactant naphtha stream before it enters the reformer reactor.

M3 represents a possible control action, which is modeled in EXCHO4(NE=5), water (steam) will only be added to the reactant gas mixture if its temperature were to exceed some specified

maximum temperature.

M5 and M6 are used to mix the air, fuel and atomizing steam before it enters the reforming reactor since only one stream can enter a SYMBØL reactor.

M7 adds the flue gas from the reformer furnace and from the auxillary burner together. This is the particular control action which is modelled in EXCHO4(NE=5), which occurs if the flue gas from the reformer is inadequate to raise the reactant gas mixture temperature to some desired value.

5.3 Equivalent Stream and Stream codes

With the new system and an additional stream code (=4) to cause streams to be stored in the SN table, stream code 2 can be used exclusively for true recycle (cut) streams. The GEMCS stream code list in the combined system should be constructed accordingly. This ensures that only that information which is definitely required by GEMCS (stream codes 1 and 2) is transferred from SYMBØL. Unless computer core storage requirements dictate otherwise, it is suggested that all equivalent streams be labelled with stream code 4 (and thus the equivalence set in the SN(IS,2) position) even if they are not needed in the calculation of coefficients. This procedure should prevent any problems associated with the calculation of coefficients, particularly if changes in other parts of the data set are made at a later date. Table 5.1 indicates the implementation of this procedure for the problem at hand by showing the list of all possible GEMCS equivalent streams of the SYMBØL streams along with the appropriate stream codes used in the self-standing modular simulation; it shows also

the actual GEMCS streams that are made equivalent to the SYMBØL streams and their respective stream codes in the combined executive.

There are a number of points worthwhile noting in this case which relate to the modification of the GEMCS information diagram and the construction of the SYMBØL information diagram for use in the combined system and to the equivalencing of streams.

(i) GEMCS streams 7, 35, 63, 72 and 71 are feed streams and hence their stream codes remain the same as originally set; GEMCS stream 3 becomes a stream with code=1 (in the self-standing GEMCS simultaion, its code was equal to 2), since it is now a true feed stream whose component flows will be determined by SYMBØL (instead of CONTO3 in the self standing modular simulation).

(ii) only GEMCS streams 5, 44, 46 and 73 (from the stream list of Table 5.2) are recycle (cut) streams in the specified GEMCS calculation order; therefore their stream codes are set equal to 2.

(iii) GEMCS streams 5 and 8 are required to be stored in the SN table since GEMCS requires them to be stored as a result of its specified calculation order. GEMCS streams 22, 29, 34, 37 and 43 (corresponding to SYMBØL streams 20, 23, 26, 30 and 33, respectively) were product streams from the separators (SEPAO1 models, NE=20, 22, 24, and 28) and from the absorber (ABSO1 model, NE=26) and as a result, were given a stream code of 3. (Note that one could have given them a stream code of 6 : once they are calculated, these product streams are not used anymore in the plant. However, since they were given a stream code of 3 initially, their stream codes were not changed in an attempt to minimize the changes in the data set). Since all the equivalent GEMCS streams of SYMBØL streams 13, 14, 20, 23, 26, 30, and 33 were already saved in the SN table, their equivalence

Table 5.1 Equivalent streams

SYMBØL number	GEMCS self-standing		GEMCS in the combined executive	
	equivalent stream number	stream code	stream mode equivalent	stream codes
1	3	2	3	1
2	7	1	7	1
3	69	3	69	3
4	35	1	35	1
5	62	2	62	4
6	81	2	81	4
7	63	1	63	1
8	72	1	72	1
9	71	1	71	1
10	70	3	70	3
11	4	2	4	4
12	48,59	2,2	48	4
13	5,6	2,7	5	2
14	8	2	8	4
15	9	7	9	4
16	10,11	7,7	10	4
17	12,13,14 15,16	2,7,2,2,7	12	4
18	17	7	17	4
19	-nil-	-	-	-
20	22	3	22	3
21	21	7	21	4
22	-nil-	-	-	-

Table 5.1 Equivalent streams (cont'd)

SYMBØL number	GEMCS self-standing		GEMCS in the combined executive	
	equivalent stream number	stream code	stream mode equivalent	stream codes
23	29	3	29	3
24	28	7	28	4
25	-nil-	-	-	-
26	34	3	23	3
27	33	7	33	4
28	36	7	36	4
29	-nil-	-	-	-
30	37	3	37	3
31	38	7	38	4
32	-nil-	-	-	-
33	43	3	43	3
34	42,44,45	2,2,7	44	2
35	46,47	2,7	46	2
36	49	2	49	4
37	-nil-	-	82	3
38	73	2	73	2
39	-nil-	-	83	4
40	-nil-	-	84	4
41	74,75,76 77,78	7,2,7 7,3	74	4

was set even though they are not required in the calculation of SYMBØL coefficients.

(iv) For the fuel/flue gas system, it would be possible to calculate, manually, the values of the stoichiometric coefficients and to supply them via the AX(-) vector. It is easier, however, to have them calculated in the program. Because of the constraint of a SYMBØL reactor (only one input stream), it was necessary to create a stream in GEMCS which contained the fuel (stream 81), air (stream 62), steam (stream 63) mixture (stream 82) to the reformer furnace (via JUNCO1, NE=42) which then was equivalent to SYMBØL stream 37. The stoichiometric coefficients for the SYMBØL reactor R6 were then calculated from stream 82 and stream 73 of the GEMCS simulation.

Similarly the fuel (stream 71), air (stream 73) and atomizing steam (stream 70) to the auxillary burner were mixed to produce stream 83, which is equivalent to SYMBØL stream 39. In addition, the component flowrates in the flue gas stream from the auxillary burner were required to calculate the stoichiometric coefficients in SYMBØL reactor R7. A stream equivalent to this was created through another GEMCS module, CØNTO7,which determined the difference between GEMCS stream 73 (flue gas from reformer furnace) and stream 74 (combined flue gas from reformer and auxillary burner) and produced stream 84, the GEMCS stream equivalent to SYMBØL stream 40 (the flue gas from the auxillary burner). The calculation of the stoichiometric coefficients it thus made possible.

The auxillary boiler was not modelled in SYMBØL since it was self-standing in the GEMCS simulation and is really only required to be calculated after the plant simulation is converged. Although its operation is indirectly determined by the main plant operation (its firing rate determined by the difference between the process steam requirements and the steam

generation rate in the process), it does not interact directly with the hydrogen production of the plant.

5.4 Data Set for GEMCS

In the self-standing GEMCS simulation, 31 streams were initially read in; in the combined simulation, 53 streams were needed to be read in to record the respective equivalent SYMBØL stream in position SN(IS,2). Note again that for GEMCS streams 22, 29, 34, 37, 43 and 49, it was not necessary to specify their equivalent SYMBØL streams (20, 23, 26, 30, 33 and 36, respectively) since these streams are not called by SYMBØL to calculate coefficients, In this case, SYMBØL streams 20, 23, 26, 30 and 33 are the output streams of reactors R8 to R12 respectively; there are no equivalent GEMCS streams for the input streams to these reactors and, as a result, the stoichiometric coefficients and the conversion of any component must be supplied, they cannot be calculated from GEMCS stream information. Similarly, SYMBØL stream 36 is the production; it is not required to set it equivalent to GEMCS stream 49 since the coefficients for separator (splitter) S6 (JUNCO1, NE=31) can be calculated using the input stream to the separator and only one of its outputs, recycle stream 12 (GEMCS stream 48). However, since they were already saved in the SN matrix (stream code 3), the equivalent streams were specified and read in to permit subsequent check on these streams. Otherwise, only 47 streams would be required to be read in initially.

It is worth mentioning again that for the extra 22 streams read in in the combined executive (compared with GEMCS alone), only SN(IS,1)

(= GEMCS stream number) and SN(IS,2)(= equivalent SYMBØL stream) needs to be specified, the remaining of the SN(IS,-) matrix can be left as blank (or zeroes) on the cards required in the data set.

The changes in stream codes have been indicated in Table 5.2 JUNCOl models (NE=42,43) and CØNTO7 (NE=44) were added in the calculation order list along with their respective equipment EN vectors as required in the combined system for the reasons already stated.

It can be seen that the GEMCS data set remains almost the same although the calculation order list may require some modifications; the stream code list requires a major modification resulting from the storage of more streams that must be saved in the SN matrix (whose dimensions must be increased to accomodate the extra streams saved); and additional stream information and perhaps additional equipment EN vectors must be inserted while some already existing EN vector must be removed (some of the controller models). Only the EN vectors of CVGP1 and/or CØNTL3 may require updating to account for any change in the calculation order or the number of modules in the list.

5.5 SYMBØL Data Set

The complete SYMBØL data set (for the combined executive) is listed with some useful comments in Table 5.2. The basic system data are as follows:

Problem number 100 (NPRØB), Case number 1 (NCASE)

41 streams (NST), 11 components (NCP), 10 sources (NSCE), 7 mixers (NMIX), 6 separators (NSEP), 0 cascade (NCSD), and 12 reactors (NRTR) which fill the nine first positions of the KK(-) vector. Note that in the combined executive, there is no title card read in with the SYMBØL data set

Table 5.2 SYMBØL Data set of the Naphtha Reformer

Plant Simulation for the Combined Executive

Equipment	Data Set [1]	Remarks
	100 1 41 11 10 7 6 0 12	Basic system data
	NSCE = 10	
S1	1 1	
S2	3 2	
	1 1 11	
S3	2 3	
	12	
S4	2 4	
	7	
S5	3 5	
	7 6 10	
S6	3 6	
	10 1 11	
S7	3 7	source 7 is a mode 3
	1 6 10	ratio of atomizing steam (=component 1)
S8	3 8	in stream 7 to the fuel (=component 10)
	7 9 10	in stream 6 will be calculated from
S9	3 9	GEMCS stream information
	10 1 11	
S10	3 10	
	1 9 10	
	NMIX = 7	
M1	2 13 11 12	
M2	2 14 2 13	
M3	2 15 3 14	
M4	2 28 4 27	
M5	3 37 5 6 7	
M6	3 39 8 9 10	
M7	2 41 38 40	
	NSEP = 6	
S1	4 18 19 21	
	21	
S2	4 21 22 24	
	24	
S3	4 24 25 27	
	27	
S4	4 28 29 31	separator S4 is a mode 4; the fraction of each
	31	component of the input stream 28 which
S5	4 31 32 34	goes to the output stream 31 will be
	34	calculated from GEMCS stream information.
S6	5 35 36 12	
	12	
	NRTP = 12	
R1	2 1 11	
	11	
R2	2 15 16	
	8	
R3	2 16 17	
	3	
R4	2 17 18	reactor R4 is a mode 2
	3	the fractional conversion of Carbon Monoxide
R5	2 34 35	(=component 3) will be calculated from GEMCS
	2	stream information
R6	2 37 38	
	10	
R7	2 39 40	
	10	
R8	2 19 20	

Table 5.2 (cont'd)

```
                1                                                    fractional conversion of component 1
                1.                                                          in reactor R8
                1.  0.  0.  0.  0.  0.  0.  0.  -1.  0.  0.  ---- stoichiometric coefficients
R9              2  22  23                                              (must specify these coefficients
                                                                       since SYMBØL stream 19 has no
                1                                                      equivalent stream in the GEMCS
                1.                                                     simulation)
                1.  0.  0.  0.  0.  0.  0.  0.  -1.  0.  0.
R10             2  25  26

                1
                1.
                1.  0.  0.  0.  0.  0.  0.  0.  -1.  0.  0.
R11             2  29  30

                1
                1.
                1.  0.  0.  0.  0.  0.  0.  0.  -1.  0.  0.
R12             2  32  33

                1
                1.
                1.  0.  0.  0.  0.  0.  0.  0.  -1.  0.  0.

KEXTRA =     1

36  5  0  0
536.90178    ---total production of hydrogen must be 536.90178 kg/hr.
1               KK(11)
18.  44.  28.  16.  2.  28.  32.  80.576  18.  1.  80.576  ---Vector of Molecular
                                                              weights AMW(I)[2].
****** LLL =    61  ******   61 coefficients have
                            been specified
1
1

TOLEPX =    1.000E-04
```

---

1 - Anything underlined in this table was written (and calculated in the case of KEXTRA and LLL) by subroutine DATAIN to facilitate the checking of input information.

2 - For the vector of molecular weight, note that component 10(fuel) has a molecular weight of 1. Since the modular GEMCS simulation uses moles unit for all component except for the fuel which is always used in mass unit (kg/hr), a 'pseudo' molecular weight of fuel is specified as 1.

(as it use to be the case in the self-standing SYMBØL executive) since there are already two title cards read in at the beginning of the GEMCS data set (in the combined executive) where the user can fully define the simulated problem under investigation.

5.5.1 Sources

For all sources, a GEMCS equivalent stream was available so that all required SYMBØL coefficients were calculated from GEMCS stream information, none needed to be supplied. As a result, after all the information about sources has been read in, the AX(-) vector is empty.

S1 : source stream 1 (naphtha feed) was made a mode 1 in order to specify the extra constraint "hydrogen production" that must be set in stream 36.

S2 : source stream 2 represents the steam added to the naphtha feed. Since the amount of steam added is in constant ratio with the amount of naphtha, (set by plant operation), it was given a mode 3, in order to specify the ratio steam in S2 to naphtha in S1.

S3 : steam added to cool the reactant gas mixture if its temperature is too high. During normal (base case) plant operation, there is usually no steam added to cool the gas mixture. As a result, this source was made a mode 2 where the total flow can be specified. However, the equivalent GEMCS stream should show the composition of this stream but its total flow should be specified as zero when the stream is read in initially in the GEMCS data set.

S4 : small amount of air added to the process. The flowrate of this stream is constant and independent of other streams in the process (set by operators to maintain the activity of the absorbant in the absorbers). It can be given a MØDE = 2 since its composition and flowrate can be specified, that is taken directly from the equivalent GEMCS stream (in this case, from the oxygen flowrate).

S5 (and S9) represents the fuel to the reformer furnace (auxillary burner); it was also given a mode 3 in order to specify the ratio fuel in S6 (S9) to the naphtha feed in S1. Note that in fact, the amount of fuel is determined by the amount of heat required in the reformer furnace for the reforming reaction (or in the auxillary burner); however, SYMBØL is a material balance program and, as such, cannot handle a constraint base on a heat requirement. For that reason, the amount of fuel required is specified here in a ratio fuel-to-naphtha feed, that is calculated from their GEMCS equivalent streams.

S7 (and S10) represents the atomizing steam added to the fuel; it was also made a mode 3 in order to specify the ratio steam in S7 (S10) to fuel in S6 (S9).

5.5.2 Mixers

No coefficients need to be supplied for any mixer so that it does not matter if equivalent GEMCS streams exist or not for these units; as a

result, the AX(-) vector is still empty when all the information for mixers is read in.

5.5.3 Separators

Only one of the output streams for each of the separators S1 to S5 has an equivalent GEMCS stream. This satisfies the requirement for calculating the coefficients for them since only the composition of the input and one output stream is required in this calculation. The system recognizes that any fraction of any component that does not go from the input to that output goes to the other output stream. This output must have an equivalent stream saves in the SN table which is the case here. Separator S6 has an equivalent GEMCS stream for its input and for each of its output streams. As a result, when the system has just finished reading in the data for all separators, the AX(-) vector is still empty since all coefficients can be calculated.

Separators S1 to S5 were made a mode 4 to specify that the fraction of each component of the input stream which goes to the specified output stream will be supplied later by calculation from equivalent GEMCS streams. S6 was made a mode 5 since it was a splitter (the fraction of the whole input stream which goes to the specified output stream must be supplied in a mode 5).

There are no cascade in this information flow diagram so that the next data read are the information about reactors.

5.5.4 Reactors

All reactors were given a mode 2 as it was desired to specify the

fractional conversion of a specified component. For reactors R1 to R7, all their input and output streams had each an equivalent GEMCS stream. As a result, all stoichiometric coefficients for these reactors, along with the fractional conversion of a specified component could be calculated from the GEMCS stream information.

For reactors R8 to R12, however, all their input streams have no equivalent stream in GEMCS. As a result, the stoichiometric coefficients and the fractional conversion must be supplied through the use of the AX(-) vector as shown below: the reaction occuring in these five reactors is simply a change of phase where water vapour is condensed:

$H_2O(v) \rightarrow H_2O(\ell)$

or component number 1 -> component number 9

with stoichiometric coefficient + 1 for $H_2O(v)$ and -1 for $H_2O(\ell)$ and the extent of reaction is 1. since the water which is separated in the separators upstream of these reactors is determined from the amount condensed (remembering that SYMBØL is a mass balance program, a change of phase cannot occur in a SYMBØL separator, but one can get around it by supplying a SYMBØL reactor, in which the change of phase can be accounted for).

5.5.5 Extra Constraints and Additional Information for the Combined Executive

All separators and reactors have no mode 1; only one source has a mode 1. As a result, only one additional constraint is required to be specified. It is desired that SYMBØL calculates all feed and cut stream flows to meet a certain hydrogen production rate; that production rate is the desired extra-constraint.

Since the modular simulation uses mole units for its component flows, KK(11) is set equal to 1 and the vector of molecular weight must be specified, containing the molecular weights of all components in the same order as in the stream information vector.

Finally, it was desired to test (for convergence) one stream only in CØNTL4 (thus KK(12)=1); that stream is the naphtha feed to the process and is identified as stream number 1 (thus KK(13)=1). The tolerance criterion (TOLERX) for convergence was set to 0.0001.

5.6 Comparison Between the Combined Executive with the Self-Standing GEMCS Simulation

Once a base case simulation was achieved with the combined system, it was decided to test its ability to achieve a simulation under other conditions. Although the equipment parameters in the GEMCS system could be changed and a simulation could have been obtained using both systems, it is best to effect a comparison of the two systems by starting with the same stream information in the recycle streams and try to meet a new constraint with the same operating (equipment) parameters in the GEMCS system. A comparison of the results of two test cases are shown in Table 5.3. Although these results do indicate that the desired simulation can be achieved in the combined system, they also indicate that the computational time is more than twice for the combined system. This demonstrates the superior power of the convergence promoter (CVGP1) over that of the combined system. It must be noted, however, that the real power of the combined system has not been tested since the production rate of hydrogen is approximately linear with the naphtha feed flowrate. It is believed

Table 5.3 Comparison Between the Combined Executive with the Self-Standing GEMCS simulation

	90% production	80% production
1. Number of iterations		
(a) self-standing GEMCS	24	26
(b) GEMCS in combined executive	13 + 14	13 + 16 + 4
2. CPU time (seconds)		
(a) self-standing GEMCS	23.55	25.22
(b) GEMCS in combined executive	53.50	72.16
3. Hydrogen production rate (moles/hr)		
(a) self-standing GEMCS	241.60580	214.76071
(b) GEMCS in combined executive	241.58427	214.76128
(c) desired production	241.60580	214.76071

that where simple control models can be included to meet constraints in a simulation, they should be used since CVGP1 will bring about a solution faster than one that iterates between SYMBØL and GEMCS, even if the time for swapping the overlays were not included (more model evaluations required in the combined system). On the other hand, in cases where simple control models may not be written to meet the constraints as in the Alcan Bayer plant simulation, the combined system will provide a solution whereas the GEMCS system will fail. Hence, the real power of the combined system will only be tested and demonstrated under that condition.

It should be noted in the example that if the test for convergence is not on the constrained flow, the solution will not be as accurate unless the tolerance is made smaller. Testing on the feed stream as was done in the test cases, does not give the same tolerance on the production because of the multiplying factor, production rate-to-feed.

The GEMCS part of the data set and the converged solution of the Naphtha Reformer Plant simulation (base case data) using the Combined GEMCS-SYMBØL executive is given in Appendix F.

PART II

THE STEADY-STATE SIMULATION OF A NAPHTHA REFORMER PLANT USING THE GEMCS MODULAR SIMULATION ALONE

6. THE STEADY-SIMULATION OF A NAPHTHA REFORMER PLANT

6.1 Introduction

The object of this steady-state simulation of an existing reformer plant is to provide a test case for the combined executive system as developed in Part I. It was produced specifically for this study as an educational experience for the writer to ensure that all details of the modular GEMCS executive and unit computations were known and understood and to ensure that the requirements for the combined system were well defined at the outset. Moreover, it provided a valuable and unique experience in applying the algorithm, as suggested by Ripps (R2), to correct plant data which had gross experimental errors:. to the writer's knowledge such an experience with real plant data has not been reported in the literature.

The technology of the modular approach to steady state simulation technology has been well-documented in a text by Crowe et. al., (C3); the details of the GEMCS executive system are available in a User's Manual (J1). Basically, simulation involves the translation of the process flow sheet into an information flow diagram in which the information concerning the flow of material and the thermodynamic state of any stream (called the SN list) is transferred from one unit computation or module to another. All incoming information to the module is assumed known and this incoming information is modified in the unit computation; the outgoing information reflects these changes and becomes the input information for the next module. The unit computation must describe mathematically the physical and chemical processes occurring in the actual physical units. Other unit computations will be included on the information flow diagram to describe control processes within the plant and the actual operating procedures, and also to aid in

mathematical operations required in effecting the overall mathematical simulation (for example, convergence promotion on an unknown recycle stream).

An overall strategy in obtaining a process simulation is depicted in Figure 6.1. This strategy was followed fairly closely in the present case. The specific objectives for the present simulation are best discussed after the processing details are known.

6.2 General

The naphtha reforming process in this study is used to produce hydrogen which, in turn, is normally used for the ammonia production. Basically, the process involves reacting naphtha (a mixture of hydrocarbons represented by a fictitious formula C_nH_{Zn} with n=5.6488 and Zn=12.789 with steam over a nickel-supported catalyst according to the reaction:

$$C_nH_{Zn} + (n-x)H_2O \rightleftharpoons (n-x)CO + \left(\frac{Zn-4x}{2} + n - x\right) H_2 + x\ CH_4 \qquad (6.1)$$

This reaction is written to indicate that almost all of the unreacted hydrocarbon in the exit gas exists as methane. This chemical reaction greatly oversimplifies the actual chemical reactions which occur as a result of the thermal and catalytic cracking of naphtha.

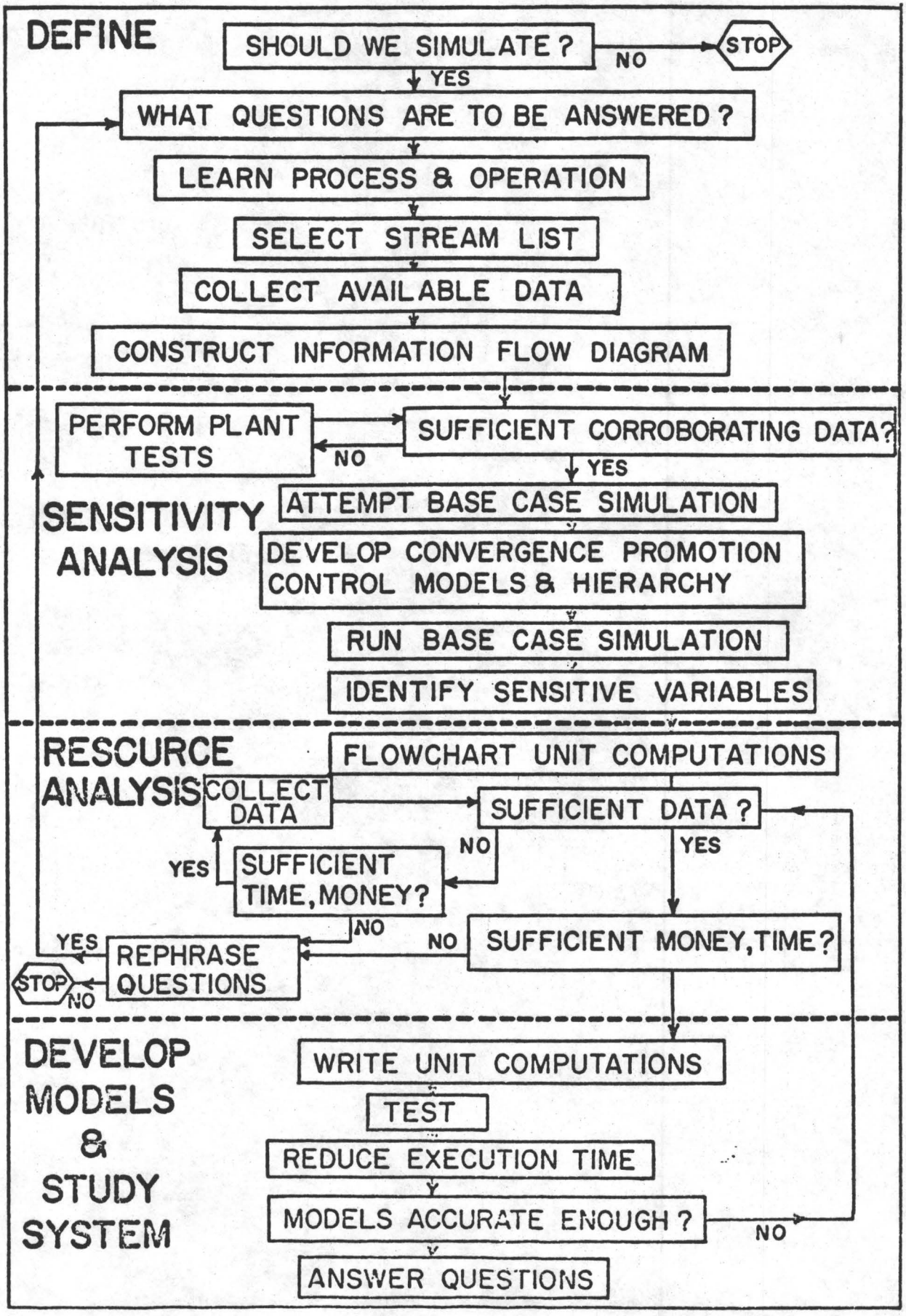
Methane and steam react according to

$$CH_4 + H_2O \rightleftharpoons CO + 3H_2 \qquad (6.2)$$

Both of these reactions are strongly endothermic and hence the equilibrium favours the production of hydrogen at high temperatures.

In addition to these reactions, the water gas shift reaction

Figure 6.1 Overall Strategy for Solving Simulation Projects



(equation 6.3) occurs to some extent in the reformer reactor:

$$CO + H_2O \leftrightarrow CO_2 + H_2 \qquad (6.3)$$

This reaction is strongly exothermic and hence favours the production of hydrogen at low temperatures. For this reason, additional shift reactors are included downstream of the reformer in an attempt to maximize the production of hydrogen for a given feed.

6.3 Process Description

The process flowsheet for the plant under consideration is shown in Figure 6.2. The reformer reactor consists of vertical tubes, packed with the nickel-supported catalyst pellets, through which the naphtha-steam mixture flows. The kg-moles of steam/kg-atoms of carbon in the feed and the operating pressure are process operating variables. Note that increased steam favours the production of hydrogen whereas increased pressures decreases it in reaction 6.1 and 6.2. The plant in question operates the reactor at about 41 atmospheres to ensure an outlet pressure from the plant of about 33 atmosphere to allow direct flow of hydrogen to the ammonia plant without the use of intermediate compression. The steam/carbon ratio is about 9 to offset the high pressure operation. These operating conditions are not typical of other reforming processes.

The reactant gases enter the reactor at about 510-520^{o}C and leave the reactor at a controlled temperature of about 810^{o}C. Heat is supplied to a vertical furnace surrounding the tubes by burning a low sulphur Bunker-C oil in multiple burners located on the wall of the furnace. Control is

established by adjusting the fuel to ensure a desired outlet temperature of the reformed gases; however, the furnace may not be fired at a rate that would allow the tube wall temperature to exceed about 950°C. At temperatures in excess of this, carbon deposition on the catalyst begins to be important as well as thermal-mechanical and corrosion problems. The rest of the process is designed to recover energy from the reaction product and flue gas streams, to maximize hydrogen production, and to remove or minimize impurities in the product hydrogen stream.

The flue gas leaves the furnace at about 850°C. In turn, it is used to preheat the steam-naphtha reactant to the desired temperature of 518°C, to produce steam at 48.4 atmospheres in a waste-heat boiler, to reheat a hydrogen recycle stream (which is required in the hydrodesulphurization reactor to remove sulphur from the naphtha feed), to vaporize and super heat the naphtha feed before it enters the hydrodesulfurizers, and finally to preheat the air required by the reformer burners.

The gaseous reaction products from the reformer are cooled to about 435°C in a waste heat boiler before they are fed to the high temperature, adiabatic, water gas shift reactor, containing an iron catalyst, where more hydrogen is produced according to reaction 6.3. Although the literature suggests that the remaining hydrocarbon should be reacted as well, this is not the experience in this plant, because, at the operating temperature of this reactor (exit temperature 445°C), the equilibrium constant for reaction 6.2 is very low.

The reaction products from this reactor are cooled successively by the feed stream to a methanator reactor downstream, a waste heat boiler, and two economizers used to preheat boiler feed water for the waste heat boilers.

Figure 6.2 Process Flowsheet of the Naphtha Reformer Plant

Symbol	Meaning
△	stream number where plant measurements were available (listed in Table 6.)
□	pressure in atmospheres
○	temperature (^{o}C)
▱	flows in kg/hr
⬭	flows in Nm^3/hr

ABSORB/DESORB. - ABSORBER/DESORBER

B.F.W. - Boiler Feed Water

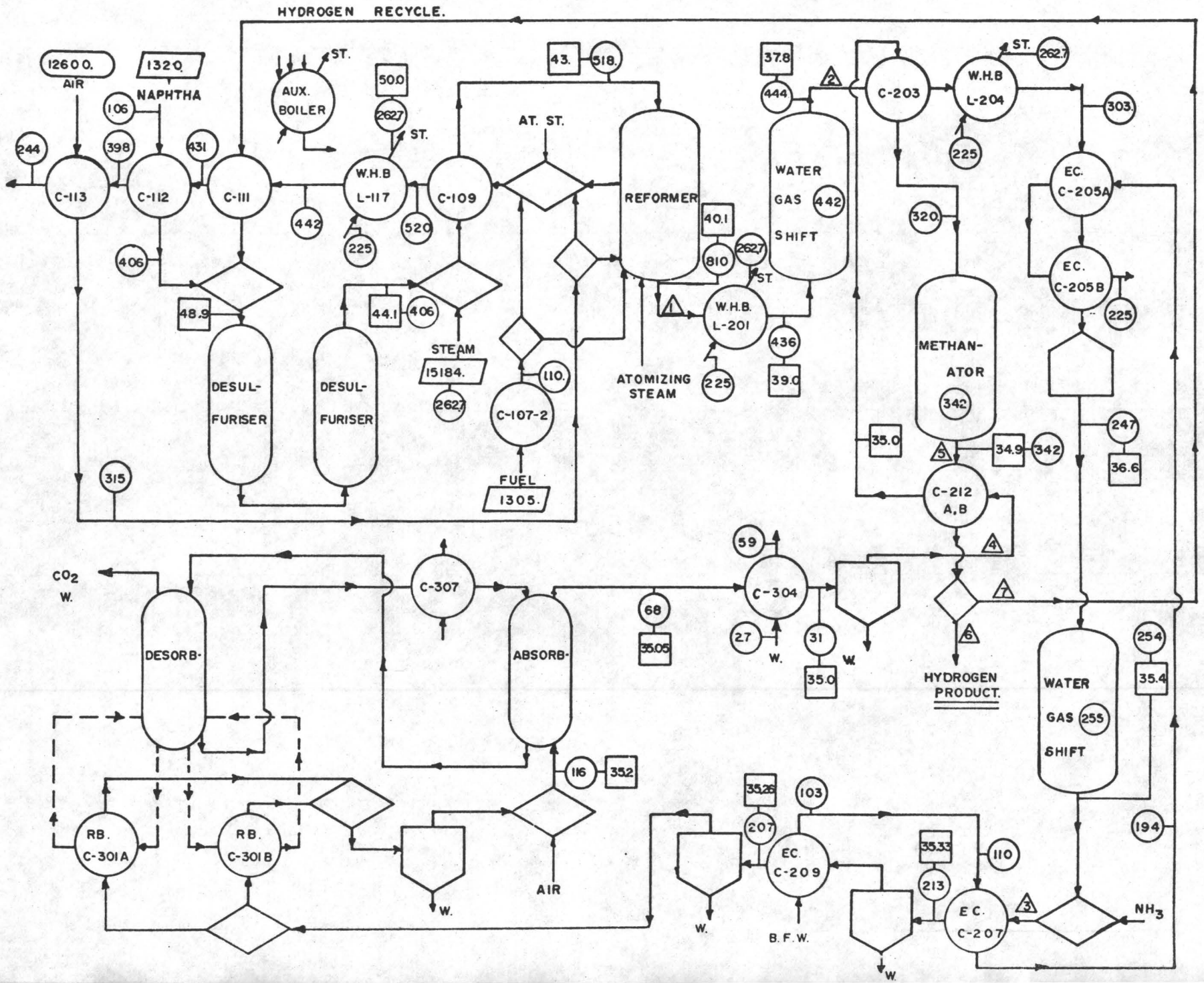
EC - Economizer

RB - Reboiler

St - steam

W. - water

W.H.B. - waste Heat Boiler


HYDROGEN RECYCLE.
12600.
AIR
1320.
NAPHTHA
106
AUX. BOILER
ST.
500
2627
ST.
43.
518.
37.8
444
C-203
W.H.B L-204
ST.
262.7
303.
225
244
398
431
C-113
C-112
C-111
442
W.H.B L-117
225
520
C-109
AT. ST.
REFORMER
40.1
810
262.7
ST.
WATER GAS SHIFT
442
320.
EC. C-205A
EC. C-205B
225
406
48.9
44.1
406
DESUL-FURISER
DESUL-FURISER
STEAM
15184.
2627
W.H.B. L-201
436
39.0
ATOMIZING STEAM
225
METHAN-ATOR
342
35.0
34.9
342
247
36.6
315
110.
C-107-2
FUEL
1305.
C-212 A,B
CO_2
W.
DESORB-
C-307
ABSORB-
68
35.05
59
C-304
27
W.
31
W.
35.0
HYDROGEN PRODUCT.
WATER GAS SHIFT
255
254
35.4
194
RB. C-301A
RB. C-301B
W.
116
35.2
AIR
35.26
207
103
EC. C-209
B.F.W.
35.33
213
110
EC. C-207
NH_3
W

This cooled stream is sent to an adiabatic low-temperature shift reactor containing a copper-supported catalyst. The object is to convert as much of the carbon monoxide as possible since any remaining carbon oxides must be converted to methane in order that the hydrogen may be used directly in an ammonia converter; otherwise the carbon oxides would poison the catalyst. Moreover, the methane concentration in the product stream should not be above about 1.8 mole percent or excessive purge is required in the ammonia plant.

A small amount of ammonia is mixed with the product gases from this reactor to neutralize the acidity of the condensed water (carbon dioxide absorption) downstream. The neutralization is required only as a corrosion control measure.

The product gases are cooled by boiler feed water in two economizers. At these temperatures, a large fraction of residual steam condenses. A water separator is provided immediately downstream of each of these cooler-condensers.

Most of the carbon dioxide is removed from the gas stream by countercurrent contacting with hot potassium carbonate solution containing arsenic oxides as promoters (Vetracoke process) in a tower packed with Interlox Saddles. The solvent is generated in a similar reboiler-desorber where the carbon dioxide is removed according to the reverse reaction:

$$CO_2 + CO_3^{=} + H_2O \;\underset{}{\overset{}{<=>}}\; 2\ HCO_3^{-} \qquad (6.4)$$

The product gases from the economizers provide the heat for the reboiled desorber and hence are cooled to about 130°C before entering the absorption tower. Condensed water is again removed in a separator downstream of the

reboilers. A small amount of air is added to the gas stream before the absorber to ensure a high oxidation state for the arsenic oxide promoter. The potassium carbonate solution is cooled in a water-cooled exchanger before it enters the absorber. Since the outlet temperature of the gases is lower than the inlet, some steam also condenses out of the product gases in the absorber; this condensate enters the absorbant.

The reactant gases are finally cooled to about 31°C in a water-cooled cooler-condenser where the water content is reduced to about 0.13 mole %. These gases are then preheated to about 320°C by an exchanger on the outlet of the methanator and the upstream exchanger before they are passed into an adiabatic methanator operating at an exit temperature of about 342°C where the following methanation reactions occur:

$$CO + 3H_2 \rightleftharpoons CH_4 + H_2O \qquad (6.5)$$

$$CO_2 + 4H_2 \rightleftharpoons CH_4 + 2H_2O \qquad (6.6)$$

The equilibrium and reactor dimensions are such that essentially total conversion to methane occurs. The product gases from the methanator preheat the reactant gases to it. The outlet temperature of the product gases is about 60°C and its pressure is about 34 atmosphere absolute.

6.4 Questions to be Answered by the Simulation

Typical of many processes, is the need to increase production of an existing plant. Some increase in productivity may already have been realized because of changing operating conditions or because the plant already

had been over-designed for the original capacity. At some point, however, constraints or bottlenecks show up which limit the capacity of the plant. At this point, there is a need to know what steps may be taken to increase production and the relative sensitivity of the production rate to these changes.

In the naphtha reforming process, increasing production means increasing the flowrate of naphtha feed. Experience indicates that the pressure drop across the naphtha vaporizer-super heater becomes excessively high so that the pressure of the hydrogen at the outlet (inlet to the ammonia plant) becomes so low that it adversely affects ammonia production. The question then is: How can the production rate be increased without decreasing the outlet pressure?

Since the questions posed for the simulation dictate the sophistication of the modeling of the units in the plant, it is worthwhile listing the overall requirements in the modeling of the units; these are suggested as follows:

(i) pressure drop across all units must be predicted under any operating condition.

(ii) The effects of changes in reactant flowrates must be accounted for.

(iii) The effect of any operating condition (control variable or result of control) or change in size or design variable must be taken into account in the modeling.

6.5 Stream Information Vector

The pertinent information which affects the production of the

process must be included in the stream information vector. It is important to include only that information which affects the production rate or the plant operation. If a variable is important in the actual operation of the plant but is known not to control the production rate of the existing system then it should not be included in the stream list or the modeling. For example, in the present plant, it is known that sulfur poisons the reformer catalyst. The hydrodesulfurization unit removes sulfur from the naphtha feedstock. It is shown, however, that this unit has been considerably over-designed and hence will not limit production. On the other hand, this unit does add pressure drop to the system. Therefore, the desulfurization reaction need not be modeled (and hence sulfur does not need to be included in the stream list), but the pressure drop across the desulfurization units should be taken into account at any production rate.

On this basis, the following stream information vector was considered to be important for the present simulation:

1. Stream number
2. Stream type (normally = 0)
3. Total flowrate
4. Temperature
5. Pressure (atmosphere)
6. Flowrate of water vapour
7. Flowrate of carbon dioxide
8. Flowrate of carbon monoxide
9. Flowrate of methane
10. Flowrate of hydrogen
11. Flowrate of nitrogen

12. Flowrate of oxygen

13. Flowrate of naphtha vapour

14. Flowrate of water as liquid

15. Flowrate of fuel as liquid

16. Flowrate of naphtha as liquid

The flowrates of all materials are in kilogram-moles/hour except the fuel which is taken as kilograms/hour. Note that the gaseous flowrates are grouped together (numbers 6 to 13) and the liquid flows are in number 14 to 16 inclusive. This order is useful in using the physical property package (Appendix C). The first five elements of the stream list are common in all GEMCS simulations.

7. THERMODYNAMICS AND PHYSICAL PROPERTY DATA

The requirements for physical property data relate to the need for modeling reactors and heat exchangers and for ascertaining the non-ideality of gaseous mixtures at the temperature and pressure conditions of the process.

7.1 Equilibrium Constants and Non-Ideality

The equilibrium constant for a reversible chemical reaction

$$aA + bB + \text{----} \rightleftarrows \ell L + mM + \text{----} \tag{7.1}$$

may be written, upon applying the 'Law of Mass action' for a gaseous heteregeneous or homogeneous reaction, thus:

$$K_a = \frac{(a_L)^{\ell}(a_M)^{m} \text{---}}{(a_A)^{a}(a_B)^{b} \text{---}} \tag{7.2}$$

where a_i is the activity of component i.

If the standard state for each substance involved in the reaction is that of the pure component at the given pressure (P) and temperature (T), then the change in Gibb's free energy (ΔF^{o}) is

$$\Delta F^{o} = -RT\ell nK_a \tag{7.3}$$

Equation 7.2 may be expressed in terms of fugacity and when the standard state is one unit of fugacity and when the reactants form an ideal gaseous solution:

$$\frac{(y_L f_L)^{\ell}(y_m f_m)^{m} \text{---}}{(y_A f_A)^{a}(y_B f_B)^{b} \text{---}} = K_f$$

that is the fugacity of the gas ($y_i f_i$) is equal to the activity of the gas as a result of the way the standard state is defined and $K_a = K_f$. For ideal gases

$$\frac{y_L^{\ell} \cdot y_M^{m} \text{ ---}}{y_A^{a} \cdot y_B^{b} \text{ ---}} \cdot p_T^{\Sigma n} = K_p \quad \text{where } \Sigma n = \ell+m+\text{---} -a-b \text{ ---} \tag{7.4}$$

Hence

$$\frac{K_f}{K_p} = K_\gamma = \frac{\gamma_L^{\ell} \gamma_M^{m} \text{ ---}}{\gamma_A^{a} \gamma_B^{b} \text{ ---}} \tag{7.5}$$

where $\gamma_i = f_i/p_i$ is defined as an activity coefficient. K_γ is independent of concentration but a function of temperature and pressure; it gives an indication of the extent of deviation from ideal behaviour and can be evaluated as a function of reduced properties.

In the present case, the pressure is about 40 atmospheres and the activity coefficients for all components, except perhaps water vapour, are very close to one. The difficulty associated with water vapour is that at the lowest temperature (reduced temperature $T_R < 0.6$) where non-ideality is suspected, extrapolation of existing fugacity charts is questionable. Therefore, it was assumed the system behaves ideally under all conditions.

The dependence of the equilibrium constant with temperature is given by

$$\frac{d \ \ell n \ K_p}{dT} = \frac{\Delta H_R^o}{RT^2} \tag{7.6}$$

where ΔH_R^o is the heat of reaction at a standard state given by

$$\Delta H_R^o = \ell H_{f,L}^o + m H_{f,M}^o + \text{---} - a H_{f,A}^o - b H_{f,B}^o \qquad (7.7)$$

where $H_{f,i}^o$ is the enthalpy of formation of each substance in the pure state at a given standard pressure and temperature.

The heat of reaction at any temperature may be calculated from the ideal enthalpies of formation (in k calories/kg mole) of the pure components involved in the reaction. These may be expressed as a function of temperature in a polynomial form for any component , viz.

$$H_{f,i} = a(i,0) + a(i,1)*T + a(i,2)*T^2 + a(i,3)*T^3 + a(i,4)*T^4 \qquad (7.8)$$

where the coefficients have been tabulated by Kjaer (K2) with T in degrees Kelvin (^oK).

The heat of reaction for any reaction (equation 7.7) is determined by the Lewis' Rule:

$$\Delta H_R = \ell H_{f,L} + m H_{f,M} + \text{---} - a H_{f,A} - b H_{f,B} - \text{---}$$

$$= \Delta a(i,0) + \Delta a(i,1)*T + \Delta a(i,2)*T^2 + \Delta a(i,3)*T^3 + \Delta a(i,4)*T^4 \qquad (7.9)$$

where $\Delta a(i,j) = \ell*a(i,j) + m*a(i,j) + \text{---} - a*a(i,j) - b*a(i,j)$ (7.10)

Thus the equilibrium constant may be obtained at any temperature by

integrating equation 7.6 to give

$$\ell n\ K_p\Big|_T - \ell n\ K_p\Big|_{T_{ref}} = \int_{T_{ref}} \frac{\Delta H}{RT^2} \cdot dT$$

$$= \frac{1}{R} * \left[-\Delta a(i,0)/_T + \Delta a(i,1)*\ell nT + \Delta a(i,2)*T \right.$$

$$\left. + \Delta a(i,3)*T^2/_2 + \Delta a(i,4)*T^3/_3 \right]_{T_{ref}}^{T} \qquad (7.11)$$

and $$K_p\Big|_{ref} = \exp\{-\Delta F_{ref}/RT_{ref}\} \qquad (7.12)$$

where ΔF_{ref} is obtained from the standard free energy of formation (kcal/kgmole) of the pure component given by

$$\Delta F_{ref} = \ell F_L + mF_M + \text{---} - aF_A - bF_B - \text{---} \qquad (7.13)$$

Kjaer (K2) tabulates these free energies at 25°C for a large number of components.

Equation 7.11 may be simplified for computing to

$$K_p = \exp\{\frac{1}{R} *[\Delta a(i,1)*\ell nT - \Delta a(i,0)/T + K_{p_{298.16}} - a(298.16)$$

$$+ \Delta a(i,2)*T + \Delta a(i,3)*T^2/_2 + \Delta a(i,4)*T^3/_3]\} \qquad (7.14)$$

The program for determining the equilibrium constants for the reactions involved in this process (subroutine EQK) is shown in Appendix B.

7.2 Heat capacity

The ideal heat capacities of the pure component gases may be evaluated from the ideal enthalpies by

$$C_{p_i} = \frac{dH_{fi}}{dT} = a(i,1) + 2*a(i,2)*T + 3*a(i,3)*T^2 + 4*a(i,4)*T^3 \qquad (7.15)$$

The heat capacity of any mixture is determined simply by weighting each component value according to its mole fraction. It was determined that the correction for non-ideality was negligible.

The heat capacity of liquid water was taken as constant at 18 kcal/kgmole. The heat capacity of naphtha liquid and vapour were provided as 35.5 and 28.045 kcal/kgmole, respectively, and assumed independent of temperature since this additional information was not available.

These physical property coefficients were incorporated into the physical property subroutines READIN, PRØCAL and PRØMIX which are described and listed in Appendix C.

7.3 Density

Since the correction for non-ideality was determined to be small, the ideal gas law was used to evaluate the density of any gas mixture.

7.4 Heat of Vaporization of Water

The heat of vaporization of water at any temperature was correlated

by Watson correlation (R1):

$$\Delta H_v = \Delta H_{v_1} \left(\frac{T_c - T}{T_c - T_1} \right)^n \tag{7.16}$$

where ΔH_{v_1} is the latent heat of vaporization (kcal/kgmole) at temperature T_1 (^oC) and T_c is the critical temperature (374.2^oC).

This method provided a correlation

$$\Delta H_v = \exp(0.33473 * \ell n(374.2-T)+7.30109) \tag{7.17}$$

which was accurate to better than 1% over the full range of temperatures (50 to 265^oC).

7.5 Vapour Pressure of Water

The vapour pressure of water was correlated by the Cox-Antoine relation:

$$\log P_v = A - B/(T+C) \tag{7.18}$$

where A = 7.96181, B=1668.21, C=228.0 for T in ^0C and P_v in mm.Hg. as determined by Holmes and Van Winkle (H2).

7.6 Heat of Vaporization of Naphtha

A typical analysis of the naphtha feedstock is provided in Appendix D. From this analysis the average molecular weight was determined to be 80.576 and its representative (fictitious) formula as $C_{5.6488}\ H_{12.789}$. The heat of vaporization was provided to be 85.3 kcal/kg. No effect of temperature was available so that this value was assumed to represent an average

value over the temperature range involved.

7.7 Vapour pressure of Naphtha

Vapour pressure versus temperature data were provided in Figure 7.1. Since the mixture has a wide boiling range, only the initial and final boiling points are provided. Note that there is some difference between the naphtha as supplied by Union Oil and Esso. Since the pressure at the naphtha vaporizer/super heater is about 48 atmospheres, the temperature range over which the vaporization occurs is 300 to 370°C (for the Union Oil naphtha). This detail is discussed further in section 8.5.3 for subroutine EXCH03.

7.8 Heat of Reaction of Naphtha

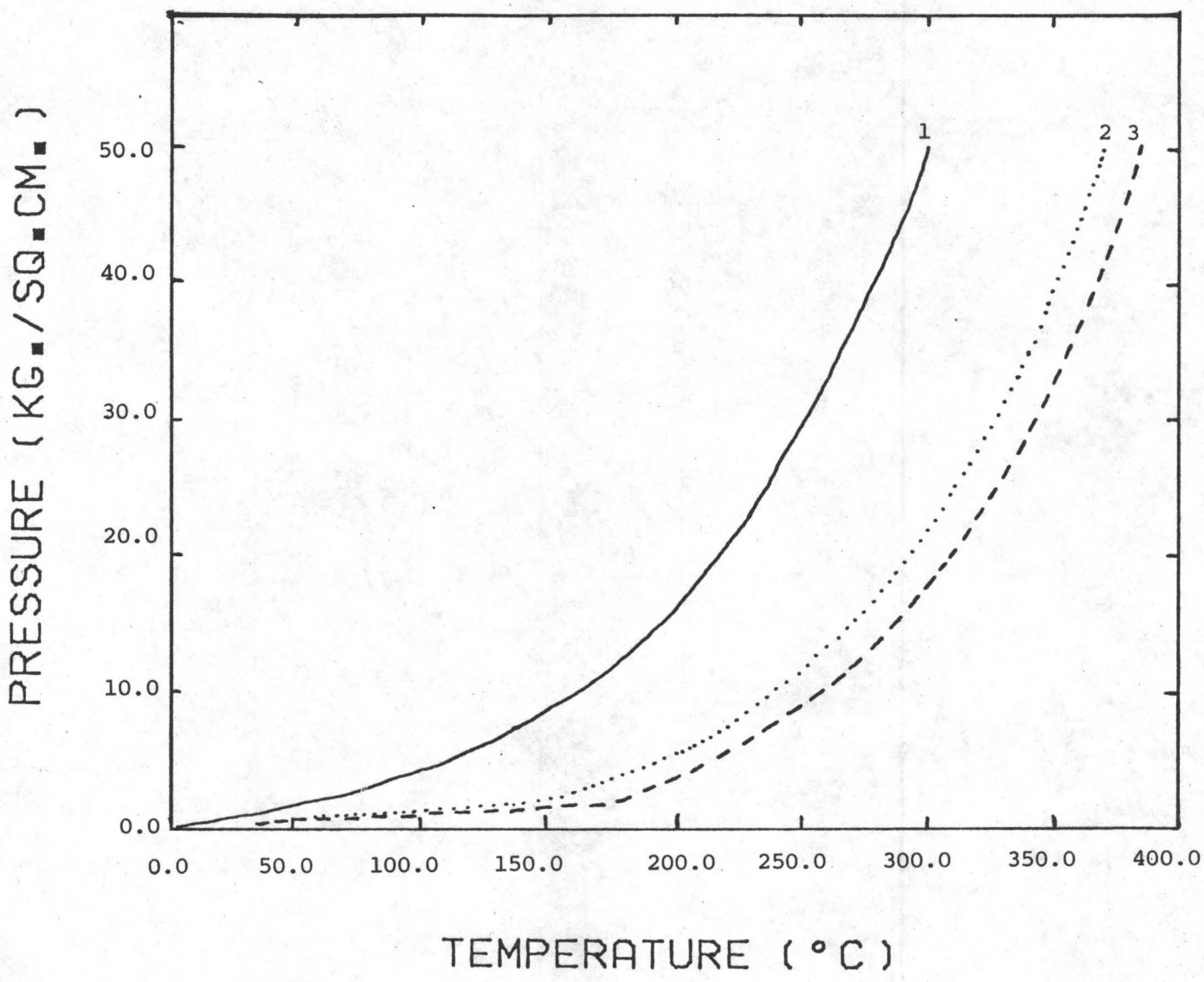
Heat of combustion of naphtha (higher heating value) was provided for naphtha as 11,500 kcal/kg which is assumed to apply at 25°C. This information can be used to calculate the heat of formation of naphtha at 25° from the chemical reaction

$$C_{5.6488}\ H_{12.579} + 8.84\ O_2 \rightarrow 5.6488\ CO_2 + 6.38\ H_2O_{(\ell)}$$

with $H_f(H_2O(\ell)) = -68.3174$ and $H_f(CO_2(g)) = -94.052$ kcal/g mole, the heat of formation of naphtha at 25°C is calculated to be -40.19 kcal/gmole. To use this information in calculating the heat of reaction at any temperature requires knowledge of how the heat of formation of naphtha changes with temperature. This is easily obtained from

$$H_f(T) = H_f(25) + \int_{25}^{T} C_p\, dT \qquad (7.19)$$

Figure 7.1 Naphtha Vapour Pressure Variation against Temperature - Initial and Final Distillate



1 - Initial Distillate of Naphtha from Union Oil

2 - Final Distillate of Naphtha from Union Oil

3 - Final Distillate of Naphtha from ESSO

In this case, the heat capacity of naphtha vapour is assumed to be constant at 28.052 kcal/kgmole ^{o}C. Therefore,

$$H_f(T) = -40190. + 28.052\ (T-25) \qquad (7.20)$$

For the reaction mixture (m moles of CH_4 recycled/mole of naphtha entering and x moles of CH_4 left/mole of naphtha entering and with all naphtha which enters reacting), the reaction may be represented by:

$$C_n H_{Zn} + mCH_4 + (n-x+m)H_2O \rightarrow (n-x+m)CO +$$

$$\left(\frac{Zn}{2} + 2m + (n-x) + m-2x\right)H_2 + xCH_4 \qquad (7.21)$$

The heat of reaction (ΔH_R) per mole of naphtha reacted is calculated from

$$\Delta H_R = (n-x+m)*(H_f(CO)-H_f(H_2O(g)) + (x-m)*H_f(CH_4)$$

$$+\left(\frac{Zn}{2} + 3m+n-3x\right)*H_f(H_2) - H_f(C_nH_{Zn}(v)) \qquad (7.22)$$

7.9 Heat of Combustion of Fuel

The higher heating value of the fuel was provided as 9800 kcal/kg. This was assumed to be the value at 25^{o}C. The composition of the fuel was given as a carbon-to-hydrogen mass ratio (C/H = 8.3).

8. PROCESS INFORMATION FLOW DIAGRAM AND ASSOCIATE MODULES

8.1 Process Information Flow Diagram

As indicated in section 6.1 the process flowsheet (Figure 6.2) must be transformed into an information flow diagram on which the streams and unit computations or models are indicated. The information flow diagram for the naphtha reforming process is shown on Figure 8.1 with the models for each unit listed on Table 8.1. Note that there are a number of control modules which reflect the control required in the process and aid in achieving the overall process simulation. The details of the models are described in the following sections. Chapter 9 indicates how the parameters in the models were obtained.

8.2 The Naphtha Reformer Reactor (REAC01)

This reactor model must account for the thermal and catalytic cracking of the naphtha with steam and the water gas shift reaction which occurs there; viz:

$$C_nH_{Zn} + (n-x)H_2O \rightarrow (n-x)CO + \left(\frac{Zn-4x}{2} + n-x\right) H_2 + x\ CH_4 \qquad (8.1)$$

where the x moles of methane remaining account for the incomplete cracking of the naphtha. This gives rise to another equilibrium reaction involving methane and steam.

$$CH_4 + H_2O \underset{\leftarrow}{\overset{K_1}{\rightarrow}} CO + 3H_2 \qquad (8.2)$$

Figure 8.1 GEMCS Information Flow Diagram of the Naphtha Reformer Plant

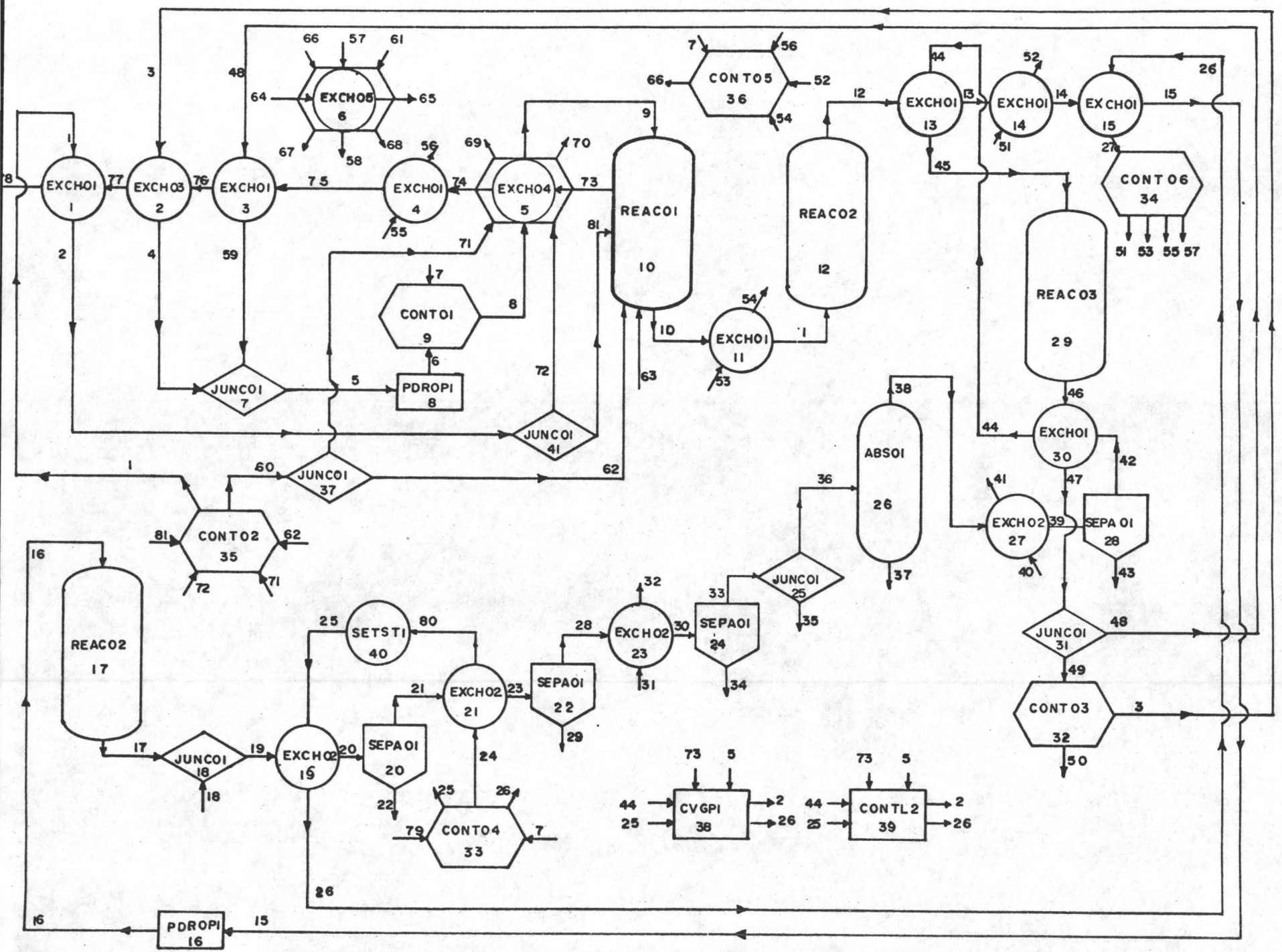

EXCH01 1
EXCH03 2
EXCH01 3
EXCH01 4
EXCH04 5
EXCH05 6
JUNC01 7
PDROPI 8
CONT01 9
REAC01 10
EXCH01 11
REAC02 12
EXCH01 13
EXCH01 14
EXCH01 15
REAC02 17
JUNC01 18
EXCH02 19
SEPA01 20
EXCH02 21
SEPA01 22
EXCH02 23
SEPA01 24
SETST1 40
CONT04 33
CONT02 35
JUNC01 37
CONT05 36
JUNC01 41
CVGPI 38
CONTL2 39
JUNC01 25
ABS01 26
REAC03 29
EXCH01 30
EXCH02 27
SEPA01 28
JUNC01 31
CONT03 32
CONT06 34
PDROPI 16

Table 8.1 Summary of Models for the Reformer Plant Simulation

Model	Number(s) on Information flow Diagram	Units simulated
1. Reactors		
REACO1	10	Naphtha Reforming Reactor
REACO2	12,17	Water Gas Shift Reactors
REACO3	29	Methanator Reactor
2. Heat Exchangers		
EXCHO1	1,3,4,11,13,14,15,30	Fluid-fluid Heat Exchangers
EXCHO2	19,21,23,27	Cooler-Condensers
EXCHO3	2	Naphtha vaporizer/superheater
EXCHO4	5	Steam-Naphtha Reactant Preheater, controls auxiliary burner or water addition to obtain desired reactant preheat temperature
EXCHO5	6	Auxiliary boiler
3. Mixers, Splitters, Separators, Absorbers		
JUNCO1	7,18,25,31,37,41	Mixers, Splitters
SEPAO1	20,22,25,28	Water Separators
ABSO1	26	Absorber/Desorber with water condensation
4. Controllers		
CØNTO1	9	Sets input reactant steam to desired steam/carbon ratio
CØNTO2	35	Sets air and fuel flowrates to reformer furnace and auxiliary burner for desired air-to-total fuel ratio
CØNTO3	32	Sets naphtha feedrate to obtain desired hydrogen production
CØNTO4	33	Sets expected inter-economizer boiler feedwater flowrates

CØNTO5	36	Determines steam production required in auxiliary boiler
CØNTO6	34	Transfers exit temperature from final aconomizers to waste heat boiler inlet water and water to auxiliary boiler
5. Miscellaneous		
PDROP1	8,16	Predicts pressure drop over a series of processing units module 8 - over hydrodesulphurizers and naphtha vaporizer Module 16 - over heat exchangers 13,14,5 Also used to predict pressure drop over individual units when called by model
SETST1	40	Set temperature change in a stream
CONTL2	39	Convergence tester
CVGP1	38	Convergence promoter

Both these reactions are strongly endothermic and hence the equilibrium in reactions 8.1 and 8.2 favours the production of hydrogen at high temperatures. Figure 8.2b shows the equilibrium constant (K_1) variation with temperature.

The water gas shift reaction

$$CO + H_2O \overset{K_2}{\rightleftharpoons} CO_2 + H_2 \qquad (8.3)$$

occurs to some extent. Since it is exothermic, the production of hydrogen is favoured at low temperatures (Figure 8.2a).

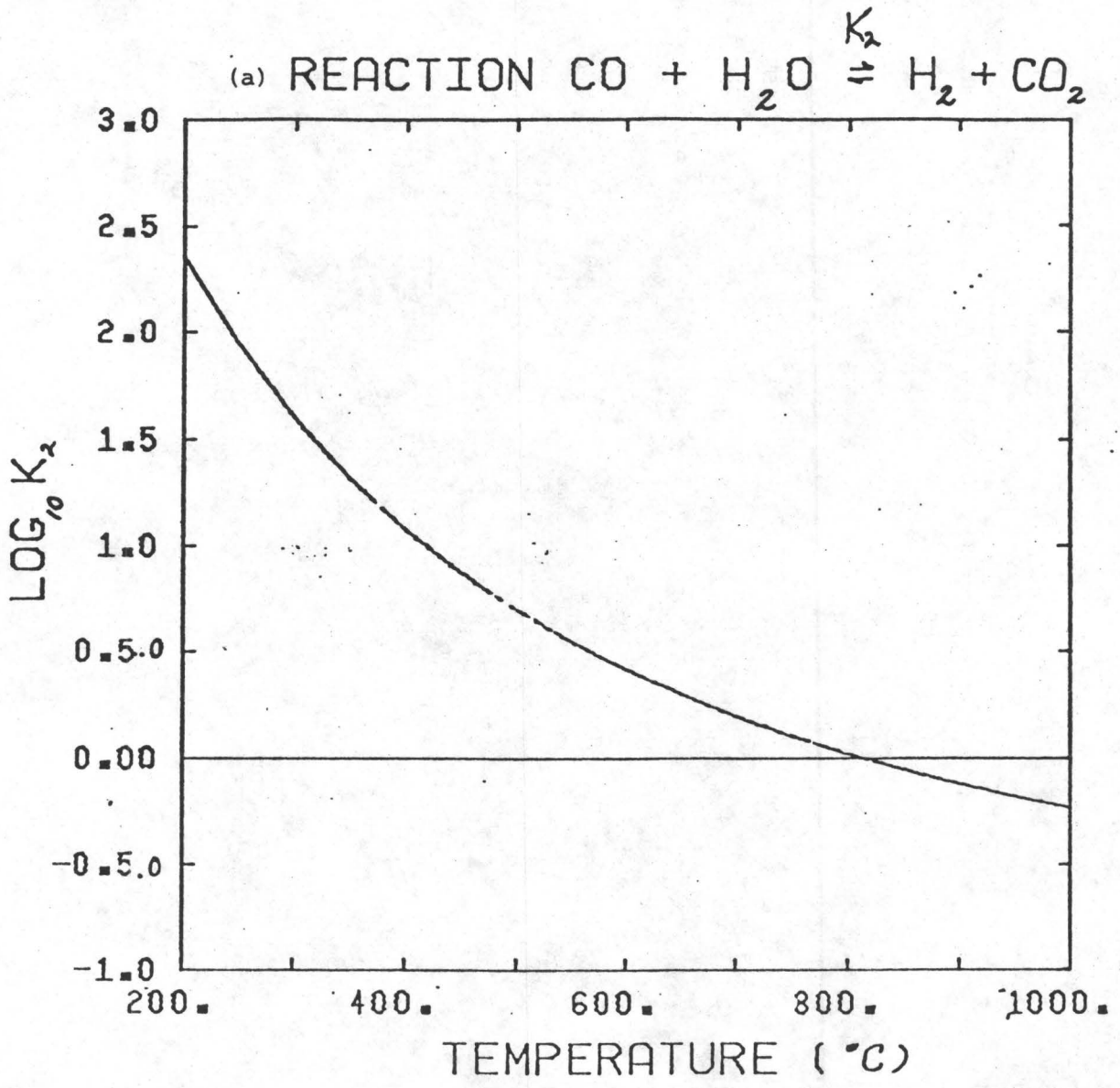
In the process there is a recycle product gas stream which contains hydrogen, methane and nitrogen. Call k,j,m moles of nitrogen, hydrogen and methane recycled per mole of naphtha fed, respectively, x, the moles of methane left per mole of naphtha fed and sn the moles of steam per mole of naphtha entering. Assume all naphtha reacts. Then moles of water vapour remaining is (sn-n+x) except for water reacted in the water gas shift reaction. Adding recycle to reaction 8.1, then

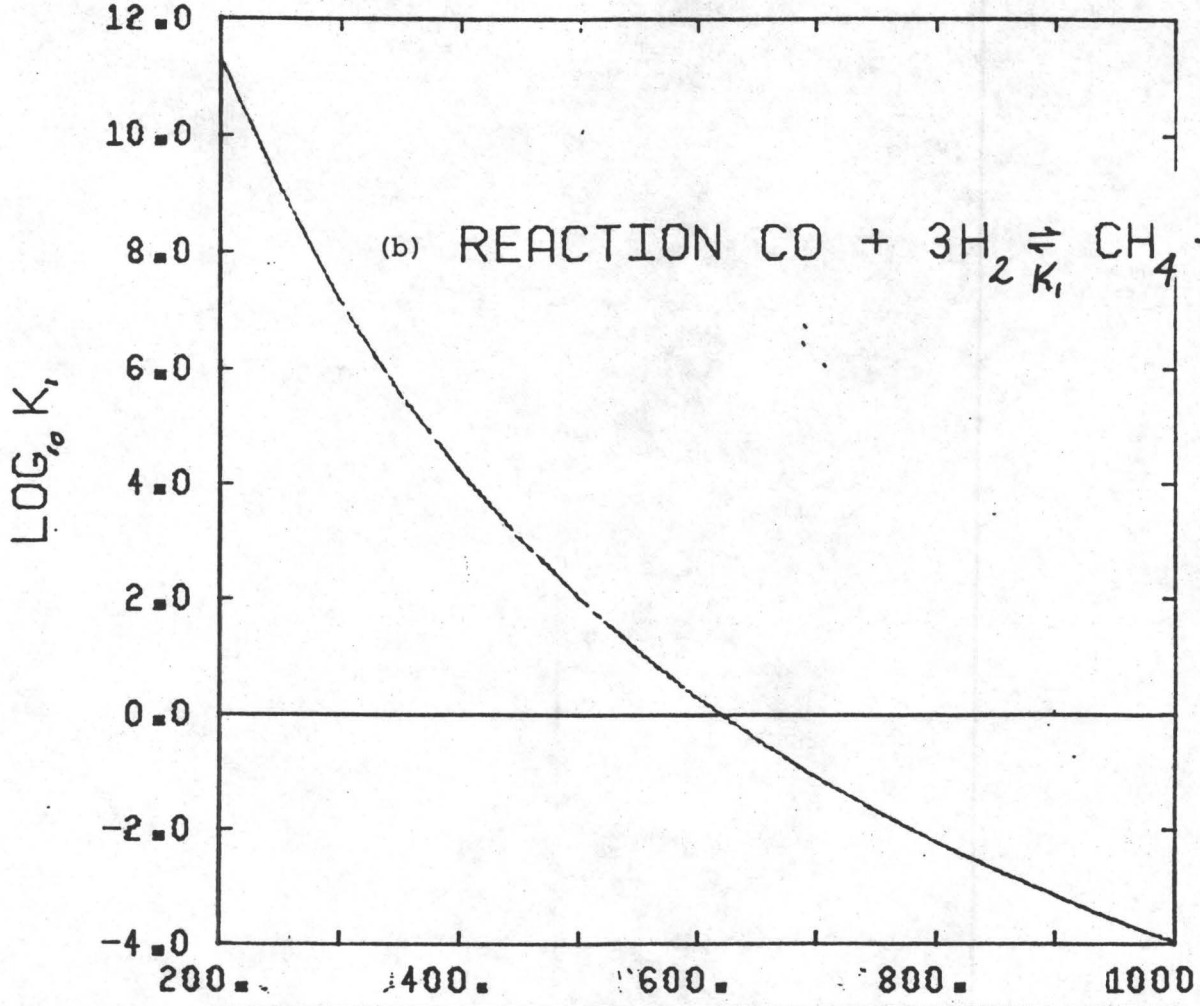
$$jH_2 + kN_2 + mCH_4 + C_nH_{Zn} + (n-x+m)H_2O \rightarrow (n-x+m)CO$$

$$+ \left(j + \frac{Zn}{2} + 3m-3x+n\right)H_2 + xCH_4 + kN_2 \qquad (8.4)$$

To account for water gas shift reaction, assume y moles of carbon dioxide are formed per mole of naphtha fed. Then y moles of carbon monoxide and water vapour disappear and y additional moles of hydrogen are formed. Table 8.2 shows the moles of each species remaining per mole of naphtha fed.

Figure 8.2 Equilibrium Constants Variation with Temperature




(b) REACTION CO + 3H2 ⇌ CH4 + H2O
K1
LOG10 K1
12.0
10.0
8.0
6.0
4.0
2.0
0.0
-2.0
-4.0
200.
400.
600.
800.
1000.

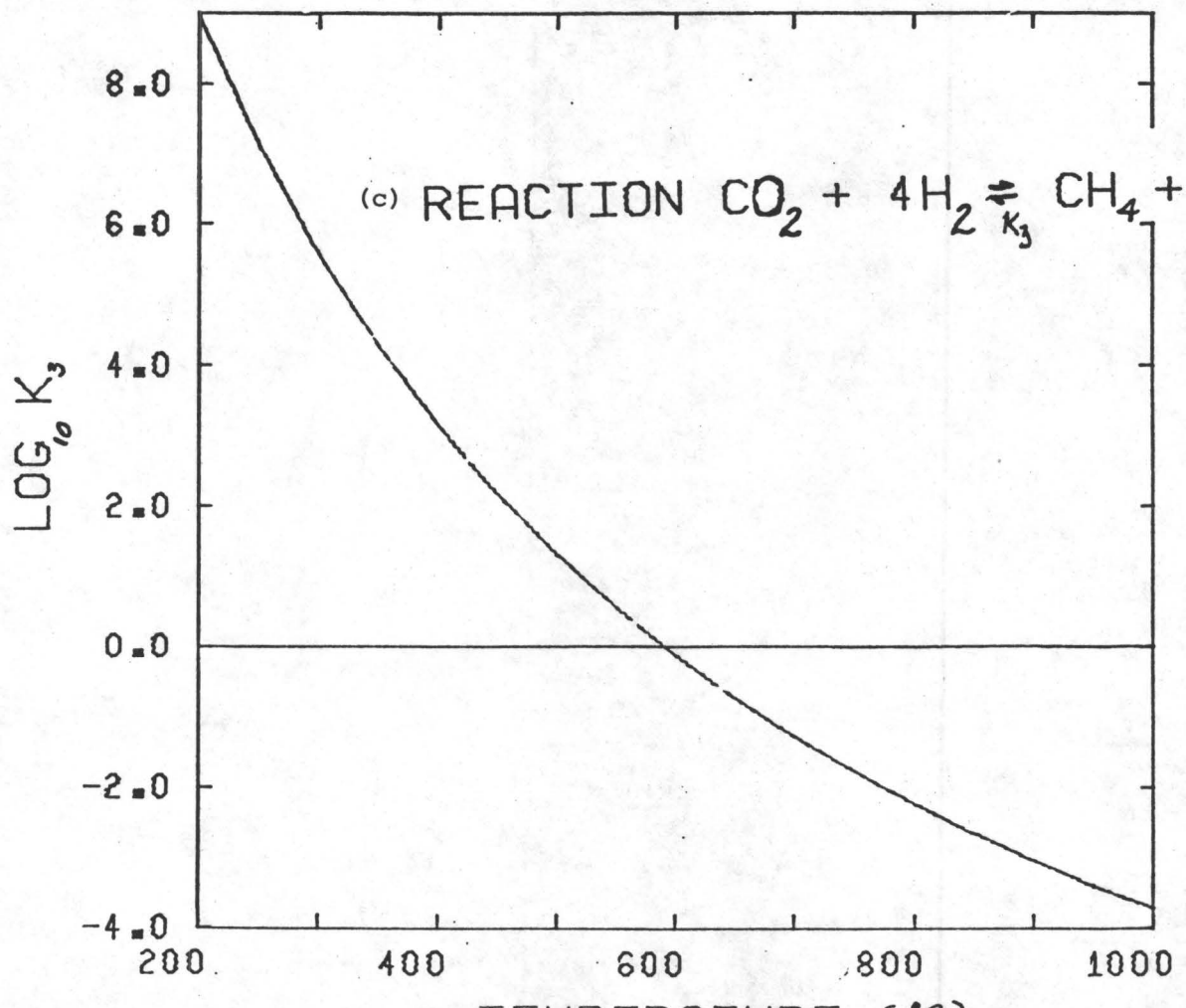

(c) REACTION CO2 + 4H2 ⇌ CH4 + 2H2O
K3
LOG10 K3
8.0
6.0
4.0
2.0
0.0
-2.0
-4.0
200
400
600
800
1000
TEMPERATURE (°C)

Table 8.2 Moles of Each Component Remaining per Mole of Naphtha Fed

CO_2	y
CH_4	x
H_2O	sn-n+x-m-y
H_2	$j + \frac{Zn+2}{2} + 3(m-x)+y$
CO	n-x+m-y
N_2	k
Total moles	$j + k + \frac{Zn}{2} + n + 3m - 2x + sn$

Now, $$K_1 = \frac{P_{CO} \cdot P_{H_2}^3}{P_{CH_4} P_{H_2O}} = \frac{[CO][H_2]^3}{[CG_4][H_2O]} \cdot \frac{P_T^2}{n_T^2} \qquad (8.5)$$

where [ ] represents the number of moles of a given component. Substituting from Table 8.2

$$K_1 = \frac{(n-x+m-y)(Zn-6x+2n+2y+6m+2j)^3}{(8)(x)(sn-n+x-y-m)(\frac{Zn}{2}-2x+n+sn+j+3m+k)^2} P_T^2 \qquad (8.6)$$

For the water gas shift reaction

$$K_2 = \frac{(y)(2j+Zn+6(m-x)+2y+2n)}{(2)(n-x+m-y)(sn-n+x-m-y)} \qquad (8.7)$$

Equation 8.7 is a quadratic in x^2 and may be solved for x for any given y. With this assumed y and x calculated from 8.7 for this y, K_1 may be evaluated from equation 8.6. This K_1 may be compared with the K_1 at the temperature in question. A convergence promotion technique (for example Newton's method) may be used to determine the correct y to yield K_1 to any desired tolerance.

The reactor may not be sufficiently long to ensure equilibrium.

This is usually accounted for by using an approach temperature, ΔT_{app}, which is substracted from the actual outlet temperature to give an equilibrium temperature T_{equil}. The composition of the exit gases corresponds to the composition calculated from Equation 8.6 and at this temperature. For K_2, since the lower the concentration is at a higher temperature, the approach temperature is at a higher temperature, the approach temperature is added to the outlet temperature. These approach temperatures are parameters in the model and must be determined from base case plant data.

The model also requires a calculation of the flue gas flowrate and temperature from the furnace supplying the heat of reaction to the naphtha reformer. This requires some knowledge of the controls the operator exercises. The desired operation, as it now exists, is to fire the furnace to ensure an outlet reaction product temperature of 800 to 810°C (usually 810°C); however, the tube wall temperature should not exceed some maximum value (950°C). If the desired outlet products temperature cannot be met at a low tube wall temperature, then the furnace is fired to maintain a wall temperature of 950°C. In the model, it is necessary to determine these temperatures at any operating condition.

For a given outlet temperature (say 810°C) and with the approach temperature known from normal operating base case analysis and for a given flowrate of feed, the outlet product gas flowrate can be determined and hence the amount of heat required to be transferred through the tubes can be calculated. This heat will be transferred by convection and radiation. This latter mechanism involves mainly a calculation of radiation from the flue gases, complicated by zero-flux or refractory reradiating surfaces. In the present model this radiation is included in a fictitious convection

coefficient, viz:

$$q = U_o A_o \Delta T_{\ell.m} \tag{8.8}$$

The furnace is made up of a large number of burners; each bank may be fired at a different rate to provide a desired wall temperature profile. If it is assumed, however, that the entire furnace flue gases are completely mixed and at a uniform temperature corresponding to the outlet temperature (perhaps plus some ΔT as recommended by Perry for furnace calculation (P1)), then

$$\Delta T_{\ell.m.} = \frac{(T_o - T_{in}) - (T_o - T_{out})}{\ell n \dfrac{(T_o - T_{in})}{(T_o - T_{out})}} \tag{8.9a}$$

where T_o = furnace flue gas temperature in furnace

T_{in} = reactant gas temperature at inlet

T_{out} = product gas temperature at outlet

$$\text{or} \qquad q = \frac{(UA)(T_{out} - T_{in})}{\ell n \dfrac{(T_o - T_{in})}{(T_o - T_{out})}} \tag{8.9b}$$

Now q^* is the heat transferred to the reacting gases (determined above).

The overall heat transfer coefficient-area product (UA) is determined from base case operating data (model fit)

---

*q is determined from the energy balance equation for the reactants:

$$\sum_{i=\text{reactants}} (w_i Cp_i)(T_{in} - T_{base}) + \sum_{\text{reactions}} \Delta H_R)_{T_{base}} + q = \sum_{i=\text{product gases}} (w_i Cp_i)(T_{out} - T_{base}) \tag{8.10}$$

$(T_{out}-T_{in})$ is already evaluated (desired temperature for T_{out}). Equation 8.9b allows T_0 to be evaluated (this is the necessary outlet flue gas temperature). The question is how much fuel must be fed to ensure this outlet temperature and the amount of heat required to be transferred.

An energy balance on the fuel/flue gas around the furnace provides this, since :

$$\sum_{\substack{i = \text{fuel} \\ \text{air} \\ \text{steam}}} (w_i Cp_i)(T_{in}-T_{base}) + \Delta H_{\substack{\text{combustion} \\ \text{at base} \\ \text{temperature}}} - q = \sum_{i = \text{flue gases}} (w_i Cp_i)(T_{out}-T_{base}) \qquad (8.11)$$

where $T_{out} \equiv T_o$ if completely mixed furnace gases are assumed. For the fuel system, the following analysis pertains: the fuel is a Bunker-C oil with a specified C/H mass ratio; call this CH. For (CH+1) grams of fuel entering, then $(\frac{CH}{12} + \frac{1}{2})$ g-moles of fuel combust. Assume complete combustion (i.e. sufficient excess air), then

$$(\frac{CH}{12}+\frac{1}{2})\ \text{fuel} + (\frac{CH}{12}+\frac{1}{4})\ O_2 \rightarrow (\frac{CO}{12})\ CO_2 + \frac{1}{2}\ H_2O \qquad (8.12)$$

or per gram of fuel, divide O_2, CO_2 and H_2O by (CH+1) to obtain

$$(\frac{CH}{12})(\frac{1}{CH+1})\ \text{moles of } CO_2 \text{ formed} \qquad (8.13a)$$

$$(\frac{1}{2})(\frac{1}{CH+1})\ \text{moles of } H_2O \text{ formed} \qquad (8.13b)$$

$$(\frac{CH+3}{12})(\frac{1}{CH+1})\ \text{moles of } O_2 \text{ required theoretically} \qquad (8.13c)$$

In combusting the fuel, the fuel is atomized in a two-fluid nozzle using steam. If AS grams of atomizing steam per gram of fuel is used, then

$$AS + \frac{1}{2(CH+1)} \quad \text{moles of } H_2O \text{ leaves the furnace per g of fuel} \tag{8.13d}$$

and if EX is the fraction excess air, then

$$(1+EX)\left(\frac{CH+3}{12}\right)\left(\frac{1}{CH+1}\right) \text{ moles } O_2 \text{ enter} \tag{8.13e}$$

$$\text{and } (EX)\left(\frac{CH+3}{12}\right)\left(\frac{1}{CH+1}\right) \text{ moles } O_2 \text{ leave} \tag{8.13f}$$

$$\text{and } (1+EX)\left(\frac{CH+3}{12}\right)\left(\frac{1}{CH+1}\right)\frac{79}{21} \text{ moles of } N_2 \text{ enter and leave per g of fuel fed} \tag{8.13g}$$

Hence, the energy balance equation 8.11 allows a calculation of the amount of fuel required for a given T_o.

For this T_o, however, the model must check first whether the tube wall temperature is exceeded under this firing rate. This can be calculated from:

$$q = (h_o A_o)\Delta T_{\ell.m.} \text{ where } \Delta T_{\ell.m.} = (T_o - T_w)_{\text{average}} \tag{8.14}$$

This requires an estimate of $(h_o A_o)$ which can be determined from base case conditions (model fit at a given operating conditions where T_{wall} and T_o are known).

If the tube wall temperature exceeds 950°C, then the actual exit temperature of the reaction products must be calculated for this condition.

This calculation requires some estimate of $(hA)_i$ since

$$q = (hA)_i \ (T_{wall}-T_{gas})_{\ell.m.} \qquad (8.15)$$

Again $(hA)_i$ must be estimated from plant measurements at a given operating condition. Equation 8.15 allows a calculation of q; substituting this value in equation 8.10 allows the calculation of the outlet temperature of the reactants. This leads to an iterative calculation since a new outlet temperature changes the equilibrium and consequently the component flowrates in the product gases.

The pressure is a key variable in this reactor since the equilibrium constant for the reforming reaction varies directly as the square of the pressure as indicated by equation 8.6. For instance, as the pressure drop increases across the reactor, the hydrogen production rate decreases. It is therefore very important to calculate accurately the pressure drop across the reactor. An iterative procedure is used to determine the pressure drop across the reactor until $(p^{n+1}-p^n)/p^{n+1}<0.001$. An initial value of pressure drop is assumed (stored in the EN vector); from this assumed outlet pressure p^n, the model calculates the outlet flows. It then determines the outlet pressure p^{n+1} by calling PDROP1 (see section 8.6.4); if the calculated outlet pressure has the same value as the assumed pressure (to within 0.1% tolerance), the model updates its EN vector and returns; if, however, the assumed and calculated outlet pressures are not the same, the model sets the assumed pressure equal to the calculated pressure and recalculates the flows, outlet pressure, etc. until the outlet pressure converges.

The listing and flowchart of subroutine REACO1, along with subroutine FRACS which is called by REACO1 to calculate the equilibrium constants, are shown in Appendix E.

8.3 The Water Gas Shift Reactor

Hydrogen is produced in two essentially adiabatic water gas shift reactors according to the reaction

$$CO + H_2O \rightleftharpoons H_2 + CO_2 \qquad (8.16)$$

Since this equilibrium reaction is exothermic, hydrogen production is more favourable at low temperatures. In addition, the unconverted methane from the reformer may react according to

$$CH_4 + H_2O \rightleftharpoons CO + 3H_2 \qquad (8.17)$$

as in the reformer.

In this reaction model the fraction of methane which reacts is specified and the amount of remaining reactants/products is then calculated. An outlet temperature is assumed and a specified approach temperature difference applied to yield a temperature at which equilibrium concentrations are calculated from

$$K_2 = \frac{(y+A)(y+B)}{(C-y)(1-y)} \qquad (8.18)$$

where y is the fraction of the carbon monoxide which reacts, A, B and C are the moles of carbon dioxide, hydrogen and water reactants per mole of carbon monoxide reactant prior to the water

gas shift reaction, respectively.

This quadratic equation is easily solved for y, from which the outlet component flows are obtained.

The heat evolution accompanying these reactions is then evaluated and the outlet temperature of the gases calculated by a heat balance. Since plant data indicate that some heat loss occurs, a specified temperature difference (model parameter from base case data) is substracted from this adiabatic temperature to account for this loss. A direct iteration is carried out until the calculated and assumed temperatures T agree to some tolerance ($\Delta T/T<0.001$).

Finally the pressure drop across the reactor is evaluated using PDROP1 (section 8.6.4). The program listing and flowchart of the Water Gas Shift Model is given in Appendix E.

8.4 The Methanator Reactor (REACO3)

The methanator is required to reduce the carbon monoxide and carbon dioxide concentrations in the final product gases to very low concentrations to avoid poisoning of the ammonia reaction catalyst. This is accomplished by the methanation reactions:

$$CO + 3H_2 \rightleftharpoons CH_4 + H_2O \qquad (8.19)$$

$$CO_2 + 4H_2 \rightleftharpoons CH_4 + 2H_2O \qquad (8.20)$$

Since the reactors have sufficient catalyst to ensure equilibrium at the outlet, the reactant/product flows are calculated from the equilibrium expressions for their reactions as follows:

Let X and Y be the moles of carbon monoxide and carbon dioxide which react. Then, let A, B, C, D, and E be the moles of carbon monoxide, carbon dioxide, hydrogen, methane, and water remaining after reaction. Then

$$K_3 \text{ (carbon monoxide reaction)} = \frac{(D)(E)}{(C)^3(A)} \cdot \frac{n_T^2}{p_T^2} \tag{8.21}$$

$$\text{and } K_4 \text{ (carbon dioxide reaction)} = \frac{(D)(E)^2}{(C)^4(A)} \cdot \frac{n_T^2}{p_T^2} \tag{8.22}$$

$$\text{where } n_T = n_T\Big|_i - 2x - 2y \tag{8.23}$$

Hence:

$$X = A - \frac{(D)(E)n_T^2}{(C)^3 K_3 p_T^2} \tag{8.24}$$

and dividing equation 8.22 by equation 8.21

$$Y = B - \frac{(E)(A)(K_3)}{(C)(K_4)} \tag{8.25}$$

Now A, B, C, D, E and n_T are functions of X and Y; therefore X and Y are assumed, and then calculated from equation 8.24 and 8.25 in turn. Since almost all of the carbon oxides react, it was found necessary to use the average of the assumed and calculated values in the convergence scheme. Y was converged first within the convergence for X. This overall procedure prevented negative values of X and Y. For any given X and Y, a heat balance allows a calculation of the outlet temperature so that

iteration on this temperature was required. This was done through subroutine CONVE. Because of heat loss, the outlet temperature was less than the calculated adiabatic temperature and a constant specified temperature difference was therefore applied.

Subroutine REACO3 is listed (and flowcharted) in Appendix E.

8.5 Heat Exchanger Models

A number of different types of heat exchangers are modeled in the process; these, along with their respective mathematical models are listed in Table 8.3. The details of these heat exchanges models are indicated below:

8.5.1 Model EXCHO1

This model predicts the outlet temperature of one fluid where that fluid increases or decreases its sensible heat content only and the other fluid may

(i) increase or decrease its sensible heat content, and hence its outlet temperature is predicted, or

(ii) be a boiling liquid, in which case its temperature is constant and hence the amount of vapour produced is predicted.

The model allows for cocurrent or countercurrent flow of fluids or multi-shell and tube passes.

Since the inlet temperatures and flowrates of the incoming fluids are assumed known in the simulation mode, the use of the effectiveness factor-transfer unit concept is the most appropriate for use here since

Table 8.3 Heat Exchangers in the Reformer Process

Equipment Number on Information flow diagram	Plant specification number	Model	Tube-Side Fluid	Shell-Side Fluid	Type of Exchanger
N.M.*	C-118	-	air	condensing steam	gas/cond.steam
1	C-113	EXCHO1	air	flue-gas	gas/gas
2	C-112	EXCHO3	naphtha	flue-gas	naphtha evaporator-superheater/gas
3	C-111	EXCHO1	hydrogen recycle	flue-gas	gas/gas
4	L-117	EXCHO1	flue gas	boiling water	waste heat boiler
5	C-109	EXCHO4	naphtha vapour/ steam reactants	flue-gas	reactant super heater
11	L-201	EXCHO1	reformer product gas	boiling water	waste heat boiler
13	C-203	EXCHO1	product gas from high temp. Water Gas Shift reactor	gas from absorber	gas/gas
14	L-204	EXCHO1	product gas from high temp. Water Gas Shift Reactor	boiling water	waste heat boiler
15	C-205A/C-205B	EXCHO1	boiler feed water	product gas from high temp. Water Gas Shift reactor	gas/water economizer
19	C-207	EXCHO2	boiler feed water	product gas from low temp. Water Gas Shift reactor	cooler-condenser (economizer)

Table 8.3 (cont'd)

21	C-209	EXCHO2	boiler feed water	product gas from low temp. Water Gas Shift reactor	cooler-condenser (economizer)
23	C-301A/C-301B	EXCHO2	boiling carbonate solution	product gas from low temp. Water Gas Shift reactor	cooler-condenser (reboiler)
N.M.*	C-307	-	cooling water	carbonate regenerated solution	liquid/liquid
27	C-304	EXCHO2	cooling water	product gas from absorber	cooler-condenser (cooler)
30	C-212A/C-212B	EXCHO1	product gas from absorber	product gas from methanator	gas/gas
N.M.*	C-107/2	-	fuel	condensing steam	liquid/cond. steam

* N.M. - exchanger not modeled in the process simulation.

it involves the least iteration in the calculation of outlet temperatures. In this case, the definition of the effectiveness factor ε depends on the relative thermal capacities (WC) of the two fluids, viz:

$$\varepsilon = \frac{T_{c_{out}} - T_{c_{in}}}{T_{h_{in}} - T_{c_{in}}} \quad \text{if } WC)_{cold} \leq WC)_{hot} \tag{8.25}$$

$$\varepsilon = \frac{T_{h_{in}} - T_{h_{out}}}{T_{h_{in}} - T_{c_{in}}} \quad \text{if } WC)_{h} \leq WC)_{c} \tag{8.26}$$

Hence, if the effectiveness factor and the thermal capacities are known, knowledge of the appropriate effectiveness factor allows the calculation of one of the outlet temperatures. The heat balance

$$WC)_{c} \, (T_{c_{out}} - T_{c_{in}}) = WC)_{h} \, (T_{h_{in}} - T_{h_{out}}) \tag{8.27}$$

allows the calculation of the second outlet temperature.

The effectiveness factor depends on the number of transfer units (NTU), defined by

$$NTU = \frac{UA}{WC)_{smaller}} \tag{8.28}$$

and the flow condition in the exchanger. The derivation of these formulas may be found in Kays and London (K1); the formulas for the most common exchangers are summarized below (and included in EXCHO1 model). Unless otherwise noted, the value of R is given by

$$R = \frac{WC)_{smaller}}{WC)_{larger}} \tag{8.29}$$

(a) True Countercurrent Exchanger

$$\varepsilon = \frac{1 - \exp(-NTU(1-R))}{1 - R\exp(-NTU(1-R))} \tag{8.30}$$

when $R = 1$, $\varepsilon = NTU/(1+NTU)$ (8.31)

(b) True Cocurrent Exchanter

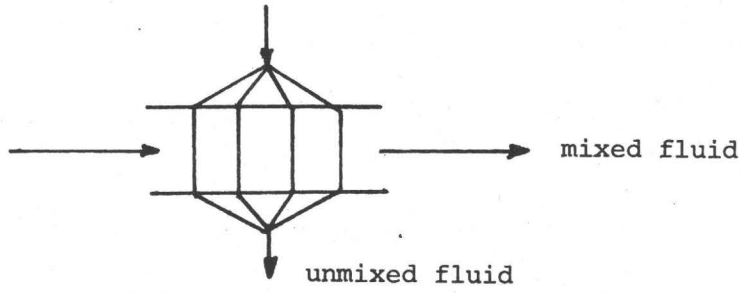
$$\varepsilon = \frac{1 - \exp(-NTU(1+R))}{1+R}$$

Both arrangements (a) and (b) reduce to

$$\varepsilon = 1 - \exp(-NTU) \tag{8.32}$$

when $R = 0$

(c) Crossflow Exchanger, one fluid mixed at any cross-section, the other Unmixed; the unmixed flow flows in parallel through a number of tubes.



There are two conditions of relative thermal capacities

(i) $WC)_{larger}$ = unmixed fluid, $WC)_{smaller}$ = mixed fluid.

Then, $\varepsilon = 1 - \exp(-\Gamma/R)$ (8.33a)

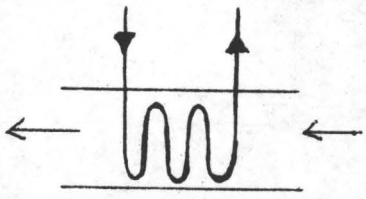
with $\Gamma = 1 - \exp(-NTU*R)$ (8.33b)

(ii) $WC\Big|_{larger}$ = mixed fluid, $WC\Big|_{smaller}$ = unmixed fluid,

Then, $$\varepsilon = \frac{1}{R} * (1-\exp(-\Gamma' * R)) \qquad (8.34a)$$

with $$\Gamma' = 1 - \exp(-NTU) \qquad (8.34b)$$

(d) Crossflow Exchanger, one mixed (at any cross-section) flows across a bank of tubes, the flow through which is in series and in a counter-current or cocurrent manner.



It is shown in Kays and London (K1, Figure 2-18, p53) that if the number of tube passes (assumed to be mixed flow, only one tube) is greater than 3 or 4, this exchanger approaches that calculated for a true conutercurrent or cocurrent exchanger. If the number of tubes is small, the effectiveness factor is calculated from

$$\varepsilon = \frac{\gamma^n - 1}{\gamma^n - R} \qquad (8.35a)$$

where $$\gamma = \frac{1-\varepsilon p R}{1-\varepsilon p} \qquad (8.35b)$$

and εp is the effectiveness factor for one tube (fluid mixed) and cross flow of the other fluid (unmixed) as given in (c). The exponent, n, is the number of tubes mounted across the flowing unmixed stream.

(e) Cocurrent-Countercurrent (Multi-pass) Exchangers

(i) One shell pass and 2, 4, 6 --- tube passes

In this case,

$$\varepsilon = \frac{2}{\left\{(1+R) + \frac{\sqrt{1+R^2}\,(1+\exp(-\alpha))}{(1-\exp(-\alpha))}\right\}} \qquad (8.36a)$$

where $\alpha = NTU*\sqrt{1+R^2}$ (8.36b)

This expression was derived for 1 shell pass, 2 tube passes and is recommended for use when there are a large number of tube passes because the results for these are very close to that for the 1, 2 exchanger.

(ii) Two shell passes, 4, 8, 12 --- tube passes

Equation 8.35 is used to calculate the effectiveness factor, where εp is calculated from equation 8.36 with n, the number of shell passes. Note that when n=1, $\varepsilon=\varepsilon p$.

In these formulations, the overall heat transfer coefficients are assumed constant over the exchanger. In the model, the overall heat transfer coefficient (UA) is assumed constant at some specified value. The variation of the individual coefficients, and hence the overall coefficient, with flowrate and physical properties is not accounted for. The UA product is inserted in EN(16).

The flowchart (and listing) of the model is shown in Appendix E. The type of exchanger is determined by the number inserted in EN(17); 1 to 7 specifies the exchanger flow arrangement for exchangers without phase change; 8 and 9 specifies an exchanger with phase change of the shell-side fluid (boiling or condensing steam, respectively).

The pressure drop across the exchanger (shell side) can be calculated by calling PDROP1 (section 8.6.4) if EN(24)=1. The constant factor K in the pressure drop equation 8.46 is then specified in EN(25).

8.5.2 Model EXCHO2

Subroutine EXCHO2 models a cooler-condenser given an overall heat transfer coefficient-area product (UA) for it. The use of the effectiveness factor (or the log-mean temperature driving force) assumes:

(i) the overall heat transfer coefficient, the heat capacities of the fluids and the flowrates are constant

(ii) The heat is transferred linearly with the heat exchange area that the fluid traverses

(iii) plug flow of fluids.

These assumptions are violated to some extent in a cooler-condenser. On the other hand, a simple model, although approximate, may be written on the basis of the following analysis.

A fictitious thermal capacity of the fluid may be evaluated on the basis of a heat balance on the partially condensing fluid, viz:

$$q = \sum_{i=1}^{NØCØMP} WC_p)_i (T_{in}-T_{out}) + W_{cond}\,\lambda_{T_{out}} = WC)_{EFF}(T_{in}-T_{out}) \qquad (8.37)$$

where it is assumed that all of the inlet gas is cooled to the outlet temperature and then part of the condensing fluid, W_{cond}, is condensed at the outlet temperature. This effective thermal capacity, $WC)_{EFF}$, then is used as thermal capacity in the appropriate effectiveness factor

calculations. This procedure requires knowledge of the amount of liquid condensed. In the initial model for the cooler-condenser, this was specified as a fraction of the incoming vapour, which was determined from plant data. This procedure, however, does not respond correctly to changes in operating conditions; therefore a better estimate of the amount condensed was determined as follows.

If it is assumed that the gas leaving the exchanger is saturated with respect to the condensing vapour at the outlet temperature of the fluid, then

$$\frac{\text{moles of vapour in the outlet gas}}{\text{moles of dry gas in the outlet gas}} = \frac{\text{vapour pressure at the outlet temperature}}{\text{total pressure - vapour pressure}} \qquad (8.38)$$

Thus, recognizing that the amount of dry gas flowing through the exchanger is constant, the amount of water condensed is determined by the outlet temperature. The outlet temperature is also determined from the effectiveness factor calculation (i.e. the actual heat transfer rate that can be achieved in the given heat exchanger). When the heat transfer rate as given by equation 8.37 is the same as that calculated from the effectiveness factor calculations, the outlet temperature (and the amount of water condensed) is correct. This iteration is programmed in EXCHO2, as indicated in the flowchart and listing in Appendix F.

The model then calculates the pressure drop of the condensing fluid across the exchanger based on the average flowrate of gases.

8.5.3 Model EXCHO3

Subroutine EXCHO3 models the naphtha vaporizer-super heater which is heated by furnace flue gas. As indicated in section 7.7 and on Figure 7.1, the vaporization process is not a constant temperature process since at any operating pressure, naphtha exhibits a range of boiling points. This means that at a pressure of 48 atmospheres, the vaporization starts at about 300°C and is completed at about 370°C (curves 1 and 2, Figure 7.1.) The variation in temperature with distance through the naphtha vaporizer-super heater is illustrated in Figure 8.3. In the model, the vaporization process is assumed to take place at a constant temperature of 300°C as illustrated. Since the heat capacity of naphtha liquid is about 20% larger than the vapour, this procedure will not introduce a large error.

As in EXCHO2, an effective thermal capacity can be calculated from

$$q = W_N\left\{C_{p_L}\Big|_N (300-T_{in}) + \lambda_{v,N,300^oC} + C_{p_v}\Big|_N * (T_{out}-300)\right\}$$

$$= WC\Big|_{EFF} (T_{out}-T_{in}) \qquad (8.39)$$

The effectiveness factor is calculated for a single-pass counter-current exchanger, thus yielding the outlet temperature of both streams. The overall heat transfer coefficient-area product (UA) is assumed constant as calculated from base-case data.

The model determines if the heat transfer rate is sufficient to vaporize and super heat all of the naphtha feed. If insufficient heat is transferred to vaporize the naphtha, the fraction vaporized is calculated and the appropriate amount of vapour is put in the SØ vector, the remaining naphtha is put in SØ(2,16). A message indicating this state is printed out.

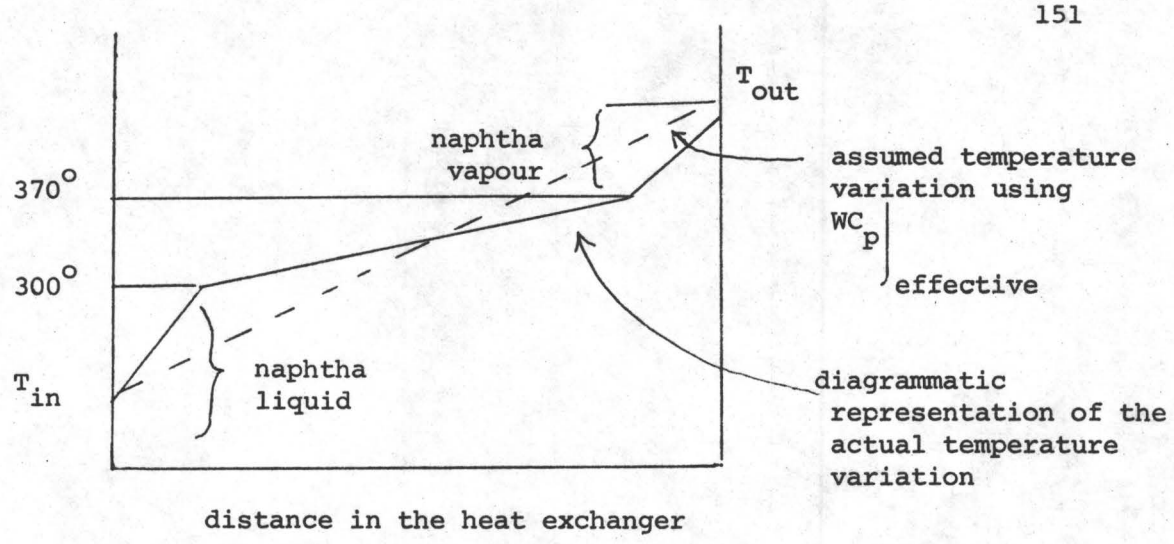


Figure 8.3 - Temperature Variation of Naphtha in the Naphtha vaporizer/superheater

Subroutine EXCHO3 program is listed and flowcharted in Appendix E.

8.5.4 Model EXCHO4

Subroutine EXCHO4 models the reactant super heater which has fraction F' and fraction (1-F') of the total exchanger area in cocurrent and countercurrent flow, respectively. Figure 8.4 indicates the arrangement of the auxillary burner, water injector and cocurrent, countercurrent exchangers.

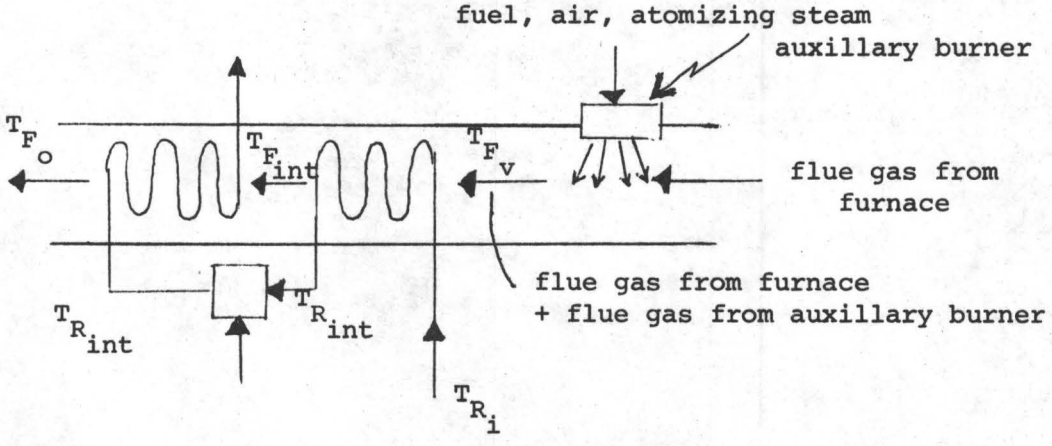


Figure 8.4 Auxillary Burner, Water Injector and Cocurrent-Countercurrent Exchanger (all Modeled in Subroutine EXCHO4).

The control procedures in the model are as follows: for T_{R_o} calculated with burner and water off:

(i) if $T_{desired} < T_{R_o} < T_{max}$, do not turn on the auxiliary burner or water

(ii) if $T_{R_o} < T_{desired}$, turn on the auxiliary burner to provide $T_{R_o} = T_{desired}$

(iii) if $T_{R_o} > T_{max}$ turn off burner and turn on the water to provide $T_{R_o} = T_{max}$.

The model, EXCH04, calls a subroutine HEX which is a model for the heat exchanger shown above. The derivation of the outlet flue gas and reactant temperatures in terms of the effectiveness factor of the two exchangers is shown in Appendix E. The appropriate equation to be used is determined by the relative thermal capacity of the two stream.

If $\left.\frac{WC_p}{WC_p}\right|_F^R < 1$, then, with $R = \left.\frac{WC}{WC}\right|_F^R$

$$T_{F_o} = T_{F_i}(1-FAC2) + T_{R_i} * FAC2 \qquad (8.40a)$$

and $$T_{R_o} = T_{R_i} + R(T_{F_i} - T_{F_o}) \qquad (8.40b)$$

If $\left.\frac{WC_p}{WC_p}\right|_F^R \leq 1$ and $R = \left.\frac{WC}{WC}\right|_F^R$,

$$T_{R_o} = T_{R_i}(1-FAC2) + FAC2*T_{F_i} \tag{8.41a}$$

and

$$T_{F_o} = T_{F_i} - R(T_{R_o} - T_{R_i}) \tag{8.41b}$$

with

$$FAC2 = \varepsilon_1 + \varepsilon_2 - \varepsilon_1\varepsilon_2(1+R) \tag{8.42}$$

where ε_1 is the effectiveness factor of the cocurrent exchanger and ε_2 is the effectiveness factor of the countercurrent one.

If the outlet temperature, as calculated above, is less than the desired temperature (condition (ii)), the auxiliary burner is turned on. The program proceeds with the following steps:

(i) Determine the adiabatic flame temperature for the particular flow of atomizing steam and excess air (independent of flowrate)

(ii) (a) Set the fuel flowrate initially to a previously calculated value or to 5 kg/hr;

(b) calculate the flue gas flow from this fuel rate; and

(c) mix burner flue gas and furnace flue gas and determine the resulting flow and temperature

(iii) For this, new flowrate and temperature, calculate outlet reactant temperature by calling HEX

(iv) Test if calculated reactant temperature is the desired temperature, if not increment fuel by DF(=5) and repeat

(v) If the reactant temperature is equal to the desired one within tolerance, move on to (vii)

(vi) If the reactant temperature is above desired, decrement the fuel rate by DF and set switch for Newton's convergence on fuel rate. Converge on desired fuel rate

(vii) Reset inlet flows of flue gas from reformer to original values and inlet and outlet stream vector of fuel, reactants, air and water; calculate the pressure drop in the reacting gas stream (viii) Exit.

If the outlet reactant temperature is above the maximum allowed temperature, the water must be turned on to give the outlet temperature equal to the maximum. Then the program proceeds with:

(i) Calculate intermediate temperature after cocurrent part of exchanger (ε_1 known from previous calculation);

(ii) Calculate approximate water needed to remove enthalpy in excess of T_{max} in reactant gases. This is an estimate of water needed.

(iii) Start convergence on water needed.

(a) Add water to reactant gases after cocurrent exchanger, vaporize it and calculate this new inlet temperature to the cocurrent exchanger.

(b) Calculate the new outlet reactant temperature after re-calculating ε_2.

(c) Test if reactant and flue gas temperatures have converged on assumed gas temperatures (in calculation of ε_2)

(d) Test if calculated outlet reactant temperature is close enough to the maximum.

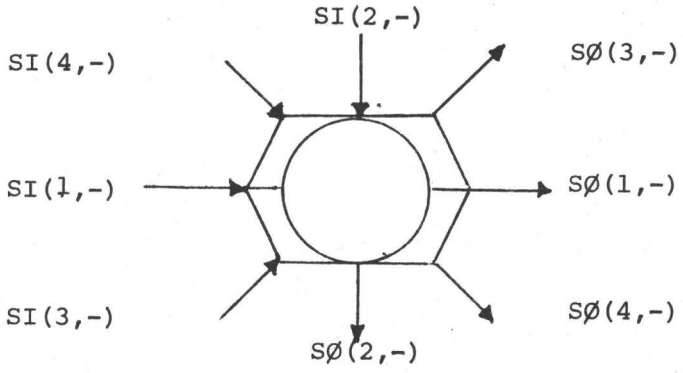
(e) If not return to (ii) and increment water flow appropriately (direct iteration) depending on the new excess or deficiency in enthalpy of reactant gases relative to the maximum temperature.

(iv) Set auxilliary burner fuel flow to zero, water flow and new outlet stream vectors of reactant and flue gases and calculates the pressure drop across the exchanger for the reacting gas stream.

A flowchart and a listing of subroutine EXCHO4 are available in Appendix E.

8.5.5 Model EXCHO5

Since the high pressure steam (reactant) requirements can not be satisfied by the production of steam in the waste heat boilers in the process, an auxillary boiler must be fired to meet them. Subroutine CØNTO5 determines the amount of steam to be produced in the auxillary boiler and this information is transferred to EXCHO5. The other input/output information is shown on Figure 8.5.



Input streams

1. Fictitious stream which will contain flue gas at adiabatic flame temperature
2. Boiler feed water (temperature available, flow to be calculated)
3. Required steam to meet process requirements (from CØNTO5)
4. Fuel (Temperature available, flow to be calculated)

Output streams

1. Flue gas from boiler
2. Steam produced in boiler
3. Air required for combustion of fuel (temperature available as input, flow to be calculated)
4. Atomized steam required for combustion process (temperature available as input, flow to be calculated)

Figure 8.5 Input/Output Information of the Auxiliary Boiler (Model EXCHO5)

The air and atomizing steam, although inputs for material and temperature information are put in the output streams because their flows are calculated; note that the fuel information (SI(4,-)) has the same characteristics. The GEMCS executive system, however, allows a maximum of four input and four output streams only.

A comprehensive model of this boiler unit requires a sophisticated description of the combustion, heat exchange and fluid mechanical processes occurring within it. Since these physical/chemical processes do not interact directly with the rest of the process, a simple model is both necessary and sufficient.

The simplest model for this combustion/heat exchange system is to assume the fuel burns adiabatically, heat exchange then occurs by convection only (linearization of the radiation processes) and the combustion products flow through the heat exchanger in plug flow. This is obviously an oversimplification of the actual phenomena occurring. The calculated fuel requirements will be correct, but the exit flue gas temperature will be in error.

Using these assumptions, the heat exchange part of the model is essentially the same as EXCHO1 (boiler) and, as a result, EXCHO5 calls EXCHO1 to perform this calculation after the flow of the flue gas and its inlet temperature are evaluated. This is the reason for having the order shown for the first two inlet and outlet streams on Figure 8.4, that is EXCHO1 assumes this order of input/output streams.

The unknown fuel rate (with accompanying air and atomizing steam) is calculated by an iterative procedure to ensure the required steam is produced. This is done by calculating the adiabatic flame temperature for the desired excess air, atomizing steam and inlet temperatures. A fuel

rate is assumed and the amount of flue gas produced is calculated as in the furnace calculation in REAC01. EXCH01 calculates the amount of steam produced. The program then proceeds to set up and use Newton convergence promotion to match the fuel rate with the desired steam production rate. The overall heat transfer coefficient-area product which is required in EXCH01 is determined from base case data and is assumed constant in the model.

A listing and flowchart of subroutine EXCH05 are available in Appendix E.

8.6 Miscellaneous Models

8.6.1 Mixer/Splitter Model (JUNC01)

This mixer model simulates the complete mixing of up to four input streams. It performs a mass and energy balance to determine the total output component flows and the mixture temperature. If only a material balance with a fixed desired temperature is required, a switch in the EN vector allows this; the desired temperature and pressure are put into EN(4) and EN(5) respectively.

If there is more than one stream leaving the junction, the fraction of the total flow going to each output stream must be specified in the EN vector. The concentration in each stream will be the same as the mixture composition. The model may function as a splitter only, that is it may have only one input stream. Subroutine JUNC01 program listing is available in Appendix E.

8.6.2 Separator Model (SEPA01)

This model has one input and two output streams. It allows a

specified splitting of the incoming component flows into these two streams by specifying the fraction of each component flow which proceeds to a given output (these fractions are specified by the user in the EN vector). The fractions are assumed constant under any operating condition. Subroutine SEPA01 program listing is available in Appendix E.

8.6.3 Absorber Model (ABS01)

This model simulates the absorber system by specifying the fraction of the incoming component flow which goes to the overhead. The outlet temperature and pressure are also specified and assuming the output gas stream is saturated with water vapour, the model calculates the amount of water vapour in the output process stream. The water which is condensed from the process gas stream is thus calculated and put in the absorbed gas stream. Subroutine ABS01 program listing is available in Appendix E.

8.6.4 Pressure Drop Calculator (PDROP1)

This model calculates the pressure drop through a particular processing unit or series of units. It can be self standing or can be called from another unit computation.

The model is based on the following analysis: The equation for calculating the pressure drop in conducts, packed beds, etc., may be cast in the form

$$\Delta P = f\left(\frac{L}{D}\right)\left(\rho \frac{v^2}{2g_c}\right) \tag{8.43}$$

where f is a friction factor, L is some (equivalent) length and D is a

characteristic dimension of the system; ρ is the density of the fluid mixture, and g_c is the universal gravitational constant. If it is assumed that, for any processing unit(s), f is essentially constant over the range of operating conditions of the plant (fully turbulent flow), then for any unit or group of units

$$\Delta P = K'\rho v^2 \tag{8.44}$$

and v is a characteristic average velocity which is easily calculated for any stream assuming an ideal gas as:

$$v = \frac{G}{\rho} = \frac{W_M * M_{ave}}{A_x * \frac{P_T M_{ave}}{RT}} = \frac{W_M RT}{P_T * A_x} \tag{8.45}$$

where W_M is the average total molar flowrate, M_{ave} is the average molecular weight of the flowing gas and T and P_T are the average temperature and pressure of the stream respectively; A_x is a characteristic cross-sectional area for the unit(s). Then

$$\Delta P = \frac{K'}{A_x^2} * \frac{P_T M_{ave}}{RT} * \left(\frac{W_M RT}{P_T}\right)^2 = K \frac{W_M^2 M_{ave} RT}{P_T} \tag{8.46}$$

Equation 8.46 accounts for the major effect of changing operating conditions (T, P_T, W_M) on the pressure drop; it allows a first order extrapolation from base case operating data since the coefficient K is determined from these data. Subroutine PDROP1 program is listed in Appendix E.

8.7 Controller Models

The simulated process is required to operate according to the actual process under any desired operating condition. This means that certain constraints and operating conditions must be built into the simulation. In addition, these are some mathematical manipulations that ensure that the simulation converges quickly to its final steady-state. These unit computations are described below. Their program listings are available in Appendix E.

8.7.1 Model CØNTO1

The operators of the plant control the desired reactant steam flowrate to the reformer. This flowrate is determined by the naphtha flowrate to the process and is specified as a desired ratio of moles of steam/atoms of carbon in the feed gas (ratio specified in the EN vector). Subroutine CØNTO1 calculates the required input flow of steam from both that ratio and the inlet reactant gases. After the required input steam is calculated, the model calls JUNCO1 to perform the heat and material balance to evaluate the temperature of the mixture. The EN vector for this model, therefore, has to be compatible with that for JUNCO1.

8.7.2 Model CØNTO2

This model determines the total fuel and air requirements as calculated in the reformer model (REACO1) and the auxillary burner (EXCHO4) and then sets these flows in the appropriate incoming stream. The fraction splits for the splitter models for the fuel (EN vector in NE=37) and for the air (EN vector in NE=41) are also set from this model.

8.7.3 Model CØNTO3

The process is required to produce a desired amount of hydrogen according to the requirements in other parts of the overall plant (ammonia production and hydrogenation processes). Since the hydrogen production is related to the naphtha feed rate, subroutine CØNTO3 controls the naphtha feed rate to ensure a desired hydrogen production. On every iteration of the simulation, whether the system has converged or not, CØNTO3 tests whether the desired hydrogen production is met and if not it adjusts the naphtha feed rate according to

$$\text{(New Naphtha Feedrate)} \equiv \text{(Old Naphtha feedrate)} * \frac{\text{Desired hydrogen production rate}}{\text{Calculated hydrogen production rate}} \tag{8.47}$$

Since this equation is being applied on each iteration, it becomes part of $f(\underline{x})$ in the convergence promoter, CVGP1, and hence will not introduce any discontinuity in $f(\underline{x})$.

8.7.4 Model CØNTO4

After the required total steam production rate is determined from CØNTO1, the boiler feed water rate, which is to be preheated in the economizers, is known since its flow is calculated from:

$$\text{(Boiler Feedwater Rate)} \equiv \text{(steam production rate)} * \text{(1. + fraction blowdown in boilers)} \tag{8.48}$$

This boiler water feedrate then needs to be set in the boiler feedwater input streams entering each of the four economizers. This is done in CØNTO4 with these four streams as output streams. Estimates of the input

water temperatures to each of the economizers are required in the initial data set since three of these are recycle streams).

8.7.5 Model CØNTO5

This subroutine determines the steam production rate required in the auxillary boiler. It sums the steam production rates from the three waste heat boilers and substract this quantity from that required as reactant. The required production rate is put in the output stream.

8.7.6 Model CØNTO6

The boiler feedwater temperature to each of the boilers is required to perform the heat exchanger calculations in them. This temperature is only known after the boiler water temperature is calculated from the final economizer in this series network. CØNTO6 transfers this temperature information to the appropriate input stream to the three waste heat boiler and the auxillary boiler. Note that this temperature information constitutes recycle of information and really adds three recycle streams to the main process.

9. ESTIMATING PARAMETERS IN MODELS FROM PLANT BASE CASE DATA

9.1 Background

The object of the steady state simulation program is to predict temperature, pressure and component flows for all streams in the plant given some set of control conditions, feedrates, etc. The mathematical models for the various operations have been formulated and these contain adjustable parameters such as the overall heat transfer coefficients in heat exchanger, the approach to equilibrium temperatures in reactors, the constant factor K in equation 8.46 for the pressure drop calculation across any equipment, etc. These parameters must be estimated from measurements of component flows, temperature and pressure at some steady-state operating conditions for the plant, known as the base case conditions.

9.2 Plant Data and Observations

A concerted effort was made to operate the reformer plant at steady-state conditions for the twenty-four hour period before samples and measurements were taken. All measured temperatures and pressures are shown on the process flowsheet, Figure 6.1. Single values of some feed flows such as the naphtha feed, the reactant steam, the fuel and air flows are also indicated on the process flowsheet. The compositions of the key streams (numbered in a triangle on the process flowsheet) are given in Table 9.1

Table 9.1 Composition of the Key Streams of the Naphtha Reformer Plant[1]

stream number on the process flowsheet	Component flows (kgmole/hr)					
	water vapour	carbon dioxide	carbon monoxide	methane	hydrogen	total flow
①	686.91964	67.92991	20,80982	3.97009	262.14018	1041.76964
②	675.75982	79.20000	9.42009	4.09018	273.05982	1041.52991
③	667.00000	87.14018	1.70982	3.85982	281.67991	1041.98973
④	0.36563	0.14019	1.44018	3.45000	281.67991	287.07589
⑤	2.08428	0.	0.	5.05572	276.98193	284.12193
⑥	2.02009	0.	0.	4.90000	268.45089	275.37098
⑦	0.06420	0.	0.	0.15572	8.53104	8.75096

1. These key streams are indicated in triangles on the process flowsheet (Figure 6.2)

Preliminary calculations using these plant data showed that some of the measured temperatures and/or measured flows in the flue gas heat exchanger train were significantly inconsistent. The following points are noted:

(i) The temperature of the flue gases at the exit of the reformer furnace was not measured (sensor defective)

(ii) The measured air rate (12600 Nm^3/hr) does not correspond to that calculated from the fuel rate (1305 kg/hr) and the reported excess air. Using the C/H mass ratio reported for the fuel (=8.3), the reported 20% excess air and the indicated fuel rate, the air flowrate is calculated to be 16914 Nm^3/hr which is far in excess of the 12600 Nm^3/hr reported.

(iii) In the reformer reactor, for any given outlet temperature of the reactants, the extent of reaction is fixed by specified equilibrium conditions; hence the amount of heat required in the endothermic reforming reaction is fixed. This heat, along with the sensible heat required to bring the reactants to their outlet temperature is now known. The combustion of Bunker-C oil on the outside of the reactant tubes must provide this heat and this heat plus whatever heat loss (call that the total heat load for the reactor) will be given up by the flue gas between its adiabatic flame temperature and the mixed gas and this, in turn, means that the exit flue gas temperature will be higher for any given reactant outlet temperature. This heat balance must apply independent of the

model used to calculate the heat transfer rate. This implies that if the firing rate, the percent excess air, the atomizing steam flowrate relative to the fuel rate and the inlet temperature of all streams are known, the exit temperature is fixed by the heat requirements on the furnace.

This calculated flue gas exit temperature with the indicated fuel rate is much too high to provide the observed flue gas temperatures at the exit of the heat exchangers in the heat exchanger train downstream of the reactor. It is observed, however, that the flue gas analysis on a dry basis corresponds very closely to that expected with the C/H mass ratio (=8.3) given for the fuel and the reported operation at 20% excess air. Table 9.2 shows the comparison between the reported analysis and the analysis from the calculated flue gas flowrate for 20% excess air and C/H mass ratio for the fuel of 8.3.

Table 9.2 Flue Gas Analysis

Components	Calculated flue gas (with 20% excess air and C/H (fuel)=8.3)			Reported Flue gas analysis	
	flows lbmole/hr	wet analysis	dry analysis	wet analysis	dry analysis
H_2O	150.21924	0.2056	-	0.1199	-
CO_2	78.25687	0.1071	0.1348	0.1136	0.1291
N_2	480.96320	0.6582	0.8285	0.7340	0.8340
O_2	21.30850	0.0292	0.0367	0.0325	0.0369

The wet analysis indicates too little water, but this could arise from not keeping the sample bottle at a temperature above the dew point of the sample. The fact that the measured carbon dioxide level is slightly low may also arise because some of it was absorbed in the condensed water vapour.

(iv) Calculating the flue gas flowrate from heat balances on the heat exchangers in the flue gas heated exchanger train, assuming the flue gas composition as calculated is correct, the flowrates of other streams and the measured temperatures to be exact, leads to wide variations in the flue gas flowrates, thus indicating that measured temperatures and/or flowrates are subject to considerable error. These calculations did indicate, however, that the flue gas flowrate had to be considerably less than that arising out of the measured fuel oil flowrate.

9.3 General Background-Parameters Estimation in Models

The problem now is: given these plant data which are subject to considerable error, what are the parameters in the models?

Generally, these parameters are estimated so as to satisfy some criterion. That criterion is usually that the sum of squares of the experimental minus the expected values (called the residuals), weighted by the inverse of the variance of the experimental measurements is a minimum. Mathematically, for multi-response data (for which there is a variance-covariance matrix), this is expressed as

$$ss = (\underline{y} - \underline{n}(\underline{\xi},\underline{\theta})^T \underline{v}^{-1}(\underline{y}-\underline{\eta}(\underline{\xi},\theta)) \qquad (9.1)$$

where $\underline{v}$ is the variance-covariance matrix; $\underline{Y}$ is a vector of plant measurements; $\underline{\eta}(\underline{\xi},\underline{\theta})$ is the vector of experimental values from the model at experimental conditions $\underline{\xi}$; and $\underline{\theta}$ is the vector of parameter values.

In many instances in plant measurements, the variance-covariance matrix is unknown; indeed, rarely is the variance of one measurement known. This was the case here; therefore, it was assumed that were a vector of observations was made (for example, the component flows), the covariance among the measurements was zero and the relative error on each measurement was the same. Equation 9.1 may then be written as:

$$ss = \sum_{i=1}^{N} \left(\frac{y_i - \eta_i(\underline{\xi},\underline{\theta})}{y_i}\right)^2 \qquad (9.2)$$

where N is the number of components.

For instance, given the experimentally measured analysis of the outlet stream from the reformer reactor and the measured total flowrate (provides experimental component flowrates as response), the model parameters (the temperature approaches for the two reactions occurring) may be estimated by minimizing ss, as calculated from Equation 9.2. In this case, there are more experimental observations than parameters to be estimated and this criterion may be used. Similarly, the parameter, UA, may be estimated if all inlet and outlet temperatures and flowrates are known. In other instances, the number of measurements may equal the number of parameters to be estimated (for example, the pressure drop across an equipment), in which case, there are no degrees of freedom and the parameter is calculated

directly.

Another difficulty arises, however, in that in using Equation 9.2, it is assumed that all error is in the response measurements and these errors are free from any bias. This bias may arise if the experimental conditions pertaining to the test do not apply to the model. For example the measured flowrate of naphtha, steam, etc. to the reactor provides an apparent atomic flowrate of carbon, hydrogen, oxygen, etc. The atomic flowrates of carbon, hydrogen, oxygen, etc. out of the reactor may be evaluated from the measured component flowrates. If the error in these material balances on atomic species is always positive or negative then a bias will have been introduced. This suggests that some means for correcting flowrates and temperatures is required to ensure that mass and heat balances pertain before parameters estimation is undertaken.

Difficulties arise in carrying out this 'correction process' as a result of

(1) Insufficient measurements were made at the time of the test to check heat and material balances

(2) The measurements contain errors of unknown, magnitude arising from:

  (a) inaccurate reading of instrument response

  (b) improper design and/or installation of instruments (for example, conduction and radiation errors in temperature sensors)

  (c) improper calibration of instruments

  (d) inconsistent measurements because plant operation is not at the steady-state condition appropriate for the feedrates and control variables.

Type (a) errors lead to small random errors; errors of type (b) (c) and (d) usually lead to gross errors.

A method of adjusting plant data to ensure closure of the heat and material balances is available and discussed in the next section.

9.4 Correction of Measured Plant Data According to Ripps

Ripps (R1) considers the problem of obtaining a consistent set of data to ensure exact closure of heat and material balances. Raw process data, since they are subject to the aforementioned errors, are usually not consistent. Ripps has provided a rational numerical method to adjust these data to make them internally consistent. His method is outlined below.

The first problem considered is the one where I measurements (vector $\underline{M}$ with m_i as the i^{-th} measurement) contain only small random errors, called measurement errors. A small random error suggests that the measurement is a good estimate of the underlying physical quantity (temperature, flow, etc.). The problem is to adjust each of the measured values, m_i, so that the altered data, x_i, satisfies mass and energy balances or some other criteria of consistency. These restrictions comprise the vector $\underline{\psi}(\underline{x})$ of length J and may be written mathematically as

$$\psi_j(\underline{x}) = 0 \ , \quad j = 1,2,3,....J \tag{9.3}$$

with $\underline{x}$ representing the vector of adjusted data. The number of restrictions, J, must be less than the number of measurements, I.

Ripps limits the restrictions to linear equations of the form:

$$\psi_j(\underline{x}) = \sum_{i=1}^{I} a_{ji} x_i = 0 \qquad j=1,2.....J \tag{9.4}$$

Mass balances are easily cast in the form of Equation 9.4. Energy balances are linear only if the $\underline{x}$ is enthalpy; hence if both temperature and flow measurements are in error a non-linear equation results. In this case, the flowrates must be consistent and they must be corrected first on the basis of the mass balances. Note also that the primary measurements are usually not the flowrates or enthalpies but some measurement from which these quantities are calculated. The variances on the primary measurements, if they are available must therefore be translated to the variances on the flowrates and enthalpies when they are used later in the mathematical analysis to follow. Since many of these translating equations are non-linear, a linearization procedure for estimating the variances must be applied if the variances of $\underline{x}$ are not known directly.

In his analysis of the problem, Ripps suggests that when only small random errors occur, it is best to make the data consistent by applying the smallest possible set of corrections. This is accomplished by choosing $\underline{x}$ such that

$$x : \begin{cases} \phi(\underline{x}) = \sum_{i=1}^{I} \left(\frac{x_i - m_i}{s_i}\right)^2 = \text{minimum} \\ \psi_j(\underline{x}) = 0 \quad ; \quad j = 1,2,3.....J<I \end{cases} \tag{9.5}$$

where s_i^2 is an estimate of the variance of the quantity to be adjusted. This formulation therefore makes the correction to the measurements according to the accuracy with which they are measured (largest correction to the least accurate measurement).

To solve Equation 9.5, the restrictions are introduced into the objective function, via. Lagrange multipliers. Thus the Lagrangian is

formed

$$\underline{L} = \underline{\phi} + \sum_{j=1}^{J} \lambda_j \psi_j \tag{9.6}$$

where λ_j are the Lagrangian multipliers which are chosen to ensure that the Lagrangian is stationary with respect to the variables $\underline{x}$ and $\underline{\lambda}$. These equations become:

$$\frac{\partial}{\partial x_i} \{\phi + \sum_{j=1}^{J} \lambda_j \psi_j\} = \frac{2}{s_i^2} (x_i - m_i) + \sum_{j=1}^{J} a_{ji} \lambda_j = 0$$

or

$$\sum_{j=1}^{J} a_{ji} \lambda_j + \frac{2}{s_i^2} x_i = \frac{2m_i}{s_i^2} \qquad i = 1,2,3 \ldots I \tag{9.7}$$

and

$$\frac{\partial}{\partial \lambda_j} \{\phi + \sum_{j=1}^{J} \lambda_j \psi_j\} \sum_{i=1}^{I} a_{ji} x_i = 0 \qquad j = 1,2,3 \ldots J \tag{9.8}$$

Following Ripps, Equations 9.7 and 9.8 are cast in the form

$$\begin{bmatrix} \underline{0} & \underline{A} \\ \underline{A}^T & \underline{S} \end{bmatrix} \begin{bmatrix} \underline{\Lambda} \\ \underline{x} \end{bmatrix} = \begin{bmatrix} \underline{0} \\ \underline{b} \end{bmatrix} \tag{9.9}$$

where $\underline{\Lambda} = \{\lambda_j\}$ $\qquad i = 1,2,3 \ldots I$ (9.9a)

$\underline{A} = \{a_{ji}\}$ (9.9b)

$\underline{S} = \{\frac{2}{s_i^2}\}$ (9.9c)

$\qquad j = 1,2,3 \ldots J$

$\underline{b} = \{\frac{2m_i}{s_i^2}\}$ (9.9d)

Solution of Equation 9.9 for $\underline{\Lambda}$ and the desired corrected measurements, $\underline{x}$, is easily obtained by matrix manipulation. Ripps suggests an inversion formula which is useful in the numerical analysis of the problem which follows. He suggests the following formula for the inverse of a partitioned matrix, viz:

$$\begin{bmatrix} \underline{O} & \underline{A} \\ \underline{A}^T & \underline{S} \end{bmatrix}^{-1} = \begin{bmatrix} \underline{B}_1 & \underline{B}_2 \\ \underline{B}_3 & \underline{B}_4 \end{bmatrix} \tag{9.10}$$

where $\underline{B}_0 = -\underline{A}\,\underline{S}^{-1}\,\underline{A}^T$ (non singular) (9.10a)

$$\underline{B}_1 = \underline{B}_0^{-1} \tag{9.10b}$$

$$\underline{B}_2 = \underline{B}_0^{-1}\,\underline{A}\,\underline{S}^{-1} = \underline{B}_3^T \tag{9.10c}$$

$$\underline{B}_4 = \underline{S}^{-1}[\underline{I} - \underline{A}^T\underline{B}_2] \tag{9.10d}$$

The advantage here is that the largest matrix to be inverted is of order J, not I+J, the order of the matrix in Equation 9.10.

The second problem involves correcting measurements where some of the data exhibit gross error or, indeed, in cases where some of the data are not available. Ripps points out that the previous scheme will distribute any gross errors over all measurements and hence may make all of them differ from their true value.

To avoid this difficulty, the first procedure is modified to allow some measurements to be discarded. This is normally possible since there are usually more mass and energy balances available than are required to effect a solution to the problem. (I is still greater than J).

Mathematically, measurements are discarded by introducing a vector delta into Equation 9.5, viz:

$$x : \begin{cases} \phi(\underline{x}) = \sum_{i=1}^{I} \delta_i \left(\frac{x_i - m_i}{s_i}\right)^2 = \text{MIN} \; ; \; \delta_i = 0 \text{ or } 1 & (9.11a) \\ \psi_j(\underline{x}) = 0 \qquad j = 1,2,3....J<1 & (9.11b) \end{cases}$$

where δ_i is 0 or 1 depending on whether the measurement is discarded (gross error assumed) or retained (small random error assumed), respectively. Note that a missing measurement is like assuming a gross error. This adjustment scheme requires that the data be classified according to the size of the expected errors. With each classification of data, according to the assumption of where gross errors are suspected, the least squares criterion is evaluated. Often, the minimum in ϕ coincides with the most correct data adjustment, although this need not be the case. The final decision as to which measurement(s) is (are) in gross error should be determined in the light of the reasonableness of the results relative to experience.

Ripps provides an algorithm for computing the adjustment to the data for any set of δ_i (vector $\underline{\Delta}$) after the results for $\underline{\Delta} = \underline{1}$ have been obtained. This is summarized as follows:

Note that $\underline{S}$ is an I×I diagonal matrix, with

$$s_i = \frac{2\delta_i}{s_i^2} \qquad (9.8e)$$

and $\underline{b}$ is a column vector of length I with

$$b_i = \frac{2m_i\delta_i}{s_i^2} \qquad (9.8f)$$

Note also that the time-consuming part of the computation is in obtaining the inverse matrix. Efficiency of computing this inverse is achieved by the following analysis:

Assume the B matrices in Equation 9.10 have already been evaluated for $\underline{\Delta} = 1$, that is

$$\begin{bmatrix} \underline{O} & \underline{A} \\ \underline{A}^T & \underline{S} \end{bmatrix}^{-1}_{\underline{\Delta}=1} = \begin{bmatrix} \underline{B}_1 & \underline{B}_2 \\ \underline{B}_3 & \underline{B}_4 \end{bmatrix} \tag{9.12}$$

Let N be the number of measurements assumed to have gross errors. In the computation scheme, the indices on the measurements to be discarded are changed so that x_{I-N+1} through to x_I are the discarded measurements. This requires interchanging some of the rows and columns on both side of Equation 9.12. This means that all elements of the matrices in the matrix equation 9.12 are known numerically. All that is required are row-column interchanges. The partitions on both sides of the interchanged matrix equation 9.12 are moved (I-N) places to the right and (I-N) places down to give:

$$\begin{bmatrix} \underline{\alpha}_1 & \underline{\alpha}_2 \\ \underline{\alpha}_3 & \underline{\alpha}_4 \end{bmatrix}^{-1}_{\underline{\Delta}=\underline{1}} = \begin{bmatrix} \underline{\beta}_1 & \underline{\beta}_2 \\ \underline{\beta}_3 & \underline{\beta}_4 \end{bmatrix} \tag{9.13}$$

This separates the measurements assumed to have small errors into the upper positions. Note the numerical values of the elements of the matrices

on both sides of the equation remain unchanged; only their position has changed and these positions only depend on the value of $\underline{\Delta}$ one is using.

The value of $\underline{\alpha}_1^{-1}$ is required in the simplified algorithm. Its value is easily found from the formulas for the inverse of a partitioned matrix, namely

$$\underline{\alpha}_0 = \underline{\alpha}_4 - \underline{\alpha}_2^T \underline{\alpha}_1^{-1} \underline{\alpha}_2 \tag{9.14a}$$

$$\underline{\beta}_1 = \underline{\alpha}_1^{-1} + \underline{\alpha}_1^{-1} \underline{\alpha}_2 \underline{\alpha}_0^{-1} \underline{\alpha}_2^T \underline{\alpha}^{-1} \tag{9.14b}$$

$$\underline{\beta}_2 = -\underline{\alpha}_1^{-1} + \underline{\alpha}_2 \underline{\alpha}_0^{-1} = \underline{\beta}_3^T \tag{9.14c}$$

$$\underline{\beta}_4 = \underline{\alpha}_0^{-1} \tag{9.14d}$$

Solving for $\underline{\alpha}_1^{-1}$ yields

$$\underline{\alpha}_1^{-1} = \underline{\beta}_1 - \underline{\beta}_2 \underline{\beta}_4^{-1} \underline{\beta}_2^T$$

The elements of the $\underline{\beta}$'s for Δ=1 are available for Equation 9.10; hence it is a simple matter to obtain $\underline{\alpha}_1^{-1}$ particularly since the order of the matrix to be inverted ($\underline{\beta}_4$) is N.

When a new given value of $\underline{\Delta}$ is substituted, α_4 becomes a zero matrix but all other matrices are unchanged and hence $\underline{\alpha}_1^{-1}$ and $\underline{\alpha}_2^T \underline{\alpha}_1^{-1} \underline{\alpha}_2$ are the same. Equations 9.13 may be used to generate the new $\underline{\beta}$ matrices and then

$$\begin{bmatrix} \underline{\beta}_1 & \underline{\beta}_2 \\ \underline{\beta}_3 & \underline{\beta}_4 \end{bmatrix} = \begin{bmatrix} \underline{0} & \underline{A} \\ \underline{A}^T & \underline{S} \end{bmatrix}^{-1} \tag{9.15}$$

that is the new data adjustment matrix for the given $\underline{\Delta}$. The entire procedure of column-row interchange, repartitioning, evaluating $\underline{\alpha}_1^{-1}$

must be carried out for any new $\underline{\Delta}$, that is for any change in streams assumed to be in gross error. The adjusted data is easily obtained from

$$\begin{bmatrix} \underline{\Lambda} \\ \hline \underline{x} \end{bmatrix} = \left[\begin{array}{c|c} \underline{\beta}_1 & \underline{\beta}_2 \\ \hline \underline{\beta}_3 & \underline{\beta}_4 \end{array}\right] \begin{bmatrix} \underline{o} \\ \hline \underline{b} \end{bmatrix} \qquad (9.16)$$

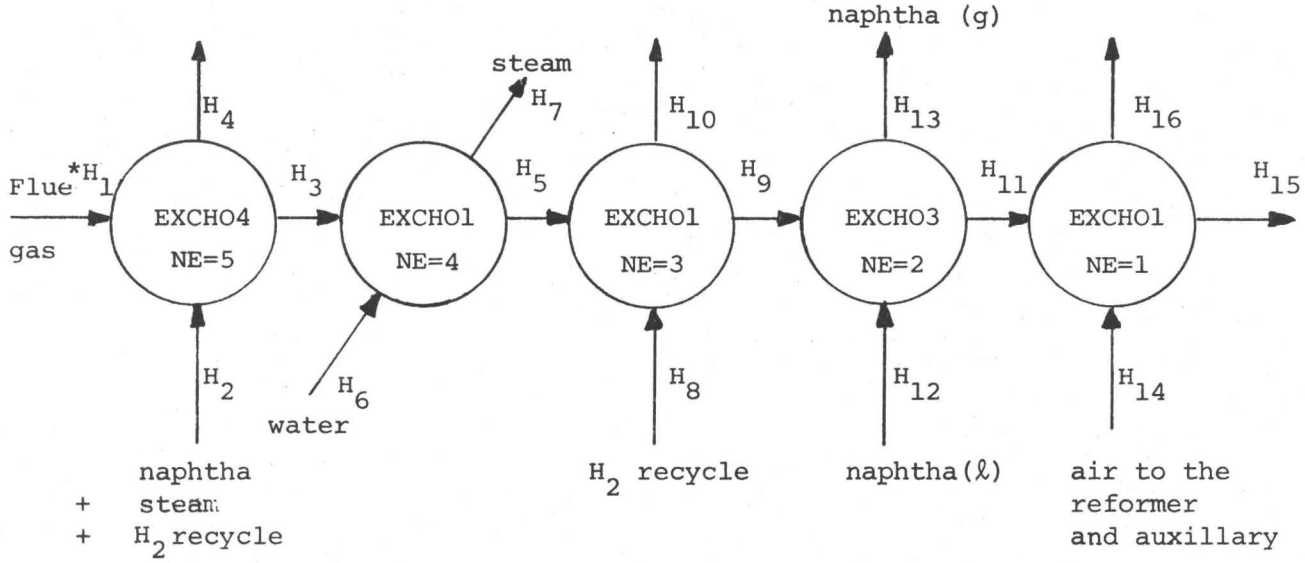
This algorithm has been used to adjust the temperature measurements in the flue gas heat exchanger train located downstream of the reformer furnace. These consistent temperatures are then used in evaluating the parameter (overall heat transfer coefficient-area product, UA) in each of the heat exchanger models. This application of Ripps' scheme is presented in the following section (9.5).

A computer program using this algorithm has been written and is available from the Department of Chemical Engineering, McMaster University.

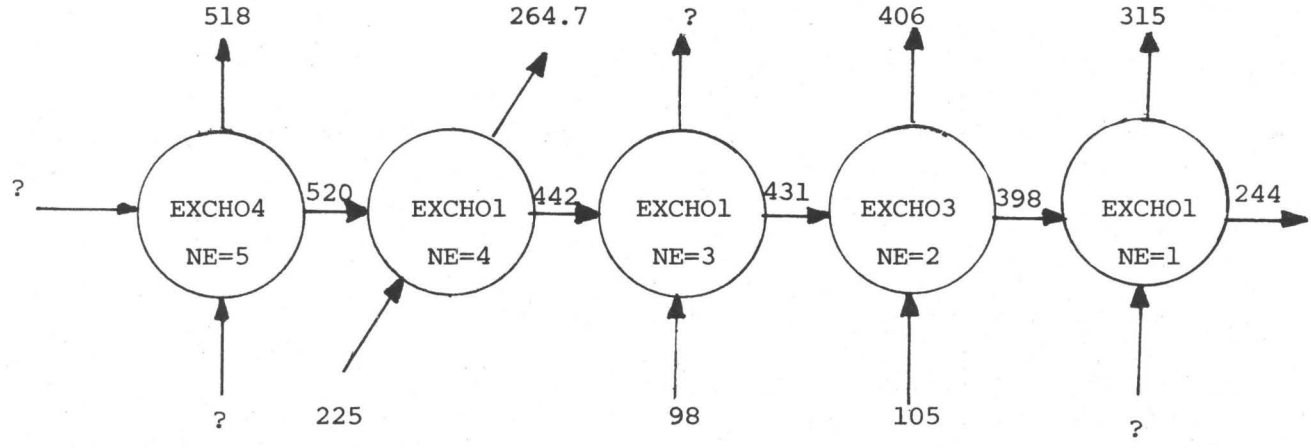
9.5 Adjustment of Temperature Measurements in the Flue Gas Heat Exchanger Train

Figure 9.1 indicates the heat exchanger network in which the flue gas is used to preheat the reactant mixture, generate steam in a waste-heat boiler, preheat the recycle gas, preheat and vaporize the naphtha feed, and preheat the combustion air, in turn. The recorded temperatures, compositions and component flowrates are shown on the figure. Note that the incoming flue gas temperature, the incoming reactant temperature, the naphtha - recycle gas mixture temperature, the incoming air temperature and the exit recycle temperature are not measured. Moreover, the flue gas flowrate is unknown and the measured air flowrate is suspected to be

Figure 9.1 Plant Data of the Flue Gas Heat-Exchanger Train



Measured Temperatures (^{o}C)



* This is the total flue gas resulting from the combustion of fuel in both the auxillary burner and the reformer furnace.

Figure 9.1 Plant Data of the Flue Gas Heat-Exchanger Train (cont'd)

Measured flows (kgmole/hr)

	flue gas	naphtha Steam H_2 recycle(2,4)	H_2 recycle (8,10)	Naphtha (12,13)	air (14,16)
H_2O(v)		844.52293	0.0622	0.	
CO_2		0.	0.	0.	
CO		0.	0.	0.	
CH4	?	0.15093	0.15093	0.	?
H_2		8.269	8.269	0.	
N_2		0.	0.	0.	
O_2		0.	0.	0.	
naphtha		16.4	0.	16.4	

steam produced : 1974 kg/hr (6)

water required (assuming reported 0.053 fraction blowdown) : 2078.6 kg/hr (7)

	flue gas corresponding to 1305 kg/hr fuel	air flowrate corresponding to 1305 kg/hr fuel	flue gas corresponding to 12600 Nm^3/hr air	air flowrate
H_2O	186.30629		138.79165	
CO_2	97.05645		72.30365	
N_2	596.50461	596.50461	444.37500	444.375
O_2	26.42742	158.56452	19.68750	118.125

in gross error, as per previous discussion (section 9.2). The other flowrates are of unknown accuracy. This presents a major problem, since the component flowrates of all streams must be known, or at least consistent with the mass balance constraints, in order to formulate linear energy balances with temperature as the only unknown (enthalpy as a function of temperature only).

To resolve this problem, the following scheme was adopted:

(a) Assume that the flowrates and composition of the reactant (steam-naphtha) mixture, the recycle gas and the naphtha feed are as measured.

(b) Assume that the temperature of the incoming reactant mixture is that calculated from an enthalpy balance of the entering steam and naphtha vapour-recycle gas mixture at the measured temperature for the naphtha vapour ($273^{o}C$).

(c) Assume the recycle gas is heated to almost the measured incoming flue gas temperature (countercurrent exchanger). Note that since the recycle gas flow is very low, any error in this temperature has a very small effect on the enthalpy content of the flue gas at the exit of this exchanger.

(d) The steam generated in the waste heat boiler was assumed to be as measured; it was assumed to be saturated at 50 kgm/cm^2 (48.392 atmospheres, $262.7^{o}C$). The incoming water flowrate was assumed to be 5.3% higher than the steam production rate to account for the reported blowdown rate (5.3% of total water flow).

(e) The flue gas flowrate presents a major problem in this analysis. Note that the flue gas arises from both the furnace and the auxillary burner located just upstream of the reactant preheater. This auxillary burner is fired at a rate to ensure that the reactants can be heated to the desired temperature (ca. 520°C) before entering the reformer. Plant data indicates that 1305 kg of fuel are fired in both the reformer furnace and auxillary burner. Plant personnel suggest that approximately 10% of the fuel is fired in the burner, but no direct measurement was made. Given the atomizing steam to fuel mass ratio as 0.089 and 20% excess air, the flue gas flowrate and composition may be calculated. Using the C/H mass ratio reported for the fuel (=8.3), this yields an air flowrate for use in the air preheater calculations. As already indicated, the flue gas analysis agrees with the 20% excess air figure.

(f) The air flowrate was assumed to correspond to that required for the fuel rate used assuming 20% excess air.

The measurements, m_i, used in Ripps' analysis correspond to the total stream enthalpy calculated from the measured temperatures and assumed flowrates. The enthalpies per unit mass for the fixed gases were calculated from Kjaer's correlations (K1) in a computer program, that of the naphtha from the liquid and vapour heat capacity data and the latent heat of vaporization, and that of water and steam from steam tables.

Since the inlet flue gas temperature was unknown, it was estimated at 850°C for preliminary calculations ; in the final analysis, it was

assumed to be in gross error. Similarly, the inlet air temperature was assumed to be 110°C with gross error. Experience with the Ripps' adjustment scheme suggested that stream 1,3,10 and 14 contained gross errors.

The same variance of 1 was given to the enthalpy of each stream (resulting from temperature measurements) at first; after the above results, the variance of the enthalpy of streams 2,4,6,7,12,13 and 16 was made much smaller (0.001) in order to keep these values as close as possible to the assumed or measured values since they do not show any particular inconsistency.

Assuming that the temperature of stream 10 (or enthalpy) contains gross errors, leads to an impossible situation (temperature of stream 10 larger than the inlet temperature of the flue gas) since a large correction in enthalpy is then applied to this stream. This arises because the mass flowrate of this recycle stream is relatively very small and any correction in enthalpy leads to a very large temperature change. On the other hand, because of the small flowrate, any error in outlet temperature or enthalpy will have a very small effect on the overall heat balance around the flue gas heated exchanger train. Therefore, to circumvent this problem, it was assumed that the outlet temperature of the recycle stream was only slightly lower than the inlet flue gas temperature, and that both the inlet and outlet temperatures of this recycle stream were accurate measurements; this was handled mathematically by giving them very small variances ($s^2 = 0.001$).

In this analysis, the amount of fuel fired in both the reformer

furnace and the auxillary burner (located downstream of the reformer furnace) is a key variable since it determines directly the total amount of flue gas in the flue gas heated exchanger train and the flowrate of air which must be heated by this flue gas in one of the heat exchangers. Flue gas flowrates corresponding to the indicated 12,600 Nm^3/hr and, in a second analysis, to the 1305 kg/hr reported fuel rate both indicated a fairly large sum of squares of residuals, thereby suggesting that both reported values seem to be in gross error.

To reconcile this problem, an independent measurement of mass flowrate of flue gas along with its composition is required. Since this measurement was not available, the temperature measurements had to be used in some way to estimate a reasonable flue gas flowrate and a flue gas temperature at the inlet to the heat exchanger train. This was done by including the reformer reactor furnace in the analysis, since what is happening there influences the mass flowrate of flue gas and its exit temperature from the furnace.

Unfortunately, another problem arises: some fuel is fired in the auxillary burner located downstream of the furnace but before the first heat exchanger. Therefore, the exit temperature from the furnace is lower than the actual temperature of the flue gas at the entrance to the heat exchanger train. The amount of fuel fired in the auxillary burner is unknown and has to be estimated.

Therefore to effect a reasonable solution to this problem, the following procedure was followed:

(i) Assume a fuel firing rate for the furnace and auxillary burner. Some limits on this estimate may be found as follows: the fuel firing rate in the furnace must provide an exit flue gas temperature above the reactant/product temperature from the reactor (810^{o}C). Including an estimated 10% heat loss in the furnace (as suggested in Perry (P1)) and knowing the heat load for the endothermic reaction and sensible heat requirements, the minimum fuel rate is calculated to be 950 kg/hr. An initial Ripps' analysis suggests that the maximum flue gas temperature need be no more than 900^{o}C and this provided an estimate of the maximum firing rate for both the furnace and auxillary burner of about 1170 kg/hr.

(ii) Calculate the flue gas flowrate based on the C/H ratio of the fuel (=8.3), complete combustion, 20 percent excess air and the specified atomizing steam-to-fuel ratio (=0.089). Calculate the amount of air required for this fuel firing rate.

(iii) Carry out the Ripps adjustment procedure using enthalpy evaluated from the flue gas flowrate calculated in (ii) and the measured temperatures. The calculated air flowrate is used in the final exchanger. This analysis included the following aspects: as previously, the temperature of the flue gas at the inlet to the heat exchanger train was not known and therefore was assumed to be in gross error. Again, the inlet water and steam temperatures in

the waste heat boiler were assumed to be measured accurately. The recycle stream was assumed to be heated to 430°C and both inlet and outlet temperatures were assumed to have very small variance (for reasons already indicated). The naphtha and steam flowrates were assumed to be correct; also, the temperatures of the naphtha liquid, vapour and naphtha/ steam mixture were assumed to be accurate (small variance). In addition to stream 1, streams 3 and 14 were assumed to be in gross error since these were shown to be in the initial analysis.

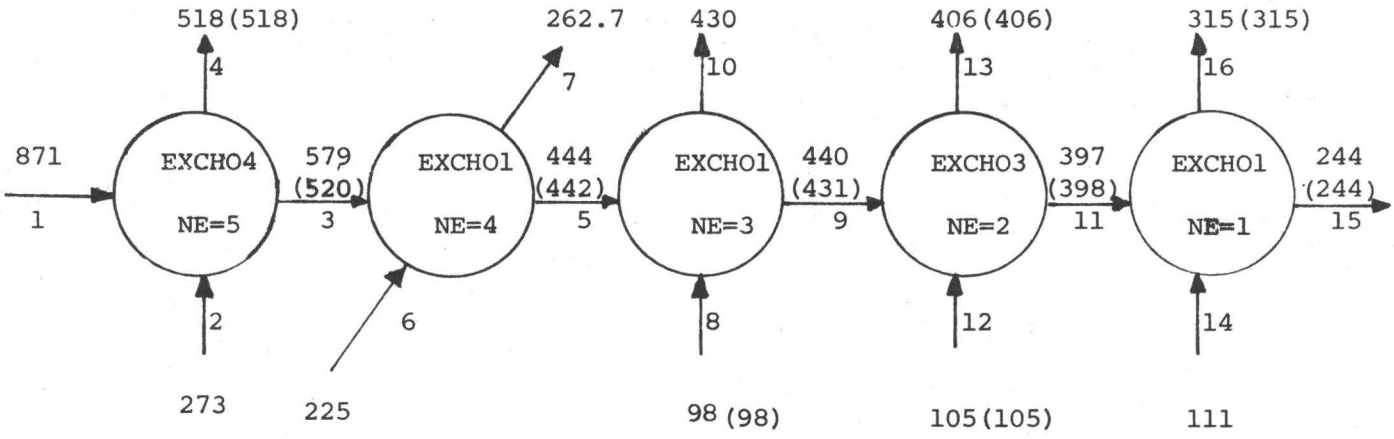
(iv) After the adjusted temperatures were known for any assumed fuel flowrate, the sum of squares of the differences between the measured and adjusted temperatures was evaluated for those streams assumed to contain small error.

(v) This procedure was repeated for a number of fuel rates and it was assumed that that fuel rate which gave the least value of the sum of squares was the correct value.

It was fairly easily determined (without the aid of an optimizing routine) that a total fuel firing rate of 1100 kg/hr was the most likely and this yielded the temperatures shown on Figure 9.2. The measured values are shown in parenthesis.

This fuel must be distributed to both the furnace and the auxillary burner since it is known that the auxillary burner is operating under the conditions pertaining to the test. Although the operator estimate 10% of the fuel goes to the auxillary burner (based on relative valve openings), such a distribution would lead to an inordinately low exit flue

Figure 9.2 Corrected Plant Data from Ripps' Analysis of the Flue Heat Exchanger Train



Calculated flows (kgmole/hr)

	Flue gas	Air
H_2O	157.35741	-
CO_2	81.97551	-
N_2	502.86551	502.86551
O_2	22.06790	133.67311

Corresponding total fuel requirements 1102.kg/hr(instead of the reported 1305 kg/hr)

Air requirements 14250 N_m^3/hr (instead of the reported 12600 N_m^3/hr)

gas temperature from the furnace (below 810°C). It was arbitrarily assumed that the exit temperature of the furnace flue gas was 830°C. This meant that 1050 kg/hr of fuel would be fired in the furnace and the remaining 50 kg/hr would raise the flue gas from 830°C to the required 871°C.

With these corrected temperatures and flowrates which produce consistent heat and mass balances, the parameters in each of the heat exchanger models and in the reformer reactor can be evaluated directly.

9.6 Parameter Estimation in Reactor Models

As indicated in section 9.3, the parameters to be estimated in the reactor models are the approach-to-equilibrium temperature. This means that although the actual temperature of the reactant is known (measured as 810°C, 442°C and 254°C in the reformer reactor, the high temperature water gas shift reactor and the low temperature water gas shift reactor, respectively), the composition of the outlet reactant stream corresponds to that which would be at equilibrium at another temperature. This procedure allows for the fact that the extent of reaction is less than the maximum possible and in an appropriate way allows for the reaction kinetics in a specific reactor. Because the reforming reaction of methane is strongly endothermic, the equilibrium constant is less at lower temperatures and therefore the approach temperature must be substracted from the actual exit temperature. The reverse is true for the water gas shift reaction. The magnitude of the approach temperature will indicate the extent to which the reaction is controlled by reaction kinetics, which, in turn, provides some measure of the activity of the catalyst.

This approach-temperature parameter for each reaction in each reactor was estimated from base case data (which provided the component flowrates to and from each reactor). The best parameter value was taken as that which minimized the sum of squares of the residuals (Equation 9.2). This minimum was obtained through the use of the Nelder and Mead (N1,K3) optimization technique based on the Simplex algorithm. The computer program was taken from Kuester and Mize (K3).

The application of this procedure to the specific reactors and the results from it are discussed below. In each case before the parameters were estimated, a material balance was performed on the plant data to ensure that the measured component flowrates were consistent. This meant doing a carbon, hydrogen and oxygen balance. The only minor inconsistency occurred with the reformer reactor. By adjusting the naphtha feed to 1320.9 kg/hr from the reported 1320 kg/hr this inconsistency could be rectified.

9.6.1 The Reformer Reactor (REAC01,NE=10)

Plant data suggested that the thermal and catalytic cracking of naphtha to methane was complete; therefore, only the methane reforming reaction was considered. In addition, the water gas shift reaction occurs to some extent. Hence, there are two approach temperatures which need to be estimated. This was done simultaneously by the Nelder and Mead program, with the results shown in Table 9.3.

Table 9.3 Parameter estimation in the Reformer Reactor (REAC01,NE=10)

Approach temperature for the methane/steam reaction = -58.5°C

Approach temperature for the water gas shift reaction = -5.1°C

	Outlet Component flows (kgmoles/hr)	
	Calculated	Plant Data
$H_2O(v)$	687.85463	686.91964
CO_2	67.97272	67.92991
CO	20.82094	20.80982
CH_4	3.96612	3.97009
H_2	262.3185	262.14018

The negative approach temperature for the methane reaction is consistent with that expected. The negative approach temperature for the water gas shift reaction is not consistent with that expected but probably arises because of the simultaneous reactions and the fact that the equilibrium is being approached from a lower temperature (assuming the outlet gas temperature is the maximum in the reactor).

9.6.2 The Water Gas Shift Reactors (REACO2,NE=12,17)

The component flowrate data were found to give consistent material balances. It was found, however, that methane was produced in the high temperature reactor and then was reacted in the low temperature one. The methanation reaction was not modeled, however, since the extent of reaction was very small. Instead it was assumed that a constant fraction of carbon monoxide was reacted to produce methane (or vice versa) and that this fraction occurred first before the equilibrium of the water gas shift reaction was considered. This fraction is a parameter to be estimated and was assumed to be constant under all conditions. Because the effect is small, doing so introduces very little error in the model. The approach-to-equilibrium temperature and the fraction of carbon monoxide produced by the methane/steam reaction were estimated via Nelder and Mead optimization technique as described earlier. The results are shown in Table 9.4a and 9.4b.

The approach temperatures are very large thus indicating the possibility of low catalyst activity in these reactors or that the reactors are underdesigned for the particular production rate employed.

Table 9.4 Parameter Estimation in the Water Gas Shift Reactors

(REACO2, NE=12.17)

(a) High Temperature Reactor (REACO2,NE=12)

Approach temperature for the water gas shift reaction = 105.8°C

Fraction of carbon monoxide produced by the methane/steam reaction = 0.03025

	Outlet Component Flows (kg moles/hr)	
	Calculated	Plant Data
$H_2O(v)$	676.68956	675.75982
CO_2	79.25775	79.20000
CO	9.41595	9.42009
CH_4	4.08608	4.09018
H_2	273.24367	273.05982

(b) Low Temperature Reactor (REACO2,NE=17)

Approach temperature for the water gas shift reaction = 91.6°C

Fraction of carbon monoxide produced by the methane/steam reaction = -.05609

	Outlet Component Flows (kg moles/hr)	
	Calculated	Plant Data
$H_2O(v)$	668.52259	667.00000
CO_2	87.19552	87.14018
CO	1.70739	1.70982
CH_4	3.85687	3.85982
H_2	281.86905	281.67991

9.6.3 The Methanator Reactor (REACO3,NE=29)

The exit gases from the methanator were found to be at equilibrium at the adiabatic temperature calculated for the reactor (see Table 9.5). The exit temperature, however, was measured to be somewhat less than this adiabatic temperature. This could arise as a result of a gross measurement error or heat loss. Because of previous experience with temperature sensors in this plant, it is suspected that the measurement is in error; however, for lack of better information, it was assumed that this resulted from heat loss. Hence, this constant temperature difference (of 17.55^{o}C) was subtracted from the adiabatic temperature to account for heat loss. It was assumed constant under all operating conditions. (The same case applied to the water gas shift reactors,where the calculated outlet temperatures were slightly higher than the measured ones.)

Table 9.5 Calculated and Measured Flows out of the Methanator Reactor (REACO3,NE=29)

	Calculated	Plant Data
$H_2O(v)$	2.09757	2.08428
CO_2	0.	0.
CO	0.	0.
CH_4	5.02964	5.05572
H_2	276.98203	276.98193
Outlet temperature	359.55^{o}C	342.^{o}C

9.6.4 The Reformer Reactor Furnace (REAC01,NE=10)

As indicated, the reformer furnace was modeled as a completely mixed reactor and hence the flue gas temperature in the furnace was assumed equal to the exit temperature. This created a problem for estimating effective convective heat transfer coefficients since heat is actually transferred by radiation and convection and the flue gas is not at a uniform temperature. The effective overall heat transfer coefficient can be defined by the temperature differences used; however, the film coefficients must be based on actual tube wall temperatures. The tube wall temperature was measured at one point by a radiometer as being 920°C. This is much greater than the flue gas exit temperature which was evaluated previously. (Note that this temperature must be close to that which must actually occur). This difficulty was circumvented by arbitrarily defining the inside and outside film coefficients based on a flue gas temperature which was 125°C higher than the calculated flue gas exit temperature. This temperature difference was assumed a constant parameter in the model. Note that it is only applicable when the desired production is much more than normal and/or the furnace firing rate is increased to such an extent that the desired outlet reactant temperature cannot be met without increasing the tube wall temperature above the 950°C maximum.

9.7 Parameter Estimation in the Heat Exchanger Models

As indicated in Section 9.3, the overall heat transfer coefficient-area product UA is the parameter to be estimated in heat exchangers that will allow the heat given up by one stream to be transferred to the other stream. A Newton convergence method was used to iterate on UA (the program

is listed in Figure 9.3); the program tests the calculated outlet temperature against the plant value (or corrected value) and chooses UA to make the calculated outlet temperature equal to the desired value to within some tolerance (0.1% relative error). The results of this procedure are indicated in Table 9.6.

Table 9.6 Overall Heat Transfer Coefficient-Area Product (UA) in Each Model

Equipment number	Model	UA (Kcalories/fr^{o}C)
1	EXCHO1	8711
2	EXCHO3	2275
3	EXCHO1	198.3
4	EXCHO1	3648
5	EXCHO4	7722
13	EXCHO1	197.3
14	EXCHO1	13703.
15	EXCHO1	7661.
19	EXCHO2	16850.
21	EXCHO2	8369.
23	EXCHO2	13054.
27	EXCHO2	17156.
30	EXCHO1	16505

Figure 9.3 Program Listing of Newton Convergence method

```
C
CCC      NEWTON CONVERGENCE TO DETERMINE UA
CCCCC
CCCCC    COMMON AND/OR DIMENSION STATEMENTS OF THE HEAT EXCHANGER MODEL
CCC      ARE INCLUDED HERE
C
         ...
         DIMENSION DF(2),U(2)
         MSW=1
         NSW=0
         UA=EN(16)
         DO 2100 IJJ=1,50
         ...
C
CC       MAIN BULK OF THE HEAT EXCHANGER PROGRAM IS INCLUDED HERE
C
         ...
         U(MSW)=UA
         DF(MSW)=SO(1,4)-436.0
         WRITE(6,3333) IJJ,UA
 3333 FORMAT(/,≠ AFTER≠,I3,≠ ITERATIONS, UA = ≠,PE12.5,/)
         IF(ABS(DF(MSW)/436.0).LE.0.001) GO TO 2110
         IF(MSW.EQ.1) GO TO 2152
         IF(NSW.EQ.1) GO TO 2151
         SLOP=(DF(1)-DF(2))/(U(1)-U(2))
         NSW=1
 2151 UA=U(2)-DF(2)/SLOP
         GO TO 2100
 2152 IF(DF(MSW).GT.0.) UA=UA+0.05*UA
         IF(DF(MSW).LT.0.) UA=UA-0.05*UA
         MSW=2
 2100 CONTINUE
 2110 CONTINUE
         ...
C
CC       REST OF THE HEAT EXCHANGER MODEL
C
         ...
```

A problem arose in estimating the parameter for the cooler-condenser (EXCHO2). In every case, the heat balances across the exchangers were not consistent. For lack of better information, it was assumed that

(i) The gross error occurred in measuring outlet temperature of the condensing stream.

(ii) The measured inlet and outlet temperatures and the flow-rate of the water in the economizers were exact.

(iii) The condensing gas had its water vapour content saturated at the calculated outlet temperature.

(iv) The inlet temperature of the condensing gas was that calculated from the previous exchanger.

It was then a matter of calculating the outlet temperature of the exit gas stream to ensure that the heat given up by the condensing stream was the same as that taken up by the water. This was done through the use of model EXCHO2. When the correct outlet temperature was obtained, the UA was easily calculated.

In the case of the reboilers on the desorption column, this heat balance was not possible and the measured temperatures had to be assumed correct. Similarly, the temperature measurements on the final cooler-condenser in the train had to be assumed correct.

9.8 Parameter Estimation for Pressure Drop Calculation in Each Model

The parameter to be estimated for the pressure drop calculation in each model is the constant factor K in equation 8.46. This was done by setting the inlet and outlet pressures of each equipment to the measured

values, each model calculated the outlet flows and temperatures corresponding to the base case data; subroutine CALFAC(FACP) was then called to determine the constant factor K (called FACP in the computer program) from equation 8.46 as follows

$$K = \frac{P_T \ \Delta_P}{W_M^2 \ M_{ave.} \ RT} \tag{9.17}$$

where W_M is the average total molar flowrate, $M_{ave.}$ is the average molecular weight of the flowing gas; T and P are the average temperature and pressure, respectively. Subroutine CALFAC is listed in Appendix E. The calculated pressure drop coefficients K for various equipment or groups of equipment (when PDRØP1 is self-standing) are listed in Table 9.7. Note that only the pressure drop in the main hydrogen stream needs to be calculated.

The GEMCS data set and the GEMCS converged solution (as part of the GEMCS-SYMBØL executive) of the Naphtha Reformer Plant simulation are given in Appendix F. Only the base case data simulation was studied. There has been no attempt to answer the questions posed for the actual simulation. Once this modular simulation was working well for base case data, it was used to build the combined executive and the time-factor did not permit further study of the Naphtha Reformer plant in itself in attempt to answer the questions posed for this simulation.

Table 9.7 Constant Factor K for Pressure Drop Calculation in Models

Equipment number	Model	$K*10^{-10}$ $\frac{atm.\ hr^2}{lit\ kg}$.
8	PDROP1	1176.06
5	EXCHO4	0.59346
10	REACO1	0.96141
11	EXCHO1	0.32557
12	REACO2	0.43544
16	PDROP1	0.57667
17	REACO2	0.55579
19	EXCHO2	0.03501
21	EXCHO2	0.04991
23	EXCHO2	0.12568
26	ABSO1	1.75460
27	EXCHO2	3.083
29	REACO3	3.45009

10. CONCLUSIONS AND RECOMMENDATIONS

In the first part of this report, a Combined GEMCS-SYMBØL executive system has been developed for the simulation of steady-state processes. The system and the procedure for its use was based on previous requirements which arose from one experience, - the Alcan Bayer process simulation. Although the experience with the test case (Naphtha Reforming Plant) proved that the system functions satisfactorily, additional experience with it may suggest additional requirements for the combined system. Moreover, experience may suggest more efficient ways to use the system. For example, the number of GEMCS iterations (or loops) employed before transferring to SYMBØL, or the tolerance on the convergence criterion in the GEMCS system should be investigated to determine optimum use of the system.

Because of its relative simplicity, particularly as related to constraints and number of recycle streams, the Naphtha Reforming Plant was probably not a good test case to test the power of the combined system. It showed, however, that the very powerful convergence promoter (CVGP1) coupled with linear control models was superior to the combined system in effecting a converged simulation. On the other hand, the real power of the combined system would probably be demonstrated in the Bayer Plant simulation where a solution was shown not to be possible using the GEMCS system alone (H1). This simulation was not attempted here because the large stream information vector (25 elements) of the original simulation must be reduced in order for it to be used with the available computer core storage. In this case, this stream list may be reduced to about twelve elements without affecting the answers to the questions

posed for the simulation. It still requires a major effort and is recommended as a future study and test of the system. Such a study would provide valuable experience in using the combined system and perhaps dictate additional requirements.

In the combined system as it is now constituted, the coefficients for the SYMBØL equations are generated from the GEMCS stream information only. Experience may prove that such coefficients should be obtained more effectively from the equipment vectors (EN vectors) of the GEMCS modules or alternatively the equipment parameters in the modules should be changed after the SYMBØL flowrates are calculated. This could be achieved by 'translator' modules, which could be included in the GEMCS calculation order list. It is felt, however, that a good basic executive system is now available and it has enough flexibility that additional features, such as indicated above, may be readily incorporated.

REFERENCES

B1. Bayley, M., (1976), User's Guide for the CDC6400 - I.P.A.C.S. (Information Processing and Academic Computing Services), Edition 1, McMaster University, Hamilton.

C1. CONTROL DATA (1973), CYBER 70 Computer Systems Models 72, 73, 74, 76, 7600 Computer Systems, 6000 Computer Systems, FORTRAN Extended Reference Manual. Version 4, Publication No. 60305600, Control Data Corporation, Sunnyvale, California.

C2. CONTROL DATA, (1969), 6400/6500/6600 Computer Systems, FØRTRAN Reference Manual, Publication No. 60174900, Control Data Corporation, Sunnyvale, California.

C3. Crowe, C.M., Hamielec, A.E., Hoffman, T.W., Johnson, A.I., Shannon, P.T., Woods, D.R., (1971, Chemical Plant Simulation, An Introduction to Computer-Aided Steady-State Process Analysis, Prentice Hall, Toronto.

C4. Crowe, C.M., and Orbach, O., (1971), Convergence Promotion in the Simulation of Chemical Processes with recycle - The Dominant Eigenvalue Method, Can. J. Chem. Eng., 49, 509-13.

C5. Crowe, C.M., and Nishio, M., (1975), Convergence Promotion in the Simulation of Chemical Processes - The General Dominant Eigenvalue Method, A.I. Ch. E., 21, p. 528-33.

H1. Hoffman, T.W., and Hutchison, H.P., (1975), The Simulation of the Bayer Process for Extracting Alumina from Bauxite Ore, Symposium Computers in the Design and Erection of Chemical Plants, Karlovy Vary, Czechoslovakia.

H2. Holmes, M.J., and Van Winkle, M., (1970), Ind. Eng. Chem. 62, No. 1., 21.

H3. Hutchison, H.P., (1975), Preliminary User's Guide to the SYMBØL Programs, University of Western Ontario, London, Canada.

H4. Hutchison, H.O., (1974), Plant Simulation by Linear Methods, Trans. Instn. Chem. Engrs., Vol. 52, p.287-90.

J1. Johnson, A.I., and Associates, (1971), GEMCS-General Engineering and Management Computation System., Dept of Chemical Engineering, McMaster University Hamilton (1970), Second Printing at the University of Western Ontario.

REFERENCES (cont'd.)

J2. Johnson, A.I., and Associates, (1968), Chemical Plant Simulation, A manual for the Digital Computer Simulation Programs MACSIM, GEMCS, Department of Chemical Engineering, McMaster University, Hamilton,

K1. Kays, W.M., and London, A.L., (1955), Compact Heat Exchangers, McGraw Hill, New York.

K2. Kjaer, J., (1963), Thermodynamic Calculations on an Electronic Digital Computer, Holder-Topse, Hellerup, Denmark.

K3. Kuester, J.L., and Mize, J.H., (1973), Optimization Techniques with FØRTRAN, McGraw Hill, New York.

N1. Nagiev, M.F., (June 1957), Material Balance in Complex and Multistage Recycle Chemical Processes, CEP, 53, 297.

N2. Nagiev, M.F., (1964), The Theory of Recycle Processes in Chemical Engineering, Pergman Press, London.

N3. Nelder, J.A., and Mead, R., (1964), A simplex Method for Function Minimization, Computer J., 7, 308-313.

P1. Perry, R.H., and Chilton, C.H., (1973), Chemical Engineers' Handbook, 5th Edition, McGraw Hill, New York.

R1. Reid, R.C., and Sherwood, T.K., (1958), The Properties of Gases and Luquids, McGraw Hill, New York.

R2. Ripps, D.L., (1965), Adjustment of Experimental Data, Chem. Eng. Symp. Series, Process Control and Applied Mathematics, 55, Vol. 61.

APPENDIX A

PROGRAM LISTINGS OF THE COMBINED GEMCS-SYMBØL

EXECUTIVE SYSTEM AND NOTES

Main ØVERLAY

Primary ØVERLAY (READA,1,0)

Notes on the use of subroutine DLOAD1 (and DLOAD2)

SUBROUTINE DLOAD2

SUBROUTINE DISKIO

SUBROUTINE SLIST

Primary ØVERLAY (GEMCS,2,0)

Notes on the Use of Subroutine CVGP1 and
CØNTL2 (or CØNTL3)

SUBROUTINE CVGP1

SUBROUTINE CØNTL3

SUBROUTINE DGELS

SUBROUTINE MODULE (NT)

SUBROUTINE STREAM(S)

Main ØVERLAY (SYMBOL,3,0)

Subroutines DATAIN,CFLØW,CFRAC,CONTL4,FEEDBK,
SCØMP,SFRAC,STOICF,RATIØ,VALUES,STINFO
and STR are explained and listed in Chapter 3

Subroutine TRGB, and the associated subroutines setting the system of linear equations to be solved such as SETTRI,SETTØT,SETCØM, SETMIX,SETVAL,SETRAT,SETT,CHKCMP,CHKLIM and ERREP are considered proprietary by their creators and for that reason, they are not included here. Copies of these may be obtained from (a fee will probably be required)

Mr. H. P. Hutchison
Lecturer,
Dept. of Chemical Engineering,
Cambridge University,
Cambridge, U.K.

```
      OVERLAY(COMBIN,0,0)
      PROGRAM TST (INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7=PUN
     1CH,TAPE20=0,TAPE21=0,TAPE22=0)
C
CCCC   COMBINED GEMCS - SYMBOL EXECUTIVE -- SEPTEMBER 1976
C
CCCC   COMMON DECLARATIONS -- SEPTEMBER 1976
C
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,NIT,SN(72,16)
      COMMON/D/KOUNT,JSWMX
      COMMON/SPEC/LLL,MMM
      III=72
      NCOUNT=0
      READ(5,*) NCASE
      DO 20 I=1,NCASE
      PRINT 101,I
  101  FORMAT(1H1,10X,4HCASE,1X,I5//)
      CALL SECOND(TSTAR)
      CALL OVERLAY(5HREADA,1,0)
      DO 10 J=1,999
      CALL OVERLAY(5HGEMCS,2,0)
      CALL OVERLAY(6HSYMBOL,3,0)
      IF(LOOP.EQ.999.AND.NIT.EQ.999) GO TO 15
   10 CONTINUE
   15 CALL SECOND(TEND)
      TIME=TEND-TSTAR
      WRITE(6,50) TIME,NCASE
   50 FORMAT(//,≠ ****** ≠,F15.4,≠  CPU SECONDS USED TO CALCULATE CASE N
     1UMBER≠,I4,≠   ******≠,//)
   20 CONTINUE
      WRITE(6,30)
   30 FORMAT(1H0,≠  ******  END OF EXECUTION OF THE COMBINED GEMCS - SYM
     1BOL SIMULATION  ******≠)
      STOP
C
CCCC   THE FOLLOWING STATEMENTS ARE ONLY DUMMY STATEMENTS, THEY ARE
CCCC   NEVER EXECUTED... THEY MUST BE SPECIFIED IN ORDER TO LOAD THE
CCCC   CORRESPONDING SYSTEM SUBROUTINES INTO THE MAIN OVERLAY
C
      BUFFER IN (20,0) (NS(1),NS(2))
      BUFFER OUT (21,0) (NS(3),NS(4))
      CALL UNIT(20)
      WRITE(6,*) NS(1)
      READ(5,101)
      REWIND 21
      END
```

```
      OVERLAY(READA,1,0)
      PROGRAM ONE
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSR,NCOUNT
      COMMON/PPDATA/ NTPC(17),LOC(17),PROP(17,16,4)
      COMMON/SBL/ KSC(10,5),KM(7,9),KSP(6,8),KSP6(1,1),KC(1,1),KR(12,6),
     1            KE(5,4),KK(15),AX(100),AMW(11),TOLERX
      COMMON/SPEC/LLL,MMM
      CALL OLOAD2
      CALL READIN
      CALL DATAIN
      BUFFER OUT (20,0)  (LLST(1),NCOUNT)
      CALL UNIT(20)
      BUFFER OUT (21,0)  (NTPC(1),PROP(17,16,4))
      CALL UNIT(21)
      BUFFER OUT (22,0)  (KSC(1,1),TOLERX)
      CALL UNIT(22)
      END
```

Notes on the Use of Subroutine DLØAD1 (and DLØAD2)

Although well defined in the user's manual,(J1) it was felt important to summarize in Table A.1 the data set format for the modular (GEMCS) simulation.

Table A.1 Preparation of the GEMCS Data Set[7]

Variable to be read	Remarks	FØRMAT
NCASE	number of cases to be run; in DLOAD1 (self-standing GEMCS). DLØAD2 does not read NCASE, it is read by the main overlay of the combined executive in free format.	I5
KRUN	run number : identity of the run for the user	I5
2 Title cards	The user identifies here (in words) the run	18A4
KPRNT(1 to 10)	printing control (2 cards) as defined in DIØAD1 and DLØAD2	5F12.5[1]
NCALC[2],NØCØMP	Number of equipment in the calculation order list, number of flow components in stream list	5F12.5[1]
LLST(1 to NCALC)	calculation order list	5F12.5[1]
MSN[6]	largest stream number on information flow diagram	5F12.5[1]
NS(1 to MSN)	vector of stream codes[3] (NS(K)=stream code of stream K)	5F12.5[1]
NSR	number of stream to be read	5F12.5[1]
SN(I,J) I=1 to NSR J=1 to JJ	each stream[4] to be read is read here, each stream starting on a new card, and of length JJ (=NØCØMP+5)	
NOE	number of equipment to be read	5F12.5[1]
EN(I)	all equipment vectors[5] are read here, each starting on a new card and of length specified by the user in EN(3). Each equipment	5F12.5

Table A.1 (cont'd)

	vector is stored into the EEN vector at position NPØINT(NE,1) (of length NPØINT(NE,2)=EN(3)) where NE is the equipment number on the information flow diagram specified in EN(1). (see note 5)	
NLP	in DLØAD2 only (for use with combined GEMCS-SYMBØL executive) - number of loop to be iterated in GEMCS before transferring to SYMBØL (large number, say 200, suggested in order to converge the plant before transferring to SYMBØL, so that any promotion taken by CVGP1 is not destroyed)	5F12.5[1]

1 - KPRNT(-), NCALC,NØCØMP,LIST,MSN,NS(-),NSR,NØE, are initially read under real number variables and the DLØAD1 (or DLØAD2) subroutine transfers them to the appropriate integer variables.

2 - If NCALC is made negative, DLØAD1 sets LØØP=999 thus going through only one iteration and in the combined executive, DLØAD2 sets both NIT and LØØP=999 meaning that only one iteration in GEMCS and only one transfer to SYMBØL (without transferring back the calculated SYMBØL to GEMCS).

3 -

Stream codes*	meaning
1	feed stream
2	cut stream
3	product stream
4	used in combined executive to represent a stream that must be saved (although not a cut stream) for calculation of SYMBØL coefficients
6	product stream
7	direct stream

* Stream with codes $\leq$ 4 are always saved in the SN table.

4 - Stream list

Element No	Element
1	GEMCS stream identifier
2	- not used in self-standing GEMCS - equivalent SYMBØL stream identifier in the combined executive
3	Total flow
4	Temperature
5	Pressure
6 to JJ(=NØCØMP+5)	flow of each component

5 - Equipment vector EN(I). EN(1 to 15) are common to all equipment modules (except for EN(4) and EN(5) which is specified by the user in typical equipment.

I	meaning of EN(I)
1	equipment number (NE)
2	equipment type (NTYPE)
3	length of EN vector
4 and 5	for use in module
6	number of input streams (maximum 4)
7 to 10	input stream identifier(s)
11	number of output streams (maximum 4)
12 to 15	output stream identifier(s)

NTYPE is the type of module to be called by subroutine MØDULE. It is suggested to use EN(4 or 5) as a printing control of any WRITE statements in the module (except maybe non-convergence statements).

6 - If MSN is made negative, then the associated output streams of each equipment will be printed throughout loop execution.

7 - Each entry in this table starts on a new card.

8 - KPRNT vector is defined as follows :

I	meaning of KPRNT(I) = 1 :
1	causes printing of NCONT,LLST,NS
2	causes printing of initial streams read
3	causes printinf of module sets (equipment vector)
4	causes printing of EN vector and input streams throughout loop execution
5	causes printing of IS, position of input/output stream to each equipment
6	causes printing of stream list
7	———
8	causes printing of SN table on entering each loop
9	causes printing of calculated SYMBØL coefficients and table of calculated of SYMBØL streams
10	causes printing of all GEMCS streams called by SYMBOL from the SN table (allows for debugging, used in the combined executive)
KPRNT(I)=0	suppresses the printing

```
      SUBROUTINE DLOAD2
C
C      UPDATED VERSION OF SUBROUTINE DLOAD1
C      FOR USE IN THE COMBINED GEMCS SYMBOL EXECUTIVE
C                                                - SEPTEMBER 1976
C
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSR,NCOUNT
C
C
C     ***** PRINTING CONTROL CHARACTERS *****
C
C     KPRNT(1)=1  CAUSES PRINTING OF NCONT,LLST,NS
C     KPRNT(2)=1  CAUSES PRINTING OF INITIAL STREAMS
C     KPRNT(3)=1  CAUSES PRINTING OF MODULES SETS
C     KPRNT(4)=1  CAUSES PRINTING OF EN VECTOR FOR SPECIFIED MODULE SET
C     KPRNT(5)=1  CAUSES PRINTING OF SI MATRIX FOR SPECIFIED MODULE SET
C     KPRNT(6)=1 CAUSES PRINTING OF STREAM VECTORS ENTERING SPECIFIED
C                MODULES (RE- PIECE OF EQUIPMENT)
C     KPRNT(8)=1 CAUSES PRINTING OF THE SN TABLE ON ENTERING LOOP
C     KPRNT(9)=1 CAUSES PRINTING OF CALCULATED SYMBOL COEFFICIENTS FROM
C                GEMCS STREAM INFORMATION AND A TABLE OF CALCULATED
C                SYMBOL STREAM FLOWS
C     KPRNT(10)=1 CAUSES PRINTING OF ALL GEMCS STREAM THAT ARE CALLED
C                 BY SYMBOL FROM THE SN TABLE TO CALCULATE SYMBOL
C                 COEFFICIENTS (ALLOWS FOR DEBUGGING ...)
C     KPRNT(-)=0 SUPRESSES ABOVE PRINTING
C
C     MSN=LARGEST STREAM NUMBER (IF NEGATED, THEN THE ASSOCIATED OUTPUT
C         STREAMS FROM EACH EQUIPMENT WILL BE PRINTED AFTER EVERY
C         CALCULATION OF EQUIPMENT
C     NCALC=NUMBER OF EQUIPMENTS IN THE CALCULATION ORDER LIST (IF
C           NEGATED, LOOP AND NIT ARE SET EQUAL TO 999 AND ONLY ONE LOOP
C           THROUGH GEMCS (PRINTING FINAL OUTPUT STREAM FROM EACH
C           CALCULATED EQUIPMENT) AND ONE PASSAGE THROUGH SYMBOL
C           WITH NO TRANSFER BACK OF SYMBOL FLOWS TO THE SN TABLE
C           ... ALLOWS FOR PRIMARY CHECK OF DATA READ IN
C
C
      DIMENSION TITLE(18), BIBLE(18)
C
C     *** NOTE*** THE DIMENSION ON ALLST MUST BE THE SAME AS LLST ALSO THE
C     DIMENSION ON ANS MUST BE THE SAME AS NS
C
      DIMENSION AKPRNT(10),ALLST(120)
       DIMENSION ANS (120)
      DO 7070 I=1,150
C      ***NOTE*** THE LIMIT ON THIS DO STATEMENT MUST CORRESPOND TO THE
C       LENGTH OF THE EN VECTOR
7070   EN(I)=0.
      DO 7073 I=1,16
      DO 7072 J=1,4
C       ***NOTE*** THE LIMIT ON THIS CO STATEMENT MUST CORRESPOND TO THE
C         DIMENSION ON SI AND SO
      SI(J,I)=0.
7072   SO(J,I)=0.
      DO 7073 IK=1,III
7073   SN(IK,I)=0.
      DO 7075 I=1,1500
 7075 EEN(I)=0.
      NCOUNT=0
      READ(5,108) KRUN
  116 FORMAT(/,1H ,*RUN NUMBER*,I5,/)
      WRITE(6,116) KRUN
      READ(5,100) TITLE
      PRINT 101,TITLE
      READ(5,100) BIBLE
      PRINT 101,BIBLE
      READ (5,112) AKPRNT
      DO 7171 I=1,10
7171   KPRNT(I)=AKPRNT(I)
C
C     REPORT ON PRINTING EXPECTED
C
```

```
      IF(KPRNT(1)-1)2,1,2
    1 WRITE(6,103)
    2 IF(KPRNT(2)-1)4,3,4
    3 WRITE(6,104)
    4 IF(KPRNT(3)-1)6,5,6
    5 WRITE(6,105)
    6 CONTINUE
C
C     READ NO. OF MODULES IN CALCULATION ORDER, AND NO. OF COMPONENTS
      READ (5,112) ANCALC,ANOCOM
      NCALC=ANCALC
      NOCOMP=ANOCOM
      IF(KPRNT(1)-1) 12,11,12
   11 WRITE(6,107)NCALC,NOCOMP
   12 CONTINUE
      IF(-NCALC)510,509,509
  509 NCALC=-NCALC
      LOOP=999
      NIT=999
      GO TO 511
  510 LOOP=1
      NIT=1
C
C     READ CALCULATION ORDER
C
511    READ (5,112) (ALLST(I),I=1,NCALC)
      DO 7172 I=1,NCALC
7172   LLST(I)=ALLST(I)
      IF (KPRNT(1)-1)14,13,14
   13 WRITE(6,109)(LLST(I),I=1,NCALC)
   14 CONTINUE
C
C     READ STREAM CODES,MSN IS THE MAXIMUM STREAM NUMBER, IF NEGATIVE
C     OUTPUT STREAMS WILL BE PRINTED DURING EXECUTION
C
      READ(5,112) AMSN
      MSN=AMSN
      IF (-MSN)520,519,519
  519 MSN=-MSN
      ISP=1
  520 IF(KPRNT(1)-1)16,15,16
   15 WRITE(6,110)MSN
   16 READ(5,112)(ANS(I),I=1,MSN)
      DO 7175 I=1,MSN
7175   NS(I)=ANS(I)
      IF(KPRNT(1)-1)18,17,18
   17 WRITE(6,109)(NS(I),I=1,MSN)
   18 JJ=NOCOMP+5
      READ(5,112) ANSP
      NSP=ANSP
      IF(KPRNT(2)-1)20,19,20
   19 WRITE(6,111)NSP
   20 DO 530 I=1,NSP
      READ(5,112)(SN(I,J),J=1,JJ)
      DO 3333 J=1,JJ
 3333 IF(SN(I,J).LE.0.) SN(I,J)=+0.
      IF(KPRNT(2)-1)22,21,22
   21 WRITE(6,113)(SN(I,J),J=1,JJ)
      WRITE (6,115)
   22 CONTINUE
  530 CONTINUE
C
C     READING MODULE SETS,NOE IN NUMBER
C
      READ(5,112) ANOE
      NOE=ANOE
      IF(KPRNT(3)-1)24,23,24
   23 WRITE(6,114) NOE
   24 DO 540 I=1,NOE
C     LIMIT ON DO LOOP MUST BE EQUAL TO DIMENSION OF EN VECTOR
      DO 541 IZ=1,150
  541 EN(IZ)=0.
      READ(5,112)(EN(N),N=1,5)
      NN=EN(3)
      NCOUNT=NCOUNT+NN
      READ(5,112)(EN(N),N=6,NN)
      IF(KPRNT(3)-1)26,25,26
```

```
   25 WRITE(6,113)(EN(N),N=1,NN)
      WRITE (6,115)
   26 MM=EN(1)+.001
      NPOINT(MM,1)=NCOUNT-NN
      NPOINT(MM,2)=NN
  540 CALL DISKIO(2,MM )
      IF(KPRNT(5).GT.0) GO TO 766
      GO TO 768
766    DO 767 I=1,NOS
767    WRITE(6,106) (NPOINT(I,J),J=1,2)
  768 CONTINUE
      READ(5,112) ANLP
      NLP=ANLP
C     FORMAT STATEMENTS FOR DLOAD
C
  100 FORMAT (18A4)
  101 FORMAT(18A4)
  102 FORMAT (10I1)
  103 FORMAT (1H0,22HPRINTING NCONT,LLST,NS)
  104 FORMAT (1H0,24HPRINTING INITIAL STREAMS)
  105 FORMAT (1H0,20HPRINTING MODULE SETS)
  106 FORMAT(2I5)
  107 FORMAT(1H0,I5,12H MODULE SETS,I5,11H COMPONENTS//
     124H CALCULATION ORDER LIST-)
  108 FORMAT (10I5)
  109 FORMAT (1X,10I5)
  110 FORMAT (1H0,I5,22HSTREAM CODES ARE READ-)
  111 FORMAT (1H0,I5,25HINITIAL STREAMS ARE READ-)
  112 FORMAT(5F12.5)
  113 FORMAT (1X,5F15.5)
  114 FORMAT (1H0,I5,21HMODULE SETS ARE READ-)
  115 FORMAT(/,1H ,*----*)
      IF(KPRNT(6)-1) 770,769,770
  769 CALL SLIST
  770 RETURN
      END
```

```
      SUBROUTINE DISKIO(IPNT,MM )
C
C     THIS SUBROUTINE SIMULATES DISKIO ON360/30                    214
C     IF IPNT=1.  READS FROM MODULE SETS TABLE
C     IF IPN=2. WRITES ONTO MODULE SETS TABLE
C
C     ***** COMMON DECLARATIONS - JUNE 22,1971  ******
C
      COMMON NS(12),SI(-,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSR,NCOUNT
C
      MQ=NPOINT(MM,1)
      ML=NPOINT(MM,2)
      IF(IPNT-1)2,1,2
    1 DO 11 I=1,ML
   11 EN(I)=EEN(MQ+I)
      GO TO 7
    2 IF(IPNT-2)4,3,4
    3 DO 33 I=1,ML
   33 EEN(MQ+I)=EN(I)
      GO TO 7
    4 WRITE(6,100)IPNT,MM
  100 FORMAT(1H0,22H DISKIO ERROR-IPNT,MM=,I5,1X,I5)
    7 RETURN
      END
```

```
      SUBROUTINE SLIST
C        THIS SUBROUTINE PRINTS THE SN LIST AT THE BEGINNING OF THE
C     PROGRAM IF KPRINT(6) IS GREATER THAN ZERO
C        THIS SUBROUTINE IS SPECIAL FOR EVERY SIMULATION
C
C
      WRITE(6,888)
  888 FORMAT(1H1)
      WRITE(6,989)
  989 FORMAT(20X,*----------------------------------------*)
      WRITE(6,999)
  999 FORMAT(35X,*STREAM LIST - NAPHTHA SIMULATION*)
      WRITE(6,989)
      WRITE(6,998)
  998 FORMAT (/,28X,*1.  STREAM NUMBER*,/,28X,*2.  STREAM TYPE*,/,28X,*3
     1.  TOTAL FLOW*,/,28X,*4.  TEMPERATURE DEGREES C*,/,28X,*5.  PRESSU
     2RE, ATM.*)
      WRITE(6,993)
  993 FORMAT(/,30X,*COMPONENT FLOWS, KGM.-MOLES/HR.*)
      WRITE(6,994)
  994 FORMAT(/,28X,*GASEOUS COMPONENTS*,/)
      WRITE(6,992)
  992 FORMAT(/,28X,*6.  WATER*,/,28X,*7.  CARBON DIOXIDE*,/,28X,*8.  CA
     1RBON MONOXIDE*,/,28X,*9.  METHANE*,/,28X,*10. HYDROGEN*,/,28X,*11.
     2NITROGEN*,/,28X,*12. OXYGEN*,/,28X,*13. NAPHTHA*)
      WRITE(6,995)
  995 FORMAT(/,28X,*LIQUID COMPONENTS*,/)
      WRITE(6,991)
  991 FORMAT(/,28X,*14. WATER*,/,28X,*15. FUEL (KGM./HR.)*,/,28X,*16. NA
     1PHTHA*)
      WRITE(6,989)
      WRITE(6,888)
      RETURN
      END
```

```
      OVERLAY(GEMCS,2,1)
      PROGRAM TWO
C
C     GEMCS/6400  -  MAIN GEMCS PROGRAM FOR USE IN THE COMBINED GEMCS -
C                    SYMBOL  EXECUTIVE
C                                                  -  SEPTEMBER 1976
C
      COMMON NS(121),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSP,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(16)
      COMMON/PPDATA/ NTPC(17),LOC(17),PROP(17,16,4)
      COMMON/D/KOUNT,JSWMX
      REWIND 20
      BUFFER IN (20,0) (LLST(1),NCOUNT)
      IF(UNIT(20).GE.0.0) STOP *TAPE20 ERROR*
      REWIND 21
      BUFFER IN (21,0) (NTPC(1),PROP(17,16,4))
      IF(UNIT(21).GE.0.0) STOP *TAPE21 ERROR*
C
C       ***NOTE*** ACHANGE IN THE SN SIZE REQUIRES A CHANGE IN III TO
C      MATCH THIS CHANGED SIZE
C       *** NOTE*** THE SIZE OF THE SN TABLE MUST BE THE NUMBER OF
C      STREAMS WHICH HAVE A 1,2,OR3 STREAM TYPE LABEL PLUS THE MAXIMUM
C      NUMBER OF OUTPUT STREAMS FROM AN EQUIPMENT MODULE WHICH HAVE A
C      7 LABEL
      ISP=0
       IOE=1
C
C     CALCULATING EQUIPMENT IN CALCULATION ORDER LIST
C
      NC=1
      IK=1
      KTEST =0
C
C     PRINT SN TABLE
C
  707 IF(LOOP.EQ.999) KTEST=1
      IF (KPRNT(8)) 710,710,709
  709 WRITE(6,112) LOOP
C
  112 FORMAT(1H0,25HSN TABLE ON ENTERING LOOP,I5)
      DO 111 IKE=1,III
      IF(SN(IKE,1).LE.0.) GO TO 111
      WRITE(6,103) (SN(IKE,J),J=1,JJ)
      WRITE (6,115)
  111 CONTINUE
C
C     READ EN VECTOR
C
  710 NE=LLST(NC)
      MM=NE
      CALL DISKIO(1,MM)
      NN=EN(1)+.001
      NIN=EN(6)+.001
      NOUT=EN(11)+.001
      NTYPE = ABS(EN(2)) + .001
      KSW=
      IF (EN(2).LE.0.) KSW=1
      IF (KPRNT(1)) 40,40,711
  711 WRITE(6,106) NE
  106 FORMAT(1H0,19HMODULE SET FOR UNIT,I5)
      WRITE(6,103)(EN(I),I=1,NN)
C
C     FINDING INPUT STREAMS
C
   40 DO 41 I=1,NIN
      S=EN(I+6)
      CALL STREAM(-S)
      IF(IS-III) 43,43,142
  142 CONTINUE
      SI(I,1)=S
C       IF NO STREAM AVAILABLESET SI(1,-) =0.
```

```
      DO 45 J=2,JJ
   45 SI(I,J)=0.
      GO TO 41
   47 DO 46 J=1,JJ
   46 SI(I,J)=SN(IS,J)
      IF(INT(S+.001)) 41,41,142
  142 CONTINUE
      M=S+.001
      IF(NS(M)-6)41,141,141
  141 CONTINUE
      SN(IS,1)=0.
   41 CONTINUE
      IF (KPRNT(4)) 714,714,713
  107 FORMAT(1H0,24HINPUT STREAMS FOR MODULE,IF)
  713 WRITE(6,107) NE
      DO 108 IKE=1,NIN
      WRITE(6,105)(SI(IKE,J),J=1,JJ)
  108 WRITE(6,115)
  714 CONTINUE
C
C     FINDING OUTPUT STREAMS
C
      DO 46 I=1,NOUT
      S=EN(I+11)
      CALL STREAM(-S)
      IF(III-IS) 46,146,146
  146 CONTINUE
      DO 49 J=2,JJ
   49 SO(I,J)=SN(IS,J)
   46 SO(I,1)=S
C
C
C     CALLING MODULES
C
      CALL MODULE(NTYPE)
C
C
   50 CONTINUE
C
C     STORING OUTPUT STREAMS AND PRINTING
C
      IF(NOUT)68,68,168
  168 CONTINUE
      DO 60 I=1,NOUT
      S=EN(I+11)
      M=S+.001
  164 IF(NS(M).NE.3) GO TO 64
      IF(ISP.EQ.2) GO TO 64
  165 WRITE(6,120)M
  120 FORMAT(1H0,26H OUTPUT FOR PRODUCT STREAM,I5)
      WRITE(6,105) (SO(I,J),J=1,JJ)
   64 IF (NS(M)-6) 66,60,66
   66 CALL STREAM(-S)
      IF(IS-III)62,62,171
C       STORES IN NEXT LOCATION AVAILABLE IN SN
  171 CALL STREAM(0.)
      IF (IS-III) 62,62,163
C      IF NO SPACE AVAILABLE IN SN WRITE (6,102)
  163 WRITE(6,102)
      GO TO 60
C       IF NS(M) =3,SO'S STORED IN SN TABLE TEMPORARILY FOR IMMEDIATE
C       USE IN NEXT CALCULATION
C       IF NS(M) =1,2,OR 3 THEN STORED PERMANENTLY
   62 DO 63 J=1,JJ
   63 SN(IS,J)=SO(I,J)
   60 CONTINUE
C     STORE INPUT STREAMS IF DESIRED
   68 CONTINUE
      IF(NIN.LE.0) GO TO 6999
      DO 6000 I=1,NIN
      S=EN(I+6)
      M=S+0.001
      IF(NS(M).GE.6) GO TO 699
      CALL STREAM(-S)
      IF(IS.LE.III) GO TO 6299
      CALL STREAM (0.)
      IF(IS.LE.III) GO TO 6299
```

```
      WRITE(6,102)
      GO TO 685
 6200 DO 6300 J=1,JJ
 6300 SN(IS,J)=SI(I,J)
  600 CONTINUE
 6899 CONTINUE
      IF(ISP)98,98,715
  715 WRITE(6,109)NE
  109 FORMAT(1HO,25HOUTPUT STREAMS FOR MODULE,I5)
      DO 116 IKE=1,NOUT
      WRITE(6,105)(SO(IKE,J),J=1,JJ)
  116 WRITE (6,115)
   98 IF(LOOP-999)99,97,97
97      IF (KSW.NE.0) GO TO 99
      PRINT 114,NE
  114 FORMAT(1HO,31HFINAL OUTPUT STREAMS FOR MODULE,I5)
      IF (NOUT) 99,99,799
799     CONTINUE
      DO 116 IKE=1,NOUT
      WRITE(6,105)(SO(IKE,J),J=1,JJ)
  116 WRITE (6,115)
   99 IF(NOUT.LT.1) GO TO 867
      DO 866 K1=1,NOUT
  866 SO(K1,2)=0.
  867 NC=NC+1
      IF(NC-NCALC) 707,707,708
C
  708 CONTINUE
      IF (LOOP-999) 71,70,70
   71 CONTINUE
      NC=0
      NIT=NIT+1
      IF(NIT-NLP) 99,99,749
   70 IF(KTEST) 727,727,728
  727 NC=0
      DO 729 IG=1,5
  729 KPRNT(IG)=0
      KTEST=1
      GO TO 99
  728 WRITE(6,104)
  749 CONTINUE
      REWIND 20
      BUFFER OUT (20,0) (LLST(1),NCOUNT)
      CALL UNIT(20)
  100 FORMAT(215)
  102 FORMAT(1H ,11HERROR IN SN)
  103 FORMAT(1H ,5F15.5)
  104 FORMAT(1HO,16HEND OF EXECUTION)
  105 FORMAT(1H ,5F15.5)
55678 FORMAT(* ------* FF.2)
  115 FORMAT(/,1H ,*-----*)
      END
```

Notes on the Use of Subroutine CVGP1 and CØNTL2 (or CØNTL3)

CVGP1 is a convergence promoter based on a mathematical analysis developed by Dr. Crowe (professor of Chemical Engineering at McMaster University) and his students (C4) (C5). CØNTL2 (CØNTL3 in the combined executive)[1] is a convergence tester which is required by GEMCS (and the combined executive) to test key streams for convergence criterion for any plant simulation which contains one to a maximum of eight streams (usually recycle or cut streams).

In the calculation order list, CVGP1 must be inserted just before CØNTL2 (or CØNTL3). CØNTL2 (or CØNTL3) should be the last unit in the calculation order list for that part of the plant which involves recycle stream(s). Table A.2 and Table A.3 summarize the equipment vector for CVGP1 and CØNTL2 (or CØNTL3); as usual, EN(1 to 15) is the same as any other model where EN(6)+EN(11) represents the total number of streams to be tested (maximum of 4 each for a total of 8 possible streams).

---

1. The only difference between CØNTL2 (used in the self standing GEMCS executive) and CØNTL3 (used in the combined GEMCS-SYMBØL executive) is that CØNTL3 also redefines NIT (the number of iterations used in GEMCS before transferring to SYMBØL) whenever LØØP is redefined (in CØNTL2 and CØNTL3).

Table A.2 Equipment list (EN vector) for CVGP1

EN vector No.	Data Inserted	Remarks
3	24	length of the EN vector
16	0	identification numbers for cut streams. Number of coefficients used[1].
17	3	Number of iterations that must be carried out before a promotion step is allowed. (must be $\geq$ 3).
18	3	Number of iterations that must be carried out since last promotion before another promotion is allowed ($\geq$ 3).
19	0.01	Tolerance on percentage change in the predicted solution from iteration to iteration[2].
20	0.0	Relaxation factor TR for direct substitution.
21	1.0	Relaxation factor on promotion step (TC). $0.7 \leq TC \leq 1.0$
22	(100*JSWMX+JOPT)	JSWMX is the number of eigencoefficients to be used[5].
23	1	Counter KØUNT which is incremented on each loop and indicates the number of loops since the beginning or since the last promotion step[6].

Table A.2 (cont'd)

1 - If it is desired to have the subroutine CVGP1 determine the largest number of eigenvalues it can use, then JOPT in EN(22) is set equal to 2 and EN(16) may be left blank. If JOPT=1, then the number of eigenvalues which will be used will be given by JSWMX in EN(22) and again EN(16) may be left blank. In both cases EN(16) is set by the subroutine and then used on subsequent iterations.

2 - On each iteration (after enough information has been accumulated), CVGP1 predicts what the values in the stream information vectors of all cut streams should be. It provides an analysis of how much different these predicted values are from the last prediction as a percentage. If this value is less than the value in EN(19), promotion will be allowed. This procedure ensures that the direct substitution behaviour of the simulation has become regular and any major perturbations have been dampened out. Also included is a test to ensure that the change that would ensure if promotion is provided is significantly larger than the change brought about by direct iteration.

3 - When direct successive substitution is used to obtain a solution of a recycle problem, a calculated value is computed using an assumed value; this calculated value then becomes the assumed value, etc. Mathematically this is expressed as:

$$\underline{y} = f(\underline{x}_o)$$

and

$$\underline{x}_N = \underline{y} \text{ for direct substitution}$$

There may be instances where one can take a larger step than $\underline{x}_N - \underline{x}_O$ or others where it is necessary to take a smaller step than this difference (solution diverging rather than converging). This is allowed through the use of TR=EN(20). In the program,

$$\frac{x_n - y}{x_o - y} = TR$$

so that for TR=0, $x_N = y$ (direct substitution)

for TR=negative value, then $\underline{x}_N$ larger than $\underline{y}$

for TR=positive value, then $\underline{x}_N$ smaller than $\underline{y}$

and for TR=1 $\underline{x}_N = \underline{x}_O$ (same solution as just generated).

4 - It may be found advisable to not take the full promotion step since it is found by experience that CVGP1 overestimates the correction required or if such a step leads to severe oscillations in the solution. This factor reduces the actual step change in cut stream values as calculated by CVGP1.

5 - JSWMX = maximum number of eigenvalues allowed to be used

JØPT = 1 - fixed number of eigenvalues (=JSWMX) will be used

JØPT = 2 CVGP1 searches for the maximum number it can use, this value must be less than JSWMX.

6 - KØUNT is a counter which ensures that a false convergence will not be suggested by CØNTL2 (or CØNTL3) during the few iterations just following a promotion step. Experience has demonstrated that this can occur. Note that KØUNT gives a direct comparison with EN(18).

Table A.3 Equipment list for CØNTL2 (or CØNTL3)

EN vector No.	Data inserted	Remarks
3	21+(NØCØMP*3) + no. of cut streams to be tested	
5	5	This is a print control which must be some number other than zero. In this case, the information will be printed every 5 iterations.
6 to 15	-	as per CVGP1
16	Number of modules in the calculation order list which are controlled by CØNTL2 (or CØNTL3)	The number of modules controlled by CØNTL2 (or CØNTL3) are those which are in the recycle part of the calculation order list.
17	Maximum number of iterations or loops allowed before the calculation is stopped	
18	0	not used
19	0.001	fractional tolerance on each element $(X_{n+1}-X_n)/x_{n+1-}<0.001$
20 and 21	1	loop number (not needed at beginning of simulation.
22 to{21+(NØCØMP*3) + no. of cut streams to be tested}	0	These spaces are required to store the cut stream information.

```
      SUBROUTINE CONTL1
C
C     MULTIPLE STREAM CONVERGENCE TESTER
C     MAXIMUM NUMBER OF STREAMS THAT CAN BE TESTED IS EIGHT (8)
C                                4 INPUT AND 4 OUTPUT STREAMS
C     WRITTEN BY T.W. HOFFMAN -- REVISION OF CONTL1 - JULY 1972
C     UPDATED VERSION FOR USE IN THE COMBINED GEMCS-SYMBOL EXECUTIVE
C
      COMMON NS(12),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(12,16)
      COMMON/GEMCS1/LLST(120),EEN(150),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSF,NCOUNT
      COMMON/D/KOUNT,JSW4X
C
C        WRITTEN BY T.W.HOFFMAN--REVISION OF CONTL1-JULY 1972
C
C     EN LIST
C     1. EQUIPMENT NUMBER
C     2. EQUIPMENT TYPE -2
C     3. THE LENGTH OF THE EN VECTOR IS   21+(NUMBER OF STREAMS TESTED)*(
C     (NOCOMP+3)
C     4.
C     5. PRINTING CONTROL-NUMBER OF LOOPS BETWEEN PRINTING OF THE STREAM
C        UNDER TEST ONLY NON ZERO COMPONENTS+ AND THEIR FRACTIONAL
C        CHANGES
C     6.  NIN - NO. OF STREAM TO BE TESTED
C     16. NO. OF MODULES IN CALC ORDER LIST CONTROLLED BY CONTL1
C     17. MAXIMUM NO. OF LOOPS
C     18.  ZERO
C     19. FRACTIONAL TOLERANCE
C     20. LOOP IDENTIFICATION NO. -STARTING LOOP NUMBER TO BE SPECIFIED
C     21. LOOP IDENTIFICATION NO. -RUNNING COUNT,SET=EN(20) AT START
C
C
      IF (ISP) 13101,13101,13102
13102 WRITE(6,13000) NE
13000 FORMAT(1H ,6HCONTL1,IF)
13101 CONTINUE
C
C        TEST IF CONVERGED ALREADY OR THIS IS THE LAST LOOP TO BE RUN
      IF(999-LOOP)13030,13030,13300
C
C        SET CONSTANTS IN PROGRAM
13301 JSW=0
      NK=EN(6)+EN(11)+0.001
C     NK=NUMBER OF STREAMS BEING TESTED =NUMBER OF INPUT STREAMS
      ML=EN(20)+.001
      LL=LOOP-ML+1
      WRITE(6,13001) LOOP
13001 FORMAT(1H0,11HEND OF LOOP,I5)
C           PRINT CONTROL CALCULATIONS
      D=FLOAT(LL)/EN(5)
      C=FLOAT(INT(D))-D
C     TEST EACH STREAM IN TURN-ALL MUST MEET CONVERGENCE TEST--
C     NOTE THAT EACH MUST PASS THE SAME TOLERANCE TEST
C
      DO 130 IJK=1,NK
      IF(IJK.GT.4) E=EN(7+IJK)
      IF(IJK.GT.4) GO TO 13501
      E=EN(6+IJK)
C        CONSTANTS FOR THIS STREAM
13501 L=2
      NNS=E+.001
      KJT=IJK-1
      SF1=0.
      M=KJT*(NOCOMP+3)+10
      IF(INT(E+.0001))13011,13023,13011
13011 CALL STREAM (E)
C     IF FIRST LOOP BYPASS TEST
      IF(LL-2)13023,13150,13150
13150 CONTINUE
      DO 13013 J=3,JJ
      S=SN(IS,J)
      DS=EN(J+M)
C     JULY TEST THOSE STREAMS THAT HAVE FLOWRATES GREATER THAN 1 UNIT/TIM
      IF(INT(S+.001)) 13016,13013,13016
```

```
13016    TEST=(S-OS)/S
         TS=ABS(TEST)
C          FIND COMPONENT WITH THE LARGEST CHANGE ON THIS ITERATION
         IF(TS.LT.SF1) GO TO 50
         SF1=TS
         KK=J
   50    IF(TS.LE.EN(19)) GO TO 13017
C          L GREATER THAN ZERO IF NOT CONVERGED
13016    L=L+1
13017    CONTINUE
C          PRINTING ON PRINT CONTROL--C
         IF(C)13013,13130,13135
13130    CONTINUE
13024    IF(ABS(C)-.001)13135,13135,13013
13135    CONTINUE
         IF( J .NE.3) GO TO 13136
         WRITE(6,3005)
 3005    FORMAT(1H ,2X,*J*,10X,*VALUE*,7X,*FRAC*)
13136    WRITE(6,13004) J,S,TEST
13004    FORMAT(1H ,36HFRACTIONAL CHANGE IN TOTAL OF STREAM,F5.0,2HIS,F10.8
        1)
13013    CONTINUE
13005    FORMAT(1H ,I3,E15.5,F11.8)
         IF(L) 13022,13020,13022
13020    WRITE(6,999) NNS,LOOP
         GO TO 60
13022    WRITE(6,998) NNS,KK,SF1,LOOP
C          JSW IS SWITCH WHICH INDICATES WHETHER A PREVIOUS TEST HAS FAILED
C        NOTE IT REMAINS ZERO IF ALL STREAMS PASS
   60    CONTINUE
C
13023    DO13021 J=3,JJ
13021    EN(J+M)=SN(IS,J)
C
         IF(JSW.GT.0) GO TO 100
C          PROTECT AGAINST CONVERGED SOLUTION AT OUTSET,OTHERWISE VARIABLES
C        ARE UNDEFINED IN CVGP1
         IF(KOUNT.LE.(JSW*X+2)) L=L+1
C----KOUNT IS IN OD2,THE CONVERGENCE PROMOTER.THIS ENSURES THAT FALSE
C        CONVERGENCE WILL NOT OCCUR
         JSW=L
  100    CONTINUE
C
         IF(LL-2) 13015,150,150
  150    CONTINUE
C          TEST IF ALL COMPONENTS ARE CONVERGED
         IF(JSW)  250,200,250
C
  200    EN(21)=EN(20)
         LOOP=999
         NC=0
         GO TO 13030
C
  250    IF(LL-INT(EN(17)*.001)) 13015,13025,13015
13025    WRITE(6,997) LOOP
         GO TO 200
13015    EN(21)=LOOP+1
         LOOP=LOOP+1
         NIT=NIT+1
         NC=NC-INT(EN(16))-1
C
  997    FORMAT(1H ,*CONVERGENCE NOT ACHIEVED WITHIN THE SPECIFIED MAXIMUM
        1NUMBER OF LOOPS,EQUAL TO,*,I5)
  998    FORMAT(1H ,   10X,*  TEST ON STREAM,*,I5,*   NOT CONVERGED,COMPONE
        1NT ,*,I5,* HAS THE GREATEST  FRACTIONAL CHANGE OF*,F15.8,* ON LOOP,
        2* ,I5)
  999    FORMAT(1H ,*TEST ON STREAM,*,I5,* CONVERGED FOR ALL COMPONENTS ON
        1LOOP,*,I5)
C
13030    MM=NE
         CALL DISKIO(2,MM)
         RETURN
         END
```

```
      SUBROUTINE CVGP1
C     REVISION OF JANUARY 23,1976.
C     ALLOWS EITHER A FIXED NO. OF COEFFICIENTS OR A STEADILY INCREASING
C     NO. TO BE USED. JOPT=1 OR 2, RESPECTIVELY.
C     WRITTEN BY C.M.CROWE, AUG.1972
C                     DEPT. OF CHEMICAL ENGINEERING,
C                     MCMASTER UNIVERSITY, HAMILTON,ONTARIO.
C        REFERENCES- CROWE + NISHIO,AICHE J 21,528(1975)
C                    ORBACH + CROWE,CAN.J.CH.E. 49,509(1971)
C     MODIFIED FROM PROGRAMS WRITTEN BY O. ORBACH AND BY T. HOFFMAN.
C     CONVERGENCE PROMOTION CAN BE  APPLIED TO (NIN+NOUT) CUT STREAMS.
C       LP ITERATIONS ARE REQUIRED BEFORE PROMOTION CAN BE APPLIED FOR
C     THE FIRST TIME, AND LK ITERATIONS ARE REQUIRED SUBSEQUENTLY.
C     X1(-)  CONTAINS THE LATEST CALCULATED VALUES OF THE CUT VARIABLES.
C     X2(-)  CONTAINS THE NEXT TO LATEST CALCULATED VALUES OF THE CUT
C     VARIABLES.
C     XA(-) CONTAINS THE PREVIOUS VALUES OF THE ASSUMED CUT VARIABLES.
C     XX(-) CONTAINS THE CURRENT PREDICTED SOLUTION.
C     XO(-) CONTAINS THE PREVIOUS PREDICTED SOLUTION.
C     DIMENSIONS OF VARIABLES FOR *JSW* EIGENVALUES SHOULD BE -
C     DX(JSW,-),XMU(JSW),XMUO(JSW),XSP(JSW),SCPD(JSW*(JSW+1)/2),YSP(AS S
C     ***** COMMON DECLARATIONS - JUNE 22,1971  ******
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSF,NCOUNT
      COMMON/D/KOUNT,JSWMX
      COMMON/Z/XA(192),XO(192),DX(12,192),X2(192),SCPD(78),YSP(78)
      DIMENSION X1(192),XMU(12),XSP(12),XX(192),XSUM(12),NWRK(12)
      DOUBLE YSP,SCPD
      DOUBLE XMU,XSP,XSUM,YSUM


C     EN VECTOR
C     2.    EQUIPMENT TYPE 32
C     3.    LENGTH OF EN LIST = 24
C     6.    NO. OF INPUT STREAMS, NIN
C     7.-10.     INPUT STREAM NUMBERS
C     11.   NO. OF OUTPUT STREAMS, NOUT
C     12.-15.    OUTPUT STREAM NUMBERS
C     16.   JSW = NO. OF EIGENCOEFFICIENTS CURRENTLY USED.
C     17. LP = NO. OF ITERATIONS BEFORE ACCELERATION IS APPLIED,GE.3
C     18.   LK = NO. OF ITERATIONS BEFORE ACCELERATION IS REAPPLIED,GE.3
C     19. EGT = TOLERANCE OF THE FRACTIONAL CHANGE OF THE PREDICTED
C     SOLUTION,IN PERCENT, SAY 0.01
C     20. TP = RELAXATION FACTOR FOR THE ITERATION (MAY BE GE.0 FOR POSI
C     DOMINANT EIGENVALUE,LE.0 FOR NEGATIVE, AND EQ.0 FOR SUCCESSIVE
C     SUBSTITUTION.
C     21. TC = RELAXATION FACTOR FOR THE ACCELERATION( LE.1,SAY 0.7 TO 1
C     22.   ENTRY = (100*JSWMX+JOPT)
C           JSWMX = MAXIMUM NO. OF EIGENCOEFFS. USED.
C           JOPT = 1   -  FIXED NO. OF EIGENCOEFFS.
C                = 2   -  VARIABLE NO. OF EIGENCOEFFS., LE.JSWMX.
C     23. KOUNT = NO. OF ITERATIONS SINCE LAST CONVERGENCE PROMOTION,
C     USED IN CONTL1 TO ENSURE THAT CONVERGENCE IS NOT ALLOWED IMMEDIATE
C     AFTER A STEP CHANGE IS MADE.

      IF(ISP) 2116,2115,2117
 2117 WRITE(6,2118) NE
 2118 FORMAT(1H ,*CVGP1*,I7)
 2116 CONTINUE
      IF (LOOP.EQ.999) RETURN
C       START PROGRAM

      NIN=EN(6)+.001
      NOUT=EN(11)+.001
      JSW=EN(16)+.001
      LP=EN(17)+.001
      LK=EN(18)+.001
      EGT=EN(19)
      TP=EN(20)
      TC=EN(21)
      JSWMX=(EN(22)+.001)/100
```

```
      JOPT=EN(22)+.01-JSWMX*100
      KOUNT = EN(23)+.01
      NK=NCOMP+2
      NKT=(NIN+NOUT)*NK
C
C     RECORD NEW VALUES OF OUT VARIABLES.
      IF (NIN.LE.0)  GO TO 51
      DO 50 K1=1,NIN
      K2=(K1-1)*NK
      DO 50 K=1,NK
      IF(SI(K1,K+3).LT.0.1E-10) SI(K1,K+3)=0.1E-10
50    X1(K2+K)=SI(K1,K+3)
51    IF (NOUT.LE.0)  GO TO 53
      DO 52 K1=1,NOUT
      IF(SO(K1,K+3).LT.0.1E-10) SO(K1,K+3)=0.1E-10
      K2=(K1+NIN-1)*NK
      DO 52 K=1,NK
52    X1(K2+K)=SO(K1,K+3)
53    IF (NIN.LE.0.A.NOUT.LE.0)  GO TO 76
      IF(LOOP.GT.1) GO TO 200
C
C     INITIALIZE VARIABLES
      IF (LP.LT.3)  LP=3
      EN(17)=LP
      IF (LK.LT.3)  LK=3
      EN(18)=LK
      IF (JSWMX.LT.1)  JSWMX=1
      IF ((JOPT+1)/2.NE.1)  JOPT=1
      EN(22)=100*JSWMX+JOPT
      IF (JOPT.EQ.1)  JSW=JSWMX
      IF (JOPT.EQ.2)  JSW=1
      EN(16)=JSW
      DO 100 K1=1,NKT
100   XA(K1)=X1(K1)
      KOUNT=1
      GO TO 76
C
200   KOUNT=KOUNT+1
      M1=MIND(LOOP-LP,KOUNT-LK)
      IF (M1+1)  86,210,220
210   DO 215 K1=1,NKT
215   X2(K1)=X1(K1)
      GO TO 86
220   NSP=JSW*(JSW+1)/2
      IF (M1-JSW)  225,240,252
225   K2=JSW-M1
C     RECORD INITIAL SET DX=S.
      DO 230 K1=1,NKT
      DX(K2,K1)=X1(K1)-X2(K1)
230   X2(K1)=X1(K1)
      GO TO 86
C
C     CALCULATE INITIAL SET OF SCALAR PRODUCTS.
C     MATRIX OF SCALAR PRODUCTS IS STORED AS VECTOR SCPD(I) IN LOWER
C     TRIANGULAR FORM, MOST RECENT RESULTS IN LAST ROW
240   IF (JOPT.EQ.2.A.JSW.GT.1)  GO TO 252
      K4=NSP+1
      DO 255 K1=1,JSW
      DO 255 K2=K1,JSW
      K4=K4-1
      SCPD(K4)=0.
      DO 255 K3=1,NKT
255   SCPD(K4)=DX(K1,K3)*DX(K2,K3) +SCPD(K4)
C
C        START THE PROMOTION CALCULATIONS
C     COMPUTE NEW VALUES OF SCALAR PRODUCTS)
252   DO 11 K=1,JSW
11    XSP(K)=0.
      SUMDEL=0.
      SUMSQ=0.
      JSW1=JSW
      DO 12 K1=1,NKT
      SUMSQ=SUMSQ+X1(K1)**2
      DEL=X1(K1)-X2(K1)
      SUMDEL=SUMDEL+DEL*DEL
      K3=JSW+1
```

```
      DO 111 K=1,JSW
      K2=K3-K
111   XSP(K)=XSP(K)+DEL*DX(K2,K1)
      X2(K1)=X1(K1)
      IF (JSW.LT.JSWMX)  GO TO 1125
      IF (JSW.EQ.1)  GO TO 12
      K3=JSW+2
      DO 112 K=2,JSW
      K2=K3-K
112   DX(K2,K1)=DX(K2-1,K1)
      GO TO 12
1125  K3=JSW+1
      DO 113 K=1,JSW
      K2=K3-K
113   DX(K2+1,K1)=DX(K2,K1)
12    DX(1,K1)=DEL
C
C     COMPUTE RELATIVE CHANGE IN CALCULATED VARIABLES.
      RMSERR=SQRT(SUMDEL/SUMSQ)*100.
      WRITE(6,997)  LOOP,KOUNT,RMSERR
997   FORMAT(/* ON LOOP *I4,*, KOUNT = *I4* AND RMS. ERROR = *1PE12.5* P
     1ERCENT.*)
C
C     COMPUTE NEW VALUES OF EIGENCOEFFICIENTS.
      IF (JSW.GT.1)  GO TO 120
      XMU(1)=-XSP(1)/SCPD(1)
      IF (JOPT.EQ.2.A.M1.LT.JSWMX)  GO TO 150
      K5=1
      GO TO 145
120   K2=JSW-JSW1
      DO 121 K1=1,JSW1
121   XMU(K1)=-XSP(K2+K1)
      K=K2+1
      K5=0
      DO 122  K1=K,JSW
      K4=K1*(K1+1)/2
      K3=K4-K1+K
      DO 122  K2=K3,K4
      K5=K5+1
122   YSP(K5)=SCPD(K2)
      CALL DGELS(XMU,YSP,JSW1,1,1.E-14,IERR,XSUM)
      IF (IERR.GT.0)  WRITE(6,998)  IERR
998   FORMAT(1H ,*IERR = *,I2,* SOLUTION FOR XMU(-) ILL-BEHAVED.*)
      IF (IERR.LE.0.O.IERR.GE.JSW)  GO TO 123
C
C     REDUCE NUMBER OF EIGENCOEFFS. TO IERR DUE TO ILL-BEHAVIOUR OF SOLN
      JSW1=IERR
      JSWMX=JSW1
      EN(16)=JSW1
      EN(22)=100*JSW1+JOPT
      IF (JSW1.GT.1)  GO TO 120
      XMU(1)=-XSP(JSW)/SCPD(NSP)
      K5=1
      GO TO 145
C
C     MOVE SCPD(-) K COLUMNS TO LEFT, AND K ROWS UP.  K=JSW-JSW1+1
123   IF (JOPT.EQ.2.A.M1.LT.JSWMX)  GO TO 150
      K=JSW-JSW1+1
      K5=0
      DO 13 K1=2,JSW1
      K3=K+K1
      K4=(K3-1)*K3/2-K1
      DO 13 K2=2,K1
      K5=K5+1
13    SCPD(K5)= SCPD(K4+K2)
      K3=K+1
      DO 14 K1=K3,JSW
      K5=K5+1
14    SCPD(K5)= YSP(K1)
      K5=K5+1
145   SCPD(K5)= SUMDEL
      JSW2=JSW
      JSW=JSW1
      GO TO 160
C
```

```
C
C     FILL IN NEW ROW OF (JSW+1) ELEMENTS IN SCPD.
150   KE=NSE
      DO 155 K1=1,JSW
      KE=KE+1
155   SCPD(KE)=XSP(K1)
      SCPD(KE+1)=SUMDEL
160   WRITE(6,996) (XMU(K),K=1,JSW)
996   FORMAT(/* EIGENCOEFFICIENTS = *(8(F10.5,2X)))
C
C     FIND PREDICTED SOLUTION.
      SUM=0.
      DO 841 K1=1,JSW
      SUM=SUM+XMU(K1)
841   XSUM(K1)=SUM
      SUM=SUM+1.
      K4=JSW+1
      DO 843 K3=1,NKT
      ZSUM=0.
      DO 842 K1=1,JSW
      K2=K4-K1
842   ZSUM=ZSUM+XSUM(K1)*DX(K2,K3)
843   XX(K3)=X1(K3)-TC*ZSUM/SUM
C
C     FIND CHANGE IN PREDICTED SOLUTION.
      IF (JOPT.EQ.1.A.M1.EQ.JSW2)  GO TO 71
      IF (JOPT.EQ.2.A.M1.EQ.1)  GO TO 71
      IF (IERR.GT.0)  GO TO 71
      WSUM=0.
      ZSUM=0.
      SUM=0.
      DO 70 K1=1,NKT
      SUM=SUM+XX(K1)**2
      ZSUM=ZSUM+(XX(K1)-X1(K1))**2
70    WSUM=WSUM+(XX(K1)-X0(K1))**2
      WSUM=SQRT(WSUM/SUM)*100.
      ZSUM=SQRT(ZSUM/SUM)*100.
      WRITE (6,99999)  WSUM,ZSUM
99999 FORMAT(//* CHANGE IN PREDICTED SOLN. = *1PE12.5/* DISTANCE FROM PR
     1EDICTED SOLN. = *1PE12.5)
      IF (WSUM.GT.AMIN1(RMSERR/5.,EGT))  GO TO 71
      IF (ZSUM.LE.WSUM*10.)  GO TO 71
C
C     TAKE PROMOTION STEP.
      DO 72 K1=1,NKT
      IF(XX(K1).LT.0.1E-10) XX(K1)=0.1E-10
72    XA(K1)=XX(K1)
      KOUNT=0
      WRITE(6,999)  JSW,LOOP
999   FORMAT(/3X,*PROMOTION STEP TAKEN OF ORDER *I3,* AT LOOP NO. *I4)
      IF (JOPT.EQ.2)  EN(16)=1
      GO TO 88
C
C     STORE LATEST PREDICTED SOLUTION.
71    DO 73 K1=1,NKT
73    XO(K1)=XX(K1)
      IF (JSW.LT.JSWMX)  EN(16) = JSW+1
C
C     ASSIGN NEW ASSUMED VARIABLES TO STRMO AND/OR STRMI.
86    DO 87 K1=1,NKT
      XA(K1)=X1(K1)*(1.+TR)-TR*XA(K1)
      IF(XA(K1).LT.0.1E-10) XA(K1)=0.1E-10
87    CONTINUE
      IF (TE.EQ.1.)  GO TO 76
88    IF (NIN.LE.0)  GO TO 90
      DO 89 K1=1,NIN
      K2=(K1-1)*NK
      SI(K1,3)=0.
      DO 89 K=1,NK
      SI(K1,K+3)=XA(K2+K)
      IF (K.LE.2)  GO TO 89
      SI(K1,3)=SI(K1,3)+SI(K1,K+3)
89    CONTINUE
90    IF (NOUT.LE.0)  GO TO 76
      DO 91 K1=1,NOUT
      K2=(K1+NIN-1)*NK
```

```
      SO(K1,7)=0.
      DO 91 K=1,MK
      SO(K1,K+3)=XA(K2+K)
      IF (K.LE.2) GO TO 91
      SO(K1,3)=SO(K1,3)+SO(K1,K+3)
91    CONTINUE
76    EN(23)=KOUNT
      MM=ME
      CALL DISKIO(2,MM)
      RETURN
      END
```

```
      SUBROUTINE DGELS(R,A,M,N,EPS,IER,AUX)

C     ..................................................................
C
C        SUBROUTINE DGELS
C
C        PURPOSE
C           TO SOLVE A SYSTEM OF SIMULTANEOUS LINEAR EQUATIONS WITH
C           SYMMETRIC COEFFICIENT MATRIX UPPER TRIANGULAR PART OF WHICH
C           IS ASSUMED TO BE STORED COLUMNWISE.
C
C        USAGE
C           CALL DGELS(R,A,M,N,EPS,IER,AUX)
C
C        DESCRIPTION OF PARAMETERS
C           R      - DOUBLE PRECISION M BY N RIGHT HAND SIDE MATRIX
C                    (DESTROYED). ON RETURN R CONTAINS THE SOLUTION OF
C                    THE EQUATIONS.
C           A      - UPPER TRIANGULAR PART OF THE SYMMETRIC DOUBLE
C                    PRECISION M BY M COEFFICIENT MATRIX.  (DESTROYED)
C           M      - THE NUMBER OF EQUATIONS IN THE SYSTEM.
C           N      - THE NUMBER OF RIGHT HAND SIDE VECTORS.
C           EPS    - SINGLE PRECISION INPUT CONSTANT WHICH IS USED AS
C                    RELATIVE TOLERANCE FOR TEST ON LOSS OF
C                    SIGNIFICANCE.
C           IER    - RESULTING ERROR PARAMETER CODED AS FOLLOWS
C                    IER=0  - NO ERROR,
C                    IER=-1 - NO RESULT BECAUSE OF M LESS THAN 1 OR
C                             PIVOT ELEMENT AT ANY ELIMINATION STEP
C                             EQUAL TO 0,
C                    IER=K  - WARNING DUE TO POSSIBLE LOSS OF SIGNIFI-
C                             CANCE INDICATED AT ELIMINATION STEP K+1,
C                             WHERE PIVOT ELEMENT WAS LESS THAN OR
C                             EQUAL TO THE INTERNAL TOLERANCE EPS TIMES
C                             ABSOLUTELY GREATEST MAIN DIAGONAL
C                             ELEMENT OF MATRIX A.
C           AUX    - DOUBLE PRECISION AUXILIARY STORAGE ARRAY
C                    WITH DIMENSION M-1.
C
C        REMARKS
C           UPPER TRIANGULAR PART OF MATRIX A IS ASSUMED TO BE STORED
C           COLUMNWISE IN M*(M+1)/2 SUCCESSIVE STORAGE LOCATIONS, RIGHT
C           HAND SIDE MATRIX R COLUMNWISE IN N*M SUCCESSIVE STORAGE
C           LOCATIONS. ON RETURN SOLUTION MATRIX R IS STORED COLUMNWISE
C           TOO.
C           THE PROCEDURE GIVES RESULTS IF THE NUMBER OF EQUATIONS M IS
C           GREATER THAN 0 AND PIVOT ELEMENTS AT ALL ELIMINATION STEPS
C           ARE DIFFERENT FROM 0. HOWEVER WARNING IER=K - IF GIVEN -
C           INDICATES POSSIBLE LOSS OF SIGNIFICANCE. IN CASE OF A WELL
C           SCALED MATRIX A AND APPROPRIATE TOLERANCE EPS, IER=K MAY BE
C           INTERPRETED THAT MATRIX A HAS THE RANK K. NO WARNING IS
C           GIVEN IN CASE M=1.
C           ERROR PARAMETER IER=-1 DOES NOT NECESSARILY MEAN THAT
C           MATRIX A IS SINGULAR, AS ONLY MAIN DIAGONAL ELEMENTS
C           ARE USED AS PIVOT ELEMENTS. POSSIBLY SUBROUTINE DGELG (WHICH
C           WORKS WITH TOTAL PIVOTING) WOULD BE ABLE TO FIND A SOLUTION.
C
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C           NONE
C
C        METHOD
C           SOLUTION IS DONE BY MEANS OF GAUSS-ELIMINATION WITH
C           PIVOTING IN MAIN DIAGONAL, IN ORDER TO PRESERVE
C           SYMMETRY IN REMAINING COEFFICIENT MATRICES.
C
C     ..................................................................
C
      DIMENSION A(1),R(1),AUX(1)
      DOUBLE PRECISION R,A,AUX,PIV,TB,TOL,PIVI
      IF(M)24,24,1
C
C     SEARCH FOR GREATEST MAIN DIAGONAL ELEMENT
```

```
    1 IER=
      PIV=0.D
      L=
      DO 3 K=1,M
      L=L+K
      TB=DABS(A(L))
      IF(TB-PIV)3,3,2
    2 PIV=TB
      I=L
      J=K
    3 CONTINUE
      TOL=EPS*PIV
C     MAIN DIAGONAL ELEMENT A(I)=A(J,J) IS FIRST PIVOT ELEMENT.
C     PIV CONTAINS THE ABSOLUTE VALUE OF A(I).
C
C
C     START ELIMINATION LOOP
      LST=0
      NM=N*M
      LEND=M-1
      DO 18 K=1,M
C
C     TEST ON USEFULNESS OF SYMMETRIC ALGORITHM
      IF(PIV)24,24,4
    4 IF(IER)5,5,7
    5 IF(PIV-TOL)6,6,7
    6 IER=K-1
      LT=J-K
      LST=LST+K
C
C     PIVOT ROW REDUCTION AND ROW INTERCHANGE IN RIGHT HAND SIDE R
      PIVI=1.D0/A(I)
      DO 8 L=K,NM,M
      LL=L+LT
      TB=PIVI*R(LL)
      R(LL)=R(L)
    8 R(L)=TB
C
C     IS ELIMINATION TERMINATED
      IF(K-M)9,19,19
C
C     ROW AND COLUMN INTERCHANGE AND PIVOT ROW REDUCTION IN MATRIX A.
C     ELEMENTS OF PIVOT COLUMN ARE SAVED IN AUXILIARY VECTOR AUX.
    9 LR=LST+(LT*(K+J-1))/2
      LL=LR
      L=LST
      DO 14 II=K,LEND
      L=L+II
      LL=LL+1
      IF(L-LR)12,10,11
   10 A(LL)=A(LST)
      TB=A(L)
      GO TO 13
   11 LL=L+LT
   12 TB=A(LL)
      A(LL)=A(L)
   13 AUX(II)=TB
   14 A(L)=PIVI*TB
C
C     SAVE COLUMN INTERCHANGE INFORMATION
      A(LST)=LT
C
C     ELEMENT REDUCTION AND SEARCH FOR NEXT PIVOT
      PIV=0.D0
      LLST=LST
      LT=0
      DO 18 II=K,LEND
      PIVI=-AUX(II)
      LL=LLST
      LT=LT+1
      DO 15 LLD=II,LEND
      LL=LL+LLD
      L=L+LT
   15 A(L)=A(L)+PIVI*A(LL)
      LLST=LLST+II
```

```
      LP=LLST+LT
      TB=DABS(A(LP))
      IF(TB-PIV)17,17,16
   16 PIV=TB
      I=LP
      II=II+1
   17 DO 18 LP=K,NM,M
      LL=LP+LT
   18 R(LL)=R(LL)+PIVI*R(LP)
      END OF ELIMINATION LOOP
C
C
C     BACK SUBSTITUTION AND BACK INTERCHANGE
   19 IF(LEND)24,23,20
   20 II=M
      DO 22 I=2,M
      LST=LST-II
      II=II-1
      L=A(LST)+.500
      DO 22 J=II,NM,M
      TB=R(J)
      LL=J
      K=LST
      DO 21 LT=II,LEND
      LL=LL+1
      K=K+LT
   21 TB=TB-A(K)*R(LL)
      K=J+L
      R(J)=R(K)
   22 R(K)=TB
   23 RETURN
C
C     ERROR RETURN
   24 IER=-1
      RETURN
      END
```

```
      SUBROUTINE MODULE(NT)
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23
     1),NT
    1 CONTINUE
      CALL JUNC01
      RETURN
    2 CONTINUE
      CALL EXCH01
      RETURN
    3 CONTINUE
      CALL EXCH02
      RETURN
    4 CONTINUE
      CALL SEPA01
      RETURN
    5 CONTINUE
      CALL PEAC01
      RETURN
    6 CONTINUE
      CALL PEAC02
      RETURN
    7 CONTINUE
      CALL PEAC03
      RETURN
    8 CONTINUE
      IISW=1
      CALL PROP1(IISW)
      RETURN
    9 CONTINUE
      CALL CONTL3
      RETURN
   10 CONTINUE
      CALL CONTL1
      RETURN
   11 CONTINUE
      CALL CONT02
      RETURN
   12 CONTINUE
      CALL CONT03
      RETURN
   13 CONTINUE
      CALL CONT04
      RETURN
   14 CONTINUE
      CALL CONT05
      RETURN
   15 CONTINUE
      CALL EXCH05
      RETURN
   16 CONTINUE
      CALL EXCH04
      RETURN
   17 CONTINUE
      CALL CONT06
      RETURN
   18 CONTINUE
      CALL CVGP1
      RETURN
   19 CONTINUE
      CALL EXCH03
      RETURN
   20 CONTINUE
      CALL ABSO1
      RETURN
   21 CONTINUE
      CALL SETST2
      RETURN
   22 CONTINUE
      CALL CONT07
      RETURN
   23 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE STREAM(S)
C
C     FINDS STREAM S IN THE SN TABLE BY LOOKING FOR SN(IS,,1)=S
C
      COMMON NS(12),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(150),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSF,NCOUNT
C
      DO 1 IS=1,III
      KLK=IS
      IF(ABS(SN(IS,1)-ABS(S))-.010) 2,1,1
    1 CONTINUE
      IS=KLK+1
      IF (S) 2,2,3
    3 CONTINUE
      WRITE(6,400)S
  400 FORMAT(1HC,14HERROR...STREAM,F3.0,16HNOT IN SN MATRIX)
    2 IF(KPRNT(5)) 70,70,71
   71 WRITE(6,401)IS
  401 FORMAT(1HC,3HIS= ,I3)
   70 RETURN
      END
```

```
      OVERLAY(SYMBOL,3,0)
      PROGRAM THREE


      MASS BALANCE PROGRAM, MARK 2.
      VERSION OF 1, JANUARY 1975  J.F. SHEWCHUK
      IMPLEMENTS THE SAME SPECIFICATIONS AS EARLIER VERSIONS.


       UPDATED VERSION FOR USE WITH THE COMBINED GEMCS - SYMBOL
      EXECUTIVE                                    - SEPTEMBER 1976

      COMMON NS(12),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,NIT,SN(72,16)
      COMMON/SBL/ KSC(10,5),KM(7,9),KSP(6,8),KSP6(1,1),KC(1,1),KR(12,6),
     1            KE(5,4),KK(15),AX(100),AMW(11),TOLERX
      COMMON/SYM/NEQ,NVAR,NCP,NEM,IROW(2925),JCOL(2925),A(3825),NW(7560)
      COMMON/SPEC/LLL,MMM
      DIMENSION JNO(2), ALPHA(100), B(100), INS(7)
      DIMENSION SUMS(5)
      REWIND 22
      BUFFER IN (22,0) (KSC(1,1),TOLERX)
      IF(UNIT(22).GE.0.0) STOP *APE23 ERROR*
      WRITE(6,*) (AX(I),I=1,LLL)
      CALL SECOND(TSTART)
      LLL=0
      NPROB=KK(1)
      NCASE=KK(2)
      NST=KK(3)
      NCP=KK(4)
      NSCE=KK(5)
      NMIX=KK(6)
      NSEP=KK(7)
      NCSD=KK(8)
      NRTP=KK(9)
      IF(KPRNT(9).LT.1) GO TO 5
      WRITE (6,920) NPROB,NCASE,NST,NCP,NSCE,NMIX,NSEP,NCSD,NRTP
    5 CONTINUE
      XMIN=1.0E-5
      NVAR=NCP+1
      NEM=0
      NEL=NST*NVAR
      NEXTRA=KK(10)
      NEQ=0
      NZZZ=NST*(NCP+1)
      CALL CHKLIM (NCP,1,99,2)
      CALL CHKLIM (NZZZ,1,900,1)
      IF (NSCE) 10,20,70
   10 CALL ERREP (22)
      GO TO 1500
   20 IF (NMIX) 10,30,130
   30 IF (NSEP) 10,40,160
   40 IF (NCSD) 10,50,300
   50 IF (NRTP) 10,60,550
   60 IF (NEXTRA) 10,760,740

C     SOURCES

   70 DO 120 N=1,NSCE
      IF(KPRNT(9).GE.1)       WRITE (6,930) N
      NEQ=NEQ+1
      MODE=KSC(N,1)
      NA=KSC(N,2)
      CALL SCOMP(NA,B)
      IF(KPRNT(9).GE.1) WRITE (6,940) MODE,NA
      IF(KPRNT(9).GE.1) WRITE (6,950) (B(I),I=1,NCP)
      CALL CHKLIM (MODE,1,3,3)
      CALL CHKLIM (NA,1,NST,4)
      CALL SETCOM (NA,3)
      CALL SETTOT (NA)
      GO TO (110,90,100),MODE

C     MODE 2

   90 NB=KSC(N,3)
```

```
      CALL CFLOW(NA,NB,E)
      IF(KPRNT(9).GE.1)  WRITE (6,960) NB,E
      CALL CHKLIM (NB,1,NVAR,5)
      CALL SETVAL (NA,NB,E)
      GO TO 11
C
C     MODE 3
C
  10  NB=KSC(N,3)
      NC=KSC(N,4)
      ND=KSC(N,5)
      CALL RATIO(NA,NB,NC,ND,E)
      IF(KPRNT(9).GE.1)  WRITE (6,970) NB,NC,ND,E
      CALL CHKLIM (NB,1,NVAR,5)
      CALL CHKLIM (NC,1,NST,4)
      CALL CHKLIM (ND,1,NVAR,5)
      CALL SETRAT (NA,NB,NC,ND,E)
  11  IF(KPRNT(9).GE.1)  WRITE (6,980) NEQ1,NEQ
12    CONTINUE
      GO TO 20
C
C     MIXERS
C
13    DO 15 I=1,NMIX
      IF(KPRNT(9).GE.1)  WRITE (6,990) I
      NEQ1=NEQ+1
      MODE=KM(I,1)
      JNO(1)=KM(I,2)
      DO 135 J=1,MODE
      INS(J)=KM(I,J+2)
  135 CONTINUE
      IF(KPRNT(9).GE.1)  WRITE (6,940) MODE,JNO(1),(INS(K),K=1,MODE)
      CALL CHKLIM (MODE,2,7,3)
      CALL CHKLIM (JNO(1),1,NST,4)
      DO 14 J=1,MODE
      CALL CHKLIM (INS(J),1,NST,4)
14    CONTINUE
      CALL SETMIX (INS,MODE,JNO,1)
      CALL SETTOT (JNO(1))
      IF(KPRNT(9).GE.1)  WRITE (6,980) NEQ1,NEQ
15    CONTINUE
      GO TO 30
C
C     SEPARATORS
C     ALL MODES
C
16    DO 290 I=1,NSEP
      IF(KPRNT(9).GE.1)  WRITE (6,1000) I
      NEQ1=NEQ+1
      MODE=KSP(I,1)
      INS(1)=KSP(I,2)
      JNO(1)=KSP(I,3)
      JNO(2)=KSP(I,4)
      IF(KPRNT(9).GE.1)  WRITE (6,940) MODE,INS(1),JNO(1),JNO(2)
      CALL CHKLIM (MODE,1,6,3)
      CALL CHKLIM (INS(1),1,NST,4)
      CALL CHKLIM (JNO(1),1,NST,4)
      CALL CHKLIM (JNO(2),1,NST,4)
      CALL SETMIX (INS,1,JNO,2)
      CALL SETTOT (JNO(1))
      CALL SETTOT (JNO(2))
      GO TO (170,170,170,170,220,240), MODE
C
C     MODES 1 TO 4
C
  17  NA=KSP(I,5)
      IF(MODE.EQ.4) GO TO 171
      CALL SCOMP(NA,B)
      GO TO 172
  171 CALL SFRAC(INS(1),NA,B)
  172 IF(KPRNT(9).GE.1)  WRITE(6,950) (B(J),J=1,NCP)
      IF(KPRNT(9).GE.1)  WRITE (6,960) NA
      CALL CHKCMP (NA,JNO,2,6)
      GO TO (180,180,180,240), MODE
C
C     MODES 1 TO 3
```

```
1[0     CALL SETCOM (NA,2)
        GO TO (2[0,21[,22[),MODE
C
C       MODE 2
C
  2[[ NB=KSP(I,6)
        CALL CFLOW(NA,NB,E)
        IF(KPRNT(9).GE.1) WRITE (6,95[) NB,E
        CALL CHKLIM (NB,1,NVAR,5)
         CALL SETVAL(NA,NB,E)
        GO TO 25[
C
C       MODE 3
C
  21[ NB=KSP(I,6)
        NC=KSP(I,7)
        ND=KSP(I,8)
        CALL RATIO(NA,NB,NC,ND,E)
        IF(KPRNT(9).GE.1) WRITE (6,97[) NB,NC,ND,E
        CALL CHKLIM (NB,1,NVAR,5)
        CALL CHKLIM (NC,1,NST,4)
        CALL CHKLIM (ND,1,NVAR,5)
        CALL SETRAT (NA,NB,NC,ND,E)
        GO TO 29[
C
C       MODE 5
C
  22[ NA=KSP(I,5)
        M=NVAR
        CALL CFRAC(NA,INS(1),M,E)
        IF(KPRNT(9).GE.1)  WRITE (6,96[) NA,E
        CALL CHKCMP (NA,JNO,2,6)
        DO 23[ J=1,NCP
        B(J)=E
23[     CONTINUE
C
C       MODES 4,5 AND 6
C
24[     NC=INS(1)
        IF(MODE.GT.5) M6=M6+1
        K=1
        DO 27[ J=1,NCP
        NB=J
        ND=J
        IF (MODE-6) 26[,25[,25[
C
C       MODE 6
C
  25[ NA=KSP6(M6,K)
        NB=KSP6(M6,K+1)
        NC=KSP6(M6,K+2)
        ND=KSP6(M6,K+3)
        CALL RATIO(NA,NB,NC,ND,B(J))
        IF(KPRNT(9).GE.1)  WRITE (6,1[1[) NA,NB,NC,ND,B(J)
        CALL CHKLIM (NA,1,NST,4)
        CALL CHKLIM (NB,1,NVAR,5)
        CALL CHKLIM (NC,1,NST,4)
        CALL CHKLIM (ND,1,NVAR,5)
        K=K+4
C
C       MODES 4,5 AND 6
C
26[     CALL SETRAT (NA,NB,NC,ND,B(J))
27[     CONTINUE
28[     IF(KPRNT(9).GE.1) WRITE (6,98[) NEQ1,NEQ
29[     CONTINUE
        GO TO 4[
C
C       CASCADES
C
3[[     DO 5[[ L=1,NCSD
        IF(KPRNT(9).GE.1)  WRITE (6,102[) L
        NEQ1=NEQ+1
        MODE=KC(L,1)
        INS(1)=KC(L,2)
```

```
      JNO(1)=KC(L,3)
      INS(2)=KC(L,4)
      JNO(2)=KC(L,5)
      IF(KPRNT(9).GE.1) WRITE (6,940) MODE,INS(1),JNO(1),INS(2),JNO(2)
      CALL CHKLIM (MODE,1,2,3)
      DO 310 I=1,2
      CALL CHKLIM (INS(I),1,NST,4)
      CALL CHKLIM (JNO(I),1,NST,4)
      CALL SETTOT (JNO(I))
310   CONTINUE
      DO 330 I=1,NCP
      NEQ=NEQ+1
      DO 320 J=1,2
      CALL SETT (INS(J),I,1.)
      CALL SETT (JNO(J),I,-1.)
320   CONTINUE
330   CONTINUE
C
C     READ THE FIRST SET OF DATA
C
      NA=KC(L,6)
      NB=KC(L,7)
      CALL SERAC(NB,NA,B)
      IF(KPRNT(9).GE.1) WRITE (6,940) NA,NB
      IF(KPRNT(9).GE.1)  WRITE (6,950) (B(J),J=1,NCP)
C
C     MODE1
C
      N1=0
      N2=0
      N1A=0
      N2A=0
      DO 370 I=1,2
      IF (INS(I)-NB) 350,340,350
340   N1=I
      N1A=3-I
350   IF (JNO(I)-NA) 370,360,370
360   N2=I
      N2A=3-I
370   CONTINUE
      IF (N1*N2) 380,380,390
380   CALL ERREP (8)
      GO TO (530,440), MODE
390   GO TO (430,440), MODE
400   IF (N1.EQ.N2) GO TO 420
      DO 410 I=1,NCP
      B(I)=1.-B(I)
      CALL SETRAT (JNO(N2A),I,INS(N1),I,B(I))
410   CONTINUE
      GO TO 530
420   DO 430 I=1,NCP
      CALL SETRAT (JNO(N2),I,INS(N1),I,B(I))
430   CONTINUE
      GO TO 530
C
C     MODE 2
C
C     READ THE SECOND SET OF DATA
440   NC=KC(L,8)
      ND=KC(L,9)
      CALL SERAC(ND,NC,ALPHA)
      IF(KPRNT(9).GE.1) WRITE (6,940) NC,ND
      IF(KPRNT(9).GE.1) WRITE (6,950) (ALPHA(J),J=1,NCP)
      IF (ND-INS(N1)) 460,450,460
450   CALL ERREP (8)
      GO TO 530
460   IF (NC-JNO(N2)) 470,510,470
470   IF (NC-JNO(N2A)) 480,490,480
480   CALL ERREP (8)
      GO TO 530
490   DO 500 I=1,NCP
      ALPHA(I)=1.-ALPHA(I)
500   CONTINUE
510   DO 520 I=1,NCP
      CALL SETPI (JNO(N2),I,INS(N1),B(I),INS(N1A),ALPHA(I))
520   CONTINUE
```

```
  530 IF(KPRNT(9).GE.1) WRITE (6,980) NEQ1,NEQ
540   CONTINUE
      GO TO 55
C
C     REACTORS
C
550   DO 730 L=1,NETR
      IF(KPRNT(9).GE.1)   WRITE (6,130) L
      NEQ1=NEQ+1
      MODE=KR(L,1)
      NIN=KR(L,2)
      NOUT=KR(L,3)
      IF(KPRNT(9).GE.1)  WRITE (6,940) MODE,NIN,NOUT
      CALL CHKLIM (MODE,1,3,3)
      CALL CHKLIM (NIN,1,NST,4)
      CALL CHKLIM (NOUT,1,NST,4)
      CALL SETTOT (NOUT)
      GO TO (590,560,580), MODE
  560 NCREF=KR(L,4)
      CALL CRPAC(NOUT,NIN,NCREF,R)
      XX=1.0-R
      IF(MMM.EQ.1) XX=R
      IF(KPRNT(9).GE.1)   WRITE(6,960) NCREF,XX
      CALL CHKLIM (NCREF,1,NCP,5)
      XX=1.0-XX
      IF(XX) 570,590,590
570   CALL ERREP (9)
      GO TO 590
  580 NCREF=KR(L,4)
      NC=KR(L,5)
      ND=KR(L,6)
      CALL RATIO(NOUT,NCREF,NC,ND,XX)
      IF(KPRNT(9).GE.1)  WRITE(6,970) NCREF,NC,ND,XX
      CALL CHKLIM (NCREF,1,NCP,5)
      CALL CHKLIM (NC,1,NST,4)
      CALL CHKLIM (ND,1,NCP,5)
  590 CALL STOICF(NOUT,NIN,B)
      IF(KPRNT(9).GE.1)  WRITE (6,950) (B(J),J=1,NCP)
      BMAX=0.0
      NNREF=0
      DO 610 J=1,NCP
      IF (ABS(B(J))-BMAX) 610,610,600
600   BMAX=ABS(B(J))
      NNREF=J
610   CONTINUE
      IF (BMAX) 620,620,630
620   CALL ERREP (10)
630   DO 640 J=1,NCP
      B(J)=B(J)/BMAX
640   CONTINUE
      DO 690 J=1,NCP
      IF (J-NNREF) 650,690,650
650   IF (B(J)) 660,670,660
660   NEQ=NEQ+1
      B1=1.0/B(J)
      B2=1.0/B(NNREF)
      CALL SETT (NIN,J,B1)
      CALL SETT (NOUT,J,-B1)
      CALL SETT (NIN,NNREF,-B2)
      CALL SETT (NOUT,NNREF,B2)
      GO TO 680
670   NEQ=NEQ+1
      CALL SETT (NIN,J,1.0)
      CALL SETT (NOUT,J,-1.0)
680   CONTINUE
      GO TO (700,710,710),MODE
700   NC=NIN
      ND=NCREF
  710 CALL SETPAT(NOUT,NCREF,NC,ND,XX)
  720 IF(KPRNT(9).GE.1)   WRITE (6,980) NEQ1,NEQ
730   CONTINUE
      GO TO 60
C
C     EXTRA CONSTRAINTS
C
740   NEQ1=NEQ+1
```

```
      IF(KPRNT(9).GE.1)    WRITE (6,1040)
      DO 745 L=1,NEXTRA
      NA=KE(L,1)
      NB=KE(L,2)
      NC=KE(L,3)
      ND=KE(L,4)
      LLL=LLL+1
      R=AX(LLL)
      IF(KPRNT(9).GE.1)     WRITE(6,1010) NA,NB,NC,ND,R
      CALL CHKLIM (NA,1,NST,4)
      CALL CHKLIM (NB,1,NVAR,5)
C
C     EXTRA CONSTRAINTS-AMMENDMENT AUG,1975
C
      IF(NC.NE.0.OR.ND.NE.0) GO TO 745
      CALL SETVAL(NA,NB,R)
      GO TO 750
  745 CONTINUE
      CALL CHKLIM (NC,1,NST,4)
      CALL CHKLIM (ND,1,NVAR,5)
      CALL SETRAT (NA,NB,NC,ND,R)
  750 CONTINUE
      IF(KPRNT(9).GE.1)    WRITE (6,980) NEQ1,NEQ
C
C     SET-UP NOW COMPLETE, WRITE SOME TECHNICAL INFORMATION
C
  760 CALL SECOND (TIME)
      TIME=TIME-TSTART
      IF(KPRNT(9).GE.1)    WRITE (6,1050) NEQ,NEL,NEM,TIME
      LERR=0
      CALL ERREP (LERR)
      IF (LERR) 910,770,910
C
C     NOW CALL THE TRIANGULARISATION SUBROUTINE
C
  770 CALL TRGB (NX,NW,A,JCOL,IROW,NEL,NEQ,NEM,XMIN)
      CALL SECOND (TIME)
      TIME=TIME-TSTART
      NWSP=3*NEL+2*NEQ+NX+1
      IF(KPRNT(9).GE.1)     WRITE (6,1060) NEQ,NEL,NX,TIME,NWSP
C
C       AMENDMENT (AUGUST 1976)
C       KK(11)=1 ... WRITE STREAM COMPONENTS IN MOLE UNITS
C       KK(11)=0 ... WRITE STREAM COMPONENTS IN WEIGHT UNITS
C
C     NOW WRITE THE MAIN OUTPUT
C
      IF(KPRNT(9).LE.0) GO TO 915
      WRITE (6,1070)
      WRITE (6,1080) NPROB,NCASE
      N1=NEQ+2
      N2=NEQ+NEL+1
      LCHECK=0
      DO 790 I=N1,N2
      IF (NW(I)) 790,780,780
  780 LCHECK=1
      A(NX+I-NEQ-1)=-9999.9999
  790 CONTINUE
      IF (LCHECK) 830,830,800
  800 WRITE (6,1090)
      WRITE (6,1100)
      DO 820 N=1,NEQ
      IF (NW(N)) 820,810,810
  810 WRITE (6,1110) N
  820 CONTINUE
  830 DO 840 N=1,NST
      IROW(N)=N
  840 CONTINUE
      DO 900 N=1,NST,5
      N2=N+4
      IF (N2-NST) 860,860,850
  850 N2=NST
  860 WRITE (6,1120) (IROW(NN),NN=N,N2)
      DO 861 L=1,5
  861 SUMS(L)=0.
```

```
      DO 9.. M=1,NVAR
      LB=1
      DO 87. L=N,N2
      JK=M+(L-1)*NVAR
      B(LB)=A(JK+NX)
      LB=LB+1
  87. CONTINUE
      N3=LB-1
      IF (M-NVAR) 880,890,890
  880 CONTINUE
      IF(KK(11).LT.1) GO TO 882
      DO 881 LL=1,N3
  881 B(LL)=B(LL)/AMW(M)
  882 WRITE(6,1130) M,(B(LL),LL=1,N3)
      IF(KK(11).LT.1) GO TO 900
      DO 883 LL=1,N3
  883 SUMS(LL)=SUMS(LL)+B(LL)
      GO TO 900
  890 IF(KK(11).LT.1) GO TO 891
      WRITE(6,1140) (SUMS(LL),LL=1,N3)
      GO TO 900
  891 WRITE(6,1140) (B(LL),LL=1,N3)
  900 CONTINUE
      WRITE (6,1150)
  910 CALL SECOND (TIME)
      TIME=TIME-START
      WRITE (6,1160) TIME,LEPR
  915 CONTINUE
      IF(LOOP.EQ.999.AND.NIT.EQ.999) GO TO 1500
      CALL CONTL4(NX,NSCE)
      IF(NIT.EQ.999) GO TO 1500
      CALL FEEDBK(NST,NX)
C
  920 FORMAT (1H1,≠SYSTEM DATA≠,/,2I6,/,7I6)
  930 FORMAT (1H0,≠SOURCE≠,I3)
  940 FORMAT (1H ,10I6)
  950 FORMAT (5(2X,E12.4))
  960 FORMAT (1H ,I6,2X,E12.4)
  970 FORMAT (1H ,3I6,2X,E12.4)
  980 FORMAT (1H ,≠EQUATIONS≠,I6,≠ TO≠,I5,/)
  990 FORMAT (1H0,≠MIXER≠,I3)
 1000 FORMAT (1H0,≠SEPARATOR≠,I3)
 1010 FORMAT (1H ,4I6,2X,E12.4)
 1020 FORMAT (1H0,≠CASCADE≠,I3)
 1030 FORMAT (1H0,≠REACTOR≠,I3)
 1040 FORMAT (1H0,≠EXTRA CONSTRAINTS≠)
 1050 FORMAT (1H0,I5,≠ EQUATIONS≠,/,I6,≠ VARIABLES≠,/,I6,≠ ELEMENTS≠,/,F
     16.2,≠ SECS≠,/)
 1060 FORMAT (1H0,I5,≠ EQUATIONS≠,/,I6,≠ VARIABLES≠,/,I6,≠ ELEMENTS≠,/,F
     16.2,≠ SECS≠,/,I6,≠ WORKSPACE≠)
 1070 FORMAT (1H1,≠SYMBOL≠,//,1X,≠MASS BALANCE PROGRAM, MARK2≠,//,30X,≠
     1MASS BALANCE≠)
 1080 FORMAT (1H0,≠PROBLEM≠,I10,38X,≠CASE≠,I10)
 1090 FORMAT (1H0,≠CERTAIN VALUES ARE UNSET, THE DUMMY VALUE≠,2X,≠******
     1** IS USED≠)
 1100 FORMAT (1H0,≠CERTAIN EQUATIONS ARE INCONSISTENT OR REDUNDANT≠,/,≠
     1THE FOLLOWING EQUATIONS MAY ASSIST DIAGNOSIS≠,//,5X)
 1110 FORMAT (2X,≠EQUATION≠,I5)
 1120 FORMAT (1H0,5HSTREAM,5(I8,5X),/)
 1130 FORMAT (1H ,I4,5(3X,E10.4))
 1140 FORMAT (1H ,5HTOTAL,2X,F10.4,5(3X,F10.4))
 1150 FORMAT (1H0,≠OUTPUT COMPLETE≠)
 1160 FORMAT (1H0,F6.2,≠ SECS≠,8X,≠ERRORS≠,I3)
 1500 CONTINUE
      END
```

APPENDIX B LISTING OF SUBROUTINE EQK

```
      SUBROUTINE EQK(N,TC,AK1,AK2,AK3,AK4)
C     CALCULATION OF EQUILIBRIUM CONSTANTS
C     N=1  AK1 FOR REACTION CH4+H2O=3*H2+CO
C     N=2  AK2 FOR REACTION CO+H2O=H2+CO2
C     N=3  AK3 FOR 3H2+CO=CH4+H2O
C     N=4      AK4 FOR 4H2+CO2=CH4+2*H2O
C     NU=STOICHIOMETRIC COEFFICIENTS FOR REACTION 1
C     MU=STOICHIOMETRIC COEFFICIENTS FOR REACTION 2
C     LU=STOICHIOMETRIC COEFFICIENTS FOR REACTION 4
C     A(I,J) COEFFICIENTS FOR ENTHALPIES ABOVE 0 C FROM KJAR IN
C     REGRESSION H=A(1,COMP)+A(1,COMP)*T +A(2,COMP)*T*T+A(3,COMP)*T*T*T
C
C      ETC..COEFICIENTS ARE LISTED IN ORDER FOR GASES CH4,H2O,H2,CO,CO2.
      DIMENSION A(5,5),SA(5),SB(6),NU(5),MU(5),LU(5),ANU(5)
      REAL NU,MU,LU
      DATA (A(1,K),K=1,5)/-19625.11,-60059.65,-2093.47,-28438.67,-96224.
     166/
      DATA(A(2,K),K=1,5)/3.3595953,7.26073636,7.0912215,6.7325148,5.3753
     134/
      DATA(A(3,K),K=1,5)/8.4959047E-03,9.8842169E-04,-3.2010327E-04,-2.3
     105115CE-05,-.26509E-03/
      DATA(A(4,K),K=1,5)/-7.1126382E-07,3.7872408E-07,2.9556076E-07,7.38
     160226E-07,-3.0163581E-06/
      DATA(A(5,K),K=1,5)/-2.5486782E-10,-1.2904534E-10,-2.9377E5E-11,-2.4
     137179 1E-10,5.3576149E-10/
      DATA NU/-1.,-1.,3.,1.,0./
      DATA MU/0.,-1.,1.,-1.,1./
      DATA LU/1.,2.,-4.,0.,-1./
C
      P=1.9865
      DO 99 I=1,5
   99 SA(I)=0.
      DO 1010 I=1,5
      GO TO (1,2,1,4),N
    1 ANU(I)=NU(I)
      GO TO 1010
    2 ANU(I)=MU(I)
      GO TO 1010
    4 ANU(I)=LU(I)
 1010 CONTINUE
C
      DO 101 J=1,5
      DO 100 I=1,5
  100 SA(J)=ANU(I)*A(J,I)+SA(J)
  101 CONTINUE
C
      T=298.16
      AA=-SA(1)/T+SA(2)*ALOG(T)+T*(SA(3)+T*(SA(4)/2.+SA(5)*T/3.))
      T=TC+273.16
      SB(1)=SA(2)/P
      SB(2)=-SA(1)/P
      SB(3)=-(SF2F(ANU)/298.16+AA)/P
      SB(4)=SA(3)/P
      SB(5)=SA(4)/(2*P)
      SB(6)=SA(5)/(3*P)
      AK=EXP(SB(1)+ALOG(T)+SB(2)/T+SB(3)+T*(SB(4)+T*(SB(5)+SB(6)*T)))
      GO TO (201,202,203,204),N
  201 AK1=AK
      GO TO 205
  202 AK2=AK
      GO TO 205
  203 AK3=1./AK
      GO TO 205
  204 AK4=AK
  205 CONTINUE
      RETURN
      END
```

```
      FUNCTION SF25(ANU)
      DIMENSION ANU(5),F25(5)
      DATA F25/-12141.,-54535.,0.,-32807.7,-94259.8/
      SF25=0.
      DO 100 I=1,5
  100 SF25=SF25+ANU(I)*F25(I)
      RETURN
      END
```

APPENDIX C

PHYSICAL PROPERTY PACKAGE

The physical property library routine is organized in three subroutines as follows:

(1) Subroutine READIN which reads from cards the coefficients for the regression equations for a particular property, for example the coefficients a_i in

$$C_p = a_1 + a_2T + a_3T^2 + a_4T^3$$

(2) Subroutine PRØCAL which calculates a property of a pure component at a given specified condition

(3) Subroutine PRØMIX which calculates the property of a mixture either for the entire flowing mixture or per mole for the mixture by using the pure component properties as calculated from PRØCAL.

These subroutines will be listed after the essential details of each subroutine is presented.

C.1 Subroutine READIN

The correlation coefficients for the properties are contained in the matrix PRØP(I,J,K), where I refers to the code number (called NØPP) which is saved in blank common in the self standing GEMCS executive; in the combined GEMCS-SYMBØL executive, it is saved in the labelled common /PPD/ since it is used only by the GEMCS overlay. It is therefore

required to set a dictionary of code numbers for each simulation. In the naphtha reformer plant simulation, for instance, NOPP=1 stands for a physical property (the heat capacity) of a gaseous stream while NOPP=2 represents the heat capacity of a liquid stream. J refers to a particular component of all NØCØMP components used in the simulation; K refers to the K^{th} coefficient in the regression equation for component J and property I.

Table C.1 shows the data set (in order) required by READIN.

Table C.1 Data Set for READIN[1]

Card Number	Variable Name(s)	Particular	FØRMAT
1	NPP	Number of unique properties to be read	I5
2	I,LØC(I)	Property identifier (=NØPP), Number of coefficients in the regression equations (must be the same for all components for property I	2I5
3	PRØP(I,1,K)	The coefficients in the regression equation (maximum of 5) for component 1 (location 6 in the stream list)	5E12.5
4	PRØP(I,2,K)	The coefficients for component 2 as in 3	5E12.5
.			
.			
.			
NØCØMP+2	PRØP(I,NØCØMP,K)	The coefficients for component NØCØMP[2].	5E12.5
NØCØMP+3	I,LOC(I)	Property identifier and number of coefficients for property I	2I5
.			
.			
.			
etc			

1 - The total number of cards must be NPP*(NØCØMP+1)+1

2 - If this is a gas property, and component NØCØMP is a liquid, a card of zeroes (blank card) must be read.

```
      SUBROUTINE READPP
C
C      THIS SUBROUTINE  READS IN THE PHYSICAL PROPERTY PACKAGE
C
      COMMON NS(12),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NTT,SN(72,16)
      COMMON/PPDATA/ NPPC(17),LOC(17),PROP(17,16,4)
      DIMENSION NO(17),NTYPE(17)
      WRITE(6,16)
      READ(5,11)NPP
      WRITE(6,15) NPP
      DO 1 I=1,NPP
      READ(5,12) I,LOC(I)
      NOPP=I
      L=LOC(I)
      WRITE(6,17) NOCOMP,NOPP
      DO 2 J=1,NOCOMP
      READ(5,13) (PROP(I,J,K),K=1,L)
      IF (KPRNT(7).EQ.0) GO TO 10
      WRITE(6,14) (PROP(I,J,K),K=1,L)
10    CONTINUE
 2    CONTINUE
 1    CONTINUE
17    FORMAT(1X,*FOR THE*,I3,* COMPONENTS,THE PARAMETERS FOR NOPP=*,I3,*
     1 ARE *,/)
14    FORMAT(5E12.5)
15    FORMAT(10X,*READING*,I7,*PHYSICAL PROPERTY SETS*)
13    FORMAT(5E12.5)
16    FORMAT(20X,*PHYSICAL PROPERTIES*)
11    FORMAT(I5)
12    FORMAT(2I5)
      RETURN
      END
```

C.2 Subroutine PRØCAL

The information required by PRØCAL (NØPP, LOC(NØPP), PRØP (NØPP,J,K),X,NØCØMP) is transmitted through blank common* and through labelled common /PPDATA/. The calculated (PPSC(NØPP,J)) is returned through blank CØMMØN. A call to PRØCAL requires that NØPP and X be defined. (LOC(NOPP) and PROP(NOPP,J,K) are assumed to be read in by READIN and transferred through labelled common /PPDATA/). X is the correlating variable and is assumed to be absolute temperature in the correlation; it is assumed to be transmitted (through common) as degrees Celsius. PRØCAL calculate the physical property, NØPP, for all NØCOMP components; these values are available as PPSC(NØPP,J) for the j^{th} component.

C.3 Subroutine PRØMIX

The information required by PRØMIX(NØPP,XMOLE(J),NØCØMP) is transmitted through blank common*. (XMOLE(J),J=1,NØCØMP) is the flow-rate of each of the NØCØMP components in the stream in the units used in the stream list and property calculation. When calling PROMIX(M), NØPP must be defined; the XMOLE(-) vector must be filled; and the temperature, X, must be specified. The variable M must be specified as either 1 or 2 as follows

---

*This is true for the self-standing GEMCS. When PROCAL and PRØMIX are used in the combined GEMCS-SYMBØL executive, only NØCØMP is transferred through blank common ; the remaining data are transferred through labelled common /PPD/ and /PPDATA/.

(a) when M=1, the physical property of the whole mixture is calculated as

$$PPMX(N\emptyset PP) = \sum_{I=1}^{N\emptyset C\emptyset MP} XMOLE(I)*PPSC(N\emptyset PP,I)$$

(b) when M=2, the molar physical property of the whole mixture is calculated as

$$PPMX(N\emptyset PP) = \sum_{I=1}^{N\emptyset C\emptyset MP} XM\emptyset LE(I)*PPSC(NOPP,I) / \sum_{I=1}^{N\emptyset COMP} XM\emptyset LE(I)$$

```
      SUBROUTINE PROCAL
C
C  CALCULATES THE PROPERTY OF A PURE COMPONENT AT A GIVEN
C                              SPECIFIED CONDITION
C
      COMMON NS(12,),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NTT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(15,0),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSE,NCOUN
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      COMMON/PPDATA/ NTPC(17),LOC(17),PROP(17,16,4)
      N=NOPP
      L=LOC(N)
      X=X+273.16
      DO 120 J=1,NOCOMP
      E=0.
      DO 110 K1=1,L
      K=L+1-K1
      E=PROP(N,J,K)+E*X
  110 CONTINUE
  120 PPSC(N,J)=E
      IF(KPRNT(8).EQ.0 ) GO TO 200
      DO 130 J=1,NOCOMP
      WRITE(6,999)N,J,PPSC(N,J)
  130 CONTINUE
  999 FORMAT(1X,*PROPERTY*,I5,*FOR COMPONENT*,I5,*IS*,F12.5)
  200 RETURN
      END
```

```
      SUBROUTINE PROMIX(N)
C
C     CALCULATES THE PROPERTY OF A MIXTURE  EITHER FOR THE ENTIRE
C     FLOWING MIXTURE (N=1) OR PER MOLE FOR THE MIXTURE (N=2) BY USING
C     THE PURE  COMPONENT PROPERTIES AS CALCULATED FROM PROCAL
C
      COMMON NS(12),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NT,SP(72,16)
      COMMON/GEMCSI/LLST(12),EEN(150),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSP,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      COMMON/PPDATA/ NTPC(17),LOC(17),PROP(17,16,4)
      IF(N.LT.1.OR.N.GT.2 ) GO TO 100
      N=NOPP
      IF(N.EQ.1 ) GO TO 100
      CALL PROCAL
      IF(N.GT.1 ) GO TO 20
      PPMX(1)=0.
      DO 11 J=1,NOCOMP
      PPMX(1)=PPMX(N)+PPSC(N,J)*XMOLE(J)
   11 CONTINUE
      GO TO 200
   20 TMOLE=0.
      DO 21 J=1,NOCOMP
      TMOLE=TMOLE+XMOLE(J)
   21 CONTINUE
      PPMX(N)=0.
      DO 22 J=1,NOCOMP
      XMOLE(J)=XMOLE(J)/TMOLE
      PPMX(N)=PPMX(N)+ PPSC(N,J)*XMOLE(J)
   22 CONTINUE
      GO TO 200
  100 CONTINUE
      WRITE(6,15) NE
   15 FORMAT(1X,*ERROR ON CALL TO PHYSICAL PROPERTIES FROM EQUIPMENT NUM
     *BER*,I5)
  200 RETURN
      END
```

Table D.1

Composition of the Naphtha feed

	Composition(%)
	0.20
	0.70
	4.40
isopentane (C_5H_{12})	18.40
n-pentane (C_5H_{12})	20.80
2-2-dimethylbutane (C_6H_{14})	0.40
cyclopentane + 2-methylpentane (C_6H_{14})	10.00
3-methylpentane (C_6H_{14})	4.30
n-hexane (C_6H_{14})	9.70
methylcyclopentane (C_6H_{12})	3.80
cyclohexane (C_6H_{12})	2.90
Dimethylpentane and other hydrocarbons (C_7H_{16})]	2.10
N.C7 (C_7H_{16})	4.20
methylcyclohexane (C_7H_{14})	3.10
C_8H_{18}	6.60
C_9H_{20}	1.30
benzene (C_6H_6)	0.65
toluene (C_7H_8)	1.10
ethylbenzene (C_8H_{10})	0.15
m- and p-xylene (C_8H_{10})	0.40
O-xylene (C_8H_{10})	0.10
C_9H_{12} (C_9H_{12})	0.05

APPENDIX D

NAPHTHA FEEDSTOCK COMPOSITION

Table D.1 lists the composition of a sample of the naphtha fed to the reforming plant. The fictious formula $C_{5.6488}H_{12.579}$ with molecular weight 80.576 was calculated from the values in this table. The naphtha feed is a mixture of hydrocarbons.

Table D.1

Composition of the Naphtha feed

hydrocarbon	Composition(%)
propane	0.20
isobutane	0.70
n-butane	4.40
isopentane (C_5H_{12})	18.40
n-pentane (C_5H_{12})	20.80
2-2-dimethylbutane (C_6H_{14})	0.40
cyclopentane + 2-methylpentane (C_6H_{14})	10.00
3-methylpentane (C_6H_{14})	4.30
n-hexane (C_6H_{14})	9.70
methylcyclopentane (C_6H_{12})	3.80
cyclohexane (C_6H_{12})	2.90
Dimethylpentane and other hydrocarbons (C_7H_{16})]	2.10
N.C7 (C_7H_{16})	4.20
methylcyclohexane (C_7H_{14})	3.10
C_8H_{18}	6.60
C_9H_{20}	1.30
benzene (C_6H_6)	0.65
toluene (C_7H_8)	1.10
ethylbenzene (C_8H_{10})	0.15
m- and p-xylene (C_8H_{10})	0.40
O-xylene (C_8H_{10})	0.10
C_9H_{12} (C_9H_{12})	0.05

APPENDIX E

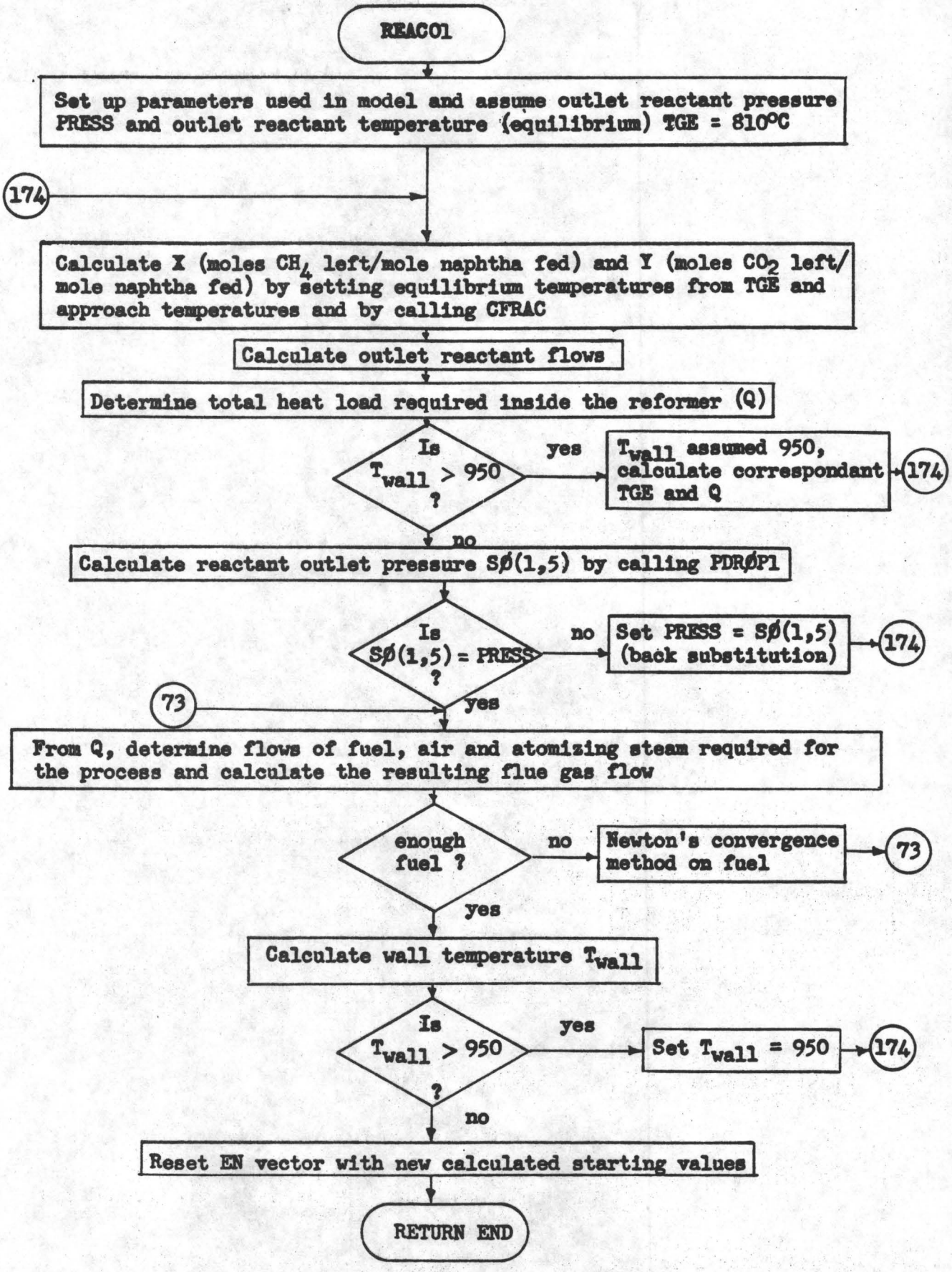
FLOWCHARTS AND LISTINGS OF THE MODELS OF THE NAPHTHA REFORMER PLANT SIMULATION

E.1 Reactor Models

REACO1 and FRACS

REACO2

REACO3

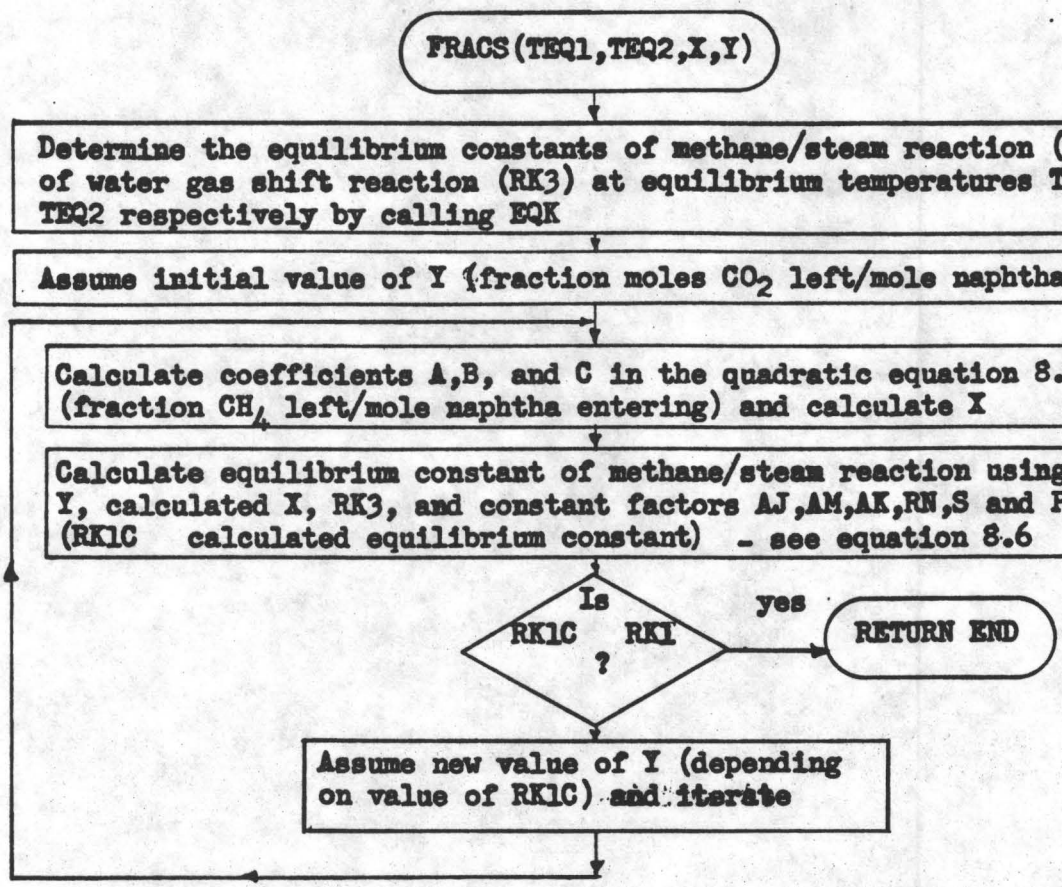

REACO1
Set up parameters used in model and assume outlet reactant pressure PRESS and outlet reactant temperature (equilibrium) TGE = 810°C
174
Calculate X (moles CH_4 left/mole naphtha fed) and Y (moles CO_2 left/mole naphtha fed) by setting equilibrium temperatures from TGE and approach temperatures and by calling CFRAC
Calculate outlet reactant flows
Determine total heat load required inside the reformer (Q)
Is $T_{wall} > 950$?
yes
T_{wall} assumed 950, calculate correspondant TGE and Q
174
no
Calculate reactant outlet pressure SØ(1,5) by calling PDRØP1
Is SØ(1,5) = PRESS ?
no
Set PRESS = SØ(1,5) (back substitution)
174
73
yes
From Q, determine flows of fuel, air and atomizing steam required for the process and calculate the resulting flue gas flow
enough fuel ?
no
Newton's convergence method on fuel
73
yes
Calculate wall temperature T_{wall}
Is $T_{wall} > 950$?
yes
Set T_{wall} = 950
174
no
Reset EN vector with new calculated starting values
RETURN END

```
      SUBROUTINE REAC01                                                 256
C
C       SIMULATES THE REFORMER REACTOR
C
C     ** EN(16)=TGE=ESTIMATED TEMP.OF GAS LEAVING THE REFORMER        **
C     ** EN(17)= APPROACH TEMPERATURE FOR WGS REACTION
C                                          CO + H2O = CO2 + H2
C     ** EN(18)=C/H=MASS RATIO CARBON/HYDROGEN IN FUEL                **
C     ** EN(19)=FRACTIO OF EXCESS AIR TO FURNACE                      **
C     ** EN(20)=ATOMIZED STEAM RATIO ENTERING FURNACE-MOLSTEAM/KGMFUEL**
C     ** EN(21)=TW=WALL TEMPERATURE-CONTROLLED                       **
C     ** EN(22)=HF=HEAT OF FORMATION OF NAPHTHA AT TB                 **
C     ** EN(23)=HFCO=HEAT OF FORMATION OF CO                          **
C     ** EN(24)=HFST=HEAT OF FORMATION OF WATER VAPOUR
C     ** EN(25)=HFCHA=HEAT OF FORMATION OF CHA                       **
C     ** EN(26)=HRWGS=HEAT OF REACTION OF WATER GAS SHIFT             **
C     ** EN(27)=TB=BASE TEMPERATURE -25C                              **
C     ** EN(28)=UA=OVER-ALL HEAT TRANSFER COEFFICIENT    **
C     ** EN(29)=HAO=HEAT TRANSFER COEFFICIENT-AREA PRODUCT FOR OUTSIDE REFO
C     ** EN(30)=HC=HEAT OF COMBUSTION OF FUEL AT TB                  **
C     ** EN(31)=WFUEL=ESTIMATED FLOW OF INLET FUEL KG/HR              **
C     ** EN(32)=RN=NUMBER OF ATOM OF CARBON IN ONE MOLECULE OF NAPHTHA**
C     ** EN(33)=Z=PROPORTION FACTOR OF HYDROGEN ATOM                  **
C     ** EN(34)=HAI=HEAT TRANSFER COEFFICIENT-AREA PRODUCT FOR INSIDE REFOR
C     ** EN(35)= APPROACH TEMPERATURE FOR THE CH4 REACTION
C                                          CH4 + H2O = 3*H2 + CO
C     ** EN(36)=HEAT OF FORMATION OF CO2
C     ** EN(37)=HEAT OF FORMATION OF LIQUID WATER
C     ** EN(38)= FRACTION HEAT LOSS (=0.06*HEAT LOAD ON REACTOR TUBES,
C                                    CF. RHODIA DESIGN CALCULATIONS)
C     ** EN(39)=CORRECTION FOR WALL TEMPERATURE
C     ** EN(40)=FACP=FACTOR FOR PRESSURE DROP CALCULATION
C     ** EN(41)=PRESSURE DROP ACROSS THE REACTOR
C
C**STREAM ORDER DEISGNATION     **
C     SI(1,-)= REACTANT GASES(NAPHTHA+STREM) ENTERING INSIDE TUBES  **
C     ** SI(2,-)= ATOMIZED STEAM ENTERING THE FURNACE              **
C     ** SO(1,-)= GASES RESULTING FROM THE REFORMING REACTION,LEAVING
C     THE REFORMER  **
C     ** SO(2,-)= FLUE GAS LEAVING THE FURNACE  **
C     ** SO(3,-) AIR ENTERING THE FURNACE(TEMPERATURE AT FED IN WHIT INPUT
C     STREAM INFO)  **
C     ** SO(4,-)= FUEL ENTERING THE FURNACE, FOR COMBUSTION (TEMPERATURE AT
C     FED IN WITH INPUT STREAM INFO)                **
C
      COMMON NS(12),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(150),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSR,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      COMMON/P/ AJ,AM,AK
      COMMON/P/ICI,ICO,FACP
C     COMMON/NAEJ/EQNU,ICC
      DIMENSION TGAUZ(2),WFUAX(2)
      DIMENSION WFUEL(2),DF(2)
      SO(1,11)=SI(1,11)
      AK=SI(1,11)/SI(1,17)
      AJ=SI(1,10)/SI(1,17)
      AM=SI(1,9)/SI(1,13)
      ISW=1
      KW=0
      I7=0
      I8=0
  212 ITGE=1
C     ** ICH1=1 NORMAL CONDITION                                      **
      ICH1=1
      ICH2=1
      I6=0
      IDEP=1
 1991 FORMAT(/,10X,'ESTIMATED OUTLET PRESSURE =',F9.5)
      TGE=EN(16)
  174 TEOF=EN(16)+EN(35)
      TEOW=EN(16)+EN(17)
      SO(1,5)=SI(1,5)-EN(41)
      PRESS=SO(1,5)
      IF(EN(5).GE.1) WRITE(6,1991) PRESS
```

```
      CALL FEACS(TEOF,TEOW,X,Y)
      IF(EN(5).GE.1.) WRITE(6,200) TEOF
  200 FORMAT(/10X,'EO. TEMP.='F12.4)
      SO(1,7)=SI(1,13)*Y
      SO(1,8)=SI(1,13)*(EN(32)-X-Y+AM)
      SO(1,9)=SI(1,13)*X
      SO(1,10)=SI(1,13)*(EN(32)*EN(33)/2.-3.*X+EN(32)+Y+AJ+3.*AM)
      S=SI(1,6)/(SI(1,13)*EN(32))
      SO(1,6)=SI(1,13)*( S     *EN(32)-EN(32)+X-Y-AM)
      DO 1221 KS=12,16
 1221 SO(1,KS)=0.
      SO(1,3)=0.
      DO 50 I=1,NOCOMP
      SO(1,3)=SO(1, 3)+SO(1, I+5)
   50 CONTINUE
      XCH4=X
C     ** HEAT BALANCE-INSIDE REFORMER                                    **
C     ** PHYSICS PROPERTIES FOR GASES INLET(NAPHTHA+STEAM)               **
      NOPP=1
      X=(SI(1,4)+EN(27))/2.
      DO 52 J=1,NOCOMP
      XMOLE(J)=SI(1,J+5)
   52 CONTINUE
      CALL PROMIX(1)
      WCIN=PRMX(NOPP)
C     ** PHYSICAL PROPERTIES FOR OUTLE GASES                             **
      X=(EN(16)+EN(27))/2.
      DO 53 J=1,NOCOMP
      XMOLE(J)=SO(1,J+5)
   53 CONTINUE
      CALL PROMIX(1)
      WCOUT=PRMX(NOPP)
CCCCC
C     ** HRX IS THE HEAT OF REACTION PER MOL OF NAPHTHA                  **
        HEAT OF COMBUSTION OF NAPHTHA (HHV) = -11500 KCAL/KGM
        MOLECULAR WEIGHT OF NAPHTHA = 80.576
CCCCC
      RN=EN(32)
      HRX=(RN-XCH4+AM)*(EN(23)-EN(24))+(XCH4-AM)*EN(25)-RN*EN(36)-EN(33
     1*RN*EN(37)/2.-80.576*11500.
      IF(ICH1 .GT. 1) GO TO 161
      A=WCIN*(SI(1,4)-EN(27))-SO(1,7)*EN(26)-SI(1,13)*HRX-WCOUT*(EN(16)
     1EN(27))
      Q=-A
      IF(EN(5).GE.1.) WRITE(6,201) Q
  201 FORMAT(/10X,'Q1=',E15.5)
      GO TO 162
C     ** CALCUL OF TGE=EN(16) AND Q                                      **
  161 Q=EN(34)*((EN(21)-SI(1,4))-(EN(21)-EN(16)))
      QQ=(EN(21)-SI(1,4))/(EN(21)-EN(16))
      Q=Q/ALOG(QQ)
      IF(EN(5).GE.1.) WRITE(6,206) Q
  206 FORMAT(/10X,'Q2=',E15.5)
      A=WCIN*(SI(1,4)-EN(27))-SO(1,7)*EN(26)+Q-SI(1,13)*HRX
      TGEC=A/WCOUT+EN(27)
      CALL CONVE(TGE,TGEC,ITGE,TGAUX)
      IF(EN(5).GE.1.) WRITE(6,202) TGE
  202 FORMAT(/10X,'OUT LET TEMP.='F12.4)
      IF(ITGE.EQ.1) GO TO 61
      IF(I6.GT. 20) GO TO 203
      EN(16)=TGE
      I6=I6+1
      GO TO 174
   60 CONTINUE
  203 WRITE(6,69) NE
   69 FORMAT(/10X,'TGE DID NOT CONVERGE IN REACC1,EQUIPMENT NO',I10)
   61 CONTINUE
      EN(16)=TGE
  162 CONTINUE
      SO(1,4)=EN(16)
CCCCCC
       CALCULATE PRESSURE DROP
CCCCCC
      SO(1,5)=EN(40)
      ICC=1
      RETURN
      END
```

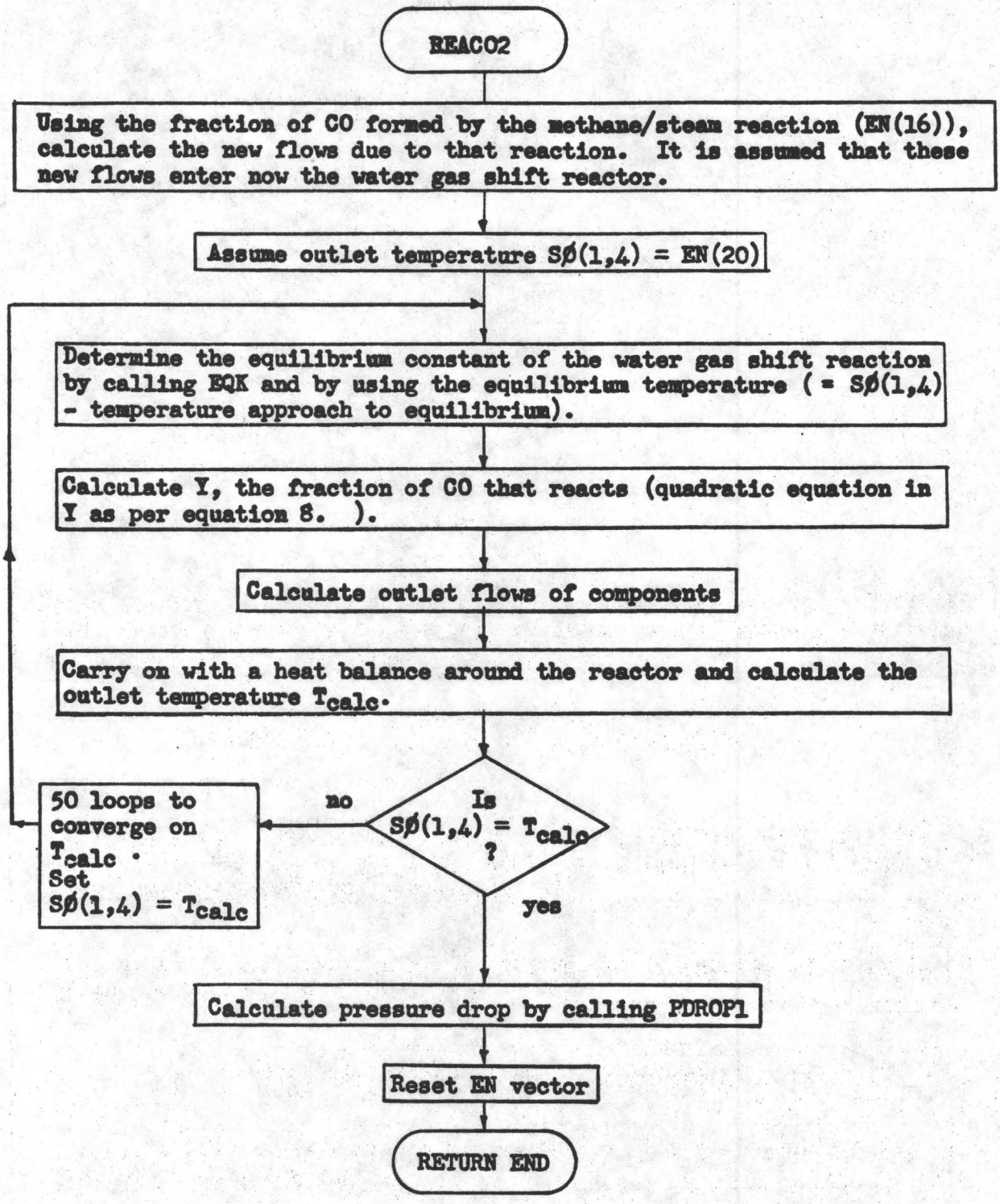
```
C       CALL CALFAC(FACP)
        IISW=2
        ICI=1
        ICO=1
        FACP=EN(40)
        CALL PDROP1(IISW)
        IF(EN(5).GE.1.) WRITE(6,1992) SO(1,5)
 1992 FORMAT(/,10X,*CALCULATED OUTLET PRESSURE =*,F9.5)
        DEPP=(SO(1,5)-PRESS)/SO(1,5)
        IF(ABS(DEPP).LE.0.001) GO TO 1995
        IF(IDEP.GT.10) GO TO 1993
        IDEP=IDEP+1
        EN(41)=SI(1,5)-SO(1,5)
        GO TO 174
 1993 WRITE(6,1994) NE
 1994 FORMAT(/,10X,*OUTLET PRESSURE DID NOT CONVERGE IN REACC1, NE =*,I3
     1)
 1995 CONTINUE
        EN(41)=SI(1,5)-SO(1,5)
C   ** MASS BALANCE IN FURNACE                                          **
        A=EN(18)
        A1=A+1.
        IWFU=1
        I5=5
        WFUEL(1)=EN(31)
   77   SO(4,15)=EN(31)
        SO(4,3)=SO(4,15)
        SO(3,12)=((A/12.+1./4.)/A1)*SO(4,15)*(1.+EN(19))
        SO(3,11)=SO(3,12)*79./21.
        SO(3,5)=SO(3,11)+SO(3,12)
        SO(2,11)=SO(3,11)
        SO(2,7)=((A/12.)/A1)*SO(4,15)
        SI(2,6)=SO(4,15)*EN(20)
        SI(2,3)=SI(2,6)
        SO(2,6)=((1./2.)/A1)*SO(4,15)+SI(2,6)
        SO(2,12)=SO(4,15)*((A/12.+.25)/A1)*EN(19)
        SO(2,3)=0.
        DO 51 I=1,NOCOMP
        SO(2,3)=SO(2,3)+SO(2, I+5)
   51 CONTINUE
C   ** HEAT BALANCE IN FURNACE                                          **
C   ** PHYSICAL PROPERTIES OF FUEL INLET                               **
        NOPP=2
        X=(SO(4,4)+EN(27))/2.
        CALL PPOCAL
        CPFUL=PPSC(NOPP,10)
        WCFUL=CPFUL*SO(4,15)
C   ** PHYSICAL PROPERTIES OF AIR INLET                                 **
        NOPP=1
        X=(SO(3,4)+EN(27))/2.
        DO 55 I=1,NOCOMP
        XMOLE(I)=SO(3,I+5)
   55 CONTINUE
        CALL PROMIX(1)
        WCAIR=PPMX(NOPP)
C   ** PHYSICAL PROPERTIES OF ATOMIZED STEAM                            **
        NOPP=1
        X=(SI(2,4)+EN(27))/2.
        CALL PPOCAL
        CPATS=PPSC(NOPP,1)
        WCATS=CPATS*SI(2,6)
C   ** PHYSCIAL PROPERTIES OF FLUE GAS                                  **
C       OUTLET TEMPERATURE CALCULATION
        IF(ICH1 .LE. 1) GO TO 162
        TFO=0/EN(29)+EN(21)
        IF(EN(5).GE.1.) WRITE(6,208) TFO
  208 FORMAT(/10X,*OUTLET TEMP.=*,F12.4)
        GO TO 167
  162 CONTINUE
        AA=(SO(1,4)-SI(1,4))*EN(26)/0.
        AB=EXP(AA)
        TFO=(ST(1,4)-AB*SO(1,4))/(1.-AB)
  167 SO(2,4)=TFO
        IF(EN(5).GE.1.) WRITE(6,209) TFO
  209 FORMAT(/10X,*OUTLET TEMP2=*,F12.4)
```

```
      C=SO(2,4)
      I4=6
C  ** PHYSICAL PROPERTIES FOR FLUE GAS                    **
      NOPP=1
    3 X=(C+EN(27))/2.
      DO 54 I=1,NOCOMP
      XMOLE(I)=SO(2,I+5)
   54 CONTINUE
      CALL PROMIX(1)
      WCFLU=PPMX(NOPP)
      IF(EN(5).GE.1.) WRITE(6,207) WCFLU
  207 FORMAT(/,10X,'W-CPOG=',F12.4)
C  ** CALCUL OF FLUE GAS FLOW                                **
      D=EN(36)*SO(4,15)+WCFUL*(SO(4,4)-EN(27))+WCATR*(SO(3,4)-EN(27))+
     1WCATS*(SO(2,4)-EN(27)) -EN(38)*Q
      WCOFL=(D-Q)/(TFO-EN(27))
C  ** CONVERGENCE OF FUEL GAS FLOW                          **
      EW=WCOFL/WCFLU
    7 I5=I5+1
      WFUEC=EW*WFUEL(ISW)
C     NEWTON CONVERGENCE METHOD                          ******
      IF(EN(5).GE.1.) WRITE(6,205) WFUEL(ISW)
  205 FORMAT(/10X,'FUEL RATE =',F12.4)
      DF(ISW)=WFUEC-WFUEL(ISW)
      IF(ABS(DF(ISW)/WFUEL(ISW)).LE.0.001) GO TO 71
      IF(I5.GT.50) GO TO 72
      IF(ISW.EQ.1) GO TO 1210
      IF(KW.EQ.1) GO TO 1205
      SLOPE=(DF(2)-DF(1))/(WFUEL(2)-WFUEL(1))
      KW=1
1205  WFUEL(2)=WFUEL(2)-DF(2)/SLOPE
      EN(31)=WFUEL(2)
      GO TO 73
1210  ISW=2
      WFUEL(ISW)=(WFUEC+WFUEL(1))/2.
      EN(31)=WFUEL(ISW)
      GO TO 73
   72 WRITE(6,79) NE
   79 FORMAT(/10X,'W-FUEL DID NOT CONVERGE IN PEACO1, EQUIPMENT NO.',I10
     1)
   71 CONTINUE
      TW=SO(2,4)-Q/EN(29)+EN(39)
      EN(21)=TW
      IF(TW .LE. 950.) GO TO 172
C  ** ICH1=2 TW GT TEMPERATURE LIMIT                        **
      ICH1=2
      EN(21)=950.
      GO TO 174
  172 CONTINUE
      IF(EN(5).GE.1.) WRITE(6,700) WFUEL(2),TFO,TGE,TW
      CALL DISKIO(2,NE)
  700 FORMAT(/10X,10F10.5)
      RETURN
      END
```


FRACS(TEQ1,TEQ2,X,Y)
Determine the equilibrium constants of methane/steam reaction (RK1) and of water gas shift reaction (RK3) at equilibrium temperatures TEQ1 and TEQ2 respectively by calling EQK
Assume initial value of Y (fraction moles CO_2 left/mole naphtha entering)
Calculate coefficients A,B, and C in the quadratic equation 8.7 in X (fraction CH_4 left/mole naphtha entering) and calculate X
Calculate equilibrium constant of methane/steam reaction using assumed Y, calculated X, RK3, and constant factors AJ,AM,AK,RN,S and PT (SØ(1,5)) (RK1C calculated equilibrium constant) - see equation 8.6
Is RK1C RK1 ?
yes
RETURN END
Assume new value of Y (depending on value of RK1C) and iterate

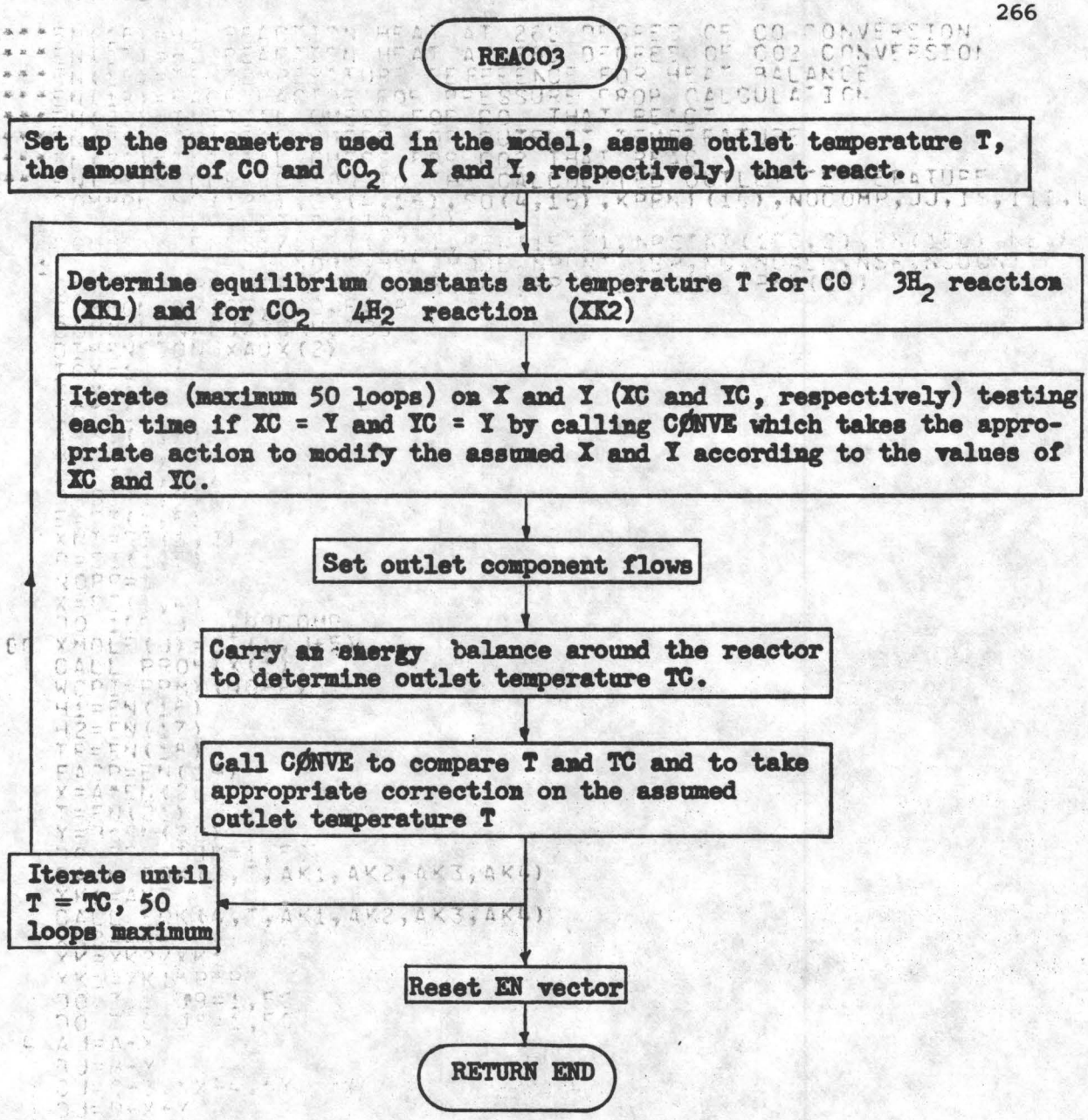
```
      SUBROUTINE FRACS(TEQ1,TEQ2,X,Y)
C
C      SUBROUTINE CALLED BY REAC01 TO ITERATE ON EQUILIBRIUM CONSTANTS
C      TO DETERMINE X AND Y
C      WHERE X = FRACTION MOLES OF CO2 LEFT/MOLE NAPHTHA FED
C            Y = FRACTION MOLES OF CH4 LEFT/MOLE NAPHTHA FED
C
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSR,NCOUNT
      COMMON/P/ AJ,AM,AK
      RN=EN(32)
      Z=EN(33)
      S=SI(1,5)/(SI(1,13)*RN)
      PT=SO(1,5)
      I=1
      CALL EQK(1,TEQ1,AK1,AK2,AK3,AK4)
      RK1=AK1
      CALL EQK(2,TEQ2,AK1,AK2,AK3,AK4)
      RK3=AK2
      IF(EN(5).GE.1.) WRITE(6,102) RK1,RK3
  102 FORMAT(/10X,*RK1=*,E15.5,*RK3=*,E15.5)
      Y=3.
      II1=0
      ICK=1
      GO TO 5
   22 I3=0
      DEK1=0.0001
      DEK2=0.0001
      YD=Y1-Y2
      IF(ABS(YD) .LE. 0.00001) GO TO 100
    1 II=1
      Y=Y1
      GO TO 5
    6 F1=RK1C
    3 YB=(Y1+Y2)/2.
      IF (I3 .GT. 100) GO TO 112
      YD=Y1-Y2
      II=2
      Y=YB
      GO TO 5
    8 IF(ABS(ERRO) .LE. DEK2) GO TO 100
      F2=RK1C
      FF=(F1-RK1)*(F2-RK1)
      IF(FF .GT. 0.) GO TO 2
      Y2=YB
      I3=I3+1
      GO TO 3
    2 Y1=YB
      F1=F2
      I3=I3+1
      GO TO 3
    5 A=2.*RK3
      B=2.*RK3*(2.*RN+2.*AM-S*RN)+6.*Y
      C=-2.*RK3*(-RN*RN-AM*AM+Y*Y+S*RN*RN+AM*S*RN-2.*RN*AM-Y*S*RN)+2.*Y
     1Y+2.*Y*AJ+Z*RN*Y+2.*RN*Y+6.*AM*Y
C     WRITE(6,*) A,B,C,X
      X=B*B-4.*A*C
      X=(B+SQRT(X))/(2.*A)
    7 D=Z*RN-6.*X+2.*RN+2.*Y+6.*AM+2.*AJ
      RK1C=(RN-X-Y+AM)*D*D*PT*PT
      RNT=Z*RN/2.-2.*X+RN+S*RN+AJ+3.*AM+AK
      RK1C=RK1C/(8.*X*(S*RN-RN+X-Y+AM)*RNT*RNT)
      ERRO=(RK1C-RK1)/RK1
        IF(ICK .GT. 1) GO TO 23
      IF(ERRO .GT. 0.) GO TO 21
      IF(II1.GT.50) GO TO 112
      Y=Y+.05
      II1=II1+1
      GO TO 5
   21 ICK=2
      Y2=Y
      Y1=Y2-0.2
      GO TO 22
   23 GO TO (6,8) II
  112 WRITE(6,103)X,Y,RK1C
```

```
      GO TO 24
  100 IF(EN(5).GE.1.) WRITE(6,101) X,Y,PK1C
  103 FORMAT(/,10X,'X=',F10.5,'Y=',F10.5,'K1=',E15.5,/,10X,'DID NOT CON
     1ERGE IN EQUILIBRIUM CONSTANT')
  101 FORMAT(/,10X,'X=',F10.5,'Y=',F10.5,'K1=',E15.5)
   24 RETURN
      END
```


REACO2
Using the fraction of CO formed by the methane/steam reaction (EN(16)), calculate the new flows due to that reaction. It is assumed that these new flows enter now the water gas shift reactor.
Assume outlet temperature SØ(1,4) = EN(20)
Determine the equilibrium constant of the water gas shift reaction by calling EQK and by using the equilibrium temperature (= SØ(1,4) - temperature approach to equilibrium).
Calculate Y, the fraction of CO that reacts (quadratic equation in Y as per equation 8.).
Calculate outlet flows of components
Carry on with a heat balance around the reactor and calculate the outlet temperature Tcalc.
Is SØ(1,4) = Tcalc ?
no
50 loops to converge on Tcalc . Set SØ(1,4) = Tcalc
yes
Calculate pressure drop by calling PDROP1
Reset EN vector
RETURN END

```
      SUBROUTINE REACO2
C
C     SIMULATES THE WATER GAS SHIFT REACTOR
C
C     EN(16)=FRACTION OF CO FORMED BY CH4
C     EN(17)=BASE TEMPERATURE = 260°C
C     EN(18)=HEAT OF REACTION FOR WATER GAS SHIFT REACTION   BASE TEMP.
C     EN(19)=HEAT OF REACTION FOR REFORMER REACTION  BASE TEMP.
C     EN(20)=ASSUMED OUTLET TEMPERATURE FOR SHIFT REACTORS :
C                                 FOR HIGH TEMPERATURE = 342°C
C                                 FOR LOW TEMPERATURE = 254°C
C     EN(21)=APPROACH TEMPERATURE FOR SHIFT REACTORS :
C                                 FOR HIGH TEMPERATURE =-105.79366
C                                 FOR LOW TEMPERATURE = -91.60266
C     EN(22)=PRESSURE DROP FACTOR :
C     EN(23)=DUE CORRECTION FACTOR FOR OUTLET TEMPERATURE
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSP,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSO(10,16),PPMX(10)
      COMMON/P/ICI,ICO,FACP
      COMMON/NAEJ/EQNU,ICC
C
C     MATERIAL BALANCE
C
      DO 90 J=1,NOCOMP
      SO(1,J+5)=0.
   90 CONTINUE
      SO(1,4)=EN(20)
      F=EN(16)
      CON=SI(1,8)+F*SI(1,9)
      SI(1,6)=SI(1,6)-F*SI(1,9)
      SI(1,10)=SI(1,10)+3.*F*SI(1,9)
      AA=SI(1,7)/CON
      BB=SI(1,10)/CON
      CC=SI(1,6)/CON
      DO 600 K=1,50
      SO(1,3)=0.
      TEQ=SO(1,4)-EN(21)
C
C     CALCULATE K3
C
C     T IN C DEGREE
C
C***  NOTE  ***
C
C     CALL TO EQK MUST BE DONE WITH CARE. CHECK THE PARAMETER N AND
C     THE OUTPUT
C
      CALL EQK(2,TEQ,AK1,AK2,AK3,AK4)
      AK3 = AK2
C
C     Y=FRACTION OF CO REACTS
C
      A = AK3 -1.
      B = -AA-BB-AK3*(CC+1.)
      C=AK3*CC-AA*BB
      D=B*B-4.*A*C
      Y1=(-B+SQRT(D))/(2.*A)
      Y2=(-B-SQRT(D))/(2.*A)
      IF(Y1.GE.0..AND.Y1.LE.1.)GO TO 10
      IF(Y2.GE.0..AND.Y2.LE.1.)GO TO 20
      WRITE(6,100)
  100 FORMAT(1X,'DO NOT HAVE ROOT BETWEEN 0. AND 1. IN SUBROUTINE REACO2
     **')
      DO 92 L=3,JJ
   92 SO(1,L)=SI(1,L)
      GO TO 600
   10 Y=Y1
      GO TO 30
   20 Y=Y2
C
C     CALCULATE FLOW COMPOUNDS
```

```
   7[  SO(1,6)=SI(1,6)-Y*CON
       SO(1,7)=SI(1,7)+Y*CON
       SO(1,8)=CON*(1.-Y)
       SO(1,9)=SI(1,9)*(1.-F)
       SO(1,10)=SI(1,10)+Y*CON
       DO 499 J=11,JJ
   499 SO(1,J)=SI(1,J)
       DO 500 J=1,NOCOMP
       SO(1,3)=SO(1,3)+SO(1,J+5)
   500 CONTINUE
C
C      HEAT BALANCE
C
C      CALCULATE PHYSICAL PROPERTS
C
       IF(K-1)700,700,800
   700 NOPP=1
       X=(SI(1,4)+EN(17))/2.
       DO 300 J=1,NOCOMP
       XMOLE(J)=SI(1,J+5)
   300 CONTINUE
       CALL PROMIX(1)
       SPFI=PPMX(NOPP)
   800 NOPP=1
       X=(SO(1,4)+EN(17))/2.
       DO 400 J=1,NOCOMP
       XMOLE(J)=SO(1,J+5)
   400 CONTINUE
       CALL PROMIX(1)
       SPFO=PPMX(NOPP)
C      SUM OF ALL INDIVIDUAL PRODUCTS (W*CP) IS IN PPMX(NOPP)
C
C      CALCULATE OUTLET TEMPERATUPE
C
       TC=(SPFI*(SI(1,4)-EN(17))-EN(18)*Y*CON-EN(19)*EN(16)*SI(1,9))/
      *SPFO+EN(17)
       IF(ABS((SO(1,4)-TC)/TC)-0.001)60,60,70
    70 SO(1,4)=TC
   600 CONTINUE
       WRITE(6,91)
    91 FORMAT(1X,*SUBROUTINE REACO 2 DID NOT CONVERGE IN 50 LOOPS*)
       GO TO 660
    60 SO(1,4)=TC
       SO(1,4)=SO(1,4)-EN(23)
       EN(20)=SO(1,4)
C
C      CALCULATE PRESSURE DROP
C
       SO(1,5)=EN(22)
       ICO=1
       EQNU=NE
       CALL CALFAC(FACP)
       IISW=2
       ICT=1
       ICO=1
       FACP=EN(22)
       CALL PDROP1(IISW)
       CALL DISKIO(2,NE)
   660 RETURN
       END
```


REACO3
Set up the parameters used in the model, assume outlet temperature T, the amounts of CO and CO_2 (X and Y, respectively) that react.
Determine equilibrium constants at temperature T for CO $3H_2$ reaction (XK1) and for CO_2 $4H_2$ reaction (XK2)
Iterate (maximum 50 loops) on X and Y (XC and YC, respectively) testing each time if XC = Y and YC = Y by calling CØNVE which takes the appropriate action to modify the assumed X and Y according to the values of XC and YC.
Set outlet component flows
Carry an energy balance around the reactor to determine outlet temperature TC.
Call CØNVE to compare T and TC and to take appropriate correction on the assumed outlet temperature T
Iterate until T = TC, 50 loops maximum
Reset EN vector
RETURN END

```
      SUBROUTINE REAC03
C
C       SIMULATES THE METHANATOR REACTOR
C
C*****EN(16)=H1 REACTION HEAT AT 260 DEGREES OF CO CONVERSION
C*****EN(17)=H2 REACTION HEAT AT 260 DEGREES OF CO2 CONVERSION
C*****EN(18)=TR TEMPERATURE REFERENCE FOR HEAT BALANCE
C*****EN(19)=FACP FACTOR FOR PRESSURE DROP CALCULATION
C*****EN(20)=INITIAL GUESS FOR CO  THAT REACT
C*****EN(21)=INICIAL GUESS FOR OUTELET TEMPERATURE
C*****EN(22)=INITIAL GUESS FOR CO2 THAT REAC
C   ** EN(23)= CORRECTION TO THE CALCULATED OUTLET TEMPERATURE
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSP,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      COMMON/P/ICI,ICO,FACP
C     COMMON/NAEJ/EQNU,ICC
      DIMENSION XAUX(2)
      ICX=1
      ICY=1
      ICT=1
      A=SI(1,8)
      B=SI(1,7)
      C=SI(1,10)
      D=SI(1,9)
      E=SI(1,6)
      XNI=SI(1,3)
      P=SI(1,5)
      NOPP=1
      X=SI(1,4)
      DO 100 J=1,NOCOMP
  100 XMOLE(J)=SI(1,J+5)
      CALL PROMIX(1)
      WCPI=PPMX(NOPP)
      H1=EN(16)
      H2=EN(17)
      TR=EN(18)
      FACP=EN(19)
      X=A*EN(20)
      T=EN(21)
      Y=B*EN(22)
      DO 750 IJK=1,50
      CALL EQK(3,T,AK1,AK2,AK3,AK4)
      XK1=AK3
      CALL EQK(4,T,AK1,AK2,AK3,AK4)
      XK2=AK4
      XK=XK2/XK1
      XK1=XK1*P*P
      DO 751 J9=1,50
      DO 700 JR=1,50
    4 AJ=A-X
      BJ=B-Y
      CJ=C-3.*X-4.*Y
      DJ=D+X+Y
      EJ=E+X+2.*Y
      FJ=XNI-2.*X-2.*Y
      XC=A-(DJ*EJ*FJ*FJ)/(CJ*CJ*CJ*XK1)
      YC=B-(AJ*EJ)/(CJ*XK)
      CALL CONVE (Y,YC,ICY,XAUX)
      IF(ICY.EQ.1) GO TO 5
  700 CONTINUE
      PRINT 997
  997 FORMAT(1X,*CONVERGENCE WAS NOT ACHIEVED ON YC IN REAC03*)
      GO TO 4
    5 CALL CONVE (X,XC,ICX,XAUX)
      IF(ICX.EQ.1) GO TO 1
  751 CONTINUE
      PRINT 998
  998 FORMAT(1X,*CONVERGENCE WAS NOT ACHIEVED ON XC IN REAC03*)
    1 SO(1,6)=E+X+2.*Y
      SO(1,7)=B-Y
      SO(1,8)=A-X
      SO(1,9)=D+X+Y
      SO(1,10)=C-3.*X-4.*Y
```

```
      DO 11 J1=11,16
      SO(1,J1)=SI(1,J1)
   11 CONTINUE
      SO(1,3)=0.
      DO 12 J2=6,16
      SO(1,3)=SO(1,3)+SO(1,J2)
   12 CONTINUE
      NOPP=1
      DO 13 J3=1,NOCOMP
      XMOLE(J3)=SO(1,J3+5)
   13 CONTINUE
      AAA=X
      X=T
      CALL PROMIX(1)
      WCPO=PPMX(NOPP)
      X=AAA
      TC=(WCPI*(SI(1,4)-TR)+H1*X+H2*Y)/WCPO+TP
      CALL CONVE (T,TC,ICT,XAUX)
      IF(ICT.EQ.1) GO TO 15
  750 CONTINUE
      PRINT 999
  999 FORMAT(1X,*CONVERGENCE WAS NOT ACHIEVED ON TC IN REAC03*)
   15 SO(1,4)=T
   15 SO(1,4)=T-EN(23)

C     PRESSURE DROP CALCULATION

      SO(1,5)=EN(19)
      EQNU=NE
      ICC=1
      CALL CALFAC(FACP)
      IISW=2
      ICI=1
      ICO=1
      FACP=EN(19)
      CALL PDROP1(IISW)
      EN(20)=X/A
      EN(21)=T
      EN(22)=Y/B
      CALL DISKIO(2,NE)
      RETURN
      END
```

E.2 Heat Exchanger Models

EXCHO1

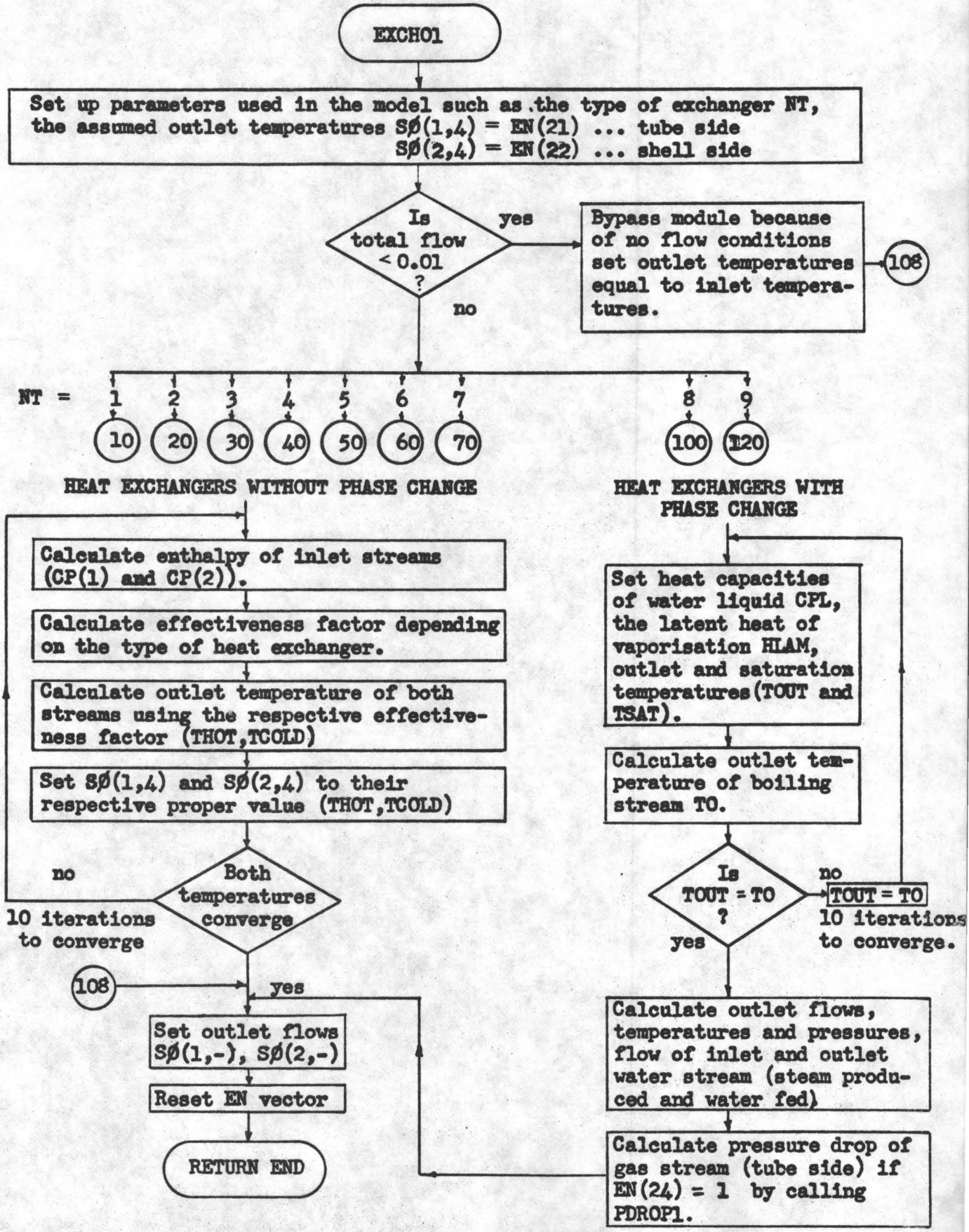
EXCHO2

EXCHO3

EXCHO4 and HEX

- Formulation to Determine Exit Temperatures for the Cocurrent-Countercurrent Heat Exchanger for EXCHO4

EXCHO5


EXCHO1
Set up parameters used in the model such as the type of exchanger NT, the assumed outlet temperatures SØ(1,4) = EN(21) ... tube side
SØ(2,4) = EN(22) ... shell side
Is total flow < 0.01 ?
yes
Bypass module because of no flow conditions set outlet temperatures equal to inlet temperatures.
108
no
NT =
1
2
3
4
5
6
7
8
9
10
20
30
40
50
60
70
100
120
HEAT EXCHANGERS WITHOUT PHASE CHANGE
HEAT EXCHANGERS WITH PHASE CHANGE
Calculate enthalpy of inlet streams (CP(1) and CP(2)).
Calculate effectiveness factor depending on the type of heat exchanger.
Calculate outlet temperature of both streams using the respective effectiveness factor (THOT,TCOLD)
Set SØ(1,4) and SØ(2,4) to their respective proper value (THOT,TCOLD)
Both temperatures converge
no
10 iterations to converge
108
yes
Set outlet flows SØ(1,-), SØ(2,-)
Reset EN vector
RETURN END
Set heat capacities of water liquid CPL, the latent heat of vaporisation HLAM, outlet and saturation temperatures (TOUT and TSAT).
Calculate outlet temperature of boiling stream TO.
Is TOUT = TO ?
no
TOUT = TO
10 iterations to converge.
yes
Calculate outlet flows, temperatures and pressures, flow of inlet and outlet water stream (steam produced and water fed)
Calculate pressure drop of gas stream (tube side) if EN(24) = 1 by calling PDROP1.

```
      SUBROUTINE EXCHG1
      THE PROGRAM CALCULATES THE OUTLET TEMPERATURE
      DIRECTLY FOR 9 TYPES OF HEAT EXCHANGERS USING
      EFFECTIVENESS FACTORS. IT CALCULATES ALSO THE INLET
      FLOW RATE TO BE SET INTO A BOILER (LAST CASE).

      SI(1,1) IS THE INLET TO TUBE SIDE.
      SI(2,1) IS THE INLET TO SHELL SIDE.
      CMAX IS THE LARGER OF THE TWO THERMAL CAPACITANCES.
      CMIN IS THE SMALLER OF THE TWO CAPACITANCES.

      EN VECTOR
      16 UA
      17 NT(TYPE OF EXCHANGER)

      EXCHANGES WITHOUT PHASIS CHANGE

        COUNTERCURRENT                                    1
        PARALLEL                                          2
        CROSSFLOW, BOTH STREAMS UN MIXED                  3
        CROSSFLOW, BOTH STREAMS MIXED                     4
        CROSSFLOW, STREAM CMIN UNMIXED                    5
        CROSSFLOW,STREAM CMAX UNMIXED                     6
        PARALLELCOUNTER FLOW WITH N SHELL PASSES          7
        ANY NUMBER OF TUBE PASSES INCLUDING 1
      EXCHANGERS WITH PHASIS CHANGE

        BOILER WITH A INLET FLOW RATE TO BE DETERMINED    8
        CONDENSER WITH A INLET FLOW RATE TO BE DETERMINED 9

      18 NUMBER OF SHELL PASSES IF COUNTERCOURRENT FLOW
      19 LOCATION OF HEAT CAPACITY COEF. FOR FLUID IN TUBE SIDE
      20 LOCATION OF HEAT CAPACITY COEF. FOR FLUID IN SHELL SIDE
         (FOR EVAPORATING OR CONDENSING SYSTEMS THE LOCATION OF
         HEAT CAPACITY OF HEATING  FLUID ON TUBE SIDE
      21  FIRST ASSUMPTION OF OUTLET TEMPERATURE TUBE SIDE
      22  FIRST ASSUMPTION OF OUTLET TEMPERATURE SHELL SIDE
      23  FRACTION BLOWDOWN
       24 LLSW = SWITCH  = 0 NO PRESSURE DROP CALCULATION
                           1 PRESSURE DROP CALCULATION
       25 FACP (IF EN(24)=1) = FACTOR FOR PRESSURE DROP CALCULATION

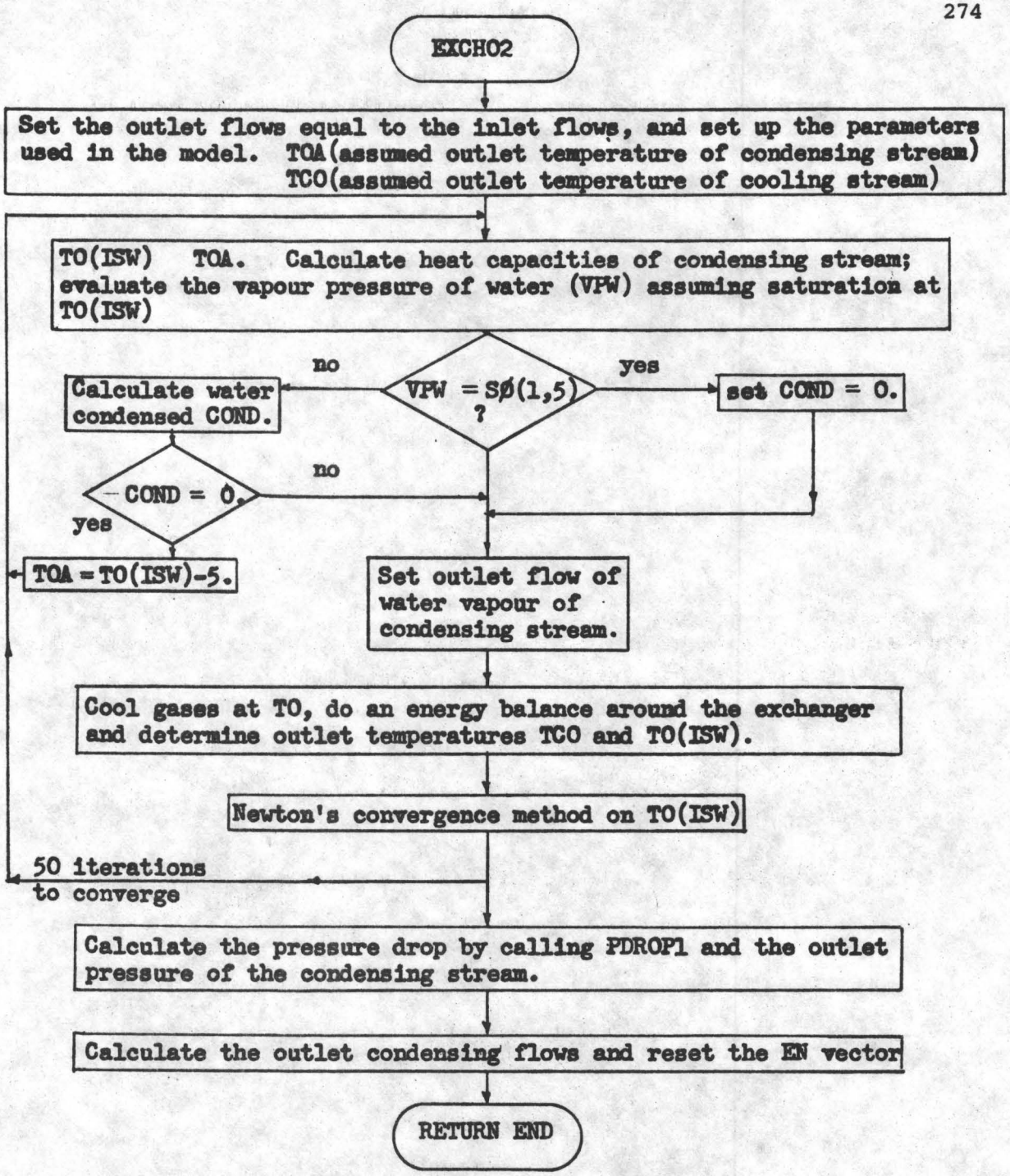
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSF,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      COMMON/P/ICI,ICO,FACP
      COMMON/NAEJ/EQNU,ICC
      DIMENSION TAVE(2), CP(2), F(2)
      HLAM(T)=7081.
      TS(P)=262.7
      SO(1,4)=EN(21)
      UA=EN(16)
      NT=EN(17)+0.01
      SO(2,4)=EN(22)
      IF(SI(1,3).LT.0.01.OR.SI(2,3).LT.0.01) GO TO 103
      GO TO (6,6,6,6,6,6,100,120)NT
    6 DO 99 II=1,10
      DO 2 KK=1,2
      TAVE(KK)=(SI(KK,4)+SO(KK,4))/2.
      X=TAVE(KK)
      NOPP=EN(KK+18)
      DO 1 J=1,NOCOMP
    1 XMOLE(J)=SI(KK,J+5)
      CALL PROMIX(1)
    2 CP(KK)=PPMX(NOPP)
      IF(SI(1,4).LT.SI(2,4))GO TO 3
      CH=CP(1)
      CC=CP(2)
      THIN=SI(1,4)
      TCIN=SI(2,4)
      ITHO=1
      GO TO 4
    3 CH=CP(2)
```

```
      CC=CP(1)
      THIN=SI(2,4)
      TCIN=SI(1,4)
      ITHOT=2
  101 CMIN=CC
      CMAX=CH
      IF((CH-CC).GT.0.)GO TO 5
      CMIN=CH
      CMAX=CC
    5 CONTINUE
  110 A=CMIN/CMAX
      B=UA/CMIN
  112 GO TO(10,20,30,40,50,60,70,100,120)NT
   10 X=(1.0-A)
      Y=EXP(-B*X)
  113 Z=(1.-Y)/(1.-A*Y)
      GO TO 90
   20 Z=(1.-EXP((-B)*(1.+A)))/(1+A)
      GO TO 90
   30 X=(1./B)**.22
      Y=(UA*CMIN*X)/(CMIN*CMAX)
      Z=1.-EXP((EXP(-Y)-1.)/(A*X))
      GO TO 90
   40 Y=UA/CMAX
      Z=B/((B/(1.-EXP(-B)))+(Y/(1.-EXP(-Y))) -1.)
      GO TO 90
   50 Z=(1.-EXP(-(1.-EXP(-B))*A))/A
      GO TO 90
   60 Z=1.-EXP(-(1./A)*(1.-EXP(-UA/CMAX)))
      GO TO 90
   70 Y=SQRT(1.+B*B)
      E=2./(1.+A+Y*((1.+EXP(-B*Y))/(1.-EXP(-B*Y))))
      R=((1.-E*A)/(1.-E))**EN(18)
      Z=(R-1)/(R-A)
   90 THOT=THIN-(Z*CMIN*(THIN-TCIN)/CH)
      TCOLD=TCIN+(Z*CMIN*(THIN-TCIN)/CC)
      SO(1,4)=TCOLD
      SO(2,4)=THOT
      IF(ITHOT.EQ.2)GO TO 92
  114 SO(1,4)=THOT
      SO(2,4)=TCOLD
   92 AVE1=(SO(1,4)+SI(1,4))/2.
      AVE2=(SO(2,4)+SI(2,4))/2.
      YY=ABS(AVE2-TAVE(2))
      ZZ=ABS(AVE1-TAVE(1))
  120 IF(YY.LT.0.5.AND.ZZ.LT.0.5)GO TO 130
   99 CONTINUE
      WRITE(6,998)NE
  998 FORMAT(//,5X,*TAVE DID NOT CONVERGE IN EXCH01,EQUIPMENT NUMBER*,
     1 I5 )
      GO TO 108
  103 SO(1,4)=SI(1,4)
  121 SO(2,4)=SI(2,4)
      WRITE(6,105)NE
  105 FORMAT(//,5X,*BY PASS MODULE*,I4,*BECUASE OF NO FLOW CONDITIONS*)
  108 CONTINUE
      DO 119 K=1,2
      SO(K,3)=SI(K,3)
  130 DO 109 J=5,JJ
      SO(K,J)=SI(K,J)
  109 CONTINUE
  119 CONTINUE
      EN(21)=SO(1,4)
      EN(22)=SO(2,4)
      GO TO 140
  997 FORMAT(//,5X,*CONVERGENCE PROMOTION NOT COMPLETED ON TO IN
     2 EQUIPMENT*, I4)
  130 P=SI(2,3)
      TSAT=TS(P)
  140 ALAMP=HLAM(TSAT)
C
C     THE AVERAGE HEAT CAPACITY OVER THE REGION OF INTEREST IS
C     0.873 KCAL/KGM C  (= 15.71 KCAL/KGMOLE C)
C
      CPL=15.710
      TOUT=EN(21)
```

```
      DO 11   I=1,10
      X=(TOUT+SI(1,4))/2.
      NOPP=EN(19)+0.001
      DO 111 J=1,NOCOMP
  101 XMOLE(J)=SI(1,J+5)
      CALL PROMIX(1)
      CPG=PPMX(NOPP)
      Z=1.-EXP(-UA/CPG)
      TO=SI(1,4)-Z*(SI(1,4)-TSAT)
      IF(ABS((TO-TOUT)/TO).LE.0.001)GO TO 112
  110 TOUT=TO
      WRITE(6,997)NE
  112 QTOT=CPG*(SI(1,4)-TO)
      DO 113 I=7,JJ
      SO(1,I)=SI(1,I)
  113 SO(2,I)=0.
      SO(2,6)=QTOT/(ALAMB+(1.+EN(23))*CPL*(TSAT-SI(2,4)))
      SI(2,14)=SO(2,6)*(1.+EN(23))
      SI(2,3)=SI(2,14)
      SO(2,3)=SO(2,6)
      SO(1,6)=SI(1,6)
      SO(1,4)=TO
      SO(2,4)=TSAT
      SO(1,3)=SI(1,3)
      SO(1,5)=EN(24)
      ICC=1
      EQNU=NE
      CALL CALFAC(FACP)
      LLSW=EN(24)+0.001
      IF(LLSW.LT.1) GO TO 114

C     CALCULATE PRESSURE DROP

      IISW=2
      ICI=1
      ICO=1
      FACP=EN(25)
      CALL PDROP1(IISW)
      GO TO 115
  114 CONTINUE
      SO(1,5)=SI(1,5)
  115 CONTINUE
      SO(2,5)=SI(2,5)
      EN(21)=TO
      GO TO 140
  120 TSAT=SI(2,4)
      ALAMB=HLAM(TSAT)
      TOUT=EN(21)
      DO 130 IJK=1,20
      X=(TOUT+SI(1,4))/2.
      NOPP=EN(19)
      DO 121 J=1,NOCOMP
  121 XMOLE(J)=SI(1,J+5)
      CALL PROMIX(1)
      CPG=PPMX(NOPP)
      Z=1.-EXP(-UA/CPG)
      TO=SI(1,4)-Z*(SI(1,4)-SI(2,4))
      IF(ABS((TO-TOUT)/TO).LE.0.001)GO TO 132
  130 TOUT=TO
      WRITE(6,997)NE
  132 QTOT=CPG*(TO-SI(1,4))
      SI(2,6)=SO(2,13)
      SO(2,6)=0.
      DO 133 I=7,JJ
      SO(1,I)=SI(1,I)
  133 SO(2,I)=SI(2,I)
      SO(2,13)=QTOT/ALAMB
      SI(2,13)=0.
      EN(21)=TO
  140 CALL DISKIO(2,NE)
      RETURN
      END
```


EXCHO2
Set the outlet flows equal to the inlet flows, and set up the parameters used in the model. TOA(assumed outlet temperature of condensing stream) TCO(assumed outlet temperature of cooling stream)
TO(ISW) TOA. Calculate heat capacities of condensing stream; evaluate the vapour pressure of water (VPW) assuming saturation at TO(ISW)
VPW = SØ(1,5) ?
no
yes
Calculate water condensed COND.
set COND = 0.
COND = 0.
no
yes
TOA = TO(ISW)-5.
Set outlet flow of water vapour of condensing stream.
Cool gases at TO, do an energy balance around the exchanger and determine outlet temperatures TCO and TO(ISW).
Newton's convergence method on TO(ISW)
50 iterations to converge
Calculate the pressure drop by calling PDROP1 and the outlet pressure of the condensing stream.
Calculate the outlet condensing flows and reset the EN vector
RETURN END

```
      SUBROUTINE EXCH22
C
C     CALCULATES THE HEAT EXCHANGED IN A COOLER-CONDENSER GIVEN THE
C     FRACTION OF WATER CONDENSED AS SPECIFIED IN EN VECTOR
C     USES EFFECTIVENESS FACTOR TO CALCULATE OUTLET TEMPERATURES
C     IF EN(17)=1 THEN VAPOR-LADEN STREAM IS PARTIALLY CONDENSED BY NON-
C     CONDENSING GAS OR LIQUID.
C     IF EN(17)=2 THEN VAPOR-LADEN STREAM IS PARTIALLY CONDENSED BY A
C     BOILING LIQUID.
C
C     EN VECTOR
C     EN(16)=UA
C     EN(17)=EXCHANGER TYPE,=1 IF EXCHANGER COOLED BY NON-CONDENSING
C            FLUID.=2 IF COOLED BY A BOILING LIQUID
C     EN(18)=NUMBER OF TUBE PASSES,ONLY NEEDED IF EN(17)=1
C     EN(19)=NUMBER OF SHELL PASSES,ONLY NEEDED IF EN(17)=1
C     EN(20)=ASSUMED TOUT OF COOLING STREAM  (=SO(2,4))
C     EN(21)=NOPP FOR COOLING FLUID=1 FOR GASES,=2 FOR LIQUID(REFORMER
C            PLANT)
C     EN(22)=FRACTION OF VAPOR IN CONDESING STREAM WHICH IS CONDENSED.
C     EN(23)=CONSTANT IN PRESSURE DROP EQUATION.
C     EN(24)=ASSUMED TOUT OF CONDENSING STREAM  (=SO(1,4))
C     EN(25) = FACP = FACTOR FOR PRESSURE DROP CALCULATION
C
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(150),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,KC,NCALC,NSR,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      COMMON/P/ICI,ICO,FACP
C     COMMON/NAEJ/EQNU,ICC
      DIMENSION DF(2),TO(2)
      DIMENSION DIFT(2),U(2)
C     LATENL HEAT OF VAPORIZATION CORRELATED VIA
C    HLAM(T)/HLAM(T1)=((TCRIT-T)/(TCRIT-T1))**N--TCRIT WATER=374.7 C
      HLAM(T)=EXP(0.33473*ALOG(374.2-T)+7.30109)
      DO 11 I=1,2
      DO 10 J=3,JJ
      SO(I,J)=SI(I,J)
   10 CONTINUE
   11 CONTINUE
C        SET UP CONSTANTS AND PARAMETERS
      UA=EN(16)
      ITYPE=EN(17)+0.001
      NTP=EN(18)+0.001
      NSP=EN(19)+0.001
      AK1=EN(23)
      TCO=EN(20)
      TOA=EN(24)
      DELT=SI(1,4)-SI(2,4)
      MSW=1
      NSW=2
      DO 2111 KMM=1,20
      NCOUNT=1
      ISW=1
      ICOUNT=0
      KSW=2
      JSW=2
      DO 555 MLM=1,50
      TO(ISW)=TOA
      NOPP=1
      TAVEE=(SI(1,4)+TO(ISW))/2.
      X=TAVEE
      DO 620 J=1,NOCOMP
  620 XMOLE(J)=SI(1,J+5)
      CALL PPOMIX(1)
      IF(EN(5).GE.1.) WRITE(6,797) PPMX(1)
  797 FORMAT(1X,'PPMX(1)=',E16.3)
C       EVALUATE VAPOUR PRESSURE AT TO AND USE SATURATION ASSUMPTION
C     TO CALCULATE MOLES OF WATER PER MOLE INERT AT EXIT OF EXCHANGER
C        PROTECT AGAINST TOO HIGH A TEMPERATURE
      IF(TOA.GE.374.2) TOA=300.
      Z=-.9668*+1668.21/(228.0+TOA)
      VPW=(10.**Z)/760.
      IF(EN(5).LT.1.) GO TO 1212
      IF((MLM/9).NE.NCOUNT) GO TO 1212
```

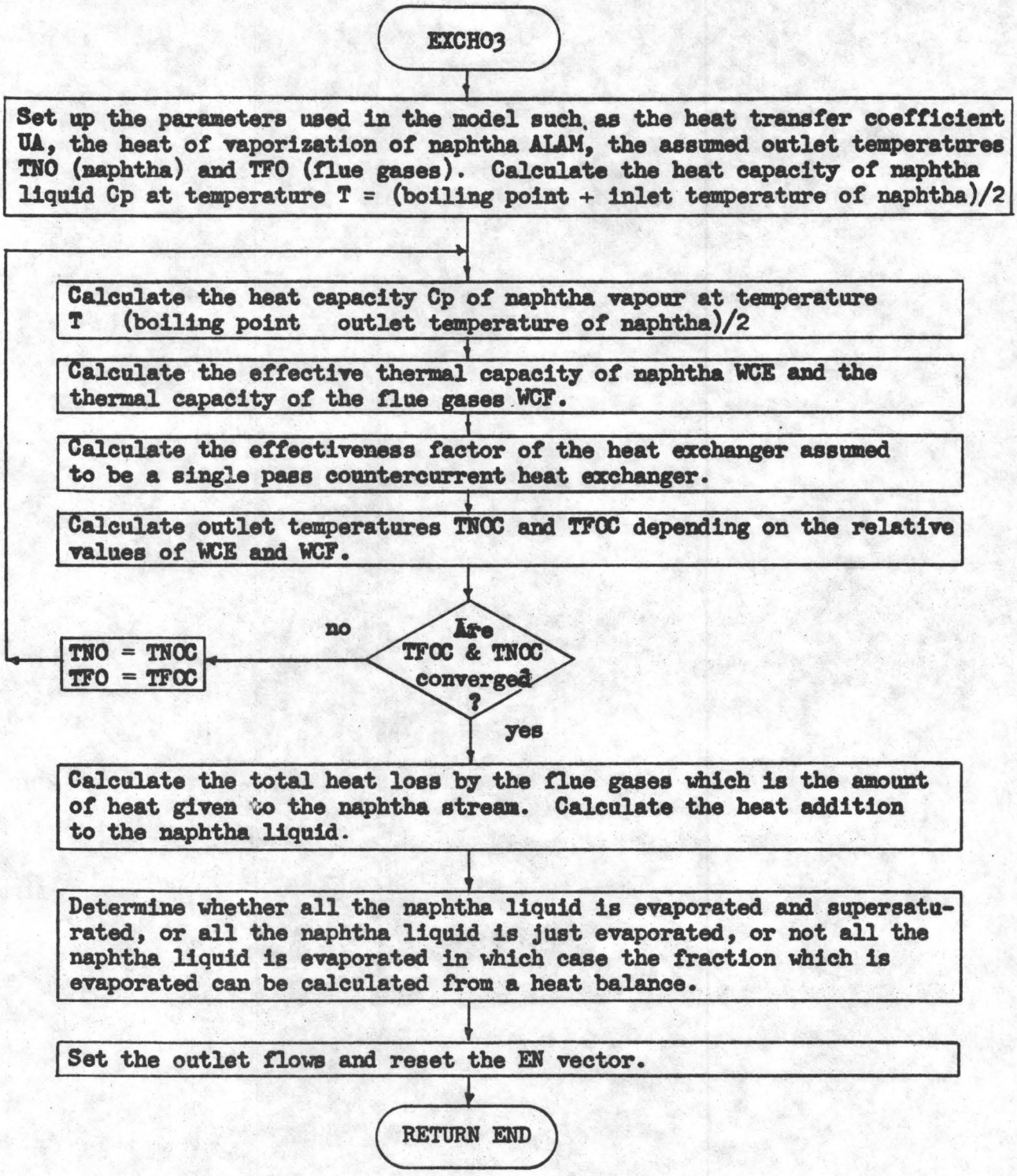
```
      WRITE(6,*) VPW
      NCOUNT=NCOUNT+1
 1212 CONTINUE
C       TEST IF OUTLET VAPOUR PRESSURE GREATER THAN SYSTEM PRESSURE
      DEP=VPW-SI(1,5)
      IF(DEP.LT.0.) GO TO 554
      WRITE(6,791) MLM
  791 FORMAT(1X,*OUTLET TEMPERATURE ASSUMED TOO HIGH,VAPOUR PRESSURE EXC
     1EEDS TOTAL PRESSURE OF SYSTEM,ON ITERATION NO.,*,I3,/)
      WCE=PPMX(1)
      Q=WCE*(SI(1,4)-TO(ISW))
      ICOUNT=ICOUNT+1
      F=0.
      COND=0.
      GO TO 144
  554 XMVO=(SI(1,3)-SI(1,6))*VPW/(-DEP)
  552 COND=SI(1,6)-XMVO
      IF(COND.LT.0.) GO TO 564
      SO(1,6)=SI(1,6)-COND
      F=COND/SI(1,6)
      QV=COND*HLAM(TO(ISW))
      IF(EN(5).GE.1.) WRITE(6,749) COND,XMVO,QV,TOA
  749 FORMAT(1X,*COND=*,F15.5,* XMVO=*,F15.5,* QV=*,E15.4,* TOA=*,F8.3)
C       COOL ALL GASES TO TO THEN CONDENSE WATER AT TO
      Q=QV+PPMX(NOPP)*(SI(1,4)-TO(ISW))
C         CALCULATE OUTLET TEMPERATURES WITH THIS MUCH CONDENSATION
C           CALCULATING WCE FOR CONDENSING STREAM
      WCE=QV/(SI(1,4)-TO(ISW))+PPMX(1)
      IF(EN(5).GE.1.) WRITE(6,796) WCE
  796 FORMAT(1X,*WCE=*,E15.3)
C       ITERATE ON TCO
  144 CONTINUE
      IF(ITYPE.EQ.2) GO TO 651
C     PROCESS GAS COOLED BY NON-CONDENSING FLUID
      DO 300 IJK=1,20
C     ALLOW ONLY 20 ITERATIONS ON CONVERGENCE OF TAVE.
      TAVE=(SI(2,4)+TCO)/2.
      X=TAVE
C     OBTAIN SUM OF WCIS-SET PHYS.PROP.
      DO 150 J=1,NOCOMP
      XMOLE(J)=SI(2,J+5)
  150 CONTINUE
      NOPP=EN(21)+0.001
C     WRITE(6,799) NOPP
C 799 FORMAT(10X,I5)
      CALL PROMIX(1)
      WC=PPMX(NOPP)
C     WRITE(6,798) WC
      IF(EN(5).GE.1.) WRITE(6,798) WC
C 798 FORMAT(1X,*WC=*,E15.3)
  798 FORMAT(1X,*WC=*,E15.3)
      TCOC=Q/WC+SI(2,4)
      IF(ABS((TCO-TCOC)/TCOC).LE.0.001) GO TO 750
      TCO=TCOC
  300 CONTINUE
  750 CONTINUE
C       CORRECTION FOR THE NUMBER OF TURE AND SHELL PASSES NOT INCLUDED
      IF(WCE-WC) 650,650,660
C     CONDENSING STREAM SMALLER WC
  650 R=WCE/WC
  651 CONTINUE
      IF(ITYPE.EQ.2) R=0.
      TU=UA/WCE
      IIS=1
      GO TO 670
C       COOLING STREAM SMALLER WC
  660 R=WC/WCE
      TU=UA/WC
      IIS=2
  670 CONTINUE
      ETU=EXP(-TU*(1.-R))
      E=(1.-ETU)/(1.-R*ETU)
      IF(IIS-1) 680,680,690
  680 TOC=SI(1,4)-E*DELT
      IF(ITYPE.EQ.2) TCOC=SI(2,4)
```

```
      IF(ITYPE.EQ.2)  GO TO 700
      TCOC=SI(2,4)+R*(SI(1,4)-TOC)
      GO TO 690
  690 TCOC=SI(2,4)+E*DELT
      TOC=SI(1,4)-R*(TCOC-SI(2,4))
  700 CONTINUE
      DF(ISW)=TO(ISW)-TOC
  748 FORMAT(1X,*DF(ISW)=*,E15.5,*  TCOC=*,F15.3)
      IF(EN(5).GE.1.) WRITE(6,748) DF(ISW),TCOC
      IF(ABS(DF(ISW)/TOC).LE.0.001) GO TO 1000
C       STRADDLE THE ROOT FIRST
      IF(ISW.EQ.1) GO TO 485
      IF(JSW.EQ.1) GO TO 485
      IF(DF(1).LT.0..AND.DF(2).GT.0.) GO TO 487
      IF(DF(1).GT.0..AND.DF(2).LT.0.) GO TO 487
      ISW=1
      KSW=0
      DF(1)=DF(2)
      TO(1)=TO(2)
  487 CONTINUE
      IF(ISW.EQ.1) GO TO 485
      IF(KSW.EQ.1) GO TO 487
      SLOPE=(DF(1)-DF(2))/(TO(1)-TO(2))
      JSW=1
      KSW=1
  483 TOA=TO(2)-DF(2)/SLOPE
      GO TO 555
  485 CONTINUE
      IF(DF(1).LE.0.) TOA=TO(1)+2.
      IF(DF(1).GT.0.) TOA=TO(1)-2.
      ISW=2
      GO TO 555
C       TOO MUCH CONDENSATION TOA INCREASED
  564 TOA=TO(ISW)-5.
      IF(EN(5).GE.1.) WRITE(6,766) TO(ISW),MLM
  766 FORMAT(1X,*CONDENSING GAS STREAM OUTLET TEMPERUATURE IS ASSUMED TO
     1O HIGH.ITS VALUE IS,*,F10.3,* ON ITERATION NO.,*,I3,/)
  555 CONTINUE
      WRITE(6,996)NE
  996 FORMAT(1X,*THE OUTLET TEMPERATURE FOR THE REBOILER,EQUIPMENT NO.*,
     *I2,*IN EXCH02,DID NOT CONVERGE.*)
  550 CONTINUE
 1000 SO(1,4)=TO(ISW)
      SO(1,14)=COND
      SO(2,4)=TCOC
      WRITE(6,*) UA,KMM,SO(1,4),COND,SO(2,4)
      DIFT(MSW)=TO(ISW)-116.
      DIFT(MSW)=TO(ISW)-103.
      DIFT(MSW)=TO(ISW)-116.
      U(MSW)=UA
      IF(ABS(DIFT(MSW)/TO(ISW)).LE.0.001) GO TO 2112
      IF(ABS(DIFT(MSW)/TCOC).LT.0.001) GO TO 2112
      IF(ABS(DIFT(MSW)/TO(ISW)).LE.0.001) GO TO 2112
      IF(MSW.EQ.1) GO TO 2152
      IF(MSW.EQ.2) GO TO 2151
      SLOP=(DIFT(1)-DIFT(2))/(U(1)-U(2))
      MSW=1
2151  UA=U(2)-DIFT(2)/SLOP
      GO TO 2111
2152  IF(DIFT(MSW).GT.0.) UA=UA-0.05*UA
      IF(DIFT(MSW).LT.0.) UA=UA+0.05*UA
      MSW=2
2111  CONTINUE
2112  CONTINUE
      EN(24)=SO(1,4)
      AK1=EN(23)

C     ** NOTE ** TO CALCULATE THE PRESSURE DROP, TAKE OFF THE WATER
C               CONDENSED ** OTHERWISE PDROP1 WILL CALCULATE THE WRONG
C               AVERAGE MOLECULAR WEIGHT OF THE GAS LEAVING EXCH02 **

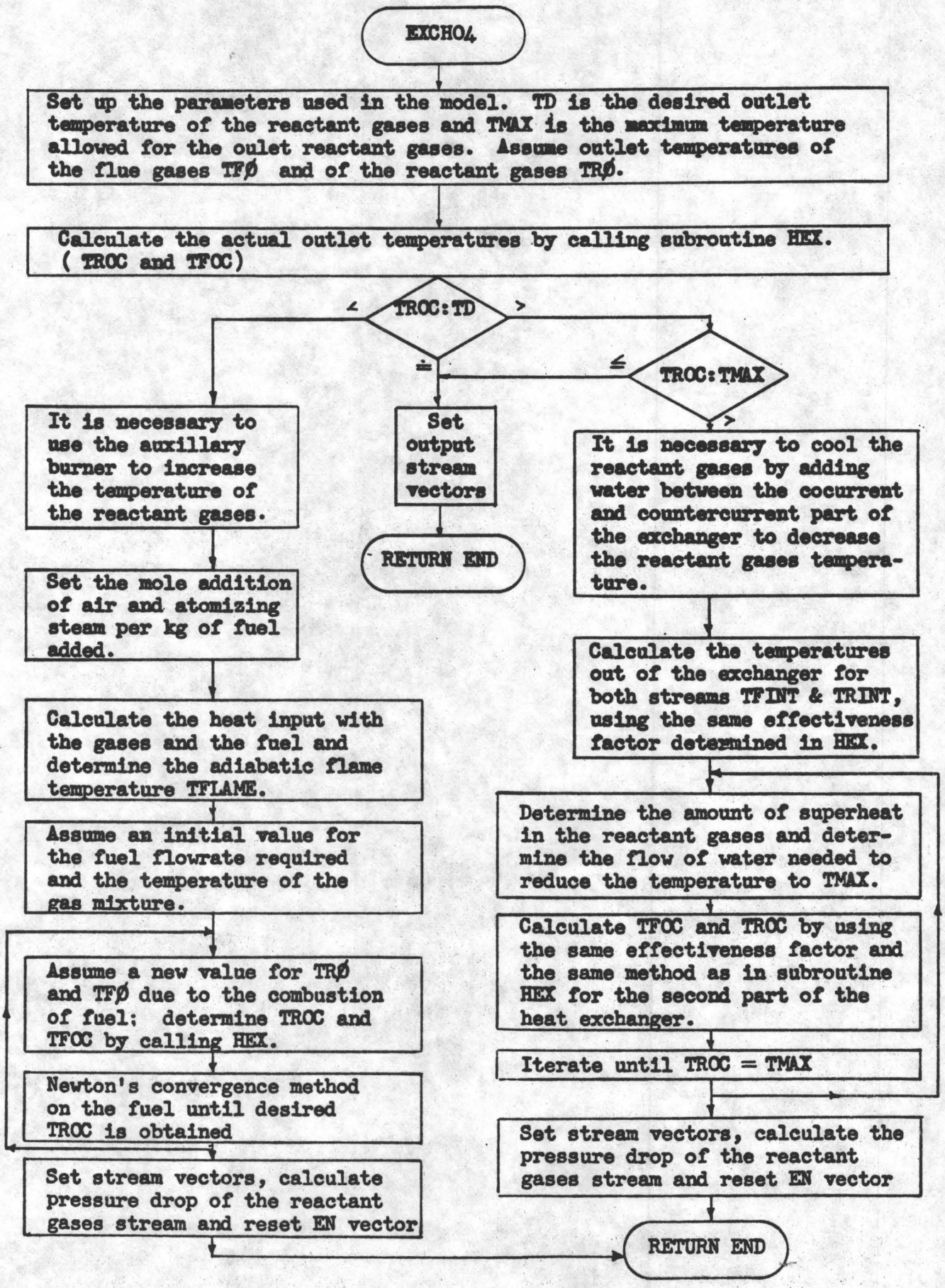
      SO(1,3)=SO(1,3)-COND
      SO(1,14)=0.
      SO(1,5)=EN(25)
C     ICC=1
```

```
      EQUIV=.F.
      CALL CALFAC(FACP)
      FACP=EN(2F)
      ICI=1
      ICO=1
      IISW=2
      CALL PCROP1(IISW)
      SO(1,3)=SO(1,3)+COND
      SO(1,14)=COND
      SO(2,3)=0.
      SO(1,3)=0.
      DO 70 J=6,JJ
      SO(1,3)=SO(1,3)+SO(1,J)
      SO(2,3)=SO(2,3)+SO(2,J)
   70 CONTINUE
      EN(20)=SO(2,4)
      CALL DISKIO(2,NE)
      RETURN
      END
```


EXCHO3
Set up the parameters used in the model such as the heat transfer coefficient UA, the heat of vaporization of naphtha ALAM, the assumed outlet temperatures TNO (naphtha) and TFO (flue gases). Calculate the heat capacity of naphtha liquid Cp at temperature T = (boiling point + inlet temperature of naphtha)/2
Calculate the heat capacity Cp of naphtha vapour at temperature T (boiling point outlet temperature of naphtha)/2
Calculate the effective thermal capacity of naphtha WCE and the thermal capacity of the flue gases WCF.
Calculate the effectiveness factor of the heat exchanger assumed to be a single pass countercurrent heat exchanger.
Calculate outlet temperatures TNOC and TFOC depending on the relative values of WCE and WCF.
Are TFOC & TNOC converged ?
no
TNO = TNOC
TFO = TFOC
yes
Calculate the total heat loss by the flue gases which is the amount of heat given to the naphtha stream. Calculate the heat addition to the naphtha liquid.
Determine whether all the naphtha liquid is evaporated and supersaturated, or all the naphtha liquid is just evaporated, or not all the naphtha liquid is evaporated in which case the fraction which is evaporated can be calculated from a heat balance.
Set the outlet flows and reset the EN vector.
RETURN END

```
      SUBROUTINE EXCH2
C     CALCULATES THE OUTLET TEMPERATURE OF A NAPHTHA SUPERHEATER
C     SI(1,4)=INLET TEMP. OF THE FLUE GAS
C     SI(2,4)=INLET TEMP. OF NAPHTHA
C     EN VECTOR
C     16. UA
C     17. FIRST ASSUMPTION OF OUTLET TEMP. OF THE FLUE GAS
C     18. FIRST ASSUMPTION OF OUTLET TEMP. OF THE NAPHTHA
C     19. INITIAL VALUE B.P. OF NAPHTHA AT PRESSURE OF SUPERHEATER VAPO
C     RIZER
C     20. FINAL B.P. OF NAPHTHA AT OPERATING PRESSURE OF SUPERHEATER
C     VAPORIZER
C     21. LATENT HEAT OF VAPORIZATION OF NAPHATHA ASSUMED TO BE CALCU
C     LATED AT INITIAL BOILING POINT
      COMMON NS(120),SI(4,16), SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSR,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(10)
      UA=EN(16)
      SO(1,4)=EN(17)
      SO(2,4)=EN(18)
      DO 130 I=5,JJ
      SO(1,I)=SI(1,I)
  130 SO(2,I)=0.
      NOPP=2
      X=(SI(2,4)+EN(19))/2.
      CALL PROCAL
      CPNL=PPSC(2,11)
C     EVALUATE WCE
      ALAM=EN(21)
      TNO=EN(18)
      TFO=EN(17)
      DO 200 IJK=1,20
      NOPP=1
      X=(TNO+EN(19))/2.
      CALL PROCAL
      CPNV=PPSC(1,8)
      FAC=SI(2,3)*(CPNL*(EN(19)-SI(2,4))+ALAM+CPNV*(TNO-EN(19)))
      WCE=FAC/(TNO-SI(2,4))
      DO 10 I=1,NOCOMP
   10 XMOLE(I)=SI(1,I+5)
      X=(TFO+SI(1,4))/2.
      NOPP=1
      CALL PROMIX(1)
      WCF=PPMX(1)
      IF(WCF-WCE) 20,20,30
C     WCF SMALLER
   20 R=WCF/WCE
      TU=UA/WCF
      GO TO 40
C     WCE IS SMALLER
   30 R=WCE/WCF
      TU=UA/WCE
   40 X=1.-R
C  ASSUME A SINGLE PASS COUNTERCURRENT EXCHANGER
      Y=EXP(-TU*X)
      E2=(1.-Y)/(1.-Y*R)
      IF(EN(5).GE.1.) WRITE(6,777) WCF,WCE,E2
  777 FORMAT(1X,*WCF=*,E12.4,5X,*WCE=*,E12.4,5X,*E2=*,F10.4)
      IF(WCF-WCE) 50,50,60
   50 TFOC=SI(1,4)-E2*(SI(1,4)-SI(2,4))
      TNOC=SI(2,4)+R*(SI(1,4)-TFOC)
      GO TO 70
   60 TNOC=SI(2,4)+E2*(SI(1,4)-SI(2,4))
      TFOC=SI(1,4)-R*(TNOC-SI(2,4))
   70 CONTINUE
      DELTE=ABS((TNOC-TNO)/TNOC)
      DELTF=ABS((TFOC-TFO)/TFOC)
      IF(DELTE.LE..001.AND.DELTF.LE.0.01) GO TO 300
C     DIRECT ITERATION
      TNO=TNOC
      TFO=TFOC
  200 CONTINUE
      WRITE(6,999) NE
```

```
  999 FORMAT(1X,'OUTLET TEMPERATURE DID NOT CONVERGE IN EXCHG3, EQUIPMEN
     'T NO.',I5)
  300 CONTINUE
      IF(EN(5).GE.1.) WRITE(6,778) TFOC,TNOC
  778 FORMAT(1X,'TFOC=',F8.3,5X,'TNOC=',F8.3)
      Q=WCP*(SI(1,4)-TFOC)
      QSL=SI(2,3)*CPNL*(EN(19)-SI(2,4))
      DT=TNOC-EN(20)
      IF(ABS(DT/TNOC).LE.0.001) GO TO 500
      IF(DT) 400,510,600
C     LIQUID NOT ALL EVAPORATED
  400 QVAP=Q-QSL
      FVAP=QVAP/(SI(2,3)*ALAM)
      SO(2,16)=FVAP*SI(2,3)
      SO(2,13)=SI(2,3)-SO(2,16)
      WRITE(6,997) FVAP,NE
  997 FORMAT(1X,'ONLY THE FRACTION',F6.3,' OF THE NAPHTHA IS VAPORIZED IN
     ' EXCHG3, EQUIPMENT NO.',I5 )
      GO TO 700
C     LIQUID JUST EVAPORATED
  500 CONTINUE
      WRITE(6,998) NE
  998 FORMAT(1X,'NAPHTHA JUST EVAPORATED IN EXCHG3, WITH NO SUPER HEAT
     'IN EQUIPMENT NO.',I5)
C       LIQUID ALL EVAPORATED AND SUPERHEATED
  600 SO(2,13)=SI(2,3)
      SO(2,16)=0.
  700 SO(2,4)=TNOC
      SO(2,3)=SI(2,3)
      SO(1,4)=TFOC
      SO(1,5)=SI(1,5)
      SO(2,5)=SI(2,5)
      SO(1,3)=SI(1,3)
      EN(18)=TNOC
      EN(17)=TFOC
      CALL DISKIO(2,NE)
      RETURN
      END
```


EXCHO4
Set up the parameters used in the model. TD is the desired outlet temperature of the reactant gases and TMAX is the maximum temperature allowed for the oulet reactant gases. Assume outlet temperatures of the flue gases TFØ and of the reactant gases TRØ.
Calculate the actual outlet temperatures by calling subroutine HEX. (TROC and TFOC)
TROC:TD
<
=
>
TROC:TMAX
≤
>
It is necessary to use the auxillary burner to increase the temperature of the reactant gases.
Set output stream vectors
RETURN END
It is necessary to cool the reactant gases by adding water between the cocurrent and countercurrent part of the exchanger to decrease the reactant gases temperature.
Set the mole addition of air and atomizing steam per kg of fuel added.
Calculate the heat input with the gases and the fuel and determine the adiabatic flame temperature TFLAME.
Calculate the temperatures out of the exchanger for both streams TFINT & TRINT, using the same effectiveness factor determined in HEX.
Assume an initial value for the fuel flowrate required and the temperature of the gas mixture.
Determine the amount of superheat in the reactant gases and determine the flow of water needed to reduce the temperature to TMAX.
Assume a new value for TRØ and TFØ due to the combustion of fuel; determine TROC and TFOC by calling HEX.
Calculate TFOC and TROC by using the same effectiveness factor and the same method as in subroutine HEX for the second part of the heat exchanger.
Newton's convergence method on the fuel until desired TROC is obtained
Iterate until TROC = TMAX
Set stream vectors, calculate pressure drop of the reactant gases stream and reset EN vector
Set stream vectors, calculate pressure drop of the reactant gases stream and reset EN vector
RETURN END

```
      SUBROUTINE EXCH04
C
C     MODELS HEAT EXCHANGER WHICH HAS FIRST PART COCURRENT WITH FLUE GAS
C     AND SECOND PART COUNTERCURRENT
C     DETERMINES OUTLET TEMPERATURE FROM HEAT EXCHANGER WHICH PREHEATS
C     REAVTANT GASES AND COMPARES THE TEMPERATURE OF REACTANT GASES
C     WITH A DESIRED CONTROLLED TEMPERATURE.THREE OPTIONS EXIST,
C        1. OUTLET TEMPERATURE=DESIRED TEMPERATURE-EXIT MODEL
C        2.OUTLET TEMPERATURE EXCEEDS A MAXIMUM ALLOWED OUTLET T,THEN
C     ADD WATER BETWEEN COCURRENT AND COUNTERCURRENT EXCHANGER TO
C     PRODUCE AN OUTLET TEMPERATURE=TMAX
C        3. OUTLET TEMPERATURE BELOW DESIRED TEMPERATURE,THEN DETERMINE
C     FUEL REQUIRED IN AUXILLARY BURNER TO BRING OUTLET TEMP. TO DESIRED
C     TEMPERATURE
C
C        STREAM ORDER DESIGNATIONS
C     SI(1,-)=REACTANT GASES
C     SI(2,-)=FLUE GAS FROM REFORMER
C     SI(3,-)=FUEL TO AUXILLARY BURNER
C     SI(4,-)=AIR TO AUXILLARY BURNER
C     SO(1,-)=REACTANT GASES
C     SO(2,-)=FLUE GAS(REFORMER+AUXILLARY BURNER TOTAL)
C     SO(3,-)=ATOMIZING STEAM REQUIRED
C     SO(4,-)=WATER REQUIRED TO COOL GASES
C
C        EN VECTOR
C     EN(16)=CARBON-TO-HYDROGEN WEIGHT RATIO IN FUEL
C     EN(17)=FRACTION EXCESS AIR
C     EN(18)=HEAT OF COMBUSTION OF FUEL AT TB KCAL/KGM(POSITIVE LHV)
C     EN(19)=BASE TEMPERATURE
C     EN(20)=ATOMIZING STEAM/KGM FUEL FED(KMOLES/KGM)
C     EN(21)=INITIAL ESTIMATE OF ADIABATIC FLAME TEMPERATURE
C     EN(22)= ASUMED INITIAL ESTIMATE OF FUEL REQUIRED(MAY BE 0.)
C     EN(23)=DESIRED OUTLET TEMP.OF REACTANT GASES
C     EN(24)=INITIAL GUESS OF REACTANT TEMPERATURE OF REACTANT GASES
C     EN(25)=INITIAL ESTIMATE OF OUTLET TEMPERATURE OD FLUE GASES
C     EN(26)= ASSUMED  TEMPERATURE OF THE MIXTURE OF BURNER AND REFORMER
C     GASES
C     EN(27)=MAXIMUM TEMPERATURE ALLOWED FOR OUTLET REACTANT TEMP.
C     EN(28)=FRACTION OF EXCHANGER WHICH IS COCURRENT
C     EN(29)=OVERALL HEAT TRANSFER COEFFICIENT-AREA PRODUCT --UA
C     EN(30) = FACP = FACTOR FOR PRESSURE DROP CALCULATION OF THE
C                     REACTANT GASES STREAM
C
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(72,16)
      COMMON/GENCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,NSN,MODE,NPLNT,ISP,NC,NCALC,NSR,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      COMMON/H/F,UA,IISW,E1,WCR,WCE,E2
      COMMON/P/ICI,ICO,FACP
C     COMMON/NAEJ/EQNU,ICC
      DIMENSION YMOL(11),ZMOL(11),SS(14),STI(14)
C
      DIMENSION FUEL(2),DT(2)
      HLAM(T)=(33.31393 -0.03524*T +0.00005*T*T)*18.*18.
C
C     STORE SI(1,-),SI(2,-) FOR FUTURE USE
      NCOM=NOCOMP+3
      DO 1 I=1,NCOM
      STI(I)=  SI(1,I+2)
    1 SS(I)=SI(2,I+2)
C     CALCULATE WHETHER AUXILLARY BURNER REQUIRED,I.E. IF TO=TC,TO.LT.TO
C     OR TO.GT.TMAX
C     CALCULATE HEAT EXCHANGE WITH FRACTION F OF HEAT EXCHANGER IN
C     COCURRENT FLOW AND (1-F) IN COUNTERCURRENT
C
      SET UP PARAMETERS NEEDED IN MODEL
      JJSW=0
      KSW=0
      WW=0.
      MM=1
      IISW=0
      DF=5.
      FUEL(2)=0.
      TMAX=EN(27)
      F=EN(28)
      ISW=0
      TO=EN(23)
```

```
      UA=EN(29)
      TADC=EN(21)
      TC=EN(26)
C     ASSUME SAME AVERAGE HEAT CAPACITY PERTAINS TO EACH HEATEXCHANGER
C
      TRO=EN(24)
      TFO=EN(25)
      CALL          HEX(TRO,TFO,TROC,TFOC,R)
C     TEST IF TRO AT,BELOW OR ABOVE TD
      DELDT=TD-TROC
      TDT =ABS(DELDT/TD)
C       IF OUTLET TEMPERATURE IS AT DESIRED VALUE  GO TO 2000
      IF(TDT.LE.0.001) GO TO 2000
      IF(JJSW) 300,250,250
C     DETERMINE IF FIRST TIME EVALUATED TDT ,JJSW=0 FIRST TIME,JJSW=-1
C     NEED WATER,IF JJSW=+1 NEED BURNER
  250 IF(DELDT) 300,300,400
  400 JJSW=1
C     NEED BURNER CALCULATION
CC    CALCULATE ADIABATIC FLAME TEMPERATURE
C     SET CONSTANTS
      CH=EN(16)
      FEX=EN(17)
      HC=EN(18)
      TB=EN(19)
C     PER KGM OF FUEL -ZMOL=FLOW IN ,YMOL =FLOW OUT
      DO 10 I=1,NOCOMP
      YMOL(I)=0.
   10 ZMOL(I)=0.
      DEN=12.*(CH+1.)
      ZMOL(1)=EN(20)
      THO2=(CH+3.)/DEN
      ZMOL(7)=THO2*(1.+FEX)
      ZMOL(6)=ZMOL(7)*79./21.
      ZMOL(10)=1.
      YMOL(1)=ZMOL(1)+0.5/DEN*12.
      YMOL(2)=CH/DEN
      YMOL(6)=ZMOL(6)
      YMOL(7)=FEX*THO2
      TADA=EN(21)
C     CALCULATE HEAT INPUT WITH GASES AND FUEL
      X=(SI(4,4)+TB)/2.
      NOPP=1
      CALL PPOCAL
      CPO=PPSC(1,7)
      CPN=PPSC(1,6)
      X=(SO(3,4)+TB)/2.
      CALL PPOCAL
      CPS=PPSC(1,1)
      NOPP=2
      X=(SI(3,4)+TB)/2.
      CALL PPOCAL
      CF=PPSC(2,10)
      FAC1=ZMOL(10)*CF*(SI(3,4)-TB)+(ZMOL(6)*CPN+ZMOL(7)*CPO)*(SI(4,4)
     *-TB)+ZMOL(1)*CPS*(SO(3,4)-TB)+HC
      DO 30 IJK=1,40
      X=(TADA+TB)/2.
      DO 20 I=1,NOCOMP
   20 XMOLE(I)=YMOL(I)
      NOPP=1
      CALL PROMIX(1)
      WCA=PPMX(1)
      TADC=FAC1/WCA +TB
      IF(ABS((TADC-TADA)/TADC).LE.0.001) GO TO 40
      TADA=TADC
C     DIRECT ITERATION ON ADIABATIC TEMPERATURE
      IF(EN(5).GE.1.) WRITE(6,777) TADC
  777 FORMAT(1X,*TADC=*,F10.3)
   30 CONTINUE
C
      WRITE(6,999) NE
  999 FORMAT(1X,*ADIABATIC FLAME TEMPERATURE DID NOT CONVERGE IN EXCH04,
     * EQUIPMENT NO.,*,I5)
   40 CONTINUE
      TASS=EN(26)
      FF=EN(22)
      DO 490 IKJ=1,50
```

```
C       FUEL RATE CALCULATION
C       ASSUMED TEMPERATURE OF COMBINED FLUE GASES
C 490   FF=ASSUMED FUEL FLOW(J-5)*FF
        DO 450 IJK=1,20
C       MIX PRODUCTS OF COMBUSTION
C
C 100   ORIGINAL SI(2,-) FLUE GAS IS STORED IN SS(I) VECTOR OR ELSE LOSE
C       IT HERE
        DO 410 I=1,NOCOMP
        J=I+5
        SI(2,J)=SS(I+3)+FF*YMOL(I)
  410   XMOLE(I)=SI(2,J)
C       DETERMINE TEMP BY HEAT BALANCE
        X=(TASS+SS(2)  )/2.
        NOPP=1
        CALL PROMIX(1)
        WCF=PPMX(1)
        DO 420 I=1,NOCOMP
  420   XMOLE(I)=FF*YMOL(I)
        X=(TADC+SS(2)  )/2.
        CALL PROMIX(1)
        WCB=PPMX(1)
        TC=SS(2)  +WCB/WCF*(TADC-SS(2))
        DELT=ABS(TC-TASS)
        IF(DELT/TC.LE.0.001) GO TO 500
        TASS=TC
  450   CONTINUE
        WRITE(6,997) NE
  997   FORMAT(1X,*THE MIXED TEMPERATURE AFTER THE BURNER DID NOT CONVERGE
       *IN EXCH04, EQUIP. NO.,*,I5)
  500   CONTINUE
C       CALCULATE NEW TRO
        SI(2,4) =TC
C       CONVERGENCE IN FF
        ISW=1
C       DIFFERENCE IN IN INLET TEMPERATURE OF FLUE GAS GIVES BETTER
C       ESTIMATE OF TRO AND TRO
        DET=SI(2,4)-SS(2)
        TRO=DET+TROC
        TFO=DET+TFOC
        CALL HEX(TRO,TFO,TROC,TFOC,R)
        FUEL(MM)=FF
C       NEWTON-S METHOD TO CONVERGE ON THE FUEL REQUIRED
        FUEL(MM)=FF
        DT(MM)=TROC-TD
  350   IF(ABS(DT(MM)/TD).LE.0.001) GO TO 480
        IF(MM.EQ.1) GO TO 442
        IF(KSW.EQ.1) GO TO 441
  360   SLOPE=(DT(1)-DT(2))/(FUEL(1)-FUEL(2))
        KSW=1
  441   FUEL(2)=FUEL(2)-DT(2)/SLOPE
        FF=FUEL(2)
        GO TO 490
  442   IF(DT(MM).LT.0.) FF=FF+DF
        IF(DT(MM).GT.0.) FF=FF-DF
        MM=2
  490   CONTINUE
        WRITE(6,995) NE
  995   FORMAT(1X,*THE FUEL REQUIREMENTS DID NOT CONVERGE IN EXCH04,EQUIPM
       *ENT NO.,*,I5)
  480   CONTINUE
C       RESET SI(2,-) VECTOR AND SI(1,-) VECTOR
        DO 425 I=3,JJ
        SI(1,I)=SI(I-2)
  425   SI(2,I)=SS(I-2)
        SO(1,3)=SI(1,3)
        SO(2,5)=SI(2,5)
C       SET SI(3,-) VECTOR
        DO 491 J=6,JJ
  491   SI(3,J)=0.
        SI(3,3)=FF
        SI(3,15)=FF
C 110   SET COMBUSTION GAS FLOW
        DO 492 J=6,JJ
        SO(1,J)=SI(1,J)
        SI(4,J)=0.
```

```
      SO(4,J)=0.
      SO(3,J)=0.
  492 SO(2,J)=SI(2,J)+WMOL(J-5)*FF
      SO(4,14)=1.
      SO(2,3)=0.
      DO 100 J=6,JJ
  100 SO(2,3)=SO(2,3)+SO(2,J)
      SO(2,4)=TFOC
      SO(1,4)=TROC
C     SET AIR
      SI(4,11)=ZMOL(6)*FF
      SI(4,12)=ZMOL(7)*FF
      SI(4,3)=SI(4,11)+SI(4,12)
C     SET WATER
      SO(4,3)=0.
      SO(3,6)=ZMOL(1)*FF
      SO(3,3)=SO(3,6)
C
C      PRESSURE DROP CALCULATION
C
      SO(1,5)=SI(1,5)
      SO(1,5)=EN(30)
      EONU=NE
      ICC=1
C     CALL CALFAC(FACP)
      FACP=EN(30)
      ICI=ICO=1
      CALL PDROP1(2)
      GO TO 3000
  300 JJSW=-1
C     NEED TO ADD WATER BETWEEN COCURRENT AND CONNTERCURRENT EXCHANGERS
C     ADD ENOUGH WATER TO MAKE TRO=TMAX. IF TROC IS LESS THEN TMAX BUT
C     GREATER THAN TO , DO NOT ADD WATER . TEST THIS FIRST
C
      IF(TROC .LE. TMAX) GO TO 2000
      SO(1,4)=TROC
C     NOW TROC IS GREATER THAN TMAX
C
C     CALCULATE INTERMEDIATE TEMPERATURE AFTER COCURRENT EXCHANGER
C     ASSUME WCP IS CONSTANT AT OVERALL VALUE PREVIOUSLY CALCULATED
C     ASSUME WCR SAME AS WELL (SLIGHT ERROR SINCE NOT THE SAME AVERAGE
C     TEMPERATURE)
C
C     E1 KNOWN FROM PREVIOUS CALCULATION
      IF(IISW-1) 350,350,360
  350 TRINT=SI(1,4) +E1*(SI(2,4)-SI(1,4))
      TFINT=SI(2,4)-R*(TRINT-SI(1,4))
      GO TO 380
  360 TFINT=SI(2,4)-E1*(SI(2,4)-SI(1,4))
      TRINT=SI(1,4)+R*(SI(2,4)-TFINT)
  380 CONTINUE
C     ADD WATER AT TRINT,SET UP CONVERGENCE ON WW
      DO 390 IKK=1,20
C      DETERMINE AMOUNT OF SUPERHEAT IN REACTANT GASES
C     FIND APPROXIMATE WATER NEEDED BY FINDING HOW MUCH HEAT CONTAINED
C     ABOVE TMAX
      NOPP=1
      X=(TFINT  +TMAX)/2.
      DO 9 J=1,NOCOMP
    9 XMOLE(J)=SI(1,J+5)
      CALL PROMIX(1)
      Q=PPMX(1)*(SO(1,4)-TMAX)
      TFINT=TFINT-Q/PPMX(1)
      WW=Q/HLAM(TRINT)+WW
      SI(1,6)=STI(4)+WW
C      CALCULATE OUTLET TEMPERATURE
      DO 200 IJK=1,20
      TAR=(TRO+TRINT)/2.
      TAF=(TFO+TFINT)/2.
C     EVALUATE SUM OF WCPS
      NOPP=1
      X=TAR
      DO 110 I=1,NOCOMP
  110 XMOLE(I)=SI(1,I+5)
      CALL PROMIX(1)
      WCR=PPMX(1)
      X=TAF
```

```
      DO 120 I=1,NOCOMP
  120 XMOLE(I)=SI(2,I+5)
      CALL PROMIX(1)
      WCF=PPMX(1)
      IF(WCR-WCF) 130,130,140
  130 R=WCR/WCF
C       WCR IS SMALLER
      TU=UA/WCR
      IISW=1
      GO TO 150
  140 R=WCF/WCR
C       WCF IS SMALLER
      TU=UA/WCF
      IISW=2
  150 CONTINUE
C         FOR COUNTER CURRENT PART OF EXCHANGER
      TUC=TU*F
      TUCC=TU-TUC
      X=1.-R
      Y=EXP(-TUCC*X)
      E2=(1.-Y)/(1.-Y*R)
C       CALCULATE OUTLET TEMPERATURE DEPENDING ON WHICH WC IS SMALLER
      IF(IISW-1) 160,160,170
C       WCR IS SMALLER
  160 TROC=TRINT+E2*(TFINT-TRINT)
      TFOC=TFINT-R*(TROC-TFINT)
      GO TO 199
  170 TFOC=TFINT-E2*(TFINT-TRINT)
      TROC=TRINT+R*(TFINT-TFOC)
  199 DELR=ABS((TROC-TRO)/TROC)
      DELF=ABS((TFOC-TFO)/TFOC)
      IF(DELR.LE.0.001.AND.DELF.LE.0.001) GO TO 240
C       DIRECT ITERATION
      TRO=TROC
      TFO=TFOC
  200 CONTINUE
  998 FORMAT(1X,*THE OUTLET TERMPERATURES DID NOT CONVERGE IN EXCH04,EQ
     *IPMENT NO.,*,I5)
      WRITE(6,998)NE
  240 CONTINUE
      SO(1,4)=TROC
      DELT=(TROC-TMAX)/TMAX
      IF(ABS(DELT).LE.0.001)GO TO 2000
      IF(EN(5).GE.1.) WRITE(6,955) WW
  955 FORMAT(1X,F12.5)
  390 CONTINUE
C
      WRITE(6,988) NE
  988 FORMAT(1X,"WATER DID NOT CONVERGE IN EXCH04,EQUIP.NO.,*,I5)
 2000 CONTINUE
      DO 2010 J=1,NOCOMP
      K=J+5
      SO(1,K)=STI(J+3)
      SI(2,K)=SS(J+3)
 2010 SO(2,K)=SI(2,K)
      DO 2015 J=3,5
      SI(1,J)=STI(J-2)
 2015 SI(2,J)=SS(J-2)
      SO(1,6)=SI(1,6)+WW
      DO 2030 I=1,4
      SO(I,3)=0.
      DO 2020 J=6,JJ
      SO(I,3)=SO(I,3)+SO(I,J)
 2020 CONTINUE
 2030 CONTINUE
C         SET WATER
      SO(4,14)=WW
      SO(4,3)=WW
      SO(1,4)=TROC
      SO(2,4)=TFOC
C
C       PRESSURE DROP CALCULATION
C
C     SO(1,5)=SI(1,5)
C     SO(1,5)=EN(30)
C     EQNU=NE
```

```
C       ICC=1
C       CALL CALFAC(FACP)
        FACP=EN(30)
        ICI=ICO=1
        CALL PROP1(2)
        SO(2,5)=SI(2,5)
 3000 CONTINUE
C         RESET EN VECTOR
        EN(21)=TACC
        EN(22)=FF
        EN(24)=TPCC
        EN(25)=TFOC
        EN(26)=TC
        CALL DISKIO(2,NE)
C           WRITE OUT CALCULATED SI VECTORS
        IF(EN(5).LT.1.) GO TO 669
        DO 667 I=1,4
        WRITE(6,666)(SI(I,J),J=1,JJ)
        WRITE(6,668)
  667 CONTINUE
  668 FORMAT(5X,'-----')
  666 FORMAT(1X,5F15.5)
  669 RETURN
        END
```

Formulation to Determine the Exit Temperatures for the Cocurrent-Countercurrent Heat Exchanger for EXCH04

Figure 8.4 shows the flow arrangement in the heat exchanger and the nomenclature used. The formulation is based on the definition of the effectiveness factors for the two parts of the exchanger.

The analysis is different depending on the relative magnitude of the heat capacities of the reactant gases WC_R and of the flue gases WC_F.

A. For $WC_R < WC_F$ and $R = WC_R/WC_F$

By definition of the effectiveness factor (ε_1) for the co-exchanger:

$$T_{R_{int}} = T_{R_i} + \varepsilon_1(T_{F_i} - T_{R_i}) \qquad (E.1)$$

and then by heat balance and equation (E.1)

$$T_{F_{int}} = T_{F_i} - R*(T_{R_i} + \varepsilon_1*(T_{F_i} - T_{R_i})) + R*T_{R_i} \qquad (E.2)$$

From the definition of the effectiveness factor (ε_2) for the counter-current part of the exchanger:

$$T_{R_0} = T_{R_{int}} + \varepsilon_2(T_{F_{int}} - T_{R_{int}}) \qquad (E.3)$$

Substituting $T_{R_{int}}$ and $T_{F_{int}}$ from equations (E.1) and (E.2) in equation (E.3), yields

$$T_{R_0} = T_{R_i} + (T_{F_i} - T_{R_i})*(\varepsilon_1+\varepsilon_2-\varepsilon_1\varepsilon_2*(1+R)) \quad (E.4)$$

An overall heat balance gives T_{F_0}, viz:

$$T_{F_0} = T_{F_i} - R*(T_{R_0} - T_{R_i}) \quad (E.5)$$

B. For $WC_R > WC_F$ and $R = WC_F/WC_R$

By definition of ε_1,

$$T_{F_{int}} = T_{F_i} - \varepsilon_1 * (T_{F_i} - T_{R_i}) \quad (E.6)$$

and a heat balance,

$$T_{T_{int}} = T_{R_i} + R*\varepsilon_1*(T_{F_i} - T_{R_i}) \quad (E.7)$$

Then by definition of ε_2,

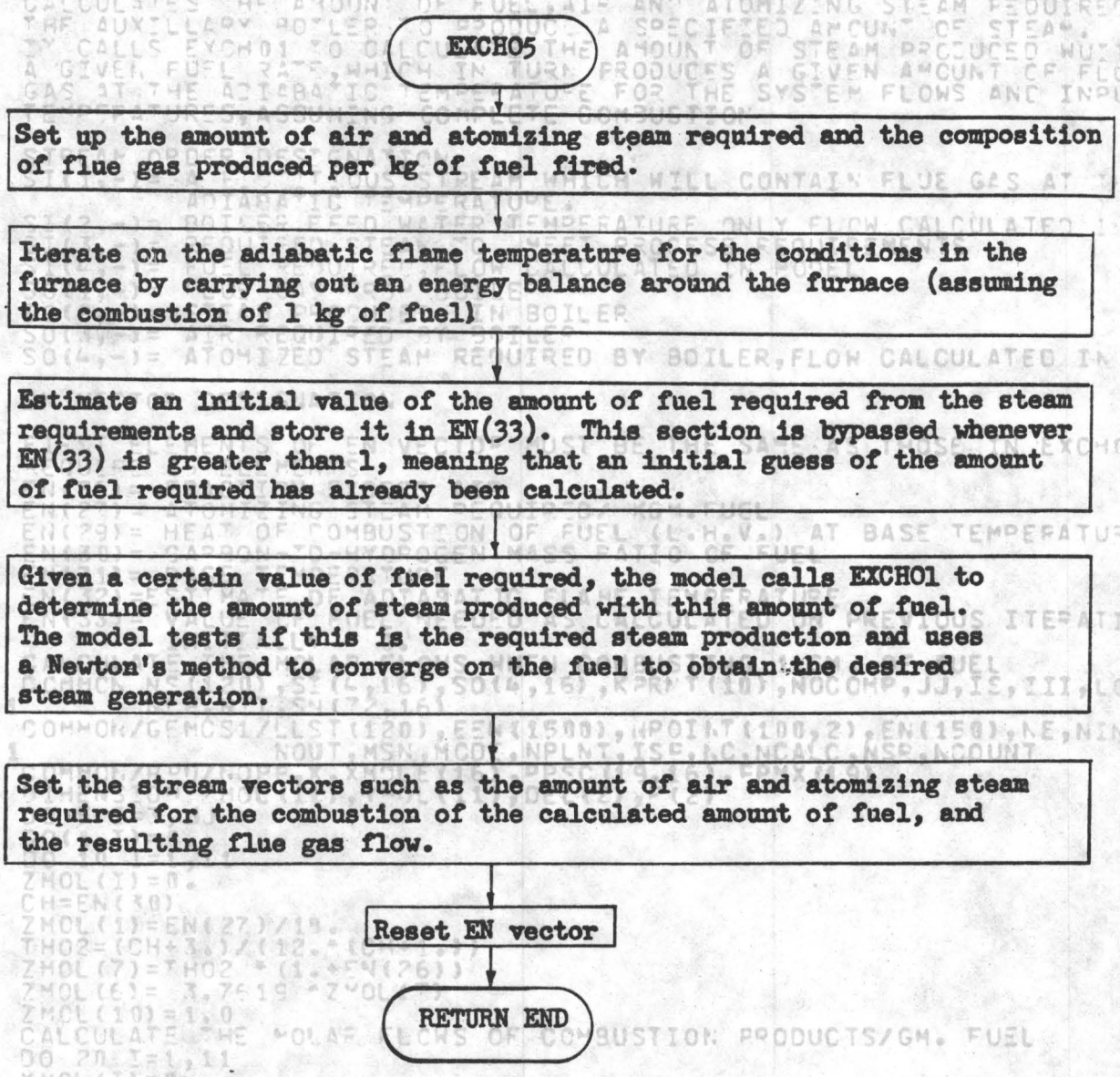
$$T_{F_0} = T_{F_{int}} - \varepsilon_2*(T_{F_{int}} - T_{R_{int}}) \quad (E.8)$$

and substituting for the intermediate temperatures

$$T_{F_0} = T_{F_i} + R^*(T_{F_i} - T_{F_0}) \qquad \text{(E.9)}$$

These formulations are incorporated into subroutine HEX which is a subroutine called from EXCHO4.

```
      SUBROUTINE HEX(TRO,TFO,TROC,TFOC,P)
C
C       SUBROUTINE CALLED BY EXCHG4 TO CALCULATE OUTLET TEMPERATURES
C     FROM REACTANT SUPERHEATER
      COMMON BS(12),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,NIT,SP(4,16)
      COMMON /GEMCSI/LLST(120),EEN(150),NPOINT(100,2),EN(150),NE,NIN,
     1               NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSE,NCOUNT
      COMMON /PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      COMMON/H/F,UA,IISW,E1,WCR,WCE,E2
C     START CALCULATION OF OUTLET TEMPERATURES FOR EXCHG2
      DO 200 IJK=1,20
      TAR=(TRO+SI(2,4))/2.
      TAF=(TFO+SI(2,4))/2.
C     EVALUATE SUM OF WCPS
      NOPP=1
      X=TAR
      DO 110 I=1,NOCOMP
  110 XMOLE(I)=SI(1,I+5)
      CALL PROMIX(1)
      WCR=PPMX(1)
      X=TAF
      DO 120 I=1,NOCOMP
  120 XMOLE(I)=SI(2,I+5)
      CALL PROMIX(1)
      WCF=PPMX(1)
      IF(WCR-WCF) 130,130,140
  130 R=WCF/WCR
C       WCR IS SMALLER
      TU=UA/WCR
      IISW=1
      GO TO 150
  140 R=WCF/WCR
C       WCF IS SMALLER
      TU=UA/WCF
      IISW=2
  150 CONTINUE
C       FOR COUNTER CURRENT PART OF EXCHANGER
C     EFFECTIVENESS FACTOR COCURRENT PART OF EXCHANGER
      TUC=TU*F
      E1=(1.-EXP(-TUC*(1.+R)))/(1.+R)
      TUCC=TU-TUC
      X=1.-R
      Y=EXP(-TUCC*X)
      E2=(1.-Y)/(1.-Y*R)
C       CALCULATE OUTLET TEMPERATURE DEPENDING ON WHICH WC IS SMALLER
      IF(IISW-1) 160,160,170
C       WCR IS SMALLER
  160 FAC2=E1+E2-E1*E2*(1.+R)
      TROC=SI(1,4)*(1.-FAC2)+SI(2,4)*FAC2
      TFOC=SI(2,4)-R*(TROC-SI(1,4))
      GO TO 199
C     WCF IS SMALLER
  170 FAC2=E1+E2-E1*E2*(1.+R)
      TFOC= SI(2,4)*(1.-FAC2)+ SI(1,4)*FAC2
      TROC= SI(1,4)+ R*(SI(2,4)-TFOC)
  199 DELR=ABS((TROC-TRO)/TROC)
      DELF=ABS((TFOC-TFO)/TFOC)
      IF(DELR.LE.0.001.AND.DELF.LE.0.01) GO TO 240
C       DIRECT ITERATION
      TRO=TROC
      TFO=TFOC
  200 CONTINUE
C
  205 FORMAT(1X,*THE OUTLET TEMPERATURE DID NOT CONVERGE IN HEX,    ,EQU
     *IPMENT NO.=*,F5)
      WRITE(6,998)NE
  240 CONTINUE
      IF(EN(5).GE.1.) WRITE(6,777) E1,E2,WCF,WCR,R
  777 FORMAT(1X,*E1=*,F7.4,*E2=*,F10.4,*WCF=*,F10.4,*WCR=*,F10.4,*R=*,F
     *10.4)
      RETURN
      END
```


EXCH05
Set up the amount of air and atomizing steam required and the composition of flue gas produced per kg of fuel fired.
Iterate on the adiabatic flame temperature for the conditions in the furnace by carrying out an energy balance around the furnace (assuming the combustion of 1 kg of fuel)
Estimate an initial value of the amount of fuel required from the steam requirements and store it in EN(33). This section is bypassed whenever EN(33) is greater than 1, meaning that an initial guess of the amount of fuel required has already been calculated.
Given a certain value of fuel required, the model calls EXCH01 to determine the amount of steam produced with this amount of fuel. The model tests if this is the required steam production and uses a Newton's method to converge on the fuel to obtain the desired steam generation.
Set the stream vectors such as the amount of air and atomizing steam required for the combustion of the calculated amount of fuel, and the resulting flue gas flow.
Reset EN vector
RETURN END

```
      SUBROUTINE EXCH05
C
C     CALCULATES THE AMOUNT OF FUEL,AIR AND ATOMIZING STEAM REQUIRED IN
C     THE AUXILLARY BOILER TO PRODUCE A SPECIFIED AMOUNT OF STEAM.
C     IT CALLS EXCH01 TO CALCULATE THE AMOUNT OF STEAM PRODUCED WHITH
C     A GIVEN FUEL RATE,WHICH IN TURN PRODUCES A GIVEN AMOUNT OF FLUE
C     GAS AT THE ADIABATIC TEMPERATURE FOR THE SYSTEM FLOWS AND INPUT
C     TEMPERATURES,ASSUMING COMPLETE COMBUSTION
C
C     STREAM ORDER DESIGNATION
C     SI(1,-)= A FICTITIOUS STREAM WHICH WILL CONTAIN FLUE GAS AT THE
C              ADIABATIC TEMPERATURE.
C     SI(2,-)= BOILER FEED WATER TEMPERATURE ONLY FLOW CALCULATED IN MOD
C     SI(3,-)= REQUIRED STEAM TO  MEET PROCESS REQUIREMENTS
C     SI(4,-)= FUEL REQUIRED,FLOW CALCULATED IN MODEL
C     SO(1,-)= FLUE GAS FROM BOILER
C     SO(2,-)= STEAM PRODUCED  IN BOILER
C     SO(3,-)= AIR REQUIRED BY BOILER
C     SO(4,-)= ATOMIZED STEAM REQUIRED BY BOILER,FLOW CALCULATED IN MODL
C
C     EN VECTOR DESIGNATION
C
C     FIRST ELEMENTS OF EN VECTOR MUST BE THE SAME AS THOSE IN EXCH01
C     REQUIRE 25 ELEMENTS
C     EN(26)= FRACTION EXCESS AIR
C     EN(27)= ATOMIZING STEAM REQUIRED/ KGM.FUEL
C     EN(29)= HEAT OF COMBUSTION OF FUEL (L.H.V.) AT BASE TEMPERATURE
C     EN(30)= CARBON-TO-HYDROGEN MASS RATIO OF FUEL
C     EN(31)= BASE TEMPERATURE
C     EN(32)=ESTIMATE OF ADIABATIC FLAME TEMPERATURE
C     EN(33)= VALUE OF FUEL NEEDED AS CALCULATED ON PREVIOUS ITERATION,
C             INITIALLY = 0.
C     CALCULATE THE MOLAR FLOWS WHEN COMBUSTING 1 GM. OF FUEL
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSR,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      DIMENSION ZMOL(11),YMOL(11),DEL(2),F(2)
      DO 5 I=3,JJ
    5 SO(1,I)=0.
      DO 10 I=1,11
   10 ZMOL(I)=0.
      CH=EN(30)
      ZMOL(1)=EN(27)/18.
      THO2=(CH+3.)/(12.*(CH+1.))
      ZMOL(7)=THO2 *(1.+EN(26))
      ZMOL(6)= 3.7619 *ZMOL(7)
      ZMOL(10)=1.0
C     CALCULATE THE MOLAR FLOWS OF COMBUSTION PRODUCTS/GM. FUEL
      DO 20 I=1,11
   20 YMOL(I)=0.
      YMOL(1)=ZMOL(1)+0.5/(CH+1.)
      YMOL(6)=ZMOL(6)
      YMOL(7)=THO2 *EN(26)
      YMOL(2)=CH/(12.*(CH+1.))
C     CALCULATE ADIABATIC FLAME TEMPERATURE FOR CONDITIONS IN FURNACE
      TB=EN(31)
C     CALCULATE AVERAGE AIR HEAT IN
      NOPP=1
      X=(SO(3,4)+TB)/2.
      CALL PROCAL
      CPO2=PPSC(1,7)
      CPN2=PPSC(1,6)
C     CALCULATE MEAN SPECIFIC HEAT OF STEAM
      X=(SO(4,4)+TB)/2.
      CALL PROCAL
      CPSTM=PPSC(1,1)
C     CALCULATE MEAN CP OF FUEL
      X=(SI(4,4)+TB)/2.
      NOPP = 2
      CALL PROCAL
      CPF=PPSC(2,10)
C     CALCULATE  SUM OF WC FOR COMUBSTION PRODUCTS AND FLAME TEMPERATURE
      TADA=EN(32)
      HI= EN(29)+ ZMOL(1)*CPSTM*(SO(4,4)-TB)+ZMOL(6)*CPN2*(SO(3,4)-TB)+
     *ZMOL(7)*CPO2*(SO(3,4)-TB)+ZMOL(10)*CPF*(SI(4,4)-TB)
```

```
      IF(EN(5).GE.1.) WRITE(6,*) HI
      DO 50 IJK=1,20
      X=(TADA+TB)/2.
      IF(EN(5).GE.1.) WRITE(6,*) X
      NCPP=1
      DO 500 I=1,NOCOMP
  500 XMOLE(I)=YMOL(I)
      IF(EN(5).GE.1.) WRITE(6,*) (XMOLE(I),I=1,NOCOMP)
      CALL PROMIX(1)
      WCPM=PPMX(NOPP)
      TADC= HI/WCPM + TB
      DLT=TADC-TADA
      IF(ABS(DLT/TADC).LE.0.001) GO TO 60
C     DIRECT ITERATION
      TADA=TADC
   50 CONTINUE
      WRITE(6,999)NE
  999 FORMAT(/X,*THE ADIABATIC FLAME TEMPERATURE DID NOT CONVERGE IN COM
     *T06,EQUIP.NO.,*,I5)
   60 CONTINUE
C     ESTIMATE  INITIAL VALUE OF THE FUEL REQUIRED FROM STEAM PRODUCTION
C     REQUIRED IF NOT ALREADY CALCULATED
      IF(EN(33).GT.1.0) GO TO 100
      HLAM=7128.
      P=SI(2,5)
      TSAT=262.7
      TM=(SI(2,4)+TSAT)/2.
      X= TM
      NCPP=2
      CALL PROCAL
      CPW=PPSC(2,9)
      SI(2,3)=SI(3,3)*(1.+EN(23))
      FUEL=(SI(2,3)*CPW*(TSAT-SI(2,4))+SI(3,3)*HLAM)/(EN(29)*0.80)
C     CALCULATE PRODUCTS OF COMBUSTION FROM FUEL RATE AND USE NEWTONS
C     METHOD OF CONVERGENCE TO PROMOTE CONVERGENCE
      EN(33)=FUEL
C     STORE THIS VALUE FOR USE IN NEXT SECTION-PREVIOUS SECTION ONLY
C     USED AT THE BEGINNING IF FUEL HAS NOT BEEN ESTIMATED PREVIOUSLY
  100 JK=1
C     SET UP NEWTON*S METHOD USING JK AS INDEX
      F(JK)=EN(33)
      ISW=0
      M=0
      FF=5.
      F(JK)=FF
      DO 800 KJI=1,50
C     LIMIT NUMBER OF ITERATIONS TO CONVERGE
      SI(1,4)=TADC
      SI(1,3)=0.
      DO 140 J=1,NOCOMP
      K=J+5
      SI(1,K)=YMOL(J)*F(JK)
  140 SI(1,3)=SI(1,3)+SI(1,K)
      CALL EXCH01
      DEL(JK)= SI(3,3)-SO(2,3)
      IF(ABS(DEL(JK)/SI(3,3)).LE.0.001) GO TO 900
C     CONVERGENCE PROMOTION
      IF(JK .EQ. 2) GO TO 261
      IF(DEL(JK)) 251,250,250
  250 FF=FF+100.
      F(JK)=FF
      GO TO 800
  251 FF=FF-100.
      JK=2
      F(JK)=FF
      GO TO 800
  261 IF(ISW.EQ.1) GO TO 798
      SLOPE= (DEL(1)-DEL(2))/(F(1)-F(2))
      ISW=1
  798 F(JK)=F(JK)-DEL(JK)/SLOPE
  800 CONTINUE
      WRITE(6,899)NE
  899 FORMAT(1X,*FUEL FLOWRATE DID NOT CONVERGE IN EXCH05,EQUIPMENT NUMB
     *ER,*,I5)
  900 CONTINUE
C     SET ALL STREAM VECTORS BOTH INPUT AND OUTPUT AS REQUIRED
C     1. FUEL IN SI(4,-)
```

```
      DO 910 J=6,JJ
      SO(3,J)=0.
      SO(4,J)=0.
  910 SI(4,J)=0.
      SI(4,15)=F(2)
      SI(4,3) =F(2)
C     2. AIR REQUIRE SO(3,-)
      SO(3,11)=ZMOL(6)*F(2)
      SO(3,12)=ZMOL(7)*F(2)
      SO(3,3)=SO(3,11)+SO(3,12)
C     3.ATOMIZING STEAM IN SO(4,-)
      SO(4,6)=  ZMOL(1)*F(2)
      SO(4,3)=  SO(4,6)
      EN(32)=TADC
      EN(33)=F(2)
      CALL DISKIO(2,NE)
      RETURN
      END
```

E.3 Controller Models

CØNTO1

CØNTO2

CØNTO3

CØNTO4

CØNTO5

CØNTO6

CØNTO7

```
C
C
C
```

```
      SUBROUTINE CONTI2
C
C     TRANSFER REQUIRED FLOWRATE OF AIR AND FUEL TO REFORMER AND TO
C     AUXILIARY BURNER AS CALCULATED IN THESE MODELS
C
C     SI(1,-)  ... FUEL TO REFORMER
C     SI(2,-)  ... FUEL TO AUXILIARY BURNER
C     SI(3,-)  ... AIR TO REFORMER
C     SI(4,-)  ... AIR TO AUXILIARY BURNER
C
C     SO(1,-)  ... TOTAL FUEL REQUIREMENTS FOR THE PROCESS
C                  SUM OF SI(1,-) AND SI(2,-)
C     SO(2,-)  ... TOTAL AIR REQUIREMENTS FOR THE PROCESS
C                  SUM OF SI(3,-) AND SI(3,-)
C
C     EN(16)  ... EQUIPMENT NUMBER WHERE FRACTION OF FUEL TO REFORMER IS
C                 SPLET FROM MAIN FLOW
C     EN(17)  ... EQUIPMENT NUMBER WHERE FRACTION AIR TO REFORMER IS
C                 SPLET FROM MAIN FLOW
C
      COMMON NS(120),ST(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSP,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      DIMENSION FRAC(2)
C
      DO 10 I=1,2
      DO 10 J=6,JJ
   10 SO(I,J)=0.
      SO(1,15)=SI(1,15)+SI(2,15)
      SO(1,3)=SO(1,15)
      SO(2,12)=SI(3,12)+SI(4,12)
      SO(2,11)=SI(3,11)+SI(4,11)
      SO(2,3)=SO(2,11)+SO(2,12)
C
      FRAC(1)=SI(1,15)/SO(1,15)
      FRAC(2)=SI(3,3)/SO(2,3)
      DO 20 I=1,2
      MM=EN(15+I)+0.001
      MO=NPOINT(MM,1)
      EEN(MO+18)=FRAC(I)
      EEN(MO+19)=1.-FRAC(I)
   20 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CONTR
C     SUBROUTINE CONTROLS PRODUCTION RATE OF HYDROGEN BY CONTROLLING THE
C     NAPHTHA FEED.IT ASSUMES THE HYDROGEN PRODUCTION-TO-NAPHTHA FEED
C     RATIO IS CONSTANT ACCORDING TO THAT CALCULATED FOR THE CURRENT
C     ITERATION
C
C     EN VECTOR
C     EN(16) = DESIRED PRODUCTION RATE
C
C     STREAM ORDER INFORMATION
C     SI(1,-) = HIDROGEN PRODUCT STREAM
C     SO(1,-) =  HYDROGEN PRODUCT STREAM
C     SO(2,-) = NAPHTHA FEED STREAM-RECYCLE STREAMAND HENCE A VALUE MUST
C     BE ESTIMATED INITIALLY-MAKE EQUAL TO INPUT STREAM TO STEAM RATE
C     CONTROLLER  INITIALLY , TEMPERATURE OF NAPHTHA SET IN S-VECTOR
C
C     CALCULATE NEW NAPHTHA RATE
C     SET ALL STREAM COMPONENTSS AT ZERO INITIALLY
C
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,NIT,SN(12,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSF,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      FEEDN = SO(2,16)*EN(16)/SI(1,16)
      DO 20 J = 6,JJ
      SO(2,J) = 0.
   20 CONTINUE
      SO(2,16) = FEEDN
      SO(2,3) = SO(2,16)
C
      DO 30 J = 3,JJ
   30 SO(1,J) = SI(1,J)
      RETURN
      END
```

```
      SUBROUTINE CONTL-
C
C     REQUIRED BY THE SIMULATION TO ENSURE THE CORRECT FLOW RATE
C     IS FED TO EACH ECONOMIZER.  ONCE STEAM REQUIREMENTS ARE KNOWN, THEN
C     FLOW OF WATER MUST EQUAL THIS FLOW PLUS THE FRACTION BLOWDOWN
C
C        STREAM ORDER DESCRIPTION
C         INPUT STREAMS
C        SI(1,-)=TOTAL STEAM REQUIREMENTS FOR THE PROCESS
C        SI(2,-)=ENTERING BOILER FEED WATER
C        OUTPUT STREAMS
C        SO(1,-)=INPUT STREAM TO ECONOMIZER WHERE WATER ENTERS DIRECTLY,
C                THAT IS WHERE IT IS THE COLDEST
C        SO(2,-),SO(3,-),SO(4,-) DOES NOT REQUIRE ANY PARTICULAR ORDER
C
C        EN VECTOR
C        EN(16)=FRACTION OF STEAM PRODUCTION WHICH IS BLOW-DOWN
C
C     SET ALL FLOWS TO ZERO INITIALLY
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(2,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSR,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      DO 15 I=1,NOUT
      DO 10 J=6,JJ
      SO(I,J)=0.
   10 CONTINUE
   15 CONTINUE
C     CALCULATE WATER FLOW REQUIRED
      SO(1,14)=SI(1,6)*(1.+EN(16))
      SO(1,3)=SO(1,14)
C     SET ALL OTHER FLOWS
      DO 20 I=2,NOUT
      SO(I,14)=SO(1,14)
      SO(I,3)=SO(1,3)
   20 CONTINUE
C     SET FLOW AND T,P OF INCOMING WATER AND FIRST OUTPUT STREAM
      SI(2,3)=SO(1,3)
      SI(2,14)=SO(1,14)
      SO(1,4)=SI(2,4)
      SO(1,5)=SI(2,5)
      RETURN
      END
```

```
      SUBROUTINE CONTRE
      DETERMINES AMOUNT OF STEAM REQUIRED TO BE PRODUCED BY AUXILLARY
      BOILER

      STREAM ORDER DESCRIPTION
      SI(1,-)=STEAM REQUIRED BY THE PROCESS
      SI(2,-),SI(3,-),SI(4,-) MAY BE IN ANY ORDER THEY REPRESENT STEAM
      PRODUCED IN WASTE HEAT BOILERS
      SO(1,-)=STEAM REQUIRED BY AUXILLARY BOILER

      EN VECTOR IS OF LENGHT=15

      SUM UP STEAM PRODUCED
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NTT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSP,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      DO 5 I=3,JJ
    5 SO(1,I)=0.
      ST=0.
      DO 10 I=2,4
      ST=ST+SI(I,6)
   10 CONTINUE
      SO(1,6)=SI(1,6)-ST
      SO(1,3)=SO(1,6)
      SO(1,4)=1.
      SO(1,5)=1.

   ** SO(1,4)=SO(1,5)=1.  SINCE TEMPERATURE AND PRESSURE ARE NOT
                          REQUIRED DOWNSTREAM

      RETURN
      END
```

```
      SUBROUTINE CONTOE
C     TRANSFERS EXIT TEMPERATURE FROM FINAL ECONOMIZER TO WASTE HEAT
C     BOILER INLET WATER AND WATER TO AUXILLARY BOILER
C
C     STREAM ORDER IS NOT IMPORTANT
C     SI(1,-)=WATER FROM FINAL ECONOMIZER
C     SO(1,-)=WATER INLET NUMBERS FOR FOUR BOILERS IN PROCESS
C
C     EN VECTOR IS 5 ELEMENTS LONG
C
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SV(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSP,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      DO 10 I=1,4
      SO(I,4)=SI(1,4)
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CONTR
C
C     THIS SUBROUTINE SUMMARIZES THE FLUE GAS PRODUCED BY THE AUXILIAR
C     BURNER ALONE NEEDED FOR THE TRANSFER OF INFO FROM GEMCS TO SYMBOL
C
C     SI(1,-)   FLUE GAS FROM THE REFORMER FURNACE ALONE
C     SI(2,-)   FLUE GAS FROM THE REFORMER + AUXILIARY BURNER
C     SO(1,-)   FLUE GAS FROM THE AUXILIARY BURNER ALONE
C     EN VECTOR IS 15 ELEMENTS LONG
C
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCSI/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSP,NCOUNT
      COMMON/PPD/NOPP,K,XMOLE(16),PPSC(19,16),PPMX(19)
      DO 10 J=6,JJ
      SO(1,J)=SI(2,J)-SI(1,J)
   10 CONTINUE
      SO(1,3)=SI(2,3)-SI(1,3)
      SO(1,4)=SI(2,4)
      SO(1,5)=SI(2,5)
      RETURN
      END
```

E.4 Miscellaneous Models

JUNCO1

PDRØP1 and CALFAC

SEPAO1

ABSO1

SETST2

CONVE

```
      SUBROUTINE JUNC01
      COMMON NS(12),ST(7,16),SO(7,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIN,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(150),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSR,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      SUBROUTINE JUNC01 ***MIXED-MASS AND/OR HEAT BALANCE***
      **APPOXIMATE HEAT BALANCE (THE OUTPUT TEMPERATURE IS CALCULATED)
      *ASSUMING THE HEAT COEFFICIENTS ARE CONSTANT WITHIN THE RANGE
      *BETWEEN THE INPUT TEMPERATURES AND THE OUT PUT TEMPERATURE

      *MAXIMUM VALUE OF NOCOMP, NUMBER OF FLOW COMPONENTS, IS 26
      *MAXIMM VALUE OF NIN, NUMBER OF INPUT STREAMS, IS 5
      *MAXIMUM VALUE OF NOUT, NUMBER OF OUTPUT STREAMS, IS 5


      EN VECTOR
      J - TEMPERATURE FOR STREAM OUT
      5 - PRESSURE FOR STREAM OUT
      16 - SWITCH IF 0, MASS BALANCE ONLY
                     1, HEAT AND MASS BALANCE
      17 - IF 1,THE MISTURES IS IN VAPOR PHASE=NOPP
              2, THE MISTURES IS IN LIQUID PHASE=NOPP
      18 - FRACTION OF THE TOTAL FLOW WHICH GOES TO THE FIRST OUTPUT STR
      19 - FRACTION OF THE TOTAL FLOW WHICH GOES TO THE SECOND OUTPUT ST
      20 - FRACTION OF THE TOTAL FLOW WHICH GOES TO THE 3 OUTPUT STREAM
      21 - FRACTION OF THE TOTAL FLOW WHICH GOES TO THE 4 OUTPUT STREAM
      22 - FRACTION OF THE TOTAL FLOW WHICH GOES TO THE 5 OUTPUT STREAM
      MASS BALANCE
      JJ=NOCOMP+5
      DO 1 J=6,JJ
      SO(NOUT,J)=0.
      DO 1 I=1,NIN
    1 SO(NOUT,J)=SO(NOUT,J)+SI(I,J)
      DO 2 I=1,NOUT
      SO(I,4)=EN(4)
      IF(EN(4).EQ.0.) SO(I,4)=SI(1,4)
      SO(I,5)=EN(5)
      IF(EN(5).EQ.0.) SO(I,5)=SI(1,5)
      SO(I,3)=0.
      DO 2 J=6,JJ
      SO(I,J)=SO(NOUT,J)*EN(I+17)
    2 SO(I,3)=SO(I,3)+SO(I,J)
      ISW=EN(16)+.001
      IF(ISW.EQ.0.) GO TO 26
      HEAT BALANCE

      IF(NIN.EQ.1.)GO TO 26
      CPTOT=0.
      ENTH=0.
      NOPP=EN(17)
      DO 21 I=1,NIN
      X=SI(I,4)
      DO 22 J=1,NOCOMP
   22 XMOLE(J)=SI(I,J+5)
      CALL PROPTX(1)
      CPTOT=CPTOT+PPMX(NOPP)
   21 ENTH=ENTH+PPMX(NOPP)*X
      TOUT=ENTH/CPTOT-273.16
      DO 23 I=1,NOUT
   23 SO(I,4)=TOUT
   24 CONTINUE
   26 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PDROP1(IISW)
C
C     EN(16)=FACP (CONSTANT FOR PRESSURE DROP CALCULATION)
C     IISW   = 1      IF  THE EQUIP. IS SELF STANDING
C     IISW   = 2      IF  PDROP1 IS CALLED BY ANOTHER SUBROUTINE
C     ICI = NUMBER OF INPUT STREAM
C     ICO = NUMBER OF OUTPUT STREAM
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(13),NOCOMP,JJ,IS,III,LOOP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(160),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSP,NCOUN
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      COMMON/P/ICI,ICO,FACP
C     COMMON/NAEJ/EQNU,ICC
      DIMENSION YMOLEI(14),YMOLEO(14),AMOL(11)
      DATA AMOL/18.,44.,28.,16.,2.,28.,32.,80.54,0.,0.,0./
      R=82.05
      GO TO (1,2),IISW
    1 TAVE=SI(1,4)+273.
      P=SI(1,5)
      FACP=EN(16)
      DO 20 J=3,JJ
      K=J-2
      YMOLEI(K)=SI(1,J)
      YMOLEO(K) = YMOLEI(K)
   20 CONTINUE
      DO 1000 J1=3,JJ
      SO(1,J1)=SI(1,J1)
 1000 CONTINUE
      SO(1,5)=EN(16)
      EQNU=NE
      ICC=1
      CALL CALFAC(FACP)
      GO TO 21
    2 DO 10 J2=3,JJ
      K=J2-2
      YMOLEI(K) = SI(ICI,J2)
      YMOLEO(K) = SO(ICO,J2)
   10 CONTINUE
      TAVE=(YMOLEI(2)+YMOLEO(2))/2.+273.
      P=YMOLEI(3)
   21 AMOLAV=0.
      DO 200 J3=6,JJ
      I=J3-2
      AMOLAV=AMOLAV+((YMOLEI(I)+YMOLEO(I))/(YMOLEI(1)+YMOLEO(1)))*AMOL(J
     *3-5)
  200 CONTINUE
      RHO=P/(R*TAVE)
      YMOLAV=(YMOLEI(1)+YMOLEO(1))/2.
      FACP=FACP*1.E-10
      DELP=(FACP*YMOLAV*YMOLAV*AMOLAV)/RHO
      P=P-DELP
      GO TO (3,4),IISW
    3 SO(1,5)=P
      RETURN
    4 SO(ICC,5) = P
      RETURN
      END
```

```
      SUBROUTINE CALFAC(FACP)
C
C     CALCULATES THE CONSTANT FACTOR USED IN THE PRESSURE
C                        DROP EQUATIONS
C
      COMMON NS(120),ST(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(2,16)
      COMMON/GEMCS1/LLST(120),EEN(150),NPOINT(100,2),EN(150),PE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSP,NCOUNT
      COMMON/NAEJ/EQNU,ICC
      DIMENSION YMOLEI(14),YMOLEO(14),AMOL(11)
      DATA AMOL/18.,44.,28.,16.,2.,28.,32.,80.54,0.,0.,0./
      P=P2.0E
      DELP=SI(ICC,5)-SO(ICC,5)
      DO 10  J=3,JJ
      K=J-2
      YMOLEI(K)=SI(ICC,J)
      YMOLEO(K)=SO(ICC,J)
   10 CONTINUE
      TAVE=(YMOLEI(2)+YMOLEO(2))/2.+273.
      P=YMOLEI(3)
   21 AMOLAV=0.
      DO 200  J3=6,JJ
      I=J3-2
      AMOLAV=AMOLAV+((YMOLEI(I)+YMOLEO(I))/(YMOLEI(1)+YMOLEO(1)))*AMOL(J
     *3-5)
  200 CONTINUE
      RHO=P/(R*TAVE)
      TMOLAV=(YMOLEI(1)+YMOLEO(1))/2.
      FACP=(DELP*RHO)/(TMOLAV*TMOLAV*AMOLAV)
      FACP=FACP*1.E+10
      PRINT 500,EQNU,ICC,FACP
  500 FORMAT(1X,*EQNU*,F5.1,10X,*FLOW NUMBER =*,I4,10X,*FACP=*,E15.9///)
      RETURN
      END
```

```
      SUBROUTINE SEPAC
C     SPLITS THE FLOW TO A SEPARATOR TO A FIXED FLOW
C     OUT AT THE TOP
C
C     STREAM CODE LIST
C     SI(1,-) = FLOW IN
C     SO(1,-)=FLOW OUT AT THE TOP
C     SO(2,-)=FLOW OUT AT THE BOTTOMS
C
C     EQUIPMENT CODE LIST
C     EN(4)=PRESSURE AT THE TOP
C     EN(5)=PRESSURE AT THE BOTTOM
C     EN(16)=FRACTION OF STEAM FROM FEED AT THE TOP
C     EN(17)=FRACTION OF CO2   FROM FEED AT THE TOP
C     EN(18)=FRACTION OF CO    FROM FEED AT THE TOP
C     EN(19)=FRACTION OF CH4   FROM FEED AT THE TOP
C     EN(20)=FRACTION OF H2    FROM FEED AT THE TOP
C     EN(21)=FRACTION OF N2    FROM FEED AT THE TOP
C     EN(22)=FRACTION OF O2    FROM FEED AT THE TOP
C     EN(23)=FRACTION OF NAPHTHA VAPOR FROM FEED AT THE TOP
C     EN(24)=FRACTION OF WATER FROM FEED AT THE TOP
C     EN(25)=FRACTION OF FUEL  FROM FEED AT THE TOP
C     EN(26)=FRACTION OF NAPHTHA LIQUID FROM FEED AT THE TOP
C     EN(27)=PRESSURE AT THE SEPARATOR
C     EN(28)=TOP TEMPERATURE
C     EN(29)=BOTTOMS TEMPERATURE
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSR,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      SO(1,3)=0.
      SO(2,3)=0.
      DO 510 J=6,JJ
      SO(1,J)= SI(1,J)*EN(10+J)
      SO(2,J)=SI(1,J)-SO(1,J)
      SO(2,3)=SO(2,3)+SO(2,J)
      SO(1,3)=SO(1,3)+SO(1,J)
  510 CONTINUE
C     CALCULATE P AND T AT THE EXIT
      SO(1,4)=EN(28)
      SO(2,4)=EN(29)
C
C     IF SET TO ZERO TAKE THE VALUE OF THE INLET STREAM
      IF(INT(EN(28)).EQ.0)SO(1,4)=SI(1,4)
      IF(INT(EN(29)).EQ.0)SO(2,4)=SI(1,4)
      SO(1,5)=EN(4)
      SO(2,5)=EN(5)
      IF (EN(4).EQ.0.) SO(1,5)=SI(1,5)
      IF (EN(5).EQ.0.) SO(2,5)=SI(1,5)
      RETURN
      END
```

```
      SUBROUTINE ABS01
C
C      SIMULATES THE ABSORBER, ... A MODIFICATION OF SEPA01 ...
C      AS FOR SEPA01,  ABS01 SPLITS THE FLOW INTO A SEPARATOR
C     TO A FIXED FLOW OUT AT THE TOP.
C      ABS01 ALSO DETERMINES THE AMOUNT OF WATER WHICH MUST BE LOST
C     (CONDENSED) IN THE ABSORBER GIVEN THE GAS TEMPERATURE AT THE
C     EXIT IS THE ABSORBER TEMPERATURE AND IT IS SATURATED WITH
C     WATER VAPOUR.
C
C     STREAM CODE LIST
C     SI(1,-) = FLOW IN
C     SO(1,-)=FLOW OUT AT THE TOP **  MAIN GAS STREAM
C     SO(2,-)=FLOW OUT AT THE BOTTOMS
C     SO(2,-)=FLOW OUT AT THE BOTTOMS  **  MAINLY LIQUID (CONDENSATION)
C
C     EQUIPMENT CODE LIST
C     EN(4)=PRESSURE AT THE TOP (NOT USED HERE, EXIT PRESSURE IS
C                                CALCULATED  BY CALLING PDROP1
C     EN(5)=PRESSURE AT THE BOTTOM
C     EN(16)=FRACTION OF STEAM FROM FEED AT THE TOP
C     EN(17)=FRACTION OF CO2    FROM FEED AT THE TOP
C     EN(18)=FRACTION OF CO    FROM FEED AT THE TOP
C     EN(19)=FRACTION OF CH4    FROM FEED AT THE TOP
C     EN(20)=FRACTION OF H2     FROM FEED AT THE TOP
C     EN(21)=FRACTION OF N2     FROM FEED AT THE TOP
C     EN(22)=FRACTION OF O2     FROM FEED AT THE TOP
C     EN(23)=FRACTION OF NAPHTHA VAPOR FROM FEED AT THE TOP
C     EN(24)=FRACTION OF WATER FROM FEED AT THE TOP
C     EN(25)=FRACTION OF FUEL  FROM FEED AT THE TOP
C     EN(26)=FRACTION OF NAPHTHA LIQUID FROM FEED AT THE TOP
C     EN(27)=PRESSURE AT THE SEPARATOR
C     EN(28)=TOP TEMPERATURE
C     EN(29)=BOTTOMS TEMPERATURE
C     EN(30)=FACP USED FOR PRESSURE DROP CALCULATION
C
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(72,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(100,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSR,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
      COMMON/P/ICI,ICO,FACP
C     COMMON/NAEJ/EQNU,ICC
      SO(1,3)=0.
      SO(2,3)=0.
      DO 510 J=7,JJ
      SO(1,J)= SI(1,J)*EN(10+J)
      SO(2,J)=SI(1,J)-SO(1,J)
      SO(2,3)=SO(2,3)+SO(2,J)
      SO(1,3)=SO(1,3)+SO(1,J)
  510 CONTINUE
C     CALCULATE P AND T AT THE EXIT
      SO(1,4)=EN(28)
      SO(2,4)=EN(29)
C
C     IF SET TO ZERO TAKE THE VALUE OF THE INLET STREAM
      IF(INT(EN(28)).EQ.0)SO(1,4)=SI(1,4)
      IF(INT(EN(29)).EQ.0)SO(2,4)=SI(1,4)
C
C      PRESSURE DROP CALCULATION
C
      SO(1,5)=EN(4)
      IF (EN(4).EQ.0.) SO(1,5)=SI(1,5)
      SO(1,5)=EN(4)
      EQNU=NE
      ICC=1
C     CALL CALFAC(FACP)
      FACP=EN(30)
      ICI=ICO=1
      CALL PDROP1(2)
      SO(2,5)=EN(5)
      IF (EN(5).EQ.0.) SO(2,5)=SI(1,5)
      TO=SO(1,4)
      Z=7.96681-1668.21/(228.0+TO)
      VPM=(10.**Z)/760.
```

```
      DEP=SO(1,5)-VPW
      IF(DEP.LT.0.) GO TO 10
      XMVO=(SI(1,3)-SI(1,6))*VPW/DEP
      COND=SI(1,6)-XMVO
      IF(COND.LT.0.) GO TO 20
      SO(1,6)=SI(1,6)-COND
      SO(1,3)=SO(1,3)+SO(1,6)
      SO(2,14)=COND
      SO(2,3)=SO(2,3)+COND
      GO TO 100
   10 CONTINUE
      WRITE(6,998) NE
  998 FORMAT(/,10X,≠A SITUATION EXISTS IN  ABS01, EQUIPMENT NO. ≠,I3,≠ I
     1N WHICH THE VAPOUR PRESSURE≠,/,10X,≠OF WATER IN THE OVERHEAD PRODU
     2CT STREAM EXCEEDS THE SYSTEM PRESSURE≠,/)
      COND=0.
      SO(1,6)=0.
      GO TO 100
   20 CONTINUE
      WRITE(6,999) NE
  999 FORMAT(/,10X,≠THERE IS INSUFFICIENT WATER IN THE GAS STREAM LEAVIN
     1G THE OVERHEAD OF  ABS01,≠,/,10X,≠EQUIPMENT NO. ≠,I3,≠ TO ENSURE S
     2ATURATION AT THE DESIRED OUTLET TEMPERATURE≠,/)
      COND=0.
      SO(1,6)=0.
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SETST2
C
C  ** EN(16) = TEMPERATURE RISE OF STREAM **
C
      COMMON NS(120),SI(4,16),SO(4,16),KPRNT(10),NOCOMP,JJ,IS,III,LOCP,
     1       NLP,NIT,SN(2,16)
      COMMON/GEMCS1/LLST(120),EEN(1500),NPOINT(150,2),EN(150),NE,NIN,
     1              NOUT,MSN,MODE,NPLNT,ISP,NC,NCALC,NSP,NCOUNT
      COMMON/PPD/NOPP,X,XMOLE(16),PPSC(19,16),PPMX(19)
C
      DO 100 I=3,JJ
  100 SO(4,I)=SI(1,I)
      SO(1,4)=SI(1,4)+EN(16)
      RETURN
      END



      SUBROUTINE CONVE(X,XC,ICX,XA)
      DIMENSION XA(2)
      IF(ABS((X-XC)/(X+XC)) - 1.E-4) 6,7,7
    6 ICX=1
      RETURN
    7 IF(ICX-1) 5,5,8
    8 XT=(X+XC)/2.
      GO TO 4
    5 XT=(XA(1)*XC-XA(2)*X)/(XA(1)+XC-XA(2)-X)
    4 XA(1)=X
      XA(2)=XC
      X=XT
      ICX=2
      RETURN
      END



      FUNCTION TS(P)
      TS=262.7
      RETURN
      END



      FUNCTION HLAM(T)
      HLAM=7081.
      RETURN
      END
```

APPENDIX F

GEMCS DATA SET AND CONVERGED SOLUTION

OF THE COMBINED EXECUTIVE

F.1 GEMCS Data Set

```
          CASE     1


RUN NUMBER  100

NAPHTHA REFORMER PLANT ** TOTAL PLANT SIMULATION **
COMBINED GEMCS-SYMBOL EXECUTIVE  -  100 PERCENT PRODUCTION

PRINTING NCONT,LLST,NS

PRINTING INITIAL STREAMS

PRINTING MODULE SETS

   46 MODULE SETS    11 COMPONENTS

CALCULATION ORDER LIST-
     8     9     5    10    11    12    13    14    33    15
    34    16    17    18    19    20    21    22    23    24
    25    26    27    28    30    13    29    30    31     5
     4    36     6    35     3     2     1     7    37    41
    40    42    43    44    38    39

   84STREAM CODES ARE READ-
     1     2     2     2     2     7     1     2     4     4
     7     2     7     2     2     7     4     1     7     7
     4     3     7     2     2     2     7     4     3     7
     1     3     4     3     1     4     3     4     7     1
     3     2     3     2     7     2     7     2     7     3
     1     3     1     3     1     3     1     3     2     1
     1     2     1     1     3     7     3     3     3     3
     1     1     2     2     2     7     7     3     1     2
     2     3     3     3

   F3INITIAL STREAMS ARE READ-
         1.00000          0.00000        636.30616        111.10000         48.39200
         0.00000          0.00000          0.00000          0.00000          0.00000
       502.68187        133.62429          0.00000          0.00000          0.00000
         0.00000

----
         2.00000          0.00000        636.30616        315.00000         48.39200
         0.00000          0.00000          0.00000          0.00000          0.00000
       502.68187        133.62429          0.00000          0.00000          0.00000
         0.00000

----
         3.00000          1.00000         16.39373        105.00000         49.34616
         0.00000          0.00000          0.00000          0.00000          0.00000
         0.00000          0.00000          0.00000          0.00000          0.00000
        16.39373

----
         4.00000         11.00000          0.00000          0.00000          0.00000
         0.00000          0.00000          0.00000          0.00000          0.00000
         0.00000          0.00000          0.00000          0.00000          0.00000
         0.00000

----
         5.00000         13.00000         25.13572        406.00000         47.37232
          .06420          0.00000          0.00000          .15572          8.53104
         0.00000          0.00000         16.38476          0.00000          0.00000
         0.00000

----
```

7.00000	2.00000	0.00000	262.70000	48.37200
1.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

8.00000	14.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

9.00000	15.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

10.00000	16.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

12.00000	17.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

17.00000	18.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

18.00000	0.00000	0.00000	20.00000	1.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

21.00000	21.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

22.00000	20.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

25.00000	0.00000	0.00000	110.00000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

26.00000	0.00000	1.00000	194.00000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

28.00000	24.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

29.00000	23.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

31.00000	0.00000	1.00000	110.00000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	1.00000	0.00000
0.00000				
33.00000	27.00000	0.00000	220.00000	40.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	1.00000	0.00000
0.00000				
34.00000	26.00000	0.00000	220.00000	40.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	1.00000	0.00000
0.00000				
35.00000	4.00000	.03600	220.00000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
.02800	.00800	0.00000	1.00000	0.00000
0.00000				
36.00000	28.00000	0.00000	110.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				
37.00000	30.00000	0.00000	110.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				
38.00000	31.00000	0.00000	110.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				
40.00000	0.00000	178.82020	127.00000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	178.82020	0.00000
0.00000				
43.00000	33.00000	0.00000	160.00000	40.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				
44.00000	34.00000	287.07589	307.46691	33.87440
.36563	.14019	1.44018	3.45000	281.67991
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				
46.00000	35.00000	284.12193	342.00000	33.77762
2.08428	0.00000	0.00000	5.05572	276.98193
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				
48.00000	12.00000	8.75142	298.20000	33.77747
.06460	0.00000	0.00000	0.15491	8.53104
.00086	0.00000	0.00000	0.00000	0.00000
0.00000				
49.00000	36.00000	0.00000	160.00000	40.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

51.00000	0.00000	1.00000	225.00000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	1.00000	0.00000
0.00000				

53.00000	0.00000	1.00000	225.00000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	1.00000	0.00000
0.00000				

55.00000	0.00000	1.00000	225.00000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	1.00000	0.00000
0.00000				

57.00000	0.00000	1.00000	225.00000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	1.00000	0.00000
0.00000				

60.00000	0.00000	1.00000	110.00000	1.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	1.00000
0.00000				

61.00000	0.00000	1.00000	110.00000	1.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	1.00000
0.00000				

62.00000	6.00000	1.00000	110.00000	1.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	1.00000
0.00000				

63.00000	7.00000	0.00000	110.00000	1.00000
1.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

64.00000	0.00000	1.00000	164.70000	48.39200
1.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

67.00000	0.00000	1.00000	111.10000	1.00000
0.00000	0.00000	0.00000	0.00000	0.00000
.79000	.21000	0.00000	0.00000	0.00000
1.00000				

68.00000	0.00000	1.00000	164.70000	48.39200
1.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

69.00000	3.00000	0.00000	262.70000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	1.00000	0.00000
0.00000				

70.00000	10.00000	0.00000	164.70000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	1.00000	0.00000
0.00000				

71.00000	0.00000	0.00000	110.00000	1.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	1.00000
0.00000				

72.00000	0.00000	0.00000	315.00000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
.79000	.21000	0.00000	0.00000	0.00000
0.00000				

73.00000	38.00000	731.43836	830.23302	48.39200
150.36120	78.33082	0.00000	0.00000	0.00000
481.41771	21.32663	0.00000	0.00000	0.00000
0.00000				

74.00000	41.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

79.00000	0.00000	1.00000	29.00000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	1.00000	0.00000
0.00000				

81.00000	5.00000	1.00000	315.00000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
.79000	.21000	0.00000	0.00000	0.00000
0.00000				

82.00000	37.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

83.00000	39.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

84.00000	40.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

43MODULE SETS ARE READ-

1.00000	2.00000	25.00000	0.00000	0.00000
2.00000	1.00000	77.00000	0.00000	0.00000
2.00000	2.00000	78.00000	0.00000	0.00000
8711.00000	1.00000	1.00000	1.00000	1.00000
315.00000	244.00000	0.00000	0.00000	0.00000

2.00000	19.00000	21.00000	0.00000	0.00000
2.00000	76.00000	3.00000	0.00000	0.00000
2.00000	77.00000	4.00000	0.00000	0.00000
2275.00000	399.70000	406.00000	300.00000	370.00000
6897.31000				

3.00000	2.00000	25.00000	0.00000	0.00000
2.00000	48.00000	75.00000	0.00000	0.00000
2.00000	59.00000	76.00000	0.00000	0.00000
198.33000	1.00000	1.00000	1.00000	1.00000
430.20000	437.40000	0.00000	0.00000	0.00000

4.00000	2.00000	25.00000	0.00000	0.00000
2.00000	74.00000	55.00000	0.00000	0.00000
2.00000	75.00000	56.00000	0.00000	0.00000
3648.00000	8.00000	1.00000	1.00000	1.00000
440.00000	262.70000	.05300	0.00000	0.00000

5.00000	16.00000	30.00000	0.00000	0.00000
4.00000	8.00000	73.00000	71.00000	72.00000
4.00000	9.00000	74.00000	70.00000	69.00000
8.30000	.15000	9600.00000	25.00000	.08900
1600.00000	50.00000	518.00000	273.00000	612.60000
875.00000	520.00000	.33333	7722.00000	.59346

5.00000	15.00000	33.00000	0.00000	0.00000
4.00000	64.00000	57.00000	66.00000	61.00000
4.00000	65.00000	58.00000	67.00000	68.00000
1500.00000	8.00000	1.00000	1.00000	2.00000
297.00000	262.70000	.05300	0.00000	0.00000
.15000	1.59800	0.00000	9800.00000	7.63600
60.00000	1100.00000	0.00000		

7.00000	1.00000	22.00000	0.00000	47.37232
2.00000	59.00000	4.00000	0.00000	0.00000
1.00000	5.00000	0.00000	0.00000	0.00000
1.00000	1.00000	1.00000	0.00000	0.00000
0.00000	0.00000			

8.00000	5.00000	16.00000	0.00000	0.00000
1.00000	5.00000	0.00000	0.00000	0.00000
1.00000	6.00000	0.00000	0.00000	0.00000
1176.06284				

9.00000	10.00000	24.00000	0.00000	42.71260
2.00000	6.00000	7.00000	0.00000	0.00000
1.00000	8.00000	0.00000	0.00000	0.00000
1.00000	1.00000	1.00000	0.00000	0.00000
0.00000	0.00000	9.12000	5.64880	

10.00000	5.00000	41.00000	0.00000	0.00000
2.00000	9.00000	63.00000	0.00000	0.00000
4.00000	10.00000	73.00000	81.00000	62.00000
810.00000	-58.47582	8.30000	.20000	.08900
910.00000	-35550.00000	-26416.00000	-57797.00000	-17889.00000
-9828.45000	25.00000	57702.00000	173106.00000	9800.00000
1100.00000	5.64880	2.26000	86553.00000	-5.14023
-94052.00000	-6831[illegible].00000	.10000	125.00000	.96141
2.80674				

11.00000	2.00000	25.00000	0.00000	0.00000
2.00000	10.00000	53.00000	0.00000	0.00000
2.00000	11.00000	54.00000	0.00000	0.00000
11849.40000	8.00000	1.00000	1.00000	2.00000
436.00000	262.70000	.05300	1.00000	.32557

12.00000	6.00000	23.00000	0.00000	0.00000
1.00000	11.00000	0.00000	0.00000	0.00000
1.00000	12.00000	0.00000	0.00000	0.00000
-.03025	260.00000	-9447.00000	51552.00000	442.00000
-105.79366	.43544	5.44051		

13.00000	2.00000	25.00000	0.00000	0.00000
2.00000	12.00000	44.00000	0.00000	0.00000
2.00000	13.00000	45.00000	0.00000	0.00000
197.92500	1.00000	1.00000	1.00000	1.00000
435.00000	320.00000	0.00000	0.00000	0.00000

14.00000	2.00000	25.00000	0.00000	0.00000
2.00000	13.00000	51.00000	0.00000	0.00000
2.00000	14.00000	52.00000	0.00000	0.00000
13702.80000	8.00000	0.00000	1.00000	2.00000
303.00000	262.70000	0.00000	0.00000	0.00000

15.00000	2.00000	25.00000	0.00000	0.00000
2.00000	14.00000	26.00000	0.00000	0.00000
2.00000	15.00000	27.00000	0.00000	0.00000
7660.62000	1.00000	1.00000	1.00000	2.00000
225.00000	247.00000	0.00000	0.00000	0.00000

16.00000	8.00000	16.00000	0.00000	0.00000
1.00000	15.00000	0.00000	0.00000	0.00000
1.00000	16.00000	0.00000	0.00000	0.00000
.57667				

17.00000	6.00000	23.00000	0.00000	0.00000
1.00000	16.00000	0.00000	0.00000	0.00000
1.00000	17.00000	0.00000	0.00000	0.00000
.05609	260.00000	-9447.00000	51552.00000	255.00000
-91.60266	.55579	.18384		

18.00000	1.00000	22.00000	0.00000	0.00000
2.00000	17.00000	18.00000	0.00000	0.00000
1.00000	19.00000	0.00000	0.00000	0.00000
1.00000	1.00000	1.00000	0.00000	0.00000
0.00000	0.00000			

19.00000	3.00000	25.00000	0.00000	0.00000
2.00000	19.00000	25.00000	0.00000	0.00000
2.00000	20.00000	26.00000	0.00000	0.00000
16850.46400	1.00000	1.00000	1.00000	150.00000
2.00000	.19250	0.00000	214.00000	.03501

20.00000	4.00000	29.00000	0.00000	0.00000
1.00000	20.00000	0.00000	0.00000	0.00000
2.00000	21.00000	22.00000	0.00000	0.00000
1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	0.00000	0.00000
0.00000	34.16154	0.00000	0.00000	

21.00000	3.00000	25.00000	0.00000	0.00000
2.00000	21.00000	24.00000	0.00000	0.00000
2.00000	23.00000	60.00000	0.00000	0.00000
8369.09200	1.00000	1.00000	1.00000	103.00000
2.00000	.19520	0.00000	207.00000	.04991

22.00000	4.00000	29.00000	0.00000	0.00000
1.00000	23.00000	0.00000	0.00000	0.00000
2.00000	28.00000	29.00000	0.00000	0.00000
1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	0.00000	0.00000
0.00000	34.06154	0.00000	0.00000	

23.00000	3.00000	25.00000	0.00000	0.00000
2.00000	28.00000	31.00000	0.00000	0.00000
2.00000	30.00000	32.00000	0.00000	0.00000
130541.00000	2.00000	1.00000	1.00000	110.00000
2.00000	.93200	0.00000	116.00000	.12568

24.00000	4.00000	29.00000	34.06797	34.06797
1.00000	30.00000	0.00000	0.00000	0.00000
2.00000	33.00000	34.00000	0.00000	0.00000
1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	0.00000	0.00000
0.00000	34.06797	0.00000	0.00000	

25.00000	1.00000	22.00000	0.00000	0.00000
2.00000	33.00000	35.00000	0.00000	0.00000
1.00000	36.00000	0.00000	0.00000	0.00000
1.00000	1.00000	1.00000	0.00000	0.00000
0.00000	0.00000			

26.00000	20.00000	30.00000	0.00000	33.92279
1.00000	36.00000	0.00000	0.00000	0.00000
2.00000	38.00000	37.00000	0.00000	0.00000
0.00000	.00161	.84448	.69383	1.00000
1.00000	0.00000	0.00000	0.00000	0.00000
0.00000	34.06797	68.00000	68.00000	1.75460

27.00000	3.00000	25.00000	0.00000	0.00000
2.00000	38.00000	40.00000	0.00000	0.00000
2.00000	39.00000	41.00000	0.00000	0.00000
17155.60000	1.00000	1.00000	1.00000	59.00000
2.00000	.84100	0.00000	31.00000	3.08300

28.00000	4.00000	29.00000	33.87440	33.87440
1.00000	39.00000	0.00000	0.00000	0.00000
2.00000	42.00000	43.00000	0.00000	0.00000
1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	0.00000	0.00000
0.00000	33.87440	0.00000	0.00000	

29.00000	7.00000	23.00000	0.00000	0.00000
1.00000	45.00000	0.00000	0.00000	0.00000
1.00000	46.00000	0.00000	0.00000	0.00000
51533.00000	42106.00000	260.00000	3.45009	.99000
342.00000	.99000	17.55000		

30.00000	2.00000	25.00000	0.00000	0.00000
2.00000	46.00000	42.00000	0.00000	0.00000
2.00000	47.00000	44.00000	0.00000	0.00000
16505.40000	1.00000	1.00000	1.00000	1.00000
307.00000	64.00000	0.00000	0.00000	0.00000

31.00000	1.00000	22.00000	98.20000	0.00000
1.00000	47.00000	0.00000	0.00000	0.00000
2.00000	48.00000	48.00000	0.00000	0.00000
0.00000	1.00000	.96920	.03080	0.00000
0.00000	0.00000			

33.00000	13.00000	16.00000	0.00000	0.00000
2.00000	7.00000	79.00000	0.00000	0.00000
3.00000	24.00000	25.00000	25.00000	0.00000
.05300				

```
 34.00000    17.00000    15.00000     0.00000     0.00000
  1.00000    27.00000     0.00000     0.00000     0.00000
  4.00000    51.00000    53.00000    55.00000    57.00000

----

 35.00000    11.00000    17.00000     0.00000     0.00000
  4.00000    62.00000    71.00000    61.00000    72.00000
  2.00000    60.00000     1.00000     0.00000     0.00000
 37.00000    41.00000

----

 36.00000    14.00000    15.00000     0.00000     0.00000
  4.00000     7.00000    56.00000    52.00000    54.00000
  1.00000    66.00000     0.00000     0.00000     0.00000

----

 37.00000     1.00000    19.00000     0.00000     0.00000
  1.00000    60.00000     0.00000     0.00000     0.00000
  2.00000    62.00000    71.00000     0.00000     0.00000
  0.00000     1.00000      .90000      .10000

----

 38.00000    18.00000    24.00000     0.00000     0.00000
  4.00000     5.00000    73.00000    44.00000    25.00000
  2.00000    26.00000     2.00000     0.00000     0.00000
  0.00000     4.00000     7.00000      .05000     0.00000
  1.00000   100.00000     1.00000     0.00000

----

 39.00000     9.00000   105.00000     0.00000     5.00000
  4.00000     5.00000    73.00000    44.00000    25.00000
  2.00000    26.00000     2.00000     0.00000     0.00000
 45.00000    30.00000     0.00000      .00010     1.00000
  1.00000     0.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     0.00000     0.00000     0.00000

----

 40.00000    21.00000    16.00000     0.00000     0.00000
  1.00000    60.00000     0.00000     0.00000     0.00000
  1.00000    75.00000     0.00000     0.00000     0.00000
  7.00000

----

 41.00000     1.00000    19.00000     0.00000     0.00000
  1.00000     2.00000     0.00000     0.00000     0.00000
  2.00000    81.00000    72.00000     0.00000     0.00000
  0.00000     1.00000      .90000      .10000

----

 42.00000     1.00000    18.00000     1.00000     1.00000
  3.00000    81.00000    62.00000    63.00000     0.00000
  1.00000    82.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     1.00000

----

 43.00000     1.00000    18.00000     1.00000     1.00000
  3.00000    72.00000    71.00000    70.00000     0.00000
  1.00000    83.00000     0.00000     0.00000     0.00000
  0.00000     0.00000     1.00000

----

 44.00000    22.00000    15.00000     0.00000     0.00000
  2.00000    73.00000    74.00000     0.00000     0.00000
  1.00000    84.00000     0.00000     0.00000     0.00000

----
```

STREAM LIST - NAPHTHA SIMULATION

---

1. STREAM NUMBER
2. STREAM TYPE
3. TOTAL FLOW
4. TEMPERATURE DEGREES C
5. PRESSURE, ATM.

COMPONENT FLOWS, KGM.-MOLES/HR.

GASEOUS COMPONENTS

6. WATER
7. CARBON DIOXIDE
8. CARBON MONOXIDE
9. METHANE
10. HYDROGEN
11. NITROGEN
12. OXYGEN
13. NAPHTHA

LIQUID COMPONENTS

14. WATER
15. FUEL (KGM./HR.)
16. NAPHTHA

PHYSICAL PROPERTIES
READING 2PHYSICAL PROPERTY SETS
FOR THE 11 COMPONENTS, THE PARAMETERS FOR NOPP= 1 ARE

H_2O(v)	7.260736380	.001976843	.000001135	.000000001
CO_2	5.375334100	.014530190	-.000009050	.000000002
CO	6.732514800	-.000046100	.000002216	-.000000001
CH_4	3.359595300	.016991809	-.000002134	-.000000001
H_2	7.091221500	-.000640007	.000000887	0.000000000
N_2	6.979502570	-.001017071	.000003056	-.000000001
O_2	5.846819580	.004126210	-.000001979	0.000000000
Naphtha(v)	28.045000000	0.000000000	0.000000000	0.000000000
	0.000000000	0.000000000	0.000000000	0.000000000
	0.000000000	0.000000000	0.000000000	0.000000000
	0.000000000	0.000000000	0.000000000	0.000000000

FOR THE 11 COMPONENTS, THE PARAMETERS FOR NOPP= 2 ARE

	0.000
	0.000
	0.000
	0.000
	0.000
	0.000
	0.000
	0.000
H_2O(ℓ)	18.000
Fuel (ℓ)	.500
Naphtha(ℓ)	35.500

NOPP = 1 means Gaseous state

NOPP = 2 means Liquid state

The SYMBØL part of the data set is given in Table 5.2

F.2 GEMCS Converged solution

```
END OF LOOP   13
TEST ON STREAM,     5 CONVERGED FOR ALL COMPONENTS ON LOOP,    13
TEST ON STREAM,    73 CONVERGED FOR ALL COMPONENTS ON LOOP,    13
TEST ON STREAM,    44 CONVERGED FOR ALL COMPONENTS ON LOOP,    13
TEST ON STREAM,    25 CONVERGED FOR ALL COMPONENTS ON LOOP,    13
TEST ON STREAM,    26 CONVERGED FOR ALL COMPONENTS ON LOOP,    13
TEST ON STREAM,     2 CONVERGED FOR ALL COMPONENTS ON LOOP,    13

FINAL OUTPUT STREAMS FOR MODULE    39
        26.00000          0.00000        889.31821        194.01299         48.39200
         0.00000          0.00000          0.00000          0.00000          0.00000
         0.00000          0.00000          0.00000        889.31821          0.00000
          .00000

----
         2.00000          0.00000        636.53862        315.33064         48.39200
         0.00000          0.00000          0.00000          0.00000          0.00000
       502.86551        133.67311          0.00000          0.00000          0.00000
          .00000

----

FINAL OUTPUT STREAMS FOR MODULE     8
         6.00000          0.00000         25.14518        409.10399         42.66634
          .06461          .00000          .00000          .15491          8.53165
          .00089          .00000        16.39373          .00000           .00000
          .00000

----
THE FLOW OF STEAM REACTANT TO THE PROCESS IS  844.5567

FINAL OUTPUT STREAMS FOR MODULE     9
         8.00000         14.00000        869.70189        272.32167         42.71260
       844.62131          .00000          .00000          .15491          8.53105
          .00089          .00000        16.39373          .00000           .00000
          .00000

----

FINAL OUTPUT STREAMS FOR MODULE     5
         9.00000         15.00000        869.70189        518.24308         41.61646
       844.62131          .00000          .00000          .15491          8.53105
          .00089          .00000        16.39373          .00000           .00000
          .00000

----
        74.00000         41.00000        754.25634        579.39401         48.39200
       157.35741         81.97551          .00000          .00000           .00000
       502.86551         22.06790          .00000          .00000           .00000
          .00000

----
        70.00000         10.00000          4.45000        164.70000         48.39200
         4.45000          0.00000          0.00000          0.00000          0.00000
         0.00000          0.00000          0.00000          0.00000          0.00000
         0.00000

----
        69.00000          3.00000          0.00000        262.70000         48.39200
         0.00000          0.00000          0.00000          0.00000          0.00000
         0.00000          0.00000          0.00000          1.00000          0.00000
         0.00000

----
```

FINAL OUTPUT STREAMS FOR MODULE 10

10.00000	16.00000	1042.93418	810.00000	38.80625
687.85488	67.97274	20.82095	3.96612	262.31660
.00089	0.00000	0.00000	0.00000	0.00000
0.00000				

73.00000	38.00000	730.74781	830.24631	48.39200
150.21924	78.25687	.00000	.00000	.00000
480.96320	21.30850	.00000	.00000	.00000
.00000				

81.00000	5.00000	608.81418	315.33064	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
480.96320	127.85098	0.00000	0.00000	0.00000
0.00000				

62.00000	6.00000	1052.22486	110.00000	1.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	1052.22486
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 11

11.00000	0.00000	1042.93418	436.14641	37.74209
687.85488	67.97274	20.82095	3.96612	262.31660
.00089	0.00000	0.00000	0.00000	0.00000
0.00000				

54.00000	0.00000	500.30619	262.70000	48.39201
500.30619	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 12

12.00000	17.00000	1042.69427	442.16397	36.57894
676.68979	79.25778	9.41595	4.08608	273.24377
.00089	0.00000	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 13

13.00000	0.00000	1042.69427	439.50533	36.57894
676.68979	79.25778	9.41595	4.08608	273.24377
.00089	0.00000	0.00000	0.00000	0.00000
0.00000				

45.00000	0.00000	287.30261	319.22729	33.87440
.37494	.14038	1.44186	3.44739	281.66915
.02889	.00000	.00000	.00000	.00000
.00000				

FINAL OUTPUT STREAMS FOR MODULE 14

14.00000	0.00000	1042.69427	302.80356	36.57894
676.68979	79.25778	9.41595	4.08608	273.24377
.00089	0.00000	0.00000	0.00000	0.00000
0.00000				

52.00000	0.00000	164.53286	262.70000	48.39200
164.53286	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 33

24.00000	0.00000	889.31821	23.00000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	889.31821	0.00000
0.00000				

25.00000	0.00000	889.31821	110.04363	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	889.31821	0.00000
0.00000				

26.00000	0.00000	889.31821	194.01299	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	889.31821	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 15

15.00000	0.00000	1042.69427	247.06840	36.57894
676.68979	79.25778	9.41595	4.08608	273.24377
.00089	0.00000	0.00000	0.00000	0.00000
0.00000				

27.00000	0.00000	889.31821	224.95694	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	889.31821	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 34

51.00000	0.00000	164.53286	224.95694	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	164.53286	0.00000
0.00000				

53.00000	0.00000	526.82242	224.95694	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	526.82242	0.00000
0.00000				

55.00000	0.00000	121.08595	224.95694	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	121.08595	0.00000
0.00000				

57.00000	0.00000	68.15770	224.95694	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	68.15770	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 16

15.00000	0.00000	1042.69427	247.06840	35.41852
676.68979	79.25778	9.41595	4.08608	273.24377
.00089	0.00000	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 17

17.00000	18.00000	1043.15268	254.06619	34.25556
663.52283	87.19555	1.79739	3.85687	291.86915
.00089	0.00000	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 18

19.00000	0.00000	1043.15268	254.06619	34.25556
668.52283	87.19555	1.70739	3.85687	281.86915
.00089	0.00000	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 19

20.00000	0.00000	1043.15268	213.58523	34.19103
547.18209	87.19555	1.70739	3.85687	281.86915
.00089	0.00000	0.00000	121.34074	0.00000
0.00000				

26.00000	0.00000	889.31821	194.01300	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	889.31821	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 20

21.00000	21.00000	921.81194	213.58523	34.19103
547.18209	87.19555	1.70739	3.85687	281.86915
.00089	0.00000	0.00000	0.00000	0.00000
0.00000				

22.00000	20.00000	121.34074	213.58523	34.19103
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	121.34074	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 21

23.00000	0.00000	921.81194	207.00000	34.12636
408.02923	87.19555	1.70739	3.85687	281.86915
.00089	0.00000	0.00000	139.15286	0.00000
0.00000				

80.00000	0.00000	889.31821	103.04363	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	889.31821	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 22

28.00000	24.00000	782.65908	207.00000	34.12638
408.02923	87.19555	1.70739	3.85687	281.86915
.00089	0.00000	0.00000	0.00000	0.00000
0.00000				

29.00000	23.00000	139.15286	207.00000	34.12638
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	139.15285	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 23

30.00000	0.00000	782.65908	116.00000	34.06176
19.93310	87.19555	1.70739	3.85687	281.86915
.00089	0.00000	0.00000	388.09613	0.00000
0.00000				

32.00000	0.00000	1.00000	110.00000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	1.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 24

33.00000	27.00000	394.55295	116.00000	34.06797
19.93310	87.19555	1.70739	3.85687	281.86915
.00889	0.00000	0.00000	0.00000	0.00000
0.00000				

34.00000	26.00000	388.09613	116.00000	34.06797
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	388.09613	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 25

36.00000	28.00000	394.59895	115.99212	34.06797
19.93310	87.19555	1.70739	3.85687	281.86915
.02889	.00800	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 26

38.00000	31.00000	290.05783	68.00000	33.92100
3.14016	.14038	1.44186	3.44739	281.86915
.02889	0.00000	0.00000	0.00000	0.00000
0.00000				

37.00000	30.00000	104.53112	68.00000	33.92279
0.00000	87.05516	.26553	.40948	0.00000
0.00000	.00800	0.00000	16.79294	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 27

39.00000	0.00000	290.05783	31.04245	33.87254
.37494	.14038	1.44186	3.44739	281.86915
.02889	0.00000	0.00000	2.76523	0.00000
0.00000				

41.00000	0.00000	178.82020	59.32831	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	178.82020	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 28

42.00000	0.00000	287.30261	31.04245	33.87440
.37494	.14038	1.44186	3.44739	281.86915
.02889	0.00000	0.00000	0.00000	0.00000
0.00000				

43.00000	33.00000	2.76523	31.04245	33.87440
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	2.76523	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 30

47.00000	0.00000	284.13812	64.05737	33.77746
2.09757	.00000	.00000	5.02963	276.96203
.02889	0.00000	0.00000	0.00000	0.00000
0.00000				

44.00000	34.00000	287.30261	306.81614	33.87440
.37494	.14038	1.44186	3.44739	281.86915
.02889	0.00000	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 13

13.00000	0.00000	1042.69427	439.50533	35.57694
676.68979	79.25778	9.41595	4.08608	273.24377
.00089	0.00000	0.00000	0.00000	0.00000
0.00000				

45.00000	0.00000	287.30261	319.22729	33.87440
.37494	.14038	1.44186	3.44739	281.86915
.02889	0.00000	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 29

46.00000	35.00000	284.13812	341.26631	33.77746
2.09757	.00000	.00000	5.02964	276.98203
.02889	0.00000	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 30

47.00000	0.00000	284.13812	64.05737	33.77746
2.09757	.00000	.00000	5.02964	276.98203
.02889	0.00000	0.00000	0.00000	0.00000
0.00000				

44.00000	34.00000	287.30261	306.81614	33.87440
.37494	.14038	1.44186	3.44739	281.86915
.02889	0.00000	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 31

49.00000	36.00000	275.38667	98.20000	33.77746
2.03296	.00000	.00000	4.87472	268.45098
.02800	0.00000	0.00000	0.00000	0.00000
0.00000				

48.00000	12.00000	8.75145	98.20000	33.77746
.06461	.00000	.00000	.15491	8.53105
.00089	0.00000	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 5

9.00000	15.00000	869.70189	518.24308	41.61646
844.62131	.00000	.00000	.15491	8.53105
.00089	.00000	16.39373	.00000	.00000
.00000				

74.00000	41.00000	764.25634	579.39401	48.39200
157.35741	81.97551	.00000	.00000	.00000
502.86551	27.06790	.00000	.00000	.00000
.00000				

70.00000	10.00000	4.45000	164.70000	48.39200
4.45000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

69.00000	3.00000	0.00000	262.70000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	1.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 4

75.00000	0.00000	764.25634	444.24835	48.39200
157.35741	81.97551	.00000	.00000	.00000
502.86551	22.06790	.00000	.00000	.00000
.00000				

56.00000	0.00000	114.99141	262.70000	48.39200
114.99141	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 36

66.00000	0.00000	64.72624	1.00000	1.00000
64.72624	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 6

65.00000	0.00000	37.85120	288.21333	48.39200
8.16948	4.10403	0.00000	0.00000	0.00000
24.73023	.85746	0.00000	0.00000	0.00000
0.00000				

58.00000	0.00000	64.72710	262.70000	48.39200
64.72710	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

67.00000	0.00000	31.39409	111.10000	1.00000
0.00000	0.00000	0.00000	0.00000	0.00000
24.73023	6.57387	0.00000	0.00000	0.00000
0.00000				

68.00000	0.00000	4.94473	164.70000	48.39200
4.94473	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 35

60.00000	0.00000	1102.22486	110.00000	1.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	1102.22486
0.00000				

1.00000	0.00000	636.53862	111.10000	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
502.86551	133.67311	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 3

59.00000	0.00000	8.75145	429.71658	33.77746
.06461	.00000	.00000	.15491	8.53105
.00089	0.00000	0.00000	0.00000	0.00000
0.00000				

76.00000	0.00000	764.25634	441.03281	48.39200
157.35741	81.97551	.00000	.00000	.00000
502.86551	22.06790	.00000	.00000	.00000
.00000				

FINAL OUTPUT STREAMS FOR MODULE 2

77.00000	0.00000	764.26634	397.70604	48.39200
157.35741	81.97551	.00000	.00000	.00000
502.86551	22.06790	.00000	.00000	.00000
.00000				

4.00000	11.00000	16.39373	406.27171	49.34616
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	16.39373	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 1

2.00000	0.00000	636.53862	315.33064	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
502.86551	133.67311	0.00000	0.00000	0.00000
0.00000				

78.00000	0.00000	764.26634	246.65603	48.39200
157.35741	81.97551	.00000	.00000	.00000
502.86551	22.06790	.00000	.00000	.00000
.00000				

FINAL OUTPUT STREAMS FOR MODULE 7

5.00000	13.00000	25.14518	409.10399	47.37232
.06461	.00000	.00000	.15491	8.53105
.00089	0.00000	16.39373	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 37

62.00000	6.00000	1052.22486	110.00000	1.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	1052.22486
0.00000				

71.00000	9.00000	50.00000	110.00000	1.00000
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	50.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 41

81.00000	5.00000	608.81418	315.33064	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
480.96320	127.85098	0.00000	0.00000	0.00000
0.00000				

72.00000	8.00000	27.72444	315.33064	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
21.90231	5.82213	0.00000	0.00000	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 40

75.00000	0.00000	889.31321	110.04363	48.39200
0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	983.31821	0.00000
0.00000				

FINAL OUTPUT STREAMS FOR MODULE 42

82.00000	37.00000	1754.68706	1.00000	1.00000
93.64801	0.00000	0.00000	0.00000	0.00000
480.96320	127.85098	0.00000	0.00000	1052.22486
0.00000				

```
FINAL OUTPUT STREAMS FOR MODULE   43
      83.00000        39.00000        82.17444         1.00000         1.00000
       4.45000         0.00000         0.00000         0.00000         0.00000
      21.90231         5.82213         0.00000         0.00000        50.00000
       0.00000

----

FINAL OUTPUT STREAMS FOR MODULE   44
      84.00000        40.00000        33.51853       579.39401        48.39200
       7.13817         3.71864         0.00000         0.00000         0.00000
      21.90231          .75941         0.00000         0.00000         0.00000
       0.00000

----

FINAL OUTPUT STREAMS FOR MODULE   38
      26.00000         0.00000       889.31821       194.01300        48.39200
       0.00000         0.00000         0.00000         0.00000         0.00000
       0.00000         0.00000         0.00000       889.31821         0.00000
       0.00000

----

       2.00000         0.00000       636.53862       315.33064        48.39200
       0.00000         0.00000         0.00000         0.00000         0.00000
     502.86551       133.67311         0.00000         0.00000         0.00000
       0.00000

----

FINAL OUTPUT STREAMS FOR MODULE   39
      26.00000         0.00000       889.31821       194.01300        48.39200
       0.00000         0.00000         0.00000         0.00000         0.00000
       0.00000         0.00000         0.00000       889.31821         0.00000
       0.00000

----

       2.00000         0.00000       636.53862       315.33064        48.39200
       0.00000         0.00000         0.00000         0.00000         0.00000
     502.86551       133.67311         0.00000         0.00000         0.00000
       0.00000

----

END OF EXECUTION
```

F.3 SYMBØL Simulation

Setting the System of Linear Equations to be Solved

```
SYSTEM DATA
  100     1
   41    11     10      7      6      0     12
    1     1
   0.             0.             0.             0.             0.
   0.             0.             0.             0.             0.
    .1321E+04
EQUATIONS      1 TO   11

    3     2
    .1520E+05      0.             0.             0.             0.
   0.             0.             0.             0.             0.
   0.
    1     1     11      .1151E+02
EQUATIONS     12 TO   23

    2     3
   0.             0.             0.             0.             0.
   0.             0.             0.             .1800E+02      0.
   0.
   12     0.
EQUATIONS     24 TO   35

    2     4
   0.             0.             0.             0.             0.
    .7840E+00      .2560E+00      0.             0.             0.
   0.
    7      .2560E+00
EQUATIONS     36 TO   47

    3     5
   0.             0.             0.             0.             0.
    .1347E+05      .4091E+04      0.             0.             0.
   0.
    7     6     10      .3888E+01
EQUATIONS     48 TO   59

    3     6
   0.             0.             0.             0.             0.
   0.             0.             0.             0.             .1052E+04
   0.
   10     1     11      .7966E+00
EQUATIONS     60 TO   71

    3     7
    .1686E+04      0.             0.             0.             0.
   0.             0.             0.             0.             0.
   0.
    1     6     10      .1602E+01
EQUATIONS     72 TO   83

    3     8
   0.             0.             0.             0.             0.
    .6133E+03      .1863E+03      0.             0.             0.
   0.
    7     9     10      .3726E+01
EQUATIONS     84 TO   95

    3     9
   0.             0.             0.             0.             0.
   0.             0.             0.             0.             .5000E+02
   0.
   10     1     11      .3785E-01
EQUATIONS     96 TO  107
```

```
     3       10
  .8010E+02      0.              0.              0.              0.
  0.             0.              0.              0.              0.
  0.
     1       9       10       .1602E+01
EQUATIONS    108 TO  119

MIXER  1
     2       13      11       12
EQUATIONS    120 TO  131

MIXER  2
     2       14      2        13
EQUATIONS    132 TO  143

MIXER  3
     2       15      3        14
EQUATIONS    144 TO  155

MIXER  4
     2       28      4        27
EQUATIONS    156 TO  167

MIXER  5
     3       37      5        6        7
EQUATIONS    168 TO  179

MIXER  6
     3       39      8        9        10
EQUATIONS    180 TO  191

MIXER  7
     2       41      38       40
EQUATIONS    192 TO  203

SEPARATOR  1
     4       18      19       21
  .8185E+00      .1000E+01       .1000E+01       .1000E+01       .1000E+01
  .1000E+01      .1000E+01       .1000E+01       .1000E+01       .1000E+01
  .1000E+01
    21
EQUATIONS    204 TO  227

SEPARATOR  2
     4       21      22       24
  .7457E+00      .1000E+01       .1000E+01       .1000E+01       .1000E+01
  .1000E+01      .1000E+01       .1000E+01       .1000E+01       .1000E+01
  .1000E+01
    24
EQUATIONS    228 TO  251

SEPARATOR  3
     4       24      25       27
  .4885E-01      .1000E+01       .1000E+01       .1000E+01       .1000E+01
  .1000E+01      .1000E+01       .1000E+01       .1000E+01       .1000E+01
  .1000E+01
    27
EQUATIONS    252 TO  275

SEPARATOR  4
     4       28      29       31
  .1575E+00      .1610E-02       .8445E+00       .8938E+00       .1000E+01
  .1000E+01      0.              .1000E+01       .1000E+01       .1000E+01
  .1000E+01
    31
EQUATIONS    276 TO  299
```

```
SEPARATOR  5
     4       31      32      34
   .1194E+00      .1000E+01      .1000E+01      .1000E+01      .1000E+01
   .1000E+01      .1000E+01      .1000E+01      .1000E+01      .1000E+01
   .1000E+01
    34
EQUATIONS    300 TO  323

SEPARATOR  6
     5       35      36      12
    12      .3080E-01
EQUATIONS    324 TO  347

REACTOR  1
     2       1      11
    11      .1000E+01
    0.              0.              0.              0.              0.
    0.              0.             -.1000E+01       0.              0.
   .1000E+01
EQUATIONS    348 TO  359

REACTOR  2
     2      15      16
     8      .1000E+01
   .2136E+01      -.2264E+01      -.4413E+00      -.4616E-01      -.3843E+00
   0.              .2423E-12       .1000E+01       .1363E-12       .7570E-14
   .6100E-12
EQUATIONS    360 TO  371

REACTOR  3
     2      16      17
     3      .5478E+00
   .1000E+01      -.2471E+01       .1589E+01      -.9550E-02      -.1087E+00
   0.              0.              0.              0.              0.
   0.
EQUATIONS    372 TO  383

REACTOR  4
     2      17      18
     3      .8187E+00
   .1000E+01      -.2376E+01       .1469E+01       .2495E-01      -.1173E+00
   0.              0.              0.              0.              0.
   0.
EQUATIONS    384 TO  395

REACTOR  5
     2      34      35
     2      .1000E+01
  -.5020E+01       .1000E+01       .6536E+01      -.4098E+01       .1582E+01
   0.              0.              0.              0.              0.
   0.
EQUATIONS    396 TO  407

REACTOR  6
     2      37      38
    10      .1000E+01
  -.2987E+00      -.1010E+01      -.8213E-13      -.4693E-13      -.5866E-14
   0.              .1000E+01      -.2363E-12      -.5280E-13       .3086E+00
  -.2363E-12
EQUATIONS    406 TO  419

REACTOR  7
     2      39      40
    10      .1000E+01
  -.9677E+00      -.3272E+01      0.              0.              0.
  -.2037E-11       .3240E+01      0.              0.              .1000E+01
   0.
EQUATIONS    420 TO  431
```

```
REACTOR  8
     2     19     20

STREAM  19. IN SYMBOL HAS NO SPECIFIED EQUIVALENT STREAM IN GEMCS
WHICH IS SAVED IN THE SN MATRIX

     1     .1000E+01

   .1000E+01     0.              0.              0.              0.
   0.            0.              0.             -.1000E+01       0.
   0.
EQUATIONS    432 TO   443

REACTOR  9
     2     22     23

STREAM  22. IN SYMBOL HAS NO SPECIFIED EQUIVALENT STREAM IN GEMCS
WHICH IS SAVED IN THE SN MATRIX

     1     .1000E+01

   .1000E+01     0.              0.              0.              0.
   0.            0.              0.             -.1000E+01       0.
   0.
EQUATIONS    444 TO   455

REACTOR 10
     2     25     26

STREAM  25. IN SYMBOL HAS NO SPECIFIED EQUIVALENT STREAM IN GEMCS
WHICH IS SAVED IN THE SN MATRIX

     1     .1000E+01

   .1000E+01     0.              0.              0.              0.
   0.            0.              0.             -.1000E+01       0.
   0.
EQUATIONS    456 TO   467

REACTOR 11
     2     29     30

STREAM  29. IN SYMBOL HAS NO SPECIFIED EQUIVALENT STREAM IN GEMCS
WHICH IS SAVED IN THE SN MATRIX

     1     .1000E+01

   .1000E+01     0.              0.              0.              0.
   0.            0.              0.             -.1000E+01       0.
   0.
EQUATIONS    468 TO   479

REACTOR 12
     2     32     33

STREAM  32. IN SYMBOL HAS NO SPECIFIED EQUIVALENT STREAM IN GEMCS
WHICH IS SAVED IN THE SN MATRIX

     1     .1000E+01

   .1000E+01     0.              0.              0.              0.
   0.            0.              0.             -.1000E+01       0.
   0.
EQUATIONS    480 TO   491

EXTRA CONSTRAINTS
    36     5     0     0     .5360E+03
EQUATIONS    492 TO   492

          492 EQUATIONS                 492 EQUATIONS
          492 VARIABLES                 492 VARIABLES
         1532 ELEMENTS                 1580 ELEMENTS
          .65 SECS                    11.94 SECS
                                      4041 WORKSPACE
```

Calculated SYMBØL Flows

SYMBOL

MASS BALANCE PROGRAM, MARK2*

MASS BALANCE

PROBLEM 100 CASE 1

STREAM	1	2	3	4	5
1	0.0000	844.5564	0.0000	0.0000	0.0000
2	0.0000	0.0000	0.0000	0.0000	0.0000
3	0.0000	0.0000	0.0000	0.0000	0.0000
4	0.0000	0.0000	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0000	.0280	480.9630
7	0.0000	0.0000	0.0000	.0090	127.8509
8	0.0000	0.0000	0.0000	0.0000	0.0000
9	0.0000	0.0000	0.0000	0.0000	0.0000
10	0.0000	0.0000	0.0000	0.0000	0.0000
11	16.3937	0.0000	0.0000	0.0000	0.0000
TOTAL	16.3937	844.5564	0.0000	.0360	608.8140

STREAM	6	7	8	9	10
1	0.0000	93.6480	0.0000	0.0000	4.4500
2	0.0000	0.0000	0.0000	0.0000	0.0000
3	0.0000	0.0000	0.0000	0.0000	0.0000
4	0.0000	0.0000	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0000	0.0000	0.0000
6	0.0000	0.0000	21.9023	0.0000	0.0000
7	0.0000	0.0000	5.8221	0.0000	0.0000
8	0.0000	0.0000	0.0000	0.0000	0.0000
9	0.0000	0.0000	0.0000	0.0000	0.0000
10	1052.2245	0.0000	0.0000	50.0000	0.0000
11	0.0000	0.0000	0.0000	0.0000	0.0000
TOTAL	1052.2245	93.6480	27.7244	50.0000	4.4500

STREAM	11	12	13	14	15
1	0.0000	.0646	.0646	844.6210	844.6210
2	0.0000	.0000	.0000	.0000	.0000
3	0.0000	.0000	.0000	.0000	.0000
4	0.0000	.1549	.1549	.1549	.1549
5	0.0000	8.5310	8.5310	8.5310	8.5310
6	0.0000	.0009	.0009	.0009	.0009
7	0.0000	0.0000	0.0000	0.0000	0.0000
8	16.3937	0.0000	16.3937	16.3937	16.3937
9	0.0000	-.0000	-.0000	-.0000	-.0000
10	0.0000	-.0000	-.0000	-.0000	-.0000
11	0.0000	-.0000	-.0000	-.0000	-.0000
TOTAL	16.3937	8.7515	25.1452	869.7016	869.7016

STREAM	16	17	18	19	20
1	687.8546	676.6896	668.5226	121.3407	0.0000
2	67.9727	79.2578	87.1955	0.0000	0.0000
3	20.8209	9.4160	1.7074	0.0000	0.0000
4	3.9661	4.0861	3.8569	0.0000	0.0000
5	262.3185	273.2437	281.9601	0.0000	0.0000
6	.0009	.0009	.0009	0.0000	0.0000
7	-.0000	-.0000	-.0000	0.0000	0.0000
8	0.0000	0.0000	0.0000	0.0000	0.0000
9	-.0000	-.0000	-.0000	0.0000	121.3407
10	-.0000	-.0000	-.0000	0.0000	0.0000
11	-.0000	-.0000	-.0000	0.0000	0.0000
TOTAL	1042.9338	1042.6939	1043.1523	121.3407	121.3407

STREAM	21	22	23	24	25
1	547.1819	139.1528	0.0000	408.0291	386.0960
2	87.1955	0.0000	0.0000	87.1955	0.0000
3	1.7074	0.0000	0.0000	1.7074	0.0000
4	3.8569	0.0000	0.0000	3.8569	0.0000
5	281.8691	0.0000	0.0000	281.8691	0.0000
6	.0009	0.0000	0.0000	.0009	0.0000
7	-.0000	0.0000	0.0000	-.0000	0.0000
8	0.0000	0.0000	0.0000	0.0000	0.0000
9	-.0000	0.0000	139.1528	-.0000	0.0000
10	-.0000	0.0000	0.0000	-.0000	0.0000
11	-.0000	0.0000	0.0000	-.0000	0.0000
TOTAL	921.8116	139.1528	139.1528	782.6589	386.0960

STREAM	26	27	28	29	30
1	0.0000	19.9331	19.9331	16.7929	0.0000
2	0.0000	87.1955	87.1955	87.0551	87.0551
3	0.0000	1.7074	1.7074	.2655	.2655
4	0.0000	3.8569	3.8569	.4095	.4095
5	0.0000	281.8691	281.8691	0.0000	0.0000
6	0.0000	.0009	.0289	0.0000	0.0000
7	0.0000	-.0000	.0080	.0080	.0080
8	0.0000	0.0000	0.0000	0.0000	0.0000
9	386.0960	-.0000	-.0000	0.0000	16.7929
10	0.0000	-.0000	-.0000	0.0000	0.0000
11	0.0000	-.0000	-.0000	0.0000	0.0000
TOTAL	386.0960	394.5628	394.5988	104.5311	104.5311

STREAM	31	32	33	34	35
1	3.1402	2.7652	0.0000	.3749	2.0976
2	.1404	0.0000	0.0000	.1404	.0000
3	1.4419	0.0000	0.0000	1.4419	.0000
4	3.4474	0.0000	0.0000	3.4474	5.0296
5	281.8691	0.0000	0.0000	281.8691	276.9819
6	.0289	0.0000	0.0000	.0289	.0289
7	0.0000	0.0000	0.0000	0.0000	0.0000
8	0.0000	0.0000	0.0000	0.0000	0.0000
9	-.0000	0.0000	2.7652	-.0000	-.0000
10	-.0000	0.0000	0.0000	-.0000	-.0000
11	-.0000	0.0000	0.0000	-.0000	-.0000
TOTAL	290.0677	2.7652	2.7652	287.3025	284.1380

STREAM	36	37	38	39	40
1	2.0330	93.6460	150.2132	4.4500	7.1392
2	.0000	0.0000	78.2568	0.0000	3.7196
3	.0000	0.0000	.0000	0.0000	0.0000
4	4.8747	0.0000	.0000	0.0000	0.0000
5	268.4509	0.0000	.0000	0.0000	0.0000
6	.0280	480.9630	480.9630	21.9023	21.9023
7	0.0000	127.8509	21.3085	5.8221	.7594
8	0.0000	0.0000	.0000	0.0000	0.0000
9	-.0000	0.0000	.0000	0.0000	0.0000
10	-.0000	1052.2245	.0000	50.0000	0.0000
11	-.0000	0.0000	.0000	0.0000	0.0000
TOTAL	275.3866	1754.6865	730.7476	82.1744	33.5185

STREAM	41
1	157.3574
2	81.9755
3	.0000
4	.0000
5	.0000
6	502.8653
7	22.0679
8	.0000
9	.0000
10	.0000
11	.0000
TOTAL	764.2661

OUTPUT COMPLETE

```
 12.35 SECS          ERRORS  0
TEST ON SYMBOL STREAM    1 ( GEMCS STREAM    3 )
****** CONVERGED FOR ALL COMPONENT ******

******              27.9830  CPU SECONDS USED TO CALCULATE CASE NUMBER    1    *****

  ******  END OF EXECUTION OF THE COMBINED GEMCS - SYMBOL SIMULATION  ******
```