STEADY-STATE MODELLING AND PARAMETRIC STUDY OF A VAPOR RECOMPRESSION DISTILLATION UNIT

STEADY-STATE MODELLING AND PARAMETRIC STUDY OF A VAPOR RECOMPRESSION DISTILLATION UNIT

BY

M.A. MENZIES, B.E.

A Thesis
Submitted to the Faculty of Graduate Studies
In Partial Fulfilment of the Requirements
For the Degree
Master of Engineering

McMaster University
December 1969 .

MASTER OF ENGINEERING (1969)
(Chemical Engineering)

McMASTER UNIVERSITY
Hamilton, Ontario.

TITLE : Steady State Modelling and Parametric Study of a Vapor Recompression Distillation Unit

AUTHOR : M.A. Menzies, B.E. (Chem.) (University of Canterbury, N.Z.)

SUPERVISOR : Professor A.I. Johnson

NUMBER OF PAGES: viii, 104

SCOPE AND CONTENTS

Steady state heat and mass balancing around an ethylene/ethane distillation unit at Polymer Corporation, Sarnia is studied using the CHESS simulation executive system.

The unit involves a single column with reboiler heat provided by recompression of the overhead vapor stream.

A new column model is developed, based on the approximate pseudo-binary method of Hengstebeck, and is shown to give good results with marked savings in computation time over the conventional tray to tray methods. Models for vapor compression and heat exchange are also presented.

The system model is fitted to plant data and a routine developed to obtain satisfactory system convergence.

A parametric study is carried out in which column pressure and distillate product enthalpy are varied to demonstrate significant improvements in plant operation.

An evaluation of the CHESS simulation system is presented.

ACKNOWLEDGEMENTS

The author gratefully acknowledges the continued assistance and guidance provided by his supervisor, Professor A.I. Johnson.

The cooperation of Polymer Corporation, and in particular Mr. Doug Preston, is appreciated in providing the necessary plant information.

Several members of the McMaster University Computing Centre have provided much needed advice and assistance, particularly in the early stages of this project.

Acknowledgements are also due to McMaster University and the National Research Council for financial support received throughout the study.

The CHESS Program

The CHESS program used in this study is a proprietory computer program written at the University of Houston by Prof. R. Motard and his students.

Arrangements to make commercial use of this program should be made with

TECH Publishing Co.,
4375 Harvest Lane,
Houston,
Texas 77004

In this study the following equipment subroutines were modified:-

ADBF

HXER

The following new subroutines were created:-

ADD1
ADD2
ADD3
ADD4
ADD5

Industrial colleagues who wish to take advantage of these contributions to the system need not have permission from McMaster University; however Prof. A.I. Johnson would be interested in learning of experiences of users of these modified programs.

TABLE OF CONTENTS

APPENDICES

TABLE INDEX

FIGURE INDEX

I. INTRODUCTION

I.I Background

The problem of steady state solution of chemical process networks is frequently encountered in both simulation and design. In large systems, particularly those involving recycle streams, the amount and complexity of calculation is usually large enough to necessitate computer solution. A special case arises where the set of equations describing the system is linear, when a direct, simultaneous solution may be obtained by matrix methods, but for the general non-linear system some iterative method must be applied to achieve the final solution.

In recent years much attention has been directed towards the modular approach to computation of such systems. The method involves replacement of the physical process units by a network of equipment subroutines or modules, each of which may represent a part or group of the original equipment units. Calculation proceeds sequentially through the network calculating each module in turn. Recycle loops are handled by iterative direct substitution, continuing around the loop until a convergence criterion is met. The calling of subroutines, transfer of information and other control functions are handled by an executive system. In addition to these basic functions the executive may include convergence promotion routines, physical properties handling schemes etc. Many modular executive systems have been described [1, 2], most employing the same fundamental algorithm and differing mainly in their degrees of sophistication and areas of application.

Past work at McMaster University [2] has involved the PACER [3], MACSIM [4], and more recently, GEMCS [5], systems. None of these contain physical property calculation schemes. Partly for this reason applications have centred around the mass balancing aspects, rather than treatment of problems where superimposed heat transfer interacts significantly with the system mass flows. Aizawa [6] has dealt with simultaneous heat and mass balancing within distillation columns in a PACER study of a styrene plant. However the study was not concerned with interactions of mass and heat flows external to the column. A study by Petryschuk [7] has also treated column internal heat and mass balancing. A description of the actual column in the present ethylene/ethane separation unit was included but the study did not deal with any aspects of the external vapor recompression cycle.

In most petrochemical distillation operations the interchange and internal recycle of heat within the system is of major importance and directly interacts with column operation. Particularly in low temperature systems, where refrigeration costs are high, considerable benefits may be gained from study of the distribution of internal heat and mass flows.

1.2 Study Objectives

The present study is concerned with simulation of interrelated mass and heat flows using the modular approach. The ethylene/ethane separation unit is a particularly suitable example since the vapor recompression cycle involves high recycle of both heat and mass flows. The simulation executive used is CHESS [8], chosen primarily for its integrated physical properties calculation package.

The study objectives can be summarized as follows:

a) To evaluate CHESS and its property calculation feature, using the CDC6400 computer system.

b) To create a model to describe the superimposed heat and mass balancing in the ethylene/ethane separation unit.

c) To carry out parametric studies on the model to seek improvement in the operation of the unit.

1.3 CHESS Simulation System

1.3.1 System Description

CHESS [8] is a modular, steady state simulation system developed by the University of Houston. It is designed primarily for application to hydrocarbon systems where there is considerable emphasis on phase equilibria and enthalpy calculation, in addition to mass balancing. Its major feature is the comprehensive physical properties calculation package which is an integral part of the system, supplying K-values, enthalpies, densities and bubble and dew point temperatures for liquid or vapor mixtures. Properties are calculated by a set of generalized thermodynamic correlations, starting from basic pure component physical constants, which are pre-programmed into the system. Presently 65 components are available. Total stream enthalpy and vapor fraction are calculated and carried as elements of the stream properties vectors. To further facilitate physical properties handling an adiabatic flash subroutine is included in the system to handle routine phase determination.

1.3.2 Implementation on CDC6400

CHESS [8] was originally written in FORTRAN IV for the IBM360 system. The CDC6400 also employs a FORTRAN IV compiler; however a moderately large number of syntactical changes to the programming were necessary to achieve satisfactory compilation. Additionally the alphameric data storage structure was altered as the CDC6400 stores 10 characters per word compared with 4 per word for the IBM360. The CDC6400 NAMELIST data input feature will not accept alphameric characters so that changes to input data formats were also necessitated.

The full system was found to require in excess of 100 K (octal) of central memory storage. Consequently CDC6400 OVERLAY techniques were employed to reduce the storage requirements to below 60K (octal), necessary to take advantage of rapid daytime turnaround on the McMaster system. The present OVERLAY structure is shown in Figure I.I. Computation of any problem involves sequential loading of the 4 primary OVERLAYs with the bulk of the calculation in OVERLAY (3, 0). Secondary OVERLAYs are needed only when large equipment subroutines are used. Hence the time lost through OVERLAY loading with the present structure is minimal.

A full listing of the CHESS system is given as Appendix IV.

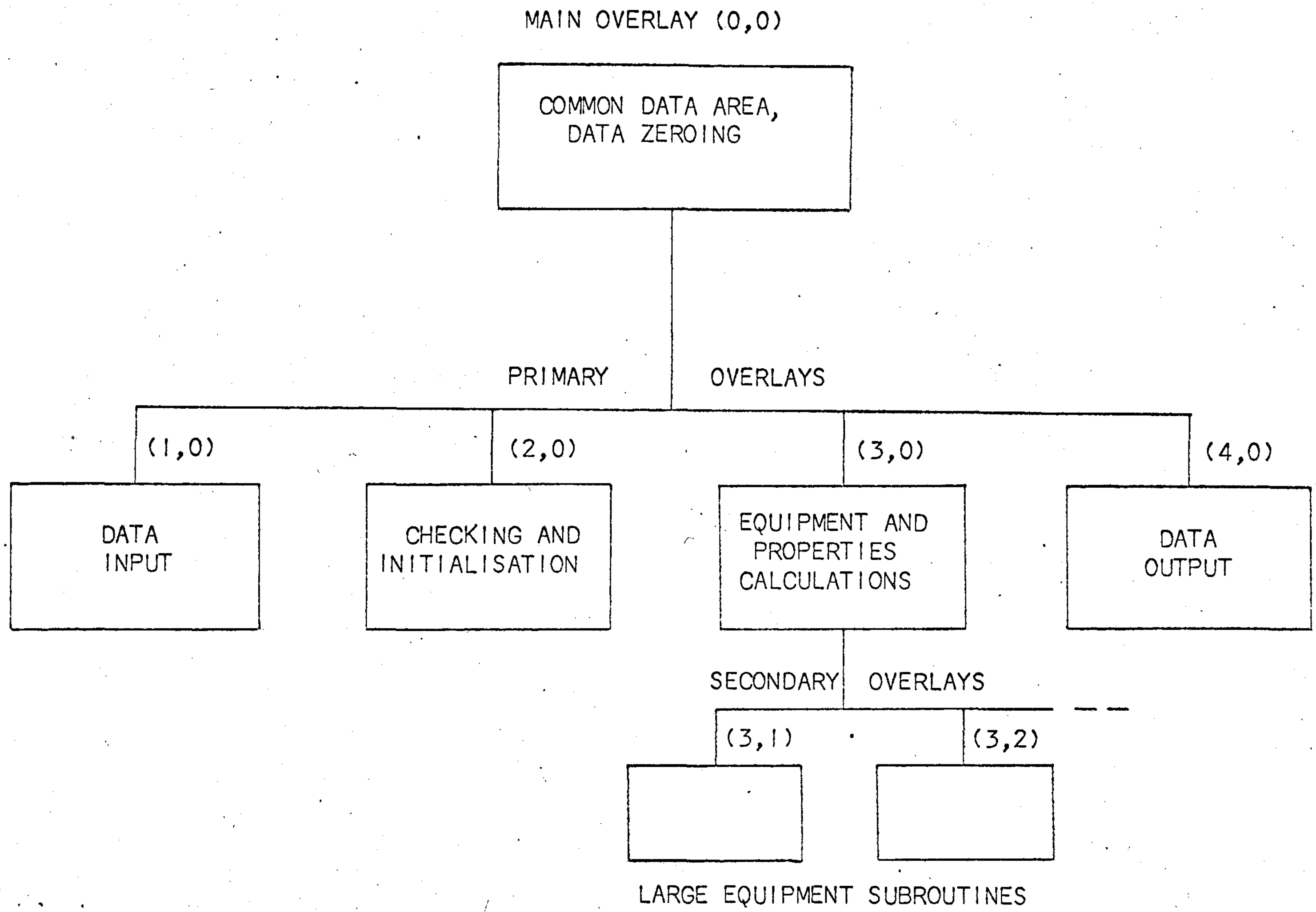


FIGURE 1.1 CHESS OVERLAY STRUCTURE FOR CDC6400

1.4 Process Description

The ethylene/ethane fractionation unit forms part of the light hydrocarbon refining network at Polymer Corporation, Sarnia. The unit flow diagram is shown in Fig. 1.2.

The column contains 60 Glitsch valve trays and operates at 200 psig to produce an ethylene overhead product stream of around 96% purity for styrene manufacture. The bottom ethane stream is recycled to a thermal dehydrogenation furance. The unit operates as a vapor recompression cycle in which reboiler heat is supplied by condensation of the compressed overhead vapor stream.

The feed, overhead product from the de-ethaniser column, is an ethylene/ethane mixture containing a trace of methane and small quantities of propylene and propane. The major part of the feed is liquid, with a small additional vapor stream. The combined streams enter on the 20th tray of the column, with approximately a 10% flash-off of the liquid stream across the control valve, due to pressure reduction from 350 psig. The overhead vapor stream is superheated in the overheads exchanger, C113, by contact with the returning liquid reflux. It passes through a surge drum, F40, and is then compressed to 500 psig by single stage reciprocating compressors, J42B and J54. The bulk of the compressed vapor is condensed in reboilers, C44 and C44A, to reboil the bottoms liquid, thus providing vapor reflux return to the column. The remainder is condensed by ammonia refrigeration in the trimmer condenser, C114. The combined condensate flows to a flash drum, F103. Drum pressure is controlled at 400 psig with the vapor flash-off going to the ethylene product line.

The liquid stream is subcooled in exchanger, C113, a liquid product stream is withdrawn and the remaining flow is fed to the top tray of the column as liquid reflux. This stream is still approximately at flash drum pressure so that there is a flash-off of around 10% across the control valve.

Column pressure is maintained by a small bypass around the compressors. Compressor discharge pressure is held by throttling the exit stream from the trimmer condenser. The reboiler flow is controlled by throttling its exit stream and is adjusted to maintain a constant temperature on tray 11 of the column. The liquid reflux flow is controlled at a constant value.

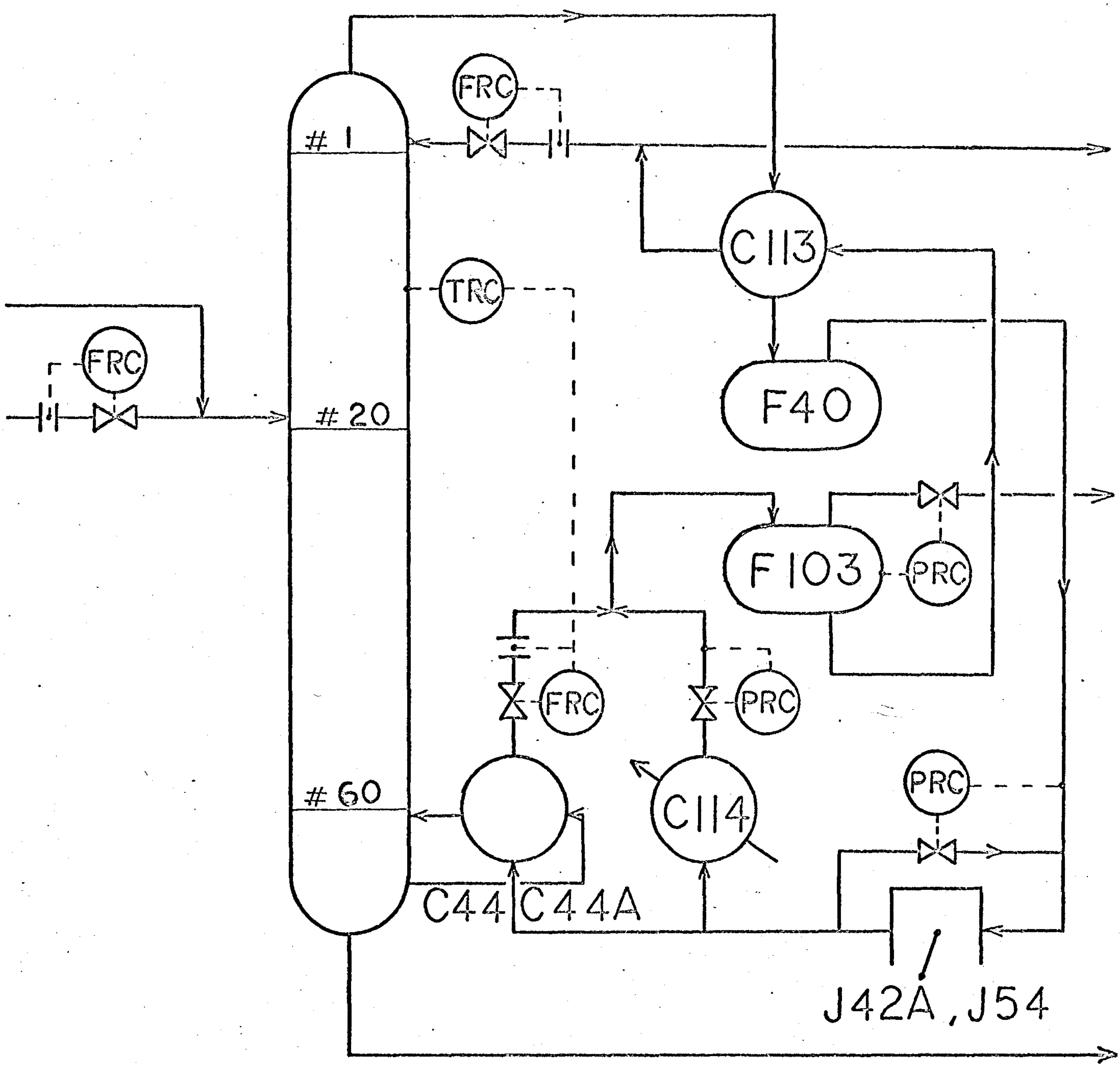


FIGURE 1.2 PROCESS FLOW DIAGRAM

2. STEADY STATE SOLUTION OF PROCESS

2.1 Equipment Modules

The modular information flow diagram developed to represent the process is shown in Fig. 2.1. There is not an exact one-to-one correspondence between equipment modules in Fig. 2.1 and process units in Fig. 1.2. The equipment modules are summarized in Table 2.1, and the configurations representing the physical process units are described in detail in subsequent sections.

Of the models employed, only the mixer, module 12, and valve, module 13, are as originally supplied with the CHESS [8] system. The adiabatic flash, modules 1 and 14, and the heat exchanger, modules 3, 7, 8, 9, 10 and 11, are modified versions of the original routines. The remainder of the models have been developed for the present simulation. Full listings of all model subroutines are given in Appendix III.

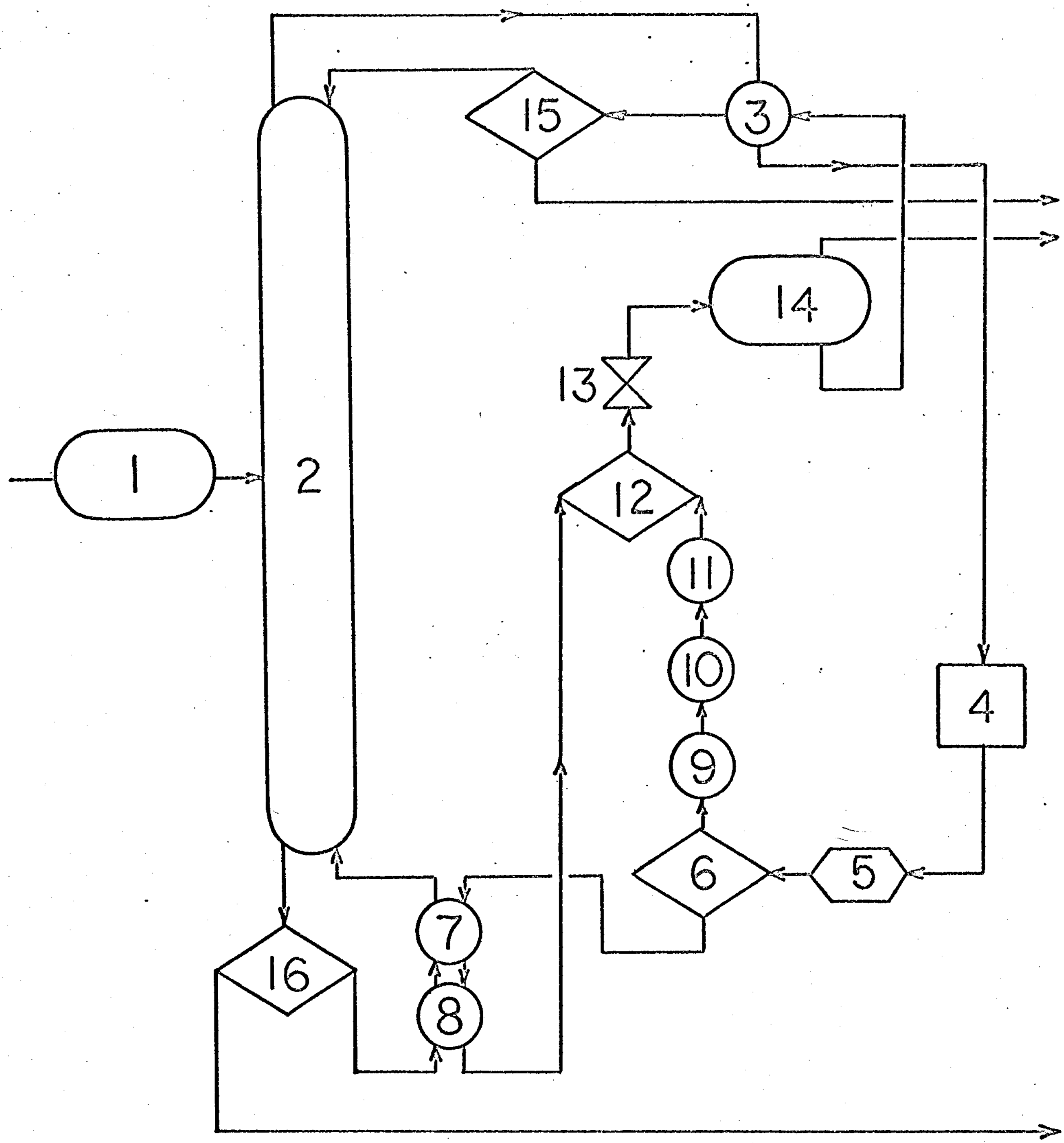


FIGURE 2.1 MODULAR INFORMATION FLOW DIAGRAM

TABLE 2.1

EQUIPMENT MODULE SUMMARY

MODULE	SUBROUTINE NAME	FUNCTION
1	ADBF	Feed flash
2	ADD1	Column
3	HXER	Overheads exchanger
4	ADD3	Compressor
5	ADD4	Constant heat loss
6	ADD5	Vapor divider & system convergence
7	HXER	Reboiler de-superheating section
8	HXER	Reboiler condensing section
9	HXER	Trimmer de-superheating section
10	HXER	Trimmer condensing section
11	HXER	Trimmer subcooling section
12	MIXR	Mixer
13	VALV	Valve
14	ADBF	Flash drum
15	ADD2	Overhead product divider
16	ADD2	Bottom product divider

2.1.1 Column

The feed condition is computed by adiabatic flash, module 1. An input enthalpy is supplied and the feed thermal condition is calculated at the column operating pressure.

The column itself, module 2, is represented by an approximate, pseudo-binary distillation model. The method is based on a design oriented procedure developed by Hengstebeck [9], and depends on a computed separation between two 'equivalent keys'.

The use of a more exact tray to tray calculation technique as used by Petryschuk [7] was impractical for the present application. The column model forms part of the overall calculation loop and the computation time requirements for tray to tray calculations would have been excessive, particularly in view of the large number of theoretical trays. A comparison of methods is presented in Section 2.4.2.

The algorithm for the column model is shown in Fig. 2.2, and the calculation scheme is detailed below.

Two key components are chosen from the feed (ethylene and ethane in the present case). For each the distribution ratio between distillate and bottoms products is assigned an initial estimate. Relative volatilities are calculated and a log-log relationship between distribution ratio and relative volatility is formulated from these two key values.

$$\ln \left(\frac{d}{b}\right)_i = C + C_1 \ln \alpha_i \tag{2.1}$$

where C and C_1 are constants and d and b are the net overhead and bottom product flows of component i, for which α_i is the relative volatility.

For all other components relative volatilities are calculated and the actual separations are estimated from equation 2.1 and the component mass balance

$$f_i = d_i + b_i \qquad (2.2)$$

where f_i is the feed rate of component i.

An equivalent binary separation is then computed for two equivalent keys which are assembled from the feed components by the following procedure.

Critical ratios are defined for each key

$$\ell n \left(\frac{d}{b}\right)_{CL} = \ell n\left(\frac{d}{b}\right)_{LK} + 0.7\left[\ell n\left(\frac{d}{b}\right)_{LK} - \ell n\left(\frac{d}{b}\right)_{HK}\right] \qquad (2.3a)$$

$$\ell n \left(\frac{d}{b}\right)_{CH} = \ell n\left(\frac{d}{b}\right)_{HK} - 0.7\left[\ell n\left(\frac{d}{b}\right)_{LK} - \ell n\left(\frac{d}{b}\right)_{HK}\right] \qquad (2.3b)$$

where subscripts LK and HK refer to the light and heavy keys and C refers to critical values.

Components with ratios between these two critical values are treated as follows. For any light component the d_i and b_i portions are included wholly in the equivalent light key, and heavy components are treated similarly.

Components with ratios outside the critical range are divided into key and non-key portions. For light components b_i is estimated from equations 2.1 and 2.2 and is multiplied by the light key critical ratio from equation 2.3(a). The product is taken to be the d_i contribution to the light key. The b_i contribution is zero.

i.e.

$$(d_i)_{ELK} = b_i\left(\frac{d}{b}\right)_i \qquad (2.4a)$$

$$(b_i)_{ELK} = 0 \qquad (2.4b)$$

where subscripts ELK and EHK refer to the equivalent light and heavy keys. Heavy components are treated similarly.

The equivalent binary feed and product compositions are estimated by a summation of the key contributions.

$$x_F = [\sum_{i=1}^{n} (d_i+b_i)]_{ELK} / ([\sum_{i=1}^{n} (d_i+b_i)]_{ELK} + [\sum_{i=1}^{n} (d_i+b_i)]_{EHK}) \qquad (2.5a)$$

$$x_D = [\sum_{i=1}^{n} (d_i)]_{ELK} / ([\sum_{i=1}^{n} (d_i)]_{ELK} + [\sum_{i=1}^{n} (d_i)]_{EHK}) \qquad (2.5b)$$

$$x_B = [\sum_{i=1}^{n} (b_i)]_{ELK} / ([\sum_{i=1}^{n} (b_i)]_{ELK} + [\sum_{i=1}^{n} (b_i)]_{EHK}) \qquad (2.5c)$$

where x_F, x_D and x_B are the equivalent light key feed, overhead and bottom mole fractions.

Constant molal flows are assumed for each column section. Then for given internal reflux ratio and feed thermal condition the theoretical tray requirements for rectifying and stripping sections can be estimated from the McCabe Thiele procedure [10]. For computer calculation it is convenient to adopt an analytical modification such as that proposed by Stoppel [11].

The overall procedure is applied iteratively, changing the estimated real key separations until the calculated tray requirements balance the number specified for each column section. Actually the change in key separations is achieved by changing in turn the slope and intercept of the line given by equation 2.1. The convergence technique is reguli-falsi [12].

At convergence the total saturated liquid and vapor exit flow rates are calculated from the final d_i and b_i values by overall mass balance.

$$v_{1i} = (R+1)\, d_i \tag{2.6a}$$

$$\bar{\ell}_{iN} = [R \sum_{i=1}^{n} (d_i) + F(1-\psi)]b_i/[\sum_{i=1}^{n} (b_i)] \tag{2.6b}$$

where v_{1i} and $\bar{\ell}_{iN}$ are component vapor and liquid flows from the top and bottom trays (1 and N). R is the internal reflux ratio, F is the total feed rate and ψ is the feed vapor fraction at column conditions.

Exit temperatures and enthalpies are calculated at saturated conditions. An overall column heat balance is calculated but with the model assumption of constant molal overflow there is no satisfactory means of simultaneously imposing it on the mass balance.

For the first application of the model a supplied reflux ratio is used. To aid convergence a 10% increase in overhead vapor flow due to liquid reflux flash-off is assumed. On subsequent applications the reflux ratio is calculated from the liquid or vapor reflux return. In general neither stream is at saturated conditions and an enthalpy balance is used to correct flows to saturated conditions and estimate the vapor flash-off. The reflux ratio is then estimated by overall mass balance.

$$R_L = L/D \tag{2.7a}$$

$$R_{\bar{V}} = (\bar{V} - F\psi)/D-1 \tag{2.7b}$$

where L and $\bar{V}$ are saturated total liquid and vapor reflux flows and D is the total net overhead product flow.

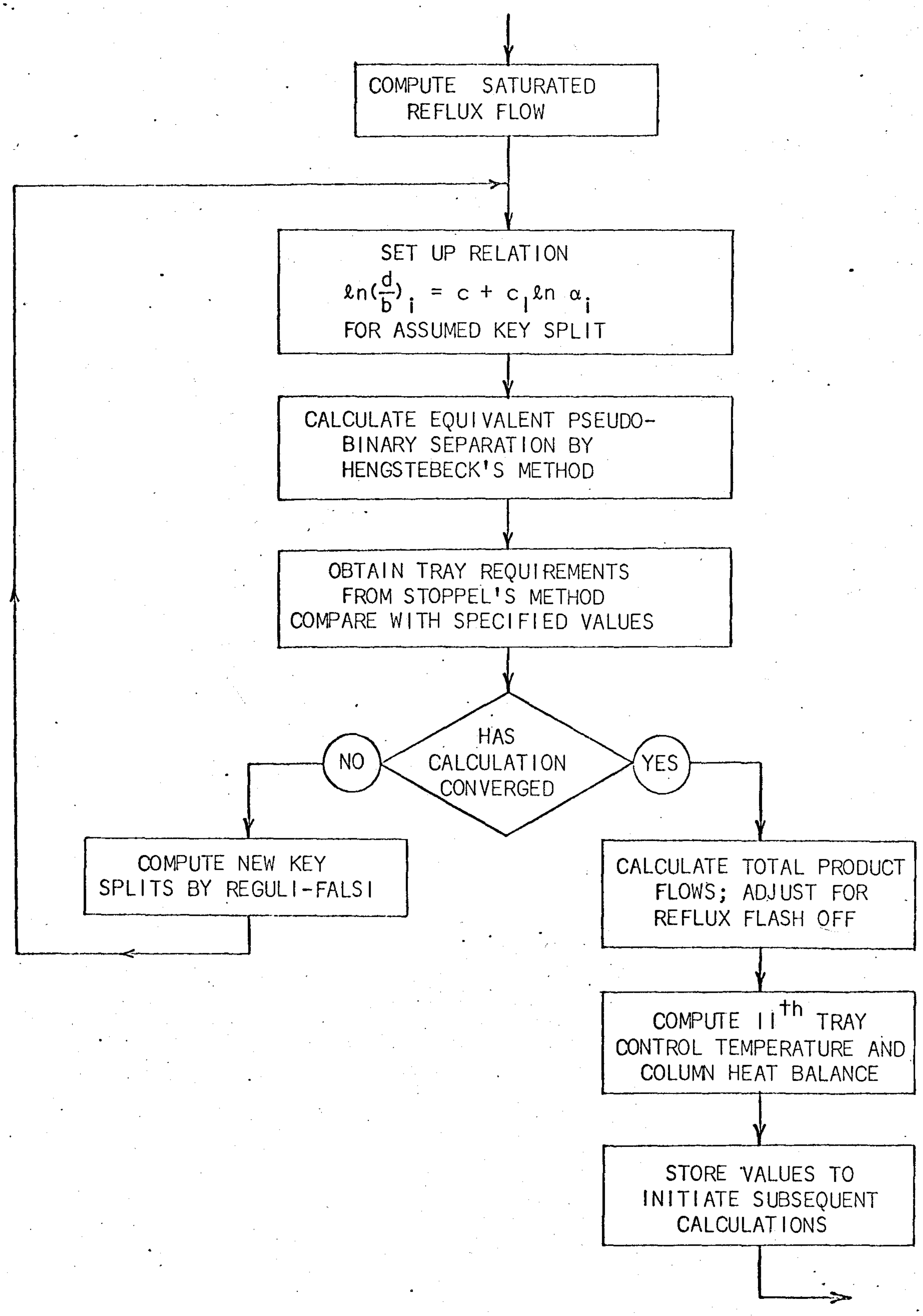


FIGURE 2.2

COLUMN MODEL ALGORITHM

The total liquid and vapor exit flows must subsequently be adjusted for this vapor flash-off.

2.1.2 Heat Exchangers

All exchangers in the network represent various modes of operation of a single heat exchanger model. The basic structure of the present model is that of the originally supplied CHESS [8] version. However it has undergone extensive modification, both to achieve satisfactory convergence and to permit additional modes of operation. The algorithm is based on a constant overall heat transfer coefficient. In the general case it involves iteration until the driving force calculated from stream temperatures corresponds to that computed from the heat duty.

i.e.

$$\Delta T_{LM} = Q/(UA) \tag{2.8}$$

where ΔT_{LM} is the logarithmic mean driving force, Q is the heat duty, U is the overall heat transfer coefficient and A is the heat transfer area.

Exit temperatures are determined by adiabatic flash at enthalpies corresponding to an estimated value of Q. The driving force is calculated from inlet and exit temperatures, with appropriate correction factors for configurations other than simple counterflow. Direct calculation, as in the 'effectiveness factor' approach [13], is not possible as the assumption of constant heat capacities is in most cases not valid and is incompatible with the system physical properties scheme.

The configurations representing the individual process exchangers are described below.

(a) Overheads exchanger - C113

This corresponds to module 3 and represents the normal iterative calculation.

(b) Reboilers - C44 and C44A

The two reboilers are identical units operating in parallel and were treated as a single unit, represented by modules 7 and 8. These correspond to the de-superheating and condensation of the ethylene vapor. Separate modules were employed as the heat transfer fluxes are widely different for each of the processes.

For the de-superheating section the heat exchanger area was calculated, rather than specified, to bring the incoming vapor stream to its dew point. No iteration is necessary. The condensing section was calculated in the normal iterative mode with the remainder of the total reboiler transfer area.

(c) Trimmer condenser - C114

Modules 9, 10 and 11 represent the de-superheating, condensing and subcooling of the inlet vapor stream. There is no second input stream; the shell side of the exchanger is assumed to be at constant temperature corresponding to the evaporating ammonia refrigerant. Modules 9 and 10 are direct calculations (as for module 7 above), bringing the ethylene stream to its dew point and bubble point respectively. Module 11 is calculated iteratively using the residual exchanger area, as for module 8.

2.1.3 Compressors - J42B and J54

The compressor model, module 4, is based on the polytropic relation [14]

$$PV^{\gamma} = \text{Constant} \tag{2.9}$$

where P is the stream pressure, V is the volumetric flow and γ the polytropic compression coefficient.

Iteration proceeds, adjusting the exit temperature until the value of PV^{γ} at inlet conditions matches that at the exit. The value of γ was assumed constant. Convection heat loss from the uninsulated compressor discharge line was modelled by module 5, representing a constant heat flux to the surroundings.

2.1.4 Surge and Flash Drums - F40 and F103

Heat gains to both insulated vessels were estimated from surface temperatures and found to be negligible. For steady state simulation the F40 surge drum fulfils no function and was neglected.

The vapor flash-off in the F103 flash drum was represented by a combination of the adiabatic valve, module 13, representing the pressure reduction, and the adiabatic flash, module 14, which computes the liquid and vapor separation.

2.1.5 Vapor Divider

The division of compressed vapor flow between reboiler and trimmer condenser was represented by a linear splitter, module 6. The module also contains a control routine which adjusts the flow split to achieve the specified control temperature for column tray 11. System convergence is handled by this routine rather than the CHESS system convergence routines which were found to be inappropriate to the present simulation, as will be seen in Section 2.3.

2.1.6 Product Dividers

To ensure a complete system mass balance it was found necessary to remove as product streams exactly the net product component flows calculated by the column model, ie. the d_i and b_i values. Otherwise component build-up or decrease was found to occur during successive calculation loops.

Module 16 places component flows b_i into its first output. Module 15 removes flows such that the combined flow in the liquid and vapor distillate product streams is equal to d_i for each component.

2.1.7 Condensate Mixer

Module 12 performs an adiabatic mixing of its two input streams, using the adiabatic flash routine.

2.2 Equipment Module Convergence

The column, adiabatic flash, heat exchanger and compressor routines all involve iterative calculation with consequent convergence problems. The objective functions are in all cases dependent on values supplied by the physical properties calculation package, making difficult the use of root finding techniques such as Newton Raphson [12]. Forms of reguli-falsi [12] were employed in all cases and were found to be reliable and to involve a minimum of programming. In difficult cases where the rate of change of objective function slope is large in the region of the root, reguli-falsi was combined with a stepping procedure which was used to initiate calculations. Stepping was continued until the root was bracketted, ie. values were obtained on both sides of the function zero. Calculation then continued using reguli-falsi, always obtaining a new point from two points on opposite sides of the root. Some loss in efficiency results in some cases but the modification ensures stability under all conditions, which is essential for this type of application.

Values from model calculations in previous loops were, where possible, used to initiate iterative calculations. Considerable reductions in computation time result.

2.3 System Convergence

For a clearer understanding of the overall system calculation it is convenient to simplify the information flow diagram to that shown in Fig. 2.3. The system reduces to an overhead loop and a bottom loop, connected by the heat transfer across the reboiler and the mass balance relations for the column. A fraction, X, of the overhead stream is condensed in the reboiler with the remainder being bypassed to the trimmer condenser. The vapor reflux and hence the internal column reflux ratio is directly determined by the heat transfer across the reboiler.

Calculation begins with the column feed flash, followed by the column itself, using, only for this first application, a specified reflux ratio. The overhead calculation loop can then be completed to obtain a value for the liquid reflux return, from which the reflux ratio is calculated for a further column calculation. The bottom loop is completed, returning to the column, which is recalculated using the vapor reflux to determine the reflux ratio. The sequence of equipment module numbers for the general overall calculation loop is -

2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 3, 15, 2, 16, 8, 7. Initial values must be supplied for the bottoms streams to both reboiler sections and convergence is aided by specification of initial values for the return condensate stream to the overheads exchanger.

The overall calculation sequence given above involves two column calculations per loop, with reflux ratio calculated alternately from the liquid and vapor reflux streams. For each calculation the 11^{th} tray control temperature is computed and with increasing number of loops the temperatures estimated by both column modes approach a single constant value. The system behaviour is shown in Fig. 2.4, where the control temperature and corresponding reflux ratio, calculated using the vapor reflux, are plotted against loop number with fraction split to the reboiler, X, as a parameter. For low X values the solution continuously decays (ie. system flows decrease), since the flow to the reboiler remains too low to maintain a condensing section driving force high enough for adequate vapor reflux generation. At high X values the sensitivity of the system to changes in X is very low and oscillatory behaviour results. The response of the converged value of control temperature to X is shown in Fig. 2.5, for the region in the vicinity of the final solution. The sharp change in slope coincides with the overhead stream leaving the reboiler at bubble point. While the exit stream contains some vapor, its temperature is relatively insensitive to small increases in heat transfer. However once the stream reaches bubble point a small increase in transfer causes an appreciable fall in exit temperature and condenser driving force, as the condensate becomes subcooled.

The system convergence routine simulates the plant control scheme by adjusting the fraction flow to the reboiler to reach a pre-specified 11^{th} tray control temperature. For any X value iteration continues, following the curves in Fig. 2.4 until the fractional temperature change between successive loops is sufficiently small (0.015°F). The X value is changed and the procedure repeated, continuing until the specified control temperature is met. New X values are obtained from the stepping reguli-falsi technique outlined in Section 2.2, following the curve in Fig. 2.5.

The CHESS convergence testing routine requires all elements of the stream properties vectors for all streams to have a fractional change between successive loops less than a prescribed value. There is no provision for testing of selected streams and/or properties. The scheme is not applicable to the two stage convergence routine developed for the present simulation.

An average of around 25 loops is needed to converge the system to a tolerance corresponding to approximately ±0.2% variation in unit flows. The average loop time on the CDC6400 is around 12 seconds. It is estimated that approximately half of this is used in physical property calculation; in particular within the adiabatic flash routine which the system relies upon for phase determination and estimation of temperature within the two-phase region.

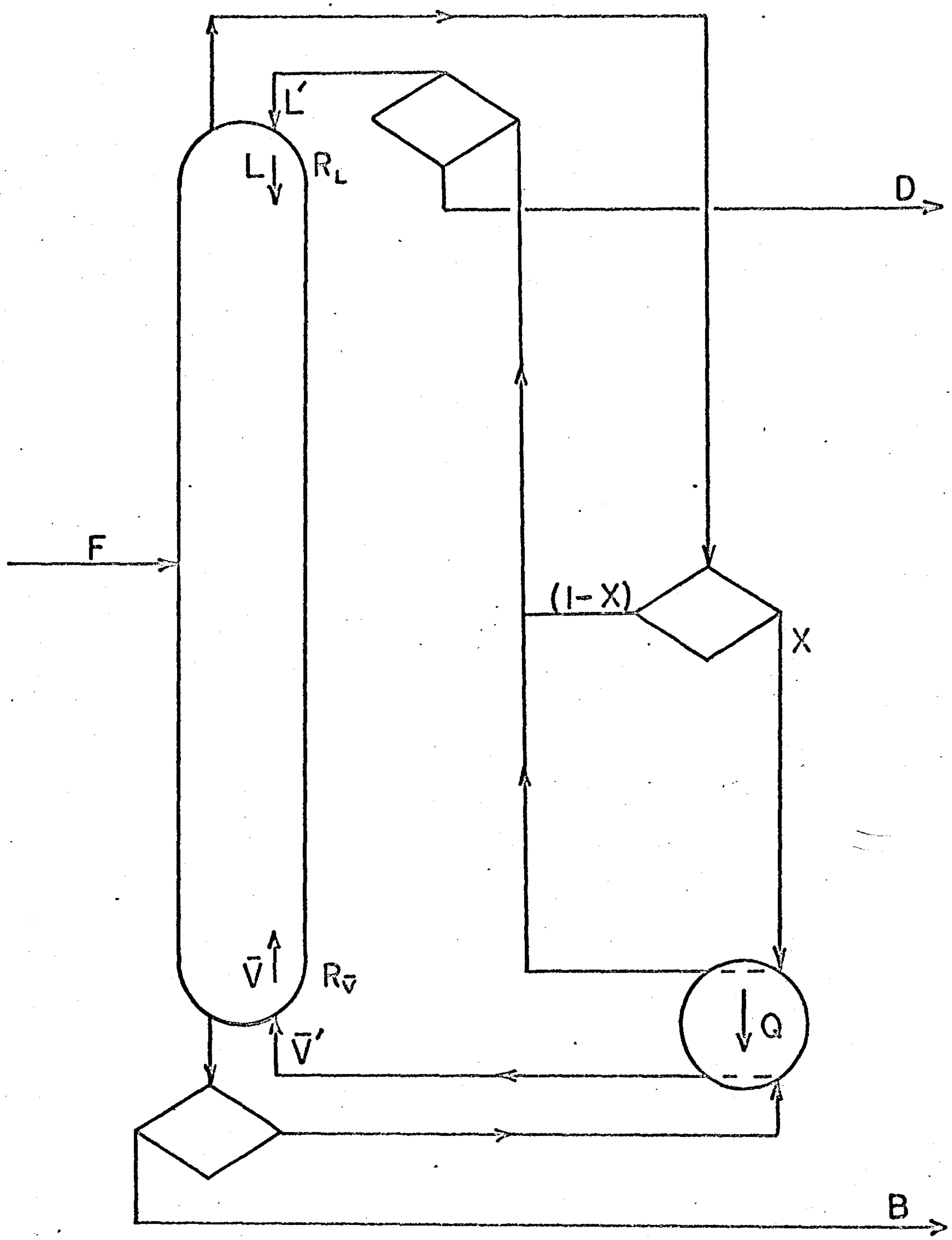


FIGURE 2.3 SIMPLIFIED INFORMATION FLOW DIAGRAM

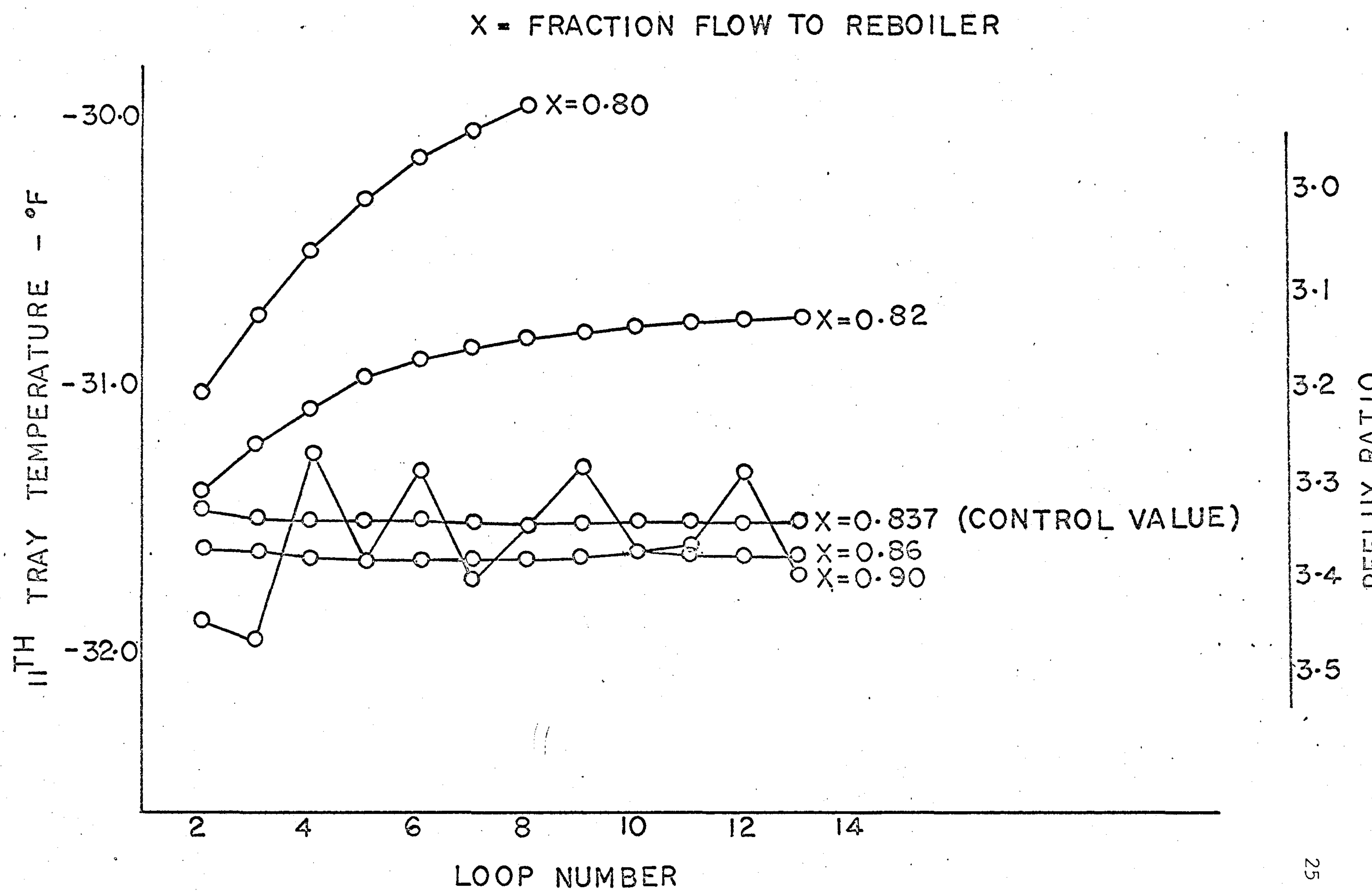
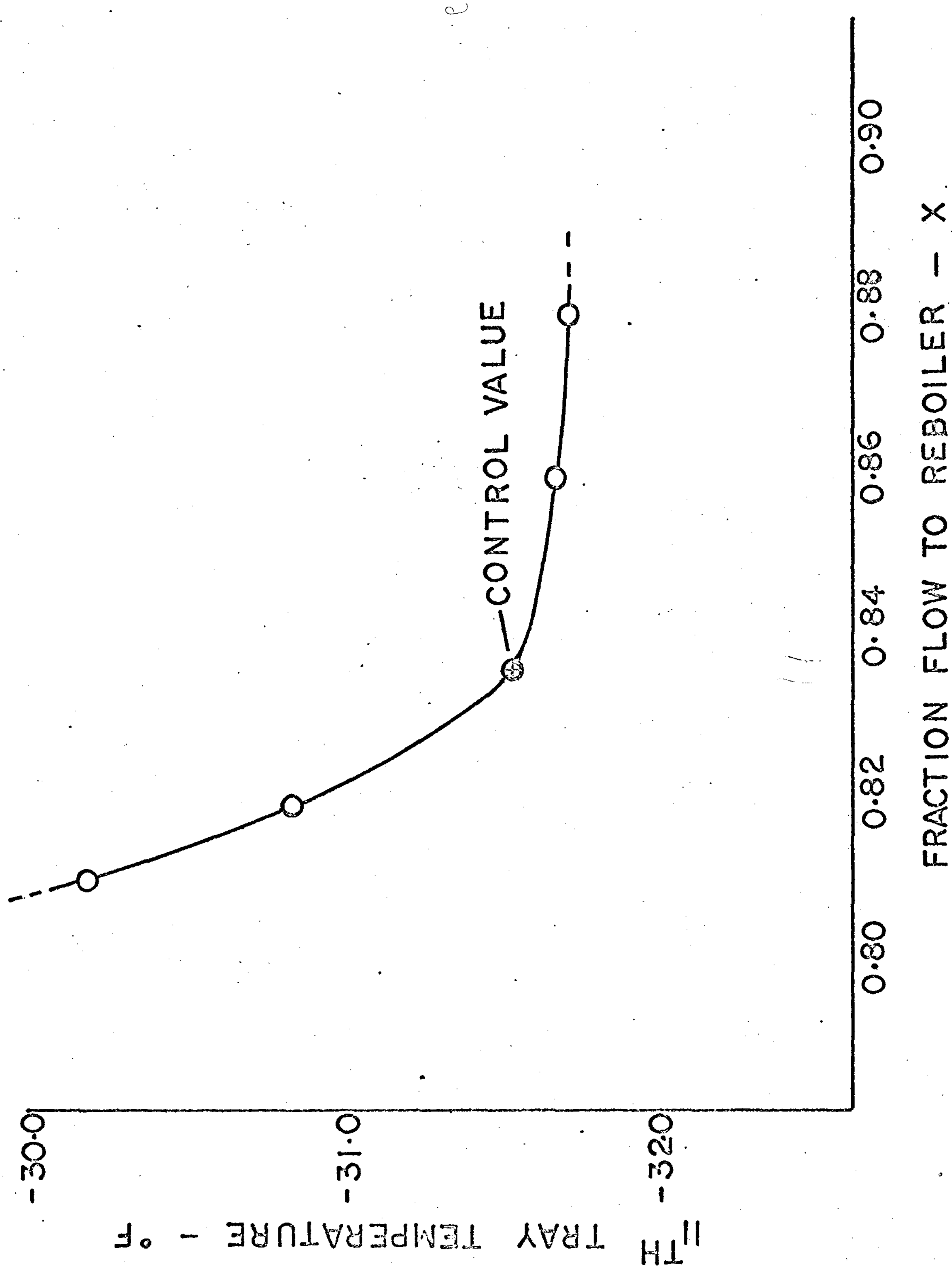


FIGURE 2.4 SYSTEM CONVERGENCE BEHAVIOUR

FIGURE 2.5 VARIATION IN CONVERGED CONTROL TEMPERATURE WITH FRACTION FLOW TO REBOILER



2.4 Base Case Parameter Determination

2.4.1 Plant Data

Polymer Corporation has supplied a set of recent operating data including flow rates, process temperatures and pressures and product analyses. Data were supplied for a four day period over which operation was reasonably steady. For purposes of parameter estimation data were averaged to minimise errors due to inaccurate recording and equipment transients. For a number of streams measurements were unavailable and data were estimated from other supplied values. Equipment details were also supplied and are given in Appendix I. The estimation of equipment parameters for the base case is summarised in the following sections. A full set of base case flows, temperatures, pressures and model parameters, together with a tabulated comparison of results with plant data are given in Appendix II.

2.4.2 Column

The numbers of theoretical trays in each column section were adjusted to match product compositions. The results were compared, for equal reflux ratios, with those from the tray to tray direct iteration program used by Petryschuk [7]. The results are summarised in Table 2.2.

The tray to tray model does not use the CHESS physical properties package so that differences cannot wholly be ascribed to the calculation methods. However it was demonstrated that the McCabe Thiele assumption of constant flows in each column section was justified; the maximum variation was around 7% for stripping section flows.

Accurate comparison of total time requirements is difficult as the number of iterations to convergence depends largely on initial values, but it is evident that a considerable time saving results from using the approximate model. In fact the time differences would be still greater if the tray to tray model had employed generalized correlations for properties rather than simplified regression expressions. There appears to be little loss in accuracy and an important saving in computer storage is also realized.

The overall rectifying section efficiency is 70% from the results. Hence the 11^{th} actual tray, on which the control temperature is measured, was taken as the 7.7^{th} theoretical tray (Stoppel's [11] analytical modification of the McCabe Thiele [10] analysis permits specification of fractional trays). The control temperature value required to converge the system to the base case conditions was within 0.5°F of the plant value.

At convergence the error in the computed overall column heat balance represented a loss of 63,000 BTU, approximately 2.5% of the overhead vapor stream enthalpy. In fact, since the column temperatures are well below ambient, there should be a small gain in heat through the insulation. This discrepancy probably results from small deviations from the assumption of constant mole flows in each column section.

TABLE 2.2

COMPARISON OF COLUMN MODELS

MODEL	ACTUAL TRAYS		THEORETICAL TRAYS		CALCULATION TIME/ITERATION
	RECT.	STRIP.	RECT.	STRIP.	FOR CDC6400-SECONDS
Tray to Tray	20	40	15.5	23	0.4
Pseudo-Binary	20	40	14	21	0.1

2.4.3 Compressor

The polytropic compression coefficient was matched to the inlet and discharge compressor temperatures. The subsequent heat loss from the discharge line was estimated from a measured temperature drop.

2.4.4 Heat Exchangers

The overheads exchanger heat transfer coefficient was estimated from well known correlations [15] for shell and tube side film coefficients, at the estimated flows. The unit is oversized for its present duty. It has a 1-4 configuration; hence the two streams exit at almost equal temperatures which are insensitive to coefficient values. The estimated coefficient produced an exit reflux temperature within 2°F of the plant value.

Overall coefficients for the three trimmer condenser sections were again estimated from correlations. No operating data were available to verify each value but under normal operating conditions the exchanger is oversized so that its performance is insensitive to coefficient values.

The estimated coefficient value for the reboiler de-superheating section combined with the high driving force to predict a very high heat flux for this section. The area required was less than 10% of the total reboiler area so that the de-superheating coefficient has a small effect on the overall reboiler operation. The condensing section however operates with a very low driving force and transfer is controlled by a low shell side boiling coefficient. This coefficient is difficult to estimate accurately but is the critical value in determining the quantity of vapor reflux produced. Hence the coefficient was set to produce the required vapor reflux.

2.4.5 Parameter Errors

It is difficult to estimate errors or further validate model parameters without a more complete set of plant data. This should include temperature profiles along exchangers, necessary to accurately fix coefficient values for exchanger sections. Data representing a different range of operation would have been useful in establishing the range over which the base case parameter set was accurate.

3. PARAMETRIC STUDY

3.1 Statement of Problem and Objectives

Improvement in the function of the unit may be achieved by reducing operating costs and/or increasing the recovery of ethylene. The present study will be directed towards a reduction in operating cost although the ethylene recovery problem is briefly considered in Section 3.2.

The major operating costs are those associated with ammonia refrigeration and vapor compression, both functions of the following variables:

a) Column pressure
b) Internal reflux ratio
c) Compressor discharge pressure
d) Flash drum pressure
e) Distillate product enthalpy

The following constraints are imposed by the process-

i) Column Pressure

The upper limit is around 350 psig, the feed pressure. The lower limit is determined by a combination of the following criteria -

a) Excessive volumetric flow to the compressors. For positive displacement machines the compressor speed is directly proportional to the volumetric throughput. Even if there is no increase in power requirements maintenance costs increase with speed.

b) Flooding and/or excessive entrainment due to increased column vapor velocities.

ii) Internal reflux ratio

The reflux ratio must be sufficiently high to maintain the desired ethylene overhead product purity.

iii) Compressor discharge pressure

The pressure should be just high enough to provide sufficient driving force across the reboiler to maintain the desired column reflux ratio.

iv) Flash drum pressure

The pressure must be between column and compressor discharge pressures. The upper value is further limited by pressure losses across control valves on the reboiler and trimmer condenser exit lines. Sufficient pressure must remain to overcome the loss across the liquid reflux control valve and the hydrostatic head due to column height.

v) Distillate enthalpy

Distillate can be withdrawn from the present system as vapor from the flash drum, as liquid through a bypass around the overheads exchanger, or as liquid after this exchanger. As will be demonstrated later it is advisable to withdraw distillate streams such that their combined enthalpy is as high as possible.

A rigorous minimization of operating cost with respect to the above operating variables is possible through a multivariable search technique such as Hooke & Jeeves [16]. However problems are envisaged with the constraints and a suitable objective function is difficult to formulate due to the interaction of the unit with the overall refining network. The following simplified analysis does however point to significant reductions in operating costs for several changes to current operating conditions.

3.2 Presentation and Discussion of Results

The following simplifying assumptions were made :-

a) Feed conditions were constant

b) Overhead product purity was constant

c) Compressor discharge pressures were adjusted to the minimum for adequate reboiler heat transfer

d) Flash drum pressures were set to constant fractions of the compressor discharge pressures

e) Base case model parameters were assumed constant over the range of investigation. Some variations in heat transfer coefficients, tray efficiencies etc. must be expected. However changes in important flows were small and the results are intended to establish unit improvement trends rather than to provide precise values.

Column pressure was varied between 175 psia and 225 psia for three operating configurations, as summarized in Table 3.1.

TABLE 3.1

OPERATING CONDITIONS FOR PARAMETRIC STUDY

COLUMN PRESSURE Psia	COMPRESSOR DISCHARGE psia	$P_{FLASH}/P_{COMPR.}$ CONFIGURATION[+] A[+]	B	C	LIQUID DISTILLATE TAKE-OFF* CONFIGURATION[+] A[+]	B	C
175	415	0.80	0.80	0.7	a	b	b
200	475	0.80	0.80	0.7	a	b	b
214	505	0.80	0.80	0.7	a	b	b
225	535	0.80	0.80	0.7	a	b	b

* a = After overheads exchanges, b = Before overheads exchanger

\+ Base case configuration

The operation of the column model, for a given feed, depends solely on the internal reflux ratio. For the ethylene/ethane system, reduction in pressure results in significant increase in relative volatility. Hence for constant product purity the reflux ratio can be reduced with pressure as shown in Fig. 3.1.

Figs. 3.2 and 3.3 show the variations in compressor power and refrigeration load with column pressure for the three configurations. The compressor power is computed from the enthalpy change between inlet and discharge. The major effect on the compressor power requirement is the reduction with column pressure corresponding to decreased molar flows at the lower reflux ratios. For a given column pressure the rectifying section vapor flow is fixed but there is a significant increase in vapor flow across the top tray due to liquid reflux flash-off. The magnitude depends on the amount by which the reflux specific enthalpy is above that for saturated conditions at top tray temperature and pressure. As long as there is some bypass of compressed overheads around the reboiler it is highly desirable to reduce this flash-off. It adds directly to the bypass flow and consequently requires both additional compression power and refrigeration. The desirable reduction in liquid reflux enthalpy can be achieved by either - (a) achieving an additional cooling duty in the overheads exchanger or - (b) withdrawing the overhead product at a higher average enthalpy.

Under present operation the overheads exchanger has more than adequate capacity and the cooling duty is limited by the attainable reduction with the 1-4 exchanger configuration. One point (214 B') is given in Figs. 3.2 and 3.3 to demonstrate the effect of changing to a 1-1 counterflow configuration. The reflux cooling is increased with a small reduction in flash-off and compressor duty. However overhead vapor enthalpy is increased by the greater exchanger duty, with the result that a smaller flow to the reboiler is required. Hence the refrigeration load is slightly increased.

Improvement (b) can be achieved by either removing the liquid overhead product through the overheads exchanger bypass (B), or increasing the vapor product flow from the flash drum (by reducing its pressure), or both (C). The resultant savings can be seen to be particularly marked for C. The maximum cost saving is around $6000/year, estimated from steam and cooling water costs for ammonia refrigeration and fuel costs for the gas fueled compressor drivers.

It should be noted that the refrigeration load in particular is very sensitive to the compressor discharge pressure which should therefore be very closely controlled.

The present operation uses the liquid portion of the distillate product for refrigeration in another unit. However, this is in effect produced by high cost ammonia refrigeration. It would be more economical to employ the maximum take-off of distillate product as vapor and replace the loss in refrigeration from an existing lower cost propane system.

The variations in compressor and column volumetric flows are shown in Figs. 3.4 and 3.5. At 175 psia, for configuration C, the increase in compressor volumetric flow is around 5%, which is within the design capacities of the present compressors. The corresponding increase in rectifying section vapor velocity is around 10%. It is doubtful whether this magnitude of increase is possible with present flows without adversely affecting column operation. The best column pressure for operating cost reduction is therefore somewhere below the present value, as limited by column hydrodynamics.

The above study has been concerned only with reduction in operating cost. However ethylene is a valuable product so that considerable profits are to be realized by reducing the ethylene loss in the bottoms stream. This can be achieved by a reduction in column pressure while maintaining the reflux ratios above the constant composition values, used above. The situation is shown in Fig. 3.6. With the higher reflux ratios however, the same reductions in total flows cannot be achieved to give the resultant savings in operating costs. Additionally the limiting column vapor velocities will be reached with much smaller reductions in pressure. Economic considerations indicate a compromise between the two objectives. The further investigation is beyond the scope of the present study.

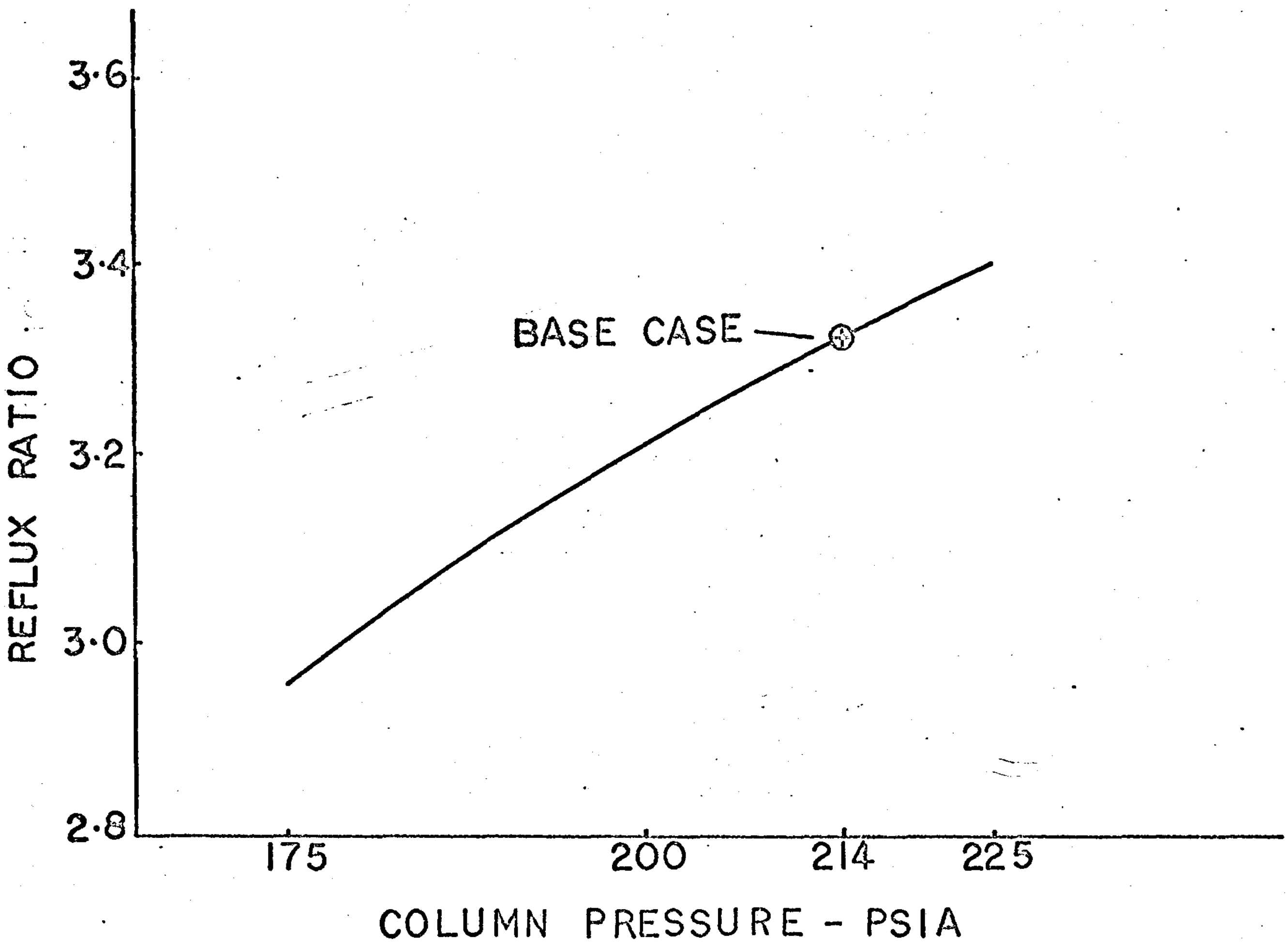


FIGURE 3.1 VARIATION IN REFLUX RATIO WITH COLUMN PRESSURE

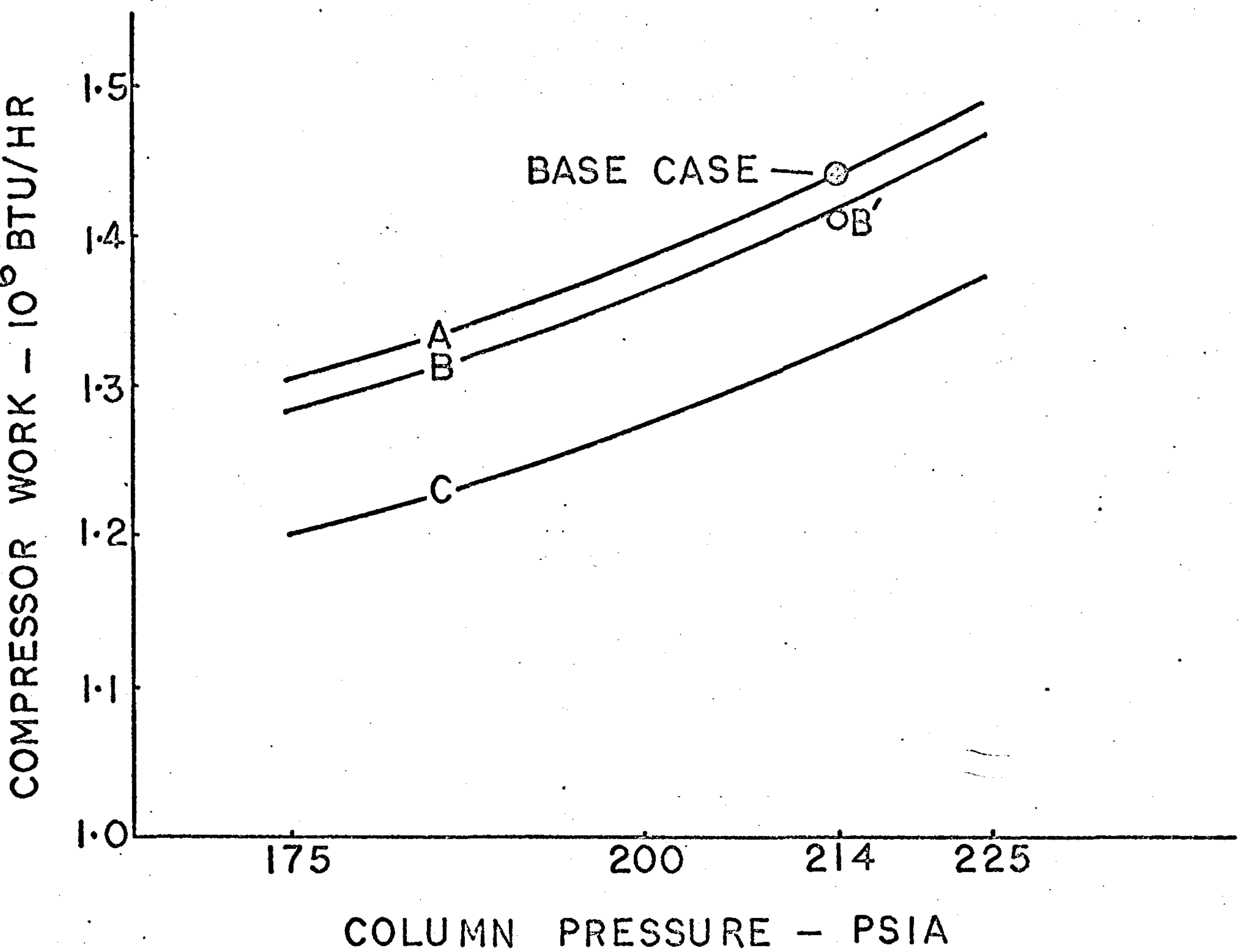


FIGURE 3.2 VARIATION IN COMPRESSOR WORK WITH COLUMN PRESSURE

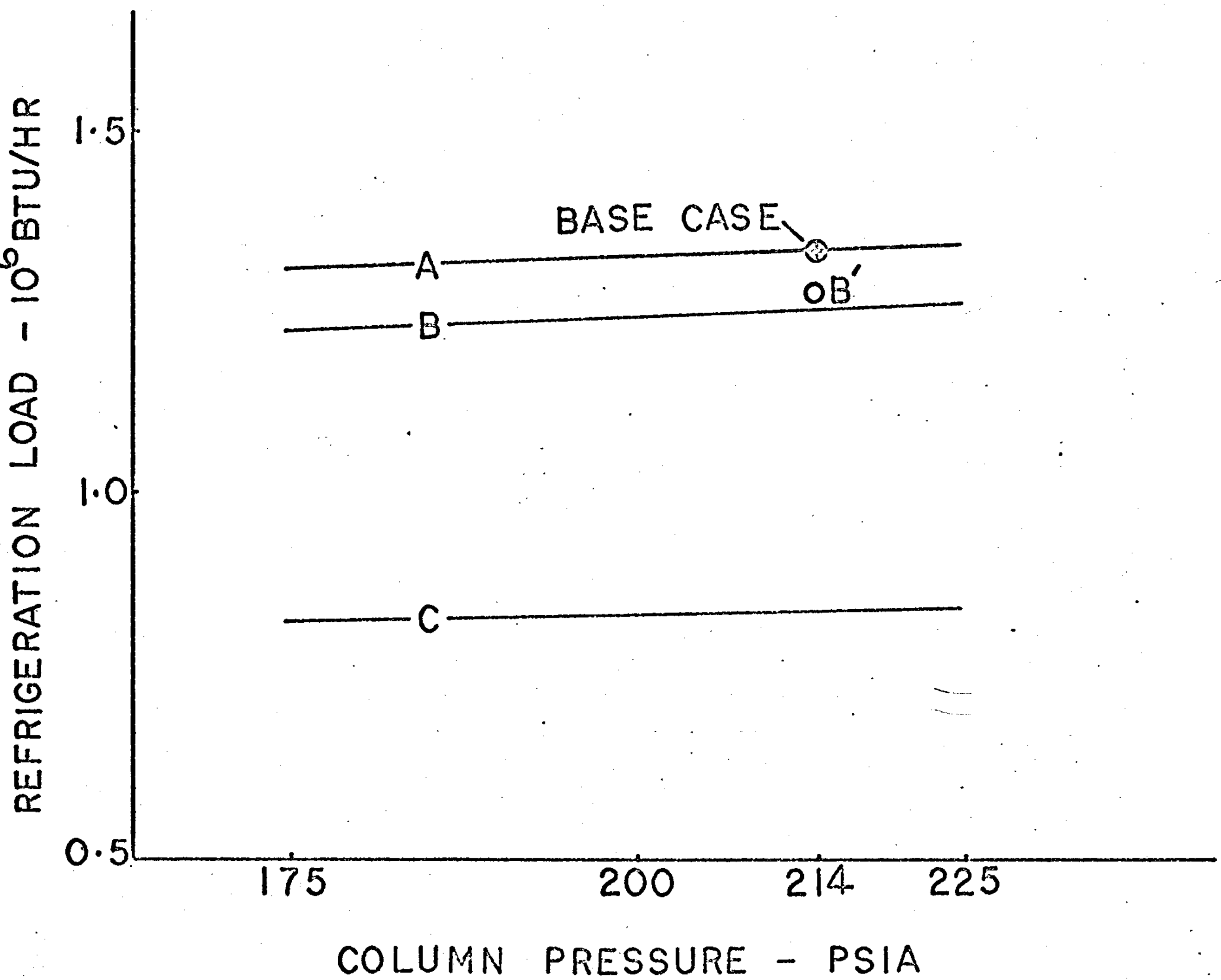


FIGURE 3.3 VARIATION IN REFRIGERATION LOAD WITH COLUMN PRESSURE

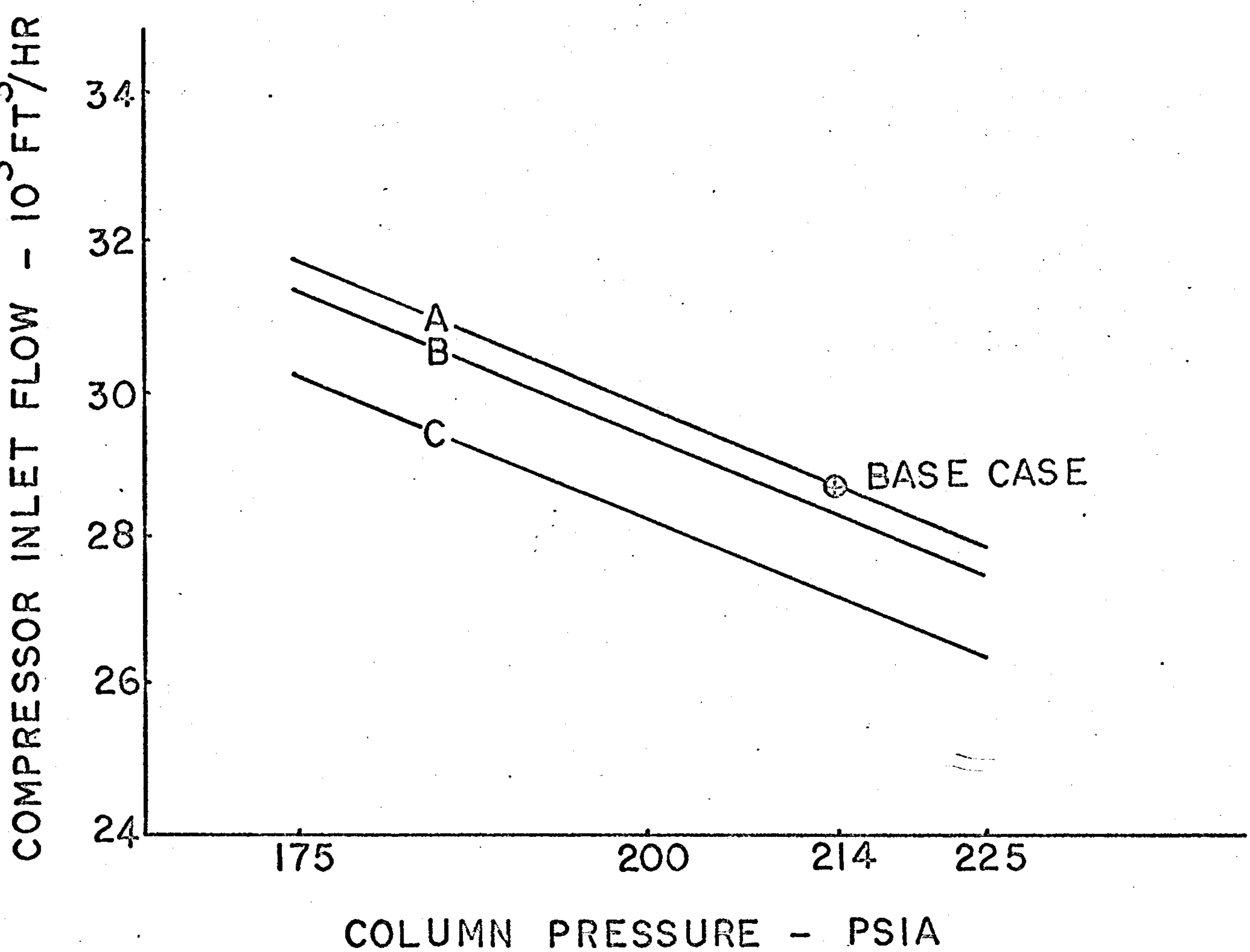


FIGURE 3.4 VARIATION IN COMPRESSOR VOLUMETRIC INLET FLOW WITH COLUMN PRESSURE

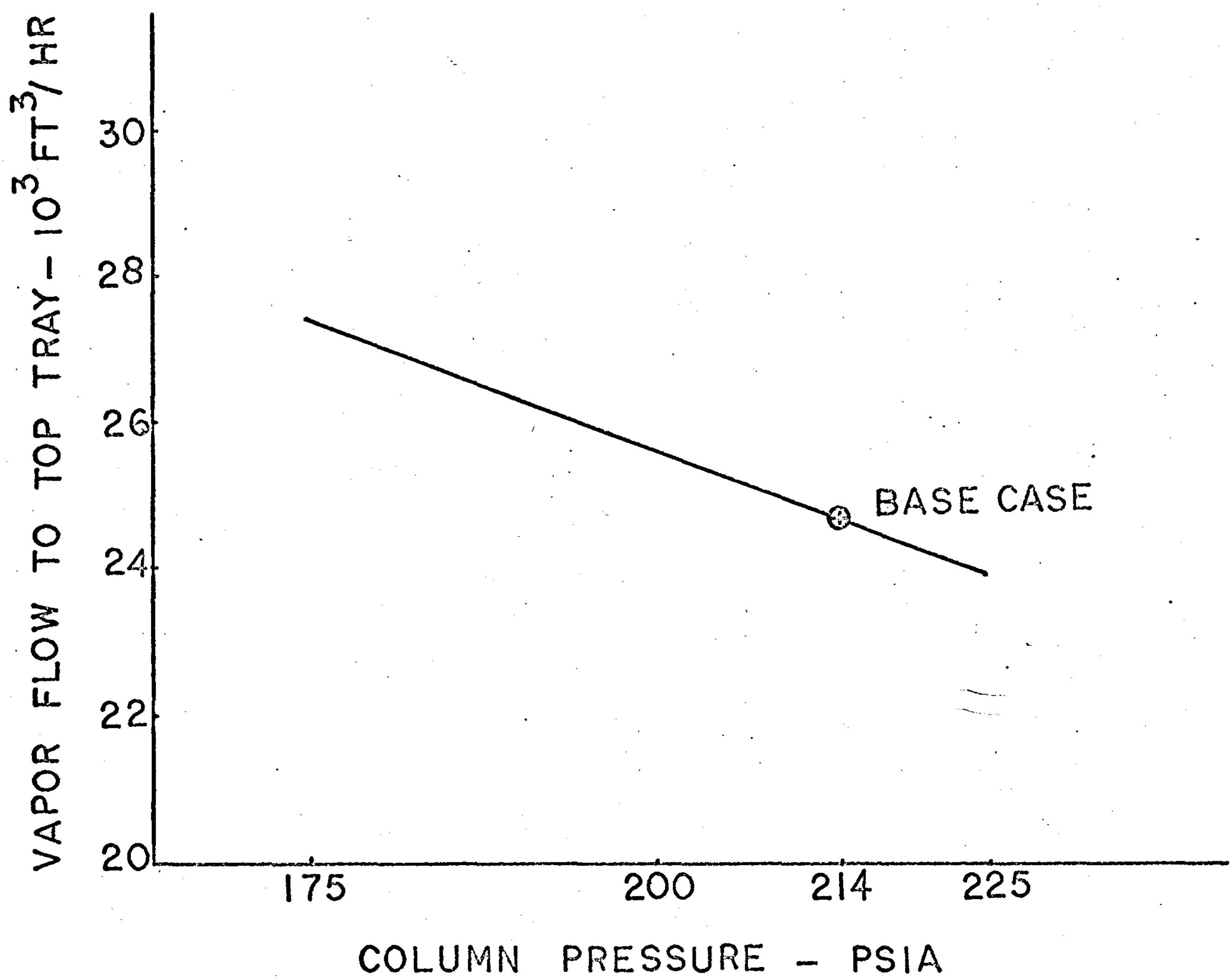


FIGURE 3.5 VARIATION IN VOLUMETRIC FLOW TO TOP TRAY WITH COLUMN PRESSURE

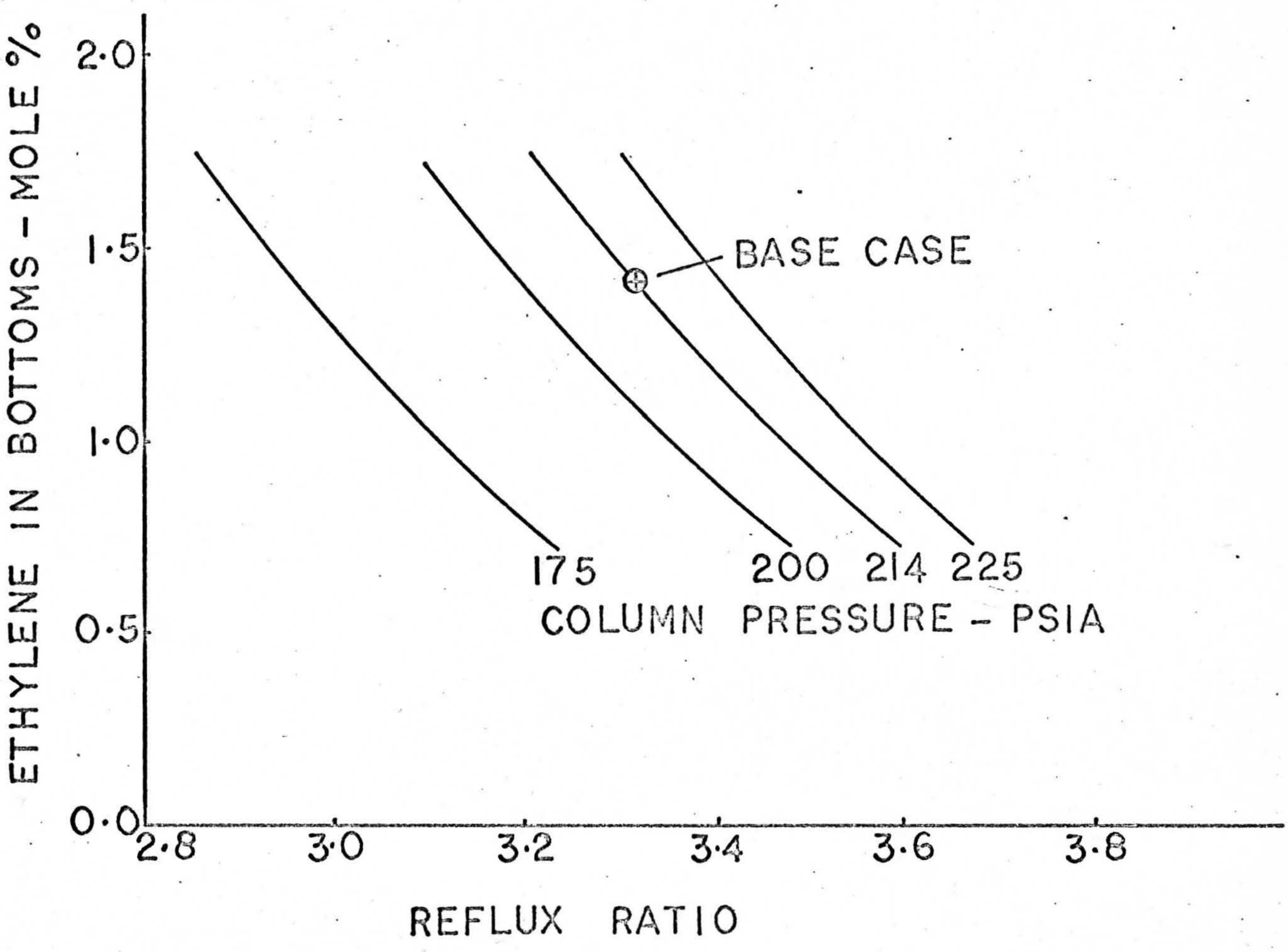


FIGURE 3.6 VARIATION IN BOTTOMS ETHYLENE CONCENTRATION WITH REFLUX RATIO AND COLUMN PRESSURE

4. CHESS SYSTEM EVALUATION

The CHESS (8) system in its present form was well suited to the present study. The physical properties package, especially when used in conjunction with the adiabatic flash routine, was a powerful and readily manipulated tool. Values are accurate but the system requires considerable computational effort in its calculation of properties from basic pure component data on all occasions. The time requirement for the present process was not excessive on the CDC6400 but could easily become so for a slower computer or more complex process. Tray to tray calculations for the column model would have been impracticable for this reason. In other studies it may be desirable to calculate physical properties from simpler regression type expressions, valid over a particular range of interest.

Considerable modification, particularly of convergence routines, was necessary for satisfactory operation of the adiabatic flash and heat exchanger models. This points to the problems involved in writing sophisticated routines for completely general application. The data structures for equipment subroutines and physical properties were well designed to facilitate programming of new models.

It was necessary to write a system convergence routine for the present study. The Wegstein convergence promotion scheme included in the present system is of doubtful value. Hence for many cases it may be beneficial to modify the system to permit the inclusion of user written convergence testing and promotion routines.

The data input and output and output formats for the system were found excellent.

A major disadvantage is that water is not available as a liquid component (due to its non-ideal behaviour) in the present physical properties package so that CHESS may not be applied to aqueous systems.

In conclusion, the CHESS system is best suited to detailed process simulation or design studies of hydrocarbon systems where accurate physical property estimation is essential. In such cases the system can be used to considerable advantage provided that a moderately large usage of computer time is possible.

5. CONCLUSIONS AND RECOMMENDATIONS

5.1 Simulation System and Equipment Models

The CHESS simulation system has proven most satisfactory for the modelling of the ethylene/ethane distillation unit.

Convergence problems within equipment modules and for the overall system were overcome by employing modified reguli-falsi [12] techniques. The CHESS system convergence testing and promotion routines were bypassed and it is suggested that both be either removed or modified to permit greater flexibility of operation.

The accuracy and user convenience of the CHESS physical properties system were found excellent for the present study. Calculation time for property estimation was not excessive on the CDC6400 but could well be a problem with larger process networks and/or slower computers. It is recommended in such cases that approximate methods, accurate within a range of interest, be employed.

The approximate, pseudo-binary column model proved much faster than conventional tray to tray calculation methods, with little loss in accuracy. The use of such a model would be particularly valuable for studies where modelling of the column(s) involved is not the major objective, or where computation time is at a premium.

5.2 Plant Operation

The two most important aspects of the operation of the unit were found to be - (a) the critical heat transfer across the condensing section of the reboiler, and (b) the flash-off of liquid reflux as it enters the column. These points in particular demonstrate the strong interaction between heat and mass flows within the unit.

More comprehensive plant data is necessary to further validate the model. The parametric study did however indicate significant savings in operating costs to be gained by making the following changes in unit operation.

a) Column pressure should be reduced until hydrodynamic limitations occur.

b) Distillate liquid product should be withdrawn before, rather than after, the overheads exchanger.

c) The proportion of vapor distillate product should be increased by a reduction in flash drum pressure.

The present parametric study was directed solely towards operating cost reduction. However increased ethylene recovery is a further important consideration. In fact the most economic operation of the unit would be determined by a compromise between these two objectives and presents an area for further investigation.

6. NOMENCLATURE

All flows are in moles/unit time unless otherwise specified

A - Heat transfer area

b - Bottom product component flow

B - Bottom product total flow

C, C_I - Constants

d - Overhead product component flow

D - Overhead product total flow

f - Feed component flow

F - Feed total flow

$\bar{\ell}$ - Stripping section component liquid flow

L - Rectifying section total liquid flow

P - Pressure

Q - Heat exchanger duty

R - Column internal reflux ratio (= L/D)

ΔT_{LM} - Exchanger logarithmic mean temperature driving force

U - Overall heat transfer coefficient

v - Rectifying section component vapor flow

V - Volumetric compressor flow

$\bar{V}$ - Stripping section total vapor flow

x - Mole fraction

X - Fraction flow to reboiler

Greek Symbols

α - Relative volatility

γ - Polytropic compression coefficient

ψ - Feed vapor fraction

Subscripts

B - Refers to bottom product

CL,CH- Refer to critical values for light and heavy keys

D - Refers to overhead product

ELK,EHK- Refer to equivalent light and heavy keys

F - Refers to feed

HK - Refers to heavy key

i - Refers to specific component

L - Refers to rectifying section total liquid flow

LK - Refers to light key

n - Refers to total number of components

$\bar{V}$ - Refers to stripping section total vapor flow

REFERENCES

1. Evans, L.B., Stewart, D.G. and Sprague, C.R., C.E.P., 64, 2, 39 (1968).

2. Crowe, C.M., Hamielec, A.E., Hoffman, T.W., Johnson, A.I., Shannon, P.T., and Woods, D.R., "Chemical Plant Simulation", McMaster University, Hamilton, Ontario (1969).

3. Shannon, P.T. and Frantz, D.R., "The PACER System Manual", Dartmouth College, Hanover, New Hampshire (1966).

4. Johnson, A.I., "A Manual for the Digital Computer Simulation Programs MACSIM and GEMCS", Dept. of Chemical Eng., McMaster University, Hamilton, Ontario (1968).

5. Johnson, A.I. and Toong, T., "GEMCS General Electric/McMaster Simulator", Dept. Chem. Eng., McMaster University, Hamilton, Ont. and Canadian General Electric, 214 King St., W., Toronto, Ont. (1968).

6. Aizawa, M., M.Eng. Thesis, Dept. Chem. Eng., McMaster University, Hamilton, Ont. (June 1966).

7. Petryschuk, W.F., Ph.D. Thesis, Dept. of Chem. Eng., McMaster University, Hamilton, Ont. (March 1967).

8. Motard, R.L., Lee, H.M., Barkley, R.W. and Ingels, D.M., "CHESS, Chemical Engineering Simulation System, System Guide", Tech. Publ. Co., 4375 Harvest Lane, Houston, Texas (1968).

9. Hengstebeck, R.J., Chemical Engineering 76, 1, 115 (1969).

10. McCabe, W.L. and Smith, J.C. "Unit Operations of Chemical Engineering", McGraw-Hill (1956).

11. Stoppel, A.E., Ind. Eng. Chem., 38, 1271 (1946).

12. Lapidus, L.,
"Digital Computation for Chemical Engineers",
McGraw-Hill (1962).

13. Kreith, F.,
"Principles of Heat Transfer", 2nd. ed.,
International Textbook Co. (1967).

14. Edmister, W.C.,
"Applied Hydrocarbon Thermodynamics", Vol. I,
Gulf Publ. Co., Houston, Texas (1961).

15. Kern, D.Q.,
"Process Heat Transfer",
McGraw-Hill (1950).

16. Kowalik, J. and Osborne, M.R.,
"Methods for Unconstrained Optimization Problems",
Am. Elsevier Publ. Co., New York (1968).

APPENDIX I

EQUIPMENT DETAILS

I.1 COLUMN

Column height	- 106 ft
Column diameter	- 4 ft. 6 in.
Tray type	- Glitsch valve
Total number of trays	- 60
Tray spacing	- 24 in. above feed, 18 in. below
Feed tray number	- 20

I.2 HEAT EXCHANGERS

TABLE I.1

HEAT EXCHANGER DETAILS

ITEM	REBOILER	TRIMMER CONDENSER	OVERHEADS
Type	Kettle-U tube	Kettle-U tube	Floating Heat - Finned tube
Horizontal	Yes	Yes	Yes
Tube side fluid	Overheads	Overheads	Reflux
Shell side fluid	Bottoms	Bottoms	Overheads
Tube passes	2	2	4
Shell passes	1	1	1
No. of tubes	1012	352	388
Tube diameter	3/4 in.	3/4 in.	3/4 in.
Pitch	1 in. - □	1 in. - □	15/16 in. - Δ
Tube length	16 ft.$^{\times}$	16 ft.	12 ft.
Transfer area	6060 ft^{2*}	1100 ft^2	725 ft^2
Heat duty $^+$	4.85	1.34	0.44

$^{\times}$ For single exchangers, C44 and C44A

* Includes both C44 and C44A

$^+$ For Base Case - Units of 10^6 BTU/hr

I.3 COMPRESSORS

The J42A and J54 compressors are both single stage reciprocating units driven by gas engines. Design speeds for the units are 320 and 320 rpm respectively. Speeds reported for the data period were 315 and 260 rpm.

I.4 SURGE & FLASH DRUMS

TABLE I.2

DRUM DETAILS

DRUM	SIZE	INSULATION
Surge Drum (F40)	15 ft. x 6 ft. D	3½ in. Foam glass
Flash Drum (F103)	20 ft. x 6 ft. D	3½ in. Foam glass

APPENDIX II

BASE CASE DESCRIPTION

II.1 Equipment Parameters

The relevant equipment parameters for the base case are summarized below.

a) Column -

Rectifying section theoretical trays	14
Stripping section theoretical trays	20
Average operating pressure	214 psia
Pressure drop for each section	2 psia
Control temperature for 11th actual tray	-31.5°F

b) Heat exchangers -

TABLE II.1

HEAT EXCHANGER BASE CASE COEFFICIENTS

EXCHANGER	SECTION	MODULE NO.	OVERALL COEFFICIENT*
OVERHEADS	-	3	45.0
REBOILER	DE-SUPERHEATING	7	120.0
REBOILER	CONDENSING	8	66.0
TRIMMER+	DE-SUPERHEATING	9	40.0
TRIMMER+	CONDENSING	10	120.0
TRIMMER+	SUBCOOLING	11	45.0

* BTU/HR FT^2 °F

\+ Ammonia side temperature constant at - 18°F (1 psig)

c) Compressor -

Polytropic compression coefficient	1.31
Discharge pressure	505 psia
Heat loss from un-insulated discharge line	0.15×10^6 BTU/hr

d) Flash Drum -

Operating pressure	405 psia

II.2 Stream Information

Streams may be identified through the process matrix which precedes the stream summary. The sign convention used is -

+ve for equipment inputs

-ve for equipment outputs

The entry EQUIP. CONXION in the stream summary also identifies the streams by giving the numbers of the equipments which the given stream connects.

C2 SPLITTER SYSTEM - P=214 (A) (BASE CASE)

PROCESS VECTORS

...... EQUIPMENT			STREAM NUMBERS				
NUMBER	SUBROUTINE	NAME					
1	ADBF	FFLS	1	-2	0	0	0
2	ADD1	HCOL	2	20	25	-3	-21
3	HXER	OHEX	3	17	-4	-18	0
4	ADD3	COMP	4	-5	0	0	0
5	ADD4	DD-1	5	-6	0	0	0
6	ADD5	VDVR	6	-7	-10	0	0
7	HXER	RBDS	7	24	-8	-25	0
8	HXER	RBCO	8	23	-9	-24	0
9	HXER	TRDS	10	-11	0	0	0
10	HXER	TRCO	11	-12	0	0	0
11	HXER	TRSC	12	-13	0	0	0
12	MIXR	OHMX	9	13	-14	0	0
13	VALV	VLV1	14	-15	0	0	0
14	ADBF	FDRM	15	-16	-17	0	0
15	ADD2	TOVR	18	-19	-20	0	0
16	ADD2	BOVR	21	-22	-23	0	0

FINAL RESULTS

C2 SPLITTER SYSTEM - P=214 (A) (BASE CASE)

STREAM SUMMARY

STREAM NUMBER	1	2	3	4	5
EQUIP. CONXION	FR 0 TO 1	FR 1 TO 2	FR 2 TO 3	FR 3 TO 4	FR 4 TO 5
VAPOR FRACTION	.2616	.2616	1.0000	1.0000	1.0000
TEMPERATURE, R	438.5000	438.5000	423.1587	449.5344	570.0333
PRESSURE, PSIA	214.0000	214.0000	212.0000	212.0000	505.0000
ENTHALPY, BTU	-807574.9100	-807574.9100	2489509.3353	2931791.4679	4370208.3522
COMPOSITION, LB-MOLES/HOUR					
METHANE	2.2000	2.2000	10.4473	10.4473	10.4473
ETHYLENE	306.0000	306.0000	1438.4034	1438.4034	1438.4034
ETHANE	217.0000	217.0000	45.7681	45.7681	45.7681
PROPYLENE	12.0000	12.0000	.0000	.0000	.0000
PROPANE	.8000	.8000	.0000	.0000	.0000
TOTAL	538.0000	538.0000	1494.6188	1494.6188	1494.6188

STREAM NUMBER	6	7	8	9	10
EQUIP. CONXION	FR 5 TO 6	FR 6 TO 7	FR 7 TO 8	FR 8 TO 12	FR 6 TO 9
VAPOR FRACTION	1.0000	1.0000	1.0000	0.0000	1.0000
TEMPERATURE, R	562.7461	562.7408	476.0437	474.4143	562.7408
PRESSURE, PSIA	505.0000	505.0000	505.0000	505.0000	505.0000
ENTHALPY, BTU	4220208.3522	3532005.2233	1799801.8225	-2323039.0665	690377.9166
COMPOSITION, LB-MOLES/HOUR					
METHANE	10.4473	8.7442	8.7442	8.7442	1.7092
ETHYLENE	1438.4034	1203.9735	1203.9735	1203.9735	235.3328
ETHANE	45.7681	38.2226	38.2226	38.2226	7.4711
PROPYLENE	.0000	.0000	.0000	.0000	.0000
PROPANE	.0000	.0000	.0000	.0000	.0000
TOTAL	1494.6188	1250.9403	1250.9403	1250.9403	244.5131

STREAM NUMBER	11	12	13	14	15
EQUIP. CONXION	FR 9 TO 10	FR 10 TO 11	FR 11 TO 12	FR 12 TO 13	FR 13 TO 14
VAPOR FRACTION	1.0000	0.0000	0.0000	0.0000	.0571
TEMPERATURE, R	476.0437	474.5919	442.7655	469.9021	460.9591
PRESSURE, PSIA	505.0000	505.0000	505.0000	505.0000	405.0000
ENTHALPY, BTU	351795.4686	-452774.5164	-651042.8631	-2973629.6722	-2975197.3925
COMPOSITION, LB-MOLES/HOUR					
METHANE	1.7092	1.7092	1.7092	10.4534	10.4534
ETHYLENE	235.3328	235.3328	235.3328	1439.3063	1439.3063
ETHANE	7.4711	7.4711	7.4711	45.6937	45.6937
PROPYLENE	.0000	.0000	.0000	.0000	.0000
PROPANE	.0000	.0000	.0000	.0000	.0000
TOTAL	244.5131	244.5131	244.5131	1495.4534	1495.4534

STREAM NUMBER	16	17	18	19	20
EQUIP. CONXION	FR 14 TO 0	FR 14 TO 3	FR 3 TO 15	FR 15 TO 0	FR 15 TO 2
VAPOR FRACTION	1.0000	0.0000	0.0000	0.0000	0.0000
TEMPERATURE. R	460.9591	460.9591	448.5651	448.5600	448.5600
PRESSURE. PSIA	405.0000	405.0000	405.0000	405.0000	405.0000
ENTHALPY. BTU	135490.5428	-3112253.3508	-3554535.4834	-581882.6757	-2972815.2923
COMPOSITION, LB-MOLES/HOUR					
METHANE	1.7111	8.7423	8.7423	.4889	8.2534
ETHYLENE	81.9195	1357.3868	1357.3868	220.9941	1136.3927
ETHANE	1.8132	43.8805	43.8805	7.8034	36.0771
PROPYLENE	.0000	.0000	.0000	.0000	.0000
PROPANE	.0000	.0000	.0000	.0000	.0000
TOTAL	85.4438	1410.0096	1410.0096	229.2864	1180.7232

STREAM NUMBER	21	22	23	24	25
EQUIP. CONXION	FR 2 TO 16	FR 16 TO 0	FR 16 TO 8	FR 8 TO 7	FR 7 TO 2
VAPOR FRACTION	0.0000	0.0000	0.0000	.5312	.7499
TEMPERATURE. R	463.0768	463.0949	463.0949	465.3284	467.0909
PRESSURE, PSIA	216.0000	216.0000	216.0000	216.0000	216.0000
ENTHALPY, BTU	-3919236.7520	-477248.6186	-3441233.3556	681607.5334	2413810.9342
COMPOSITION, LB-MOLES/HOUR					
METHANE	.0000	.0000	.0000	.0000	.0000
ETHYLENE	25.4430	3.0217	21.7882	21.7882	21.7882
ETHANE	1701.5275	207.3076	1494.8058	1494.8058	1494.8058
PROPYLENE	98.4670	12.0000	86.5268	86.5268	86.5268
PROPANE	6.5645	.8000	5.7685	5.7685	5.7685
TOTAL	1832.0020	223.1294	1608.8892	1608.8892	1608.8892

II.3 Comparison with Plant Data

TABLE II.2

BASE CASE COMPARISON WITH PLANT DATA

MEASUREMENTS	BASE CASE VALUE	PLANT VALUE
COMPOSITIONS - Mole fraction		
Overheads ethylene purity	0.962	0.960
Bottoms ethane fraction	0.928	0.922
FLOWS[1] - lb.moles/hr		
Overheads to compressor	1495	1560[2]
Overheads to reboiler	1251	1200
Liquid reflux to column	1181	1255
TEMPERATURES[3] - °F		
Overheads vapor ex-column	-36.8	-40.0
Overheads to reboiler	102.7	106.0
Overheads ex-trimmer	-17.2	-18.0
Overheads ex-reboiler	14.4	13.0
Bottoms ex-reboiler	7.1	4.0
Reflux ex-overheads exchanger	-11.4	-12.0
Column 11th tray set point	-31.5	-31.0

1 Plant measurements within approx. ±5% pf true values

2 Estimated from compressor speeds & Cylinder capacities

3 Plant measurements within approx. ±2°F of true values

Pressures are not quoted as simulation values were set to match average values recorded in the plant.

APPENDIX III

EQUIPMENT SUBROUTINE LISTINGS

Listings are presented for all equipment subroutines used in the present simulation.

```
      SUBROUTINE ADBF
C     ------
C     CHESS EQUIPMENT SUBROUTINE
C
C      MODIFIED BY M.A. MENZIES
C      MC MASTER UNIVERSITY - 1969
C      ------
C
C     DOUBLE PRECISION REMOVED FOR CDC6400 VERSION
C
C     EQUIPMENT PARAMETER LIST=
C     1. - NODE NUMBER
C     2.   - 0. ADIABATIC MODE
C            1. ISOTHERMAL MODE
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
C
C     ******
C
      LOGICAL FLAG,FFLAG
      DATA FLAG/.FALSE./
      DIMENSION SDR(40), DUM(20)
      REAL KS(20),FI(20),FJ(20),OLDKS(20),OLEQ(20),EQR(20)
      REAL NEWDIF
      EQUIVALENCE (SOCOMP,SDR)
      INTEGER TCNT,COUNT
      TCNT=1
      KCNT=0
      IBEST=0
      CALL ZERO(SOENTH,2)
      CALL ZERO(SOMOLE,2)
      CALL ZERO(SDR,40)
      DO 1777 I=1,60
 1777 OLDKS(I)=0.0
      SOPRES(1)=SIPRES(1)
      SOPRES(2)=SIPRES(1)
      TEMP=SITEMP(1)
      SOVPFR(1)=1.0
      SOVPFR(2)=0.0
      COUNT=0
      DO 1 I=1,NOCOMP
    1 IF(SICOMP(I,1).GT. 1.E-7 ) COUNT=COUNT+1
      IF(SIMOLE(1).LT.1.E-7 ) RETURN
      IF(EQPAR(2,NE)-1.E-5)4,4,2
    2 SOTEMP(1)=TEMP
      SOTEMP(2)=TEMP
      SOMOLE(1)=SIMOLE(1)
      SOMOLE(2)=SIMOLE(1)
      DO 3 I=1,NOCOMP
      TEMPA=SICOMP(I,1)
      SOCOMP(I,1)=TEMPA
    3 SOCOMP(I,2)=TEMPA
      TT=TEMP
      IF(COUNT -  1)  6,5,6
    5 IF(ABS(EQPAR(2,NE)-1.)-1.E- 5) 7,7,4
    7 CALL ENTH(1,SIENTH(1),DUM)
      SOVPFR(1)=SIVPFR(1)
      GO TO 9
    6 IF(ABS(EQPAR(2,NE)-3.0)-1.E-5)110,110,11
    4 CALL DEWTP(1,DWT,DUM)
      DO 401 IQ=1,NOCOMP
  401 KS(IQ) = DUM(IQ)
      IF(DWT.LT.0.1.AND.SIVPFR(1).LT.0.000001) GO TO 12
      SITEMP(1)=DWT
      SAVE=SIVPFR(1)
      SIVPFR(1)=1.
      CALL ENTH(1,HV,DUM)
      SITEMP(1)=TEMP
      SIVPFR(1)=SAVE
      IF((SIENTH(1)+ABS(SIENTH(1)/5000.)).LT.HV) GO TO 14
   13 SIVPFR(1)=1.0
      ASSIGN 9 TO NRT
      GO TO 15
    9 SOMOLE(1)=SIMOLE(1)
      DO 16 J=1,NOCOMP
   16 SOCOMP(J,1)=SICOMP(J,1)
      SOENTH(1)=SIENTH(1)
      RETURN
   14 IF(COUNT -  1)  18,17,18
   17 BBT=DWT
      GO TO 19
   18 CALL BUBTP(1,BBT,DUM)
      DO 1801 IQ = 1,NOCOMP
 1801 EQR(IQ) = DUM(IQ)
      IF(BBT.LT.0.1) GO TO 20
   19 IF( ABS(BBT-DWT)/BBT.GT.DERROR/100.) GO TO 1902
      DO 1901 IQ=1,20
 1901 EQR(IQ)=0E0
 1902 SITEMP(1)=BBT
      SAVE=SIVPFR(1)
      SIVPFR(1)=0.0
      CALL ENTH(1,HL,DUM)
      SITEMP(1)=TEMP
      SIVPFR(1)=SAVE
      IF((SIENTH(1)-ABS(SIENTH(1)/5000.)).GT.HL) GO TO 21
   20 SIVPFR(1)=0.0
   12 ASSIGN 22 TO NRT
   23 GO TO 15
   22 IF(NOUT -  1)  24,26,24
   24 SOMOLE(2)=SIMOLE(1)
      DO 25 I=1,NOCOMP
   25 SOCOMP(I,2)=SICOMP(I,1)
      SOENTH(2)=SIENTH(1)
      RETURN
   26 SOVPFR(1)=0.0
```

```
      GO TO 9
C
C **** FOR ONE COMPONENT SYSTEM. CALCULATE V DIRECTLY
C
   21 IF(COUNT -  1)  28,27,28
   27 SITEMP(1)=DWT
      SOTEMP(1)=DWT
      SOTEMP(2)=DWT
      V=(SIENTH(1)-HL)/(HV-HL)
      IF(NOUT -  1)  30,29,30
   29 SOVPFR(1)=V
      GO TO 9
   30 SOMOLE(1)=SIMOLE(1)*V
      SOMOLE(2)=SIMOLE(1)-SOMOLE(1)
      DO 31 I=1,NOCOMP
      SOCOMP(I,1)=SICOMP(I,1)*V
   31 SOCOMP(I,2)=SICOMP(I,1)-SOCOMP(I,1)
      SOENTH(1)=SIENTH(1)*V
      SOENTH(2)=SIENTH(1)-SOENTH(1)
      RETURN
   28 TNEG=BBT.
      TPOS=DWT
      FNEG=(HL-SIENTH(1))/ABS(SIENTH(1))
      FPOS=(HV-SIENTH(1))/ABS(SIENTH(1))
      TEMP=(DWT*FNEG-BBT*FPOS)/(FNEG-FPOS)
      TLOW=BBT
      THIGH=DWT
      DO 1778 IQ = 1,20
 1778 EQR(IQ)=0E0
      GO TO 2
   10 FLAG=.TRUE.
      DO 32 I=1,NOCOMP
      OLDIF=EQR(I)-OLEQ(I)
      NEWDIF=KS(I)-OLDKS(I)
      IF((ABS(NEWDIF)/KS(I)).GT.1.E-5.AND.SICOMP(I,1).GT.1.E-5)
     I  FLAG=.FALSE.
      OLDKS(I)=KS(I)
      IF(KCNT.LE.2) GO TO 33
      IF((ABS(NEWDIF)/KS(I)).LT.1.E-5) GO TO 33
      QWEG=1.-OLDIF/NEWDIF
      IF(ABS(QWEG).LT.1.E-7) GO TO 33
      QWEG=1./QWEG
      IF(QWEG.GT..5) QWEG=.5
      KS(I)=QWEG*EQR(I)+(1.-QWEG)*KS(I)
   33 OLEQ(I)=EQR(I)
   32 EQR(I)=KS(I)
      IF(FLAG           .OR.KCNT.GT.10) GO TO 34
  110 DPSUM=0.E0
      BPSUM=0.E0
      DO 35 I=1,NOCOMP
      IF(KS(I).LT.EXP(-30.)) KS(I)=EXP(-30.)
      FJ(I)=1./KS(I)-1.
      FI(I)=KS(I)-1.
      TREMP=SICOMP(I,1)
      DPSUM=DPSUM+FJ(I)*TREMP
   35 BPSUM=BPSUM+FI(I)*TREMP
      SUM=0.
      IF(DPSUM) 36,36,37
   36 DO 38 I=1,NOCOMP
      SOCOMP(I,1)=SICOMP(I,1)
      SOCOMP(I,2)=SICOMP(I,1)/KS(I)
   38 SUM=SUM+SOCOMP(I,2)
      SOMOLE(1)=SIMOLE(1)
      SOMOLE(2)=SUM
      V=1.
      GO TO 11
   37 IF(BPSUM) 39,39,40
   39 DO 41 I=1,NOCOMP
      SOCOMP(I,2)=SICOMP(I,1)
      SOCOMP(I,1)=SICOMP(I,1)*KS(I)
   41 SUM=SUM+SOCOMP(I,1)
      SOMOLE(2)=SIMOLE(1)
      SOMOLE(1)=SUM
      V=0.
      GO TO 11
   40 V=.5
      DO 42 COUNT=1,20
      SUM=0.0
      DSUM=0.0
      DO 43 I=1,NOCOMP
      IF(V.GT.1.) V=1.0
      IF(V.LT.0.) V=0.0
      TEMPO=FI(I)/(FI(I)*V+1.)
      TEMPA=TEMPO*SICOMP(I,1)
      DSUM=DSUM+TEMPA*TEMPO
   43 SUM=SUM+TEMPA
      TEMPA=SUM/DSUM
      IF((ABS(TEMPA))-1.E-5)45,45,44
   44 IF(V.GT.0.999999.AND.TEMPA.GT.0.0) GO TO 36
      IF(V.LT.0.000001.AND.TEMPA.LT.0.0) GO TO 39
   42 V=V+TEMPA
      IF(V.LE.1.0.AND.V.GE.0.0) GO TO 46
      SUM=0.0
      IF(V) 39,36,36
   45 IF(V.GT.0.999999)GO TO 36
      IF(V.LT.0.000001) GO TO 39
   46  IVPFR(1)=V
      DSUM=0.E0
      SUM=0.E0
      DO 47 I=1,NOCOMP
      TEMPA=SICOMP(I,1)/(V*FI(I)+1.)*(1.-V)
      IF(TEMPA.LT.0.) TEMPA=0.
      TEMPO=SICOMP(I,1)-TEMPA
      IF(TEMPO.GT.0.) GO TO 461
      TEMPA=SICOMP(I,1)
      TEMPO=0.
  461 DSUM=DSUM+TEMPO
      SUM=SUM+TEMPA
      SOCOMP(I,2)=TEMPA
   47 SOCOMP(I,1)=TEMPO
      SOMOLE(1)=DSUM
      SOMOLE(2)=SUM
   11 CALL KVAL(-2,AD,KS)
      CALL KVAL(-1,AD,KS)
      KCNT=KCNT+1
      GO TO 10
```

```
   34 IF(V - 0.999999)49,48,48
   48 SOMOLE(2)=0.0
      CALL ZERO(SDR(21),NOCOMP)
      GO TO 51
   49 IF(V - 0.000001) 52,52,51
   52  OMOLE(1)=0.0
      CALL ZERO(SDR,20)
   51 CALL ENTH(-2,HL,DUM)
      CALL ENTH(-1,HV,DUM)
      SOENTH(1)=HV
      SOENTH(2)=HL
      IF(ABS(EQPAR(2,NE)-1.).LT.0.01) GO TO 53
      FTEMP=(HL+HV-SIENTH(1))/ABS(SIENTH(1))
      AA=ABS(FTEMP)
      IF(AA.LT.0.003) GO TO 53
      IF(TCNT.LE.10) GO TO 55
      WRITE(6,56)AA
   56 FORMAT(39H0***ADIABATIC FLASH DID NOT CONVERGE***,*  - FRACTIONAL
     1ERROR=*,F8.5)
      WRITE (6,57) NE
   57 FORMAT (1H0,*UNIT NUMBER =*,I5)
      GO TO 53
   55 IF(TCNT.LT.3.OR.IBEST.EQ.1) GO TO 552
      IF((ABS(FNEG/FPOS)).LT.20.) GO TO 551
      TT=TLOW+0.8*(TPOS-TLOW)
      IBEST=1
      GO TO 60
  551 IF((ABS(FPOS/FNEG)).LT.20.) GO TO 552
      TT=THIGH-0.8*(THIGH-TNEG)
      IBEST=1
      GO TO 60
  552 IF(FTEMP)72,72,73
   72 T1=TPOS
      F1=FPOS
      GO TO 74
   73 T1=TNEG
      F1=FNEG
C
   74 TT=(TEMP*F1-T1*FTEMP)/(F1-FTEMP)
C
      IF((ABS(TT-TEMP)).LT.0.0015) GO TO 53
      IF(FTEMP)75,75,76
   75 IF(FTEMP.LT.FNEG) GO TO 60
      TNEG=TEMP
      FNEG=FTEMP
      GO TO 60
   76 IF(FTEMP.GT.FPOS) GO TO 60
      TPOS=TEMP
      DPOS=FTEMP
   60 CONTINUE
      TEMP=TT
      TCNT=TCNT+1
      SITEMP(1)=TT
      SOTEMP(1)=TT
      SOTEMP(2)=TT
      KCNT=0
      DO 1779 IQ  = 1,20
 1779 EQR(IQ)=0E0
      GO TO 11
   53 CONTINUE
      IF(NOUT.EQ.1) GO TO 66
      RETURN
   66 SOVPFR(1)=SOMOLE(1)/(SOMOLE(1)+SOMOLE(2))
      SIENTH(1)=SOENTH(1)+SOENTH(2)
      GO TO 9
C **** CALL INTERNAL FUNCTION *GETTP* ****
   15 SITEMP(1)=TEMP
      CALL TSUBH(1,TEMP,DUM)
      SITEMP(1)=TEMP
      SOTEMP(1)=TEMP
      SOTEMP(2)=TEMP
      GO TO NRT,(9,22)
      END
```

```
      SUBROUTINE HXER
C     ------
C     CHESS EQUIPMENT SUBROUTINE
C
C      MODIFIED BY M.A. MENZIES
C      MC MASTER UNIVERSITY - 1969
C     ------
C
C     EQUIPMENT PARAMETER LIST-
C     1. - NODE NUMBER
C     2.  - OVERALL HEAT TRANSFER COEFFICIENT - BTU/HR FT2 DEG. F
C     3.  - AREA/SHELL - FT2
C     4.  - NUMBER OF SHELLS IN SERIES
C     5.  - NUMBER OF SHELL PASSES/SHELL
C     6.  - NUMBER OF TUBE PASSES/SHELL
C     7.  - MODE -
C                   0. SIMPLE EXCHANGER
C                   1. WATER COOLED EXCHANGER
C                   2.-500. REFRIGERATOR,REFRIGERANT TEMP. = MODE,DEG. R
C                   500.+ WATER COOLED EXCHANGER WITH SPEC. OUTLET TEMP.
C                         =MODE,DEG. R
C     8.  - WATER FLOW REQUIRED - GAL/HR
C     9.  - PRESSURE DROP FOR FIRST INLET STREAM - PSIA
C     10. - PRESSURE DROP FOR SECOND INLET STREAM - PSIA
C     11. - HEAT DUTY FOR FIRST INLET STREAM - MM BTU/HR
C     12. - 0. NORMAL CASE
C           1. CALCULATE AREA REQUIRED TO BRING FIRST INLET TO DEW POINT
C           2. CALCULATE AREA TO BRING FIRST INLET TO BUBBLE POINT
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
      COMMON/HXMS/SAVENE(10),SAVEDT(10)
C
C     ******
C
      DIMENSION DUM(20)
      INTEGER COUNT,APRIME,B,C,SAVENE
      REAL M2,M3,M4,M5,M6
      LOGICAL FLAG,FFLAG
      DIMENSION SIDUM(8,11),SODUM(8,11)
      EQUIVALENCE (SIDUM,SINUM),(SODUM,SONUM)
      NNEG=0
      NPOS=0
      IREST=0
      COUNT=0
      U=EQPAR(2,NE)
      AR=EQPAR(3,NE)
      B=EQPAR(4,NE)
      APRIME=EQPAR(5,NE)
      C=EQPAR(6,NE)
      XMODE=EQPAR(7,NE)
      ICD=EQPAR(12,NE)
      TEMPO=APRIME*B
      IF(SIMOLE(1).EQ.0..OR.(SIMOLE(2).EQ.0..AND.XMODE.EQ.0.))
     1GO TO 99
      SAVEP1=SIPRES(1)-EQPAR(9,NE)
      SAVEP2=SIPRES(2)-EQPAR(10,NE)
      NNOUT=NOUT
      IF(ICD.EQ.0) GO TO 903
C
C
C     CALCULATE Q TO REACH DEW OR BUBBLE POINT
C      ICD=1 - DEW POINT
C      ICD=2 - BUBBLE POINT
C
  900 SVVF=SIVPFR(1)
      SVTP=SITEMP(1)
      SIPRES(1)=SAVEP1
      IF(ICD.EQ.2) GO TO 901
      SIVPFR(1)=1.
      CALL DEWTP(1,SITEMP(1),DUM)
      GO TO 902
  901 SIVPFR(1)=0.
      CALL BUBTP(1,SITEMP(1),DUM)
  902 CALL ENTH(1,HSAT,DUM)
      SIVPFR(1)=SVVF
      SITEMP(1)=SVTP
      Q=SIENTH(1)-HSAT
      GO TO 99
C
C
C
C     FOLLOWING REFERS TO EQUIPMENT NOS. SPECIFIC TO C2 SPLITTER SYSTEM
C
  903 IF(NE.EQ.10) EQPAR(3,10)=6060.-EQPAR(3,9)
      IF(NE.EQ.13) EQPAR(3,13)=1100.-EQPAR(3,11)-EQPAR(3,12)
      AR=EQPAR(3,NE)
C
C
C     FIND QMAX FOR FIRST INPUT STREAM  -  (MAX. HEAT TRANSFER)
C
      DO 91 I=1,11
   91 SIDUM(3,I)=SIDUM(1,I)
      SIPRES(1)=SAVEP1
      SITEMP(1)=550.
      IF(XMODE.EQ.0.) SITEMP(1)=SITEMP(2)
      IF(XMODE.GT.2..AND.XMODE.LT.500.) SITEMP(1)=XMODE
      EQPAR(2,NE)=1.
      NOUT=2
      CALL ADBF
      QMAX1=ABS(SIENTH(1)-SOENTH(1)-SOENTH(2))
      DTMX1=QMAX1/(U*AR*B)
C
C     FIND MAX. TEMP. DIFF. FOR EXCHANGER  -  (ZERO HEAT TRANSFER)
C
      DTMX2=ABS(SITEMP(1)-SITEMP(3))
```

```
      DTMAX1=AMIN1(DTMX1.DTMX2)
C
      IF(XMODE.GT.0.) GO TO 94
C
C     FIND QMAX FOR SECOND INPUT STREAM  -  (MAX. HEAT TRANSFER)
C
      DO 92 I=1.11
   92 SIDUM(1.I)=SIDUM(2.I)
      SIPRES(1)=SAVEP2
      SITEMP(1)=SITEMP(3)
      CALL ADBF
      QMAX2=ABS(SIENTH(2)-SOENTH(1)-SOENTH(2))
      DTMAX2=QMAX2/(U*AR*B)
   94 DO 93 I=1.11
   93 SIDUM(1.I)=SIDUM(3.I)
      SOPRES(1)=SAVEP1
      SOPRES(2)=SAVEP2
      NOUT=NNOUT
      DTMAX=AMIN1(DTMAX1.DTMAX2)
C
   99 DÓ 2 J=1.2
      DO 811 I=3.11
  811 SODUM(J.I)=SIDUM(J.I)
      DO 2 I=I.NOCOMP
    2 SOCOMP(I.J)=SICOMP(I.J)
      IF(SIMOLE(1)) 3.3.4
    3 WRITE (6.5)
    5 FORMAT(57H0*** HX CALCULATION BY-PASSED. FIRST INPUT STREAM IS ZER
     1O)
      GO TO 6
    4 IF(XMODE.EQ.0.) GO TO 8
      IF(XMODE.GT.2.0.AND.XMODE.LT.500.) GO TO 801
    7 T2=565.
      IF(SOTEMP(1).LT.550..OR.T2.GT.SITEMP(1)) GO TO 9
      SITEMP(2)=550.
      SOTEMP(2)=565.
      GO TO 10
  801 SITEMP(2)=XMODE
      SOTEMP(2)=SITEMP(2)+1.0
      T2=SOTEMP(2)
      GO TO 10
    8 IF(SIMOLE(2)) 12.12.10
   12 WRITE (6.13)
   13 FORMAT(59H0*** HX CALCULATION BY-PASSED. SECOND INPUT STREAM IS ZE
     1RO )
      GO TO 6
   10 FFLAG=.FALSE.
      J=2
      IF(SITEMP(1).GT.SITEMP(2)) J=1
      EQPAR(2.NE)=0.
      IF(ICD.NE.0) GO TO 191
      DO 14 I=1.11
      IF(I.GT.10) GO TO 18
      IF(SAVENE(I).EQ.NE) GO TO 16
   14 CONTINUE
   16 ISW=1
      FACTOR=0.97
      DT=SAVEDT(I)
      IF(DT.GT.DTMAX) GO TO 181
      GO TO 15
   18 ISW=0
      FACTOR=0.85
  181 DT=DTMAX*0.98
   15 DDT=DT
C
   19 Q=U*AR*DT*B
  191 FLAG=.FALSE.
      IF(J.EQ.2) Q=-Q
      SOENTH(1)=SIENTH(1)-Q
      SOENTH(2)=SIENTH(2)+Q
      DO 21 K=1.2
      DO 20 I=1.11
      SIDUM(K+2.I)=SIDUM(K.I)
   20 SODUM(K+2.I)=SODUM(K.I)
      DO 21 I=1.NOCOMP
      SICOMP(I.K+2)=SICOMP(I.K)
   21 SOCOMP(I.K+2)=SOCOMP(I.K)
      DO 201 I=1.11
  201 SIDUM(1.I)=SODUM(1.I)
      DO 202 I=1.NOCOMP
  202 SICOMP(I.1)=SOCOMP(I.1)
      IF(SITEMP(1).EQ.0.0) SITEMP(1)=500.
      EQPAR(2.NE)=0.0
      IF(XMODE-550.)23.23.22
   22 SITEMP(1)=XMODE
      EQPAR(2.NE)=1.0
      FFLAG=.TRUE.
   23 CALL ADBF
      SOVPFR(3)=SIVPFR(1)
      T1=SITEMP(1)
      IF(FFLAG)          GO TO 24
      SOTEMP(3)=T1
      IF(XMODE.EQ.0.) GO TO 29
      GO TO 24
   29 DO 30 I=1.11
   30 SIDUM(1.I)=SODUM(4.I)
      SIVPFR(1)=SIVPFR(4)
      DO 31 I=1.NOCOMP
   31 SICOMP(I.1)=SOCOMP(I.4)
      IF(SITEMP(1).EQ.0.)SITEMP(1)=500.
      CALL ADBF
      T2=SITEMP(1)
       OTEMP(4)=T2
      SOVPFR(4)=SIVPFR(1)
   24 EQPAR(2.NE)=U
      IF(.NOT.FFLAG) GO TO 37
   36 SOTEMP(3)=T1
      Q=SOENTH(1)-SIENTH(3)
      SOENTH(3)=SIENTH(3)+Q
      Q=ABS(Q)
   37 DO 39 K=1.2
      DO 38 I=1.11
      SIDUM(K.I)=SIDUM(K+2.I)
   38 SODUM(K.I)=SODUM(K+2.I)
      DO 39 I=1.NOCOMP
      SICOMP(I.K)=SICOMP(I.K+2)
```

```
39 SOCOMP(I,K)=SOCOMP(I,K+2)
41 IF(FLAG)                GO TO 43
45 COUNT=COUNT+1
46 S=ABS(SITEMP(1)-SITEMP(2))
   R=(SITEMP(1)-T1)/(T2-SITEMP(2))
   IF(J-I) 48,47,48
47 S=(T2-SITEMP(2))/S
   GO TO 49
48 S=(T1-SITEMP(1))/S
   R=1./R
49 TEMP=1.-R*S
   IF((1.-S).LE.0..OR.TEMP.LE.0.) GO TO 9
   GO TO 50
 9 CONTINUE
   WRITE (6,51)
51 FORMAT(104H0*** HEAT EXCHANGER CALCULATION FAILED DUE TO IMPROPER
  1TEMPERATURE LIMITS, INPUT T TRANSFERRED TO OUTPUT)
   SOTEMP(1)=SITEMP(1)
   SOTEMP(2)=SITEMP(2)
   EQPAR(2,NE)=U
 6 WRITE (6,52) NE
52 FORMAT (1H ,*UNIT NUMBER =*,I5)
   RETURN
50 IF(ABS(R-1.).LT.1.E-3) GO TO 53
   GO TO 54
53 F1=(1.-S)/S
   GO TO 55
54 F1=(R-1.)/ALOG((1.-S)/TEMP)
55 M5=SQRT(R*R+1)
   IF(APRIME.GE.C) GO TO 56
   GO TO 57
56 F2=F1
   GO TO 58
57 IF(ABS(R-1.).LT.1.E-3) GO TO 59
   GO TO 60
59 M2=TEMPO*F1+1.
   GO TO 61
60 M3=(TEMP/(1.-S))**(1./TEMPO)
   M2=(R-M3)/(1.-M3)
61 M4=2.*M2-1.-R
   M6=(M4+M5)/(M4-M5)
   IF(ABS(M4-M5).LT.1.E-5.OR.M6.LE.0.) GO TO 62
   GO TO 63
62 IF(FFLAG)               GO TO 43
   DT=DT*0.96
   GO TO 200
63 F2=M5/ALOG(M6)/TEMPO
58 IF(J.EQ.I) GO TO 65
   GO TO 66
65 DT=F2*(T2-SITEMP(2))
   GO TO 43
66 DT=F2*(T1-SITEMP(1))
43 IF(XMODE.GT.550.) EQPAR(8,NE)=Q/124.95
   EQPAR(I1,NE)=(SIENTH(I)-SOENTH(I))/1.E6
   IF(FFLAG)               GO TO 67
   GO TO 68
67 EQPAR(3,NE)=Q/(U*DT*B)
   RETURN
```

```
  68 IF(FLAG)               GO TO 69
     IF(ICD.EQ.0) GO TO 681
     TEMP=0.
     EQPAR(3,NE)=Q/(U*DT*B)
     AR=EQPAR(3,NE)
     DDT=DT
     GO TO 70
 681 TEMP=(DT-DDT)/DT
     ABTM=ABS(TEMP)
     IF(ABTM.LT.0.001) GO TO 70
     GO TO 71
  70 IF(F2/F1.LT..75) GO TO 72
     IF(ICD.NE.0) GO TO 80
     GO TO 69
  72 WRITE(6,73) NE
  73 FORMAT(/71H *** NO. OF SHELLS IN SERIES OR NO. OF SHELL PASSES SHO
    1ULD BE INCREASED./11H0UNIT NO. =,I5)
     IF(ICD.NE.0) GO TO 80
  69 DO 75 J=1,11
     IF(J.GT.10) GO TO 80
     IF(SAVENE(J).EQ.NE.OR.SAVENE(J).EQ.0) GO TO 77
  75 CONTINUE
  77 SAVEDT(J)=DT
     SAVENE(J)=NE
  80 CONTINUE
     RETURN
  71 IF(COUNT.GE.10) GO TO 400
     IF(TEMP)101,101,110
 101 IF(NNEG.GT.0) GO TO 102
     NNEG=1
     TNEG=DDT
     FNEG=TEMP
 102 IF(NPOS.GT.0) GO TO 120
     DT=DDT*FACTOR
     GO TO 200
 110 IF(NNEG.EQ.0.AND.DDT.GT.(DTMAX*0.979)) GO TO 402
     IF(NPOS.GT.0) GO TO 111
     NPOS=1
     TPOS=DDT
     FPOS=TEMP
     IF(ISW.EQ.0) GO TO 120
 111 IF(NNEG.GT.0) GO TO 120
     DT=DDT/FACTOR
     IF(DT.GT.(DTMAX*0.975)) DT=0.975*DTMAX
     GO TO 200
 120 IF(COUNT.LT.5.OR.IBEST.EQ.1) GO TO 1201
     IF((ABS(FPOS/FNEG)).LT.10.) GO TO 1202
     DT=TPOS+0.4*(TNEG-TPOS)
     IBEST=1
     GO TO 200
1202 IF((ABS(FNEG/FPOS)).LT.10.) GO TO 1201
     DT=TNEG-0.4*(TNEG-TPOS)
     IBEST=1
     GO TO 200
1201 IF(TEMP)121,121,122
 121 T1=TPOS
     F1=FPOS
     GO TO 123
```

```
  122 TI=TNEG
      FI=FNEG
C
C     CALCULATE NEW VALUE BY REGULI-FALSI
C
  123 DT=(DDT*FI-TI*TEMP)/(FI-TEMP)
C
      IF(TEMP)125,125,130
  125 IF(NNEG.GT.1) GO TO 126
      NNEG=2
      GO TO 200
  126 IF(TEMP.LT.FNEG) GO TO 200
      TNEG=DDT
      FNEG=TEMP
      GO TO 200
  130 IF(NPOS.GT.1) GO TO 131
      NPOS=2
      GO TO 200
  131 IF(TEMP.GT.FPOS) GO TO 200
      TPOS=DDT
      FPOS=TEMP
  200 CONTINUE
      DDT=DT
      GO TO 19
C
  400 WRITE(6,401)TEMP,NE
  401 FORMAT(65H *** HEAT EXCHANGER CALC. FAILED TO CONVERGE, CURRENT VA
     1LUES USED,* - FRACTIONAL ERROR=*,F10.5/*  UNIT NUMBER= *,I3)
      GO TO 70
  402 WRITE(6,403)TEMP
  403 FORMAT(*0*,*DT IS GREATER THAN 98 PERCENT OF DTMAX - USE 98 PERCEN
     1T VALUE  -  FRACTIONAL ERROR=*,F10.5)
      GO TO 6
      END
```

```
      SUBROUTINE  MIXR
C     ------
C     CHESS EQUIPMENT SUBROUTINE
C
C      MODIFIED BY M.A. MENZIES
C      MC MASTER UNIVERSITY - 1969
C     ------
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
C
C     ******
C
      DIMENSION DUM(1)
      DIMENSION SIIDUM(8,11),SOIDUM(8,11)
      EQUIVALENCE (SINUM,SIIDUM),(SONUM,SOIDUM)
      SOMOLE(1)=0.
      DO 2 J=1,NOCOMP
    2 SOCOMP(J,1)=0.
      DO 3 I=1,NIN
      SOMOLE(1)=SOMOLE(1)+SIMOLE(I)
      DO 3 J=1,NOCOMP
    3 SOCOMP(J,1)=SOCOMP(J,1)+SICOMP(J,I)
C
      DO 4 I=1,NIN
      IF(SIMOLE(I) .GT. 0.) GO TO 5
    4 CONTINUE
    5 SMALL=SIPRES(I)
      J=I+1
      IF(J.GT.NIN) GO TO 10
      DO 6 I=J,NIN
    6 IF(SIPRES(I).LT.SMALL.AND.SIMOLE(I).GT.0.) SMALL=SIPRES(I)
   10 SOPRES(1)=SMALL
      SOENTH(1)=0.
      DO 7 I=1,NIN
    7 SOENTH(1)=SOENTH(1)+SIENTH(I)
      SMALL=SIVPFR(1)
      I=1
    8 I=I+1
      IF(I.GT.NIN.OR.SIVPFR(I).NE.SMALL) GO TO 9
      GO TO 8
    9 IF(SOTEMP(1).EQ.0.) SOTEMP(1)=560.
      IF(I.GT.NIN) GO TO 17
      DO 12 I=3,11
      SIIDUM(NIN+1,I)=SIIDUM(1,I)
   12 SIIDUM(1,I)=SOIDUM(1,I)
      DO 13 I=1,NOCOMP
      SICOMP(I,NIN+1)=SICOMP(I,1)
```

```
 13 SICOMP(I,1)=SUCOMP(I,1)
    CALL ADBF
    DO 14 I=3,11
 14 SIIDUM(1,I)=SIIDUM(NIN+1,I)
    DO 15 I=1,NOCOMP
 15 SICOMP(I,1)=SICOMP(I,NIN+1)
    RETURN
 17 SOVPFR(1)=SMALL
    CALL TSUBH(-1,SOTEMP(1),DUM)
    RETURN
    END
```

```
      SUBROUTINE VALV
C     ------
C     CHESS EQUIPMENT SUBROUTINE
C
C      MODIFIED BY M.A. MENZIES
C      MC MASTER UNIVERSITY - 1969
C     ------
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLF(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
C
C     ******
C
      DIMENSION SIDUM(8,11),SODUM(8,11)
      EQUIVALENCE (SIDUM,SINUM),(SODUM,SONUM)
      DO 7 I=3,11
    7  IDUM(3,I)=SIDUM(1,I)
      IF(SIPRES(1).LT.EQPAR(2,NE)) GO TO 1
      GO TO 2
    1 WRITE (6,3) NE
    3 FORMAT(30H0*** UPSTREAM PRESSURE TOO LOW,/14H0EQUIPMENT NO.,I4)
      DO 8 I=3,11
    8 SODUM(1,I)=SIDUM(1,I)
      DO 5 I=1,NOCOMP
    5 SOCOMP(I,1)=SICOMP(I,1)
      GO TO 6
    2 SAVE3=EQPAR(2,NE)
      SIPRES(1)=SAVE3
      EQPAR(2,NE)=0.0
      CALL ADBF
      EQPAR(2,NE)=SAVE3
    6 DO 9 I=3,11
    9 SIDUM(1,I)=SIDUM(3,I)
      RETURN
      END
```

```
      SUBROUTINE ADD1
C     ------
C     CHESS EQUIPMENT SUBROUTINE
C
C      WRITTEN BY - M.A. MENZIES
C      MC MASTER UNIVERSITY - 1969
C      ------
C     HENGSTEBECK*S APPROX. PSEUDO-BINARY DISTILLATION PROCEDURE
C     EMPLOYS EFFECTIVE KEYS + USES RELATIVE VOLATILITIES
C     STAGE REQUIREMENTS ARE CALCULATED FROM STOPPELS METHOD
C
C     ******  COMMON DECK  ******
C
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
      COMMON/PRSV/TOP(20),BOT(20)
C
C     ******
C
C     EQUIPMENT PARAMETER LIST
C     1.  NODE NUMBER
C     2.  REFLUX RATIO
C     3.  NUMBER OF IDEAL STAGES IN RECT. SECTION
C     4.  NUMBER OF IDEAL STAGES IN STRIP. SECTION
C     5.  SOLUTION TOLERANCE ON NUMBER OF STAGES
C     6.  ESTIMATED LIGHT KEY MOLE FRACTION  - DISTILLATE
C     7.  ESTIMATED HEAVY KEY MOLE FRACTION  - DISTILLATE
C     8.  ESTIMATED LIGHT KEY MOLE FRACTION  - BOTTOMS
C     9.  ESTIMATED HEAVY KEY MOLE FRACTION  - BOTTOMS
C     10. LIGHT KEY SUBSCRIPT NUMBER
C     11. HEAVY KEY SUBSCRIPT NUMBER
C     02. MODE OF OPERATION-
C                            0. FEED + REFLUX RATIO KNOWN
C                            1. L KNOWN - CALCULATE REFLUX RATIO
C                            2. VBAR KNOWN - CALCULATE REFLUX RATIO
C     13. STORED VALUE FOR  QQ  (MODES 1. + 2. ONLY)
C     14. STORED PREVIOUS VALUE OF D
C     15. STORED PREVIOUS VALUE OF B
C     16. COMPUTED TEMP. ON THEOR. PLATE 7.7
C
C     THIS SUBROUTINE IS AT PRESENT SET UP TO CALCULATE MOLE FLOWS
C      IN PRODUCT LIQUID + VAPOR STREAMS
C     THE ASSUMED MOLE FRACTIONS OF THE KEYS IN THE PRODUCTS ARE USED
C      TO INITIALISE THE CALCULATIONS
C     CONSTANT LIQUID + VAPOR FLOWS IN EACH COLUMN SECTION ARE ASSUMED
C      FOR CALCULATION OF TOTAL PRODUCT FLOWS
C     THE CALCULATION PROCEDURE IS BASED ON THE LOG. RELATION
C      BETWEEN  D(I)/B(I)  AND  ALPHA(I) WHERE-
C       D(I)=V(I)-L(I)
C       B(I)=LBAR(I)-VBAR(I)
C     K-VALUES ARE CALCULATED FOR LIQUID + VAPOR PRODUCT STREAMS
C      AT THEIR BUBBLE + DEW POINTS RESPECTIVELY
C
      DIMENSION DUM(1)
      DIMENSION TKV(20),BKV(20)
      DIMENSION FEED(20)
      DIMENSION ALPHA(20),DBLN(20)
      DIMENSION FACT(2),TSV(2)
      EQUIVALENCE (FEED(1),SICOMP(1,1))
C
      QUAD(A,B,C)=(SQRT(B**2-4.*A*C))/(2.*A)
C
      RR=EQPAR(2,NE)
      TRR=EQPAR(3,NE)
      TRS=EQPAR(4,NE)
      TOLP=EQPAR(5,NE)
      Y1=EQPAR(6,NE)
      Y2=EQPAR(7,NE)
      X1=EQPAR(8,NE)
      X2=EQPAR(9,NE)
      NLK=EQPAR(10,NE)
      NHK=EQPAR(11,NE)
      IMODE=EQPAR(12,NE)
      QQ=EQPAR(13,NE)
      D=EQPAR(14,NE)
      B=EQPAR(15,NE)
      NLOOP=20
C
      SOPRES(1)=SIPRES(1)-2.
      SOPRES(2)=SIPRES(1)+2.
      SOPRES(7)=SIPRES(1)-2.
      SOPRES(8)=SIPRES(1)+2.
      TSV(1)=SOTEMP(1)
      TSV(2)=SOTEMP(2)
      SOTEMP(1)=SITEMP(1)
      SOTEMP(2)=SITEMP(1)
      SOTEMP(7)=SITEMP(1)
      SOTEMP(8)=SITEMP(1)
      SOVPFR(1)=1.0
      SOVPFR(2)=0.0
      SOVPFR(7)=1.0
      SOVPFR(8)=0.0
      DO 3 I=1,NOCOMP
      TOP(I)=0.
    3 BOT(I)=0.
C     ******
C     CALCULATE Q VALUE(EQUIV. TO VAP. FRAC.) FOR ACTUAL FEED
C     USE THIS VALUE FOR THE PSEUDO-BINARY SYSTEM
C
      IF(IMODE.GT.0) GO TO 1
      CALL DEWTP(1,DWT,TKV)
      CALL BUBTP(1,BBT,BKV)
      SAVV=SIVPFR(1)
      SAVT=SITEMP(1)
      SIVPFR(1)=1.0
      SITEMP(1)=DWT
      CALL ENTH(1,HV,DUM)
      SIVPFR(1)=0.0
      SITEMP(1)=BBT
```

```
      CALL ENTH(1,HL,DUM)
      SIVPFR(1)=SAVV
      SITEMP(1)=SAVT
      QQ=(SIFNTH(1)-HL)/(HV-HL)
      EOPAR(13,NE)=QQ
C
C     CALCULATE KEY MOLE FLOWS CORRESPONDING TO GIVEN MOLE FRACTIONS
C
      EOPAR(12,NE)=1.0
      B=(FEED(NLK)-Y1*FEED(NHK)/Y2)/(X1-X2*Y1/Y2)
      D=SIMOLE(1)-B
      GO TO 4
    1 IF(IMODE.GT.1) GO TO 2
      EOPAR(12,NE)=2.0
      GO TO 4
    2 EOPAR(12,NE)=1.0
    4 TOP(NLK)=D*Y1
      TOP(NHK)=D*Y2
      BOT(NLK)=B*X1
      BOT(NHK)=B*X2
      IF(IMODE.EQ.0) GO TO 555
C
C     CALCULATE APPROPRIATE SATURATED REFLUX FLOW BY MASS + ENTHALPY BAL
C
      II=IMODE
      ASSIGN 121 TO JRET
      SOTEMP(II)=TSV(II)
      GO TO 120
  121 RFLOW=SIMOLE(II+1)-DFLOW
      III=II+1
  555 CONTINUE
C
C     CALCULATE INITIAL ALPHA VALUES BASED ON ASSUMED KEYS ONLY
C
      ASSIGN 5 TO NRET
      GO TO 1000
C
C     SET UP LINEAR LN(D/B)  VS  LN(ALPHA) RELATION
C
    5 SLOPE=(ALOG(TOP(NLK)/BOT(NLK))-ALOG(TOP(NHK)/BOT(NHK)))/(ALOG(ALPH
     1A(NLK))-ALOG(ALPHA(NHK)))
      XINT=ALOG(TOP(NHK)/BOT(NHK))-(ALOG(ALPHA(NHK)))*SLOPE
      NIT=1
      ICAL=1
C     ******
C     BEGIN NORMAL CALCULATION MODULE
C
C     CALCULATE LN(D/B)#S FOR ALL COMPS FROM THEIR ALPHA#S
C
    6 DO 7 I=1,NOCOMP
    7 DBLN(I)=SLOPE*ALOG(ALPHA(I))+XINT
C
C     CALCULATE B#S + D#S FOR ALL NON KEYS
C
      D=0.0
      B=0.0
      DO 8 I=1,NOCOMP
      BOT(I)=FEED(I)/(EXP(DBLN(I))+1.)
      TOP(I)=FEED(I)-BOT(I)
      D=D+TOP(I)
      B=B+BOT(I)
    8 CONTINUE
C
C     CALCULATE CRITICAL LN(D/B)#S
C
      DBCL=DBLN(NLK)+.7*(DBLN(NLK)-DBLN(NHK))
      DBCH=DBLN(NHK)-.7*(DBLN(NLK)-DBLN(NHK))
C
C     CALCULATE KEY PORTIONS FOR LIGHT NON KEYS + SUM FOR EFFECTIVE LK
C
      BINFL=FEED(NLK)
      BINDL=TOP(NLK)
      BINBL=BOT(NLK)
      II=NLK-1
      IF(II.EQ.0) GO TO 15
      DO 10 I=1,II
C     11 - IF LN(D/B) IS LESS THAN CRIT VALUE INCLUDE WHOLLY IN LK
C     12 - IF GREATER CALCULATE PORTION TO BE INCLUDED
      IF(DBLN(I)-DBCL)11,11,12
   11 A1=FEED(I)
      A2=TOP(I)
      A3=BOT(I)
      GO TO 13
   12 A2=BOT(I)*EXP(DBCL)
      A3=0.
      A1=A2
   13 BINFL=BINFL+A1
      BINDL=BINDL+A2
      BINBL=BINBL+A3
   10 CONTINUE
C
C     REPEAT FOR HEAVY NON KEYS SIMILARLY
C
   15 BINFH=FEED(NHK)
      BINDH=TOP(NHK)
      BINBH=BOT(NHK)
      II=NHK+1
      IF(II.GE.NOCOMP) GO TO 21
      DO 20 I=II,NOCOMP
C     17 - IF LN(D/B) IS GREATER THAN CRITICAL VALUE INCLUDE WHOLLY IN HK
C     18 - IF LESS CALCULATE PORTION TO BE INCLUDED
      IF(DBCH-DBLN(I))17,17,18
   17 A1=FEED(I)
      A2=TOP(I)
      A3=BOT(I)
      GO TO 19
   18 A2=0.
      A3=TOP(I)*EXP(-DBCH)
      A1=A3
   19 BINFH=BINFH+A1
      BINDH=BINDH+A2
      BINBH=BINBH+A3
   20 CONTINUE
C     ******
C     CALCULATE PARAMETERS FOR EFFECTIVE BINARY
C
```

```
   21 XF=BINFL/(BINFL+BINFH)
      XD=BINDL/(BINDL+BINDH)
      XW=BINBL/(BINBL+BINBH)
      R1=ALOG(BINDL/BINBL)
      R2=ALOG(BINDH/BINBH)
      RV1=EXP((R1-XINT)/SLOPE)
      RV2=EXP((R2-XINT)/SLOPE)
      RVV=RV1/RV2
C     RVV IS THE NORMALISED REL. VOL. OF THE EFFECTIVE LK
C     ******
C     CALCULATE RATIOS TO IMPROVE REL. VOL. ESTIMATES
C     CALCULATE RATIOS CORR. TO THE CENTRES OF THE RECT. + STRIP. SECTS.
C     USE GEOMETRIC MEANS
C
      EX1=TRR/(TRR+TRS)
      EX2=1.0-EX1
      ALFFD=((TKV(NLK)**EX2)*(BKV(NLK)**EX1))/((TKV(NHK)**EX2)*(BKV(NHK)
     1**EX1))
      ALFTP=TKV(NLK)/TKV(NHK)
      ALFBT=BKV(NLK)/BKV(NHK)
      RVR=SQRT(ALFFD*ALFTP)/ALPHA(NLK)
      RVS=SQRT(ALFFD*ALFBT)/ALPHA(NLK)
C     M*****
C     NOW APPLY STOPPELS CALCULATION FOR THEORETICAL STAGES
C
C
C     CALCULATE REFLUX RATIO
C
      IF(IMODE.EQ.1) RR=RFLOW/D
      IF(IMODE.EQ.2) RR=((RFLOW+STMOLE(1)*STVPFR(1))/D)-1.0
C
      R1=RR+1.
      R11=1./R1
      Q1=1.-QQ
C     CALCULATE INTERSECTIONS
C
C     FEED LINE + RECT OL
C
      RV=RVV*RVR
      XQ=(XF/QQ-XD/R1)/(RR/R1+Q1/QQ)
      YQ=XQ*RR*R11+XD*R11
C
C     EQUIL LINE + RECT OL
C
      RV=RVV*RVR
      SS=RR*R11
      XNT=R11*XD
      ASSIGN 94 TO KINT
      GO TO 96
   94 XE=AA+QU
      XO=AA-QU
      YE=SS*XE+XNT
      YO=SS*XO+XNT
C
C     EQUIL LINE + STRIP OL
C
      RV=RVV*RVS
      SS=(YQ-XW)/(XQ-XW)
      XNT=XW*(1.-SS)
      ASSIGN 95 TO KINT
      GO TO 96
   95 XED=AA+QU
      XOD=AA-QU
      YED=SS*XED+XNT
      YOD=SS*XOD+XNT
      IF(YQ.GT.YED)WRITE(6,851)RR
  851 FORMAT(*0BELOW MINIMUM REFLUX - RR=*,F6.3)
C
C     CALCULATE TR TS
C
      TR=ALOG(((XD-YQ)*(XF-XQ))/((YE-XD)*(XQ-XO)))/ALOG((YO*XE)/(XO*YE))
      TS=ALOG(((YQ-YOD)*(XED-XW))/((YED-YQ)*(XW-XOD)))/ALOG((YOD*XED)/(X
     1OD*YED))
      DTRAY=(TRR+TRS)-(TR+TS)
      RATR=TRR/TR
      RATS=TRS/TS
C     ******
C     CONVERGENCE ROUTINE
C
      GO TO (63,67),ICAL
C     ICAL=1 - CHANGE BOTH SLOPE + INTERCEPT
C     ICAL=2 - CHANGE INTERCEPT ONLY
   63 IF(ABS(DTRAY)-TOLP)64,64,65
   64 IF(ABS(RATR-1.)-(TOLP/(2.*TRR)))81,81,61
   61 ICAL=2
      GO TO 71
   67 IF(ABS(RATR-1.)-(TOLP/(2.*TRR)))68,68,66
   68 IF(ABS(DTRAY)-TOLP)81,81,69
   69 ICAL=1
   65 AA=ALOG(BINDL/BINBH)
      CC=(AA-XINT)/SLOPE
      SLOPE=SLOPE*(TRR+TRS)/(TR+TS)
      XINT=AA-SLOPE*CC
      GO TO 70
   66 CC=XINT
C     NEW VALUE OF XINT FROM *REGULI-FALSI*
      XINT=(XXNT*(RATR-1.)-XINT*(RTR-1.))/(RATR-RTR)
      XXNT=CC
      RTR=RATR
      GO TO 70
   71 XXNT=XINT
      RTR=RATR
C     SECOND STARTING VALUE OF XINT FOR *REGULI-FALSI*
      XINT=XINT*RATR
C
C     CALCULATE NEW KEY SPLITS
C
   70 DBL=SLOPE*ALOG(ALPHA(NLK))+XINT
      DBH=SLOPE*ALOG(ALPHA(NHK))+XINT
      BOT(NLK)=FEED(NLK)/(EXP(DBL)+1.)
      BOT(NHK)=FEED(NHK)/(EXP(DBH)+1.)
      TOP(NLK)=FEED(NLK)-BOT(NLK)
      TOP(NHK)=FEED(NHK)-BOT(NHK)
C
C     CALCULATE A NEW SET OF ALPHA VALUES
C
```

```
      ASSIGN 80 TO NRET
      GO TO 1000
   80 NIT=NIT•1
      IF(NIT.GE.NLOOP) GO TO 90
      GO TO 6
   81 CONTINUE
C     ******
C     CALCULATE FINAL PRODUCT COMPONENT MOLE FLOWS
C
      DFACT=RR•1.0
C
C     ON THE FIRST APPLICATOIN OF THE MODEL ASSUME A 10 PERCENT INCREASE
C      IN OVERHEAD VAPOR FLOW DUE TO LIQUID REFLUX FLASH-OFF
C
      IF(IMODE.EQ.0) DFACT=DFACT*1.1
      BFACT=(D*RR•SIMOLE(1)*(1.0-SIVPFR(1)))/B
      DO 82 I=1•NOCOMP
      SOCOMP(I•1)=DFACT*TOP(I)
   82 SOCOMP(I•2)=BFACT*BOT(I)
      SOMOLE(1)=D*DFACT
      SOMOLE(2)=B*BFACT
C
C     CALCULATE PRODUCT TEMPERATURES • ENTHALPIES
C
   88 CALL DEWTP(-1•SOTEMP(1)•TKV)
      CALL BUBTP(-2•SOTEMP(2)•BKV)
      CALL ENTH(-1•SOENTH(1)•DUM)
      CALL ENTH(-2•SOENTH(2)•DUM)
      IF(IMODE.EQ.0) GO TO 130
C
C     ADJUST PRODUCT FLOWS FOR FLASH-OFF ON TOP • BOTTOM TRAYS
C
      FLOW1=SOMOLE(1)
      FLOW2=SOMOLE(2)
      ASSIGN 122 TO JRET
      DO 125 II=1•2
      GO TO 120
  122 SOMOLE(II)=SOMOLE(II)*FACT(II)
      DO 124 I=1•NOCOMP
      SOCOMP(I•II)=SOCOMP(I•II)*FACT(II)
  124 CONTINUE
      III=-II
      CALL ENTH(III•SOENTH(II)•DUM)
  125 CONTINUE
C
C     CALCULATE TEMP. • COMP. ON TRAY 7.7 - USE KEYS ONLY
C
  130 BB=((YO*XE)/(XO*YE))**7.7
      X7=(BB*YE*(XQ-XO)•YO*(XE-XQ))/(BB*(XQ-XO)•(XE-XQ))
      DO 777 I=1•NOCOMP
  777 SOCOMP(I•7)=0.
      SOMOLE(7)=1.0
      SOCOMP(NLK•7)=X7
      SOCOMP(NHK•7)=1.-X7
      SOPRES(7)=SIPRES(1)-1.
      SOVPFR(7)=1.0
      CALL DEWTP(-7•DWT•TKV)
      SOVPFR(7)=0.
      CALL BUBTP(-7•BBT•TKV)
      T7=0.5*(BBT•DWT)
      IF(IMODE.EQ.2) EQPAR(16•NE)=T7
C
C     CALCULATE OVERALL COLUMN HEAT BALANCE
C
      QIN=SIENTH(1)•SIENTH(2)•SIENTH(3)
      QOUT=SOENTH(1)•SOENTH(2)
      QIN=QIN*1.E-6
      QOUT=QOUT*1.E-6
      QGAIN=QOUT-QIN
      WRITE(6•108)IMODE•RR•T7•QIN•QGAIN
  108 FORMAT(/1H •43(1H*)/2H *•I3•F8.4•F9.3•5X•2F7.3•3H  */1H •43(1H*)/)
C
C     PLACE VALUES BACK IN EQUIP. PAR. LIST
C
      EQPAR(2•NE)=RR
      EQPAR(6•NE)=TOP(NLK)/D
      EQPAR(7•NE)=TOP(NHK)/D
      EQPAR(8•NE)=BOT(NLK)/B
      EQPAR(9•NE)=BOT(NHK)/B
      EQPAR(14•NE)=D
      EQPAR(15•NE)=B
      RETURN
C
C     ******
C     INTERNAL FUNCTION TO CALCULATE ALPHA VALUES
C     K-VALUES CALCULATED FOR PRODUCT STREAMS AT BUB./DWT. POINTS
C     ALPHA VALUES THEN FOUND FROM GEOMETRIC MEAN
C
 1000 SUMTP=0.0
      SUMBT=0.0
      DO 85 I=1•NOCOMP
      SOCOMP(I•7)=TOP(I)
      SOCOMP(I•8)=BOT(I)
      SUMTP=SUMTP•TOP(I)
   85 SUMBT=SUMBT•BOT(I)
      SOMOLE(7)=SUMTP
      SOMOLE(8)=SUMBT
      CALL DEWTP(-7•DWT•TKV)
      CALL BUBTP(-8•BBT•BKV)
      ALF=SQRT(TKV(NHK)*BKV(NHK))
      DO 86 I=1•NOCOMP
   86 ALPHA(I)=SQRT(TKV(I)*BKV(I))/ALF
      GO TO NRET•(5•80)
C     ******
C     INTERNAL FUNCTION TO CALC. EQUIL. • OP. LINE INTERSECTIONS
C
   96 A1=SS*(RV-1.)
      A2=SS•XNT*(RV-1.)-RV
      QU=(SQRT(A2**2-4.*A1*XNT))/(2.*A1)
      AA=-0.5*A2/A1
      GO TO KINT•(94•95)
C     ******
C     INTERNAL FUNCTION TO CALCULATE FLASH ON TOP OR BOTTOM TRAYS
C
C     CALCULATE APPROX. TO SPECIFIC ENTHALPY OF SAT. REFLUX STREAM
C
```

```
  120 VSAV=SOVPFR(II)
      IF(II.EQ.1) SOVPFR(II)=0.
      IF(II.EQ.2) SOVPFR(II)=1.
      III=-II
      CALL ENTH(III,HSAT,DUM)
      SOVPFR(II)=VSAV
      HSAT=HSAT/SOMOLE(II)
C
C     CALCULATE CONSTANT OF PROPORTIONALITY BETWEEN GROSS + NET PRODUCT
C
      FACT(II)=(SIMOLE(II+1)-SIENTH(II+1)/HSAT)/(SOMOLE(II)-SOENTH(II)/H
     1SAT)+1.
      DFLOW=SOMOLE(II)*(FACT(II)-1.)
      GO TO JRET,(121,122)
C
   90 WRITE(6,524)NE,NLOOP
  524 FORMAT(*0COLUMN MODEL,NE= *,I2,* NO CONVERGENCE AFTER *,I2,* LOOPS
     1-PRESENT VALUES WILL BE USED*)
      GO TO 81
      END
```

```
      SUBROUTINE ADD2
C     ------
C     CHESS EQUIPMENT SUBROUTINE
C
C      WRITTEN BY - M.A. MENZIES
C      MC MASTER UNIVERSITY - 1969
C      ------
C
C     THIS SUBROUTINE SPLITS THE INPUT STREAM INTO TWO OUTPUTS
C
C     THE COMPONENT FLOWS IN THE FIRST OUTPUT ARE SET EQUAL TO THE
C      COMPUTED NET FLOWS FROM THE COLUMN MODEL
C
C     EQUIPMENT PARAMETER LIST-
C       1.  NODE NUMBER
C       2.  STREAM NUMBER OF FLASH DRUM VAPOR BLEED-OFF
C       3.  1. IF TOP PRODUCT DIVIDER + 0. IF BOTTOM PRODUCT DIVIDER
C
C     ******  COMMON DECK  ******
C
      COMMON/STMA/SEXTSV(23,100),SINTSV(10,100),NSMAX,MAXSEX,MAXSIN
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
      COMMON/PRSV/TOP(20),BOT(20)
      COMMON/SYSC/LIMIT,LIMIT2,LIMIT3,LOOP,LOOPS
C
C     ******
C
      DIMENSION DUM(1)
      DIMENSION SIIDUM(8,5),SOIDUM(8,5)
      EQUIVALENCE (SINUM,SIIDUM),(SONUM,SOIDUM)
      NFLS=EQPAR(2,NE)
      ISW=EQPAR(3,NE)
C
C     WHEN USED AS TOP PRODUCT DIVIDER THE AMOUNT REMOVED IN OUTPUT 1
C      MUST BE ADJUSTED TO TAKE ACCOUNT OF THE VAPOR BLEED-OFF FROM
C      THE FLASH DRUM TO ENSURE MASS BALANCE
C        I.E.  FIRST OUTPUT = COMPUTED D  -  BLEED-OFF
C
      SOMOLE(1)=0.
      SOMOLE(2)=0.
      IF(ISW.NE.1) GO TO 3
      DO 20 I=1,NOCOMP
      SOCOMP(I,1)=TOP(I)-SEXTSV(I+3,NFLS)
      SOCOMP(I,2)=SICOMP(I,1)-SOCOMP(I,1)
      SOMOLE(1)=SOMOLE(1)+SOCOMP(I,1)
   20 SOMOLE(2)=SOMOLE(2)+SOCOMP(I,2)
      GO TO 22
    3 DO 21 I=1,NOCOMP
      SOCOMP(I,1)=BOT(I)
      SOCOMP(I,2)=SICOMP(I,1)-SOCOMP(I,1)

      SOMOLE(1)=SOMOLE(1)+SOCOMP(I,1)
   21 SOMOLE(2)=SOMOLE(2)+SOCOMP(I,2)
   22 DO 1 J=1,2
      DO 1 I=3,5
    1 SOIDUM(J,I)=SIIDUM(1,I)
      CALL ENTH(-1,SOENTH(1),DUM)
      CALL ENTH(-2,SOENTH(2),DUM)
      RETURN
      END
```

```
      SUBROUTINE ADD3
C     ------
C     CHESS EQUIPMENT SUBROUTINE
C
C      WRITTEN BY - M.A. MENZIES
C      MC MASTER UNIVERSITY - 1969
C     ------
C
C     THIS ROUTINE MODELS A SINGLE STAGE VAPOR COMPRESSOR
C     THE COMPRESSION IS ASSUMED POLYTROPIC
C      I.E. (P*V)**GAMMA = CONSTANT
C
C     EQUIPMENT PARAMETER LIST-
C     1. - NODE NUMBER
C     2. - DESIRED OUTLET PRESSURE - PSIA
C     3. - POLYTROPIC COMPRESSION COEFFICIENT
C     4. - ENTHALPY CHANGE   - MM BTU/HR
C     5. - VOLUMETRIC INFLOW  MCF/HR
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSD/KEFLAG(50).KSFLAG(100).KTRACE.DERROR.NPFREQ.IPUNCH
      COMMON/EQPA/EQPAR(25.50).NEMAX.MAXEQP
      COMMON/CONTL/NIN.NOUT.NOCOMP.NE.NEN
      COMMON/STRMIN/SINUM(8).SIFLAG(8).SIVPFR(8).SITEMP(8).
     1SIPRES(8).SIENTH(8).SIVISC(8).SITHK(8).SILZ(8).SIVZ(8).
     2SIMOLE(8).SICOMP(20.8).SIKV(20.8)
      COMMON/STMOUT/SONUM(8).SOFLAG(8).SOVPFR(8).SOTEMP(8).
     1SOPRES(8).SOENTH(8).SOVISC(8).SOTHK(8).SOLZ(8).SOVZ(8).
     2SOMOLE(8).SOCOMP(20.8).SOKV(20.8)
C
C     ******
C
      DIMENSION DUM(1)
C
      F(T.P)=((ZZ*T)**GAM)*(P**GAM1)
C
      DO 1 I=1.NOCOMP
    1 SOCOMP(I.1)=SICOMP(I.1)
      SOMOLE(1)   =SIMOLE(1)
      SOVPFR(1)=SIVPFR(1)
C
      SOPRES(1)=EQPAR(2.NE)
      POUT=SOPRES(1)
      PR=SOPRES(1)/SIPRES(1)
      GAM=EQPAR(3.NE)
      GAM1=1.-GAM
      CALL ZDENS(1.ZZ.DUM)
      ZIN=ZZ
      FIN=F(SITEMP(1).SIPRES(1))
C
C     MAKE FIRST ESTIMATE OF OUTLET TEMP.
C
      ICAL=0
      TOUT=SITEMP(1)*(PR**0.15)
    2 ICAL=ICAL+1
      SOTEMP(1)=TOUT
      CALL ZDENS(-1.ZZ.DUM)
      FOUT=F(TOUT.POUT)
      IF(ICAL.GT.1) GO TO 3
      TSV=TOUT
      ERRSV=FOUT-FIN
      TOUT=SITEMP(1)*(PR**0.35)
      GO TO 2
    3 ERR=FOUT-FIN
      IF((ABS(ERR)/FIN).LT.1.E-4) GO TO 10
C
C     ESTIMATE NEW OUTLET TEMP BY REGULI-FALSI
C
      TT=(TOUT*ERRSV-TSV*ERR)/(ERRSV-ERR)
C
C     SELECT BEST PIVOT
C
      IF((ABS(ERR)).GT.(ABS(ERRSV))) GO TO 4
      TSV=TOUT
      ERRSV=ERR
    4 TOUT=TT
      GO TO 2
C
C     CALCULATE ENTHALPY CHANGE
C
   10 CALL ENTH(-1.SOENTH(1).DUM)
      EQPAR(4.NE)=(SOENTH(1)-SIENTH(1))*1.E-6
C
C     CALCULATE VOLUMETRIC INFLOW
C
      EQPAR(5.NE)=SIMOLE(1)*1.E-3*10.73*ZIN*SITEMP(1)/SIPRES(1)
      RETURN
      END
```

```
      SUBROUTINE ADD4
C     ------
C     CHESS EQUIPMENT SUBROUTINE
C
C      WRITTEN BY - M.A. MENZIES
C      MC MASTER UNIVERSITY - 1969
C     ------
C     THIS ROUTINE ADDS OR REMOVES A CONSTANT HEAT FLUX FROM THE INPUT
C     STREAM + CALCULATES THE STREAM CONDITIONS AT THE NEW ENTHALPY
C
C     EQUIPMENT PARAMETER LIST-
C     1. - NODE NUMBER
C     2. - HEAT FLUX (+ FOR GAIN,- FOR LOSS) - MM BTU/HR
C
C     ****** COMMON DECK ******
C
      COMMON/STMA/SEXTSV(23,100),SINTSV(10,100),NSMAX,MAXSEX,MAXSIN
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
C
C     ******
C
      DIMENSION SIDUM(8,11),SODUM(8,11)
      EQUIVALENCE (SIDUM,SINUM),(SODUM,SONUM)
      DO 1 I=3,11
    1 SIDUM(3,I)=SIDUM(1,I)
      SAVE=EQPAR(2,NE)
      SIENTH(1)=SIENTH(1)+SAVE*1.E6
      EQPAR(2,NE)=0.0
      CALL ADBF
      EQPAR(2,NE)=SAVE
      DO 2 I=3,11
    2 SIDUM(1,I)=SIDUM(3,I)
      RETURN
      END
```

```
      SUBROUTINE ADD5
C     ------
C     CHESS EQUIPMENT SUBROUTINE
C
C      WRITTEN BY - M.A. MENZIES
C      MC MASTER UNIVERSITY - 1969
C     ------
C
C     THIS ROUTINE COMPUTES THE SPLIT OF COMPRESSED VAPOR BETWEEN
C      REBOILER + TRIMMER CONDENSER
C     THE VALUE OF THE SPLIT IS THE CONTROL VARIABLE FOR SYSTEM CONVERGE
C      WHICH IS ALSO HANDLED BY THIS ROUINE
C
C
C     EQUIPMENT PARAMETER LIST -
C      1. - NODE NUMBER
C      2. - SPECIFIED TEMP. ON PLATE II FOR CONVERGENCE
C      3. - INITIAL VALUE OF REBOILER/TRIMMER SPLIT
C      4. - TEMP. ERROR FOR CONVERGENCE
C      5. - 1. FOR PUNCHED CONV. OUTPUT + 0. OTHERWISE
C
C
C     ****** COMMON DECK ******
C
      COMMON/SYSC/LIMIT,LIMIT2,LIMIT3,LOOP,LOOPS
      COMMON/STMA/SEXTSV(23,100),SINTSV(10,100),NSMAX,MAXSEX,MAXSIN
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
C
C     ******
C
      DIMENSION SIIDUM(8,10),SOIDUM(8,10)
      EQUIVALENCE (SINUM,SIIDUM),(SONUM,SOIDUM)
      DATA CHESS,REC/5HCHESS,1H /
C
      IF(LOOP.GT.1) GO TO 1
      IP=EQPAR(5,NE)
      FUNC=0.
      IF(IP.EQ.1) WRITE(7,105)
  105 FORMAT(*LOOP    SPLIT    TEMP      ERROR*)
      NNEG=0
      NPOS=0
      IVAL=0
      XSPLIT=EQPAR(3,NE)
      TERR=EQPAR(4,NE)
      GO TO 23
C
C     HAS FUNCTION CONVERGED FOR ANY PARTICULAR  XSPLIT
C
    1 TEMP=EQPAR(16,2)
      ICON=0
```

74

```
      IF(IVAL.NE.0) GO TO 2
      TT=TEMP
      IVAL=1
      GO TO 19
    2 IF(IVAL.NE.1) GO TO 3
      IVAL=2
      GO TO 31
    3 FNCT=ABS(TEMP-0.5*(TLAST+TT))
      IF(FNCT.LT.(1.5*TERR)) GO TO 4
   31 TLAST=TT
      TT=TEMP
      GO TO 19
    4 IVAL=0
      FUNC=EQPAR(2,NE)-TEMP
      WRITE(6,107)XSPLIT,FUNC
  107 FORMAT(/* SPLIT + TEMP. ERROR*,2F10.4/)
      IF((ABS(FUNC)).LT.TERR) GO TO 50
C
C     FUNCTION HAS CONVERGED FOR PARTICULAR XSPLIT
C
      IF(FUNC)5,5,7
    5 IF(NNEG.GT.0) GO TO 6
      NNEG=1
      XNEG=XSPLIT
      FNEG=FUNC
    6 IF(NPOS.GT.0) GO TO 10
      XSPLIT=XSPLIT+0.015
      GO TO 19
    7 IF(NPOS.GT.0) GO TO 8
      NPOS=1
      XPOS=XSPLIT
      FPOS=FUNC
    8 IF(NNEG.GT.0) GO TO 10
      XSPLIT=XSPLIT-0.015
      GO TO 19
C
   10 IF(FUNC)11,11,12
   11 X1=XPOS
      F1=FPOS
      GO TO 14
   12 X1=XNEG
      F1=FNEG
C
C     CALCULATE NEW VALUE BY REGULI-FALSI
C
   14 XSPT=(XSPLIT*F1-X1*FUNC)/(F1-FUNC)
      ICON=1
C
      IF(FUNC)15,15,17
   15 IF(NNEG.GT.1) GO TO 16
      NNEG=2
      GO TO 19
   16 IF(FUNC.LT.FNEG) GO TO 19
      XNEG=XSPLIT
      FNEG=FUNC
      GO TO 19
   17 IF(NPOS.GT.1) GO TO 18
      NPOS=2
      GO TO 19
   18 IF(FUNC.GT.FPOS) GO TO 19
      XPOS=XSPLIT
      OPOS=FUNC
C
   19 CONTINUE
      IF(IP.EQ.1) WRITE(7,101)LOOP,XSPLIT,TEMP,FUNC
  101 FORMAT(I3,F8.4,F10.3,F8.4)
      IF(ICON.NE.1) GO TO 23
      XSPLIT=XSPT
C
   23 DO 21 J=1,2
      IF(J.EQ.1) XX=XSPLIT
      IF(J.EQ.2) XX=1.-XSPLIT
      DO 20 I=3,10
   20 SOIDUM(J,I)=SIIDUM(1,I)
      SOENTH(J)=SIENTH(1)*XX
      SOMOLE(J)=SIMOLE(1)*XX
      DO 21 I=1,NOCOMP
   21 SOCOMP(I,J)=SICOMP(I,1)*XX
      RETURN
   50 EQPAR(3,NE)=XSPLIT
      WRITE(6,102) XSPLIT
  102 FORMAT(//1H ,*SYSTEM CONVERGED AT SPLIT OF *,F8.4)
      CALL OVERLAY(CHESS,4,0,REC)
      END
```

APPENDIX IV

CHESS SYSTEM LISTINGS

Listings of all CHESS executive subroutines are presented. Equipment subroutines from the original CHESS system which have been used in the present simulation are listed in Appendix III. Subroutines ADBF and HXER have been modified for the present application and their original forms are not given. The remaining original equipment subroutines are presented here.

CHESS SIMULATION SYSTEM

CDC6400 OVERLAY VERSION

PROGRAMS MAIN00 - MAIN40 CORRESPOND TO OVERLAYS 00 - 40
AND HANDLE THE OVERALL SEQUENCING OF SYSTEM CALCULATIONS

```
      PROGRAM MAIN00 (INPUT=1001,OUTPUT,PUNCH=1001,TAPE5=INPUT,TAPE6=OUT
     1PUT,TAPE7=PUNCH)
C
C     ******  COMMON DECK  ******
C
      COMMON /SYSA/KPM(10,50),KSEM(3,100),N3MAX
      COMMON /SYSAA/TITLE(20),COMPNT(20),KOMNAM(80)
      COMMON /SYSB/KE1(50),NE1MAX,KF2(50),NE2MAX,KF3(50),NE3MAX,
     1KF4(10),NE4MAX,KRET,KRET2,KRET3
      COMMON /SYSC/LIMIT,LIMIT2,LIMIT3,LOOP,LOOPS
      COMMON /SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON /EQPB/NECALL(50),NEXFUN(50),NAME(50)
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON /STMA/SEXTSV(23,100),SINTSV(10,100),NSMAX,MAXSEX,MAXSIN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
      COMMON /PHD/APC(20),ATC(20),AVC(20),AMW(20),AOMEG(20),ADEL(20),
     1AVW(20),APH(20),BET(20),GAM(20),DTA(20),EXFLAG
      COMMON /KHSAV/BASEA(20),BASEB(20),ZCD(20),ALD(20)
      COMMON /HXMS/NESAVE(20)
      COMMON /MTST/ISAVEX(400),SAVEQ(200)
      COMMON /OVR/NRET
C
C     ******
C
      INTEGER COMPNT,TITLE
      LOGICAL EXFLAG
C
      CALL SUB00
      END
```

```
      PROGRAM MAIN10
      DATA CHESS,REC /5HCHESS,1H /
      CALL DREAD1
      CALL OVERLAY(CHESS,2,0,REC)
      END
```

```
      SUBROUTINE SUB00
      DATA CHESS,REC /5HCHESS,1H /
      CALL OVERLAY(CHESS,1,0,REC)
      END
```

```
      PROGRAM MAIN20
      DATA CHESS,REC /5HCHESS,1H /
      CALL COMPID
      CALL INIT
      CALL DCHECK(IRET)
      IF(IRET.EQ.0) GO TO 1
      WRITE(6,2)
    2 FORMAT(18H DCHECK ERROR-EXIT)
      CALL EXIT
    1 CALL DPRINT
      CALL OVERLAY(CHESS,3,0,REC)
      END
```

```
      PROGRAM MAIN30
      CALL SUBSET
      END
```

```
      PROGRAM MAIN31
      COMMON /OVR/NRET
      CALL MSEQ
      GO TO (1,2),NRET
    1 CALL RCYOV
    2 CALL SETOV
      END
```

```
      PROGRAM MAIN32
      COMMON /OVR/NRET
      CALL AD15
      GO TO (1,2),NRET
    1 CALL RCYOV
    2 CALL SETOV
      END
```

```
      PROGRAM MAIN33
      COMMON /OVR/NRET
      CALL AD16
      GO TO (1,2),NRET
    1 CALL RCYOV
    2 CALL SETOV
      END
```

```
      PROGRAM MAIN34
      COMMON /OVR/NRET
      CALL AD17
      GO TO (1,2),NRET
    1 CALL RCYOV
    2 CALL SETOV
      END
```

```
      PROGRAM MAIN35
      COMMON /OVR/NRET
      CALL AD18
      GO TO (1,2),NRET
    1 CALL RCYOV
    2 CALL SETOV
      END
```

```
      PROGRAM MAIN36
      COMMON /OVR/NRET
      CALL AD19
      GO TO (1,2),NRET
    1 CALL RCYOV
    2 CALL SETOV
      END
```

```
      PROGRAM MAIN40
      CALL BIPRNT
      CALL PTEQPT(1)
      CALL SUB00
      END
```

```
      SUBROUTINE ABSR
C
C     ****** COMMON DECK ******
C
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
C
C     ******
C
      DIMENSION DUM(1)
      INTEGER COUNT
      LOGICAL FLAG
      REAL    EQR(20),KS(20),LOVRV
      CALL ZERO(EQR,20)
      SOVPFR(1)=1.
      SOVPFR(2)=0.
      DT=2.
      IF(SITEMP(1).LT.SITEMP(2)) DT=-2.
      SOTEMP(1)=SITEMP(2) +DT
      SOTEMP(2)=SITEMP(2)
      TEMP=SIPRES(2)
      IF(SIPRES(1).LT.SIPRES(2)) TEMP=SIPRES(1)
      SOPRES(1)=TEMP
      SOPRES(2)=TEMP
      HIN=SIENTH(1)+SIENTH(2)
      NP1=EQPAR(2,NE) + 1.1
      SOMOLE(1)=SIMOLE(1)
      SOMOLE(2)=SIMOLE(2)
      DO 2 I=1,NOCOMP
      SOCOMP(I,1)=SICOMP(I,1)
    2 SOCOMP(I,2)=SICOMP(I,2)
      CALL KVAL(1,ANS,KS)
      DO 9 COUNT=1,30
      CALL FNTH(-1,SOENTH(1),DUM)
      SOENTH(2)=HIN-SOENTH(1)
      CALL TSUBH(-2,SOTEMP(2),DUM)
      CALL KVAL(-2,ANS,KS)
C** ABSORB. ROUTINE STARTS HERE .....
      LOVRV=SOMOLE(2)/SIMOLE(1)
      SUM=0.
      DO  5 I=1,NOCOMP
      TEMP=LOVRV/KS(I)
      IF(TEMP.GT.50.) TEMP=50.
      TEMPO=TEMP**NP1
      TEMPO=(TEMPO-TEMP)/(TEMPO-1.)
      IF(TEMPO.GT.1.0) TEMPO=1.
      TEMP=SICOMP(I,1)*TEMPO
      SOCOMP(I,1)=SICOMP(I,1)-TEMP
      SOCOMP(I,2)=SICOMP(I,2)+TEMP
    5 SUM=SUM+TEMP
      SOMOLE(1)=SIMOLE(1) -SUM
      SOMOLE(2)=SIMOLE(2) +SUM
C**ABSORB. ROUTINE COMPLETE  **
      FLAG=.TRUE.
      DO 7 I=1,NOCOMP
      IF(ABS(KS(I)-EQR(I))/KS(I).GT.1.E-5) FLAG=.FALSE.
    7 EQR(I)=KS(I)
      IF(FLAG) RETURN
    9 CONTINUE
      WRITE(  6,11) NE
   11 FORMAT(43H0*** ABSORBER CALCULATION FOR EQUIPMENT NO.,I3,46H DID N
     2OT CONVERGE. CURRENT VALUES WILL BE USED)
      RETURN
      END
```

```
      SUBROUTINE CLEAN
C
C..THIS SUBROUTINE ZEROS FOLLOWING COMMON STORAGES BY CALLING SUBROUTINE
C..   *ZEROX(IARRAY,ISIZE)*.
C
C     ****** COMMON DECK ******
C
      COMMON/SYSAA/TITLE(20),COMPNT(20),KOMNAM(80)
      COMMON/SYSA/KPM(10,50),KSEM(3,100),N3MAX
      COMMON/SYSB/KE1(50),NE1MAX,KE2(50),NE2MAX,KE3(50),NE3MAX,
     1KE4(10),NE4MAX,KRET,KRET2,KRET3
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/EQPB/NECALL(50),NEXEQN(50),NAME(50)
      COMMON/HXMS/NESAVE(20)
      COMMON/STMA/SEXTSV(23,100),SINTSV(10,100),NSMAX,MAXSEX,MAXSIN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
      COMMON/MTST/ISAVEX(400),SAVEQ(200)
      COMMON/PHD/APC(20),ATC(20),AVC(20),AMW(20),AOMEG(20),ADEL(20),
     1AVW(20),APH(20),BET(20),GAM(20),DTA(20),EXFLAG
C
C     ******
C
      INTEGER TITLE,COMPNT
      CALL ZEROX(TITLE,120)
      CALL ZEROX(KPM,801)
      CALL ZEROX(KE1,167)
      CALL ZEROX(KEFLAG,154)
      CALL ZEROX(EQPAR,1252)
      CALL ZEROX(NECALL,150)
      CALL ZEROX(SEXTSV,3303)
      CALL ZEROX(SINUM,408)
      CALL ZEROX(SONUM,408)
      CALL ZEROX(NESAVE,20)
      CALL ZEROX(ISAVEX,400)
      DO 1 I=1,200
      SAVEQ(I)=1.0
      CALL ZEROX(APC,220)
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DCHECK(IRET)
C     ****** COMMON DECK ******
C
      COMMON/SYSA/KPM(10,50),KSEM(3,100),N3MAX
      COMMON/STMA/SEXTSV(23,100),SINTSV(10,100),NSMAX,MAXSEX,MAXSIN
C
C     ******
C
C
C
      LOGICAL FLAG
      FLAG=.FALSE.
      J=0
      DO 10 I=1,NSMAX
      IF(KSEM(1,I).EQ.0) GO TO 10
      IF(KSEM(2,I).EQ.0.AND.KSEM(3,I).EQ.0) GO TO 2
      IF(KSEM(2,I).EQ.0.AND..NOT.(ABS(SINTSV(2,I)-1.).LT.0.001.OR.
     1 ABS(SINTSV(2,I)-3.).LT.0.001))  GO TO 4
      IF(KSEM(3,I).EQ.0.AND.ABS(SINTSV(2,I)-2.).GT.0.001) GO TO 6
   10 CONTINUE
      GO TO 8
    2 WRITE( 6,22) I
    3 FLAG=.TRUE.
      GO TO 10
    4 WRITE( 6,24) I
      GO TO 3
    6 WRITE( 6,26) I
      GO TO 3
    8 IF(.NOT.FLAG) GO TO 30
      WRITE( 6,28)
      IRET=1
      RETURN
   30 IRET=0
      RETURN
   22 FORMAT(12H0 STREAM NO.,I4,30H HAS NO CONNECTING EQUIPMENTS.)
   24 FORMAT(12H0 STREAM NO.,I4,104H APPEARS AS A FEED STREAM IN THE PRO
     1CESS MATRIX BUT IS NOT CODED AS SUCH IN THE STREAM VARIABLES MATRI
     2X.)
   26 FORMAT(12H0 STREAM NO.,I4,106H APPEARS INTHE PROCESS MATRIX AS A P
     1RODUCT STREAM BUT IS NOT CODED AS SUCH IN THE STREAM VARIABLES MAT
     2RIX.)
   28 FORMAT(1H0,///54H *** COMPUTATION FOR THIS DATA SET WILL BE TERMIN
     1ATED.)
      END
```

```
      SUBROUTINE  DIST
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
C
C     ******
C
      DIMENSION DUM(1)
      DO 18 NSN=1,NOCOMP
      NSS=NSN+3
      SOCOMP(NSN,1)=SICOMP(NSN,1)*EQPAR(NSS,NE)
   18 SOCOMP(NSN,2)=SICOMP(NSN,1)-SOCOMP(NSN,1)
C
C  CALCULATE AMOUNT OF OVERHEADS AND BOTTOMS
   19 SOMOLE(1)=0.
   20 SOMOLE(2)=0.
      DO 23 NSN=1,NOCOMP
      DO 23  I =1,2
   23 SOMOLE(I)=SOMOLE(I)+SOCOMP(NSN,I)
      SOVPFR(1)=1.
      SOVPFR(2)=0.
      DO 24 I=1,2
      SOTEMP(I)=SITEMP(1)
      SOPRES(I)=SIPRES(1)
      CALL ENTH(-I,SOENTH(I),DUM)
   24 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DPRINT
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSAA/TITLE(20),COMPNT(20),KOMNAM(80)
      COMMON/SYSA/KPM(10,50),KSEM(3,100),N3MAX
      COMMON/SYSB/KE1(50),NE1MAX,KE2(50),NE2MAX,KE3(50),NE3MAX,
     1KE4(10),NE4MAX,KRET,KRET2,KRET3
      COMMON/SYSC/LIMIT,LIMIT2,LIMIT3,LOOP,LOOPS
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/EQPB/NECALL(50),NEXEQN(50),NAME(50)
      COMMON/STMA/SEXTSV(23,100),SINTSV(10,100),NSMAX,MAXSEX,MAXSIN
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
C
C     ******
C
      INTEGER COMPNT,TITLE
C
   18 WRITE(6,19)
   19 FORMAT(1H1,/40X,46(1H*)/40X,2H**,42X,2H**,/40X,2H**,18X,5HCHESS,
     119X,2H**,/40X,2H**,42X,2H**,/40X,2H**,42X,2H**,/40X,2H**,2X,38HCHE
     2MICAL ENGINEERING SIMULATION SYSTEM,2X,2H**,/40X,2H**,42X,2H**,/40
     3X,2H**,42X,2H**,/40X,2H**,4X,34HDEVELOPED AT UNIVERSITY OF HOUSTON
     4,4X,2H**,/40X,2H**,5X,31HFOR OPERATION ON IBM SYSTEM 360,6X,2H**,/
     540X,2H**,42X,2H**,/40X,2H**,5X,32HMODIFIED AT MC MASTER UNIVERSITY
     6,5X,2H**,/40X,2H**,5X,32HFOR OPERATION ON CDC SYSTEM 6400,5X,2H**,
     7/40X,2H**,42X,2H**,/40X,46(1H*)/40X,46(1H*)//////)
C
C**PRINT PROCESS MATRIX HEADING, THEN PROCESS MATRIX
      WRITE(  6,22) TITLE
   22 FORMAT(*1*,20A4//18X,***PROCESS VECTORS***//* ...... EQUIPMENT ...
     1...*,13X,*STREAM NUMBERS*/* NUMBER  SUBROUTINE  NAME* )
      DO 24  I=1,NEMAX
      IF(KPM(1, I).LE.0) GO TO 24
      WRITE(  6,26) KPM(1,I),NAME(I),NEXEQN(I),(KPM(J,I),J=2,N3MAX)
   24 CONTINUE
   26 FORMAT(/I5,7X,A4,5X,A4,6X,9I5)
C
C**WRITE STREAM AND EQUIPMENTS CONNECTION
      WRITE(  6,27) TITLE
   27 FORMAT(*1*,20A4//23X,***STREAM CONNECTIONS***//22X,*STREAM*,4X,*EQ
     1UIPMENT*/30X,*FROM     TO* )
      DO 28 I=1,NSMAX
      IF(KSEM(1,I).EQ.0) GO TO 28
      WRITE(  6,29) (KSEM(J,I),J=1,3)
   28 CONTINUE
   29 FORMAT(/20X,I6,2I7)
      WRITE(6,30) TITLE,NOCOMP,NE2MAX,NE3MAX,(COMPNT(I),I=1,NOCOMP)
   30 FORMAT(*1*,20A4//12X,***OTHER SYSTEM VARIABLES***//
     1* NUMBER OF COMPONENTS*,24X,I4//* NUMBER OF ITEMS IN RECYCLE LIST(
     2KE2)*,8X,I4/* NUMBER OF ITEMS IN RECYCLE LIST(KE3)*,8X,I4//
     3* COMPONENT NUMBERS USED*,24X,I2,9(1H,,I2)/(46X,I2,9(1H,,I2)))
C
      I=2
      IF(NE2MAX.NE.0) WRITE(  6,40) I,(KE2(J),J=1,NE2MAX)
      I=3
      IF(NE3MAX.NE.0) WRITE(  6,40) I,(KE3(J),J=1,NE3MAX)
      IF(NE4MAX.NE.0) WRITE(  6,41)   (KE4(J),J=1,NE4MAX)
   41 FORMAT(*0STREAMS USED IN CONV. ROUTINE(KE4)*,12X,10(I2,1H,))
   40 FORMAT(*0RECYCLE LIST KE*,I1,30X,25(I2,1H,)/47X,25(I2,1H,))
      WRITE(  6,42) DERROR, LOOPS
   42 FORMAT(*0TOLERANCE, **DERROR***,25X,F10.4//* MAX. LOOPS IN RECYCLE
     1 CALC.*,17X,I4)
C
C**PRINT OUT REMAINING INPUT DATA BY CALLING *PTPRNT*.
C
      IF(KTRACE.EQ.-3) RETURN
      IF(KTRACE.NE.-1) CALL PTPRNT
      IF(KTRACE.NE.-2) CALL PTEOPT(0)
      RETURN
      END
```

```
      SUBROUTINE DREAD1
C... TYPE 1 SHOULD CONTAIN WORD *CLEAN* IN COL. 1-5.
C... ALL OTHERS ARE TREATED AS TYPE 2 AND DO NOT REQUIRE /PMLIST/.
C
C     ****** COMMON DECK ******
C
      COMMON/SYSAA/TITLE(20),COMPNT(20),KOMNAM(80)
      COMMON/SYSB/KE1(50),NE1MAX,KE2(50),NE2MAX,KE3(50),NE3MAX,
     1KE4(10),NE4MAX,KRET,KRET2,KRET3
      COMMON/SYSC/LIMIT,LIMIT2,LIMIT3,LOOP,LOOPS
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPB/NECALL(50),NEXEQN(50),NAME(50)
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/PHD/APC(20),ATC(20),AVC(20),AMW(20),AOMEG(20),ADEL(20),
     1AVW(20),APH(20),BET(20),GAM(20),DTA(20),EXFLAG
C  AVAILABLE *30* EQUIPMENT SUBROUTINE NAMES.
      INTEGER NMLIST(30)
      DATA NMLIST/4HDVDR,4HDIST,4HMIXR,4HADBF,4HREAC,4HVALV,4HHXER,4HPUM
     1P,4HABSR,4HMSEQ,4HFHTR,4HADD1,4HADD2,4HADD3,4HADD4,4HADD5,4HADD6,4
     1HADD7,4HADD8,4HADD9,4HAD10,4HAD11,4HAD12,4HAD13,4HAD14,4HAD15,4HAD
     116,4HAD17,4HAD18,4HAD19/
C
C  FOLLOWING DUMMY ARRAYS ARE USED FOR *NAMELIST* RIGID-INPUT OF
C      *KPM*,*EQPAR*,*SEXTSV*,AND *SINTSV* ARRAYS.
C
      COMMON/SYSA/ KPM1(10),KPM2(10),KPM3(10),KPM4(10),KPM5(10),KPM6(10)
     1,KPM7(10),KPM8(10),KPM9(10),KPM10(10),KPM11(10),KPM12(10),KPM13(10
     2 ),KPM14(10),KPM15(10),KPM16(10),KPM17(10),KPM18(10),KPM19(10),
     3 KPM20(10),KPM21(10),KPM22(10),KPM23(10),KPM24(10),KPM25(10),
     4 KPM26(10),KPM27(10),KPM28(10),KPM29(10),KPM30(10),KPM31(10),
     5 KPM32(10),KPM33(10),KPM34(10),KPM35(10),KPM36(10),KPM37(10),
     6 KPM38(10),KPM39(10),KPM40(10),KPM41(10),KPM42(10),KPM43(10),
     7 KPM44(10),KPM45(10),KPM46(10),KPM47(10),KPM48(10),KPM49(10),
     8 KPM50(10),KSEM(3,100),N3MAX
C
      COMMON/EQPA/ EQP1(25),EQP2(25),EQP3(25),EQP4(25),EQP5(25),EQP6(25)
     1,EQP7(25),EQP8(25),EQP9(25),EQP10(25),EQP11(25),EQP12(25),EQP13(25
     2 ),EQP14(25),EQP15(25),EQP16(25),EQP17(25),EQP18(25),EQP19(25),
     3 EQP20(25),EQP21(25),EQP22(25),EQP23(25),EQP24(25),EQP25(25),
     4 EQP26(25),EQP27(25),EQP28(25),EQP29(25),EQP30(25),EQP31(25),
     5 EQP32(25),EQP33(25),EQP34(25),EQP35(25),EQP36(25),EQP37(25),
     6 EQP38(25),EQP39(25),EQP40(25),EQP41(25),EQP42(25),EQP43(25),
     7 EQP44(25),EQP45(25),EQP46(25),EQP47(25),EQP48(25),EQP49(25),
     8 EQP50(25),NEMAX,MAXEQP
      COMMON/STMA/ SEX1(23),SEX2(23),SEX3(23),SEX4(23),SEX5(23),SEX6(23)
     1,SEX7(23),SEX8(23),SEX9(23),SEX10(23),SEX11(23),SEX12(23),
     2 SEX13(23),SEX14(23),SEX15(23),SEX16(23),SEX17(23),SEX18(23),
     3 SEX19(23),SEX20(23),SEX21(23),SEX22(23),SEX23(23),SEX24(23),
     4 SEX25(23),SEX26(23),SEX27(23),SEX28(23),SEX29(23),SEX30(23),
     5 SEX31(23),SEX32(23),SEX33(23),SEX34(23),SEX35(23),SEX36(23),
     6 SEX37(23),SEX38(23),SEX39(23),SEX40(23),SEX41(23),SEX42(23),
     7 SEX43(23),SEX44(23),SEX45(23),SEX46(23),SEX47(23),SEX48(23),
     8 SEX49(23),SEX50(23),SEX51(23),SEX52(23),SEX53(23),SEX54(23),
     9 SEX55(23),SEX56(23),SEX57(23),SEX58(23),SEX59(23),SEX60(23),
     A SEX61(23),SEX62(23),SEX63(23),SEX64(23),SEX65(23),SEX66(23),
     B SEX67(23),SEX68(23),SEX69(23),SEX70(23),SEX71(23),SEX72(23),
     C SEX73(23),SEX74(23),SEX75(23),SEX76(23),SEX77(23),SEX78(23),
     D SEX79(23),SEX80(23),SEX81(23),SEX82(23),SEX83(23),SEX84(23),
     E SEX85(23),SEX86(23),SEX87(23),SEX88(23),SEX89(23),SEX90(23),
     F SEX91(23),SEX92(23),SEX93(23),SEX94(23),SEX95(23),SEX96(23),
     G SEX97(23),SEX98(23),SEX99(23),SEX100(23)
      COMMON/STMA/ SIN1(10),SIN2(10),SIN3(10),SIN4(10),SIN5(10),SIN6(10)
     1,SIN7(10),SIN8(10),SIN9(10),SIN10(10),SIN11(10),SIN12(10),
     2 SIN13(10),SIN14(10),SIN15(10),SIN16(10),SIN17(10),SIN18(10),
     3 SIN19(10),SIN20(10),SIN21(10),SIN22(10),SIN23(10),SIN24(10),
     4 SIN25(10),SIN26(10),SIN27(10),SIN28(10),SIN29(10),SIN30(10),
     5 SIN31(10),SIN32(10),SIN33(10),SIN34(10),SIN35(10),SIN36(10),
     6 SIN37(10),SIN38(10),SIN39(10),SIN40(10),SIN41(10),SIN42(10),
     7 SIN43(10),SIN44(10),SIN45(10),SIN46(10),SIN47(10),SIN48(10),
     8 SIN49(10),SIN50(10),SIN51(10),SIN52(10),SIN53(10),SIN54(10),
     9 SIN55(10),SIN56(10),SIN57(10),SIN58(10),SIN59(10),SIN60(10),
     A SIN61(10),SIN62(10),SIN63(10),SIN64(10),SIN65(10),SIN66(10),
     B SIN67(10),SIN68(10),SIN69(10),SIN70(10),SIN71(10),SIN72(10),
     C SIN73(10),SIN74(10),SIN75(10),SIN76(10),SIN77(10),SIN78(10),
     D SIN79(10),SIN80(10),SIN81(10),SIN82(10),SIN83(10),SIN84(10),
     E SIN85(10),SIN86(10),SIN87(10),SIN88(10),SIN89(10),SIN90(10),
     F SIN91(10),SIN92(10),SIN93(10),SIN94(10),SIN95(10),SIN96(10),
     G SIN97(10),SIN98(10),SIN99(10),SIN100(10),NSMAX,MAXSEX,MAXSIN
C
C
C
C     ******
C
      LOGICAL EXFLAG
      INTEGER SNAME(100),FNAME(50),TITLE,COMPNT
      DIMENSION KNM(4)
C
      DIMENSION KPM(10,50),EQPAR(25,50),SEXTSV(23,100),SINTSV(10,100)
      EQUIVALENCE (KPM,KPM1),(EQPAR,EQP1),(SEXTSV, SEX1),(SINTSV, SIN1)
C
      NAMELIST/PMLIST/  KPM1 ,KPM2 ,KPM3 ,KPM4 ,KPM5 ,KPM6 ,KPM7 ,KPM8 ,
     1KPM9 ,KPM10,KPM11,KPM12,KPM13,KPM14,KPM15,KPM16,KPM17,KPM18,KPM19,
     2KPM20,KPM21,KPM22,KPM23,KPM24,KPM25,KPM26,KPM27,KPM28,KPM29,KPM30,
     3KPM31,KPM32,KPM33,KPM34,KPM35,KPM36,KPM37,KPM38,KPM39,KPM40,KPM41,
     4KPM42,KPM43,KPM44,KPM45,KPM46,KPM47,KPM48,KPM49,KPM50,
     5 NOCOMP,COMPNT,KOMNAM,IPUNCH
C
      NAMELIST/EQLIST/ FNAME,EQP1,EQP2,EQP3,EQP4,EQP5,EQP6,EQP7,EQP8,
     1 EQP9,EQP10,EQP11,EQP12,EQP13,EQP14,EQP15,EQP16,EQP17,EQP18,EQP19,
     2EQP20,EQP21,EQP22,EQP23,EQP24,EQP25,EQP26,EQP27,EQP28,EQP29,EQP30,
     3EQP31,EQP32,EQP33,EQP34,EQP35,EQP36,EQP37,EQP38,EQP39,EQP40,EQP41,
     4EQP42,EQP43,EQP44,EQP45,EQP46,EQP47,EQP48,EQP49,EQP50
C
      NAMELIST/SEXLST/ SNAME,SEX1,SEX2,SEX3,SEX4,SEX5,SEX6,SEX7,SEX8,
     1 SEX9 ,SEX10,SEX11,SEX12,SEX13,SEX14,SEX15,SEX16,SEX17,SEX18,SEX19
     2,SEX20,SEX21,SEX22,SEX23,SEX24,SEX25,SEX26,SEX27,SEX28,SEX29,SEX30
     3,SEX31,SEX32,SEX33,SEX34,SEX35,SEX36,SEX37,SEX38,SEX39,SEX40,SEX41
     4,SEX42,SEX43,SEX44,SEX45,SEX46,SEX47,SEX48,SEX49,SEX50,SEX51,SEX52
     5,SEX53,SEX54,SEX55,SEX56,SEX57,SEX58,SEX59,SEX60,SEX61,SEX62,SEX63
     6,SEX64,SEX65,SEX66,SEX67,SEX68,SEX69,SEX70,SEX71,SEX72,SEX73,SEX74
     7,SEX75,SEX76,SEX77,SEX78,SEX79,SEX80,SEX81,SEX82,SEX83,SEX84,SEX85
     8,SEX86,SEX87,SEX88,SEX89,SEX90,SEX91,SEX92,SEX93,SEX94,SEX95,SEX96
     9,SEX97,SEX98,SEX99,SEX100
C
      NAMELIST/SINLST/ SNAME,SIN1,SIN2,SIN3,SIN4,SIN5,SIN6,SIN7,SIN8,
     1 SIN9 ,SIN10,SIN11,SIN12,SIN13,SIN14,SIN15,SIN16,SIN17,SIN18,SIN19
```

```
     2,SIN20,SIN21,SIN22,SIN23,SIN24,SIN25,SIN26,SIN27,SIN28,SIN29,SIN30
     3,SIN31,SIN32,SIN33,SIN34,SIN35,SIN36,SIN37,SIN38,SIN39,SIN40,SIN41
     4,SIN42,SIN43,SIN44,SIN45,SIN46,SIN47,SIN48,SIN49,SIN50,SIN51,SIN52
     5,SIN53,SIN54,SIN55,SIN56,SIN57,SIN58,SIN59,SIN60,SIN61,SIN62,SIN63
     6,SIN64,SIN65,SIN66,SIN67,SIN68,SIN69,SIN70,SIN71,SIN72,SIN73,SIN74
     7,SIN75,SIN76,SIN77,SIN78,SIN79,SIN80,SIN81,SIN82,SIN83,SIN84,SIN85
     8,SIN86,SIN87,SIN88,SIN89,SIN90,SIN91,SIN92,SIN93,SIN94,SIN95,SIN96
     9,SIN97,SIN98,SIN99,SIN100
C
      NAMELIST/KELIST/ KE2,KE3,KE4,LOOPS,NPFREQ,KTRACE,DERROR
      NAMELIST/FLLIST/KEFLAG,KSFLAG
      NAMELIST/NSCOMP/APC,ATC,AVC,AMW,AOMEG,ADEL,AVW,APH,BET,GAM,DTA
C
      INTEGER CHX(2)
      DATA CHX/4HCLEA,1HN/
      DIMENSION ICARD(20)
C     READ FIRST CARD  -  IS*CLEAN* OPTION REQUIRED
      READ(5,17)ICARD
      IF(EOF,5)1,2
    1 CALL EXIT
    2 IF(ICARD(1).EQ.CHX(1).AND.ICARD(2).EQ.CHX(2)) GO TO 3
      ITYPE=2
      GO TO 4
    3 ITYPE=1
      CALL CLEAN
      WRITE(6,5)
    5 FORMAT(18H DATA AREAS ZEROED)
    4 CONTINUE
C
C... READ ALPHANUMERIC HEADER CARD *20A4* FIELD (*TITLE(20)*)
C
   17 FORMAT(20A4)
C... *CLEAN*          = TYPE 1
C...  OTHERS          = TYPE 2
   23 READ(  5,17) TITLE
      WRITE(  6,17) TITLE
      IF(ITYPE.NE.1) GO TO 63
C
C..READ REMAINING CONTROL CONSTANTS, COMPONENT ID, AND PROCESS MATRIX
C..     FROM  NAMELIST/PMLIST/
C
      READ(  5,PMLIST)
C     ****
C     THIS SECTION READS ALPHAMERIC DATA INTO PMLIST (KPM + KOMNAM)
C     CHANGE IS NECESSARY AS CDC6400 NAMELIST DOES NOT ACCEPT ALPHA DATA
C
      NN=0
      DO 150 I=1,50
      IF(KPM(1,I).NE.0) NN=NN+1
  150 CONTINUE
      DO 151 I=1,NN
  151 READ(5,152)J,KPM(2,J),KPM(3,J)
  152 FORMAT(I2,2X,A4,2X,A4)
C
      NN=0
      DO 153 I=1,20
      IF(COMPNT(I).GT.62) NN=NN+1
  153 CONTINUE
      IF(NN.EQ.0) GO TO 156
      DO 154 I=1,NN
      READ(5,155)J,(KNM(II),II=1,4)
      K=4*(J-1)
      DO 157 II=1,4
  157 KOMNAM(K+II)=KNM(II)
  154 CONTINUE
  155 FORMAT(I2,1X,4A4)
  156 CONTINUE
C     ****
      MAXSEX=NOCOMP + 3
      MAXSIN= 6
      MAXEQP=25
C
C  SET UP EQUIPMENT SUBROUTINE CALLING NUMBER FROM EQUIPMENT NAME
C
   51 DO 56 K2=1,50
      IF(KPM(1,K2).EQ.0) GO TO 56
      NEX=KPM(1,K2)
      NEMAX=K2
C
C  NEX IS EQUIPMENT NUMBER OF PROCESS MATRIX ROW 1- KPM(1,K2)
C  KPM(3,K2) IS THE EXTERNAL NAME GIVEN TO THIS UNIT.
C
      I=1
   53 IF(KPM(2,K2).EQ.NMLIST(I)) GO TO 55
      IF(I.GT.30) GO TO 54
      I=I+1
      GO TO 53
   54 WRITE(6,108)KPM(2,K2),NEX
  108 FORMAT(* OREAD1 ERROR - ILLEGAL NAME *,A4,* FOR EQUIP.*,I3,* -EXIT
     1*)
      CALL EXIT
   55 NECALL(NEX)=I
      NAME(NEX)=KPM(2,K2)
      NEXEQN(NEX)=KPM(3,K2)
   56 CONTINUE
C
C  IF THE NUMBER OF EQUIPMENT SUBROUTINE NAMES OR AN EQUIPMENT SUBROU-
C   TINE NAME ITSELF IS CHANGED, SUBROUTINE *EQCALL* MUST BE MODIFIED.
C
      IF(IPUNCH) 58,59,57
   57 READ(5,FLLIST)
      IF(IPUNCH.EQ. 1) GO TO 59
   58 LSAVE=IPUNCH
      IPUNCH=2
      WRITE(  7,PMLIST)
      IPUNCH=LSAVE
      IF(IPUNCH.EQ.2) WRITE( 7,FLLIST)
   59 CONTINUE
      DO 62 J=1,NEMAX
      IF(KPM(1,J).EQ.0) GO TO 62
      DO 61 I=2,8
   61 KPM(I,J)=KPM(I+2,J)
   62 CONTINUE
   63 CALL ZEROX(ENAME,50)
C
C..READ *EQPAR*(BY USING EQP1-EQP50) AND *ENAME*.
```

```
      READ(  5,EQLIST)
      DO 65 I=1,NEMAX
      J=ENAME(I)
      IF(J.EQ.0) GO TO 65
      IF(ABS(EQPAR(1,J)).LT.1.E-20) EQPAR(1,J)=J
C     WRITE(  6,66) J,(EQPAR(K,J),K=1,25)
   65 CONTINUE
C  66 FORMAT(* EQPAR(1,*,I2,*)=*,10G10.2/(13X,10G10.2))
      CALL ZEROX(KEFLAG,50)
C
C..READ *SEXTSV*(SEX1 -SEX100) AND *SNAME*
C
      CALL ZEROX(SNAME,100)
C
      READ(  5,SEXLST)
      DO 68 I=1,100
      J=SNAME(I)
      IF(J.EQ.0) GO TO 68
      IF(ABS(SEXTSV(1,J)).LT.1.E-20) SEXTSV(1,J)=J
C     WRITE(  6,69) J,(SEXTSV(K,J),K=1,23)
   68 CONTINUE
C  69 FORMAT(* SEXTSV(1,*,I2,*)=*,2F3.0,11G10.3/30X,10G10.3 )
      DO 70 I=1,100
      IF(ABS(SEXTSV(1,I)) .GT. 1.E-20) NSMAX=I
   70 CONTINUE
C
C..READ *SINTSV*( SIN1-SIN100) AND *SNAME*
C
      CALL ZEROX(SNAME,100)
      READ(  5,SINLST)
      DO 72 I=1,NSMAX
      J=SNAME(I)
      IF(J.EQ.0) GO TO 72
      IF(ABS(SINTSV(1,J)).LT.1.E-20) SINTSV(1,J)=J
C     WRITE(  6,73) J,(SINTSV(K,J),K=1,10)
   72 CONTINUE
   73 FORMAT(10H SINTSV(1,,I2,2H)=,F3.0,F4.0,8G10.3)
      CALL ZEROX(KSFLAG,100)
C
C..READ EQUIPMENT LISTS FOR RECYCLE CALCULATIONS, KE2, KE3 AND KE4
C.. THEIR CONTROL CONSTANTS NE2MAX, NE3MAX AND NE4MAX, WILL BE CALC.
   75 READ(  5,KELIST)
C     WRITE(  6,KELIST)
      NE2MAX=0
      NE3MAX=0
      NE4MAX=0
      DO 76 I=1,50
      IF(KE2(I).EQ.0) GO TO 77
   76 NE2MAX=NE2MAX+1
   77 DO 78 I=1,50
      IF(KE3(I).EQ.0) GO TO 79
   78 NE3MAX=NE3MAX+1
   79 DO 80 I=1,10
      IF(KE4(I).EQ.0) GO TO 81
   80 NE4MAX=NE4MAX+1
C
   81 IF(ITYPE.NE. 1) RETURN
C
      N3MAX=8
      DO 83 K=2,8
      I=10-K
      DO 82 J=1,NEMAX
      IF(KPM(I,J).NE.0) GO TO 85
   82 CONTINUE
   83 N3MAX=N3MAX-1
   85 CONTINUE
C..TEST EXISTENCE OF *NON-STANDARD* COMPONENT
      EXFLAG=.FALSE.
      DO 87 I=1,NOCOMP
      IF(COMPNT(I).GT. 62) GO TO 88
   87 CONTINUE
      GO TO 89
   88 READ(5,NSCOMP)
      EXFLAG=.TRUE.
   89 CONTINUE
C
C  BUILD UP THE STREAM CONNECTION MATRIX KSEM.
      DO 187 M1=1,NEMAX
      IF(KPM(1,M1).LE.0) GO TO 187
  188 M2=KPM(1,M1)
      DO 189 M3=2,N3MAX
      IF(KPM(M3,M1)) 191,187,190
  190 M4=(KPM(M3,M1))
      KSEM(3,M4)=M2
      GO TO 192
  191 M4=-(KPM(M3,M1))
      KSEM(2,M4)=M2
  192 KSEM(1,M4)=M4
  189 CONTINUE
  187 CONTINUE
      RETURN
  200 STOP
C  ........PUNCH OUT DATA BLOCK TO RESUBMIT THE CASE STUDY.............
      ENTRY DPUNCH
      IF(IPUNCH.EQ.0.OR.IPUNCH.EQ.1) RETURN
      WRITE(7,EQLIST)
      WRITE(7,SEXLST)
      WRITE(7,SINLST)
      WRITE(7,KELIST)
      IF(EXFLAG) WRITE(7,NSCOMP)
      RETURN
      END
```

```
      SUBROUTINE  DVDR
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLF(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLF(8),SOCOMP(20,8),SOKV(20,8)
      DIMENSION  SIIDUM(8,10),SOIDUM(8,10)
      EQUIVALENCE (SINUM,SIIDUM),(SONUM,SOIDUM)
      IF(KEFLAG(NE).EQ.1) GO TO 6
      SUM=0.
      DO 2 I=1,NOUT
    2 SUM=SUM+ EQPAR(I+1,NE)
      DO 3 I=1,NOUT
    3 EQPAR(I+1,NE)=EQPAR(I+1,NE)/SUM
    6 DO 5 J=1,NOUT
      DO 4 I=3,10
    4 SOIDUM(J,I)=SIIDUM(1,I)
      SOENTH(J)  =SIENTH(1) *EQPAR(J+1,NE)
      SOMOLE(J)  =SIMOLE(1) *EQPAR(J+1,NE)
      DO 5 I=1,NOCOMP
      SOCOMP(I,J)=SICOMP(I,1)*EQPAR(J+1,NE)
    5 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE EQCALL
C
      DATA CHESS,REC/5HCHESS,6HRECALL/
C
C
C     MODIFIED TO ALLOW OVERLAYING OF FOLLOWING SUBROUTINES ON CDC6400
C     MSEQ,AD15-AD19
C
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPB/NECALL(50),NEXEQN(50),NAME(50)
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
C
C     ******
C
      NAMELIST/DATAIN/SINUM,SIVPFR,SITEMP,SIPRES,SIENTH,SIMOLE,SICOMP
      IF(KTRACE.GT. 0 ) WRITE(6,60) NE,NAME(NE)
      IF(KTRACE.GE.2) WRITE(6,DATAIN)
   60 FORMAT(22HONOW CALLING EQUIPMENT,I4,3H - ,A4)
      NEQUIP=NECALL(NE)
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,
     1 23,24 25,26,27,28,29,30),NEQUIP
    1 CALL DVDR
      RETURN
    2 CALL DIST
      RETURN
    3 CALL MIXR
      RETURN
    4 CALL ADBF
      RETURN
    5 CALL REAC
      RETURN
    6 CALL VALV
      RETURN
    7 CALL HXER
      RETURN
    8 CALL PUMP
      RETURN
    9 CALL ABSR
      RETURN
   10 CALL OVERLAY(CHESS,3,1,REC)
   11 CALL FHTR
      RETURN
   12 CALL ADD1
      RETURN
   13 CALL ADD2
      RETURN
   14 CALL ADD3
      RETURN
   15 CALL ADD4
      RETURN
   16 CALL ADD5
      RETURN
   17 CALL ADD6
      RETURN
   18 CALL ADD7
      RETURN
   19 CALL ADD8
      RETURN
   20 CALL ADD9
      RETURN
   21 CALL AD10
      RETURN
   22 CALL AD11
      RETURN
   23 CALL AD12
      RETURN
   24 CALL AD13
      RETURN
   25 CALL AD14
      RETURN
   26 CALL OVERLAY(CHESS,3,2,REC)
   27 CALL OVERLAY(CHESS,3,3,REC)
   28 CALL OVERLAY(CHESS,3,4,REC)
   29 CALL OVERLAY(CHESS,3,5,REC)
   30 CALL OVERLAY(CHESS,3,6,REC)
      END
```

```
      SUBROUTINE EQPRNT
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSA/KPM(10,50),KSFM(3,100),N3MAX
      COMMON/SYSAA/TITLE(20),COMPNT(20),KOMNAM(80)
      COMMON/SYSB/KE1(50),NE1MAX,KE2(50),NE2MAX,KE3(50),NE3MAX,
     1KE4(10),NE4MAX,KRET,KRET2,KRET3
      COMMON/SYSC/LIMIT,LIMIT2,LIMIT3,LOOP,LOOPS
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/EQPB/NECALL(50),NEXEQN(50),NAME(50)
      COMMON/STMA/SEXTSV(23,100),SINTSV(10,100),NSMAX,MAXSEX,MAXSIN
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
C
C     ******
C
      INTEGER COMPNT,PFLAG(100) ,TITLE
C
      DIMENSION NEXT(5),ARV(5),NTG(55),NGX(50),NET(50)
      DATA NTG/4HDIVI,4HDERS,4H    ,4H    ,4H    ,4HFRAC,4HTION,4HATOR,4
     1HS   ,4H    ,4HMIXE,4HRS  ,4H    ,4H    ,4H    ,4HADIA,4HBATI,4HC
     1FI ,4HSH U,4HNITS,4HREAC,4HTORS,4H    ,4H    ,4H    ,4HP.C.,4H VAL,
     14HVES ,4H    ,4H    ,4HEXCH,4HANGE,4HR/CO,4HNDEN,4HSERS,4HPUMP,4HS
     1/CO,4HMPRE,4HSSOR,4HS   ,4HABSO,4HRBER,4HS   ,4H    ,4H    ,4HMULT
     1,4HISTA,4HGE U,4HNITS,4H    ,4HFIRE,4HD HE,4HATER,4HS   ,4H    /
   25 FORMAT (//)
      NOCOP3=NOCOMP+3
      J=0
  161 WRITE (6,168)
  168 FORMAT(1H ,23X,*EQUIPMENT SUMMARY - EQUIPMENT LIST*/*0*,19X,
     1*EQ. =*,9X,*EXT. NAME*,8X,*SUB. NAME*)
      DO 88 I=1,NEMAX
      IF(EQPAR(1,I).EQ.0.) GO TO 88
      J=J+1
       PFLAG(J)=I
   88 CONTINUE
      DO 95 I=1,J
      NKL1=PFLAG(I)
      NGX(I)=NEXEQN(NKL1)
   95 NET(I)=NAME(NKL1)
      WRITE (6,90) (PFLAG(K),NGX(K),NET(K),K=1,J)
   90 FORMAT (*0*,20X,I2,13X,A4,13X,A4)
      WRITE (6,91) TITLE
   91 FORMAT(*1*,20A4,//1H ,21X,*EQUIPMENT SUMMARY - INDIVIDUAL DETAILS*
     1 //)
      NOX1=-4
      NOX2=0
      DO 160 IE=1,30
      NOX1=NOX1+5
      NOX2=NOX2+5
      NOSOAP=1
      IF(IE.GT.11) GO TO 100
      DO 170 KE=1,NEMAX
      IF(NECALL(KE).EQ.IE) GO TO 171
  170 CONTINUE
      GO TO 160
```

```
  171 WRITE (6,29) (NTG(J),J=NOX1,NOX2)
   29 FORMAT(1H0,3X,5A4)
      GO TO 102
  100 DO 163 KE=1,NEMAX
      IF(EQPAR(1,KE).GT.0.1.AND.NECALL(KE).EQ.IE) GO TO 164
  163 CONTINUE
      GO TO 160
  164 WRITE (6,28)
      WRITE (6,101) NAME(KE)
  101 FORMAT(1H0,3X,A4)
  102 DO 159 JE=1,NEMAX
      IF(EQPAR(1,JE).EQ.0.) GO TO 159
      IF(NECALL(JE).EQ.IE) GO TO 162
      IF(JE.NE.NEMAX) GO TO 159
      GO TO 30
  162 PFLAG(NOSOAP)=JE
      NEXT(NOSOAP)=NEXEQN(JE)
      IF(NOSOAP.EQ.5) GO TO 167
      NOSOAP=NOSOAP+1
      IF(JE.EQ.NEMAX) GO TO 166
      GO TO 159
   30 IF(NOSOAP.LE.1) GO TO 159
  166 NOSOAP=NOSOAP-1
  167 WRITE (6,43) (PFLAG(K),K=1,NOSOAP)
   43 FORMAT(*0EQUIPMENT NO.    *,5(I9,5X))
      WRITE (6,44) (NEXT(K),K=1,NOSOAP)
   44 FORMAT (*0EXTERNAL NAME   *,5(7X,A4,3X))
      IF(IE.GT.11) GO TO 42
      GO TO (31,32,33,34,35,36,37,38,39,40,41),IE
   31 DO 46 JN=2,7
      NGU=JN-1
      CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,JN)
   45 IF(JN.NE.2) GO TO 50
      WRITE (6,49) NGU,(ARV(K),K=1,NOSOAP)
   49 FORMAT (* *,*FRXN. =*,2X,I2,2X,5F14.4)
      GO TO 46
   50 WRITE (6,51) NGU,(ARV(K),K=1,NOSOAP)
   51 FORMAT (* *,9X,I2,2X,5F14.4)
   46 CONTINUE
      WRITE (6,25)
      GO TO 158
   32 DO 52 JN=4,NOCOP3
      CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,JN)
   53 NGU=JN-3
      IF(JN.NE.4) GO TO 55
      WRITE (6,54) NGU,(ARV(K),K=1,NOSOAP)
   54 FORMAT (* *,*OVHD. COMP*,I2,2X,5F14.4)
      GO TO 52
   55 WRITE (6,56) NGU,(ARV(K),K=1,NOSOAP)
   56 FORMAT (* *,10X,I2,2X,5F14.4)
   52 CONTINUE
      WRITE (6,25)
      GO TO 158
   33 WRITE (6,25)
      GO TO 158
   34 CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,2)
   57 WRITE (6,58) (ARV(K), K=1,NOSOAP)
   58 FORMAT (* *,*CONDITION.  *,5F14.3)
```

```
 59 FORMAT(* 0. PHASE DET.*/* 1. CONST. T.*/* 2. ADIABATIC*//)
    WRITE (6,59)
    GO TO 158
 35 CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,2)
 62 WRITE (6,63) (ARV(K),K=1,NOSOAP)
 63 FORMAT (* *,*KEY COMP CONV*,1X,5F14.4)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,3)
 64 WRITE (6,65) (ARV(K),K=1,NOSOAP)
 65 FORMAT (* *,*KEY COMP =*,4X,5F14.4)
    DO 66 JN=4,NOCOP3
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,JN)
 67 NGU=JN-3
    IF(JN.NE.4) GO TO 68
    WRITE (6,69) NGU,(ARV(K),K=1,NOSOAP)
 69 FORMAT (* *,*STOICH. FAC.*,I2,5F14.4)
    GO TO 66
 68 WRITE (6,70) NGU,(ARV(K),K=1,NOSOAP)
 70 FORMAT (* *,12X,I2,5F14.4)
 66 CONTINUE
    WRITE (6,25)
    GO TO 158
 36 CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 2)
 71 WRITE (6,72) (ARV(K),K=1,NOSOAP)
 72 FORMAT (* *,*DOWNSTM. P*,4X,5F14.4)
    WRITE (6,25)
    GO TO 158
 37 CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,2)
 73 WRITE (6,74) (ARV(K),K=1,NOSOAP)
 74 FORMAT (* *,*U*,13X,5F14.4)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,3)
 75 WRITE (6,76) (ARV(K),K=1,NOSOAP)
 76 FORMAT (* *,*AREA*,10X,5F14.4)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,4)
 77 WRITE (6,78) (ARV(K),K=1,NOSOAP)
 78 FORMAT (* *,*= SHELLS    *,3X,5F14.4)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,5)
 79 WRITE (6,80) (ARV(K),K=1,NOSOAP)
 80 FORMAT (* *,*SHELL PASSES*,2X,5F14.4)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,6)
 81 WRITE(6,82) (ARV(K),K=1,NOSOAP)
 82 FORMAT (* *,*TUBE PASSES*,3X,5F14.4)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,7)
 83 WRITE(6,84) (ARV(K),K=1,NOSOAP)
 84 FORMAT (* *,*TYPE . *,7X,5F14.4)
 85 FORMAT(* 0. SIMPLE EXCH.*/* 1. WATER COOLED EXCH.*/* 2. WATER C
   -OOLED COND.*)
    WRITE(6,85)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,8)
 96 WRITE (6,97) (ARV(K),K=1,NOSOAP)
 97 FORMAT (* *,*WATER USAGE*,3X,5F14.4)
 98 FORMAT(*    (GAL/HR)*)
    WRITE (6,98)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,9)
 99 WRITE (6,103) (ARV(K),K=1,NOSOAP)
103 FORMAT (* *,*DELTA P-STR 1*,1X,5F14.4)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,10)
104 WRITE (6,105) (ARV(K),K=1,NOSOAP)
105 FORMAT (* *,*DELTA P-STR 2*,1X,5F14.4)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,11)
    WRITE(6,205)  (ARV(K),K=1,NOSOAP)
205 FORMAT(* Q-STREAM 1*,4X,5F14.4)
    WRITE(6,206)
206 FORMAT(*    (MM BTU/HR)*)
    WRITE (6,25)
    GO TO 158
 38 CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 2)
106 WRITE (6,107) (ARV(K),K=1,NOSOAP)
107 FORMAT (* *,*COMP. STAGES*,2X,5F14.4)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 3)
108 WRITE (6,109) (ARV(K),K=1,NOSOAP)
109 FORMAT (* *,*WORK CAPACITY*,1X,5F14.4)
    WRITE (6,110)
110 FORMAT(*    (BTU/HR)*)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 4)
111 WRITE (6,112) (ARV(K),K=1,NOSOAP)
112 FORMAT (* *,*OUTLET PRES.*,2X,5F14.4)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 5)
113 WRITE (6,114) (ARV(K),K=1,NOSOAP)
114 FORMAT (* *,*POWER TYPE .*,2X,5F14.4)
115 FORMAT(*   (+)-STEAM*/*   (0)-ELEC.*/*   (-)-FUEL GAS*)
    WRITE (6,115)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 6)
118 WRITE (6,119) (ARV(K),K=1,NOSOAP)
119 FORMAT (* *,*H-OUTLET STEAM*,5F14.4)
120 FORMAT(*    (BTU/LB)*)
    WRITE (6,120)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 7)
121 WRITE (6,122) (ARV(K),K=1,NOSOAP)
122 FORMAT (* *,*FUEL USAGE*,4X,5F14.4)
123 FORMAT(*    (MSCF/HR)*)
    WRITE (6,123)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 8)
124 WRITE (6, 97) (ARV(K),K=1,NOSOAP)
    WRITE(6,98)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 9)
127 WRITE (6,128) (ARV(K),K=1,NOSOAP)
128 FORMAT (* *,*STEAM USAGE*,3X,5F14.4)
129 FORMAT(*    (M LBS/HR)*)
    WRITE (6,129)
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,10)
130 WRITE (6,131) (ARV(K),K=1,NOSOAP)
131 FORMAT (* *,*KW USAGE*,6X,5F14.4)
    WRITE (6,25)
    GO TO 158
 39 CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 2)
132 WRITE (6,135) (ARV(K),K=1,NOSOAP)
    WRITE (6,25)
    GO TO 158
 40 CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 2)
134 WRITE (6,135) (ARV(K),K=1,NOSOAP)
135 FORMAT (* *,*= OF STAGES*,3X,5F14.4)
    DO 137 JN=3,7
    CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,JN)
136 WRITE (6,138) (ARV(K),K=1,NOSOAP)
138 FORMAT (* *,*INPUT STAGE =*,1X,5F14.4)
137 CONTINUE
```

```
      WRITE (6,25)
      GO TO 158
   41 CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 2)
  139 WRITE (6,140) (ARV(K),K=1,NOSOAP)
  140 FORMAT (* *,*HEAT DUTY*,5X,5F14.4)
  141 FORMAT(*   (MM BTU/HR)*)
      WRITE (6,141)
      CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 3)
  142 WRITE (6,143) (ARV(K),K=1,NOSOAP)
  143 FORMAT (* *,*DELTA PRES.*,3X,5F14.4)
  144 FORMAT(*   (PSIA)*)
      WRITE (6,144)
      CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 4)
  145 WRITE (6,146) (ARV(K),K=1,NOSOAP)
  146 FORMAT(* *,*TEMP. OUT*,5X,5F14.4)
  165 FORMAT(*   (DEG R.)*)
      WRITE (6,165)
      CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 7)
  147 WRITE(6,148) (ARV(K),K=1,NOSOAP)
  148 FORMAT (* *,*Q ABSORBED*,4X,5F14.4)
      WRITE(6,141)
      CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 8)
  150 WRITE (6,151) (ARV(K),K=1,NOSOAP)
  151 FORMAT (* *,*FUEL USAGE*,4X,5F14.4)
  152 FORMAT(*   (SCF/HR)*//)
      WRITE (6,152)
      GO TO 158
   42 CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV, 2)
  153 WRITE (6,154) (ARV(K),K=1,NOSOAP)
  154 FORMAT (* *,*PARAMETERS .*,2X,5F14.4)
      DO 155 JN=3,MAXEQP
      CALL TRANSF(1,NOSOAP,PFLAG,50,EQPAR,25,50,ARV,JN)
  156 WRITE (6,157) (ARV(K),K=1,NOSOAP)
  157 FORMAT (* *,14X,5F14.4)
  155 CONTINUE
      WRITE (6,25)
  158 NOSOAP=1
  159 CONTINUE
      WRITE (6,28)
   28 FORMAT (1H0,17(*.....*)//)
  160 CONTINUE
      RETURN
      END


      SUBROUTINE EQUIP
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSA/KPM(10,50),KSEM(3,100),N3MAX
      COMMON/STMA/SEXTSV(23,100),SINTSV(10,100),NSMAX,MAXSEX,MAXSIN
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
C
C     ******
C
C  *2* ENTRY POINTS.  EQUIP  +  REQUIP
      DIMENSION SIIDUM(8,10) ,SOIDUM(8,10)
      DIMENSION ISTIN(408),ISTOT(408)
      EQUIVALENCE (SINUM,SIIDUM,ISTIN),(SONUM,SOIDUM,ISTOT)
      CALL ZEROX(ISTIN,408)
      CALL ZEROX(ISTOT,408)
      K =1
      J =1
      DO 20 L2=2,N3MAX
      L3 =KPM(L2,NE)
C  L3 IS THE STREAM NUMBER OF KPM-COLUMN(NE)
C
   21 IF(L3) 24,23,22
C
C  SET INPUT STREAMS IN COMMON/STRMIN/
   22 DO 25 I =1,NOCOMP
   25 SICOMP(I ,K )=SEXTSV(I +3,L3)
      SIMOLE(K )    =SEXTSV(3,L3)
      DO 27 I =1,MAXSIN
   27 SIIDUM(K ,I )=SINTSV(I ,L3)
   28 K =K +1
      GO TO 20
   24 NIN=K -1
C  NIN = NUMBER OF INPUT STREAMS
C
C  SET OUTPUT STREAMS IN COMMON/STMOUT/
   30 L3=-L3
      DO 32 I =1,NOCOMP
   32 SOCOMP(I ,J )=SEXTSV(I +3,L3)
      SOMOLE(J )    =SEXTSV(3,L3)
      DO 34 I =1,MAXSIN
   34 SOIDUM(J ,I )=SINTSV(I ,L3)
   35 J =J +1
   20 CONTINUE
   23 NOUT=J -1
C  NOUT = NUMBER OF OUTPUT STREAMS
      RETURN
C
C  *** REQUIP ***
C   SET CALC. OUTPUT STREAM VALUES IN SEXTSV + SINTSV
C


      ENTRY REQUIP
      DO 39 I =1,NOUT
      J =SOIDUM(I ,1) + .01
      DO 41  K =1,NOCOMP
   41 SEXTSV( K +3,J )=SOCOMP( K ,I )
      SEXTSV( 3,J )=SOMOLE(I )
      DO 43  K =1,MAXSIN
   43 SINTSV( K ,J )=SOIDUM(I , K )
      IF(SOMOLE(I ).LT.1.E-20) GO TO 39
      SEXTSV( 2,J )=SINTSV( 6,J )/SEXTSV( 3,J )
   39 CONTINUE
      DO 46 I=1,NIN
      K=SIIDUM(I,1) + .01
      DO 44 J=1,NOCOMP
   44 SEXTSV(J+3,K)=SICOMP(J,I)
      SEXTSV( 3,K)=SIMOLE(I)
      DO 45 J=1,MAXSIN
   45 SINTSV( J,K)=SIIDUM(I,J)
      IF(SIMOLE(I).LT.1.E-20) GO TO 46
      SEXTSV( 2,K)=SINTSV( 6,K)/SEXTSV( 3,K)
   46 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE FHTR
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSO/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
C
C     ******
C
      REAL HVALUE
      DATA HVALUE/900./
      DO 2 I=1,NOCOMP
    2 SOCOMP(I,1)=SICOMP(I,1)
      SOMOLE(1)=SIMOLE(1)
      SOPRES(1)=SIPRES(1)-EQPAR(3,NE)
      PSAVE=SIPRES(1)
      SIPRES(1)=SOPRES(1)
      VSAVE=SIVPFR(1)
      TSAVE1=SITEMP(1)
      TSAVE2=EQPAR(4,NE)
      HSAVE=SIENTH(1)
      Q=EQPAR(2,NE)*1.E6
      NOUT=2
      SITEMP(1)=TSAVE2
      ESAVE=EQPAR(2,NE)
      EQPAR(2,NE)=1.
      CALL ADBF
      TEMP=SOENTH(1)+SOENTH(2)-SIENTH(1)
      IF( TEMP.LT.Q) GO TO 4
      NOUT=1
      SIENTH(1)=HSAVE+Q
      EQPAR(2,NE)=0.
      CALL ADBF
      GO TO 6
    4 Q=TEMP
    6 SIPRES(1)=PSAVE
      SITEMP(1)=TSAVE1
      SIENTH(1)=HSAVE
      EQPAR(2,NE)=ESAVE
      SOVPFR(1)=SOMOLE(1)/(SOMOLE(1)+SOMOLE(2))
      DO 8 I=1,NOCOMP
    8 SOCOMP(I,1)=SICOMP(I,1)
      SOMOLE(1)=SIMOLE(1)
      SOENTH(1)=HSAVE+Q
      SIVPFR(1)=VSAVE
      EQPAR(8,NE)= Q /(HVALUE*.75)
      EQPAR(7,NE)= Q / 1.E6
      NOUT=1
      RETURN
      END
```

```
      SUBROUTINE INIT
C
C     ******  COMMON DECK  ******
C
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STMA/SEXTSV(23,100),SINTSV(10,100),NSMAX,MAXSEX,MAXSIN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
C
C     ******
C
      DIMENSION DUM(1)
      DIMENSION SIIDUM(8,10)
      EQUIVALENCE (SINUM,SIIDUM)
      DO 9 I=1,NSMAX
      IF(ABS(SEXTSV(1,I)).LT.1.E-5.OR.ABS(SEXTSV(3,I)).LT.1.E-20)GO TO 9
      IF(ABS(SINTSV(4,I)).LT.1.E-5) GO TO 5
      VF=SINTSV(3,I)
      IF(VF.LT.0.0001.OR.VF.GT.0.9999) GO TO 10
      IF(SINTSV(6,I).NE.0.) GO TO 9
      GO TO 11
   10 DO 1 J=1,NOCOMP
    1 SICOMP(J,1)=SEXTSV(J+3,I)
      SIMOLE(1) = SEXTSV(  3,I)
      DO 2 J=1,MAXSIN
    2 SIIDUM(1,J)=SINTSV(  J,I)
      CALL ENTH(1,SINTSV(6,I),DUM)
      SEXTSV(2,I)=SINTSV( 6 ,I)/SEXTSV( 3,I)
      GO TO 9
    5 WRITE(  6,7) I
    7 FORMAT(108H0*** SUBROUTINE **INIT** CANNOT COMPUTE INITIAL ENTHALP
     1Y OF FOLLOWING STREAM SINCE TEMPERATURE NOT SPECIFIED,5X,I3)
   11 WRITE(6,12)I
   12 FORMAT(5H0*** ,*INIT - VAPOR FRACTION FOR STREAM*,I3,* INDICATES A
     1 TWO-PHASE MIXTURE BUT NO ENTHALPY IS SUPPLIED - BYPASS ENTHALPY C
     2ALCULATION*)
    9 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE COMPID
C
C   PURE COMPONENT ID NUMBERS...
C
C       1. HYDROGEN        18. N-TETRADECANE       35. 1-HEXENE
C       2. METHANE         19. N-PENTADECANE       36. CYCLOPENTANE
C       3. ETHANE          20. N-HEXADECANE        37. METHYLCYCLOPENTANE
C       4. PROPANE         21. N-HEPTADECANE       38. CYCLOHEXANE
C       5. I-BUTANE        22. ETHYLENE            39. METHYLCYCLOHEXANE
C       6. N-BUTANE        23. PROPYLENE           40. BENZENE
C       7. I-PENTANE       24. 1-BUTENE            41. TOLUENE
C       8. N-PENTANE       25. CIS-2-BUTENE        42. O-XYLENE
C       9. NEO-PENTANE     26. TRANS-2-BUTENE      43. M-XYLENE
C      10. N-HEXANE        27. I-BUTENE            44. P-XYLENE
C      11. N-HEPTANE       28. 1,3-BUTADIENE       45. ETHYLBENZENE
C      12. N-OCTANE        29. 1-PENTENE           46. NITROGEN
C      13. N-NONANE        30. CIS-2-PENTENE       47. OXYGEN
C      14. N-DECANE        31. TRANS-2-PENTENE     48. CARBON MONOXIDE
C      15. N-UNDECANE      32. 2-METHYL-1-BUTENE 49. CARBON DIOXIDE
C      16. N-DODECANE      33. 3-METHYL-1-BUTENE 50. HYDROGEN SULFIDE
C      17. N-TRIDECANE     34. 2-METHYL-2-BUTENE 51. SULFUR DIOXIDE
C
C      52. 2-METHYL-C5     56. 1-HEPTENE           60. C2-CYCLO-C6
C      53. 3-METHYL-C5     57. PROPADIENE          61. ISOPRENE
C      54. 2,2-DI-C1-C4    58. 1,2-BUTADIENE       62. WATER
C      55. 2,3-DI-C1-C4    59. C2-CYCLO-C5
C
C       ******  COMMON DECK  ******
C
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/SYSAA/TITLE(20),COMPNT(20),KOMNAM(80)
      COMMON/PHD/APC(20),ATC(20),AVC(20),AMW(20),AOMEG(20),ADEL(20),
     1AVW(20),APH(20),BET(20),GAM(20),DTA(20),EXFLAG
      COMMON /KHSAV/BASEA(20),BASEB(20),ZCD(20),ALD(20)
C
C       ******
C
      LOGICAL EXFLAG
      INTEGER COUNT,COMPNT
C
C      STANDARD COMPONENT NAMES
C
      INTEGER SCNAME(248)
      DATA (SCNAME(I),I=1,156)/
     1            4H HYD,4HROGE,4HN   ,4H    ,4H MET,4HHANE,4H    ,4H
     1 ,4H ETH,4HANE ,4H    ,4H    ,4H PRO,4HPANE,4H    ,4H    ,4H I-B,4
     2HUTAN,4HE   ,4H    ,4H N-B,4HUTAN,4HE   ,4H    ,4H I-P,4HENTA,4HNE
     3 ,4H    ,4H N-P,4HENTA,4HNE  ,4H    ,4H NEO,4H-PEN,4HTANE,4H    ,
     44H N-H,4HEXAN,4HE   ,4H    ,4H N-H,4HEPTA,4HNE  ,4H    ,4H N-O,4HC
     5TAN,4HE   ,4H    ,4H N-N,4HONAN,4HE   ,4H    ,4H N-D,4HECAN,4HE
     6,4H    ,4H N-U,4HNDEC,4HANE ,4H    ,4H N-D,4HODEC,4HANE ,4H    ,4H
     7 N-T,4HRIDE,4HCANE,4H    ,4H N-T,4HETRA,4HDECA,4HNE  ,4H N-P,4HENT
     8A,4HDECA,4HNE  ,4H N-H,4HEXAD,4HECAN,4HE   ,4H N-H,4HEPTA,4HDECA,4
     8HNE  ,4H ETH,4HYLEN,4HE   ,4H    ,
     9  4H PRO,4HPYLE,4HNE  ,4H    ,4H 1-B,4HUTEN,4HE   ,4H    ,4H CIS,
     A4H-2-B,4HUTEN,4HE   ,4H TRA,4HNS-2,4H-BUT,4HENE ,4H I-B,4HUTEN,4HE
     B  ,4H    ,4H 1,3,4H-BUT,4HADIE,4HNE  ,4H 1-P,4HENTE,4HE   ,4H
     C,4H CIS,4H-2-P,4HENTE,4HNE  ,4H TR-,4H2-PE,4HNTEN,4HE   ,4H 2-C,4H
     D1-1-,4HBUTE,4HNE  ,4H 3-C,4H1-1-,4HBUTE,4HNE  ,4H 2-C,4H1-2-,4HBUT
     EE,4HNE  ,4H 1-H,4HEXEN,4HE   ,4H    ,4H CYC,4HLOPE,4HNTEN,4HE   ,4
     FH C1-,4HCYCL,4HO-C5,4H    ,4H CYC,4HLOHE,4HXANE,4H    ,4H C1-,4HCY
     GCL,4HO-C6,4H    /
      DATA (SCNAME(I),I=157,248)/
     G            4H BEN,4HZENE,4H    ,4H    ,4H TOL,4HUENE,4H    ,4H    ,
     H4H O-X,4HYLEN,4HE   ,4H    ,4H M-X,4HYLEN,4HE   ,4H    ,4H P-X,4HY
     ILEN,4HE   ,4H    ,4H ETH,4HYLBE,4HNZEN,4HE   ,4H NIT,4HROGE,4HN
     J,4H    ,4H OXY,4HGEN ,4H    ,4H    ,4H CO ,4H    ,4H    ,4H    ,4H
     K CO2,4H    ,4H    ,4H    ,4H H2S ,4H    ,4H    ,4H    ,4H SO2, 4H
     L    ,4H    ,4H 2-M,4HETHY,4HL-C5,4H    ,4H 3-M,4HETHY,4HL-C5
     M,4H    ,4H 2,2,4H-DI-,4HC1-C,4H4   ,4H 2,3,4H-DI-,4HC1-C,4H4   ,4H
     N 1-H,4HEPTE,4HNE  ,4H    ,4H PRO,4HPADI,4HENE ,4H    ,4H 1,2,4H-BU
     OT,4HADIE,4HNE  ,4H C2-,4HCYCL,4HO-C5,4H    ,4H C2-,4HCYCL,4HO-C6,4
     PH    ,4H ISO,4HPREN,4HE   ,4H    ,4H WAT,4HER  ,4H    ,4H    /
C
C      CHAO-SEADER MODIFIED ACENTRIC FACTORS - DIMENSIONLESS
      REAL OMEGA(62)
      DATA OMEGA/    2*0.,.1064,.1538,.1825,.1953,.2014,.2387,.195,.2927
     1,.3403,.3992,.4439,.4869,.5210,.5610,.6002,.6399,.6743,.7078,.7327
     2,.0949,.1451,.2085,.2575,.2230,.1975,.2028,.2198,.2060,.2090,.2000
     3,.1490,.2120,.2463,.2051,.2346,.2032,.2421,.2130,.2591,.2904,.3045
     4,.2969,.2936,.0206,.0299,-.0067,.1768,.0868,.2402,.2771,.2746,.231
     54    ,.2466,.3471,.1193,.0987,.2709,.3046,.213,.348/
C
C      CHAO-SEADER MODIFIED HILDEBRAND SOLUBILITY PARAMETER
C            (CAL./ML.)** 1/2
      REAL DEL(62)
      DATA DEL/     3.25,5.45,5.88,6.00,2*6.73,3*7.021,7.266,7.43,7.551,7
     1.649,7.721,7.79,7.84,7.89,7.92,7.96,7.99,8.03,5.8,6.2,4*6.76,6.94,
     26*7.055,7.4,8.107,7.849,8.196,7.826,9.158,8.915,8.987,8.818,8.769,
     38.787,2.58,4.,3.13,6.,5.634,6.,7.018,7.132,6.712,6.967,7.168,6.854
     4,7.95,7.739,7.743,7.277,7.39/
C
C      VOLUME AT 25 DEG.C. , ML./ G-MOLE
      REAL V25(62)
      DATA V25/      31.,52.,68.,84.,105.5,101.4,117.4,116.1,123.3,136.6,
     1147.5,163.5,179.6,196.,212.2,228.6,244.9,261.3,277.8,294.1,310.4,
     261.,79.,95.3,91.2,93.8,95.4,88.,110.4,107.8,109.,108.7,112.8,106.7
     3,125.8,94.7,113.1,108.7,128.3,89.4,106.8,121.2,123.5,124.,123.1,
     4 36.0,28.4,35.2,53.6,43.6,45.2,132.9,130.6,122.7,131.2,141.7,61.6,
     5 83.7,128.8,143.1,100.37,18.076/
C
C      CHAO-SEADER CHARACTERISTIC MOLAR VOLUMES - ML./ G-MOLE
      REAL VW(62)
      DATA VW/      .955,5.,7.88,10.35,13.37,13.,15.36,15.27,15.89,17.64,2
     10.05,22.49,24.94,27.42,29.9,32.39,34.88,37.39,39.89,42.41,44.92,6.
     288,9.69,12.17,11.71,12.,12.17,11.27,14.55,14.26,14.41,14.31,14.77,
     314.14,16.9,12.72,15.33,14.87,17.67,12.26,14.83,17.03,17.28,17.34,1
     47.23,2.534,2.871,2.584,6.365,5.081,6.516,17.727,17.473,16.297,
     517.519,19.223,7.721,10.936,17.713,19.916,13.297,2.552/
C
C      CRITICAL TEMPERATURES, DEG. K.
      REAL TC(62)
      DATA TC/      33.27,190.7,305.43,369.97,408.14,425.17,461.,469.78,43
     13.76,507.9,540.16,569.4,595.,619.,640.,659.,677.,695.,710.,725.,73
     25.,283.06,365.1,419.6,428.,428.,417.89,425.,474.,481.16,479.16,472
```

```
     3.16,461.16,477.16,503.99,511.76,532.77,553.46,572.16,562.61,594.,6
     432.2,619.2,618.2,619.7,126.2,154.8,81.7,194.7,211.4,263.2,498.06,
     5504.33,489.39,500.28,535.5,392.78,458.06,569.44,602.61,484.28,
     6647.33/
C
C      CRITICAL PRESSURES, ATM.
      REAL PC(62)
      DATA PC/     12.79,45.8,48.2,42.01,36.,37.47,32.9,33.31,31.57,29.92
     1,27.01,24.64,22.5,20.8,19.2,17.9,17.,16.,15.,14.,13.,50.5,45.4,39.
     27,41.,41.,39.45,42.7,39.9,35.3,35.1,35.,34.5,35.9,32.1,44.55,37.36
     3,38.2,34.32,48.6,40.,36.,35.,34.,37.,33.5,50.1,34.5,72.9,88.9,77.7
     4,29.94,30.83,30.65,30.99,28.05,45.92,40.12,33.53,30.88,38.,218.37/
C
C      CRITICAL VOLUMES,  CC./GMOLE
      REAL VC(62)
      DATA VC/     65.0,99.5,148.,200.,263.,255.,308.,311.,303.,368.,426.
     1,486.,543.,602.,660.,718.,780.,830.,890.,950.,1000.,124.,181.,240.
     2,236.,240.,235.,221.,295.,295.,295.,301.,291.,286.,350.,260.,319.,
     3308.,344.,260.,316.,369.,376.,378.,374.,90.1,74.4,93.1,94.,95.,122
     4.,367.,367.,359.,358.,405.,146.,221.,375.,419.,266.,56./
C
C      MOLECULAR WEIGHTS
      REAL MW(62)
      DATA MW/     2.016,16.042,30.068,44.094,2*58.12,3*72.146,86.172,
     1100.198,114.224,128.25,142.276,156.302,170.328,184.354,198.38,
     2212.406,226.432,240.458,28.052,42.078,4*56.104,54.088,6*70.13,
     384.156,70.13,2*84.156,98.182,78.108,92.134,4*106.16,28.016,32.,28.
     401,44.01,34.08,64.06,4*86.2,98.2,40.1,54.1,98.2,112.2,68.1,18.02/
C
C      DENSITIES AT 15 DEG. C.,   G./ML.
      REAL DENS(62)
      DATA DENS/     .07,.2,.376,.5076,.5633,.5847,.6246,.63089,.5967,.66
     1384,.68801,.70654,.72146,.7339,.7440,.7525,.7600,.7663,.7720,.7734
     2,.7780,.3490,.5226,.6014,.6271,.61,.6005,.6274,.64565,.6607,.6534,
     3.6558,.6326,.6776,.67779,.75018,.7534,.78314,.77371,.88417,.87146,
     4.88440,.86836,.86532,.87141,.808,1.140,.804,1.101,.790,1.434,.6579
     5,.669,.654,.6664,.7015,.657,.658,.771,.7922,.6861,1.0/
C
C      **** COEFFICIENTS OF ZERO PRESSURE HEAT CONTENT. *****
      REAL APHA(62)
      DATA APHA/     6.952,3.381,2.247,2.410,3.332,4.453,4.816,5.910,4.37
     17,7.477,9.055,10.626,12.198,13.770,15.342,16.914,18.486,20.064,21.
     263,23.202,24.774,.944,.753,-.24,-1.778,2.34,1.65,-1.29,1.788,-3.35
     31,1.49,.495,3.270,.130,2.063,-12.957,-12.114,-15.935,-15.07,-8.65,
     4-8.213,-3.789,-6.533,-5.334,-8.398,6.903,6.085,6.726,5.316,7.07,6.
     5157,1.361,2.621,.593,1.298,2.344,3.0159,2.8487,-12.282,-15.559,
     6.4687,7.70/
      REAL BETTA(62)
      DATA BETTA/     -.04576E-2,18.044E-3,38.201E-3,57.195E-3,75.214E-3,
     172.27E-3,91.585E-3,88.449E-3,94.61E-3,104.422E-3,120.352E-3,136.29
     28E-3,152.248E-3,168.198E-3,184.148E-3,200.098E-3,216.048E-3,231.99
     37E-3,247.948E-3,263.898E-3,279.848E-3,3.735E-2,5.691E-2,8.65E-2,8.
     4078E-2,7.22E-2,7.702E-2,8.35E-2,101.454E-3,109.623E-3,99.696E-3,10
     53.985E-3,99.735E-3,99.118E-3,123.004E-3,13.087E-2,15.380E-2,16.454
     6E-2,18.972E-2,11.578E-2,13.357E-2,14.291E-2,14.905E-2,14.220E-2,15
     7.935E-2,-.03753E-2,.3631E-2,.04001E-2,1.4285E-2,.3128E-2,1.384E-2,
     812.5712E-2,12.3504E-2,13.3E-2,12.6929E-2,14.4802E-2,4.503E-2,
     9 6.4329E-2,17.682E-2,21.3801E-2,9.487E-2,4.594E-4/
      REAL GAMA(62)
      DATA GAMA/     .09563E-5,-43.E-7,-110.49E-7,-175.33E-7,-237.34E-7,-
     1222.14E-7,-289.62E-7,-273.88E-7,-305.87E-7,-324.71E-7,-375.28E-7,-
     2425.93E-7,-476.62E-7,-527.31E-7,-578.E-7,-628.69E-7,-679.38E-7,-73
     30.02E-7,-780.76E-7,-831.45E-7,-882.14E-7,-1.993E-5,-2.91E-5,-5.11E
     4-5,-4.074E-5,-3.403E-5,-3.981E-5,-5.582E-5,-554.27E-7,-603.45E-7,-
     5582.63E-7,-574.04E-7,-551.51E-7,-504.37E-7,-674.01E-7,-7.447E-5,-8
     6.915E-5,-9.203E-5,-10.989E-5,-7.54E-5,-8.23E-5,-8.354E-5,-8.831E-5
     7,-7.984E-5,-10.003E-5,.1930E-5,-.1709E-5,.1283E-5,-.8362E-5,.1364E
     8-5,-.9103E-5,-4.8147E-5,-4.7104E-5,-5.2878E-5,-4.9133E-5,-7.9864E-
     95,-2.556E-5,-3.418E-5,-10.2304E-5,-12.3808E-5,-5.553E-5,2.521E-6/
      REAL DELTA(62)
      DATA DELTA/     -.2079E-9,20*0.,4.22E-9,5.88E-9,12.07E-9,7.89E-9,6.
     107E-9,8.02E-9,14.24E-9,11.855E-9,12.911E-9,10.948E-9,12.414E-9,12.
     2041E-9,9.9E-9,14.404E-9,16.41E-9,20.03E-9,19.27E-9,24.09E-9,18.54E
     3-9,19.20E-9,18.8E-9,20.05E-9,17.03E-9,23.95E-9,-.6861E-9,.3133E-9,
     4-.5307E-9,1.784E-9,-.7867E-9,2.057E-9,4*0.,17.18E-9,5.678E-9,7.043
     5E-9,22.835E-9,26.997E-9,1.2629E-8,-8.587E-10/
C
      NAMELIST/NSCOMP/APC,ATC,AVC,AMW,AOMEG,ADEL,AVW,APH,BET,GAM,DTA
C
      DO 10  I=1,NOCOMP
      J=COMPNT(I)
      IF(J .GT. 62) GO TO 8
      I1=4*(I-1)+1
      IF((KOMNAM(I1)+KOMNAM(I1+1)+KOMNAM(I1+2)+KOMNAM(I1+3)) .NE. 0)
     1 GO TO 7
      DO 6 COUNT=1,4
      IK=I1+COUNT-1
      IS=4*(J-1) + COUNT
    6 KOMNAM(IK) = SCNAME(IS)
    7 PP=14.696E0*PC(J)
      TT=1.8*TC(J)
      AOMEG(I)=OMEGA(J)
      ADEL(I)=DEL(J)
      AVW(I)=VW(J)
      APC(I)=PP
      ATC(I)=TT
      AVC(I)=(VC(J)*.45359)/28.32
      AMW(I)=MW(J)
      ALD(I)=DENS(J)
      APH(I)=APHA(J)
      BET(I)=BETTA(J)
      GAM(I)=GAMA(J)
      DTA(I)=DELTA(J)
      GO TO 9
    8 TT=ATC(I)
      AVC(I)=AVC(I)*.45359/28.32
      PP=APC(I)
    9 BASEB(I)=.0867*TT/PP
      ZCD(I)=PP*AVC(I)/(10.73E0*TT)
   10 BASEA(I)=SQRT(.4278*TT**2.5/PP)
      IF(EXFLAG) WRITE (6,NSCOMP)
      RETURN
      END
```

```
      SUBROUTINE KHZT(ARG,ANS,LIST)
C
C     THIS IS A COMPREHENSIVE THERMO. DATA SUBROUTINE WITH 7 ENTRY POINTS
C                   ?. ENTRY ZDENS (ARG,ANS)
C                   3. ENTRY ENTH  (ARG,ANS)
C                   4. ENTRY KVAL  (ARG,ANS,LIST)
C                   5. ENTRY TSUBH (ARG,ANS)
C                   6. ENTRY BUBTP (ARG,ANS,LIST)
C                   7. ENTRY DEWTP (ARG,ANS,LIST)
C
C
C
C     ******  COMMON DFCK  ******
C
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/SYSAA/TITLE(20),COMPNT(20),KOMNAM(80)
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
      COMMON/PHD/APC(20),ATC(20),AVC(20),AMW(20),AOMEG(20),ADEL(20),
     1AVW(20),APH(20),BET(20),GAM(20),DTA(20),EXFLAG
      COMMON /KHSAV/BASEA(20),BASEB(20),ZCD(20),ALD(20)
C
C     ******
C
      INTEGER ARG
      REAL LIST(20)
      INTEGER TITLE
      REAL LNPHI(20),LNACT(20),LNNU(20)
      REAL LHC,MPOLY,MADDY
      INTEGER COUNTI,COUNT,COUN,COMPNT,VPFRAC
      REAL KV(20),NEWX(20),X(20),AV25(20)
      LOGICAL FFLAG,FLAG,FXFLAG,AKFLAG
      EQUIVALENCE (TRE,TEMTUR)
C
C     INTERNAL FUNCTIONS      ***** DELHV + DELHL *****
C     AS STATEMENT FUNCTION   DELHVL
C
      DELHVL(H,Z)=(1.5*ASODR*ALOG(1.+H)+1.-Z)*TEMTUR*1.986
C
      DPOLY(A,B,F,H,Z)=A+(B+(F+H*Z)*Z)*Z
      DADDY(A,B,F,H,Z)=A+B*SRED1+F*SRED2+H*ARED+Z*SRED4
      MPOLY(A1,A2,A3,A4,Z)=A1+(A2+(A3+A4*Z)*Z)*Z
      MADDY(A1,A2,A3,A4,A5)=A1+A2*SRED1+A3*SRED2+A4*ARED+A5*SRED4
C
C     CHAO-SEADER COEFFICIENTS FOR LIQUID FUGACITY
C       AS MODIFIED BY GRAYSON AND STREED.
      REAL COEFFT(3,10)
      DATA COEFFT/      1.50709,1.36822,2.05135,2.74283,-1.54831,-2.1089
     19,-.0211,2*0.,.00011,.02889,-.19396,0.,-.01076,.02282,.008585,
     2.10486,.08852,0.,-.02529,0.,2*0.,-.00872,2*0.,-.00353,2*0.,.00203/
C
C     CONSTANTS FOR YEN AND WOODS CORRELATION
C
      REAL FRI(5,3)
      DATA FRI/      -.0817,.3274,-.5014,.3870,-.1342,.0933,-.3445,.4042
     1,-.2083,.05473,.089,-.4344,.7915,-.7654,.3367/
      REAL FRJ(5,3)
      DATA FRJ/      -.0230,-.0124,.1625,-.2135,.08643,.022,-.003363,
     1 -.0796,.08546,-.0217,.0674,-.06109,.06261,-.2378,.1665/
      REAL FRK(4,3)
      DATA FRK/      .05626,-.3518,.6194,-.3809,.01937,-.03055,.06310,0.0
     1,-.01393,-.003459,-.1611,0.0/
      REAL FRL(5,3)
      DATA FRL/      -21.0,55.174,-33.637,-28.109,26.277,-16.0,30.699,
     1 19.645,-81.305,47.031,-6.55,7.8027,15.344,-37.04,20.169/
C
C     ...PRESET VALUES...
      DATA LNPHI,LNACT,LNNU/20*0E0,20*0E0,20*0E0/
C
C                              ***** ZDENS *****
C
      ENTRY ZDENS
      ASSIGN 20 TO LOC
      GO TO 1000
   20 IF(COUNT.NE. 0) GO TO 21
      ANS=0.
      RETURN
   21 ASSIGN 22 TO LOC
      GO TO 3000
   22 IF (VPFRAC .NE. 1) GO TO 24
      ASSIGN 23 TO LOC
      GO TO 5001
   23 ANS=ZFAC
      RETURN
   24 IF (VPFRAC .NE. 0) GO TO 26
      ASSIGN 25 TO LOC
      GO TO 6000
   25 ANS=ZLIQ
      RETURN
   26 WRITE(  6,27) NE
   27 FORMAT(100H0*** ZFACTOR CANNOT BE CALCULATED BECAUSE VAPOR FRACTIO
     1N IS IMPROPERLY SPECIFIED. .5 WILL BE ASSUMED./4H NE=,I4)
      ANS=0.5
      RETURN
C
C                              ***** ENTH *****
C
      ENTRY ENTH
      ASSIGN 30 TO LOC
      GO TO 1000
   30 IF(COUNT .NE. 0) GO TO 32
   31 ANS=0.
      RETURN
   32 IF(TEMTUR.LT.1.) GO TO 31
      LOS=1
      GO TO 14000
   33 ANS=GETH
      RETURN
C
C                              ***** KVAL *****
C
```

```
      ENTRY KVAL
      ASSIGN 40 TO LOC
      GO TO 1000
   40 IF(COUNT .NE. 0) GO TO 41
      ANS=0.
      RETURN
   41 IF(VPFRAC .NE. 0) GO TO 43
      LOS=1
      GO TO 7000
   43 IF(VPFRAC .NE. 1) GO TO 45
      LOS=1
      GO TO 8000
   45 WRITE(6, 46)  NE
   46 FORMAT(98H0*** K-VALUES CANNOT BE CALCULATED. VAPOR FRACTION IMPRO
     1PERLY SPECIFIED.  LAST VALUES WILL BE USED./4H NE=,I4)
   47 DO 48 I=1,NOCOMP
   48 LIST(I)=KV(I)
   49 ANS=1.0
      RETURN
C
C                            ***** TSUBH *****
C
      ENTRY TSUBH
      ASSIGN 50 TO LOC
      GO TO 1000
   50 IF( COUNT.NE.0.AND. ABS(HCONT).GT.1.E-4 ) GO TO 51
      ANS=0.
      RETURN
   51 IF(TEMTUR.LT. 1.E-4 ) TEMTUR=800.
      TT=TEMTUR
      DO 56     COUNT=      1,32
      LOS=2
      GO TO 14000
   52 HTRY=GETH
      SUMKX=HCONT - HTRY
      IF(ABS(SUMKX/HCONT) .GT. 1.E-5 ) GO TO 55
      ANS=TEMTUR
      RETURN
   55 ASSIGN 56 TO LOC
      GO TO 2000
   56 CONTINUE
      WRITE(  6,57) NE
   57 FORMAT(98H0*** TEMPERATURE AT INDICATED ENTHALPY CANNOT BE FOUND.
     1ASSUMED TEMPERATURE OF STREAM WILL BE USED./4H NE=,I4)
      ANS=TT
      RETURN
C
C                            ***** BUBTP *****
C
      ENTRY BUBTP
      ASSIGN 60 TO LOC
      GO TO 1000
   60 IF(COUNT .NE. 0) GO TO 62
   61 ANS=0.
      RETURN
   62 IF(COUNT .EQ. 1) GO TO 72
      LOS=1
      GO TO 4000
   63 IF(PRSSUR .GT. PCRIT ) GO TO 61
      TT=TEMTUR
      ENPHI=0E0
      DO 630 I=1,20
  630 LNPHI(I)=0E0
      CALL ZERO ( NEWX,20)
      DO 164  COUNT=1,40
      ASSIGN 64 TO LOC
      GO TO 12000
   64 ASSIGN 65 TO LOC
      GO TO 11000
   65 DO 66    I=1,NOCOMP
      TEMP=NEWX(I)
      NEWX(I)=X(I)
   66 X(I)= TEMP
      COUN=0
   67 FFLAG=.FALSE.
      IF(COUNT .EQ. 1) GO TO 68
      LOS=2
      GO TO 8000
   68 ASSIGN 69 TO LOC
      GO TO 9000
   69 DO 160 I=1,NOCOMP
      TEMP=KV(I)*NEWX(I)
      IF(TEMP.LT.1.E-10) GO TO 160
      IF(ABS((TEMP-X(I))/TEMP).GT. 1.E-5) FFLAG=.TRUE.
  160 X(I)=TEMP
      COUN=COUN+1
      IF( COUN.GT.20) GO TO 165
      IF(FFLAG) GO TO 67
      SUMKX=0.
      DO 161 I=1,NOCOMP
      TEMP=X(I)
      X(I)=NEWX(I)
      NEWX(I)=TEMP
      IF(X(I).LT.1.E-10) GO TO 161
      SUMKX=SUMKX+(1.-KV(I))*X(I)
  161 CONTINUE
      IF(ABS(SUMKX)/TMOLE .GT. 1.E-5 ) GO TO 163
      DO 162 I=1,NOCOMP
  162 LIST(I)=KV(I)
      ANS=TEMTUR
      RETURN
  163 ASSIGN 164 TO LOC
      GO TO 2000
  164 CONTINUE
  165 WRITE(  6,166) NE
  166 FORMAT(84H0*** BUBBLE POINT TEMPERATURE CANNOT BE DETERMINED. ASSU
     1MED TEMPERATURE WILL BE USED./4H NE=,I4)
      ANS=TEMTUR
      RETURN
C
C                            ***** DEWTP *****
C
      ENTRY DEWTP
      ASSIGN 70 TO LOC
      GO TO 1000
   70 IF(COUNT.NE. 0) GO TO 72
```

```
   71 ANS=0.
      RETURN
   72 LOS=2
      GO TO 4000
   73 IF(PRSSUR.GT.PCRIT) GO TO 71
      T1=TEMTUR
      ENACT=0E0
      DO 730 I=1,20
  730 KNACT(I)=0E0
      CALL ZERO (NEWX, 20)
      DO 174  COUNT=1,40
      ASSIGN 74 TO LOC
      GO TO 3000
   74 ASSIGN 75 TO LOC
      GO TO 5001
   75 ASSIGN 76 TO LOC
      GO TO 10009
   76 DO 77 I=1,NOCOMP
      TEMP= NEWX(I)
      NEWX(I)= X(I)
   77 X(I)=TEMP
      COUN=0
   78 FFLAG=.FALSE.
      LOS=2
      GO TO 7000
   79 DO 170 I=1,NOCOMP
      TEMP=NEWX(I)/KV(I)
      IF(TEMP.LT.1.E-10) GO TO 170
      IF(ABS((TEMP-X(I))/TEMP).GT. 1.E-5 ) FFLAG=.TRUE.
  170 X(I)=TEMP
      COUN=COUN+1
      IF( COUN.GT.20) GO TO 175
      IF( FFLAG) GO TO 78
      SUMKX=0.
      DO 171 I=1,NOCOMP
      TEMP=X(I)
      X(I)=NEWX(I)
      NEWX(I)=TEMP
      IF(X(I).LT.1.E-10) GO TO 171
      SUMKX=(1./KV(I)-1.)*X(I)+SUMKX
  171 CONTINUE
      IF(ABS(SUMKX)/TMOLE .GT. 1.E-5 ) GO TO 173
      DO 172 I=1,NOCOMP
  172 LIST(I)= KV(I)
      ANS=TEMTUR
      RETURN
  173 ASSIGN 174 TO LOC
      GO TO 2000
  174 CONTINUE
  175 WRITE(  6,176) NE
  176 FORMAT(81H0*** DEW POINT TEMPERATURE CANNOT BE DETERMINED, ASSUMED
     1 TEMPERATURE WILL BE USED,/4H NE=,I4)
      ANS=TEMTUR
      RETURN
C
C     INTERNAL FUNCTION      ***** COMVEC *****
C
 1000 IF(ARG .GT. 0) GO TO 1003
      J=-ARG
      DO 1001 I=1,20
 1001 X(I)=SOCOMP(I,J)
      HCONT=SOENTH(J)
      TMOLE=SOMOLE(J)
      TEMTUR=SOTEMP(J)
      PRSSUR=SOPRES(J)
      VPFRAC=SOVPFR(J) + .0001
      GO TO 1006
 1003 J=ARG
      DO 1004 I=1,20
 1004 X(I)=SICOMP(I,J)
      HCONT=SIENTH(J)
      TMOLE=SIMOLE(J)
      TEMTUR=SITEMP(J)
      PRSSUR=SIPRES(J)
      VPFRAC=SIVPFR(J) + .0001
 1006 COUNT=0
      DO 1007 I=1,NOCOMP
      IF(X(I) .GT. 1.E-10) COUNT=COUNT+1
 1007 CONTINUE
      GO TO LOC,(20,30,40,50,60,70)
C
C     INTERNAL FUNCTION      ***** ITER *****
C
 2000 IF(COUNT.EQ.1) GO TO 2001
      IF(ABS((SUMKX-OLDSUM)/SUMKX).LT.1.E-4) GO TO 2001
      TEMP=-SUMKX*(TEMTUR-OLDTEM)/(SUMKX-OLDSUM)
      TEM=OLDSUM*SUMKX
      IF(ABS(TEMP).GT.TMAX) TEMP=ABS(TEMP)/TEMP*TMAX
      IF(TEM.LT.0E0) TMAX=TMAX/2.
      DTEM=DT*TEMP
      IF(DTEM*TEM.LE.0E0) GO TO 2001
      DT=TEMP
      GO TO 2002
 2001 DT=-20.
      IF(COUNT.EQ.1) TMAX=500.
      IF(SUMKX.GT.0.) DT=20.
 2002 OLDTEM=TEMTUR
      OLDSUM=SUMKX
      TEMTUR=TEMTUR+DT
 2003 IF(TEMTUR .GE.0.) GO TO 2004
      DT=DT/2.
      TEMTUR=TEMTUR + ABS(DT)
      GO TO 2003
 2004 CONTINUE
      TEMTUR=AMAX1(TEMTUR,300.)
      GO TO LOC,(56,164,174)
C
C     INTERNAL FUNCTION      ***** CALCAB *****
C
 3000 CONTINUE
      A=0E0
      B=0E0
      SUMX=0.
      ASQDB=1E0
      DO 3001 I=1,NOCOMP
      IF(X(I).LT.  1.E-10) GO TO 3001
```

```
      B=B+BASEB(I)*X(I)
      A=A+BASEA(I)*X(I)
      SUMX=SUMX+X(I)
 3001 CONTINUE
      IF(SUMX.LT.1.E-10) GO TO 3002
      A=A/SUMX/TEMTUR**1.25
      B=B/SUMX/TEMTUR
      ASQDB=A*A/B
 3000 CONTINUE
      GO TO LOC,(4001,8001,14001,22,74 )
C
C     INTERNAL FUNCTION       ***** PCRIT *****
C                                                    RETURN PCRIT
 4000 ASSIGN 4001 TO LOC
      GO TO 3000
 4001 TRASH=(4.94/ASQDB)**.6666667
      PCRIT=.0867/(TRASH*B)
      GO TO      (63,73),LOS
C                                                    ENTRY LZFAC 5000
C     INTERNAL FUNCTION       ***** ZFAC *****             ZFAC  5001
C                                                    RETURN ZFAC
C
C     ORIGINAL NEWTON-RAPHSON ITERATIVE SOLUTION FOR REDLICH-KWONG
C     HAS BEEN REPLACED FOR CDC6400 VERSION
C     WITH AN ANALYTICAL SOLUTION FOR THE CUBIC IN Z
C
C     THE VAPOR COMPRESSIBILITY  ZFCTOR IS ALWAYS THE FIRST ROOT-ZZ
C     THE SAME ENTRY POINT IS NOW USED FOR CALLS  5000 +5001
C
 5000 CONTINUE
 5001 BP=B*PRSSUR
      ZB1=-1.
      ZB2=BP*(ASQDB-1.0-BP)
      ZB3=-ASQDB*BP*BP
      ZB1OV3=ZB1/3.0
      ZALF=ZB2-ZB1*ZB1OV3
      ZBET=2.0*ZB1OV3**3-ZB2*ZB1OV3+ZB3
      ZBETOV2=ZBET/2.
      ZALFOV3=ZALF/3.
      ZCUAOV3=ZALFOV3**3
      ZSQBOV2=ZBETOV2**2
      ZDEL=ZSQBOV2+ZCUAOV3
C     FOR ZDEL +VE THERE IS ONLY ONE REAL ROOT
C     FOR ZDEL -VE THERE ARE THREE REAL ROOTS
      IF(ZDEL)5003,5003,5004
 5004 ZEPS=SQRT(ZDEL)
      ZRCU=-ZBETOV2+ZEPS
      ZSCU=-ZBETOV2-ZEPS
      ZSIR=1.0
      ZSIS=1.0
      IF(ZRCU)5007,5008,5008
 5007 ZSIR=-1.0
 5008 IF(ZSCU)5009,5010,5010
 5009 ZSIS=-1.0
 5010 ZZR=ZSIR*(ZSIR*ZRCU)**0.33333333
      ZZS=ZSIS*(ZSIS*ZSCU)**0.33333333
      ZZ=ZZR+ZZS-ZB1OV3
      GO TO 5100
```

```
 5003 ZQUOT=ZSQBOV2/ZCUAOV3
      ZROOT=SQRT(-ZQUOT)
      ZTERM=1.0-ZROOT**2
      IF(ZBET)5011,5012,5012
 5012 ZPEI=(1.570796+ATAN(ZROOT/SQRT(ZTERM)))/3.0
      GO TO 5013
 5011 ZPEI=ATAN(SQRT(ZTERM)/ZROOT)/3.0
 5013 ZFACT=2.0*SQRT(-ZALFOV3)
      ZZ=ZFACT*COS(ZPEI)-ZB1OV3
      IF(ZZ.LT.0.2) WRITE(6,5555)ZZ,NE,ARG
 5555 FORMAT(*0WARNING - VAPOR COMPRESSIBILITY LESS THAN 0.2 (*,F5.3,*)
     1 - EQUIPMENT*,I3,* STREAM*,I3)
 5100 ZFCTOR=ZZ
      ZFAC=ZFCTOR
      H=BP/ZFAC
C
      GO TO LOC,(23,8002,14002,75)
C
C
C     INTERNAL FUNCTION       ***** LIQDEN *****
C                                                    RETURN ZLIQ
C... YEN AND WOODS CORRELATION ...
 6000 CONTINUE
 6100 AAMW=0.0E0
      PST=0.0E0
      PSV=0.0E0
      ZCE=0.0E0
      DO 6101 I=1,NOCOMP
      AAMW=AAMW+X(I)*AMW(I)/TMOLE
      PST=PST+X(I)*ATC(I)/TMOLE
      PSV=PSV+X(I)*AVC(I)/TMOLE
 6101 ZCE=ZCE+X(I)*ZCO(I)/TMOLE
      PSP=(ZCE*10.73E0*PST)/PSV
      ACON=DPOLY(17.4425E0,-214.578E0,989.625E0,-1522.06E0,ZCE)
      IF(ZCE.GT..26E0) GO TO 6111
      BCON=DPOLY(-3.28257E0,13.6377E0,107.4844E0,-384.211E0,ZCE)
      GO TO 6120
 6111 BCON=DPOLY(60.2091E0,-402.063E0,501.E0,641.E0,ZCE)
 6120 DCON=.93E0-BCON
      TROD=TEMTUR/PST
C     IF(TROD.GE.1.E0) WRITE(6,6121)
C6121 FORMAT (*WARNING - ATTEMPTING TO CALCULATE LIQUID DENSITIES AT RED
C    1UCED TEMPERATURES ABOVE 1*)
      ARED=1.0E0-TROD
      IF(TROD.GE.1.E0) ARED=0.0E0
      SRED1=ARED**(1./3.)
      SRED2=SRED1*SRED1
      SRED4=SRED2*SRED2
      RHORS=DADDY(1E0,ACON,BCON,0E0,DCON)
      E27=DADDY(.714E0,-1.626E0,-.646E0,3.699E0,-2.198E0)
      IF(TROD.GE.1.0E0) GO TO 6300
      F27=-ALOG(TROD)
      F27=.268E0*(TROD**2.0967)/(1.0E0+.8E0*(F27**.441))
      G27=.05E0+4.221E0*((1.01E0-TROD)**.75)*EXP(-7.848E0*(1.01E0-TROD)
     1)
      GO TO 6301
 6300 F27=.268E0*(TROD**2.0967)
      G27=.05E0
```

```
 6301 H27=DADDY(-10.6E0,45.22E0,-103.79E0,114.44E0,-47.38E0)
      IF(ZCE.LT..25E0) GO TO 6102
      IF(ZCE.GT..30E0) GO TO 6104
      DELPZ=(ZCE-.25E0)/.012E0
      AFAC=3.1E0+DPOLY(-.21417E-1,-.133624E0,.0619168E0,-.010875E0,
     1    DELPZ)*DELPZ
      GO TO 6103
 6104 AFAC=1.8E0
      GO TO 6103
 6102 IF(ZCE.LT..23E0) GO TO 6105
      DELPZ=(ZCE-.23E0)/.005E0
      AFAC=3.15E0+DPOLY(-.283392E-2,.358331E-2,-.31658E-2,.416557E-3,
     1     DELPZ)*DELPZ
      GO TO 6103
 6105 AFAC=3.15E0
 6103 PPS=EXP(2.302585E0*AFAC*(1.E0-(1.E0/TROD)))
      DELP=(PRSSUR/PSP)-PRS
      TT=DELP
      IF(DELP.LT..2E0) TT=0.2E0
      DELDUM=E27+F27*ALOG(TT)+G27*EXP(H27*TT)
      IF(DELP.LT.0.2E0) DELDUM=DELDUM*(DELP/.2E0)
 6113 IF(ABS(ZCE-.27E0).GT.1.E-10) GO TO 6107
      RDZC=0E0
      GO TO 6112
 6107 J=3
      IF(ZCE.GT..27E0) J=1
      IF(ZCE.LT..27E0.AND.ZCE.GT..24E0) J=2
      RI1=MADDY(FRI(1,J),FRI(2,J),FRI(3,J),FRI(4,J),FRI(5,J))
      RJ1=MADDY(FRJ(1,J),FRJ(2,J),FRJ(3,J),FRJ(4,J),FRJ(5,J))
      RK1=MPOLY(FRK(1,J),FRK(2,J),FRK(3,J),FRK(4,J),TROD)
      RL1=MADDY(FRL(1,J),FRL(2,J),FRL(3,J),FRL(4,J),FRL(5,J))
 6110 TT=DELP
      IF(DELP.LT..20E0) TT=.20E0
      RDZC=RI1+RJ1*ALOG(TT)+RK1*EXP(RL1*TT)
      IF(DELP.LT..20E0) RDZC=RDZC*(DELP/0.2E0)
 6112 RHO=RHORS+DELDUM+RDZC
      ROCRIT=(PSP*AAMW)/(ZCE*10.73E0*PST)
      RO=ROCRIT*RHO
      ZLIQ=(PRSSUR*AAMW)/(10.73*TEMTUR*RO)
 6007 LHC=B*PRSSUR/ZLIQ
 6008 GO TO LOC,(25,14004)
C
C      INTERNAL FUNCTION        ***** LIQPRM *****
C
 7000 ASSIGN 7001 TO LOC
      GO TO 12000
 7001 ASSIGN 7002 TO LOC
      GO TO 11000
 7002 ASSIGN 7003 TO LOC
      GO TO  9000
 7003 GO TO     (47,79),LOS
C
C      INTERNAL FUNCTION        ***** VAPPRM *****
C
 8000 ASSIGN 8001 TO LOC
      GO TO 3000
 8001 ASSIGN 8002 TO LOC
      GO TO 5001
 8002 ASSIGN 8003 TO LOC
      GO TO 10000
 8003 ASSIGN 8004 TO LOC
      GO TO 9000
 8004 GO TO     (47,69) ,LOS
C
C      INTERNAL FUNCTION        ***** EQR *****
C
 9000 CONTINUE
      DO 9001  I=1,NOCOMP
      AKV=2.302585E0*LNNU(I)+LNACT(I)-LNPHI(I)
      DKV=AMAX1(DKV,-30.)
      DKV=AMIN1(DKV,30.)
 9001 KV(I)=EXP(DKV)
      GO TO LOC,(7003,8004,69)
C
C      INTERNAL FUNCTION        ***** GASFUG *****
C
10000 ZT=ZFCTOR -1.
      ENPHI=0E0
      IF(H.GT..999E0) GO TO 10002
      ENPHI=ALOG(ZFCTOR*(1.-H))
10002 ASCON=ASQDB*ALOG(1.+H)
      BT= B*TEMTUR
      AT= A*TEMTUR**1.25 /2.
      DO 10001 I=1,NOCOMP
      IF(AT.LT.1E-30) AT=BASEA(I)
      IF(BT.LT.1E-30) BT=BASEB(I)
      BTT=BASEB(I)/BT
10001 LNPHI(I)= ZT*BTT-ENPHI- ASCON*(BASEA(I)/AT-BTT)
      GO TO LOC,(76,8003)
C
C      INTERNAL FUNCTION        ***** LIQACT *****
C
11000 SUMDEL=0E0
      SUMV=0E0
      DO 11001 I=1,NOCOMP
      AV25(I)=AVW(I)*(5.7+3.0*TEMTUR/ATC(I))
      IF(X(I).LT.1.E-10) GO TO 11001
      TEM=X(I)*AV25(I)
      SUMDEL=TEM*ADEL(I) + SUMDEL
      SUMV = TEM + SUMV
11001 CONTINUE
      IF(SUMV.LT.1.E-30) GO TO 11003
      SUMDEL= SUMDEL/ SUMV
11003 DO 11002 I=1,NOCOMP
11002 LNACT(I)=AV25(I)*(ADEL(I)-SUMDEL)**2/TEMTUR/1.1033
      GO TO LOC,(65,7002)
C
C      INTERNAL FUNCTION        ***** LIQFUG *****
C
12000 CONTINUE
      DO 12001  I=1,NOCOMP
      TRED=TEMTUR/ATC(I)
      PRED=PRSSUR/APC(I)
      J= 3 - (2/COMPNT(I))
      IF(ABS(AOMEG(I)).LT..03.AND.J.EQ.3) J=2
      ENNU=((COEFFT(J,5)*TRED+COEFFT(J,4))*TRED+COEFFT(J,3))*TRED+
```

```
      1COEFFT(J,2)/TRED+COEFFT(J,1)+((COEFFT(J,8)*TRED+COEFFT(J,7))*TRED
      2+COEFFT(J,6))*PRED+(COEFFT(J,10)*TRED+COEFFT(J,9))*PRED*PRED-
      3ALOG(PRED)/2.302585
C THIS VARIATION SUGGESTED BY GRAYSON AND STREED.
      IF(TRED.GT.1E0) TRED=1E0
12001 LNNU(I)=ENNU+AOMEG(I)*((-3.15224*TRED*TRED+8.65808)*TRED-4.23893
     1 -1.2206/TRED-.025*(PRED-.6))
      GO TO LOC,(64,7001)
C
C     INTERNAL FUNCTION       ***** ZPH *****
C                                                     RETURN HBASE
13000 SAPH=0E0
      SBET=0E0
      SGAM=0E0
      SDTA=0E0
      DO 13001  I=1,NOCOMP
      IF(X(I).LT.1.E-10) GO TO 13001
      SAPH=SAPH+X(I)*APH(I)
       BET=SBET+X(I)*BET(I)
      SGAM=SGAM+X(I)*GAM(I)
      SDTA=SDTA+X(I)*DTA(I)
13001 CONTINUE
      TOK=TRE/1.8
      HBASE=(((SDTA/4.*TOK+SGAM/3.)*TOK+SBET/2.)*TOK+SAPH)*TRE
      GO TO LOC,(14006)
C
C     INTERNAL FUNCTION       ***** GETH *****
C                                                     RETURN GETH
14000 ASSIGN 14001 TO LOC
      GO TO 3000
14001 IF(VPFRAC .NE. 1) GO TO 14003
      ASSIGN 14002 TO LOC
      GO TO 5001
14002 HDEL=DELHVL(H,ZFCTOR) * TMOLE
      GO TO 14005
14003 IF(VPFRAC .NE. 0) GO TO 14007
      ASSIGN 14004 TO LOC
      GO TO 6000
14004 HDEL=DELHVL(LHC,ZLIQ )* TMOLE
14005 ASSIGN 14006 TO LOC
      GO TO 13000
14006 GETH=HBASE- HDEL
      GO TO 14009
14007 WRITE(6,14008)  NE
14008 FORMAT(99H0*** ENTHALPY CANNOT BE CALCULATED. VAPOR FRACTION IS IM
     1PROPERLY SPECIFIED. A VALUE OF ZERO ASSUMED,/4H NE=,I4)
      GETH= 0.
14009 GO TO       (33,52) ,LOS
      END
```

```
      SUBROUTINE  MSEQ
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSC/LIMIT,LIMIT2,LIMIT3,LOOP,LOOPS
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
      COMMON/HXMS/NHX(10),SAVENE(5),SIZE(5)
C
C     ******
C
      INTEGER BB,COUNT,FIRST,STEP,STGLST(8),SAVENE,SIZE
      REAL     SAVEIN(27,80),SAVEOT(27,2),INT(27,24)
      LOGICAL FLAG,FLAGA,SW,SAVESW(5)
      REAL     SIDUM(8,6) ,SODUM(8,6)
      EQUIVALENCE (SIDUM,SINUM),(SODUM,SONUM)
      CALL ZEROX(STGLST,8)
      CALL ZERO(INT, 648)
      N=EQPAR(2,NE) + 0.1
      IF(N.LE.25) GO TO 5
      WRITE(  6,3) NE
    3 FORMAT(111H0*** MULTISTAGE EQUILIBRIUM CALCULATION ALLOWS ONLY 25
     1STAGES. CALCULATIONS WILL BE BY-PASSED FOR EQUIPMENT NO.,I3)
      RETURN
    5 N=N-1
      BB=0
      NOCOP7=NOCOMP+7
      DO 9 I=1,5
      IF(SAVENE(I).EQ.0) GO TO 10
      IF(SAVENE(I).NE.NE) GO TO 9
      SW=SAVESW(I)
      DO 7 J=1,N
      DO 7 K=1,NOCOP7
    7 INT(K,J)=SAVEIN(K,J+BB)
      EQPAR(2,NE) =2.
      GO TO 14
    9 BB=BB+SIZE(I)
   10 SW=.TRUE.
      EQPAR(2,NE)=0.
      IF( N+BB .LE.80) GO TO 14
      WRITE(  6,12) NE
   12 FORMAT(69H0*** CAPACITY OF SAVEIN MATRIX EXCEEDED IN **MSEQ**. EQU
     1IPMENT NO. IS,I4)
      RETURN
   14 PMIN=1.E4
      DO 16 I=1,NIN
      IF(SIPRES(I).LT.PMIN.AND.SIMOLE(I).GT.0.) PMIN=SIPRES(I)
   16 STGLST(I)=EQPAR(I+2,NE) + 0.1
      FLAGA= .FALSE.
      DO 19 I=1,NIN
```

```
      IBB=I+BB
      DO 17 J=1,6
   17 SAVEIN(J,IBB)=SIDUM(I,J)
      SAVEIN(7,IBB)=SIMOLE(I)
      DO 18 J=1,NOCOMP
   18 SAVEIN(J+7,IBB)=SICOMP(J,I)
   19 CONTINUE
      SUM=0.
      DO 20 I=1,NIN
   20 SUM=SUM+SIMOLE(I)
      SINUM(1)=0.
      SIFLAG(1)=0.
      SIENTH(1)=0.
      SITEMP(1)=0.
      SIPRES(1)=0.
      SIMOLE(1)=0.
      CALL ZERO(SICOMP,20)
      DO 22 I=1,2
      IF(SOTEMP(I).LT.1.) SOTEMP(I)=560.
      DO 21 J=1,6
   21 SAVEOT(J,I)=SODUM(I,J)
      SAVEOT(7,I)=SOMOLE(I)
      DO 22 J=1,NOCOMP
   22 SAVEOT(J+7,I)=SOCOMP(J,I)
      LASTC=6
      IF(LOOP.LE.1.OR.LOOP.GE.LOOPS) LASTC=20
      DO 560 COUNT=1,LASTC
      IF( SW ) GO TO 24
      FIRST=1
      STEP=1
      II=1
      JJ=2
      GO TO 25
   24 FIRST=N
      STEP=-1
      II=2
      JJ=1
   25 DO 27 I=1,NIN
      J=STGLST(I)
      IF(J.EQ.0) GO TO 27
      IF(J.EQ.1.AND.SW.OR.J.EQ.N+1.AND..NOT.SW) GO TO 27
      IF(SW) J=J-1
      DO 26 K=2,NOCOP7
   26 INT(K,J)=INT(K,J)+ SAVEIN(K,I+BB)
   27 CONTINUE
   28 I=FIRST
   29 DO 30 J=2,6
   30 SIDUM(1,J)= INT(J,I)
      SIMOLE(1) = INT(7,I)
      DO 31 J=1,NOCOMP
   31 SICOMP(J,1)=INT(J+7,I)
      IF( SW ) GO TO 32
      K=I+1
      GO TO 33
   32 K=I-1
   33 SIPRES(1)=PMIN
      IF(I.EQ.1.AND..NOT.SW.OR.I.EQ.N.AND.SW) GO TO 34
      GO TO 48
   34 IF(SITEMP(1).LT.1..AND.SAVEOT(4,II).LT.1.) GO TO 35
      IF(SAVEOT(4,II).GT.1.) SITEMP(1)=SAVEOT(4,II)
      GO TO 36
   35 SITEMP(1)=560.
   36 CONTINUE
      CALL ADBF
      SAVEOT(3,1)=SOVPER(1)
      SAVEOT(3,2)=SOVPER(2)
      FLAG=.TRUE.
C
C** FORCE OVERALL MATERIAL BALANCE ON COLUMN TERM. STAGE
C
      IF(COUNT-COUNT/2*2.NE.0) GO TO 39
      TEMPO=SOMOLE(II)+SAVEOT(7,JJ)
      IF(TEMPO.LE.0.) GO TO 40
      TEMPO=(SUM-TEMPO)/SUM*SOMOLE(II)/TEMPO + 1.
      IF(TEMPO.GT.2.) TEMPO=2.
      IF(TEMPO.LT.1.) TEMPO=1.
      KK=II
   37 DO 38 J=1,NOCOMP
   38 SOCOMP(J,KK)=SOCOMP(J,KK)*TEMPO
      SOMOLE(KK)=SOMOLE(KK)*TEMPO
      SOENTH(KK)=SOENTH(KK)*TEMPO
      IF(KK.EQ.JJ) GO TO 39
      KK=JJ
      GO TO 37
   39 GO TO 41
   40 TEMPO=1.
C
   41 DO 46 J=4,NOCOP7
      IF(J-7) 42,43,44
   42 TEMP=SODUM(II,J)
      GO TO 45
   43 TEMP=SOMOLE(II)
      GO TO 45
   44 TEMP=SOCOMP(J-7,II)
   45 IF(ABS(TEMP-SAVEOT(J,II)).GT.DERROR*TEMP.AND.TEMP.GT.0.0001)
     1 FLAG=.FALSE.
   46 SAVEOT(J,II)=TEMP
      IF(EQPAR(10,NE).NE.0.) WRITE(6,47) II,(SAVEOT(J,II),J=1,NOCOP7)
   47 FORMAT(80H0*** THIS OUTPUT COMES FROM MULTISTAGE EQ. ROUTINE WHEN
     IEQPAR(10,NE) IS NOT ZERO,10H SAVEOT(J,,I1,5H) ARE,/(10X,7G10.3))
      GO TO 49
   48 CALL ADBF
      SAVEOT(3,1)=SOVPER(1)
      SAVEOT(3,2)=SOVPER(2)
   49 CONTINUE
      IF(I.NE.1.AND.SW.OR.I.NE.N.AND..NOT.SW) GO TO 50
      GO TO 53
   50 DO 51 J=2,6
   51 INT(J,K)=INT(J,K)+SODUM(JJ,J)
      INT(4,K)=SOTEMP(JJ)
      INT(7,K)=INT(7,K)+SOMOLE(JJ)
      DO 52 J=1,NOCOMP
   52 INT(J+7,K)=INT(J+7,K)+SOCOMP(J,JJ)
   53 CONTINUE
      DO 54 J=2,6
   54 INT(J,I)=SODUM(JJ,J)
```

```
      INT(7,I)=SOMOLE(JJ)
      DO 55 J=1,NOCOMP
   55 INT(J+7,I)=SOCOMP(J,JJ)
      I=I+STEP
      IF(I.LT.1 .OR. I.GT.N) GO TO 56
      GO TO 29
   56 IF(FLAG.AND.FLAGA) GO TO 58
      FLAGA=.FALSE.
      IF(FLAG) FLAGA=.TRUE.
      EQPAR(2,NE)=3.
  560 SW=.NOT.SW
      SW=.NOT.SW
      IF(COUNT.GE.20) WRITE(6,57) NE,LASTC
   57 FORMAT(58H0*** MULTI-STAGE EQUILIBRIUM CALCULATION FOR EQUIPMENT N
     1O.,I3,23H DID NOT CONVERGE AFTER,I3,12HTH ITERATION,/28H CURRENT V
     1ALUES WILL BE USED)
   58 DO 61 I=1,2
      DO 59 J=1,6
   59 SODUM(I,J)=SAVEOT(J,I)
      SOMOLE(I) =SAVEOT(7,I)
      DO 60 J=1,NOCOMP
   60 SOCOMP(J,I)=SAVEOT(J+7,I)
   61 CONTINUE
      DO 64 I=1,NIN
      DO 62 J=1,6
   62 SIDUM(I,J)=SAVEIN(J,I+BB)
      SIMOLE(I) =SAVEIN(7,I+BB)
      DO 63 J=1,NOCOMP
   63 SICOMP(J,I)=SAVEIN(J+7,I+BB)
   64 CONTINUE
      BB=0
      DO 67 I=1,6
      IF(I.GT.5.OR.SAVENE(I).EQ.0) GO TO 68
      IF(SAVENE(I).NE.NE) GO TO 67
   65 SAVESW(I)=.NOT.SW
      DO 66 J=1,N
      DO 66 K=1,NOCOP7
   66 SAVEIN(K,J+BB)=INT(K,J)
      GO TO 69
   67 BB=BB+SIZE(I)
   68 IF(I.GT.5) GO TO 69
      IF(BB+N.GT.80) GO TO 69
      SAVENE(I)=NE
      SIZE(I)=N
      GO TO 65
   69 EQPAR(2,NE)=N+1
      RETURN
      END
```

```
      SUBROUTINE PUMP
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
C
C     ******
C
      DIMENSION DUM(1)
      DO 1 I=1,NOCOMP
    1 OCOMP(I,1)=SICOMP(I,1)
      SOMOLE(1)   =SIMOLE(1)
      SOTEMP(1)   =SITEMP(1)
      SOVPFR(1)=SIVPFR(1)
      IF(SIVPFR(1)-0.999) 12,3,3
    3 N=EQPAR(2,NE)
      PR=(EQPAR(4,NE)/SIPRES(1))**(1./N)
      TEMP=PR**0.2126-1.0
      CALL ZDENS(1,DENST,DUM)
      W=DENST*1.986*SITEMP(1)*4.74615*TEMP*SIMOLE(1)
      W=W*N
      IF(W.LE.EQPAR(3,NE)) GO TO 5
      PR=(TEMP*EQPAR(3,NE)/W+1.)**4.74615
      SOPRES(1)=SIPRES(1)*PR**N
      W=EQPAR(3,NE)
      GO TO 6
    5 SOPRES(1)=EQPAR(4,NE)
    6 CALL ENTH(-1,SOENTH(1),DUM)
      EQPAR(8,NE)=(W-(SOENTH(1)-SIENTH(1)))/124.95
    7 IF(EQPAR(5,NE).GE.0.) GO TO 8
      EQPAR(7,NE)=W*.35036383E-2
      GO TO 10
    8 IF(EQPAR(5,NE).EQ.0..AND.EQPAR(6,NE).EQ.0.) GO TO 9
      EQPAR(9,NE)=W/(EQPAR(5,NE)-EQPAR(6,NE))/.85
      GO TO 10
    9 EQPAR(10,NE)=W*2.928E-4/.85
   10 CONTINUE
      RETURN
   12 TEMP=(EQPAR(4,NE)-SIPRES(1))/SIPRES(1)
      CALL ZDENS(1,DENST,DUM)
      W=DENST*TEMP*1.986*SITEMP(1)*SIMOLE(1)
      IF(W.GT.EQPAR(3,NE)) GO TO 14
      SOPRES(1)=EQPAR(4,NE)
      GO TO 15
   14 SOPRES(1)=TEMP/W*EQPAR(3,NE)*SIPRES(1) +SIPRES(1)
      W=EQPAR(3,NE)
   15 CALL ENTH(-1,SOENTH(1),DUM)
      GO TO 7
      END
```

```
      SUBROUTINE RCYCLE
C
C     ENTRY  RCYOV  REQUIRED FOR RETURN FROM LOWER LEVEL 6400 OVERLAY
C
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSA/KPM(10,50),KSFM(3,100),N3MAX
      COMMON/SYSB/KE1(50),NE1MAX,KE2(50),NE2MAX,KE3(50),NE3MAX,
     1KE4(10),NE4MAX,KRET,KRET2,KRET3
      COMMON/SYSC/LIMIT,LIMIT2,LIMIT3,LOOP,LOOPS
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/STMA/SEXTSV(23,100),SINTSV(10,100),NSMAX,MAXSEX,MAXSIN
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
      COMMON/OVR/NRET
C
C     ******
C
      INTEGER LKE2,LKE3
      NAMELIST/DATOUT/SONUM,SOVPFR,SOTEMP,SOPRES,SOENTH,SOMOLE,SOCOMP
      DATA LKE2,LKE3/4HKE2,,4HKE3,/
      IF(NE2MAX .EQ. 0) GO TO 4
      DO 2 I=1,NE2MAX
    2 KE1(I)=KE2(I)
      NE1MAX=NE2MAX
      NE2MAX=0
      WRITE( 6,3) LKE2
    3 FORMAT(68H1*** BEGIN TRIAL AND ERROR RECYCLE CALCULATIONS WITH EQU
     1IPMENT LIST ,A4)
      GO TO 10
    4 IF(NE3MAX .EQ. 0) GO TO 8
      DO 5 I=1,NE3MAX
    5 KE1(I)=KE3(I)
      NE1MAX=NE3MAX
      NE3MAX=0
      WRITE( 6,3) LKE3
      GO TO 10
    8 WRITE( 6,9)
    9 FORMAT(66H0*** RECYCLE COMPUTATION REQUIRED BUT EQUIP. LIST WAS NO
     1T SUPPLIED)
      KRET2=2
      RETURN
   10 LOOP=1
C
   35 LIMIT2=0
      WRITE(6,36)LOOP
C         *I* IS COUNTING INDEX FOR KE1
   36 FORMAT(11H0... BEGIN ,4HLOOP,I4,4H ...)
C
      I=1
  102 CONTINUE
   37 NE=KE1( I)
      CALL EQUIP
      NRET=1
      CALL FQCALL
      GO TO 101
      ENTRY RCYOV
  101 IF(KTRACE.EQ.3) WRITE(6,DATOUT)
      CALL TEST
      CALL REQUIP
C  IF LIMIT IS ZERO, CONVERGENCE OBTAINED FOR EQUIP, KE1( I)
C  NUMBER OF ITEMS EXCEEDING DERROR IS VALUE OF LIMIT
      IF(LIMIT) 86,86,68
   68 LIMIT2=LIMIT2+1
   86 I=I+1
      IF(I.LE.NE1MAX) GO TO 102
   87 IF(LIMIT2.GT.0) GO TO 113
C
C  CONVERGENCE OBTAINED IF LIMIT2 IS ZERO
C
      GO TO 13
C
C  CHECK NUMBER OF LOOPS THRU KE1
C
  113 IF(LOOPS-LOOP.LE.0) GO TO 116
      IF(NPFREQ.EQ. 0 ) GO TO 115
      IF(LOOP/NPFREQ*NPFREQ.EQ.LOOP) CALL PTPRNT
  115 LOOP=LOOP+1
C  GO THRU KE1 LIST AGAIN
C
      GO TO 35
C  MAXIMUM NUMBER OF LOOPS TRIED WITHOUT CONVERGING.
C
  116 WRITE( 6,117)
  117 FORMAT(77H0*** EQUIPMENT CALCULATION LOOP DID NOT CONVERGE, COMPUT
     1ATIONS WILL CONTINUE.)
C
   13 DO 14 I=1,NE1MAX
      K=KE1(I)
      KEFLAG(K)=1
      DO 12 J=2,N3MAX
      L=KPM(J,K)
      IF(L) 11,14,12
   11 L=-L
      IF(KSFLAG(L).EQ.0) KSFLAG(L)=1
   12 CONTINUE
   14 CONTINUE
      WRITE( 6,16)
   16 FORMAT(32H0*** END OF RECYCLE CALCULATIONS)
      KRET2=1
      RETURN
      END
```

```
      SUBROUTINE REAC
C
C     ******  COMMON DFCK  ******
C
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STRMIN/SINUM(8),SIFLAG(8),SIVPFR(8),SITEMP(8),
     1SIPRES(8),SIENTH(8),SIVISC(8),SITHK(8),SILZ(8),SIVZ(8),
     2SIMOLE(8),SICOMP(20,8),SIKV(20,8)
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
C
C     ******
C
      DIMENSION DUM(1)
      KY=EQPAR(3,NE)+0.1
      FK=EQPAR(2,NE)*SICOMP(KY,1)
      DO 1  I=1,NOCOMP
      IF((EQPAR(I+3,NE)*FK+SICOMP(I,1))) 2,1,1
    2 FK=SICOMP(I,1)/(-EQPAR(I+3,NE) )
    1 CONTINUE
      SUM=0.0
      DO 3 I=1,NOCOMP
      SOCOMP(I,1)=SICOMP(I,1)+EQPAR(I+3,NE)*FK
    3 SUM=SUM+SOCOMP(I,1)
      SOMOLE(1)=SUM
      SOTEMP(1)=SITEMP(1)
      SOPRES(1)=SIPRES(1)
      SOVPFR(1)=SIVPFR(1)
      CALL ENTH(-1,SOENTH(1),DUM)
      RETURN
      END
```

```
      SUBROUTINE SCAN
C
C     ******  COMMON DFCK  ******
C
      COMMON/SYSA/KPM(10,50),KSFM(3,100),N3MAX
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/STMA/SEXTSV(23,100),SINTSV(10,100),NSMAX,MAXSEX,MAXSIN
C
C     ******
C
      DO 8  I  =1,NSMAX
      IF(SEXTSV(1,I  ))    5,5,3
    3 IF(SINTSV(2,I  )-1.) 7,6,7
C
C  FLAG UNUSED STREAM NUMBERS AS -1
    5 KSFLAG(I  )=-1
      GO TO 8
C
C  FLAG FEED STREAMS EQUAL TO 1
    6 KSFLAG(I  )= 1
      GO TO 8
C
C  FLAG PRODUCT AND INTERMEDIATE STREAMS EQUAL TO 0
    7 KSFLAG(I  )= 0
    8 CONTINUE
      DO 9 I  =1,NEMAX
C
C  FLAG UNUSED EQUIPMENT NUMBERS AS -1, OTHERS AS 0
      KEFLAG(I)=0
      IF(KPM(1,I) .GT. 0 ) GO TO 9
      KEFLAG(I  ) = -1
    9 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SUBSET
C
C     ENTRY  SETOV  REQUIRED FOR RETURN FROM LOWER LEVEL 6400 OVERLAY
C
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSAA/TITLE(20),COMPNT(20),KOMNAM(80)
      COMMON/SYSA/KPM(10,50),KSEM(3,100),N3MAX
      COMMON/SYSB/KE1(50),NE1MAX,KE2(50),NE2MAX,KE3(50),NE3MAX,
     1KE4(10),NE4MAX,KRET,KRET2,KRET3
      COMMON/SYSC/LIMIT,LIMIT2,LIMIT3,LOOP,LOOPS
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/EQPB/NECALL(50),NEXEQN(50),NAME(50)
      COMMON/STMA/SEXTSV(23,100),SINTSV(10,100),NSMAX,MAXSEX,MAXSIN
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/OVR/NRET
C
C     ******
C
      INTEGER TITLE,COMPNT
      DATA CHESS,REC/5HCHESS,1H /
C
      NAMELIST/FLLIST/ KEFLAG,KSFLAG
      NAMELIST/DATOUT/SONUM,SOVPFR,SOTEMP,SOPRES,SOENTH,SOMOLE,SOCOMP
      IF(IPUNCH.GE.1) GO TO 61
      KRET2=0
      SAVE2 =NE2MAX
      SAVE3 =NE3MAX
C
C  FEED STREAM FLAGS ARE (+1). UNUSED EQUIP. AND STREAM ARE (-1).
C  UNKNOWN STREAMS ARE (0) AS ARE EQUIP. NOT COMPUTED.
C   KES IS *+1* IF ANY EQUIPMENT IS CALCULATED IN KPM SCAN IN SUBSET.
      CALL SCAN
   21 KES=0
      LOOP=0
C  START PROCESS MATRIX SCAN COLUMN BY COLUMN
C
      I2=1
   22 CONTINUE
      NE=KPM(1,I2)
      IF(NE) 24,24,26
   26 IF(KEFLAG(NE).NE.0) GO TO 24
C
C  START SCAN OF COLUMN OF UNCALC*D EQUIP. LOOK AT SIGN OF STREAM
C
   27 DO 42 I=2,N3MAX
C
C  CHECK SIGN OF STREAM IN PROCESS MATRIX
C
   28 IF(KPM(I ,I2)) 44,42,29
C
C  IF STREAM IS INPUT SEE IF IT IS KNOWN. IF NOT GO TO NEXT EQUIPMENT.
C
   29 I6=KPM(I ,I2)
   40 IF(KSFLAG(I6).LE.0) GO TO 24
   42 CONTINUE
C  OUTPUT STREAM REACHED WITH INPUTS KNOWN.
C    CALL OUT SUBROUTINE OF EQUIPMENT *NE*. CALC. OUTPUT STREAMS
   44 CALL EQUIP
      NRET=2
      CALL EQCALL
      GO TO 101
      ENTRY SETOV
  101 IF(KTRACE.EQ.3) WRITE(6,DATOUT)
      CALL REQUIP
   46 KES=1
   47 KEFLAG(NE)=1
C  SET OUTPUT STREAM FLAGS FROM 0 TO 1
C
   48 DO 50 I7=2,N3MAX
      I8=KPM(I7,I2)
      IF(I8.GE.0) GO TO 50
      I8=-I8
      KSFLAG(I8)=1
   50 CONTINUE
   24 I2=I2+1
      IF(I2.LE.NEMAX) GO TO 22
C  THIS COMPLETES A SCAN OF PROCESS MATRIX COLUMN BY COLUMN
C
C     IF KES IS 1 RESCAN PROCESS MATRIX
C
C     IF KES IS 0, SEE IF ALL EQUIP. HAS BEEN CALC. IF NOT GO TO RCYCLE
   55 IF(KES.GT.0) GO TO 21
C
C     ARE ALL EQUIP. KNOWN<  IF NOT GO TO RCYCLE
      DO 58 I9=1,NEMAX
      IF(KEFLAG(I9).NE.0) GO TO 58
      IF(IPUNCH.LT.0 .AND. KRET2.EQ.0) WRITE(7,FLLIST)
C  START TRIAL + ERROR CALCULATION OF RECYCLE STREAMS.
C
   61 CALL RCYCLE
C
C  IF KRET2 IS 2, EQUIP. LIST FOR RECYCLE CALC. NOT AVAIL. IN *RCYCLE*
      IF(KRET2.NE.2) GO TO 21
      GO TO 67
   58 CONTINUE
C
C  *SUBSET* LOOP COMPLETE.
   67 NE2MAX=SAVE2
      NE3MAX=SAVE3
      CALL OVERLAY(CHESS,4,0,REC)
      END
```

```
      SUBROUTINE TEST
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSB/KE1(50),NE1MAX,KE2(50),NE2MAX,KE3(50),NE3MAX,
     1KE4(10),NE4MAX,KRET,KRET2,KRET3
      COMMON/SYSC/LIMIT,LIMIT2,LIMIT3,LOOP,LOOPS
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
      COMMON/STMA/SEXTSV(23,100),SINTSV(10,100),NSMAX,MAXSEX,MAXSIN
      COMMON/STMOUT/SONUM(8),SOFLAG(8),SOVPFR(8),SOTEMP(8),
     1SOPRES(8),SOENTH(8),SOVISC(8),SOTHK(8),SOLZ(8),SOVZ(8),
     2SOMOLE(8),SOCOMP(20,8),SOKV(20,8)
      COMMON/MTST/SAVEX(10,20),SAVEF(10,20),SAVEQ(10,20)
C
C     ******
C
      DIMENSION DUM(1)
      DIMENSION SOIDUM(8,10)
      EQUIVALENCE (SONUM(1),SOIDUM(1,1))
      LOGICAL FLAG,FFLAG
C
C
      LIMIT=0
      LIMIT3=0
      I=1
   20 J=SOIDUM(I,1) + 0.01
C
C  J IS OUTPUT STREAM NUMBER OF EQUIP.(NE)
C  USE *DERROR* TO CHECK OUTPUT STREAM VALUES AGAINST *SEXTSV*,*SINTSV*.
C
      TMOLE=SEXTSV(3,J)
      IF(ABS(TMOLE).LT.1.E-20) GO TO 21
      IF(ABS((TMOLE-SOMOLE(I))/TMOLE).GT.DERROR) LIMIT3=LIMIT3+1
      GO TO 22
   21 IF(ABS(SOMOLE(I)).GT.DERROR) LIMIT3=LIMIT3+1
      GO TO 22
   24 CONTINUE
      DO 26 K=3,MAXSIN
      SINK=SINTSV(K,J)
      IF(ABS(SINK).LT.1.E-20) GO TO 25
      IF(ABS((SINK-SOIDUM(I,K))/SINK).GT.DERROR) LIMIT3=LIMIT3+1
      GO TO 26
   25 IF(ABS(SOIDUM(I,K)).GT.DERROR) LIMIT3=LIMIT3+1
   26 CONTINUE
      IF(LIMIT3.LE.0) GO TO 28
      IF(KTRACE.EQ.3) WRITE(6,30)LOOP,J,NE,LIMIT3
      LIMIT3=0
      KIMIT=LIMIT+1
   28 IF( I .GE. NOUT ) RETURN
      I=I+1
      GO TO 20
   30 FORMAT(*0IN LOOP*,I4,* OUTPUT STREAM*,I4,* OF EQUIP.*,I4,* HAS*,
     1I4,* UNCONV. STREAM VALUES*)
C
C  WEGSTEIN*S ITERATIVE METHOD ..........................................
   22 LIMIT4=0
      FLAG=.FALSE.
      FFLAG=.FALSE.
C...  TEST PERMISSIBLE STREAM NUMBERS ... 0 + NE4MAX .LE. 10
      IF(NE4MAX.LE.0 .OR. NE4MAX.GT.10) GO TO 42
      DO 40 IK=1,NE4MAX
      IF(IK.GT.10) GO TO 42
      IF(J.EQ.KE4(IK)) GO TO 41
   40 CONTINUE
      GO TO 42
   41 FLAG=.TRUE.
   42 DO 55 K=1,NOCOMP
      SOCOM=SOCOMP(K,I)
      COMPK=SEXTSV(K+3,J)
      IF(ABS(COMPK).LT. 1.E-20) GO TO 27
      IF(ABS((COMPK-SOCOM      )/COMPK).LT.DERROR) GO TO 43
      GO TO 47
   27 IF(ABS(SOCOM      ).GT.DERROR) GO TO 47
   43 IF(.NOT.FLAG) GO TO 55
C...  NOW TEST LOOP COUNTER AND BRANCH FOR 1, 2 , .GE.3
      IF(LOOP-2) 44,45,46
   44 SAVEX(IK,K)=SOCOM
      GO TO 55
   45 SAVEF(IK,K)=SOCOM
      GO TO 55
   46 SAVEX(IK,K)=COMPK
      SAVEF(IK,K)=SOCOM
      GO TO 55
   47 LIMIT4=LIMIT4+1
      IF(LOOP.LT.3) GO TO 43
      IF(.NOT.FLAG) GO TO 55
C...  CALCULATE WEGSTEIN*S COEFFICIENTS - PHI + Q ...
      QOLD=SAVEQ(IK,K)
      PHI=(COMPK-SAVEX(IK,K))/(SOCOM      -SAVEF(IK,K))
      IF(1.-PHI) 48,50,48
   48 Q=1./(1.-PHI)
      IF(Q.LT.0.0)GO TO 52
      IF(KTRACE.GT.0) WRITE(6,49) J,I,Q
   49 FORMAT(*0WEGSTEIN/TEST2 STREAM*,I4,*-*,I3,5X,*Q=*,F10.3)
      Q=1.E-10
      GO TO 53
   50 IF(KTRACE.GT.0) WRITE(6,51) J,I,PHI
   51 FORMAT(*0WEGSTEIN/TEST1 STREAM*,I4,*-*,I3,5X,*PHI=*,E15.5)
      GO TO 46
   52 IF(Q.GT.-10.0) GO TO 53
      IF(KTRACE.GT.0) WRITE(6,49) J,I,Q
      Q=-10.0
   53 X2=Q*COMPK+(1.-Q)*SOCOM
      IF(KTRACE.GT.0) WRITE(6,36) J,K,COMPK,SOCOM,X2,Q,SAVEX(IK,K),
     -                           SAVEF(IK,K)
   36 FORMAT(*0STREAM*,I4,*-*,I3,5X,*X=*,E13.5,5X,*F=*,F13.5,5X,*XNEW=*,
     1 F13.5,5X,*Q=*,F10.3/ 16X,*SAVEX=*,E13.5,* SAVEF=*,E13.5)
      IF(X2.LE.0.) X2=SOCOM/2.
      FFLAG=.TRUE.
      IF(Q*QOLD.LT.0..OR.LOOP.EQ.3) GO TO 54
      IF((X2-COMPK)*(SAVEF(IK,K)-COMPK-1.E-10).LT.0.) GO TO 54
      SOCOMP(K,I)=COMPK
      GO TO 37
   54 SOCOMP(K,I)=X2
```

```
   37 SAVEX(IK,K)=COMPK
      SAVEF(IK,K)=SOCOM
      SAVEQ(IK,K)=Q
   55 CONTINUE
   56 LIMIT3=LIMIT3+LIMIT4
      IF(FLAG.AND.FFLAG) GO TO 57
      GO TO 24
   57 TMOLE=0.
      DO 58 K=1,NOCOMP
   58 TMOLE=TMOLE+SOCOMP(K,I)
      IF(KTRACE.GT.0) WRITE(6,59) J,SOMOLE(I),TMOLE
   59 FORMAT(*0STREAM*,I4,* OLD AND NEW T-MOLES..*,2F15.5)
      SOMOLE(I)=TMOLE
      CALL ENTH(-I,SOENTH(I),DUM)
      GO TO 24
      END
```

```
      SUBROUTINE TRANSX(I,LTH,KFLOG,N1,IV,IA)
C
C     SAME FUNCTION AS *TRANSF* FOR INTEGER + LITERAL ARRAYS
      DIMENSION KFLOG(N1),IV(N1),IA(5)
      DO 7 K=I,LTH
      K1=K-I+1
      K2=KFLOG(K)
    7 IA(K1)=IV(K2)
      RETURN
      END
```

```
      SUBROUTINE TRANSF(I,LTH,KFLAG,N1,SV,N2,N3,A,L)
C
C**THIS SUBROUTINE TRANSFERS DATA FROM TWO DIMENSIONAL ARRAY SV(N2,N3)
C** TO A ONE DIMENSIONAL A(1)...A(5) FOR PRINT-OUT.
      DIMENSION  KFLAG(N1), SV(N2,N3), A(5)
      DO 5 K=I,LTH
      K1=K-I+1
      K2=KFLAG(K)
    5 A(K1)=SV(L,K2)
      RETURN
      END
```

```
      SUBROUTINE TPRINT
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSA/KPM(10,50),KSFM(3,100),N3MAX
      COMMON/SYSAA/TITLE(20),COMPNT(20),KOMNAM(80)
      COMMON/SYSB/KE1(50),NE1MAX,KE2(50),NE2MAX,KE3(50),NE3MAX,
     1KF4(10),NE4MAX,KRFT,KRET2,KRET3
      COMMON/SYSC/LIMIT,LIMIT2,LIMIT3,LOOP,LOOPS
      COMMON/SYSD/KEFLAG(50),KSFLAG(100),KTRACE,DERROR,NPFREQ,IPUNCH
      COMMON/EQPA/EQPAR(25,50),NEMAX,MAXEQP
      COMMON/EQPB/NECALL(50),NEXEQN(50),NAME(50)
      COMMON/STMA/SEXTSV(23,100),SINTSV(10,100),NSMAX,MAXSEX,MAXSIN
      COMMON/CONTL/NIN,NOUT,NOCOMP,NE,NEN
C
C     ******
C
      INTEGER COMPNT,PFLAG(100) ,TITLE
      DIMENSION IPNT(5),JPNT(5),RPNT(5)
      INTEGER LABEL(32)
      DATA LABEL/4HVAPO,4HR FR,4HACTI,4HON  ,4HTEMP,4HFRAT,4HURE,,4H R
     1,4HPRES,4HSURE,4H, PS,4HIA  ,4HENTH,4HALPY,4H, BT,4HU   ,4HVISC,4H
     2OSIT,4HY, P,4HFH  ,4HK, B,4HTU-F,4HT-HR,4H-F  ,4HZ-FA,4HCTOR,4H, L
     3I,4HQ,  ,4HZ-FA,4HCTOR,4H, VA,4HP,  /
C
C***** THIS PORTION WAS TO PRODUCE RELOAD DECK.....TEMPORARILY DELETED<
C
      RETURN
C
C*****  BIPRNT  *****
C
      ENTRY BIPRNT
C...COMMON AREA PUNCH DELETED....
C
      WRITE(6,100)TITLE
  100 FORMAT(1H1,3X,15H FINAL RESULTS ,//13X,20A4,//32X,16H STREAM SUMMA
     2RY )
      GO TO 102
C
C*****  PTPRNT  *****
C
      ENTRY PTPRNT
      WRITE(6,101)TITLE
  101 FORMAT(1H1,3X,15H INPUT DATA    ,//13X,20A4,//32X,16H STREAM SUMMA
     2RY )
  102 CONTINUE
      J=0
      DO 12 I=1,NSMAX
      IF(SEXTSV(1,I).EQ.0.) GO TO 12
      J=J+1
       PFLAG(J)=I
   12 CONTINUE
      DO 24 I=1,J,5
      LTH=I+4
      IF(LTH.GT.J) LTM=J
      WRITE(  6,14) ( PFLAG(K),K=I,LTH)
   14 FORMAT(*0STREAM NUMBER   *,5(I9,5X))
```

```
C
      DO 15 K=I,LTH
      KPNT=K-I+1
      MPNT= PFLAG(K)
      IPNT(KPNT)=KSEM(2,MPNT)
   15 JPNT(KPNT)=KSEM(1,MPNT)
      LTHP=LTH-I+1
      WRITE(  6,16)  (IPNT(K),JPNT(K),K=1,LTHP)
   16 FORMAT(*0EQUIP. CONXION *,5(5H  FR ,I2,4H TO ,I2,1X))
      DO 18 L=3,MAXSIN
      CALL TRANSF(I,LTH, PFLAG,100,SINTSV,10,100,RPNT,L)
      K1 =4*L-11
      K2 =K1 +3
   18 WRITE(  6,19)(LABEL(K) , K=K1,K2),  (RPNT(K),K=1,LTHP)
   19 FORMAT(* *,4A4,5F14.4)
      WRITE(  6,20)
   20 FORMAT(*0*,40X,*COMPOSITION, LB-MOLES/HOUR* / )
      NOCOP3=NOCOMP+3
      DO 22 L=4,NOCOP3
      CALL TRANSF(I,LTH, PFLAG,100,SEXTSV,23,100,RPNT,L)
      K1=4*L-15
      K2=K1+3
   22 WRITE(  6,19)(KOMNAM(K),K=K1,K2),(RPNT(K),K=1,LTHP)
      CALL TRANSF(I,LTH, PFLAG,100,SEXTSV,23,100,RPNT,3)
      WRITE(  6,23)  (RPNT(K),K=1,LTHP)
   23 FORMAT(*0*,11X,*TOTAL*5F14.4)
   24 WRITE(  6,26)
   26 FORMAT(//////)
      RETURN
      END
```

```
      SUBROUTINE PTEQPT(NN)
C
C     ORIGINALLY ENTRY IN TPRINT
C     SEPARATED FOR CDC6400 OVEPLAY OPERATION
C
C     PRODUCES EQUIPMENT SUMMARY
C
C     ******  COMMON DECK  ******
C
      COMMON/SYSAA/TITLE(20),COMPNT(20),KOMNAM(80)
C
C     ******
C
      INTEGER TITLE,COMPNT
      IF(NN.GT.0) GO TO 3
      WRITE(6,1)TITLE
    1 FORMAT(1H1,3X,15H INPUT DATA     ,//13X,20A4,//)
      GO TO 4
    3 WRITE(6,2)TITLE
    2 FORMAT(1H1,3X,15H FINAL RESULTS ,//13X,20A4,//)
    4 CONTINUE
      CALL EQPRNT
      RETURN
      END
```

```
      SUBROUTINE ZERO (A,N)
      DIMENSION A(N)
      DO 3 I=1,N
    3 A(I)=0.
      RETURN
      END
```

```
      SUBROUTINE ZEROX(K,N)
      DIMENSION K(N)
      DO 1 I=1,N
    1 K(I)=0
      RETURN
      END
```