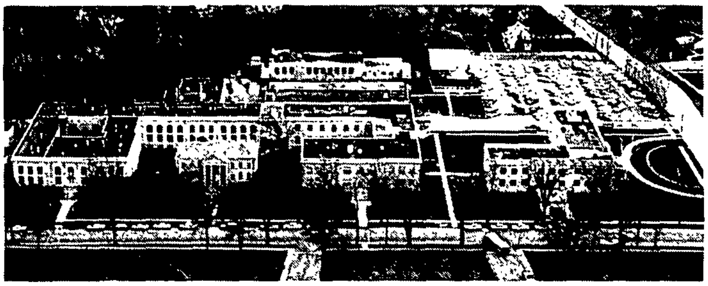
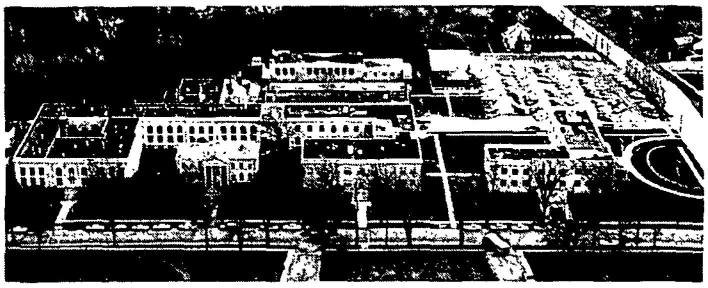


THE INSTITUTE OF PAPER CHEMISTRY, APPLETON, WISCONSIN

A GUIDE TO MAPPS SIMULATION

SYSTEMS GROUP
PAPER MATERIALS AND SYSTEMS DIVISION

JANUARY, 1983



THE INSTITUTE OF PAPER CHEMISTRY, APPLETON, WISCONSIN

A GUIDE TO MAPPS SIMULATION

SYSTEMS GROUP
PAPER MATERIALS AND SYSTEMS DIVISION

JANUARY, 1983

TABLE OF CONTENTS

1. INTRODUCTION

MAPPS (Modular Analysis of Pulp and Paper Systems) is a computer program developed at The Institute of Paper Chemistry for simulating processes in integrated, kraft pulp and paper mills. The main thrust behind the development of MAPPS is to provide a tool for simulating the steady state behavior of the "whole mill." Particular emphasis is given to the calculation of material and energy flows.

Since no two mills are the same, a modular analysis approach is used. A module is a subprogram containing a mathematical description of the material and energy flows and important performance parameters of a particular process or piece of equipment. Modules are connected through streams carrying material and energy flows. Many modules can be connected to "simulate" a particular mill configuration.

For simulation purposes, the whole mill can be divided into five subsystems - the pulp mill, bleach plant, paper mill, steam and power system, and waste treatment system. MAPPS currently has a library of modules for the pulp mill, paper mill, and steam and power system. Bleach plant and waste treatment system modules will be developed and added to MAPPS.

MAPPS uses a sequential solution algorithm. Modules are executed in a sequential order specified by the user. The executive program, which contains the solution algorithm, is a modified version of the GEMCS (General Engineering and Management Computation System) program developed by A. I. Johnson at McMaster University.

Three programs have had a direct influence in the development of MAPPS as a simulator for the pulp and paper industry. Two of these are GEMCS based: a program developed by S. S. Treiber and T. J. Boyle at McGill University and a program developed by L. J. Pitzner at The Institute of Paper Chemistry. The third is a dynamic simulator, DYSCO, developed by L. A. Lopez and B. Carnahan at the University of Michigan, which was converted to a pulp and paper simulator by J. A. Newcombe and P. E. Parker at The Institute of Paper Chemistry.

This user's manual is intended to be an instructional text and reference source for developing MAPPS simulations. The manual begins with a discussion of the principles of sequential modular simulation. The next section discusses stream and module vectors, which are used to transfer and store information. Next, the available modules are described; this section is divided into a discussion of utility modules and process modules. The methodology of obtaining a converged solution for a simulation is discussed in the next section. This is followed by a discussion of the procedure for assembling the data needed to develop a MAPPS simulation. The last two sections discuss running MAPPS in either an interactive or batch mode.

2. PRINCIPLES OF SEQUENTIAL SIMULATION

MAPPS uses a sequential solution algorithm to solve for stream and module parameters. Modules are executed sequentially, with the process sometimes requiring several iterations through the modules.

The basic concepts underlying sequential modular simulation are best described in the context of an example. Two subprocesses in a pulp mill, digester blowdown followed by brown stock washing, will be considered. The process flow diagram is given in Fig. 2.1.

In this process, pulp and black liquor from the digester enter the blow tank, where a drop in pressure and temperature causes steam to be flashed. The pulp is then washed free of the black liquor in a set of brown stock washers. For simulation purposes, the process flow diagram is converted into the process flowsheet of Fig. 2.2.

In the process flowsheet, process streams are represented by numbered stream vectors, and processes are represented by numbered modules. A stream vector is an ordered set of numbers characterizing the physical state of a process stream. Typical elements are temperature, pressure, and constituent flow rates. A module consists of mathematical equations modeling the operation of a process, subprocess, or piece of equipment. These equations describe material and energy balances, using both fundamental engineering principles and empirical data or correlations. A module may also contain equations that estimate the values of performance parameters. The module equations are included in a FORTRAN subprogram that calculates the stream vectors leaving a process module, given the stream vectors entering the module.

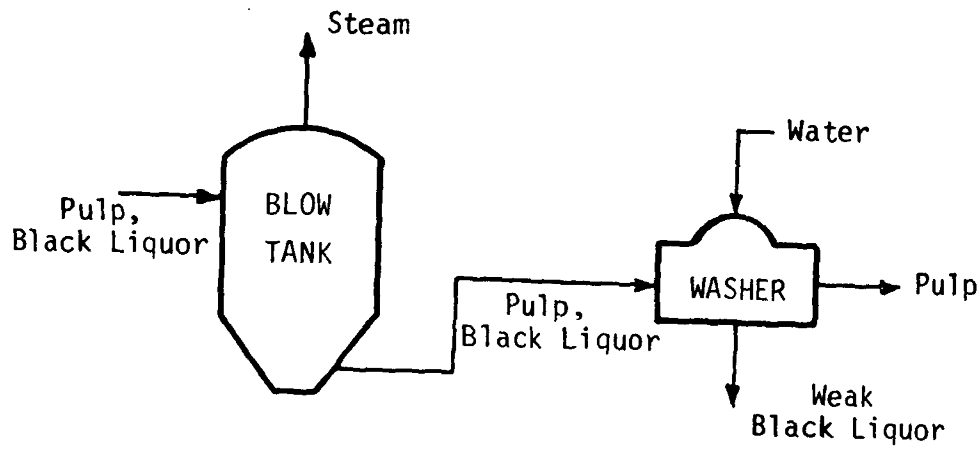


Figure 2.1 Process Flow Diagram of Blowdown Followed by Brown Stock Washing

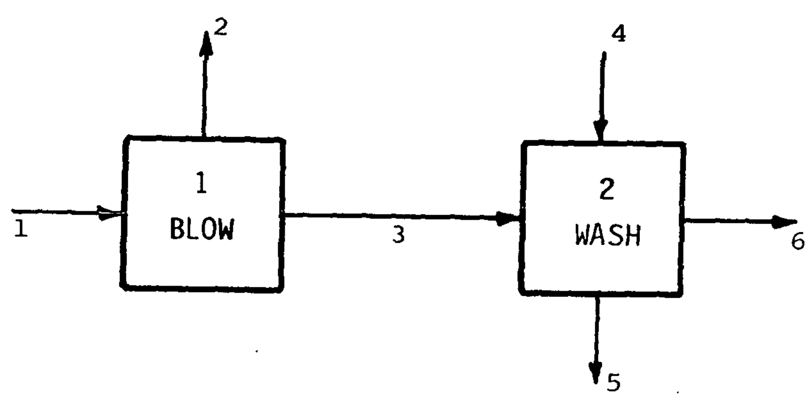


Figure 2.2 Process Flowsheet of Blowdown Followed by Brown Stock Washing

The blow tank module calculates the outlet steam vector and the pulp and black liquor stream vector, given the inlet stream vector. Note, however, that it is insufficient to specify only the inlet stream. It is also necessary to specify the pressure to which the inlet stream will be flashed. This parameter characterizes the operation of the blow tank and is not associated with the inlet stream. Similarly, for the washer, parameters such as displacement ratio and mat consistency must be specified. Therefore, each module has a vector of parameters characterizing the performance of the particular piece of equipment or process.

3. STREAM AND MODULE VECTORS

Information is transferred and stored in vectors, which are one-dimensional arrays. Two types of vectors are used - stream and module vectors. Several of the elements of these vectors require units to describe their magnitude. The following sections describe the units used, the characteristics of stream and module vectors, and the use of stream vectors in flowsheets.

3.1 UNITS

When constructing a data file for a MAPPS simulation, the user can use either English Engineering or SI units. A description of the units for several parameters is presented in Table 3.1.

3.2 STREAM VECTORS

Stream vectors are used to communicate information between module subprograms. Material and energy flows, thermodynamic properties, or other parameters can be carried in these vectors. Currently, MAPPS recognizes ten (10) different types of streams. Eight carry material and energy flows, one carries only energy flows, and one carries information. The ten types of streams are described in Table 3.2. The bleach and waste streams are not currently used by any modules.

The vector elements for each type of stream are defined in Table 3.3. Each element is identified with a label up to eight characters long. In all cases, element 1 is reserved for the stream number, and element 2 is reserved for the stream type, which is designated by a number.

Table 3.1 Units Used in MAPPS

Parameter	English Engineering	SI
Mass	lb_m	kg
Length	ft	m
Time	hr	hr
Temperature	F	C
Force	lb_f	N
Pressure	psia	kPa
Energy	Btu	kJ
Heat capacity	Btu/(lb_m-F)	kJ/(kg-C)

Table 3.2 Description of Stream Types

Stream Type	Stream Label	Description
(1)	WATER	for the flow of steam and condensate
(2)	GASEOUS	for the flow of air, flue gas, and volatile constituents
(3)	PULPING	for the flow of wood, pulp, and black liquor
(4)	RECOVERY	for the flow of constituents related to the recovery of cooking chemicals (smelt through white liquor)
(5)	BLEACHING	for the flow of constituents related to the bleaching process
(6)	PAPER	for the flow of pulp slurry in the paper mill area
(7)	WASTE	for the flow of constituents related to waste treatment
(8)	FUEL	for the flow of various fuel types, such as oil, coal, natural gas, and wood
(9)	ENERGY	for the transfer of thermal, mechanical, or electrical energy that is not accompanied by the transfer of mass
(10)	INFO	for the transfer of information

Table 3.3 Description of Vector Elements

WATER (1)		GASEOUS (2)		PULPING (3)	
1	NUMBER	1	NUMBER	1	NUMBER
2	TYPE	2	TYPE	2	TYPE
3	TEMP	3	TEMP	3	TEMP
4	PRESS	4	PRESS	4	PRESS
5	HEAT CAP	5	HEAT CAP	5	HEAT CAP
6	QUALITY	6		6	
7	SP ENTH	7		7	
8	ENTHALPY	8	ENTHALPY	8	ENTHALPY
9	EXERGY	9	EXERGY	9	EXERGY
10	FLOW	10	FLOW	10	FLOW
11	WATER (L)	11	WATER (G)	11	WATER (L)
12		12	O2	12	CELLULOS
13		13	N2	13	LIGNIN
14		14	CO2	14	DIS CELL
15		15	CO	15	DIS LIGN
16		16	SO2	16	SHIVES
17		17	INERT	17	INERT
18		18	CL2	18	
19		19	CLO2	19	TALL OIL
20		20	TURPENTN	20	TURPENTN
21		21	CH3OH(G)	21	CH3OH(L)
22		22	CH3SH	22	CH3SH
23		23	H2S	23	H2S
24		24	NA2CO3	24	NAOH BND
25		25	NA2SO4	25	NEUT OH-
26		26	CAO	26	OH-
27		27		27	SH-
28		28		28	S-2
29		29		29	S2O3-2
30		30		30	SO4-2
31		31		31	CO3-2
32		32		32	NA+
33		33		33	K+
34		34		34	CA+2
35		35		35	CL-
36		36		36	
37		37		37	
38		38		38	
39		39		39	
40		40		40	

Table 3.3 Description of Vector Elements (Continued)

	RECOVERY (4)		BLEACHING (5)		PAPER (6)
1	NUMBER	1	NUMBER	1	NUMBER
2	TYPE	2	TYPE	2	TYPE
3	TEMP	3	TEMP	3	TEMP
4	PRESS	4	PRESS	4	PRESS
5	HEAT CAP	5	HEAT CAP	5	HEAT CAP
6		6		6	
7		7		7	
8	ENTHALPY	8	ENTHALPY	8	ENTHALPY
9	EXERGY	9	EXERGY	9	EXERGY
10	FLOW	10	FLOW	10	FLOW
11	WATER(L)	11	WATER(L)	11	WATER(L)
12	CACO3	12	CELLULOS	12	FIBER
13	CA(OH)2	13	LIGNIN	13	FINES
14	CAO	14	DIS CEL	14	DIS SLDS
15	CAO BRNT	15	DIS LIGN	15	ADDITIVE
16	CASO4	16	SHIVES	16	DIRT
17	INERT	17	INERT	17	
18	NA2CO3	18	CL2	18	
19	NA2S	19	CLO2	19	
20	NA2SO4	20	O2	20	
21	CARBON	21	OX CELL	21	
22		22	CL LIGN	22	
23		23	CL BOUND	23	
24		24	NAOH BND	24	
25		25	H+	25	
26	OH-	26	OH-	26	
27	SH-	27	CLO-	27	
28	S-2	28	CLO2-	28	
29	S2O3-2	29	CLO3-	29	
30	SO4-2	30	SO4-2	30	
31	CO3-2	31	CO3-2	31	
32	NA+	32	NA+	32	
33	K+	33	K+	33	
34	CA+2	34	CA+2	34	
35	CL-	35	CL-	35	
36		36		36	
37		37		37	
38		38		38	
39		39		39	
40		40		40	

	WASTE (7)	FUEL (8)	ENERGY (9)
1	NUMBER	NUMBER	NUMBER
2	TYPE	TYPE	TYPE
3	TEMP	TEMP	
4	PRESS	PRESS	
5	HEAT CAP	HEAT CAP	
6		HT COMB	ELEC/MEC
7			THERMAL
8	ENTHALPY	ENTHALPY	ENERGY
9	EXERGY	EXERGY	EXERGY
10	FLOW	FLOW	
11		MOISTURE	
12		CARBON	
13		HYDROGEN	
14		OXYGEN	
15		NITROGEN	
16		SULFUR	
17		INERT	
18		SODIUM	
19		POTASIUM	
20		CALCIUM	
21		CHLORINE	
22			
23			
24			
25			

INFO (10)

1 NUMBER
2 TYPE
3 INFO
4 INFO
.
.
.
.

Thermodynamic properties appear in elements 3 through 9. Note that some thermodynamic property elements are not currently being used, as indicated by the absence of a label. A description of the thermodynamic properties is presented in Table 3.4. The energy-related properties have units of Btu/hr or kJ/hr. For example, ENTHALPY is the product of the mass flow rate and the specific enthalpy.

Exergy, which is also called "available energy," is defined as

$$\text{EXERGY} = \text{ENTHALPY} - (\text{mass flow rate}) \times T_o \times (\text{specific entropy})$$

where T_o is the thermodynamic reference temperature (537 R or 298 K). ELEC/MEC denotes electrical or mechanical energy transfer. THERMAL denotes thermal energy transfer. The element ENERGY, which appears in an energy stream, represents the sum of the electrical, mechanical, and thermal energy.

Specific enthalpy is based on a 77 F (25 C) reference temperature. The following chemical species were used to define the zero enthalpy reference state: molecular oxygen, molecular nitrogen, water, carbon dioxide, sulfur dioxide, hydrogen ion, sodium ion, potassium ion, calcium ion, and chlorine ion. Using this reference state, the enthalpy of a particular organic species is equal to its heating value.

Using a 77 F reference temperature, the enthalpy of liquid water is zero at 77 F. However, the Steam Tables are based on a 32 F reference. To compare enthalpy values from MAPPS with those obtained from the Steam Tables, 45.0 Btu/lb_m (104.7 kJ/kg) should be added to the enthalpy calculated by MAPPS.

In material streams, element 10 contains the total mass flow rate. The constituent mass flow rates are located in elements 11 through 11 + n, where n is the number of constituents.

In a FUEL stream, the constituent mass flow rates are expressed in terms of an ultimate analysis of the fuel. The heat capacity (element 5) and the heat of combustion of the fuel (element 6) are based on "dry" fuel. When the fuel contains free moisture, in the "as received" condition, this moisture should appear in element 11. Element 17, which is labeled INERT, is used for ash or other non-reacting constituents, such as dirt. In Table 3.5, fuel streams for Douglas-fir, no. 6 fuel oil, and a typical black liquor are illustrated.

3.3 MODULE VECTORS

MAPPS uses module vectors to communicate with individual modules. The elements of the module vectors are referred to as "given" or "calculated" parameters. Given module parameters are supplied by the user. Calculated module parameters are calculated during execution of the module subprogram. Module vectors begin with a set of given parameters and end with calculated parameters.

The standard format for the first five given parameters is

(1) module number

(2) module type

(3) number of parameters

(4) number of given streams

(5) number of calculated streams

The module number is obtained from the flowsheet. The module type is a numerical identifier used by the executive program to identify the particular module subprogram. Module types are presented in Tables 4.1-4.6.

Table 3.5 Examples of Fuel Streams[a]

Element		Douglas-fir Dry	Douglas-fir 50% Moisture[b]	No. 6 Fuel Oil	Black Liquor[c]
5	HEAT CAP	0.33	0.33	0.50	0.50
6	HT COMB	9050	9050	18,300	6000
10	FLOW	100	100	100	100
11	MOISTURE		50.00		35.0
12	CARBON	52.3	26.15	85.6	25.2
13	HYDROGEN	6.3	3.15	9.7	2.5
14	OXYGEN	40.5	20.25	2.0	22.9
15	NITROGEN	0.1	0.05		
16	SULFUR			2.3	2.2
17	INERT	0.8	0.40	0.4	
18	SODIUM				12.2
19	POTASIUM				
20	CALCIUM				
21	CHLORINE				

[a]English Engineering units.
[b]HEAT CAP and HT COMB "as received" are 0.665 and 4525, respectively.
[c]Typical black liquor, 65% solids.

The number of parameters depends on the particular module and is usually a fixed value; however, in some modules, a simple arithmetic calculation must be made. Parameters 4 and 5 represent the number of "given" and "calculated" streams, respectively. The term "given streams" refers to those streams where the stream vector elements are used in the module calculations. "Calculated streams" refer to those streams where the stream vector elements are calculated during module execution. Usually, given streams flow into a module, and calculated streams flow out of a module.

Module vectors also use a standard format for parameters 6 through 15. Beginning with parameter 6, the given stream numbers are listed, followed by the calculated stream numbers.

3.4 USING STREAM VECTORS IN FLOWSHEETS

MAPPS, which uses a sequential solution algorithm, treats most inlet streams to a module as given streams (i.e., streams with known or user-specified components). Most outlet streams are calculated during execution of the module. Exceptions to this treatment of streams can occur. An inlet stream can be calculated during execution of the module, and an outlet stream can be given. To flag "calculated" inlet streams and "given" outlet streams so that mass and energy balances can be correctly calculated, the stream type (element 2) is given a negative value.

For example, air flowing into the recovery furnace module (FURN01) is a "calculated" inlet stream having its second vector element equal to -2 (since air is a gaseous stream, type 2). In the extraction turbine module (TURB01), the steam extracted at each extraction port is specified. Thus, this stream is a "given" outlet stream having stream element 2 equal to -1 (since steam is a water stream, type 1).

Another special situation occurs when a module needs information from a stream that is already being used by another module. The same stream is used in both locations, but in this special situation, the stream is treated as an information stream and is not included in the mass and energy balance of the module. This situation occurs in several controller modules where the flow rate of a particular mass constituent must be adjusted, based on its flow rate at a particular point in the flowsheet.

As previously described, a flowsheet consists of stream vectors, constructed using solid lines, with the arrow pointing in the mass or energy flow direction. To accommodate the special stream vector situations just described, the following procedure has been adopted. "Calculated" inlet streams and "given" outlet streams are drawn as broken solid lines, and information streams appear as dotted lines; the arrow always points in the flow direction. Using this methodology, a flowsheet is also an information flow diagram illustrating how MAPPS handles the flow of information, material, and energy.

As an example of these concepts, consider the flowsheet in Fig. 3.1. This flowsheet comprises four modules and twelve streams. Stream 3, a material (or energy) stream leaving module 1, is used as an information stream entering module 3. Stream 2 is a "calculated" inlet process stream, and stream 9 is a "given" outlet process stream.

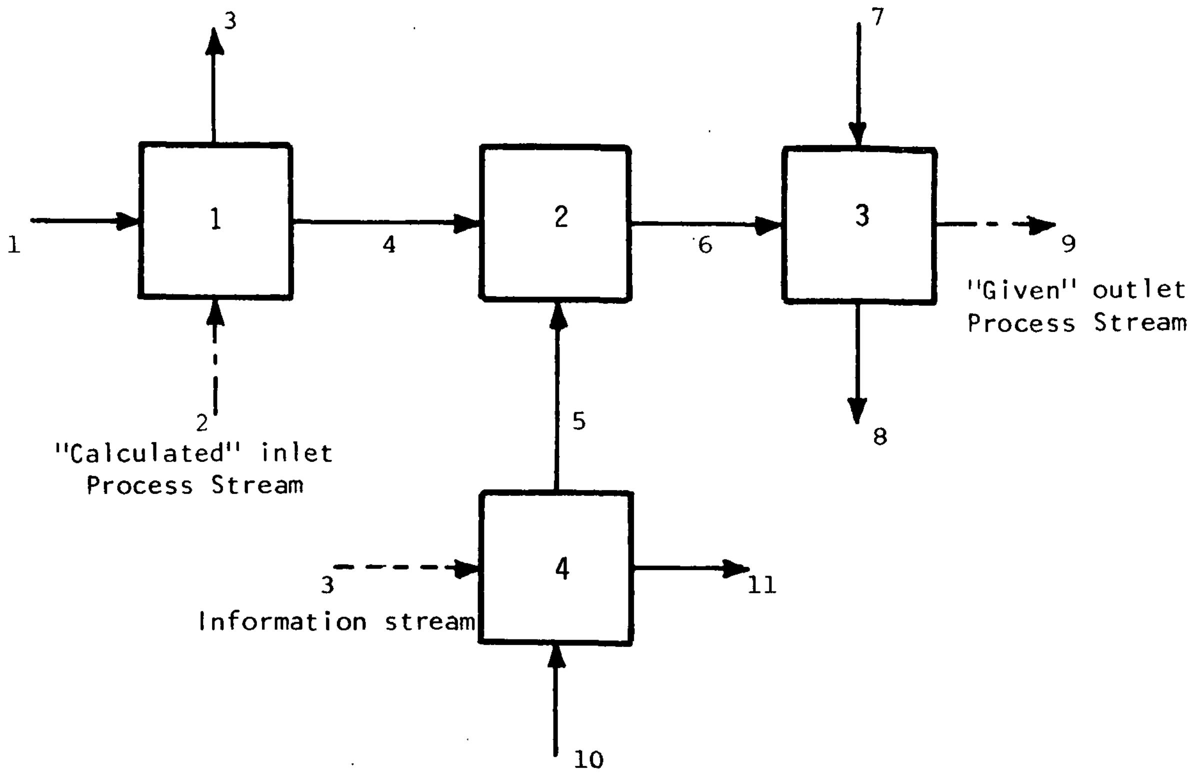


Figure 3.1 Example Flowsheet Illustrating the Different Types of Stream Vectors

4. MODULES

The modules available in MAPPS are divided into utility and process modules. Utility modules perform a variety of useful functions, which includes converging flowsheets, printing module and stream vectors, and performing economic analyses. Process modules are used to simulate processes and pieces of equipment in the mill. This section briefly describes the available modules. A description of each module is provided in Tables 4.1-4.6.

4.1 UTILITY MODULES

The convergence control module, CONVRG, instructs the executive program to repeatedly execute modules in a user-specified order until the values of the elements in a designated stream have converged. The use of this module in a flowsheet is described in Section 5 of this manual.

Four modules are available for printing results. PRNTMV prints the contents of selected module vectors, and PRNTSV prints selected stream vectors. The user designates the modules and streams to be printed by entering their numbers in the vectors for these modules. When the module and stream vectors are printed, the individual vector elements are labeled, and their values are printed.

PRNTMC and PRNTSC are used to compare the results of two or more simulations executed as multiple runs. The usage of these modules is similar to PRNTMV and PRNTSV, but instead of printing the values of vector elements, these modules print the fractional difference between vector elements of successive runs.

Several other utility modules are available to the user. ECONOM conducts a cash flow analysis, based on operating revenues and costs, and capital costs.

The plant life, tax rate, and interest rate must be specified. This module also prints the results of the cash flow analysis.

SUMARY is a module that can be used to summarize module and stream parameters from the complete flowsheet. This module allows selective printing of these parameters.

SWITCH can be used to switch independent and dependent variables. The user normally supplies MAPPS with the values of independent variables, and MAPPS computes the values of the dependent variables. With this module, the user can supply MAPPS with the values of one to ten dependent variables. MAPPS will then compute corresponding values of selected independent variables.

4.2 PROCESS MODULES

Process modules are divided into five categories - general purpose, pulp mill, bleach plant, paper mill, and steam and power system modules. A description of the modules in each category is presented in Tables 4.2-4.6.

General purpose modules are used to simulate processes occurring in several subsystems in an integrated mill. These modules include a splitter, mixer, pump, valve, heat exchangers, separators, and flow controllers.

Several modules can be categorized in either of two different subsystems. For example, the recovery steam generator (often referred to as the recovery boiler) can be considered part of the pulp mill or steam and power system. This generator is divided into a recovery furnace module and a boiler module, and it is considered part of the steam and power system.

Table 4.1 Utility Modules

Label (Type)	Description
CONVRG (1)	Controls the repeated execution of modules in a calculation loop until the component values of a test stream have converged. The test stream, convergence tolerance, and maximum number of iterations are specified.
CONVRT (9)	Converts one stream type to another. The correspondence between the constituents of the two stream types is specified.
ECONOM (6)	Conducts a cash flow analysis, based on operating revenues and costs, and capital costs. The plant life, tax rate, and interest rate are specified. A summary of the operating revenues and costs is printed.
PRNTMC (4)	Prints module vector comparisons for two consecutive simulations. The modules to be compared are specified.
PRNTMV (2)	Prints module vectors with labels. The modules to be printed are specified.
PRNTSC (5)	Prints stream vector comparisons for two consecutive simulations. The streams to be compared are specified.
PRNTSV (3)	Prints stream vectors with labels. The streams to be printed are specified.
SUMARY (7)	Summarizes selected module and stream parameters. The parameters of interest, with corresponding stream or module numbers, are specified.
SWITCH (8)	Switches independent and dependent variables. The user estimates the independent variables and specifies the dependent variables.

Table 4.2 General Purpose Process Modules

Label (Type)	Description
COMP01 (33)	Simulates a multiple-stage compressor. The outlet pressure and compressor efficiency are specified.
CONSIS (21)	Simulates a consistency controller that maintains a target ratio of solids flow to total flow by dilution. The components defined by 'solids' are user-identified.
CONSPL (22)	Simulates a controller-splitter that creates an outlet stream by taking material from a supply stream. The target amount of material in the outlet stream may be user-specified or determined from the difference between two control streams.
FLSH01 (34)	Simulates an adiabatic flash. A boiling point correction is made, based on the dissolved solids concentration. The flash pressure is specified.
HEXCH1 (32)	Simulates a simple, indirect fluid-to-fluid heat exchanger without phase change. The outlet temperature of the cold fluid is specified. The heat transfer rate and heat exchanger effectiveness are calculated.
HEXCH2 (23)	Simulates an indirect, fluid-to-fluid heat exchanger without phase change. An effectiveness model is used to calculate the outlet temperatures of the fluid streams. Either the effectiveness or the product of the overall heat transfer coefficient and the area is specified.
HEXCH3 (24)	Simulates a heat exchanger where a water, gaseous, pulping, recovery, bleaching, paper, or waste stream is heated using condensing steam. Either the effectiveness or the target outlet temperature is specified.
MIXR01 (25)	Simulates the adiabatic mixing of two to nine streams of compatible type. The minimum nonzero inlet stream pressure is assigned to the outlet stream.
PCONT1 (26)	Simulates a proportional flow controller. The adjusted (i.e., updated) flow rate is equal to the old flow rate plus the product of a user-specified gain and the difference between the value of a designated parameter (e.g., temperature) and a target value.
PUMP01 (27)	Simulates a motor-driven pump that pumps a water, pulping, recovery, bleaching, paper, or waste stream. The pressure rise and motor and pump efficiencies are specified.
PVALVE (28)	Simulates a pressure-reducing valve that throttles a water, gaseous, pulping, recovery, bleaching, paper, or waste stream. The pressure drop across the valve is specified.

Table 4.2 General Purpose Process Modules (Continued)

SEPAR1 (29)	Simulates a simple separator where the separation efficiency (reject flow/total flow) of each inlet stream constituent is specified. If the inlet stream is wet steam, it simulates a steam-water separator by setting the separation efficiency equal to one minus the steam quality.
SEPAR2 (30)	Simulates a separator, where the separation efficiency for each constituent is equal to one minus the ratio of the normalized flow rate in the accept stream to the normalized flow rate in the feed stream. Each constituent flow rate is normalized by dividing its flow rate by the total flow rate.
SPLIT1 (31)	Simulates a simple splitter that divides a single inlet stream into two to nine outlet streams. The split ratio of all but one of the outlet streams is specified.

Table 4.3 Pulp Mill Process Modules

Label (Type)	Description
BLOX01 (50)	Simulates black liquor oxidation. The excess air and oxidation efficiency are specified.
CAUS01 (41)	Simulates the slaking and causticizing steps of chemical recovery. The excess lime, causticizing efficiency, and white liquor temperature are specified.
CLAR01 (42)	Simulates a liquor clarifier. The consistency of the outlet solids stream is specified.
DCEV01 (43)	Simulates a direct-contact evaporator. The temperature difference between the outlet flue gas and black liquor streams is specified.
DIGR01 (44)	Simulates a set of batch digesters or the impregnation and cooking zones of a continuous digester. Steam may be supplied directly or indirectly at one or two pressures. The digester volume and cooking temperature are specified. The pulping model, for 11 wood species, was developed by J. V. Hatton. Mixtures of two of these species may be specified.
DIGR02 (56)	Simulates a set of batch digesters or the impregnation and cooking zones of a continuous digester. Steam may be supplied directly or indirectly at one or two pressures. The digester volume and cooking temperature are specified. The pulping model was developed by T. J. McDonough at IPC.
EVAP01 (45)	Simulates a multiple-effect evaporator, which can include a concentrator. The number of effects, mean heat transfer coefficient, and mean heat transfer area per effect are specified. The strong liquor concentration and the steam consumption are calculated.
KILN01 (47)	Simulates a lime kiln. The lime conversion efficiency and availability, and the outlet temperatures of the lime and flue gas are specified. The fuel consumption is calculated.
MKUP01 (48)	Simulates a water and chemical makeup control strategy. The alkali charge on wood, white liquor sulfidity, and white liquor active alkali concentration are specified. Makeup water and chemicals are adjusted to meet these specifications.
SMLT01 (51)	Simulates the smelt dissolving and clarifying processes.
WASH01 (49)	Simulates a single-stage washer. It analyzes the washing process in terms of dilution and thickening. Displacement is not considered.
WASH02 (52)	Simulates a set of countercurrent vacuum drum washers. The model is taken from Perkins, Welch and Mappus (*Tappi*, Oct., 1954). It analyzes the washing process in terms of dilution, thickening, and displacement.

Table 4.3 Pulp Mill Process Modules (Continued)

WASH03 (53)	Simulates a countercurrent diffusion washer. The model is taken from Williams, McKibbens, and Riese (*Tappi*, Sept., 1965). It analyzes the washing process in terms of one-dimensional diffusion of solute from cooked chips.
WOOD01 (46)	Generates a wood stream using the pulping stream format. The wood temperature, total flow rate, and constituent mass fractions (water, cellulose, lignin, tall oil, and turpentine) are specified.

Table 4.4 Bleach Plant Process Modules (Future Development)

Label (Type)	Description
CLORO1 (71)	Simulates a chlorination stage in a bleaching sequence.
EXTRO1 (72)	Simulates a caustic extraction stage in a bleaching sequence.
DIOXO1 (73)	Simulates a chlorine dioxide stage in a bleaching sequence.
HYPOO1 (74)	Simulates a hypochlorite stage in a bleaching sequence.
OXYGO1 (75)	Simulates an oxygen stage in a bleaching sequence.

Table 4.5 Paper Mill Process Modules

Label (Type)	Description
AIRCNT (91)	Simulates the air damper controller of a dryer hood. The air is split into two streams. The split ratio is adjusted to maintain a target air temperature in a specified stream.
DRYER1 (92)	Simulates a conventional drum paper-dryer system including the hood, web, blowers, and steam drums. It can be used to simulate a whole dryer or a single dryer section.
REFNR1 (93)	Simulates a stock refiner. The power required and the stock temperature rise are calculated.
STCNT1 (94)	Simulates a steam controller for a dryer system. The overall steam economy of the dryer system is specified, and the steam flow rate required to maintain the outlet moisture of the paper at a target value is calculated.

Table 4.6 Steam and Power System Process Modules

Label (Type)	Description
BOIL01 (111)	Simulates the boiler section of a steam generator that generates steam from feedwater using flue gas. A simplified model is used neglecting sootblowing and blowdown steam. The outlet flue gas temperature is specified.
BOIL02 (112)	Simulates the boiler section of a steam generator that generates steam from feedwater using flue gas. The outlet flue gas temperature is specified. The boiler efficiency is calculated.
CONDEN (113)	Simulates a water-cooled condenser that condenses superheated steam to saturated liquid.
DEAER1 (114)	Simulates a deaerator. Clean reusable process water, makeup water, and steam are mixed to produce boiler feedwater. The steam needed to bring the feedwater to the saturated liquid state is calculated.
DESUP (115)	Simulates a desuperheater used to desuperheat steam to the specified degrees of superheat. The required amount of water is calculated.
FUEL01 (116)	Simulates a fuel controller. The outlet fuel flow rate is adjusted to meet a target value of a steam or power stream.
FUEL02 (117)	Simulates a fuel controller that controls the flow rate of two fuels. The fuel flows are adjusted to generate a target steam flow rate in a user-designated stream.
FURN01 (118)	Simulates the combustion section of a recovery furnace. Black liquor may enter as a pulping stream or a fuel stream. The excess air, sulfate reduction ratio, and smelt temperature are specified. The products of combustion and the heat released are calculated.
FURN02 (119)	Simulates the combustion section of a power furnace, supplied with two different fuels. The excess air and ash temperature are specified. The products of combustion and the heat released are calculated.
GNRATR (120)	Simulates an electrical generator that generates electrical power from mechanical power. The generator efficiency is specified.
STCNT2 (121)	Simulates a steam controller that splits inlet steam into two process steam streams. The flow rate of steam in each process stream is adjusted to satisfy process demands.

Table 4.6 Steam and Power System Process Modules (Continued)

STLOAD (122)	Simulates a process steam load, i.e., a demand for process steam. The required process steam is divided into steam consumed by the process and recoverable condensate.
TURBO1 (123)	Simulates a steam turbine. The turbine has inlet and outlet steam lines, two extraction ports, and an output power shaft. The efficiency of each turbine section, the extraction steam flow rates, and the extraction and outlet pressures are specified. The shaft power is calculated.
TURBO2 (124)	Simulates a simple steam turbine with inlet and outlet steam lines, and an output power shaft. The shaft power is calculated using a specified turbine efficiency and outlet steam pressure.

5. THE SOLUTION METHODOLOGY

In sequential modular simulation, the modules are executed one at a time in a specified order. In the example simulation of blowdown followed by brown stock washing, presented in Section 2, stream 1 (pulp and black liquor) is specified by the user, and module 1 (blow tank) calculates stream 2 (steam) and stream 3 (pulp and black liquor). Stream 4 (wash water), which is specified by the user, and stream 3 are given to module 2 (washer). This module then calculates stream 5 (weak black liquor) and stream 6 (pulp). Note that the connection of modules 1 and 2 through stream 3 implies that module 1 should be calculated before module 2.

A complication frequently arises when stream loops are present in a flowsheet. For example, consider the two-stage countercurrent washer configuration of Fig. 5.1. In this case, stream 3 (pulp and black liquor) is calculated by module 1 (blow tank). Stream 4 is no longer user specified but is the filtrate stream (i.e., weak black liquor) calculated by module 3, the second-stage washer. A loop appears because stream 4 is being recycled. Stream 6 cannot be accurately calculated by module 2 unless stream 4 is known, and stream 4 cannot be calculated by module 3 unless stream 6 is known.

This impasse is broken when the user specifies an initial estimate for one of the stream vectors in the loop, e.g., stream 4. Module 2 now can calculate stream 6, and module 3 can calculate an improved value of stream 4 (assuming that the user had specified stream 7). This execution sequence (module 2 followed by module 3) can be repeated indefinitely, with each iteration providing an improved value for stream 4. The overall module calculation order can therefore be written as (1, 2, 3, 2, 3, 2, 3, ...). It is difficult to estimate the length of this sequence because the number of repetitions of the group (2, 3) required to achieve a

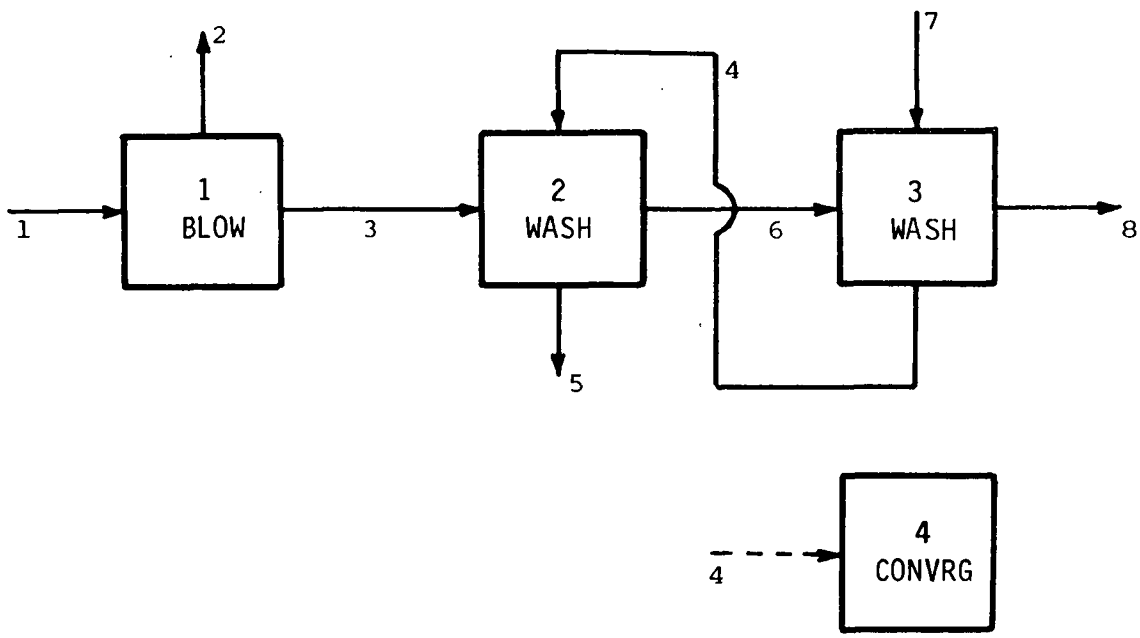


Figure 5.1 Process Flowsheet of Blowdown Followed by Two-Stage Countercurrent Washing

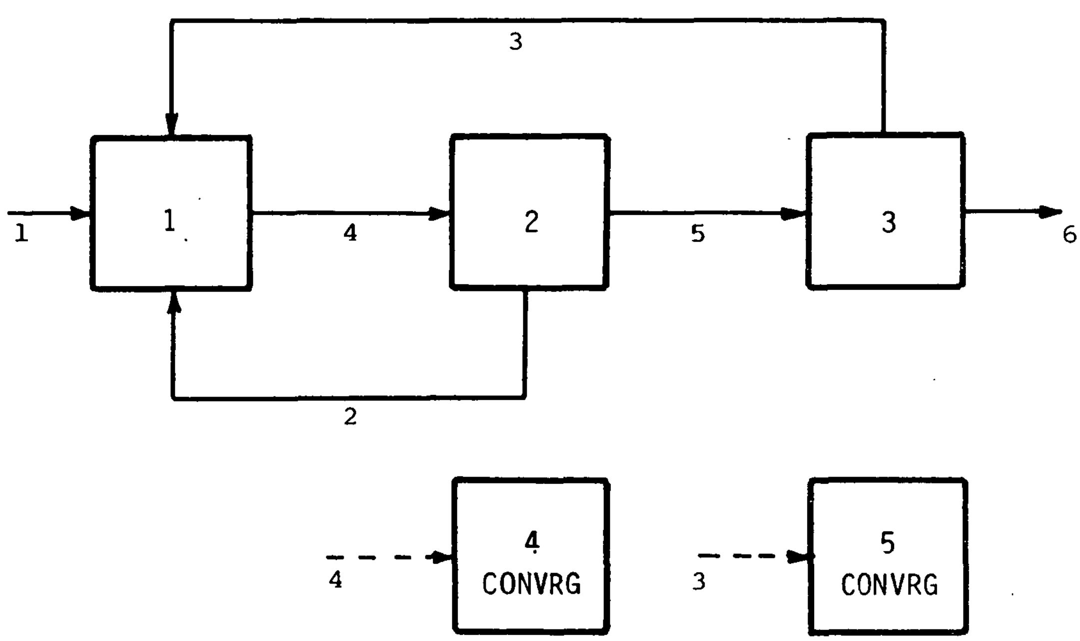


Figure 5.2 Process Flowsheet with Nested Loops

desired precision is not known a priori. To overcome this difficulty, a special utility module for converging loops (module 4, CONVRG) is added to the flowsheet. With this module, the calculation order can be written more compactly as (1, 2, 3, 4). When module 4 is executed, it tests component values of stream 4 for convergence. If these values have not converged within a specified tolerance, module 4 directs program control back to module 2, which is the beginning of the calculation loop. Note that the component values of the test stream (stream 4) must be supplied to the convergence module (module 4), as indicated by the dashed stream vector.

The situation can become more complicated when nested loops are present, as illustrated in Fig. 5.2. In this example, stream 1 is user specified, and the remaining streams are to be determined. Modules 1, 2, and 3 are process modules, and modules 4 and 5 are utility CONVRG modules. Three convergence strategies will be described. In the first strategy, streams 2 and 3 are estimated, and the calculation order is (1, 2, 3, 4). In the second, stream 4 is estimated, and the calculation order is (2, 3, 1, 4). This order requires estimating one stream instead of two, as before. In the third strategy, streams 2 and 3 are estimated, but the inner loop is converged on every pass through the outer loop. The calculation order is ((1, 2, 4), 3, 5), with the inner and outer loops indicated by inner and outer parentheses, respectively. Note that a CONVRG module is placed at the end of each calculation loop. For this simple example, a sophisticated convergence strategy would be unnecessary, but it is recommended in cases where the inner loop is much smaller than the outer loop or in cases where convergence difficulties are encountered.

A stream that must be used to execute a module, but is calculated by another module later in the calculation order, is referred to as a "tear" stream. The flowsheet topology and the calculation order determine the number of tear streams. These streams will often correspond to process recycle streams.

The following rules may provide guidance in the selection and use of tear streams.

1) Recycle streams or countercurrent streams are good candidates for tear streams.
2) The module calculation order should be chosen so that the number of tear streams is not greater than the number of loops in the flowsheet.
3) If a small loop is nested inside a larger one, converging the inner loop on each iteration through the outer loop will reduce execution time.
4) If convergence difficulties are encountered, an alternative choice of tear streams or a different calculation order may be warranted.
5) If the user chooses not to initialize a tear stream, it will be assigned a default value of zero. A good estimate of the initial value will, however, result in faster convergence.

6. DATA ASSEMBLY

Before executing a MAPPS simulation, the process flowsheet must be described, stream and module constants defined, and decisions made concerning type and quantity of output information. The procedure for assembling this information is presented below. A simple example problem, illustrating the procedure, follows.

6.1 PROCEDURE FOR ASSEMBLING THE DATA

The following seven steps outline the procedure for assembling the required data.

STEP 1 - Draw a MAPPS Flowsheet. First, select the appropriate MAPPS process modules to "simulate" the real system. Connect the modules with streams wherever necessary. The required stream connections are described under the documentation for each module. Next, assign a unique number, from 1 to 200, to each module. Last, assign a unique number, between 1 and 300, to each stream.

STEP 2 - Determine the Module Calculation Order and Add CONVRG Modules. The solution to a MAPPS problem is achieved through successive substitution. Unit modules are executed sequentially until all modules have been executed at least once. This constitutes one iteration. Iterations are then performed until constituent values of a user-designated stream converge within a designated tolerance, or until the specified maximum number of iterations is reached.

The sequence for executing unit modules is determined by the user. To reduce computer execution time, the most efficient sequence should be defined. The sequence may be determined by examining the connecting streams between modules. Begin by executing modules at the beginning of the flowsheet that have process input streams associated with them. From then on, when two modules are connected by a stream, always begin by executing the module that has the connecting stream as an output stream. When recycle or countercurrent streams are present, follow this rule as closely as possible.

The convergence of inner loops may be accelerated by the addition of CONVRG modules. The convergence of the outermost loop is also achieved with a CONVRG module.

STEP 3 - Add Utility Modules. Utility modules perform a variety of useful functions. They generate stream and module data, perform economic analyses, and fulfill several other functions. The user must select the desired modules and add them to the flowsheet and calculation order list. Generally, the print modules PRNTSV and PRNTMV are added at the end of the list. However, they may be embedded in an inner loop for diagnostic purposes.

STEP 4 - Specify Module Vectors. The parameter values of all of the module vectors must be specified. Refer to the module documentation for complete instructions.

STEP 5 - Specify Stream Vectors. All process input streams must be identified along with any tear streams the user wishes to initialize. Values for some of the elements in these vectors must then be specified. The remaining element values are calculated by the simulator. Each type of stream needs a slightly different set of vector elements specified by the user. Table 6.1 defines the critical elements for each type of stream.

STEP 6 - Specify Program Options. The user has six program options that can be used when running a simulation. These options are described in Table 6.2.

Table 6.1 Required Elements to be Specified for Each Type of Stream

Element No.	Stream Type 1	1	1	2-7	8	9	10
1	x	x	x	x	x	x	x
2	x	x	x	x	x	x	x
3	x		x	x	x		
4		x	x	x	x		
5					x		
6	x	x			x	x	
7						x	
8						x	
9							
10							
11-m	x	x	x	x	x		

NOTES:

- x corresponds to "required" input data.
- m denotes the maximum number of constituents.

- For stream types 2-8, pressure (element 4) is not used to calculate the other elements. However, for completeness of the stream definition, it should be specified.

- For a water stream (type 1), if pressure and temperature are specified and the water is in a saturation state (i.e., saturated liquid and/or vapor in equilibrium), the water is assumed to be saturated vapor.

Option 1 is used to select English Engineering (value = 0) or SI (value = 1) units. All of the other options are actuated by setting their value (i.e., flag) to one (1).

Options 2-6 generate output data that are stored in computer disk "files." The names of the generated files are listed in Table 6.2. Option 2 causes an output file to be generated that contains the output data requested by the print modules. Option 3 generates a specially formatted file of the original (i.e., prerun) data file. Using Option 4, a specially formatted postrun file of the data file is generated. This file contains the converged values of the components in the streams specified in the original data file. In addition, it also contains converged values for tear streams not included in the original file.

Options 5 and 6 may be used for diagnostic purposes. Option 5 generates a report containing a detailed analysis of the flowsheet topology. Streams are categorized as process inlet or outlet, connecting, or tear streams. Duplicate stream and module numbers are identified.

Option 6 causes mass and energy balances to be analyzed around each module after the last iteration. It then prints these balances for the ten modules having the least accurate mass balance and the ten with the least accurate energy balance.

MAPPS can be run in either an interactive or batch mode. In an interactive run, the user is asked to select the unit system (Option 1). The four output files are automatically generated. In a batch mode, the user must specify the program options to be used during execution.

6.2 DATA ASSEMBLY FOR AN EXAMPLE PROBLEM

The following simple example problem will be used to illustrate the seven-step procedure for assembling data for a MAPPS run. Consider the screening system, comprising a mixing tank and screen, shown in Fig. 6.1. The problem is to determine the steady-state constituent flow rates in the accepts and rejects stream for the following operating condition.

Inlet flows:	Screen splits (rejects/feed):
water = 8000 lb_m/hr	water = 0.5
fiber = 400 lb_m/hr	fiber = 0.2
fines = 40 lb_m/hr	fines = 0.6

The temperature and pressure of all of the streams are assumed to remain constant at 80 F, 14.7 psia.

STEP 1 - Draw a MAPPS Flowsheet. The process flowsheet is presented in Fig. 6.2. The process modules are MIXR01 and SEPAR1. One possible set of module and stream numbers is illustrated.

STEP 2 - Determine the Module Calculation Order and Add CONVRG Modules. As shown in Fig. 6.3, stream 2 is selected as a tear stream, and one CONVRG module is added to the process flowsheet. The solution order is (1, 2, 3).

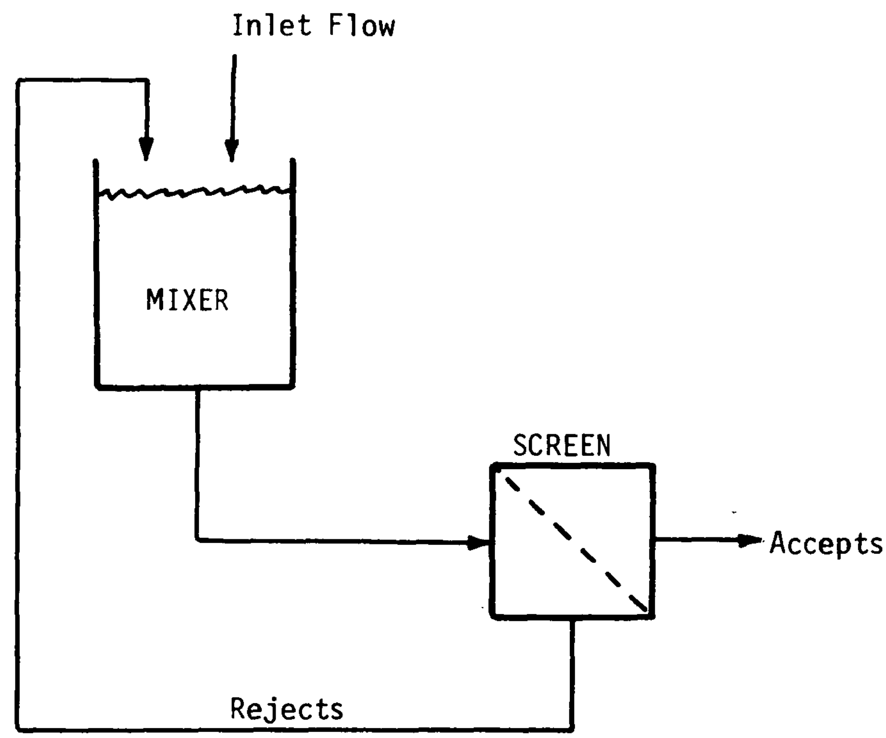


Figure 6.1 Process Flow Diagram of Example Problem

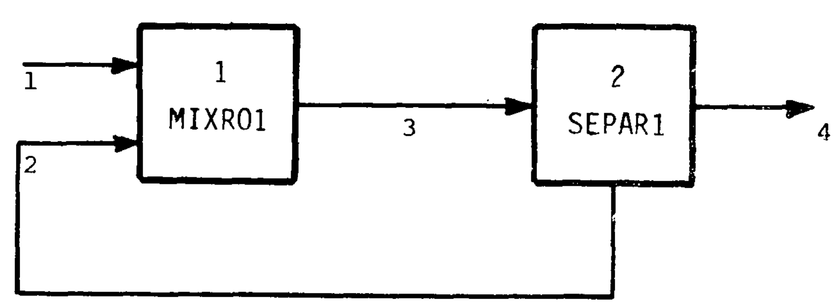


Figure 6.2 MAPPS Flowsheet of Example Problem

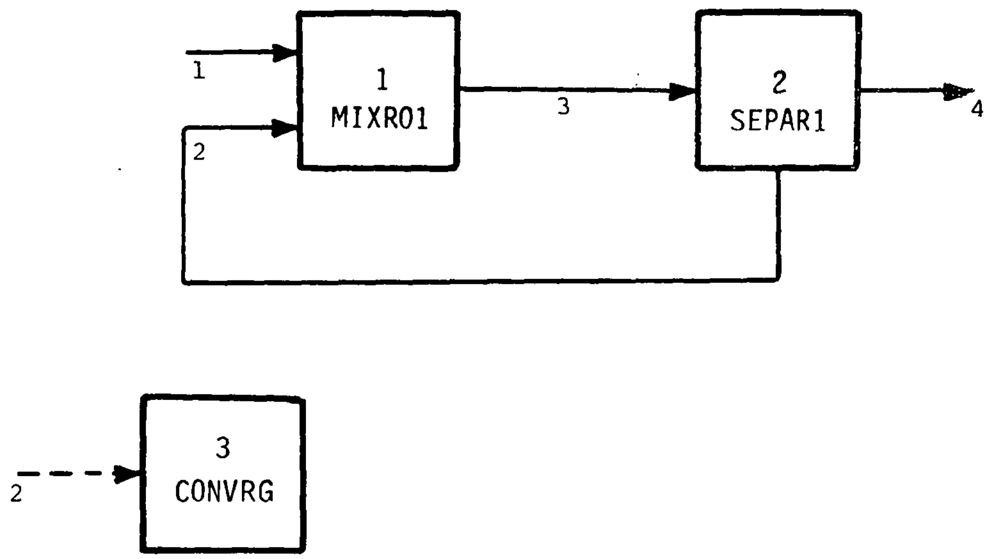


Figure 6.3 MAPPS Flowsheet with CONVRG Added

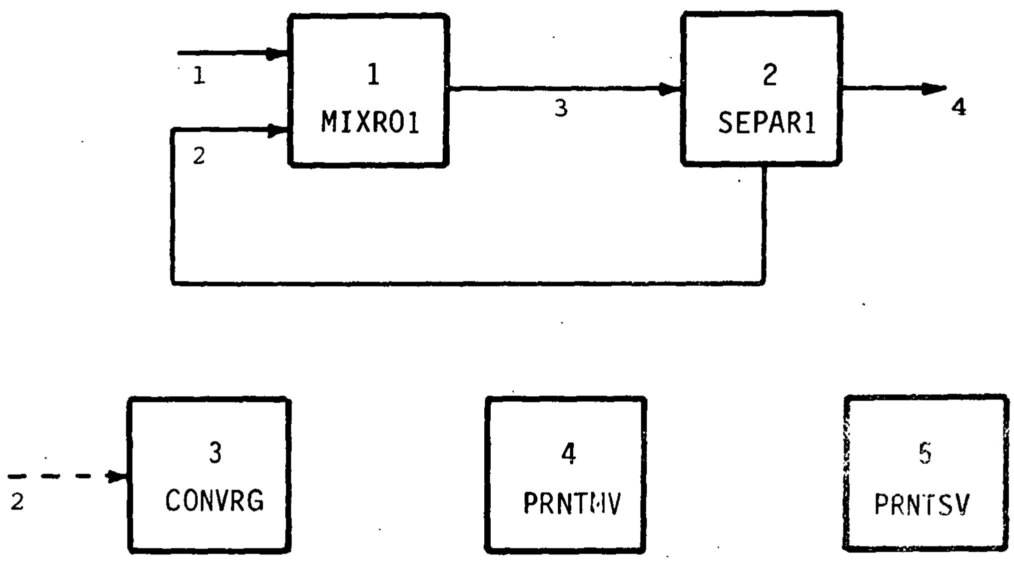


Figure 6.4 Complete MAPPS Flowsheet of Example Problem

STEP 3 - Add Utility Modules. The print modules PRNTMV and PRNTSV are added to the flowsheet. The final MAPPS flowsheet, including all of the modules and streams, is presented in Fig. 6.4. The final calculation order is ((1, 2, 3), 4, 5).

STEP 4 - Specify Module Vectors. The parameter values for the module vectors are presented below. Documentation for the five modules is presented in the Appendix, Section 9.1.

Module Vector for MIXR01

Parameter	Value	Comment
1	1	Module number (from Fig. 6.4)
2	25	Module type
3	16	Number of parameters
4	2	Number of given streams
5	1	Number of calculated streams
6	1	Stream number of first inlet stream[a]
7	2	Stream number of second inlet stream
8	3	Stream number of outlet stream

Module Vector for SEPAR1

Parameter	Value	Comment
1	2	Module number (from Fig. 6.4)
2	29	Module type
3	18	Number of parameters = 15 + 3
4	1	Number of given streams
5	2	Number of calculated streams
6	3	Stream number of feed stream
7	2	Stream number of reject stream
8	4	Stream number of accept stream
9-15		(Not used)
16	0.5	First constituent separation efficiency
17	0.2	Second constituent separation efficiency
18	0.6	Third constituent separation efficiency

[a]The stream type for the first stream into MIXR01 must be initialized.

Module Vector for CONVRG

Parameter	Value	Comment
1	3	Module number (from Fig. 6.4)
2	1	Module type
3	60	Number of parameters
4	1	Number of given streams
5	0	Number of calculated streams
6	2	Stream number of test stream
7-15		(Not used)
16	1	Number of first module in loop
17	0.001	Fractional convergence tolerance
18	0	Absolute convergence tolerance
19	30	Maximum number of iterations

Module Vector for PRNTMV

Parameter	Value	Comment
1	4	Module number (from Fig. 6.4)
2	2	Module type
3	18	Number of parameters = 15 + 3
4	0	Number of given streams
5	0	Number of calculated streams
6	0	Print control parameter
7-15		(Not used)
16	1	Module number of first module
17	2	Module number of second module
18	3	Module number of third module

Module Vector for PRNTSV

Parameter	Value	Comment
1	5	Module number (from Fig. 6.4)
2	3	Module type
3	19	Number of parameters = 15 + 4
4	0	Number of given streams
5	0	Number of calculated streams
6	0	Print control parameter
7	1	Format control parameter
8-15		(Not used)
16	1	Stream number of first stream
17	2	Stream number of second stream
18	3	Stream number of third stream
19	4	Stream number of fourth stream

STEP 5 - Specify Stream Vectors. For this example problem, stream 1, the primary inlet stream, requires initialization. Stream 2, the tear stream, will also be initialized. The values for both streams are listed below.

Element	Stream 1 Value	Stream 2 Value	Comment
1	1	2	Stream number (from Fig. 6.4)
2	6	6	Stream type (6 for paper)
3	80	80	Temperature
4	14.7	14.7	Pressure
5			Will be initialized by MAPPS
6-7			(Not used)
8-10			Will be initialized by MAPPS
11	8000	400	Water mass flow rate
12	400	80	Fiber mass flow rate
13	40	32	Fines mass flow rate

Note that the component values in stream 2 are only "estimates."

STEP 6 - Specify Program Options. Since English Engineering units are used, value = 0 for Option 1. To cause modules PRNTMV and PRNTSV to generate output data, value = 1 for Option 2. Options 3-6 will also be set equal to one to generate the remaining output files.

7. RUNNING MAPPS IN A BATCH MODE

In this section, the methodology of constructing a data file for a batch run is presented. These concepts are illustrated by using the example screening problem discussed in Section 6. For this problem, the input data file and the resulting output data files are presented.

7.1 DATA FILE CONSTRUCTION

When running MAPPS in a batch mode, the user must construct the input data file. This file contains all the necessary information for running a simulation. The information must be written in a specific format. It includes the flowsheet configuration, module parameter values, stream component values, module calculation order, and miscellaneous information. After the data are assembled, each type or group of data is preceded by a "data label." The data are written as an ordered string of numbers or characters on the following line(s). The individual sets of data, with their corresponding labels, are then assembled to form a complete data file.

7.1.1 Data Labels

Six different data labels are used to distinguish between groups or types of data. Each label must begin in column one. Only the first character is necessary. Other characters on the line are treated as comments. The labels include: TITLE, PROGRAM OPTIONS, CALCULATION ORDER, MODULE, STREAM, REMARK, and END.

7.1.2 Data Lines

The data lines follow the data labels. They are strings of alphanumeric characters, blanks, and commas, positioned in columns 2 through 80. With the exception of the TITLE, END, and REMARK labels, all other labels may be followed by as many data lines as necessary. TITLE may be followed by only one data line, and END and REMARK are not followed by data lines. Strings of numerical data may be formatted in a sequential or indexed manner, or in a combination of these two ways. Individual numbers may be in integer, decimal, or exponential form. They must be separated by at least one blank or comma.

a. Sequential String Format

Data are entered sequentially, beginning with the first vector element and ending with the last nonzero element or the last element.

b. Indexed String Format

Data are entered as a pair of numbers separated by an equal sign. The first number of a pair is the position in the vector, and the second number is the value of the parameter. Zero values need not be assigned.

c. Mixed String Format

Any combination of the above two formats may be used. When a nonindexed data element follows an indexed data element, the last index is incremented by 1 and the number stored in the corresponding vector position.

The following examples illustrate these formats. In each example, the data vector is initialized with the same set of values.

Sequential

```
a)  5    3,0    0    240000.    0,    32000    0    0    19.
b)  5,   3      0,   0.  240000      0.0   32E3,    0.,
     0.0       19,   0.0,0.0, 0, 0
```

Indexed

```
a) 1 = 5,      2 = 3.,     3 = 0     4 = 0.     5 = 2.4E4
   6 = 0,      7 = 32000.,8 = 0      9 = 0     10 = 19.
b) 7 = 3.2E4   2 = 3       8 = 0.    1 = 5      5 = 32000.
  10 = 19
```

Mixed

a) 1 = 5, 3. 0,   4 = 0   5 = 240000.   6 = 0
   32000,   8 = 0   0,19

b) 5 = 2.4E4, 0, 32000.   1 = 5   8 = 0.0,   10 = 19
   2 = 3.0

7.1.3 Overall File Format

The TITLE label always indicates the beginning of a data file. The END label always indicates the end of a data file. The PROGRAM OPTIONS label and then the CALCULATION ORDER label should follow the TITLE label. The MODULE and STREAM labels can be in any order between the CALCULATION ORDER label and the END label. REMARK labels can be placed anywhere in the data file. This label is intended for making comments. The recommended file format is presented in Table 7.1.

When listing the module numbers following the CALCULATION ORDER label, parentheses can be inserted to indicate calculation loops. The inclusion of parentheses will then indicate the solution algorithm. Several examples of the use of parentheses are presented in Section 5.

Table 7.1 Recommended File Format

TITLE

<Alphanumeric information in columns 2-80>

PROGRAM OPTIONS

<Values for flags in columns 2-80>

CALCULATION ORDER

<Module numbers in the order in which they are to be executed in columns 2-80>

MODULE <with alphanumeric information through column 80>

<module parameters in columns 2-80>

Note: Each module requires its own label.

STREAM <with alphanumeric information through column 80>

<stream parameters in columns 2-80>

Note: Each stream requires its own label.

REMARK <with alphanumeric information through column 80>[a]

END

---

[a]REMARK label can be placed anywhere in the data file.

7.2 RESULTS FOR EXAMPLE SCREENING PROBLEM

The input data file for the example screening problem is presented below.

```
TITLE
 MAPPS EXAMPLE PROBLEM
PROGRAM OPTIONS
   0   1   1   1   1   1   0   0   0   0
CALCULATION ORDER
   1   2   3   4   5
MODULE   1 - MIXR01
    1             25            8             2             1
    1             2             3
MODULE   2 - SEPAR1
    2             29            35            1             2
    3             2             4             0             0
    0             0             0             0             0
    0.500000      0.200000      0.600000
MODULE   3 - CONVRG
    3             1             60            1             0
    2             0             0             0             0
    0             0             0             0             0
    1             0.00100000    0             30
MODULE   4 - PRNTMV
    4             2             18            0             0
    0             0             0             0             0
    0             0             0             0             0
    1             2             3
MODULE   5 - PRNTSV
    5             3             19            0             0
    0             1             0             0             0
    0             0             0             0             0
    1             2             3             4
STREAM   1 - INLET
    1             6             80            14.7000       0
    0             0             0             0             0
    8000          400           40
STREAM   2 - TEAR
    2             6             80            14.7000       0
    0             0             0             0             0
    400           80            32
END
```

The PRNTMV and PRNTSV modules produce the following output.

```
    MAPPS EXAMPLE PROBLEM

     1 MIXR01

     2 SEPAR1
    16  EFF1     0.50000      17  EFF2     0.20000      18  EFF3     0.60000

     3 CONVRG
    16  AMOD      1.0000      17  TOLRF    1.00000E-03 18  TOLRA    0.
    19  AMAX      30.000      20 *COUNT    0.           21 *ALOOP    12.000
    22 *STREAM    2.0000

     1 NUMBER     1.0000       2.0000      3.0000       4.0000
     2 TYPE       6.0000       6.0000      6.0000       6.0000
     3 TEMP       80.000       80.000      80.000       80.000
     4 PRESS      14.700       14.700      14.700       14.700
     5 HEAT CAP 0.96481      0.98677     0.97560      0.96481
     6          0.           0.          0.           0.
     7          0.           0.          0.           0.
     8 ENTHALPY  24429.       24153.      48579.       24426.
     9 EXERGY     67.984       67.217      135.19       67.977
    10 FLOW       8440.0       8159.0      16598.       8439.0
    11 WATER(L)   8000.0       7999.1      15998.       7999.1
    12 FIBER      400.00      100.000      500.00       400.00
    13 FINES      40.000       59.963      99.939       39.976
    14 DIS SLDS 0.           0.          0.           0.
    15 ADDITIVE 0.           0.          0.           0.
    16 DIRT     0.           0.          0.           0.
    ** DATE  121382  **  TIME 16:44:26  **        CPU TIME (SEC)    0.8 **

    ** DATE  121382  **  TIME 16:45: 4  **  TOTAL CPU TIME (SEC)    7.1 **
```

8. RUNNING MAPPS IN AN INTERACTIVE MODE

Running MAPPS on the IPC Burroughs 6900 System requires a basic command of the Burroughs' editor, CANDE (Command AND Edit). An interactive version of MAPPS has been prepared, which permits a user with a minimum knowledge of CANDE to use MAPPS. This version of MAPPS allows a user to interactively construct a data file, run the simulator, and store the generated output files under titles of his choice. The user is prompted extensively. Many errors in the data are identified, and the user is given the opportunity to make corrections.

This version of MAPPS performs several functions. It may be used solely for data file construction. When used for this purpose, the data are stored in a file for later use. The interactive version may also be used to run a simulation, using an already existing data file. The user may also run multiple simulations, examining or changing stream or unit parameters between runs.

When the interactive version is used to construct a data file, two output files will be generated, MAPPS/OUTPUT/DIAGNOSIS and MAPPS/OUTPUT/PRERUNDATA. When this version is used to execute a simulation, all the output files will be generated, with the exception of the error messages, which will be written to the terminal. When performing multiple runs, files will be generated in the same manner as a single simulation. The information in the DIAGNOSIS and PRERUNDATA files will apply to the data used in the first run, while the information in the RESULTS and POSTRUNDATA files will apply to data used in the last run.

A typical user session for the example problem follows. A minimum set of CANDE commands, necessary for the session, is given in Table 8.1. Control commands, which may be necessary for a session, are given in Table 8.2. Each of these commands begins with a question mark. The 'DS' command may be used to terminate a program at any time. The 'JA' command will list the numbers of the actively running jobs, started from the user's terminal. The 'TI' command will give the amount of CPU, I/O, and elapsed time for a currently running job. Basic terms, used in the CANDE and control commands, are defined in Table 8.3. More complete definitions of these and other CANDE commands may be found in *The Complete CANDE Primer* by Donald J. Gregory.

In the following example session, CANDE commands issued by the user are doubly underscored, and user responses to the simulator are singly underscored. System responses usually begin with a '#' and simulator responses begin with '**'.

Table 8.1 CANDE Commands[a]

HELLO	< usercode > < password>	Initiates a new user session.
START	< filename >	Initiates a job entitled < filename >.
FILES		Lists all the names and types of files in the user's library.
FILES	< directory name >	Lists all the names and types of files in the user's library, under the designated directory name.
LIST	< filename >	Displays contents of a specific file.
WRITE	< filename >	Prints file on the IPC line printer *after the session is terminated.*
WRITE	< directory name >/=	Same as above, except that all files under a directory name are printed.
REMOVE	< filename >	Removes a file from the user's library.
REMOVE	< directory name>/=	Same as above, except that all the files under a directory name are removed.
BYE		Terminates a user session.

[a]The complete command may be typed, but only the underscored portion is necessary.

Table 8.2 Control Commands

? < mix number > DS Terminates a job.

? JA Displays job structure and mix number of jobs STARTed from station.

? < mix number > TI Displays system use times associated with a currently running job. (i.e., CPU, I/O, and elapsed time.)

Table 8.3 Definition of Terms

< usercode > < password> - To access the host computer, each user must have a valid usercode and a password. For security purposes, one's password may be changed, but the usercode will always remain the same.

Library - a collection of files stored under and accessible through the usercode.

File - a collection of information stored on disk.

< filename > - the title of a file. It is composed of alphanumeric characters and slashes "/" (optional); e.g., MAPPS/OUTPUT/RESULTS.

< directory name > - the leading components of a filename, which are separated by slashes. Filenames are constructed in this manner to group related files and to facilitate operations on them. In the example above, MAPPS and MAPPS/OUTPUT would be considered directory names.

< mix number > - A number assigned by the system to each program when it begins running. The mix number for the example program is 1370.

```
HELLO SG3 XXX

#B6900:2063 CANDE 33.260; YOU ARE PERKO1(12)
Cande Reference Cards are now available in the computer center.
#SESSION 9098  14:06:16  10/01/82
START (SG3)MAPPS

#RUNNING 1369
#JOB 1370 IN QUEUE 0
#
#?

 ** DO YOU WANT TO:
    (E)  ENTER A NEW DATA FILE
    (M)  USE THE EXISTING 'MAPPS/INPUT/DATA' AS THE INPUT DATA FILE
    (D)  USE A DIFFERENT EXISTING INPUT DATA FILE?
E

 ** ENTER A TITLE FOR THE SIMULATION ON ONE LINE,
    1 TO 79 CHARACTERS LONG. **
EXAMPLE PROBLEM

 ** ENTER 'E' FOR ENGLISH UNITS OR 'S' FOR SI UNITS.
E

 ** ENTER THE UNIT MODULE CALCULATION ORDER LIST, SEPARATING EACH UNIT
    NUMBER WITH A BLANK, COMMA, OR PARENTHESIS.  USE AS MANY LINES AS
    NEEDED.  TERMINATE DATA WITH A 'SLASH' (/).
1 2 3 4 5/

 ** ENTER PARAMETERS FOR MODULE NUMBER    1.

    USE AN INDEXED FORMAT FOR THE FIRST VALUE ENTERED,  AND AN INDEXED,
    SEQUENTIAL, OR MIXED FORMAT (AS DEFINED IN THE MANUAL) FOR THE
    REMAINING VALUES.  TERMINATE YOUR DATA WITH A 'SLASH' (/).
1=1 25 16 2 1 1 2 3/

 ** ENTER PARAMETERS FOR MODULE NUMBER    2.
1=2 29 18 1 2 3 2 4 16=.5 .2 .6/

 ** ENTER PARAMETERS FOR MODULE NUMBER    3.
1=3 1 60 1 0 2 16=1 .001  0   30/

 ** ENTER PARAMETERS FOR MODULE NUMBER    4.
1=4 2 18  16=1   2   3/

 ** ENTER PARAMETERS FOR MODULE NUMBER    5.
1=5  3  19  0  0  0  1  0  16=1  2  3  4/

 ** DO YOU WANT TO CHANGE ANY UNIT MODULE PARAMETERS?
    ENTER MODULE NUMBER OR 'NONE'
NONE
```

```
 ** THESE ARE THE PROCESS INPUT STREAMS.  EACH MUST BE INITIALIZED.
         1
 ** ENTER INFORMATION FOR STREAM NUMBER     1

    USE AN INDEXED FORMAT FOR THE FIRST VALUE ENTERED,  AND AN INDEXED,
    SEQUENTIAL, OR MIXED FORMAT (AS DEFINED IN THE MANUAL) FOR THE
    REMAINING VALUES.  TERMINATE YOUR DATA WITH A 'SLASH' (/).
1=1 6 80 14.7 11=8000 400 40/

** THESE ARE THE TEAR STREAMS. **
        2

 ** AT THIS TIME YOU MAY:
    (M)   DISPLAY OR CHANGE MODULE PARAMETERS OR ADD A NEW MODULE
    (S)   DISPLAY OR CHANGE STREAM PARAMETERS OR INITIALIZE A NEW STREAM
    (C)   DISPLAY OR CHANGE CALCULATION ORDER LIST
    (E)   EXECUTE A SIMULATION
    (X)   TERMINATE A SESSION

    ENTER THE CORRESPONDING LETTER OF THE OPTION YOU DESIRE:
S

 ** ENTER STREAM NUMBER. **
2

 ** ENTER INFORMATION FOR STREAM NUMBER     2

    USE AN INDEXED FORMAT FOR THE FIRST VALUE ENTERED,  AND AN INDEXED,
    SEQUENTIAL, OR MIXED FORMAT (AS DEFINED IN THE MANUAL) FOR THE
    REMAINING VALUES.  TERMINATE YOUR DATA WITH A 'SLASH' (/).
1=2 6 80 14.7 11=400 80 32/

 ** AT THIS TIME YOU MAY:
    (M)   DISPLAY OR CHANGE MODULE PARAMETERS OR ADD A NEW MODULE
    (S)   DISPLAY OR CHANGE STREAM PARAMETERS OR INITIALIZE A NEW STREAM
    (C)   DISPLAY OR CHANGE CALCULATION ORDER LIST
    (E)   EXECUTE A SIMULATION
    (X)   TERMINATE A SESSION

     ENTER THE CORRESPONDING LETTER OF THE OPTION YOU DESIRE:
E

            ** THE TOPOLOGY IS BEING EVALUATED **

 ** AT THIS TIME YOU MAY:
    (M)   DISPLAY OR CHANGE MODULE PARAMETERS OR ADD A NEW MODULE
    (S)   DISPLAY OR CHANGE STREAM PARAMETERS OR INITIALIZE A NEW STREAM
    (C)   DISPLAY OR CHANGE CALCULATION ORDER LIST
    (E)   EXECUTE A SIMULATION
    (X)   TERMINATE A SESSION
```

```
 ** TOPOLOGY IS BEING WRITTEN TO OUTPUT FILE **

 ** INITIAL DATA ARE BEING WRITTEN TO OUTPUT FILE **

                    ** EXECUTING **

 ** TOTAL CPU TIME (SEC)      0.17

 ** AT THIS TIME YOU MAY:
    (M)   DISPLAY OR CHANGE MODULE PARAMETERS OR ADD A NEW MODULE
    (S)   DISPLAY OR CHANGE STREAM PARAMETERS OR INITIALIZE A NEW STREAM
    (C)   DISPLAY OR CHANGE CALCULATION ORDER LIST
    (E)   EXECUTE A SIMULATION
    (X)   TERMINATE A SESSION

    ENTER THE CORRESPONDING LETTER OF THE OPTION YOU DESIRE:
S

 ** ENTER STREAM NUMBER **
4

       1 NUMBER   =    4.00              9 EXERGY   =   68.0
       2 TYPE     =    6.00             10 FLOW     =   8.439E+03
       3 TEMP     =    80.0             11 WATER    =   8.000E+03
       4 PRESS    =    14.7             12 FIBER    =   400.
       5 CP       =   0.965             13 FINES    =   40.0.
       8 ENTHALPY =   2.443E+04

 ** DO YOU WANT TO CHANGE OR ENTER ANY NEW PARAMETERS FOR THIS STREAM? **
(Y) OR (N)
N

 ** AT THIS TIME YOU MAY:
    (M)   DISPLAY OR CHANGE MODULE PARAMETERS OR ADD A NEW MODULE
    (S)   DISPLAY OR CHANGE STREAM PARAMETERS OR INITIALIZE A NEW STREAM
    (C)   DISPLAY OR CHANGE CALCULATION ORDER LIST
    (E)   EXECUTE A SIMULATION
    (X)   TERMINATE A SESSION

    ENTER THE CORRESPONDING LETTER OF THE OPTION YOU DESIRE:
X

 ** FINAL DATA ARE BEING WRITTEN TO OUTPUT FILE **

                  ** JOB TERMINATED **
```

```
** THESE FILES WERE GENERATED. **
   (1)  MAPPS/OUTPUT/PRERUNDATA  --  SUMMARY OF INPUT DATA
   (2)  MAPPS/OUTPUT/DIAGNOSIS   --  TOPOLOGY AND DIAGNOSTICS
   (3)  MAPPS/OUTPUT/RESULTS     --  RESULTS FROM PRINT MODULES
   (4)  MAPPS/OUTPUT/POSTRUNDATA --  SUMMARY OF OUTPUT DATA

** TO ARCHIVE ANY ONE OF THESE FILES UNDER A NEW TITLE, ENTER THE
   NUMBER OF THE TITLE TO BE CHANGED.
   TO CHANGE THE SECOND LEVEL DIRECTORY NAME FOR ALL THE FILES,
   ENTER 'A'.
   TO MAKE NO CHANGES, ENTER 'N'.
A

** THE FIRST LEVEL DIRECTORY NAME ('MAPPS') AND THE LAST LEVEL DIRECTORY
   NAME WILL REMAIN THE SAME.
   ENTER A NEW SECOND LEVEL DIRECTORY NAME.
EXAMPLE

**  TOTAL CPU TIME (SEC)   5.2 **
```

FILES MAPPS/EXAMPLE

```
(SG3) ON RESEARCH
.  MAPPS
.  .  EXAMPLE
.  .  .  RESULTS : DATA
.  .  .  DIAGNOSIS : DATA
.  .  .  PRERUNDATA : DATA
.  .  .  POSTRUNDATA : DATA
#
```

LIST MAPPS/EXAMPLE/DIAGNOSIS

```
#FILE (SG3)MAPPS/EXAMPLE/DIAGNOSIS ON RESEARCH

 MAPPS EXAMPLE PROBLEM

 MODULE CALCULATION ORDER

               1   2   3   4   5

 TOPOLOGY INFORMATION BY STREAM CATEGORY

       UNUSED  STREAMS:


       INLET   STREAMS:

               1

       OUTLET  STREAMS:

               4

       CONNECT STREAMS:

               3

       TEAR    STREAMS:

               2

 TOPOLOGY INFORMATION BY STREAM NUMBER

       STREAM     CATEGORY     FROM     TO    TYPE    TYPE

           1      INLET           0      1       6       6
           2      TEAR            2      1       6       6
           3      CONNECT         1      2       6       6
           4      OUTLET          2      0       6       6

       DUPLICATE STREAMS

           2      TEAR            2      3       6      10

       NOTE: ASTERISKS DENOTE POTENTIAL CONFLICTS
```

TOPOLOGY INFORMATION BY MODULE NUMBER

MODULE	NAME	STREAMS		
1	MIXR01	GIVEN:	1	2
		CALC.:	3	
2	SEPAR1	GIVEN:	3	
		CALC.:	2	4
3	CONVRG	GIVEN:	2	
		CALC.:		
4	PRNTMV	GIVEN:		
		CALC.:		
5	PRNTSV	GIVEN:		
		CALC.:		

LIST MAPPS/EXAMPLE/PRERUNDATA

```
#FILE (SG3)MAPPS/EXAMPLE/PRERUNDATA ON RESEARCH

TITLE
 MAPPS EXAMPLE PROBLEM
PROGRAM OPTIONS
   0   1   1   1   1   1   0   0   0   0
CALCULATION ORDER
   1   2   3   4   5
MODULE   1 - MIXR01
    1             25            16            2             1
    1             2             3
MODULE   2 - SEPAR1
    2             29            18            1             2
    3             2             4             0             0
    0             0             0             0             0
    0.500000      0.200000      0.600000
MODULE   3 - CONVRG
    3             1             60            1             0
    2             0             0             0             0
    0             0             0             0             0
    1             0.00100000    0             30
MODULE   4 - PRNTMV
    4             2             18            0             0
    0             0             0             0             0
    0             0             0             0             0
    1             2             3
MODULE   5 - PRNTSV
    5             3             19            0             0
    0             1             0             0             0
    0             0             0             0             0
    1             2             3             4
STREAM   1 - INLET
    1             6             80            14.7000       0.964810
    0             0             24429         67.9844       8440
    8000          400           40
STREAM   2 - TEAR
    2             6             80            14.7000       0.852344
    0             0             1309.20       3.64342       512
    400           80            32
END
```

LIST MAPPS/EXAMPLE/RESULTS

```
#FILE (SG3)MAPPS/EXAMPLE/RESULTS ON RESEARCH

 MAPPS EXAMPLE PROBLEM


  1 MIXR01

  2 SEPAR1
 16  EFF1     0.50000      17  EFF2     0.20000      18  EFF3     0.60000

  3 CONVRG
 16  AMOD      1.0000      17  TOLRF    1.00000E-03 18  TOLRA    0.
 19  AMAX      30.000      20 *COUNT    0.           21 *ALOOP     12.000
 22 *STREAM    2.0000

  1 NUMBER     1.0000       2.0000       3.0000       4.0000
  2 TYPE       6.0000       6.0000       6.0000       6.0000
  3 TEMP       80.000       80.000       80.000       80.000
  4 PRESS      14.700       14.700       14.700       14.700
  5 HEAT CAP 0.96481      0.98677      0.97560      0.96481
  6          0.           0.           0.           0.
  7          0.           0.           0.           0.
  8 ENTHALPY  24429.       24153.       48579.       24426.
  9 EXERGY    67.984       67.217       135.19       67.977
 10 FLOW      8440.0       8159.0       16598.       8439.0
 11 WATER(L)  8000.0       7999.1       15998.       7999.1
 12 FIBER     400.00      100.000       500.00       400.00
 13 FINES     40.000       59.963       99.939       39.976
 14 DIS SLDS 0.           0.           0.           0.
 15 ADDITIVE 0.           0.           0.           0.
 16 DIRT     0.           0.           0.           0.
 ** DATE  121382  **  TIME 16:44:26  **       CPU TIME (SEC)    0.8 **

 ** DATE  121382  **  TIME 16:45: 4  **  TOTAL CPU TIME (SEC)    7.1 **
```

LIST MAPPS/EXAMPLE/POSTRUNDATA

```
#FILE (SG3)MAPPS/EXAMPLE/POSTRUNDATA ON RESEARCH

TITLE
 MAPPS EXAMPLE PROBLEM
PROGRAM OPTIONS
   0   1   1   1   0   1   0   0   0   0
CALCULATION ORDER
   1   2   3   4   5
MODULE  1 - MIXR01
    1           25            16            2            1
    1           2             3
MODULE  2 - SEPAR1
    2           29            18            1            2
    3           2             4             0            0
    0           0             0             0            0
    0.500000    0.200000      0.600000
MODULE  3 - CONVRG
    3           1             60            1            0
    2           0             0             0            0
    0           0             0             0            0
    1           0.00100000    0             30           0
    12          2             6             80           14.7000
    0.986767    0             0             24150.4      67.2090
    8158.08     7998.14       100.000       59.9391
MODULE  4 - PRNTMV
    4           2             18            0            0
    0           0             0             0            0
    0           0             0             0            0
    1           2             3
MODULE  5 - PRNTSV
    5           3             19            0            0
    0           1             0             0            0
    0           0             0             0            0
    1           2             3             4
STREAM  1 - INLET
    1           6             80            14.7000      0.964810
    0           0             24429         67.9844      8440
    8000        400           40
STREAM  2 - TEAR
    2           6             80            14.7000      0.986766
    0           0             24153.2       67.2168      8159.04
    7999.07     100.000       59.9634
END
```

```
WRITE MAPPS/EXAMPLE/=
```
[a]

```
#RUNNING 1382
#

BYE

#END SESSION 9098  ET=09:15.5 PT=11.5 IO=9.6
#USER = SG3  14:15:31  10/01/82
```

[a]This command will cause all files in the user's library beginning with MAPPS/EXAMPLE to be printed on the IPC line printer.

9. APPENDIX

9.1 SELECTED MODULE DOCUMENTATION

```
      SUBROUTINE MIXR01
C
C  DOCUMENTATION OF MIXR01  (TYPE 25)
C
C     WRITTEN BY - PAUL W. OXBY  (AUG 82)
C     REVISED BY -
C
C  DESCRIPTION:
C
C     MIXR01 SIMULATES THE ADIABATIC MIXING OF TWO TO NINE STREAMS OF
C     COMPATIBLE TYPE. THE MINIMUM NONZERO INLET STREAM PRESSURE IS
C     ASSIGNED TO THE OUTLET STREAM.
C
C  GIVEN STREAMS:
C
C     1) 1ST INLET  (ANY TYPE)
C     2) 2ND INLET  (FOR TYPE, SEE COMMENTS)
C     I) ITH INLET  (FOR TYPE, SEE COMMENTS)
C
C  CALCULATED STREAMS:
C
C     1) OUTLET     (FOR TYPE, SEE COMMENTS)
C
C  GIVEN MODULE PARAMETERS:
C
C     01) - 15)  = STANDARD PARAMETERS
C     16) TYPE   = OUTLET STREAM TYPE (OPTIONAL, SEE COMMENTS)
C
C  CALCULATED MODULE PARAMETERS:  NONE
C
C  COMMENTS:
C
C    -THIS MODULE SHOULD BE USED TO MIX STREAMS OF THE SAME TYPE.
C     IF STREAMS OF DIFFERENT TYPE MUST BE MIXED, THE TYPE OF THE
C     OUTLET STREAM WILL BE EQUAL TO THE MAXIMUM OF THE TYPES OF THE
C     INLET STREAMS. THE USER MAY OVERRIDE THIS FEATURE BY EXPLICITLY
C     SPECIFYING THE OUTLET STREAM TYPE AS A MODULE PARAMETER. THIS
C     PARAMETER SHOULD OTHERWISE BE SET EQUAL TO ZERO.
C
C    -IF STREAMS OF DIFFERENT TYPE ARE MIXED, THE NONZERO CONSTITUENTS
C     OF THESE STREAMS MUST CORRESPOND IN STREAM LOCATION WITH THE
C     CONSTITUENTS OF THE OUTLET STREAM TYPE. IN PARTICULAR, WATER
C     STREAMS MAY BE MIXED WITH ANY OTHER STREAM TYPE EXCEPT ENERGY.
C
```

```
C  ANALYSIS ASSUMPTIONS:
C
C    -THE MIXING IS ADIABATIC WITH NO CHEMICAL REACTIONS.
C
C    -THE OUTLET STREAM PRESSURE IS EQUAL TO THE MINIMUM OF THE
C     NONZERO INLET STREAM PRESSURES.
C
C  SUBPROGRAMS USED:  THERMO
C
      COMMON /MAPA/ JJ1,JJ2(10),SI(10,40),SO(10,40),EN(100)
      COMMON /MAPB/ KPRNT(10),IO(10),LOOP,NC,NCALC,LLST(200)
      EQUIVALENCE (EN(16),TYPE  )
C
C  FIND MAXIMUM STREAM TYPE AND PRESSURE
C
      NIN    = EN(4)
      ISTYPE = 0.
      PMAX   = 0.
      DO 10 I = 1,NIN
         IF (ABS(SI(I,2)).GT.ISTYPE) ISTYPE = ABS(SI(I,2))
   10    IF (SI(I,4).GT.PMAX) PMAX = SI(I,4)
      IF (TYPE.NE.0.) ISTYPE = TYPE
      SO(1,2) = ISTYPE
      IF (ISTYPE.EQ.0) GO TO 90
      IF (ISTYPE.EQ.9) GO TO 70
C
C  SUM INLET STREAM COMPONENTS AND FIND MINIMUM NONZERO STREAM PRESSURE
C
      JJ   = JJ2(ISTYPE)
      PMIN = PMAX
      DO 20 I = 1,NIN
         IF (SI(I,4).LT.PMIN .AND. SI(I,4).NE.0.) PMIN = SI(I,4)
         DO 20 J = 8,JJ
   20       SO(1,J) = SO(1,J)+SI(I,J)
      SO(1,4) = PMIN
C
C  CALCULATE THE OUTLET STREAM TEMPERATURE AND HEAT CAPACITY
C
      IF (SO(1,10).EQ.0.) GO TO 90
      GO TO (30,40,40,40,40,40,40,50,70,90),ISTYPE
C
C     HANDLE STREAM TYPE 1 (WATER)
C     ALSO CALCULATE SPECIFIC ENTHALPY AND QUALITY
C
   30 SO(1,7) = SO(1,8)/SO(1,10)
      CALL THERMO (1,6)
      GO TO 90
C
C     HANDLE STREAM TYPES 2 TO 7
C
   40 CALL THERMO (1,2)
      GO TO 90
```

```
C
C     HANDLE STREAM TYPE 8 (FUEL)
C     ALSO CALCULATE HEAT OF COMBUSTION (DRY BASIS)
C
   50 SMCP = 0.
      SMHC = 0.
      DO 60 I = 1,NIN
         SMCP = SMCP + (SI(I,10)-SI(I,11))*SI(I,5)
   60    SMHC = SMHC + (SI(I,10)-SI(I,11))*SI(I,6)
      SO(1,5) = SMCP / (SO(1,10)-SO(1,11))
      SO(1,6) = SMHC / (SO(1,10)-SO(1,11))
      CALL THERMO (1,2)
      GO TO 90
C
C     HANDLE STREAM TYPE 9 (ENERGY)
C
   70 DO 80 J = 6,9
         DO 80 I = 1,NIN
   80       SO(1,J) = SO(1,J)+SI(I,J)
C
   90 RETURN
      END
```

MODULE PARAMETER WORKSHEET: MIXR01

NO.	DESCRIPTION	NOMINAL	ACTUAL
1	MODULE NUMBER		____
2	MODULE TYPE		25
3	NUMBER OF PARAMETERS		16
4	NUMBER OF GIVEN STREAMS (N)		____
5	NUMBER OF CALCULATED STREAMS		1
6	STREAM NUMBER OF FIRST INLET STREAM (TYPE = 1-9)		____
7	STREAM NUMBER OF SECOND INLET STREAM (TYPE = 1-9)		____
.			
.			
.			
5+N	STREAM NUMBER OF LAST INLET STREAM (TYPE = 1-9)		____
6+N	STREAM NUMBER OF OUTLET STREAM (TYPE = 1-9)		____
.			
.			
.			
16	OUTLET STREAM TYPE (OPTIONAL)	0	____

```
      SUBROUTINE SEPAR1
C
C  DOCUMENTATION OF SEPAR1  (TYPE 29)
C
C     WRITTEN BY - ROBERT R. MCCONNELL  (JULY 81)
C     REVISED BY - PAUL W. OXBY (DEC 82)
C
C  DESCRIPTION:
C
C     SEPAR1 SIMULATES A SIMPLE SEPARATOR WHERE THE SEPARATION
C     EFFICIENCY (REJECT FLOW/TOTAL FLOW) OF EACH INLET STREAM
C     CONSTITUENT IS SPECIFIED. IF THE INLET STREAM IS WET STEAM, IT
C     SIMULATES A STEAM-WATER SEPARATOR BY SETTING THE SEPARATION
C     EFFICIENCY EQUAL TO ONE MINUS THE STEAM QUALITY.
C
C  GIVEN STREAMS:
C
C     1) INLET STREAM   (TYPE 1 TO 7)
C
C  CALCULATED STREAMS:
C
C     1) REJECT STREAM  (SAME TYPE AS INLET STREAM)
C     2) ACCEPT STREAM  (SAME TYPE AS INLET STREAM)
C
C  GIVEN MODULE PARAMETERS:
C
C     01) - 15)  = STANDARD PARAMETERS
C     16) SE(11) = 11TH COMPONENT SEPARATION EFFICIENCY
C     17) SE(12) = 12TH COMPONENT SEPARATION EFFICIENCY
C    5+I) SE( I) =  ITH COMPONENT SEPARATION EFFICIENCY
C
C  CALCULATED MODULE PARAMETERS:  NONE
C
C  COMMENTS:
C
C    -THIS SEPARATOR MODEL CAN BE USED TO SIMULATE CLEANERS, SCREENS,
C     SAVEALLS, PRESSES, TABLE ROLLS, FOILS, SUCTION BOXES, OR ANY
C     OTHER SEPARATION PROCESS.
C
C  ANALYSIS ASSUMPTIONS:
C
C    -THE SEPARATION PROCESS IS ISOTHERMAL AND ISOBARIC.
C
C  SUBPROGRAMS USED:  THERMO
C
      COMMON /MAPA/ JJ1,JJ2(10),SI(10,40),SO(10,40),EN(100)
      COMMON /MAPB/ KPRNT(10),IO(10),LOOP,NC,NCALC,LLST(200)
      DIMENSION SE(40)
      EQUIVALENCE (EN(16),SE(11))
```

```
C
C  INITIALIZE OUTLET STREAMS
C
      ISTYPE = SI(1,2)
      IF (ISTYPE.EQ.0) GO TO 40
      DO 10 J = 2,7
         SO(1,J) = SI(1,J)
   10    SO(2,J) = SI(1,J)
C
C  FOR WATER, SET THE SEPARATION EFFICIENCY EQUAL TO THE QUALITY.
C
      IF (ISTYPE.EQ.1) SE(11) = 1. - SI(1,6)
C
C  CALCULATE REJECT AND ACCEPT CONSTITUENT FLOWRATES
C
      JJ = JJ2(ISTYPE)
      JJMAX = EN(3)-5.
      IF (JJ.GT.JJMAX) JJ = JJMAX
      DO 20 J = 11,JJ
         SO(1,J) = SI(1,J) * SE(J)
   20    SO(2,J) = SI(1,J) - SO(1,J)
C
C  SELECT THE APPROPRIATE OPTION FOR THERMO
C
      IOPTN = 1
      IF (ISTYPE.NE.1) GO TO 30
      SO(1,6) = 0.
      SO(2,6) = 1.
      IOPTN = 4
      IF (SI(1,6).EQ.1.) IOPTN = 6
C
C  CALCULATE THE OUTLET STREAM ENTHALPIES
C
   30 CALL THERMO (1,IOPTN)
      CALL THERMO (2,IOPTN)
   40 RETURN
      END
```

MODULE PARAMETER WORKSHEET: SEPAR1

NO.	DESCRIPTION	NOMINAL	ACTUAL
1	MODULE NUMBER		____
2	MODULE TYPE		19
3	NUMBER OF PARAMETERS (15+N)[a]		____
4	NUMBER OF GIVEN STREAMS		1
5	NUMBER OF CALCULATED STREAMS		2
6	STREAM NUMBER OF FEED STREAM (TYPE = 1-9)		____
7	STREAM NUMBER OF REJECT STREAM (TYPE = 1-9)		____
8	STREAM NUMBER OF ACCEPT STREAM (TYPE = 1-9)		____
9-15	NOT USED		0
16	FIRST CONSTITUENT SEPARATION EFFICIENCY		____
17	SECOND CONSTITUENT SEPARATION EFFICIENCY		____
18	THIRD CONSTITUENT SEPARATION EFFICIENCY		____
.			
.			
.			
15+N	LAST CONSTITUENT SEPARATION EFFICIENCY		____

[a]N = Number of constituents.

```
      SUBROUTINE CONVRG
C
C  DOCUMENTATION OF CONVRG  (TYPE 1)
C
C     WRITTEN BY - PAUL W. OXBY  (JAN 82)
C     REVISED BY -
C
C  DESCRIPTION:
C
C     CONVRG CONTROLS THE REPEATED EXECUTION OF MODULES IN A CALCULATION
C     LOOP UNTIL THE COMPONENT VALUES OF A TEST STREAM HAVE CONVERGED.
C     THE TEST STREAM, CONVERGENCE TOLERANCE, AND MAXIMUM NUMBER OF
C     ITERATIONS ARE SPECIFIED.
C
C  GIVEN STREAMS:
C
C     1) CONVERGENCE TEST STREAM  (ANY TYPE)
C
C  CALCULATED STREAMS:  NONE
C
C  GIVEN MODULE PARAMETERS:
C
C     01) - 15)  = STANDARD PARAMETERS
C     16) AMOD   = NUMBER OF THE FIRST MODULE IN THE LOOP
C     17) TOLRF  = FRACTIONAL CONVERGENCE TOLERANCE
C     18) TOLRA  = ABSOLUTE   CONVERGENCE TOLERANCE
C     19) AMAX   = MAXIMUM NUMBER OF ITERATIONS
C
C  CALCULATED MODULE PARAMETERS:
C
C     20) COUNT  = ITERATION COUNTER
C     21) ALOOP  = NUMBER OF ITERATIONS
C     22) - 61)  = TEST STREAM COMPONENTS
C
C  COMMENTS:
C
C    -THIS MODULE IS PLACED AT THE END OF A CALCULATION LOOP IN THE
C     CALCULATION ORDER LIST. THE MODULE NUMBER OF THE FIRST MODULE IN
C     THE LOOP IS SPECIFIED. THIS EFFECTIVELY SPECIFIES THE NUMBER OF
C     MODULES IN THE CALCULATION LOOP.
C
C    -CONVERGENCE OF A CALCULATION LOOP IS CHECKED BY COMPARING THE
C     COMPONENTS OF A TEST STREAM WITHIN THE LOOP FOR TWO CONSECUTIVE
C     ITERATIONS. A LOOP WILL HAVE CONVERGED WHEN THE FRACTIONAL
C     CHANGE IN ALL THE COMPONENTS COMPARED IS LESS THAN A SPECIFIED
C     FRACTIONAL CONVERGENCE TOLERANCE (TOLRF).
C
C    -IF A NONZERO VALUE OF THE ABSOLUTE CONVERGENCE TOLERANCE (TOLRA)
C     IS SPECIFIED, THEN THE LOOP WILL HAVE CONVERGED WHEN THE
C     MAGNITUDE OF THE TEST STREAM FLOWRATE IS LESS THAN THE ABSOLUTE
C     CONVERGENCE TOLERANCE. THIS OPTION IS NECESSARY, WHEN A TEST
C     STREAM FLOWRATE IS BEING FORCED TO ZERO BY A CONTROLLER.
```

```
C
      COMMON /MAPA/ JJ1,JJ2(10),SI(10,40),SO(10,40),EN(100)
      COMMON /MAPB/ KPRNT(10),IO(10),LOOP,NC,NCALC,LLST(200)
      EQUIVALENCE (EN(16),AMOD  ),(EN(17),TOLRF ),(EN(18),TOLRA ),
     *            (EN(19),AMAX  ),(EN(20),COUNT ),(EN(21),ALOOP )
      DATA NPARM /21/
C
C  INITIALIZE CONSTANTS
C
      IF (LOOP.GE.999) RETURN
      LMSG   = IO(6)
      IMOD   = EN(1)
      ISTYPE = ABS(SI(1,2))
      JJ = JJ1
      IF (ISTYPE.NE.0) JJ = JJ2(ISTYPE)
      JJMAX = IFIX(EN(3))-NPARM
      IF (JJ.GT.JJMAX) WRITE (LMSG,610) IMOD
      IF (JJ.GT.JJMAX) JJ = JJMAX
      IF (COUNT.EQ.0.) LOOP = LOOP+1
      COUNT = COUNT+1.
      ALOOP = COUNT
      IF (COUNT.EQ.1.) GO TO 20
C
C  CHECK CONVERGENCE TOLERANCES
C
      JJ3 = JJ
      IF (TOLRA.EQ.0.) GO TO 5
      IF (ABS(SI(1,10)).GT.TOLRA) GO TO 20
      IF (ABS(EN(10+NPARM)).GT.TOLRA) GO TO 20
      JJ3 = 7
    5 DO 10 J = 3,JJ3
         IF (J.EQ.8.OR.J.EQ.9) GO TO 10
         S = SI(1,J)
         IF (S.EQ.0.) GO TO 10
         TEST = ABS(EN(J+NPARM)/S-1.)
         IF (TEST.GT.TOLRF) GO TO 20
   10    CONTINUE
C
C  LOOP HAS CONVERGED - LEAVE LOOP
C
      COUNT = 0.
      LOOP  = LOOP-1
      GO TO 60
C
C  LOOP HAS NOT CONVERGED - ITERATE AGAIN
C
   20 IF (COUNT.GE.AMAX) GO TO 50
      MOD  = AMOD+0.1
      NCM1 = NC-1
      DO 30 I = 1,NCM1
         IF (LLST(NC-I).EQ.MOD) GO TO 40
   30    CONTINUE
      GO TO 50
   40 NC = NC-I-1
      GO TO 60
```

```
C
C  LOOP HAS NOT CONVERGED - TERMINATE PROGRAM
C
   50 LMSG = IO(6)
      IMOD = EN(1)
      WRITE (LMSG,600) IMOD
      COUNT = 0.
      LOOP  = 999
      NC = 0
C
C  STORE THE TEST STREAM VALUES
C
   60 DO 70 J = 1,JJ
   70    EN(J+NPARM) = SI(1,J)
      RETURN
  600 FORMAT(' ** ERROR **  MODULE',I4,
     *       ' (CONVRG) : ITERATIONS HAVE NOT CONVERGED')
  610 FORMAT(' ** WARNING **  MODULE',I4,
     *       ' (CONVRG) : PARAMETER VECTOR MAY BE TOO SHORT')
      END
```

MODULE PARAMETER WORKSHEET: CONVRG

NO.	DESCRIPTION	NOMINAL	ACTUAL
1	MODULE NUMBER		
2	MODULE TYPE		1
3	NUMBER OF PARAMETERS		61
4	NUMBER OF GIVEN STREAMS		1
5	NUMBER OF CALCULATED STREAMS		0
6	STREAM NUMBER OF TEST STREAMS		
7-15	NOT USED		
16	NUMBER OF FIRST MODULE IN LOOP		
17	FRACTIONAL CONVERGENCE TOLERANCE	0.001	
18	ABSOLUTE CONVERGENCE TOLERANCE		
19	MAXIMUM NUMBER OF ITERATIONS	20	
*			

*20-61 CALCULATED MODULE PARAMETERS

```
      SUBROUTINE PRNTMV
C
C  DOCUMENTATION OF PRNTMV  (TYPE 2)
C
C     WRITTEN BY - PAUL W. OXBY  (JAN 82)
C     REVISED BY -
C
C  DESCRIPTION:
C
C     PRNTMV PRINTS MODULE VECTORS WITH LABELS. THE MODULES TO BE
C     PRINTED ARE SPECIFIED.
C
C  GIVEN STREAMS:  NONE
C
C  CALCULATED STREAMS:  NONE
C
C  GIVEN MODULE PARAMETERS:
C
C     01) = MODULE NUMBER
C     02) = MODULE TYPE = 2
C     03) = NUMBER OF PARAMETERS = 15 + NUMBER OF MODULES PRINTED
C     04) = NUMBER OF GIVEN STREAMS = 0
C     05) = NUMBER OF CALC. STREAMS = 0
C     06) = PRINT CONTROL = 0 = PRINT ON LAST  PASS THRU FLOWSHEET
C                         = 1 = PRINT ON EVERY PASS THRU FLOWSHEET
C     07) -  15) = NOT USED
C     16) - 100) = MODULE NUMBERS OF MODULES TO BE PRINTED
C
C  CALCULATED MODULE PARAMETERS:  NONE
C
C  COMMENTS:
C
C    -IF ONLY THE CONVERGED RESULTS ARE TO BE PRINTED, THEN THE PRINT
C     CONTROL PARAMETER (NUMBER 6) SHOULD BE SET TO ZERO. IF THE
C     INTERMEDIATE RESULTS IN A CALCULATION LOOP ARE TO BE PRINTED,
C     THEN THIS MODULE SHOULD BE PLACED IN THE LOOP AND THE PRINT
C     CONTROL PARAMETER SHOULD BE SET TO ONE.
C
      DOUBLE PRECISION LABLM1,LABLM2
      COMMON /MAPA/ JJ1,JJ2(10),SI(10,40),SO(10,40),EN(100)
      COMMON /MAPB/ KPRNT(10),IO(10),LOOP,NC,NCALC,LLST(200)
      COMMON /MAPC/ NPOINT(200),EEN(3000),SN(40,300)
      COMMON /MAPD/ LABLM1(150),LABLM2(1000),MPOINT(2,150)
      DIMENSION VALUE(100),INDEX(100)
C
C  INITIALIZE CONSTANTS
C
      IF (KPRNT(2).EQ.0) RETURN
      IF (LOOP.NE.999 .AND. EN(6).EQ.0.) RETURN
      KMOD  = IFIX(EN(3))-15
      LO    = IO(5)
```

```
C  ITERATE THROUGH THE MODULES TO BE PRINTED
C
      DO 20 K = 1,KMOD
         IMOD = EN(K+15)
         IF (IMOD.EQ.0) RETURN
         MQ = NPOINT(IMOD)
         IF (MQ.EQ.0) GO TO 20
         IMTYPE = ABS(EEN(MQ+2))
         WRITE (LO,600)
         WRITE (LO,600) IMOD,LABLM1(IMTYPE)
         LEN  = MPOINT(2,IMTYPE)
         IF (LEN.EQ.0) GO TO 20
         IPNT = MPOINT(1,IMTYPE)
         LEN2 = 0
         DO 10 J2 = 1,LEN
            J = LEN+1-J2
            VALUE(J) = EEN(MQ+15+J)
            INDEX(J) = 15+J
   10       IF (VALUE(J).NE.0. .AND. LEN2.EQ.0) LEN2 = J
         WRITE (LO,600) (INDEX(J),LABLM2(IPNT+J),VALUE(J),J=1,LEN2)
   20    CONTINUE
      RETURN
C
C  FORMAT STATEMENTS
C
  600 FORMAT (3(I3,1X,A8,1P,G12.5))
      END
```

MODULE PARAMETER WORKSHEET: PRNTMV

NO.	DESCRIPTION	NOMINAL	ACTUAL
1	MODULE NUMBER		____
2	MODULE TYPE		2
3	NUMBER OF PARAMETERS (15+N)[a].		____
4	NUMBER OF GIVEN STREAMS		0
5	NUMBER OF CALCULATED STREAMS		0
6	PRINT CONTROL PARAMETER	0	____
7-15	NOT USED		____
16	MODULE NUMBER OF FIRST MODULE		____
17	MODULE NUMBER OF SECOND MODULE		____
18	MODULE NUMBER OF THIRD MODULE		____
.			
.			
.			
10+N	MODULE NUMBER OF LAST MODULE		____

[a]N = number of modules to be printed.

```
      SUBROUTINE PRNTSV
C
C  DOCUMENTATION OF PRNTSV  (TYPE 3)
C
C     WRITTEN BY - PAUL W. OXBY  (SEPT 82)
C     REVISED BY -
C
C  DESCRIPTION:
C
C     PRNTSV PRINTS STREAM VECTORS WITH LABELS. THE STREAMS TO BE
C     PRINTED ARE SPECIFIED.
C
C  GIVEN STREAMS:  NONE
C
C  CALCULATED STREAMS:  NONE
C
C  GIVEN MODULE PARAMETERS:
C
C     01) = MODULE NUMBER
C     02) = MODULE TYPE = 3
C     03) = NUMBER OF PARAMETERS = 15 + NUMBER OF STREAMS PRINTED
C     04) = NUMBER OF GIVEN STREAMS = 0
C     05) = NUMBER OF CALC. STREAMS = 0
C     06) = PRINT  CONTROL = 0 = PRINT ON LAST  PASS THRU FLOWSHEET
C                          = 1 = PRINT ON EVERY PASS THRU FLOWSHEET
C     07) = FORMAT CONTROL = 0 = PRINT STREAM VECTORS ROWWISE
C                          = 1 = PRINT STREAM VECTORS COLUMNWISE
C     08) -  15) = NOT USED
C     16) - 100) = STREAM NUMBERS OF STREAMS TO BE PRINTED
C
C  CALCULATED MODULE PARAMETERS:  NONE
C
C  COMMENTS:
C
C    -IF ONLY THE CONVERGED RESULTS ARE TO BE PRINTED, THEN THE PRINT
C     CONTROL PARAMETER (NUMBER 6) SHOULD BE SET TO ZERO. IF THE
C     INTERMEDIATE RESULTS IN A CALCULATION LOOP ARE TO BE PRINTED,
C     THEN THIS MODULE SHOULD BE PLACED IN THE LOOP AND THE PRINT
C     CONTROL PARAMETER SHOULD BE SET TO ONE.
C
      DOUBLE PRECISION LABLS1,LABLS2,LABLS3
      COMMON /MAPA/ JJ1,JJ2(10),SI(10,40),SO(10,40),EN(100)
      COMMON /MAPB/ KPRNT(10),IO(10),LOOP,NC,NCALC,LLST(200)
      COMMON /MAPC/ NPOINT(200),EEN(3000),SN(40,300)
      COMMON /MAPE/ LABLS1(200),KSTRM(40,10),THERM(6,100)
      DIMENSION VALUE(40),LABLS2(40)
C
C  INITIALIZE CONSTANTS
C
      IF (KPRNT(2).EQ.0) RETURN
      IF (LOOP.NE.999 .AND. EN(6).EQ.0.) RETURN
      KSTR  = IFIX(EN(3))-15
      NBLK  = (KSTR+4)/5
      LO    = IO(5)
      IF (EN(7).NE.0.) GO TO 30
```

```
C
C  PRINT STREAMS IN ROWS
C
      DO 20 K = 1,KSTR
         ISTR = EN(K+15)
         IF (ISTR.EQ.0) RETURN
         ISTYPE = ABS(SN(2,ISTR))+0.1
         IF (ISTYPE.EQ.0) ISTYPE = 1
         WRITE (LO,600)
C
C     PRINT STREAM LABELS AND VALUES
C
         JJ = JJ2(ISTYPE)
         DO 10 J = 1,JJ
            VALUE(J) = SN(J,ISTR)
   10       LABLS2(J) = LABLS1(KSTRM(J,ISTYPE))
         WRITE (LO,600) (J,LABLS2(J),VALUE(J),J=1,JJ)
   20    CONTINUE
      RETURN
C
C  PRINT STREAMS IN COLUMNS
C
   30 DO 50 K = 1,NBLK
         I1 = (K-1)*5+1
         I2 = I1+4
         IF (K.EQ.NBLK) I2 = KSTR
         ISTR = EN(I1+15)
         IF (ISTR.EQ.0) RETURN
         ISTYPE = ABS(SN(2,ISTR))+0.1
         IF (ISTYPE.EQ.0) ISTYPE = 10
         JJ = JJ2(ISTYPE)
         WRITE (LO,600)
C
C     PRINT STREAM LABELS AND VALUES
C
         DO 50 J = 1,JJ
            LABLS3 = LABLS1(KSTRM(J,ISTYPE))
            N = 0
            DO 40 I = I1,I2
               ISTR = EN(I+15)
               IF (ISTR.EQ.0) GO TO 50
               N = N+1
   40          VALUE(N) = SN(J,ISTR)
   50       WRITE (LO,610) J,LABLS3,(VALUE(I),I=1,N)
      RETURN
C
C  FORMAT STATEMENTS
C
  600 FORMAT (3(I3,1X,A8,1P, G12.5))
  610 FORMAT (  I3,1X,A8,1P,5G12.5)
      END
```

MODULE PARAMETER WORKSHEET: PRNTSV

NO.	DESCRIPTION	NOMINAL	ACTUAL
1	MODULE NUMBER		____
2	MODULE TYPE		3
3	NUMBER OF PARAMETERS (15+N)[a]		____
4	NUMBER OF GIVEN STREAMS		0
5	NUMBER OF CALCULATED STREAMS		0
6	PRINT CONTROL PARAMETER	0	____
7	FORMAT CONTROL PARAMETER	0	____
8-15	(NOT USED)		____
16	STREAM NUMBER OF FIRST STREAM		____
17	STREAM NUMBER OF SECOND STREAM		____
18	STREAM NUMBER OF THIRD STREAM		____
.			
.			
.			
15+N	STREAM NUMBER OF LAST STREAM		____

[a]N = number of streams to be printed.