188
4/13/78
ms NTIS

AR. 2017

CONS/0385-2
NASA CR-135284
BCS 40180-2

A SIMULATION MODEL FOR WIND ENERGY STORAGE SYSTEMS

MASTER

Volume II: Operation Manual

A. W. Warren, R. W. Edsinger, J.D. Burroughs
ENERGY TECHNOLOGY APPLICATIONS DIVISION
BOEING COMPUTER SERVICES COMPANY
A Division of The Boeing Company

August 1977

Prepared for the
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Lewis Research Center
Cleveland, Ohio 44135

Contract NAS 3-20385

As a part of the
ENERGY RESEARCH AND
DEVELOPMENT ADMINISTRATION
Division of Energy Storage Systems

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

NOTICE

This report was prepared to document work sponsored by the United States Government. Neither the United States nor its agent, the United States Department of Energy, nor any Federal employees, nor any of their contractors, subcontractors or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

TABLE OF CONTENTS

	<u>Page</u>
1.0 INTRODUCTION	1
1.1 GENERAL APPROACH	1
1.2 SIMWEST LIBRARY	2
1.2.1 Overview	2
1.2.2 Storage Subsystems	6
1.2.3 Logic Components	6
1.2.4 SIMWEST Output	13
2.0 MODEL GENERATION	17
2.1 MODEL DESCRIPTION	18
2.1.1 Phrases and Delimiters	20
2.1.2 Command Phrases	22
2.2 NAMING CONVENTION	30
2.2.1 Variable, Parameter, and Table Naming Convention	31
2.3 MODEL SCHEMATIC	32
2.3.1 Standard Schematic Form	32
2.3.2 Input Quantity Labeling	32
2.3.3 Component Connection Paths	34
2.3.4 Additional Pages	34
2.3.5 Guidelines for Schematic Layout	36
2.4 WARNING MESSAGES	36
2.5 MODEL GENERATION LIMITATIONS	39
3.0 SIMULATION PROGRAM	41
3.1 MODEL INPUT DATA	41
3.1.1 Scalar Data	41
3.1.2 Tabular Data	42
3.2 INITIAL CONDITION AND INTEGRATION CONTROLS	44
3.3 INITIAL CONDITION STORAGE COMMANDS	45
3.4 SIMULATION COMMANDS	45

NOTICE

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Department of Energy, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

TABLE OF CONTENTS (Continued)

	<u>Page</u>
3.5 PLOT DESIGNATION COMMANDS	48
3.6 ITERATION AND DIAGNOSTIC CONTROL	50
3.7 DEFINE COMMANDS	51
3.8 EXAMPLE OUTPUT	52
4.0 JOB CONTROL PROCEDURES	55
4.1 MODEL GENERATION AND ANALYSIS EXECUTION	55
4.2 PROGRAM MAINTENANCE AND LIBRARY UPDATES	57
5.0 DIAGNOSTICS	65
5.1 WARNING MESSAGES	65
5.2 DIAGNOSTIC MESSAGES FOR LIBRARY COMPONENTS	67
6.0 CREATION OF NEW LIBRARY COMPONENTS	71
6.1 LIBRARY COMPONENT CODING	71
6.1.1 Component Call Sequence	71
6.1.2 Additions and Modifications to Component Library	76
6.1.3 Coding Conventions	78
6.2 FILOAD PROGRAM	82
6.2.1 FILOAD Program Commands	82
6.2.2 Input Name Lists	85
6.2.3 Output Name Lists	86
6.2.4 Table Name Lists	88
7.0 LIBRARY COMPONENT DESCRIPTIONS	89
7a. INPUT/OUTPUT NAME LISTS	89
7b. INPUT PARAMETER SPECIFICATION	90
7c. COMPONENT LOGIC	90
7.1 ADMITTANCE	93
7.2 TEST FUNCTION GENERATOR	100
7.3 BATTERY	103
7.4 BURNER	110
7.5 COST MONITOR	117
7.6 COMPRESSOR (PNEUMATIC)	123
7.7 PNEUMATIC STORAGE VESSEL (CONSTANT PRESSURE)	131
7.8 FLYWHEEL/CLUTCH	141

TABLE OF CONTENTS (Continued)

	<u>Page</u>
7.9 ONE DIMENSION TABLE LOOKUP	152
7.10 TWO DIMENSION TABLE LOOKUP	154
7.11 AC INDUCTION GENERATOR	157
7.12 FIXED RATIO TRANSMISSION	163
7.13 HISTOGRAM	168
7.14 HYDRO STORAGE VESSEL	172
7.15 HYDRAULIC TURBINE	181
7.16 ADIABATIC HEAT EXCHANGER	188
7.17 ADIABATIC HEAT EXCHANGER - DISCHARGING CYCLE	207
7.18 INTEGRATOR WITH SATURATION	211
7.19 DC-AC INVERTER	214
7.20 FIRST ORDER LAG	219
7.21 LEAD LAG	221
7.22 ELECTRICAL LOAD	224
7.23 MULTIPLY AND ADD	230
7.24 MULTIPLY, DIVIDE, AND ADD	232
7.25 MULTIPLY AND ADD	235
7.26 AC INDUCTION MOTOR	237
7.27 POWER ACCUMULATOR	244
7.28 POWER DIVIDER	254
7.29 PRIORITY INTERRUPT	264
7.30 HYDRAULIC PUMP	266
7.31 AC-DC RECTIFIER	273
7.32 RANDOM NUMBERS	279
7.33 SATURATION FUNCTION	281
7.34 SINGLE POLE SWITCH	284
7.35 TWO POLE SWITCH	287
7.36 THREE POLE SWITCH	290
7.37 FOUR POLE SWITCH	293
7.38 TAPE/FILE READ	296
7.39 SECOND ORDER TRANSFER FUNCTION	302
7.40 TIME CONVERSION	305
7.41 THERMAL LOAD	308
7.42 AMBIENT TEMPERATURE	313
7.43 VARIABLE RATIO TRANSMISSION	318
7.44 THERMAL STORAGE CHAMBER	324
7.45 TURBINE (PNEUMATIC)	334
7.46 UTILITY	341
7.47 WIND	347
7.48 TURBINE/GENERATOR	353
7.49 WIND TURBINE	358

TABLE OF CONTENTS (Continued)

	<u>Page</u>
8.0 EXAMPLES	365
8.1 WIND TURBINE AND FILE READ MODEL	365
8.2 BATTERY STORAGE MODEL	375
8.3 FLYWHEEL STORAGE MODEL	384
8.4 HYDRO AND THERMAL STORAGE MODEL	392
8.5 PNEUMATIC STORAGE MODEL	402

LIST OF FIGURES

	<u>Page</u>
1-1 SIMWEST Program Organization	3
1-2 Pneumatic Storage Subsystem	8
1-3 Pumped Hydro Storage Subsystem	9
1-4 Flywheel Storage	10
1-5 Battery Storage	10
1-6 Thermal Storage	11
1-7 Example of Power Divider and Accumulator Use	14
1-8 Plumbrook Configuration for Parameter Study	15
2.1-1 Analyst's Sketch of Wind Turbine Model Schematic	19
2.1-2 Lineprinter-Drawn Wind Turbine Model Schematic	21
2.2-1 Character Assignment Input/Output or Table Name	33
2.3-1 Component Connection Paths	35
3.6 Typical Diagnostic Output	51
3.8 Sample Printer Output	53
4.1-1 SIMWEST Program Execution Structure	56
4.1-2 XQTEASY Job Control File	58
4.1-3 XQTANALYSIS Job Control File	58
4.2-1 XQTFILOAD Job Control File	60
4.2-2 MAPFILOAD Procedure File	62
4.2-3 MAPEASY Procedure File	62
4.2-4 MAPANALYSIS Procedure File	62
4.2-5 MAPNSMPPT Procedure File	63
6.1-1 Sample Component Code	79
6.1-2 Sample Component Code	80
6.2-1 List of Standard Component Symbols	84
7.0 Sample Connections for Logic Components	92
7.3 Battery Circuit Diagram	103
7.7 Constant Pressure Air Storage	132
7.12 Fixed Gear Power Loss	163
7.16-1 Koutz - Glendenning Adiabatic Compressed Air Storage Scheme (Single-Stage Heat-of-Compression Storage)	189

LIST OF FIGURES (Continued)

	<u>Page</u>
7.16-2 Enthalpy - Temperature Diagram for HX	191
7.16-3 Storage Temperature Versus Tube Length	191
7.19 Inverter Functional Diagram	214
7.31 Rectifier Functional Diagram	273
7.43 Transmission Model - Lookup Table	318
7.44 Temperature - Enthalpy Diagram	326
7.48 Output Power Versus Wind Velocity	353
7.49 Generalized Machine Power Output Performance	361
8.1-1 Wind Turbine and File Read Example	366
8.1-2 Input Data for File Read Model	367
8.1-3 Wind Turbine and File Read Model Schematic	368
8.1-4 Input Data for Analysis Program	369
8.1-5 Tape Read Formatted Load Data	371
8.1-6 Wind Turbine Power Histogram	372
8.1-7 Wind Power Output Versus Wind Velocity	373
8.1-8 Weekly Load Profile	374
8.2-1 Battery Storage Example	376
8.2-2 Battery Model Input Data	377
8.2-3 Battery Model Schematic	378
8.2-4 Input Data for Battery Simulation	379
8.2-5 Cost Monitor Output for Battery Model	380
8.2-6 Wind Profile for Battery Simulation	381
8.2-7 Wind Power Supplied to Load	382
8.2-8 Battery Potential Energy Storage	383
8.3-1 Flywheel Storage Example	385
8.3-2 Flywheel Model Input Data	384
8.3-3 Flywheel Model Schematic	386
8.3-4 Flywheel Simulation Data	388
8.3-5 Wind Power Supplied to Flywheel Storage	389
8.3-6 Flywheel Kinetic Energy Storage	390

LIST OF FIGURES (Continued)

	<u>Page</u>
8.3-7 Flywheel Model Cost Monitor Output	391
8.4-1 Hydro and Thermal Storage Example	393
8.4-2 Hydro and Thermal Model Input Data	392
8.4-3 Hydro and Thermal Model Schematic	394
8.4-4 Hydro and Thermal Simulation Data	395
8.4-5 Hydro Reservoir Energy Storage	397
8.4-6 Percent Cumulative Load Delivered	398
8.4-7 Thermal Energy Storage	399
8.4-8 Percent Cumulative Thermal Load Delivered	400
8.4-9 Ambient Temperature Simulation Over One Week	401
8.5-1 Pneumatic Storage Example	403
8.5-2 Pneumatic Storage Model Input Data	402
8.5-3 Pneumatic Storage Simulation Data	404
8.5-4 Average Temperature in Heat Exchanger Cell 2	405
8.5-5 Heat Exchanger Outlet Temperature (Charging)	406
8.5-6 Air Mass in Pneumatic Storage	407
8.5-7 Air Mass Temperature in Pneumatic Storage Vessel	408
8.5-8 Heat Exchanger Outlet Temperature (Discharging)	409

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

LIST OF TABLES

	<u>Page</u>
1-1 SIMWEST Library Components	4
1-2 Partial List of Component Inputs and Outputs	7
2.1-1 Model Generation Program Language Delimiters	20
3.4-1 Print Control Values	47
4.2 SIMWEST Maintenance File Directory	61
6.1-1 Component Subroutine Call Sequence Order	77

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

FOREWARD

This report presents results of work conducted by Boeing Computer Services Company under NASA Contract NAS3-20385, "Wind Energy Storage Model Development." This program was conducted under the sponsorship of the Advanced Physical Methods Branch, Office of Conservation, ERDA, under the direction of Dr. G. C. Chang, and was administered by the NASA-Lewis Research Center Thermal and Mechanical Storage Section with Mr. L. H. Gordon as Project Manager. This report is in three volumes.

- I. Technical Report
- II. Operation Manual
- III. Program Descriptions

The Boeing Program Manager for this work was R. W. Edsinger, and A. W. Warren was the principal investigator.

For completeness, the summary sections 1.1 and 1.2 of Volume I have been repeated in the Operation Manual, Volume II.

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

1.0 INTRODUCTION

Energy storage systems for the utilization of intermittent power sources have received increased study over the past few years. However, the type and degree of storage required for optimal utilization of wind energy and the total costs and utility of the resulting wind turbine/generator/storage system have not been thoroughly analyzed. The purpose of the SIMWEST (Simulation Model for Wind Energy Storage) program described in this document is to provide a tool for performing this needed analysis. It is a tool to aid in the design of an optimal wind energy system for a given application and to allow the resulting system to be evaluated and verified through realistic simulation.

SIMWEST consists of a library of system components and a precompiler program which allows these components to be put together in building block form. The present library contains components for five types of energy storage systems. They are pumped hydro, battery, thermal, flywheel, and pneumatic. The SIMWEST program, as described here runs on the UNIVAC 1100 series of computers.

Other computer programs exist for the simulation of wind systems and various forms of energy storage. However SIMWEST is the only program capable of simulating total wind systems containing any one or combination of the above types of storage and at the same time having the flexibility and depth required to perform thorough and meaningful parameter studies.

1.1 GENERAL APPROACH

The structure and much of the software for the SIMWEST program is based on a computer program called EASY, which was previously developed by Boeing Computer Services, Inc. (BCS). SIMWEST consists of two basic programs, and a library of system, environmental, and load components. The first program, the Model Generation Program, is a precompiler which generates computer models (in FORTRAN) of complex wind energy generator/storage/application systems, from user specifications using SIMWEST library components. The second program exercises the resulting computer model to perform cost and potential utilize-

tion analysis. It handles input, output, integration of system dynamics, and iterates to obtain convergence of variables involved in implicit loops. The combination of these two programs provides a powerful tool for analyzing alternate storage system designs.

Figure 1-1 shows the general organization of the SIMWEST program. In addition to the two programs described above, there is a third which performs file maintenance. It is used to incorporate user supplied data for new subsystem models. Although the program is shown to be made up of a number of subprograms, it can be executed as a single batch program by supplying the model description control cards and the control cards describing the desired analysis to be performed and the desired tabular and/or plotted output.

1.2 SIMWEST LIBRARY

1.2.1 Overview

The SIMWEST library is listed in Table 1-1. It is made up of five types of components: physical, environmental, load, logical, and utility.

Physical components encompass such things as motors, generators, transmissions, flywheels, etc. These components model actual physical hardware which might be used in a wind energy system. The selection of the particular SIMWEST library set of physical components was based on the requirement that it be capable of modeling five types of energy storage systems mentioned previously: thermal, flywheel, battery, pumped hydro and pneumatic.

The degree of detail in the component models is based upon two design criteria. First, all models should contain sufficient detail to simulate all physical characteristics and constraints having significant impact on the systems overall cost effectiveness. Second, the models should be designed to minimize computer time and required user specification. It is assumed that a typical SIMWEST simulation might cover a time span of one year. Thus

FIGURE 1-1 SIMEST PROGRAM ORGANIZATION

TABLE 1-1 SIMWEST LIBRARY COMPONENTS

<u>PHYSICAL</u>		<u>ENVIRONMENTAL</u>	
WIND TURBINE	WT	WIND	WD
TURBINE/GENERATOR	WP	AMBIENT TEMP	TP
AC INDUCTION GEN.	GE		
FIXED RATIO TRANSMISSION	GR	<u>LOAD</u>	
RECTIFIER	RE	ELECTRICAL LOAD	LO
BATTERY	BA	THERMAL LOAD	TL
INVERTER	IV		
ADMITTANCE	AD	<u>LOGIC</u>	
COMPRESSOR (PNEUMATIC)	CO	POWER DIVIDER	PD
ADIABATIC HEAT EXCHANGER (INPUT CYCLE)	HX	POWER ACCUMULATOR	PA
ADIABATIC HEAT EXCHANGER (OUTPUT CYCLE)	HY	PRIORITY INTERRUPT SWITCH	PI
PNEUMATIC STORAGE VESSEL	CS		SW, SX, SY, SZ
BURNER	BN	<u>UTILITY</u>	
TURBINE (PNEUMATIC)	TU	COST MONITOR	CM
INDUCTION MOTOR	MO	SATURATION	SA
VARIABLE RATIO TRANSMISSION	TR	RANDOM NUMBER GENERATOR	RN
FLYWHEEL/CLUTCH	FL	TEST FUNCTIONS	AF
PUMP (HYDRO)	PU	TABLE LOOKUP	FU, FV
TURBINE (HYDRO)	HT	TRANSFER FUNCTION	IT, LA, LL, TF
HYDRO STORAGE	HS	ARITHMETIC ELEMENT	MA, MB, MC
THERMAL STORAGE	TS	HISTOGRAM	HG
UTILITY	UT	TAPE READ	TA
		TIME CONVERSION	TI

from a computer run time and economic impact point of view a simulation step size of between 15 minutes and one hour seemed reasonable.

As a result of the above two design criteria many physical components, such as the electrical components, were modeled mainly in terms of power flow and steady state response. This lack of detail is consistent with the 15 minute time step and with the concept that the important transients are on the time scale of demand curves or weather patterns, i.e. an hour or more, rather than on the time scale of electric motor transients of a few seconds. If short electric transients were to be modeled, much detail would need to be added to the component models which would greatly increase the user's task of specifying the model. Further, the simulation time step would have to be reduced to a fraction of a second so the model would not only be much larger but computer runs would be much costlier.

Environmental components are those which simulate environmental conditions. In the present SIMWEST library these conditions are wind speed and ambient temperature. These variables are generally used as inputs to physical components. Environmental component output can either be computed from measurement data provided by the user on a data tape, or from randomly generated data, based on user furnished profiles.

The load components in the SIMWEST library are used to simulate various types of power demand. They also monitor how well the system meets the simulated demand and compute the value of the energy delivered by the wind energy system to the load. Like the environmental components these components may be computed from actual measurement data or from randomly generated data based on user furnished load profiles.

The library's logical components are the power dividers, power accumulators, switches and priority interrupts. Although physical hardware could generally be built to serve the function of the logical components, they are not meant to represent any particular piece of existing hardware. Instead, they are

idealized components that allow the user the flexibility of modeling the wide variety of control logic which a wind energy storage system would require. In practice, the control function might be performed by a control room operator using a predefined control strategy or by use of a minicomputer.

Finally, the utility components include such things as the tape read, the histogram and the cost monitor. These components serve only to help the user run the simulation and analyze its results.

1.2.2 Storage Subsystems

Figures 1-2 through 1-6 give example configurations of the five types of storage subsystems which can be modeled with the present SIMNEST library. For illustrative purposes the number of variables shown passed between components is limited. A description of the variables being passed is given in Table 1-2.

A total wind energy system will generally be made up of elements from a number of different subsystems (see Figure 1-8). In addition, the SIMNEST program can be used for models which include networks of storage subsystems of the same type or a network of wind generators.

1.2.3 Logic Components

The capability for modeling complex system control logic is provided by the power divider, power accumulator and priority interrupt components. Both the divider and accumulator operate on a priority basis. The priority interrupt is used by other system components to change the priority setting of the divider and accumulator.

The power divider has one input power port and four output power ports (not all output ports need be used for a given simulation). The divider also has

TABLE 1-2 PARTIAL LIST OF COMPONENT INPUTS AND OUTPUTS

<u>SYMBOLS</u>	
P	POWER
RE	POWER REQUEST
MP	MAXIMUM POWER
RS	ROTOR SPEED
T	TEMPERATURE
TA	AMBIENT TEMPERATURE
M	MASS FLOW RATE
H	RESERVOIR HEIGHT
LD	THERMAL LOAD DELIVERED
WV	WIND VELOCITY
GR	GEAR RATIO
EF	EFFICIENCY
INT	INTERRUPT FLAG
TSO	MINIMUM AIR TEMPERATURE
PR	PRESSURE
PS	PRIORITY SEQUENCE
WY	WEEK OF YEAR
DW	DAY OF WEEK
TD	TIME OF DAY
SP	SURPLUS POWER
VAR	FILE READ VARIABLE

FIGURE 1-2 PNEUMATIC STORAGE SUBSYSTEM

FIGURE 1-3 PUMPED HYDRO STORAGE SUBSYSTEM

FIGURE 1-4 FLYWHEEL STORAGE

FIGURE 1-5 BATTERY STORAGE

FIGURE 1-6 THERMAL STORAGE

an input request associated with each of its output power ports. These power requests generally come from a component with which the output power port is directly or indirectly connected. The user specifies priorities of either 0,1,2,3, or 4 to be associated with each of the output ports. If the input power to the power divider exceeds that requested of the port with highest priority (priority 1) then the excess power goes to the port with the next lower priority. This process continues until either all power is distributed or all requests of non-zero priority ports are met. A port with zero (0) priority will never receive power. Such ports are included so that the port may be connected to a component but transmit power only in critical situations, say, when a battery has been in discharge state for a critical amount of time. In these situations, the connected component would have to change the zero priority setting of the power divider by use of a priority interrupt.

Two or more ports may be assigned the same priority in which case the user may specify weights to be associated with each port. Then if there is not enough power available to satisfy all requests of equal priority the power is divided between them in proportion to the user specified weights.

The power accumulator is similar to the divider except that instead of distributing power from a single input port between four output ports, it accumulates power from four input ports and sends it out through a single output port. The power accumulator also accepts output power requests from the component connected to its single output power port and it outputs requests for each of its input ports in order to service the output power request.

In addition to the actual power delivered to each input port, the power accumulator also accepts information as to the maximum power that can be delivered to that port. These values are used by the accumulator to determine how to distribute its power request between its four input ports. If the input power request exceeds the maximum deliverable power for the port of highest priority, then the remainder is shifted to the port with the next lower priority. This process continues until either the power request has been completely distributed between the highest priority input ports or all input

ports have requests equal to their maximum deliverable power. An example illustrative of the use of power dividers and power accumulators is given in Figure 1-7.

Here the wind turbine distributes power first to satisfy the request from load 1. If there is power left over, it then tries to satisfy the request from the battery. Finally, if the battery is full or if its charging rate is met, then the excess power goes to the flywheel. The battery is connected to the wind turbine and also has a priority zero connection to the utility. Thus, if the battery remains in a discharge state for more than a specified amount of time, it can change the utility priority (from 0 to 1) to receive the needed power.

Also in Figure 1-7, we see that load 2 prefers to draw power from the flywheel before turning to the battery. This configuration tends to keep the flywheel as discharged as possible, using it primarily as a means to absorb large influxes of power.

Figure 1-7 is a rather simple configuration used for illustrative purposes. A more complex configuration is shown in Figure 1-8.

1.2.4 SIMWEST Output

There are three basic forms of SIMWEST output to facilitate the analysis of wind energy storage systems; line printer plots, histograms of system variables and time sequenced output of variable values. To enhance the usefulness of these outputs each SIMWEST library component is associated with a number of output variables. Prior to simulating a given system the user specifies which of these variables he wants plotted. For plotted output he may select to have the independent variable be the plot time or any of the other variables. For example he may want to plot the energy of pneumatic storage as a function of time and/or as a function of storage versus temperature. If the user wants a time sequenced listing of all variable values

FIGURE 1-7 EXAMPLE OF POWER DIVIDER & ACCUMULATOR USE

FIGURE 1-8 PLUMBROOK CONFIGURATION FOR PARAMETER STUDY

he just specifies the time step between printouts. The listing of all variables has proven to be a useful tool in understanding the performance of the storage system under consideration and a valuable aid in validating the system design.

2.0 MODEL GENERATION

The Model Generation program design is based on the assumption that the system analyst will begin by constructing a schematic diagram of the system he wishes to analyze. This schematic will be comprised primarily of standard SIMWEST library components. Standard library components include wind turbine, wind models, AC induction motors, inverters, rectifiers, etc. If a particular system cannot be modeled with existing standard components, the analyst may construct the model by including appropriate FORTRAN statements in his system description.

All interconnections between standard components are accomplished by the Model Generation program. The analyst merely specifies each standard component in the schematic diagram and all of the components that provide inputs to that component. The Model Generation program then generates names and the proper interconnections between the specified components. This is accomplished by matching the input quantities required by each standard component to the output quantities of the specified input components.

After processing the complete system model description, the Model Generation program generates a schematic diagram of the model showing the interconnections between standard components and the quantities such as power, pressure, temperature, mass flow rates, etc., that pass through each interconnection. This schematic is produced on the lineprinter to provide a rapid graphic check on the program's interpretation of the model description.

In addition, the program produces a list of input data that will be required by each component to complete the model description. Both the scalar parameters and tabular data required for the analysis are included in this list. The program assumes that any quantity not supplied by another component will be supplied as a fixed parameter by the analyst. Thus requests for non-parameter items in the input data list will reveal any connection that was omitted from the system model description.

2.1 MODEL DESCRIPTION

The Model Generation program is a precompiler program which accepts model description instructions and from these instructions generates a FORTRAN model of a system. These instructions, referred to as "program commands," are made up of one or more words. In addition, the system model description contains numeric values, standard component names, and standard input and output quantity names.

The Model Generation commands may be best introduced with a simple example of their use to describe a wind turbine system. Figure 2.1-1 shows an analyst's schematic of a wind turbine model that has been constructed using standard components on a SIMWEST schematic form. The standard component names used in this sample are:

- WD - Wind Model
- WT - Wind Turbine
- TI - Time Conversion
- HG - Histogram Generator
- GR - Fixed Ratio Transmission
- LO - Electrical Load
- GE - AC Induction Generator

The SIMWEST description of this model would be as follows:

Example 2.1

LIST STANDARD COMPONENTS

MODEL DESCRIPTION	WIND TURBINE TEST CASE	
LOCATION=15	TI	
LOCATION=11	WD	INPUTS=TI
LOCATION=31	WT	INPUTS=WD

SIMWEST SCHEMATIC FORM

FIGURE 2.1-1 ANALYST'S SKETCH OF WIND TURBINE MODEL SCHEMATIC

Example 2.1 (Cont.)

```
LOCATION=35      GR      INPUTS=WT
LOCATION=51      HG      INPUTS=WT(P =F IN)
LOCATION=39      GE      INPUTS=GR
LOCATION=19      LO      INPUTS=GE,TI
END OF MODEL
PRINT
```

The model description consists of a statement as to the location of each component in the schematic and a list of all components that provide inputs to that component. The location of the component in the schematic is used for a line printer drawn schematic of the model, such as shown in Figure 2.1-2. In the line printer schematic the input and output quantities such as powers (P2 WT, P2 GE, P2 GR) are shown on the various connecting lines.

2.1.1 Phrases and Delimiters

The system model description is interpreted by the Model Generation program as a series of "phrases", which can appear in a free field format in any position on a data card. Phrases must be separated by any one of the delimiter symbols shown in Table 2.1-1.

Table 2.1-1

Model Generation Program Language Delimiters

- = equal sign
- , comma
- (left parenthesis
-) right parenthesis
- three or more blanks

WIND TURBINE TEST CASE

FIGURE 2.1-2 LINEPRINTER-DRAWN WIND TURBINE MODEL SCHEMATIC

2.1.2. Command Phrases

The Model Generation command phrases are described in this section in a logical sequence similar to that in which they appear in system model descriptions.

MODEL DESCRIPTION

The MODEL DESCRIPTION command phrase indicates the start of a new system model. This phrase may be followed, (on the same card), by a title of up to 60 characters. This title will be used to identify various program output schematics, lists and program listings. In Example 2.1, the title was "Wind Turbine Test Case."

LOCATION

The LOCATION command phrase indicates the start of the description of a new component in the system model. This command must be followed by a numeric value phrase that specifies the location of the new component on the model schematic. Thus in the example of Figure 2.1-1, the location number of the wind model WD was 11 and the wind turbine WT was 31, etc. To be a valid component location, the last two digits of this number must comprise a number between 1 and 80. The hundreds column is used to specify additional pages as needed for the schematic. Thus the numbers:

1, 13, 51, 80

would be valid location numbers for components on the first page, (PAGE 0), of a system schematic. These same locations on the second page of the schematic, (PAGE 1), would be:

101, 113, 151, 180

The location number phrase is followed by the name of the component at that location. Component names are discussed in Section 2.2.

A LOCATION statement should be given only once for each component. That is, once a LOCATION statement is started for a component the complete description of all inputs to that component should be given.

INPUTS

The INPUTS command phrase indicates that the following phrases contain the names of the components that provide inputs to the component at the specified location. Thus in the example of Figure 2.1-1, the electric load at location 19 which receives inputs from generator GE and the time source TI was described as:

```
LOCATION=19    LO    INPUTS=GE,TI
```

In this example the command phrase INPUTS is followed by two component names. As many component names as are necessary to specify the inputs to a particular system component may be included in each component description.

For some system components there are multiple input and/or output ports. For example, a power divider has four input power ports. When specifying the connections between such components, it is advisable to specify which ports are to be connected. This is done by adding the port numbers to be connected after the name of the input component. Thus, the wind turbine to transmission connection could have been more explicitly described as:

```
LOCATION=35    GR    INPUTS=WT(2,1)
```

This says that port 2 of the wind turbine (WT) drives port 1 of the transmission (GR). Any quantiles which have no port numbers are considered "universal ports" for input connections. Thus, the GR input of GR is connected

up to GR WT, and the RS input of WT is connected up to RSIGR by the above command. If the port designations are omitted, as they were in example 2.1., the connections will be made to the first available input port starting with the minimum port number. Once a connection has been made to an input port, those input quantities that are connected are unavailable for further connections. An exception is made when the physical quantities of both input and output are specified. This method of specifying connections is described in the following paragraphs.

For certain components, such as control elements, the inputs to the component can be any physical quantity in the model. For these components, the input component names must be supplemented by the name of the particular output quantity that is to provide the input.

As an example, consider a component that represents a linear first order lag transfer function. If the transfer function component's input, FIN, was to be the rotor speed of the wind turbine WT in example 2.1, then the statement:

```
LOCATION=53      LA      INPUTS=WT(RS=FIN)
```

would indicate to the program that of the outputs of the wind turbine, the output rotor speed, RS, was to be used as the input, FIN, to the transfer function, LA.

To summarize, there are three levels of connection specification:

1. Default (only component names are specified)

Connections are made between all unconnected inputs and outputs for the first ports for which a match of physical quantity names occurs.

2. Ports Specified

Connections are made between matching physical quantities for all unconnected inputs and outputs of the specified ports.

3. Physical Quantities Specified

Connections are made between only those quantities specified. Previous connections can be over-ridden, providing the three character physical quantity name of the previously connected variable is used. For example, the phrase

```
LOCATION=19      LO      INPUTS=GE,GE(P,2=MP2),TI
```

will replace the input parameter MP1LO by MP26E and then override the connection MP2GE and substitute P2 GE as the LO input.

END OF MODEL

The END OF MODEL command phrase indicates that model description has been completed and that the Model Generation program should proceed with the generation of the model subroutines.

PRINT

The PRINT command phrase causes the program to: (1) draw a schematic of the system model, as shown in Figure 2.1-2; (2) print a list of input requirements for the model; and (3) print a source listing of the FORTRAN subroutines that were generated for the model. The Model Generation program then terminates.

PUNCH

The PUNCH command phrase has the same effect as the PRINT command, but in addition a FORTRAN source deck of the system model is produced.

FORTRAN STATEMENTS

The FORTRAN STATEMENTS command phrase allows the system analyst to supplement the library components with FORTRAN statements. Using this feature, the analyst can introduce his own program logic, DO loops, etc., as necessary to model any system feature not obtainable with standard library components.

One of the common uses of the FORTRAN STATEMENTS command is to input large tables into the model. Two function subprograms TBLU1 and TBLU2 are provided for this use. They perform linear interpolation from one and two dimension tables, respectively. TBLU1 is in general called in the form

$$F = \text{TBLU1}(X, \text{TAB}(4), \text{TAB}(4+N), I, \pm N),$$

where F is the interpolated value at the desired point X, TAB is a one dimension table with dimension N, TAB(4) is the independent variable and TAB(4+N) is the dependent variable list, I = 0 for equal spaced data, I = 1 for unequal spaced data, and the dimension N is specified as the last variable if linear extrapolation is desired, and -N is specified if truncation is desired outside the table limits. Similarly, TBLU2 is in general called using the form

$$F = \text{TBLU2}(X, Y, \text{TAB}(4+M), \text{TAB}(4), \text{TAB}(4+M+N), IX, IY, \pm N, \pm M, N, M),$$

where X and Y are the values of the primary and secondary independent variables, N and M are the dimensions of the primary and secondary variable arrays, IX and IY are indicators for equal spaced or unequal spaced data as above, and the sign convention on N and M is positive for extrapolation, negative for truncation.

The FORTRAN STATEMENTS command would normally be used only when some portion of the system cannot be modeled with library components. Then using this feature of the program, the analyst must perform many of the detail connections

and naming of variables, that are normally accomplished by the Model Generation program. In return for these added tasks, the analyst gains a great deal of additional freedom and flexibility in forming details of his system model. Non-executable code such as common blocks must precede the first component definition and executable code should come after a component has been defined for the iteration logic to work properly.

ADD STATES
ADD VARIABLES
ADD PARAMETERS
ADD TABLES

The ADD commands may be used in conjunction with the FORTRAN STATEMENTS to add states, variables, parameters, and tables that occur within the FORTRAN statements, to the system model. Quantities that are not specified by one of these commands cannot be accessed or manipulated by the Analysis Program.

Before discussing these commands, a few definitions of terms are in order.

States:

States are those quantities in the system model that are described by first order differential equations. The state variables are the result of integrating the set of first order differential equations that comprise the dynamic system model. The number of states equals the order of the system model. The states are dynamic, time varying quantities during most simulation studies. The initial values, (initial conditions), of the states must be input as part of the system model description.

Variables:	Variables are all other dynamic time varying quantities in the system model that are not states. In general, variables are related to states by algebraic relationships.
Parameters:	Parameters are constant scalar quantities in the system model. Parameters can be manipulated by the analyst to alter the system model. All parameter values* should be input as part of the system model description.
Tables:	Tables are constant nonscalar quantities in the system model. Tables are used to represent algebraic functional relationships with one or two independent variables. All table values must be input as part of the system model description.

The format for the ADD commands is that the command is followed by one or more phrases that contain the names of the states, variables, parameters, or tables. In addition to each table name, a number, specifying the amount of storage to be allocated for that table must be given. This number is positive if the table is two dimensional and negative if one dimensional, with absolute value determined by the formula:

$$N = 3 + I + J + D$$

N = the total storage required by the table,
in words.

* For certain components, default values are provided for some parameters.

I = the number of data points in the primary independent variable table.

J = the number of data points in the secondary independent variable table. (J=0 if there is only one independent variable.)

D = the number of data points in the dependent variable table. (D=1 if there is only one independent variable. D={%J if there are two independent variables.

The following example illustrates the use of FORTRAN STATEMENTS in the parameter study model:

Example 2.2

MODEL DESCRIPTION PARAMETER STUDY

.
.
.

ADD TABLES = WIND,802

LOCATION = 41 TI

FORTRAN STATEMENTS

C READ WIND VELOCITY DATA

 WVWD = TBLU2(TD TI,DY TI,WIND(35),WIND(4),

 1 WIND(59),0,0,24,-31,24,31)

LOCATION = 71 WD INPUTS = TI

.
.
.

In this model, Fortran is used to input wind velocity data. The wind table, denoted WIND, consists of up to 31 days of hourly wind speeds. Hence, as described previously, the total storage required is $3+24+31+24*31=802$. The Fortran is inserted after time of day and day of the year are computed in T1. In this case, $N=24$, $M=31$, the data is equal spaced, and extrapolation is used to provide velocity data over each 24 hour period. The variable WIND is the name at the wind input to WD generated by the precompiler. Fortran insertion in the model ends when the LOCATION=71 ... command is read and a call to the subroutine WD is then generated.

LIST STANDARD COMPONENTS

The LIST STANDARD COMPONENTS command phrase causes the program to print a list of all standard components. For each standard component, lists of inputs, outputs, and tables for that component are provided. For each input, the physical quantity name and port number is given. For each output, the physical quantity name, port number, and the letter S, if the quantity is a state is given. For each table, the table name, the number of independent variables and the maximum amount of storage allowed is provided. This command is usually given as the first command of a model description and will result in a list of all standard component information as the first output from the Model Generation program.

2.2 NAMING CONVENTION

All standard components are given names consisting of two characters, the first of which is alphabetical. Thus we have WT for wind turbine, GE for generator, WD for wind model, etc. Where multiple components of the same type are required, the second character is used to distinguish between the different models of the same basic component type. A specific component in a model can be distinguished from other components of the same type by adding one

more character to the standard component name. This character is usually numeric but can also be alphabetical or blank. Thus a given model can contain up to 37 different components of the same standard component type. For example, a model with ten different wind turbines might have these components designated as:

WT1,WT2,WT3,.....,WT8,WT9,WT10

2.2.1 Variable, Parameter, and Table Naming Conventions

All of the input, output, and tabular quantities required by each component in a system model must have unique FORTRAN names. These quantities are given names consisting of up to three characters that describe the physical quantity they represent.

Since a single component may have several inputs or outputs of the same physical quantity, the program adds the port number to the second or third character of the physical quantity name to prevent such a duplication.

The physical quantities that are outputs of a given component are identified by adding the three character name of that component to the three character name of the physical quantity. In this way, unique six character FORTRAN names are generated for all output quantities of the system model components.

Input quantities to a component that are driven by another component carry the names of the component that drives them. Any inputs that are not driven by other model components are assumed to be parameters and are assigned the name of the component for which they are an input.

If a component should require tabular data as an input, unique table names are generated just as scalar input quantity names by adding the component

name to the table name. A pictorial representation of the character assignment in component, variable, and table names is given in Figure 2.2-1.

2.3 MODEL SCHEMATIC

The Model Generation program produces a schematic diagram of the system being modeled. This schematic is crude but is inexpensive and does not have the flow time delays associated with more elaborate plotting methods. Its purpose is to provide a means of rapidly locating errors in the model description.

In order to construct a schematic diagram in an efficient manner with a reasonable size program, it was necessary to establish some simple rules for symbol generation, component connection paths, and labeling. If these rules are kept in mind when laying-out a schematic for the system, the SIMWEST produced schematic will match that developed by the analyst. If the rules are violated by the analyst's schematic, the SIMWEST schematic should still be correct, but may contain some unusual component connection paths and some labeling information may be overwritten.

2.3.1 Standard Schematic Form

The SIMWEST schematic diagrams are produced on a standard 11" by 14" line-printer page with 80 component locations per page. A standard form containing only the location numbers can be obtained by executing the Model Generation program with the single program command, PRINT. This form can then be reproduced and the copies used as forms for drawing system model schematics.

2.3.2 Input Quantity Labeling

The names of the physical quantities that are input to one component from another component are listed adjacent to the downstream component symbol.

INPUT/OUTPUT OR TABLE NAMES

FIGURE 2.2-1 CHARACTER ASSIGNMENT INPUT/OUTPUT OR TABLE NAME

These labels are placed near the connecting line that joins the two components. Since these names are composed of the physical quantity name and the name of the component that generates the information, the source of the input is evident from the name itself. Parameter and tabular inputs to a component are not shown on the schematic. These constant inputs are described in the Input Requirements List.

2.3.3 Component Connection Paths

In order to keep the core requirements and run time of the SIMWEST schematic drawing subroutine small, it was necessary to limit the types of connecting paths between components to a few basic routes. These paths are shown in Figure 2.3-1. Connections between components on the same horizontal or vertical line are straightforward. However, connections between components that do not share a horizontal or vertical line require at least a two segment path. These paths have been arbitrarily chosen to follow a clockwise route. It is therefore advisable that components that are on diagonal locations be placed in a clockwise sequence. If counter-clockwise flow between components is necessary, it can be accommodated by placing the components on the same horizontal or vertical lines.

The SIMWEST schematic drawing subroutine makes no attempt to go around components that get in the way of a connection path. Such components are "run-over" by the connecting line.

2.3.4 Additional Pages

The SIMWEST schematic diagram may be broken down into as many pages as are necessary. No attempt is made to draw connecting paths between components located on different pages. It is therefore advisable to minimize the number of connecting paths between pages. This can usually be done by grouping

POSSIBLE OUTPUT PATHS

POSSIBLE INPUT PATHS

FIGURE 2.3-1 COMPONENT CONNECTION PATHS

components with many interconnections on the same page and placing page boundaries between such groups of components.

2.3.5 Guidelines for Schematic Layout

The following guidelines may help in creating schematic layouts that can be duplicated by the SIMWEST program.

1. Try to place connected components on the same horizontal or vertical line.
2. Avoid placing components on adjacent location points.
3. Place diagonal components so that flow is clockwise.
4. Group components to minimize flow paths between pages.

2.4 WARNING MESSAGES

One or more of the following warning messages will occur if the program is unable to interpret a portion of the model description or encounters problems in assembling the system model. These messages will be preceded by: *** WARNING *** or *** NOTICE ***. The symbols xxx and zzz are used to indicate phrases from the model description that are included as part of the warning message. The following messages are listed in alphabetical order:

1. CAN'T IDENTIFY xxx AS A STANDARD COMPONENT

xxx will contain the first two characters of the phrase which cannot be identified as a command or standard component. This message will often follow other warning messages as the program makes successive attempts to interpret the given phrase.

2. CAN'T IDENTIFY xxx AS A VALID INPUT COMPONENT TO zzz

The component xxx cannot be found in the list of components for the current system model.

3. CAN'T LOCATE xxx AS AN INPUT COMPONENT TO LOCATION n

This message indicates that the component xxx, which provides inputs to location n in the schematic, has not been assigned a location number. Check for a missing LOCATION statement or misspelling of the component name.

4. COMPONENT xxx DEFINITION WASN'T COMPLETED BEFORE STARTING THE DEFINITION OF COMPONENT zzz

The command INPUTS was not given between the component names xxx and zzz. Check for proper spelling of INPUTS and a valid delimiter after the phrase xxx.

5. COMPONENT xxx HAS ALREADY BEEN DEFINED

The component xxx was defined in a previous LOCATION statement.

6. LOCATION NO. xxx FOR COMPONENT zzz HAS LAST TWO DIGITS OUTSIDE THE ALLOWABLE RANGE 1 TO 80. NO SYMBOL WILL BE PLACED IN SCHEMATIC FOR THIS COMPONENT

This message will occur at the end of the model description for a component zzz which has an invalid location number. The system model may still be valid but the schematic will not contain this component.

7. NO xxx OUTPUTS MATCH UNSATISFIED zzz INPUTS

Check that it was intended to drive component zzz with component xxx or that the inputs to zzz have been previously satisfied by other component connections.

8. TABLE NAME xxx MUST BE FOLLOWED BY A NUMERIC DIMENSION RATHER THAN zzz

When using the ADD TABLES command, it is necessary to provide the maximum amount of storage to be allocated for the table as well as the table name. This storage value must be a numeric quantity.

9. xxx IS NOT A VALID INPUT QUANTITY OR PORT DESIGNATION FOR COMPONENT zzz

The phrase xxx cannot be located as one of the input quantities or input ports of the component zzz. No connections will occur. Check the list of standard components for the proper spelling or port designations for this component.

10. xxx IS NOT A VALID LOCATION NUMBER

The LOCATION command must be followed by a numeric location number.

11. xxx IS NOT A VALID PORT DESIGNATION FOR INPUT COMPONENT zzz. ERRONEOUS CONNECTIONS MAY OCCUR.

The phrase xxx cannot be located as a valid input port for the component zzz. Connections will be attempted using the upstream output port that was identified.

2.5 MODEL GENERATION LIMITATIONS

Certain limitations exist in the Model Generation program due to array dimensions within the program. For most applications these limits should not be encountered. However, if they should be encountered they can usually be extended at the expense of larger core requirements to execute the program. The following table describes these limitations:

<u>Limitation Description</u>	<u>Maximum Value</u>
Standard components in library	150
Components per model	200
Inputs per any standard component	50
Outputs per any standard component	50
Tables per any standard component	15
Tables per model	100
Table dimension (words)	960

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

3.0 SIMULATION PROGRAM

Once a model has been generated as described in Section 2, the user must describe the simulation he wishes to perform. This involves specifying the various parameters detailing the model components and setting the models initial conditions. It involves defining input data tables and the type and quantities of printed output, both tabular and plotted. The user must also specify the number of iterations he wishes to perform at each time step and the maximum number of component diagnostics. This section describes in detail the commands for specifying the simulation and gives some example output.

3.1 MODEL INPUT DATA

A dynamic system model requires that the values of numerous model parameters, tables and initial conditions, be provided to complete the model description. Sections 3.1.1, 3.1.2 and 3.2 describe the methods used to specify parameter values, tables, and initial conditions.

3.1.1 Scalar Data

PARAMETER VALUES (Default values = .99999)

This program command allows the numeric values of parameters to be loaded into the system model. The PARAMETER VALUES command is followed by one or more parameter names followed by a numeric value. Each name and its value are separated by one of the standard delimiter symbols. This command is used to specify the values of all system model parameters at the beginning of an analysis. It may also be used at any point between analyses to modify the value of one or more model parameters. A default value of .99999 is provided by the Model Generation program for all parameters not so specified.

Example 3.1-1

PARAMETER VALUES = CYCLES = 6.01, TO TI = 0, EC WP = .2,
CR CM = 15, LE CM = 30, MDEHS = 4.E5,

3.1.2 Tabular Data

If tabular data is required by the system model, it should be loaded before any of the simulation commands described in Section 3.4 are issued. Tables may be modified between analyses by loading new values. The tables required by a SIMWEST generated model are specified in the Input Requirements List. These tables may have either one or two independent variables. All data items are in a free field format with each item separated by one of the standard delimiters: comma [,], equal sign [=], left or right parenthesis () , or three or more consecutive blank spaces. The data items required for each table are placed on cards as follows:

Card 1	TABLE	table name	NX	NZ
Card 2*	Z	table values		
Card 3*	X	table values		
Card 4*	Y	table values		

where: Table Name - The six character table name generated by the Model Generation program.

NX - The number of points in the primary independent variable table.

NZ** - The number of points in the secondary independent variable table.

Z table ** - Table of NZ secondary independent variable values.

X table - Table of NX independent table values.

Y table - 1 or NZ tables of NX dependent variable values.

* As many cards as required may be used. Each table must start with a new card and NZ, NX, and NX*NZ points must be given per table.

** These items are omitted for tables with one independent variable.

A copy of all tabular input data is printed as it is interpreted from data cards. The following example shows the data cards for a one and a two independent variable table.

Example 3.1-2

Card 1	TABLE, TABONE, 10
Card 2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10
Card 3	11, 12, 13, 14, 15, 16, 17, 18, 19, 110
Card 4	TABLE, TABTWO, 5, 4
Card 5	10.3, 20.4, 30.5, 40.6
Card 6	1, 2, 3, 4, 5
Card 7	11, 12, 13, 14, 15
Card 8	21, 22, 23, 24, 25
Card 9	31, 32, 33, 34, 35
Card 10	41, 42, 43, 44, 45

The printout of these tables would be:

TABLE TABONE									
PRIMARY INDEPENDENT VARIABLE TABLE									
1,000	2,000	3,000	4,000	5,000	6,000	7,000	8,000	9,000	10,00
DEPENDENT VARIABLE TABLE									
11,00	12,00	13,00	14,00	15,00	16,00	17,00	18,00	19,00	110,00
TABLE TABTWO									
SECONDARY INDEPENDENT VARIABLE TABLE									
10,30	20,40	30,50	40,60						
PRIMARY INDEPENDENT VARIABLE TABLE									
1,000	2,000	3,000	4,000	5,000					
DEPENDENT VARIABLE TABLE									
11,00	12,00	13,00	14,00	15,00					
21,00	22,00	23,00	24,00	25,00					
31,00	32,00	33,00	34,00	35,00					
41,00	42,00	43,00	44,00	45,00					

3.2 INITIAL CONDITION AND INTEGRATION CONTROLS

INITIAL CONDITIONS (Default value = 0)

INT CONTROLS (Default value = 1.0)

These program commands may be used to specify initial condition values and the integrator status, (either active (=1) or frozen (=0)). Default values of 0, for initial conditions and 1 for integration controls are furnished by the simulation program. However, it is strongly recommended that values appropriate to the particular system model be furnished for the initial conditions.

Each of these commands is followed by phrases of the form of a state name followed by a numeric value.

Example 3.2-1:

INITIAL CONDITIONS = MA HS = 1.6E6, E TS = 600, VDELO = 0,

INT CONTROLS = MA HS = 0, E TS = 1, VDELO = 1,

ALL STATES (Default Condition)

NO STATES

These program commands may be used to activate or freeze all system integrators. These commands are normally used together with the INT CONTROLS command to specify the desired integrator configuration.

3.3 INITIAL CONDITION STORAGE COMMANDS

XIC-X
XIC-XIC1
XIC-XIC2
XIC-XIC3
XIC1-XIC
XIC2-XIC
XIC3-XIC

These program commands are used to transfer data from the current state vector, X, to the initial condition vector, XIC, and between the XIC vector and three auxiliary initial condition vectors XIC1, XIC2, XIC3.

Example 3.3-1

XIC1-XIC, XIC-X, XIC2-XIC

The three program commands shown above would take the current operating point (initial condition vector) and store it in vector XIC1; then transfer the current state, X, into XIC; and then store that value of XIC in XIC2.

3.4 SIMULATION COMMANDS

SIMULATE

This program command initiates simulation operation. Associated with this command are the program values:

<u>Default Values:</u>		
TINC	= time increment, hours	0.1
TMAX	= duration of the simulation run hours	1.0
OUTRATE	= output rate	1
PRATE	= print rate	1
PRINT CONTROL	= print control variable	0

These program commands specify the integration time increment, duration of simulation run, the simulation output rate, the printing rate, and the quantity of printing, at each point in time. These quantities should be specified before the first issuance of the SIMULATE command.

The Time increment, TINC, provides the integrator time step size, in hours, for the integrator. TINC also provides the report interval for which data will be available for printing or plotting. The default value for TINC is 0.1.

The duration of a simulation calculation in hours, is specified by the TMAX parameter. The default value of TMAX is 1.

The output rate parameter, OUTRATE, determines the sampling rate at which simulation data is added to plots. Thus, if OUTRATE is set equal to 10, data will be plotted every 10th time increment, TINC. The default value of OUTRATE is 1. OUTRATE should only be set to positive integer values.

The number of data samples plotted for a simulation analysis is thus given by:

$$\text{No. of Plotted Samples} = \frac{\text{TMAX}}{\text{TINC} \times \text{OUTRATE}} + 1$$

For most simulation operation, the plotted output specified by the DISPLAY commands is the primary output and no line printer output is used. However, for diagnosing problems in a simulation, the line printer options provided by the PRINT CONTROL parameter allow large amounts of detailed information about the simulated system to be obtained.

The value of the PRINT CONTROL parameter controls the quantity of data printed at each print report interval as shown in Table 3.4-1. Options 1 through 4 give "snap-shots" of all states, rates, variables, and parameters of the system model at a particular point in time. Option 5 provides tabular lists

of up to 10 specified quantities.* The default value for PRINT CONTROL is 0.

TABLE 3.4-1

Print Control Values	
PRINT CONTROL	Resultant Lineprinter Output.
0	None (Default Condition)
1	All states, rates, and time
2	All states, rates, variables, and time
3	All states, rates, variables, and parameters at time = 0
4	All states, rates, variables, and parameters
5	Time and the quantities specified via PRINT VARIABLES command.

The PRATE parameter determines the sampling rate at which the simulation data specified by the PRINT CONTROL parameter is presented on the line-printer. Thus, if PRATE is set equal to 5, data will be printed on the line printer every 5th time it is added to the output plots. The rate of output to the lineprinter can never be greater than that to the plots. The default value of PRATE is 1. PRATE should only be set to positive integer values.

The number of data samples printed for a simulation analysis is thus given by:

$$\text{No. of Printed Samples} = \frac{\text{TMAX}}{\text{TINC} \times \text{OUTRATE} \times \text{PRATE}} + 1$$

*See the PRINT VARIABLES command description below.

Example 3.4-1:

```
PRINT CONTROL = 2, TINC = .01, TMAX = 10.,  
OUTRATE = 10, PRATE = 10, SIMULATE
```

In the example, the simulation would run for 10 hours. Plotted output would occur every .1 hour, (10* .01), and printed output would occur every 1. hour (10* 10* .01).

PRINT VARIABLES

This program command allows up to ten variables to be specified for printing under option 5 of the PRINT CONTROL. This command is followed by from one to ten state, rate, or variable names separated by delimiters. This command wipes out all previously stored PRINT VARIABLES names.

Example 3.4-2:

```
PRINT VARIABLES = MA HS, E TS, VDELO
```

3.5 PLOT DESIGNATION COMMANDS

```
DISPLAY1  
DISPLAY2  
DISPLAY3  
DISPLAY4  
DISPLAY5  
DISPLAY6
```

These program commands may be used to define the quantities to be displayed by lineprinter plots for simulation calculations. These commands must be issued before the simulation analysis is requested. From one to five plots may be specified per display. Each plot is specified by stating the dependent variable and the independent variable separated by the letters VS. If

desired, the independent and dependent axis scale ranges can also be specified. The independent scale range is specified by the word **XRANGE** followed by the minimum and maximum values for this scale. The dependent scale similarly is specified by the word **YRANGE**. If scale ranges are not specified, values will be used that span the given data.

SI MANUAL SCALES

SI AUTO SCALES (Default Condition)

The **SI MANUAL SCALES** command allows the plotted output requested by the **DISPLAY** commands to be plotted on manual scales specified by the **YRANGE** and **XRANGE** commands. The **SI AUTO SCALES** command can be used to return plotting to the automatic scaling mode. Auto scales are selected so that they span each plotted quantity. The auto scale option is the default used until manual scales are requested. The **PRINTER PLOTS** command is also required to obtain plots.

Example 3.5-1:

```
SI MANUAL SCALES, PRINTER PLOTS
DISPLAY1
WV2WD, VS, TIME, YRANGE = 10,40
P1 PD, VS, TIME, YRANGE = 0,1000
P2 PD, VS, TIME, YRANGE = 0,1000
DISPLAY2
P2 IV, VS, TIME
RE2BA, VS, TIME
RE1LO, VS, TIME
DISPLAY3
P1 PD, VS, P2 PD, YRANGE = 0,1000, XRANGE = 0,1000
```

TITLE

The TITLE command allows a title to be placed on all plotted output. Up to 74 characters may follow the delimiter that follows the TITLE command. The TITLE command may be changed before each analysis. Once defined, the title remains in effect until a new title is entered.

Example 3.5-2:

```
TITLE = BATTERY TEST MODEL
```

3.6 ITERATION AND DIAGNOSTIC CONTROL

There are three built-in parameters in any SIMWEST model : CYCLES, DLINEs and RESET. These parameters are specified similar to component parameters using the PARAMETER VALUES command.

CYCLES controls the number of iterations through the model to obtain steady state. If $CYCLES \leq 0$, then only one pass is made through the model. If CYCLES is a positive Integer then the number of iterations through the model is equal to $CYCLES + 1$. If cycles is positive, but not an integer, then the number of iterations is equal to the smallest integer value exceeding cycles. A maximum of 20 iterations are permitted per time step. Most of the simple models of Section 8 require between four and six iterations per time step to attain steady state. A complex model with cascaded logic components may require more.

The task of finding the correct value for CYCLES is facilitated by the program printing at each time step all variables which have a greater than 5% change in value in the last iteration.

Since output statistics are only updated the last iteration, many of the variables printed indicating nonconvergence are just statistics, and as such should be ignored.

DLINES controls the amount of convergence related printout to be controlled as well as the amount of diagnostic printout put out by the library components. If DLINES > 0 then the total number of diagnostic printouts is no greater than DLINES. Figure 3.6 shows a typical section of diagnostic printout using DLINES > 0. If DLINES < 0 then only library component diagnostics are printed with no greater than - DLINES of output. Typically, DLINES = 100 is sufficient to catch most simulation errors per run.

PC3PA	NONCONVERGENCE, OLD VALUE=	4.209	NEW VALUE=	6.109
SR UT	NONCONVERGENCE, OLD VALUE=	177.720	NEW VALUE=	264.359
PC3PA	NONCONVERGENCE, OLD VALUE=	6.109	NEW VALUE=	7.054
SR UT	NONCONVERGENCE, OLD VALUE=	264.359	NEW VALUE=	312.718
FLYWHEEL KINETIC ENERGY	39.875 FALLS BELOW MINIMUM REQUIREMENT			40.000
FLYWHEEL CLUTCH LOSS	1.217 EXCEEDS DELIVERABLE POWER			.400
SP NO	NONCONVERGENCE, OLD VALUE=	553.569	NEW VALUE=	601.998
SPCFL	NONCONVERGENCE, OLD VALUE=	543.583	NEW VALUE=	590.798
PC3PA	NONCONVERGENCE, OLD VALUE=	7.054	NEW VALUE=	8.152
SR UT	NONCONVERGENCE, OLD VALUE=	312.718	NEW VALUE=	370.217

FIGURE 3.6 TYPICAL DIAGNOSTIC OUTPUT

RESET controls the initialization value for the random number generators if several simulations are run back to back. If RESET > 0 (Default) then the same random numbers are used for each simulation. If RESET ≤ 0 then the random numbers at the start of each simulation are obtained from the last value at the end of the previous simulation.

3.7 DEFINE COMMANDS

DEFINE STATES
 DEFINE RATES
 DEFINE PARAMETERS
 DEFINE VARIABLES

These program commands may be used to define the alphanumeric names that will be used to refer to states, rates, parameters, and variables. All system models formed by the Model Generation program have model related names generated for all states, variables, and parameters in the model. State variable derivatives, (Rates), are generated as R1, R2,... for all models. R1, R2, ... refer to the rates of the first, second,... states respectively. If it is desired to replace these machine generated names with other names, the DEFINE command may be used to substitute any eight character name of the analyst's choosing. These names are associated with the corresponding numeric quantities located in the labeled commons /CX/, /CXDOT/, /CP/, and /CV/. The appropriate location for each quantity is printed out along with the quantity name prior to each simulation. Each of these commands is followed by phrases containing the location numeric followed by an alphanumeric name with one to eight characters the first of which must be alphabetic.

Example 3.7:

```

DEFINE STATES
1 = PRESSURE, 2 = STROKE, 5 = VELOCITY, 7 = ANGLE

DEFINE PARAMETERS
5 = MASS, 35 = DCT AREA

DEFINE VARIABLES, 1 = T OUTLET, 2 = LIQ H2O

```

Note that the program commands, numeric values and alphanumeric names must be separated by delimiters which are: comma [,], equals [=], left parentheses ([), right parenthesis [)], or three or more consecutive spaces.

3.8 EXAMPLE OUTPUT

Figure 3.8 shows a sample of the output print format generated using PRINT CONTROL = 3. This sample is taken from the Wind Turbine and File Read run

ANALYSIS AND FLIGHT EVALUATION

DATE 072077

PAGE 9

TIME 0 .0000		STATES	
1 VDELU	0 .00000		
TIME 0 0.0000		RATES	
1 M1	0 0.0000		
VARIABLES			
1 F T1	0 .00000	2 TO T1	0 .00000
6 MT T1	0 1.0000	7 MY T1	0 1.0000
11 M TAN	0 .45000	12 MVZNO	0 18.477
16 IIRNO	0 -1.0000	17 P2 MT	0 241.84
21 MAPMT	0 795.36	22 PT MT	0 12000
26 IU GN	0 995.48	27 PL GN	0 12.104
31 SAMNG	0 1.0000	32 AV NG	0 241.84
36 M3 GE	0 1005.2	37 PL GE	0 12.224
41 SM GE	0 27.114	42 YANTAL	0 572.02
46 MELLU	0 936.24	47 LUZLU	0 572.02
51 FIMLU	0 1.0000	52 EN LU	0 1.4575
3 TM T1	0 .00000	4 DM T1	0 1.0000
8 YANTAN	0 18.477	9 N TAN	0 446.08
13 MY MD	0 18.477	14 AV MD	0 20.354
18 TO MT	0 72000	19 CO MT	0 12500
23 MPUNT	0 241.84	24 SP MT	0 30.230
28 LFZGN	0 94941	29 MPZGN	0 10000+11
33 SO HG	0 56005	34 P2 GE	0 217.61
38 LFZGE	0 69900	39 MPZGE	0 750.00
43 N TAL	0 446.08	44 TO TAL	0 .00000
48 SRELU	0 71.502	49 VDELU	0 27.170
50 DY T1	0 1.0000	51 TO TAN	0 .00000
52 M MD	0 58.844	53 GR MT	0 74.341
54 P2 GN	0 224.65	55 P2 GE	0 97710
56 M3GR	0 23.631	57 EE GE	0 28991
58 MPNGE	0 25000	59 M TAL	0 45000
60 PC LO	0 50.011		
PARAMETERS			
1 TO T1	0 .00000	2 NSTTAA	0 .00000
6 M3LAT	0 1000.0	7 SM MT	0 60.000
11 LPMAT	0 .41000	12 CM MT	0 .99999
16 MP1LU	0 11000+11	17 CC GN	0 1000.0
21 F3 M3	0 .00000	22 F4 HG	0 1.0000
26 P8 HG	0 .00000	27 F9 HG	0 .00000
31 F13HG	0 .00000	32 F14HG	0 .00000
36 FUPH	0 1000.0	37 FLUNG	0 50.000
41 GA GE	0 .20000	42 GR GE	0 .50000+02
46 NSTIAL	0 .00000	47 ITFTAL	0 2.0000
51 SFDLU	0 0.0000	52 VE LU	0 2.3000+01
3 LFTAM	0 1.0000	4 VO MT	0 21.000
8 LC MT	0 1.0000	9 AD MT	0 23000+02
13 CC MT	0 10000	14 CM MT	0 2000.0
18 CM GN	0 200.00	19 F1 HG	0 .00000
23 F5 HG	0 .00000	24 F4 HG	0 .00000
28 FLUNG	0 .00000	29 F13HG	0 .00000
33 F13HG	0 .00000	34 F14HG	0 .00000
38 NMPGE	0 750.00	39 M3YGE	0 1000.0
43 VO GE	0 400.00	44 CC GE	0 1000.0
48 MC LO	0 57400+02	49 CT LO	0 4.0000
53 CYCLES	0 2.0100	54 ULINES	0 .99999
55 V8 MT	0 28.150	56 LAMMT	0 9.4000
57 EF1GN	0 .99999	58 P2 MG	0 .00000
59 PT NG	0 .00000	60 F12MG	0 .00000
61 FA MG	0 67.857	62 RASGE	0 10000+01
63 CM GE	0 200.00	64 NM LU	0 .00000
65 RESLT	0 .99999		
TIME 0 12.00		STATES	
1 VDELU	0 97.200		
TIME 0 0.0123		RATES	
1 M1	0 0.0123		
VARIABLES			
1 F T1	0 12.000	2 TO T1	0 12.000
6 MT T1	0 1.0000	7 MY T1	0 1.0000
11 M TAN	0 .45000	12 MVZNO	0 13.444
16 IIRNO	0 -1.0000	17 P2 MT	0 30.804
21 MAPMT	0 795.36	22 PT MT	0 23555+04
26 IU GN	0 137.67	27 PL GN	0 1.0000
31 SAMNG	0 47.000	32 AV NG	0 30004
36 M3 GE	0 1000.7	37 PL GE	0 7.7533
41 SM GE	0 4234.0	42 YANTAL	0 1024.2
46 MELLU	0 1002.0	47 LUZLU	0 1024.2
51 FIMLU	0 1.0000	52 EN LU	0 1.4575
3 TM T1	0 12.000	4 DM T1	0 1.0000
8 YANTAN	0 13.444	9 N TAN	0 446.08
13 MY MD	0 13.444	14 AV MD	0 20.354
18 TO MT	0 18454	19 CO MT	0 12.125
23 MPUNT	0 795.36	24 SP MT	0 4061.1
28 LFZGN	0 40594	29 MPZGN	0 10000+11
33 SO HG	0 24270+08	34 P2 GE	0 27.445
38 LFZGE	0 74853	39 MPZGE	0 750.00
43 N TAL	0 446.08	44 TO TAL	0 .00000
48 SRELU	0 8401.5	49 VDELU	0 4236.0
50 DY T1	0 1.0000	51 TO TAN	0 .00000
52 M MD	0 58.844	53 GR MT	0 74.341
54 P2 GN	0 224.65	55 P2 GE	0 97710
56 M3GR	0 23.631	57 EE GE	0 28991
58 MPNGE	0 25000	59 M TAL	0 45000
60 PC LO	0 50.010		
TIME 0 24.00		STATES	
1 VDELU	0 439.05		

FIGURE 3.8 SAMPLE PRINTED OUTPUT.

described in Section B.1, which is a very simple model. At each print time the output quantities are indexed by number and component name as they occur in the model. For example, first all the variables for component TI are printed, then all variables for component TAW, etc. The parameter values at time = 0 show both the input values and the default parameters. After T = 0 only the states, rates, and output variables are printed. Since all the model connection variables and output variables are printed, this mode is especially valuable for program debugging and analysis at a fixed time. The printer plots, samples of which are shown in Section 8, are useful for monitoring the time behavior of critical parameters such as energy in storage and percent of load delivered by storage.

4.0 JOB CONTROL PROCEDURES

In this section, we describe job control procedures for running and maintaining the SIMWEST programs. For the convenience of the user, a number of procedure files have been set up which simplify the user control cards required. In Section 4.1, we describe the control cards for executing the model generation and analysis programs. Section 4.2 describes the procedures to maintain the programs and update the component library.

4.1 MODEL GENERATION AND ANALYSIS EXECUTION

Figure 4.1-1 shows an overview of the program structure to execute a simulation run. The program FILoad is only executed when the component library is updated, and is thus described in the next section. The user input data for the model generation program is put on a file called EASYCARDS. A procedure file called XQTEASY is then used to generate the model Fortran and compile this model. Similarly, the user input data for the analysis program is put on a file called NONSIMCARDS, and a file called XQTANALYSIS maps the relocatable elements into absolute file elements, and executes both the simulation and printer plot programs.

A job control stream to execute these programs in a batch environment is given by:

```
@RUN ...  
@DELETE,C EASYCARDS.  
@ASG,UP EASYCARDS.  
@DATA,IL EASYCARDS.  
.  
.  
.  
INPUT DATA DECK  
FOR MODEL  
.  
.  
.  
@END  
@ASG,A XQTEASY.
```

*SCRTCH9 IS FORTRAN SOURCE CODE OUTPUT

FIGURE 4.1-1 SIMWEST PROGRAM EXECUTION STRUCTURE


```

@ADD,PL XQTEASY.
@DELETE,C NONSIMCARDS.
@ASG,UP NONSIMCARDS.
@DATA,IL NONSIMCARDS.
.
.
.
INPUT DATA DECK
FOR ANALYSIS
.
.
.
@END
@ASG,A XQTANALYSIS.
@ADD,PL XQTANALYSIS.
@FIN

```

The job control procedures XQTEASY and XQTANALYSIS are shown in Figures 4.1-2 and 4.1-3. If a user is creating data inputs from a terminal, then it may be somewhat simpler to create new job control procedures similar to XQTEASY and XQTANALYSIS, but substituting his data input file names for EASYCARDS and NONSIMCARDS, respectively. If the same model is used for a series of runs, then only the analysis program is required for execution. However, it is safer and also relatively inexpensive to execute both programs when using the above job stream. Whenever the file read component is desired, the user must either substitute his file for F1 or F2, or add the following job cards to XQTANALYSIS:

```

@ASG,A MYFILE.
@USE M, MYFILE.

```

where MYFILE is the user time history file and M is a unit number between 13 and 18. (See 7.38 for a discussion of the tape/file read component.)

4.2 PROGRAM MAINTENANCE AND LIBRARY UPDATES

Whenever the component library is updated, the user must compile the Fortran code and run the FILoad program to furnish the model generation program com-

```

@HDG SIMWEST MODEL GENERATION
@ASG,AX MGABS.
@ASG,A M18.
@USE 18,M18.
@ASG,T M7.
@USE 7,M7.
@ASG,T SCRTCH8.
@USE 8,SCRTCH8.
@DELETE,C SCRTCH9.
@ASG,UP SCRTCH9.
@USE 9,SCRTCH9.
@ASG,T SCRTCH10.
@USE 10,SCRTCH10.
@ASG,T SCRTCH11.
@USE 11,SCRTCH11.
@ASG,T SCRTCH12.
@USE 12,SCRTCH12.
@ASG,A EASYCARDS.
@USE 5,EASYCARDS.
@XQT MGABS.EASY
@ASG,AX ASRO.
@ASG,AX ASSI.
@ADD,PL 9.
@FREE 18.,7.,8.,9.,10.,11.,12.

```

FIGURE 4.1-2 XQTEASY JOB CONTROL FILE

```

@HDG SIMWEST ANALYSIS
@ASG,AX MAPANALYSIS.
@ADD,PL MAPANALYSIS.
@ASG,AX ASABS.
@ASG,AX F1.
@USE 11,F1.
@ASG,AX F2.
@USE 12,F2.
@ASG,T SCRTCH25.
@USE 25,SCRTCH25.
@DELETE,C SCRTCH26.
@ASG,UP SCRTCH26.
@USE 26,SCRTCH26.
@ASG,AX NONSIMCARDS.
@USE 5,NONSIMCARDS.
@XQT ASABS.NONSIM
@XQT ASABS.NSMPPT
@FREE 11.,12.,25.,26.

```

FIGURE 4.1-3 XQTANALYSIS JOB CONTROL FILE

ponent input and output name lists. A job control stream to compile a new component denoted DC and add it to the component library is given by:

```

@ASG,T SOURCE.
@DATA,IL SOURCE.
@FOR,IS COSI,DC,CORO.DC
.
.
.
USER FORTRAN SUBROUTINE DC
.
.
.
@END
@ASG,A COSI.
@ASG,A CORO.
@ADD,PL SOURCE.
@ASG,A CMPLCO.
@ED,U CMPLCO.
ADD SOURCE.
EXIT

```

If an old component is to be updated, then one can edit the source code on CMPLCO and recompile entirely, or copy the edited subroutine including the @FOR,IS control card onto a new file and recompile. A job stream to execute the FILOAD program is given by:

```

@DELETE,C FILOADCARDS.
@ASG,UP FILOADCARDS.
@DATA,IL FILOADCARDS.
.
.
.
USER INPUT DATA
FOR FILOAD
.
.
.
@END
@ASG,A XQTFILOAD.
@ADD,PL XQTFILOAD.

```

The procedure file XQTFILOAD is shown in Figure 4.2-1. If a mistake is made in the input cards, it is possible that the output file M18 will be destroyed. As a backup, M18 is copied onto M18X before FILOAD is run.

```

@HDS FILOAD EXECUTION
@ASG,AX FILOAD4.
@DELETE,C M19.,M18X.
@ASG,UP M19.,F/O/TRK/1000/      . FILOAD DATA FILE
@ASG,UP M18X.,F/O/TRK/1000/
@ASG,AX DUMPF9.
@ASG,AX M18.
@COPY M18.,M18X.
@USE 18,M18.
@USE 19,M19.
@USE 9,DUMPF9.
@ASG,AX FILOADCARDS.
@USE 3,FILOADCARDS.
@XQT FILOAD4.
@COPY 19.,18.
@FREE 18.,19.,9.

```

FIGURE 4.2-1 XQTFILOAD JOB CONTROL FILE

It may be necessary to modify the Fortran code for one or more of the SIMWEST programs from time to time. In this case, the procedure is to edit the source file, recompile the file, and remap the relocatables to obtain absolute file elements. Table 4.2 gives the file names for using this procedure.

TABLE 4.2 SIMWEST MAINTENANCE FILE DIRECTORY

	<u>LIBRARY UPDATE</u>	<u>MODEL GENERATION</u>	<u>ANALYSIS</u>	<u>PRINTER PLOTS</u>
FORTRAN SOURCE FILES - SDF	UTIL FILOAD	CMPLNG CMPLNG2	CMPLAS CMPLAS2	NSMPPT
FORTRAN SOURCE FILES - ELT	FSSJ	MGSJ	ASSJ	ASSJ
RELOCATABLE FILES	FSRO	MGRO	ASRO	ASRO
COLLECTION PROCEDURES	MAPFILOAD	MAPEASY	MAPANALYSIS	MAPNSMPPT

The collection procedures are shown in Figures 4.2-2 to 4.2-5. As an example, the jobstream to edit, recompile and remap a subroutine in OMPLNG is given by:

```

@ASG,A CMPLNG.
@ED,U CMPLNG.
.
.
.
FORTRAN EDITING
.
.
.
@ASG,A MGS1.
@ASG,A MGRO.
@ADD,PL CMPLNG.
@ASG,A MAPEASY.
@ADD,PL MAPEASY.

```

```

@ASG,AX FSRO.
@PREP FSRO.
@ASG,AX MAPFSS1.
@ASG,AX FILOAD4.
@MAP,1 MAPFSS1,FILOAD,FILOAD4.
  IN FSRO,FILOAD
  LIB FSRO.
END

```

FIGURE 4.2-2 MAPFILOAD PROCEDURE FILE

```

@ASG,AX MGRO.
@PREP MGRO.
@ASG,AX MAPMGSI.
@ASG,AX MGABS.
@ASG,AX FSRO.
@PREP FSRO.
@MAP,1 MAPMGSI,EASY,MGABS,EASY
  IN MGRO,EASY
  LIB FSRO.,MGRO.,FSRO.
END

```

FIGURE 4.2-3 MAPEASY PROCEDURE FILE

```

@ASG,AX FSRO.
@ASG,AX ASRO.
@ASG,AX CORO.
@PREP CORO.
@PREP FSRO.
@PREP ASRO.
@ASG,AX MAPASS1.
@ASG,AX ASABS.
@MAP,1 MAPASS1,NONSIM,ASABS,NONSIM
  IN ASRO,NONSIM
  IN ASRO,BLOCKDA
  IN ASRO,MODEL
  LIB FSRO.,ASRO.,FSRO.,CORO.
END

```

FIGURE 4.2-4 MAPANALYSIS PROCEDURE FILE

```

@ASG,AX FSRO.
@ASG,AX ASRO.
@PREP FSRO.
@PREP ASRO.
@ASG,AX MAPASS!.
@ASG,AX ASABS.
@MAP,1 MAPASS!.NSMPPT,ASABS.NSMPPT
  IN ASRO.NSMPPT
  LIB FSRO.,ASRO.,FSRO.
END

```

FIGURE 4.2-5 MAPNSMPPT PROCEDURE FILE

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

5.0 DIAGNOSTICS

Diagnostic messages are printed by both the model generation and the simulation program. In addition, individual library components generally have diagnostic printout associated with them. The diagnostics associated with the model generation program are discussed in Section 2. Section 5 describes the diagnostics associated with the simulation program and the library components.

5.1 WARNING MESSAGES

One or more of the following warning messages will occur if the program encounters difficulty in interpreting analysis instructions or performing an analysis. These messages will be preceded by the flag:

*** WARNING ***

Message symbols xxx, zzz, or nnn are used to indicate phrases from the analysis description that are included as part of the warning message. The following messages are listed in alphabetical order:

1. A VALID PARAMETER NAME MUST PRECEDE THE NUMERIC VALUE nnn

This message indicates that a valid parameter name was not identified preceding the numeric value nnn. Check for missing delimiters or misspelled parameter name.

2. xxx CAN'T BE SET EQUAL TO zzz. VALUE MUST BE NUMERIC

Check for missing numeric value or delimiters.

3. CAN'T IDENTIFY xxx AS A VALID PRINT VARIABLE

Check spelling of xxx or for missing delimiters.

4. CAN'T IDENTIFY xxx VALUE WILL BE IGNORED.

This will result in not setting the quantity intended by xxx to its new value. Check for spelling of xxx or for missing delimiters.

5. CAN'T INTERPRET xxx

The phrase xxx cannot be recognized as a valid program command, program name, or program value. Check spelling of xxx or for missing delimiters.

6. nnn EXCEEDS THE ALLOWABLE INDEX RANGE FOR xxx THIS QUANTITY WILL NOT BE DEFINED

The number nnn was outside the allowable range of states, rates, variables, or parameters. Therefore, the name xxx cannot be assigned as a name for the nnnth state, rate, variable or parameter.

7. NON-ALPHA NAME ON THIS CARD --- xxx. WILL IGNORE THIS CARD.

The table inputs routine expected an alphanumeric table name but encountered a numeric value on the data card printed. Check the sequence and number of tabular data cards to assure that they match those required by the model's tables and table input formats. See Section 3.1.2 for correct formats.

8. NON-NUMERIC DATA ON THIS CARD --- xxx. WILL READ NEXT TABLE

The table input routine expected a numeric value but encountered an alphanumeric name on the data card printed. Check that the sequence and number of tabular data cards matches the model's tables and table input formats. See Section 3.1.2 for correct formats.

9. nnn PRIMARY and xxx SECONDARY INDEPENDENT VARIABLE POINTS EXCEEDS THE zzz WORD STORAGE LIMIT FOR THE FOLLOWING TABLE. SOME DATA WILL BE LOST.

The maximum amount of data allowed for each table is given in the Input Requirements List produced by the Model Generation program. Check that given data falls within this limit or for data card errors.

5.2 DIAGNOSTIC MESSAGES FOR LIBRARY COMPONENTS

A diagnostic message associated to a component is printed when a variable gets out of bounds during analysis. Adjustment of component parameters may be necessary.

In component alphabetical order, these diagnostic messages are:

- AD: INPUT POWER xxxx TOO HIGH RELATIVE TO ADMITTANCE xxxx AND RATED VOLTAGE
xxx
ADMITTANCE POWER LOSS xxxx EXCEEDS INPUT POWER xxxx
- BA: POWER REQUEST xxxx EXCEEDS BATTERY CAPABILITY. CHECK VC, VO, AND RT.
- BN: BN INLET AIR MASS FLOW RATE xxxx GREATER THAN MAXIMUM ALLOWABLE xxxx
- CO: MAX ITERATIONS FOR COMPRESSOR EFFICIENCY. NP, XNP, RS = xxxx, xxxx,
xxxx
- CS: CS STORAGE TEMPERATURE xxxx GREATER THAN ALLOWABLE xxxx
CS MASS OF AIR IN STORAGE xxxx BELOW MINIMUM ALLOWABLE xxxx
CS MASS OF AIR IN STORAGE xxxx EXCEEDS MAXIMUM ALLOWABLE xxxx
- FL: FLYWHEEL POWER LOSS xxxx EXCEEDS CHARGING POWER xxxx
FLYWHEEL LOSS xxxx EXCEEDS DISCHARGING POWER xxxx
FLYWHEEL CLUTCH LOSS xxxx EXCEEDS MAXIMUM INPUT POWER xxxx
FLYWHEEL CLUTCH LOSS xxxx EXCEEDS DELIVERABLE POWER xxxx

FLYWHEEL KINETIC ENERGY xxxx EXCEEDS CAPACITY xxxx
FLYWHEEL KINETIC ENERGY xxxx FALLS BELOW MINIMUM REQUIREMENT xxxx

GE: GENERATOR OUTPUT EXCEEDS RATED POWER

HS: HS INLET MASS FLOW RATE xxxx OR OUTLET MASS FLOW RATE xxxx IS GREATER
THAN MAXIMUM xxxx
HS RESERVOIR VOLUME xxxx EXCEEDED MAXIMUM ALLOWABLE xxxx
HS RESERVOIR VOLUME xxxx DROPPED BELOW MINIMUM xxxx

HT: HT TURBINE CHARACTERISTIC PARAMETER OUT OF RANGE
HT INLET MASS FLOW RATE xxxx GREATER THAN MAXIMUM DESIGN VALUE

HX: HX EXIT TEMPERATURE xxxx GREATER THAN MAXIMUM ALLOWABLE xxxx

IV: IV POWER LOSS xxxx EXCEEDS INPUT POWER xxxx CHECK RATED DC VOLTAGE VDC

MB: WARNING-DIVISOR IN MB EQUALS 0., HAS BEEN SET = 1.

MO: MOTOR INPUT POWER xxxx ,GT. RATED INPUT POWER xxxx
MOTOR SLIP xxxx EXCEEDS RATED POWER SLIP xxxx
STATOR RESISTANCE xxxx OR DAMPING xxxx TOO HIGH FOR MOTOR

RE: RE POWER LOSS xxxx EXCEEDS INPUT POWER xxxx
RE, AC INPUT POWER xxxx TOO LARGE IN RELATION TO TRANSFORMER REACTANCE
xxxx AND RATED AC VOLTAGE xxxx

TA: FILE DATA OUT OF RANGE. INITIAL VALUE = xxxx ON UNIT xx
TIME POINT PAST TABLE RANGE. LAST VALUE = xxxx ON UNIT xx
READ ERROR OR END OF FILE ON UNIT xx

TR: TRANSMISSION POWER LOSS xxxx EXCEEDS INPUT xxxx

TRANSMISSION POWER LOSS xxxx EXCEEDS MAXIMUM INPUT POWER

TS: TS WORKING FLUID FLOW RATE xxxx GREATER THAN MAXIMUM ALLOWED xxxx

TS INPUT POWER xxxx GREATER THAN MAXIMUM ALLOWED CHARGE RATE xxxx

TS STORAGE TEMPERATURE xxxx OUTSIDE MINIMUM xxxx OR MAXIMUM xxxx

TU: TURBINE BACK PRESSURE xxxx GREATER THAN STORAGE VESSEL PRESSURE xxxx

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

6.0 CREATION OF NEW LIBRARY COMPONENTS

The addition of new standard components to the SIMWEST library involves two steps. The first is the design of the component. This design must conform to certain design conventions if the new component is to be compatible with existing components. Section 6.1 discusses these design conventions and the addition of the component subroutine to the SIMWEST library. The second step involves the addition of the new component's input and output description to the SIMWEST file M18. File M18 is used by the precompiler to generate subroutine calling sequences for the library components. Section 6.2 discusses the use of the FILOAD program to accomplish this task.

6.1 LIBRARY COMPONENT CODING

6.1.1 Component Call Sequence

The items in the component subroutine call sequence must be arranged in the following order:

1. Tables
2. Output Quantities
3. Input Quantities

Tables or inputs may not be present in the subroutine call sequence. However those items that are present must follow the sequence given above.

Dummy argument names for the call sequence quantities that are used within each subroutine should be chosen to match the physical quantity names placed in the input, output, and table name lists. Exceptions to this policy may be made when integer names (names starting with I through N) must be avoided or when additional letters will clarify the name.

The subroutine name must contain only two characters and must not duplicate the name of an existing standard component.

Tables

The table arrays must be dimensioned within the component subroutine. They must be dimensioned with only one subscript; e.g. `DIMENSION TABLE (1)`. When table data is passed to the component subroutine, the first word in the array contains the name of the table. The second word contains the number of values given for the primary independent variable. The third word contains the number of values given for the secondary independent variable. Both of these numbers are stored as REAL quantities and must be converted to INTEGER before they can be used as a subscript. This can be done by a statement such as:

```
NX = TABLE (2) - number of primary independent variables
NZ = TABLE (3) - number of secondary independent variables
```

If there is a secondary independent variable, the secondary independent variable array will begin with the fourth word in the array. Thus if this array is designated as `z(1), z(2), ...,` then:

```
z(1) = TABLE (4)
z(2) = TABLE (5)
z(3) = TABLE (6)
.      .
.      .
.      .
```

The primary independent variable array begins with word `NZ + 4`. Thus if this array is designated as `X (1), X(2), ... ,` then:

```
X(1) = TABLE (NZ + 4)
X(2) = TABLE (NZ + 5)
.      .
.      .
.      .
```


The dependent variable array begins with word $NX + 4$ if there is no secondary independent variable. Thus if this array is designated as $Y(1), Y(2), \dots$, then:

$Y(1) = \text{TABLE } (NX + 4)$

$Y(2) = \text{TABLE } (NX + 5)$

• •
• •
• •

If there is a secondary independent variable array and this array was designated $Y(I, J)$, with $1 \leq I \leq NX$ and $1 \leq J \leq NZ$, then $Y(I, J)$ would be related to the table array as:

$Y(I, J) = \text{TABLE } (NX + NZ + 3 + I + (J - 1) * NX)$

Normally the individual elements in the table are not used directly but are passed to a table look-up routine. In this case the starting address of the X, Y, and Z tables would be referred to as:

$Z(1) = \text{TABLE } (4)$ secondary independent variable table

$X(1) = \text{TABLE } (NZ + 4)$ primary independent variable table

$Y(1,1) = \text{TABLE } (NX + NZ + 4)$ dependent variable table

If more than one table is used by a component subroutine, the table names must appear in the same sequence in the table name list stored in MIB file as in the subroutine call sequence.

Example 6.1: Given a component, HA, that requires the tables TPH and TPC as an inputs. The call sequence of this subroutine would appear as:

SUBROUTINE HA(TPH, TPC, ...

Output Quantities

The term "output quantity" refers to information that is calculated and then "output" by a particular component subroutine. This is not to be confused with the "outlet quantities" of the component. The outlet quantities are associated with a particular component port as a result of assigning a positive direction of power or information flow through the component. Some outlet quantities may be calculated by the component subroutine and thus become output quantities of that component. While other outlet quantities may be furnished to the component subroutine and thus become input quantities to that subroutine.

The output quantities should be grouped together by port. That is, all outlet, (port two quantities), then all inlet, (port one quantities), etc. If a component has multiple outlet ports, the output quantities associated with each outlet port should be grouped together and listed before any inlet port output quantities.

Certain output quantities may be internal to the component and not associated with any port. In other cases the same output quantity may be associated with several ports. In such cases, no port designation is assigned to the output quantity. Such quantities are referred to as "universal port" quantities. As such, they are allowed to connect to any other similar physical quantity regardless of the input quantities port number. This is not the case for quantities with specified port numbers. Once a connection has been made between an input and output quantity with given port numbers, only connections of matching physical quantities with those port numbers occur. Manual override of this provision can be made by specifying particular physical quantity connections.

Three quantities are required for each state variable output. The first is the state variable, the second is the state variable derivative, (rate),

and the third is an integer quantity, the integrator control variable.

Example: Given a component, HA, with the following outputs:

Physical Quantity	Port No.	
T	3	} Outlet Ports
T	4	
P	1 (State Variable)	} Inlet Ports
P	2 (State Variable)	

The call sequence arguments for these outputs would be:

SUBROUTINE HA(TPH,TPC,T3,T4,P1,P1DOT,IP1,P2,P2DOT,IP2,...

Input Quantities

The term "input quantity" refers to information that is provided to a particular component subroutine. This is not to be confused with the "inlet quantities" of the component. The inlet quantities are associated with a particular component port as a result of assigning a positive direction of power or information, through the component. Some inlet quantities may be calculated by the component subroutine and thus become output quantities of that component, while other inlet quantities may be furnished to the component subroutine and thus become input quantities to that subroutine.

The input quantities should be grouped together by port. That is, all inlet, (port one quantities), then all outlet, (port two quantities), etc. Port designations for two port components which have the same physical quantity on both inlet and outlet will be: port 1 for upstream or inlet port and port 2 for downstream or outlet port. It is important that the inlet port quantities be listed before any outlet port quantities. If a component has multiple inlet ports, the input quantities associated with each inlet port should be grouped together and listed before any outlet port quantities.

Certain input quantities may be internal to the component and not associated with any port. In other cases the same input quantity may be associated with several ports. In such cases, no port designation is assigned to the input quantity. Such quantities are referred to as "universal port" quantities. As such, they are allowed to connect to any other similar physical quantity regardless of the output quantity's port number. This is not the case for quantities with specified port numbers. Once a connection has been made between an input and output quantity with given port numbers, only connections of matching physical quantities with those port numbers occur. Manual override of this provision can be made by specifying particular physical quantity connections.

Example: Given the component HA described in the above example, with the following inputs:

Physical Quantity	Port No.	
T	1	Inlet Ports
T	2	
P	3	Outlet Ports
P	4	
AKH	(universal port quantity)	

The call sequence for these inputs would follow the output arguments, giving the complete call sequence:

```
SUBROUTINE HA(TPH,TPC,T3,T4,P1,P1DOT,IP1,P2,P2DOT,IP2,T1,T2,P3,P4,AKH)
```

The call sequence for standard component subroutines should follow the order shown in Table 6.1-1.

6.1.2 Additions and Modifications to Component Library

Section 4.2 describes the job control procedures to add a new component to the component library, compile the source code that describes the new component and add the relocatable binaries to the component library CORO.

TABLE 6.1-1

COMPONENT SUBROUTINE
CALL SEQUENCE ORDER

1. Tables
2. Output Quantities
 - 2.1 All Outlet Port Quantities*
 - 2.2 All Inlet Port Quantities* (feedback variables)
 - 2.3 All Other Output Quantities
3. Input Quantities
 - 3.1 All Inlet Port Quantities*
 - 3.2 All Outlet Port Quantities* (feedback variables)
 - 3.3 All Other Input Quantities

* Group quantities with the same port number together. If multiple inlet or outlet ports exist, arrange port quantities in order of increasing port numbers.

6.1.3 Coding Conventions

There are several coding rules which apply to any component coded. First of all, the calling sequence must be ordered so that it agrees with that constructed from the Fiload program. Hence the calling sequence begins with table arrays, is followed by output variables, and then by input parameters. State variables require three sequential parameters in the calling sequence: the state variable, the state derivative, and an integer valued integration control. With the exception of the latter, all parameters in the calling sequence are real valued. In general one cannot use any local variables or arrays to store information from call to call since there may be several components in the model which call a given subroutine. In other words, local variables can only be used for scratch calculations, unless the computed information is based on COMMON block inputs.

Most of the coding conventions and techniques used are illustrated in Figures 6.1-1 and 6.1-2. Figure 6.1-1 shows the code for the simple power curve component WP. Following the call sequence are a number of comment cards including the component purpose and calling sequence. The table PW is treated as a single dimension Fortran array. Power output is obtained from the table interpolation subroutine TBLU1. (Use of the table interpolation routines TBLU1 and TBLU2 is explained in Section 2.1). The rest of the code shows the conventions used to compute output statistics and add costs for the cost summary. IMPL is an integer variable which indicates the iteration control status:

IMPL = 0 the first time in a simulation that the model (EQMOD) is
called
= 1 if more iterations and hence subroutine calls are expected
at a given time step
>1 the final iteration through the model.

Hence when IMPL = 0, subroutine variables are initialized, default values are assigned, etc. The statistics are only updated at the final iteration

```

00100 1*  CNP  SUBROUTINE WP ( PH,BI,PD,AMI,AMP,SP,CO,VO,WVO,WVI,WV,CCI,CHI,EC)
00101 2*  C
00102 3*  C
00103 4*  C  PURPOSE  MODEL THE WIND TURBINE AND GENERATOR USING A POWER CURVE
00104 5*  C
00105 6*  C  WRITTEN BY A.N. WARREN  VERSION 1, MARCH 3 1977
00106 7*  C
00107 8*  C  CALL SEQUENCE
00108 9*  C  TABLES
00109 10*  C  PH = WIND GENERATION POWER IN KW VERSUS WIND VELOCITY IN MPH
00110 11*  C
00111 12*  C  OUTPUTS
00112 13*  C  BI = OUTPUT BUS CURRENT, AMPS
00113 14*  C  PD = POWER OUTPUT, KW
00114 15*  C  AMI = MAX. OBSERVED CURRENT, AMPS
00115 16*  C  AMP = MAX. OBSERVED POWER, KW
00116 17*  C  SP = TOTAL OUTPUT ENERGY, KWH
00117 18*  C  CO = OPERATING COST, $
00118 19*  C
00119 20*  C  INPUTS
00120 21*  C  VO = RATED BUS VOLTAGE, VOLTS
00121 22*  C  WVO = POWER CUTIN VELOCITY, MPH
00122 23*  C  WVI = POWER CUTOFF VELOCITY, MPH
00123 24*  C  WV = WIND VELOCITY, MPH
00124 25*  C  CCI = CAPITAL COST / YEAR, $
00125 26*  C  CHI = MAINTENANCE COST / YEAR, $
00126 27*  C  EC = CONTROL ENERGY RATE, $/KWH
00127 28*  C
00128 29*  C  DIMENSION PH(1)
00129 30*  C  COMMON / CIMP / IMPL
00130 31*  C  COMMON / COST / CC,CM,COP / CTIME / TIME / CSTMUL / DUM(6),TINC,TMAX
00131 32*  C
00132 33*  C  POWER OUTPUT CALCULATIONS
00133 34*  C
00134 35*  C  PO = 0.
00135 36*  C  IF (WV,LT,WVO .OR. WV,GT,WVI) GO TO 10
00136 37*  C  N = PH(2)
00137 38*  C  PO = TBLU1(WV,PH(4),PH(4+N),1,4*N)
00138 39*  C  10 BI = PO*1000/VO
00139 40*  C
00140 41*  C  STATISTICS
00141 42*  C
00142 43*  C  IF (IMPL,GT,0) GO TO 20
00143 44*  C  CO = 0.
00144 45*  C  AMI = 0.
00145 46*  C  AMP = 0.
00146 47*  C  SP = 0.
00147 48*  C  TMAX1 = TMAX*.99999
00148 49*  C  20 IF (IMPL,LE,1) RETURN
00149 50*  C  AMI = AMAX1(AMI,BI)
00150 51*  C  AMP = AMAX1(AMP,PO)
00151 52*  C  SP = SP + PO*.5*TINC
00152 53*  C  CO = CO + EC*.5*TINC
00153 54*  C  COST SUMMATION
00154 55*  C  IF (TIME,LT,TMAX1) RETURN
00155 56*  C  CC = CC + CCI
00156 57*  C  CM = CM + CHI
00157 58*  C  COP = COP + CO
00158 59*  C  RETURN
00159 60*  C  END
00160 61*  C
00161 62*  C
00162 63*  C
00163 64*  C
00164 65*  C
00165 66*  C
00166 67*  C
00167 68*  C
00168 69*  C
00169 70*  C
00170 71*  C
00171 72*  C
00172 73*  C
00173 74*  C
00174 75*  C
00175 76*  C
00176 77*  C
00177 78*  C
00178 79*  C
00179 80*  C
00180 81*  C
00181 82*  C
00182 83*  C
00183 84*  C
00184 85*  C
00185 86*  C
00186 87*  C
00187 88*  C
00188 89*  C
00189 90*  C
00190 91*  C
00191 92*  C
00192 93*  C
00193 94*  C
00194 95*  C
00195 96*  C
00196 97*  C
00197 98*  C
00198 99*  C
00199 100*  C
00200 101*  C
00201 102*  C
00202 103*  C
00203 104*  C
00204 105*  C
00205 106*  C
00206 107*  C
00207 108*  C
00208 109*  C
00209 110*  C
00210 111*  C
00211 112*  C
00212 113*  C
00213 114*  C
00214 115*  C
00215 116*  C
00216 117*  C
00217 118*  C
00218 119*  C
00219 120*  C
00220 121*  C
00221 122*  C
00222 123*  C
00223 124*  C
00224 125*  C
00225 126*  C
00226 127*  C
00227 128*  C
00228 129*  C
00229 130*  C
00230 131*  C
00231 132*  C
00232 133*  C
00233 134*  C
00234 135*  C
00235 136*  C
00236 137*  C
00237 138*  C
00238 139*  C
00239 140*  C
00240 141*  C
00241 142*  C
00242 143*  C
00243 144*  C
00244 145*  C
00245 146*  C
00246 147*  C
00247 148*  C
00248 149*  C
00249 150*  C
00250 151*  C
00251 152*  C
00252 153*  C
00253 154*  C
00254 155*  C
00255 156*  C
00256 157*  C
00257 158*  C
00258 159*  C
00259 160*  C
00260 161*  C
00261 162*  C
00262 163*  C
00263 164*  C
00264 165*  C
00265 166*  C
00266 167*  C
00267 168*  C
00268 169*  C
00269 170*  C
00270 171*  C
00271 172*  C
00272 173*  C
00273 174*  C
00274 175*  C
00275 176*  C
00276 177*  C
00277 178*  C
00278 179*  C
00279 180*  C
00280 181*  C
00281 182*  C
00282 183*  C
00283 184*  C
00284 185*  C
00285 186*  C
00286 187*  C
00287 188*  C
00288 189*  C
00289 190*  C
00290 191*  C
00291 192*  C
00292 193*  C
00293 194*  C
00294 195*  C
00295 196*  C
00296 197*  C
00297 198*  C
00298 199*  C
00299 200*  C
00300 201*  C
00301 202*  C
00302 203*  C
00303 204*  C
00304 205*  C
00305 206*  C
00306 207*  C
00307 208*  C
00308 209*  C
00309 210*  C
00310 211*  C
00311 212*  C
00312 213*  C
00313 214*  C
00314 215*  C
00315 216*  C
00316 217*  C
00317 218*  C
00318 219*  C
00319 220*  C
00320 221*  C
00321 222*  C
00322 223*  C
00323 224*  C
00324 225*  C
00325 226*  C
00326 227*  C
00327 228*  C
00328 229*  C
00329 230*  C
00330 231*  C
00331 232*  C
00332 233*  C
00333 234*  C
00334 235*  C
00335 236*  C
00336 237*  C
00337 238*  C
00338 239*  C
00339 240*  C
00340 241*  C
00341 242*  C
00342 243*  C
00343 244*  C
00344 245*  C
00345 246*  C
00346 247*  C
00347 248*  C
00348 249*  C
00349 250*  C
00350 251*  C
00351 252*  C
00352 253*  C
00353 254*  C
00354 255*  C
00355 256*  C
00356 257*  C
00357 258*  C
00358 259*  C
00359 260*  C
00360 261*  C
00361 262*  C
00362 263*  C
00363 264*  C
00364 265*  C
00365 266*  C
00366 267*  C
00367 268*  C
00368 269*  C
00369 270*  C
00370 271*  C
00371 272*  C
00372 273*  C
00373 274*  C
00374 275*  C
00375 276*  C
00376 277*  C
00377 278*  C
00378 279*  C
00379 280*  C
00380 281*  C
00381 282*  C
00382 283*  C
00383 284*  C
00384 285*  C
00385 286*  C
00386 287*  C
00387 288*  C
00388 289*  C
00389 290*  C
00390 291*  C
00391 292*  C
00392 293*  C
00393 294*  C
00394 295*  C
00395 296*  C
00396 297*  C
00397 298*  C
00398 299*  C
00399 300*  C
00400 301*  C
00401 302*  C
00402 303*  C
00403 304*  C
00404 305*  C
00405 306*  C
00406 307*  C
00407 308*  C
00408 309*  C
00409 310*  C
00410 311*  C
00411 312*  C
00412 313*  C
00413 314*  C
00414 315*  C
00415 316*  C
00416 317*  C
00417 318*  C
00418 319*  C
00419 320*  C
00420 321*  C
00421 322*  C
00422 323*  C
00423 324*  C
00424 325*  C
00425 326*  C
00426 327*  C
00427 328*  C
00428 329*  C
00429 330*  C
00430 331*  C
00431 332*  C
00432 333*  C
00433 334*  C
00434 335*  C
00435 336*  C
00436 337*  C
00437 338*  C
00438 339*  C
00439 340*  C
00440 341*  C
00441 342*  C
00442 343*  C
00443 344*  C
00444 345*  C
00445 346*  C
00446 347*  C
00447 348*  C
00448 349*  C
00449 350*  C
00450 351*  C
00451 352*  C
00452 353*  C
00453 354*  C
00454 355*  C
00455 356*  C
00456 357*  C
00457 358*  C
00458 359*  C
00459 360*  C
00460 361*  C
00461 362*  C
00462 363*  C
00463 364*  C
00464 365*  C
00465 366*  C
00466 367*  C
00467 368*  C
00468 369*  C
00469 370*  C
00470 371*  C
00471 372*  C
00472 373*  C
00473 374*  C
00474 375*  C
00475 376*  C
00476 377*  C
00477 378*  C
00478 379*  C
00479 380*  C
00480 381*  C
00481 382*  C
00482 383*  C
00483 384*  C
00484 385*  C
00485 386*  C
00486 387*  C
00487 388*  C
00488 389*  C
00489 390*  C
00490 391*  C
00491 392*  C
00492 393*  C
00493 394*  C
00494 395*  C
00495 396*  C
00496 397*  C
00497 398*  C
00498 399*  C
00499 400*  C
00500 401*  C
00501 402*  C
00502 403*  C
00503 404*  C
00504 405*  C
00505 406*  C
00506 407*  C
00507 408*  C
00508 409*  C
00509 410*  C
00510 411*  C
00511 412*  C
00512 413*  C
00513 414*  C
00514 415*  C
00515 416*  C
00516 417*  C
00517 418*  C
00518 419*  C
00519 420*  C
00520 421*  C
00521 422*  C
00522 423*  C
00523 424*  C
00524 425*  C
00525 426*  C
00526 427*  C
00527 428*  C
00528 429*  C
00529 430*  C
00530 431*  C
00531 432*  C
00532 433*  C
00533 434*  C
00534 435*  C
00535 436*  C
00536 437*  C
00537 438*  C
00538 439*  C
00539 440*  C
00540 441*  C
00541 442*  C
00542 443*  C
00543 444*  C
00544 445*  C
00545 446*  C
00546 447*  C
00547 448*  C
00548 449*  C
00549 450*  C
00550 451*  C
00551 452*  C
00552 453*  C
00553 454*  C
00554 455*  C
00555 456*  C
00556 457*  C
00557 458*  C
00558 459*  C
00559 460*  C
00560 461*  C
00561 462*  C
00562 463*  C
00563 464*  C
00564 465*  C
00565 466*  C
00566 467*  C
00567 468*  C
00568 469*  C
00569 470*  C
00570 471*  C
00571 472*  C
00572 473*  C
00573 474*  C
00574 475*  C
00575 476*  C
00576 477*  C
00577 478*  C
00578 479*  C
00579 480*  C
00580 481*  C
00581 482*  C
00582 483*  C
00583 484*  C
00584 485*  C
00585 486*  C
00586 487*  C
00587 488*  C
00588 489*  C
00589 490*  C
00590 491*  C
00591 492*  C
00592 493*  C
00593 494*  C
00594 495*  C
00595 496*  C
00596 497*  C
00597 498*  C
00598 499*  C
00599 500*  C
00600 501*  C
00601 502*  C
00602 503*  C
00603 504*  C
00604 505*  C
00605 506*  C
00606 507*  C
00607 508*  C
00608 509*  C
00609 510*  C
00610 511*  C
00611 512*  C
00612 513*  C
00613 514*  C
00614 515*  C
00615 516*  C
00616 517*  C
00617 518*  C
00618 519*  C
00619 520*  C
00620 521*  C
00621 522*  C
00622 523*  C
00623 524*  C
00624 525*  C
00625 526*  C
00626 527*  C
00627 528*  C
00628 529*  C
00629 530*  C
00630 531*  C
00631 532*  C
00632 533*  C
00633 534*  C
00634 535*  C
00635 536*  C
00636 537*  C
00637 538*  C
00638 539*  C
00639 540*  C
00640 541*  C
00641 542*  C
00642 543*  C
00643 544*  C
00644 545*  C
00645 546*  C
00646 547*  C
00647 548*  C
00648 549*  C
00649 550*  C
00650 551*  C
00651 552*  C
00652 553*  C
00653 554*  C
00654 555*  C
00655 556*  C
00656 557*  C
00657 558*  C
00658 559*  C
00659 560*  C
00660 561*  C
00661 562*  C
00662 563*  C
00663 564*  C
00664 565*  C
00665 566*  C
00666 567*  C
00667 568*  C
00668 569*  C
00669 570*  C
00670 571*  C
00671 572*  C
00672 573*  C
00673 574*  C
00674 575*  C
00675 576*  C
00676 577*  C
00677 578*  C
00678 579*  C
00679 580*  C
00680 581*  C
00681 582*  C
00682 583*  C
00683 584*  C
00684 585*  C
00685 586*  C
00686 587*  C
00687 588*  C
00688 589*  C
00689 590*  C
00690 591*  C
00691 592*  C
00692 593*  C
00693 594*  C
00694 595*  C
00695 596*  C
00696 597*  C
00697 598*  C
00698 599*  C
00699 600*  C
00700 601*  C
00701 602*  C
00702 603*  C
00703 604*  C
00704 605*  C
00705 606*  C
00706 607*  C
00707 608*  C
00708 609*  C
00709 610*  C
00710 611*  C
00711 612*  C
00712 613*  C
00713 614*  C
00714 615*  C
00715 616*  C
00716 617*  C
00717 618*  C
00718 619*  C
00719 620*  C
00720 621*  C
00721 622*  C
00722 623*  C
00723 624*  C
00724 625*  C
00725 626*  C
00726 627*  C
00727 628*  C
00728 629*  C
00729 630*  C
00730 631*  C
00731 632*  C
00732 633*  C
00733 634*  C
00734 635*  C
00735 636*  C
00736 637*  C
00737 638*  C
00738 639*  C
00739 640*  C
00740 641*  C
00741 642*  C
00742 643*  C
00743 644*  C
00744 645*  C
00745 646*  C
00746 647*  C
00747 648*  C
00748 649*  C
00749 650*  C
00750 651*  C
00751 652*  C
00752 653*  C
00753 654*  C
00754 655*  C
00755 656*  C
00756 657*  C
00757 658*  C
00758 659*  C
00759 660*  C
00760 661*  C
00761 662*  C
00762 663*  C
00763 664*  C
00764 665*  C
00765 666*  C
00766 667*  C
00767 668*  C
00768 669*  C
00769 670*  C
00770 671*  C
00771 672*  C
00772 673*  C
00773 674*  C
00774 675*  C
00775 676*  C
00776 677*  C
00777 678*  C
00778 679*  C
00779 680*  C
00780 681*  C
00781 682*  C
00782 683*  C
00783 684*  C
00784 685*  C
00785 686*  C
00786 687*  C
00787 688*  C
00788 689*  C
00789 690*  C
00790 691*  C
00791 692*  C
00792 693*  C
00793 694*  C
00794 695*  C
00795 696*  C
00796 697*  C
00797 698*  C
00798 699*  C
00799 700*  C
00800 701*  C
00801 702*  C
00802 703*  C
00803 704*  C
00804 705*  C
00805 706*  C
00806 707*  C
00807 708*  C
00808 709*  C
00809 710*  C
00810 711*  C
00811 712*  C
00812 713*  C
00813 714*  C
00814 715*  C
00815 716*  C
00816 717*  C
00817 718*  C
00818 719*  C
00819 720*  C
00820 721*  C
00821 722*  C
00822 723*  C
00823 724*  C
00824 725*  C
00825 726*  C
00826 727*  C
00827 728*  C
00828 729*  C
00829 730*  C
00830 731*  C
00831 732*  C
00832 733*  C
00833 734*  C
00834 735*  C
00835 736*  C
00836 737*  C
00837 738*  C
00838 739*  C
00839 740*  C
00840 741*  C
00841 742*  C
00842 743*  C
00843 744*  C
00844 745*  C
00845 746*  C
00846 747*  C
00847 748*  C
00848 749*  C
00849 750*  C
00850 751*  C
00851 752*  C
00852 753*  C
00853 754*  C
00854 755*  C
00855 756*  C
00856 757*  C
00857 758*  C
00858 759*  C
00859 760*  C
00860 761*  C
00861 762*  C
00862 763*  C
00863 764*  C
00864 765*  C
00865 766*  C
00866 767*  C
00867 768*  C
00868 769*  C
00869 770*  C
00870 771*  C
00871 772*  C
00872 773*  C
00873 774*  C
00874 775*  C
00875 776*  C
00876 777*  C
00877 778*  C
00878 779*  C
00879 780*  C
00880 781*  C
00881 782*  C
00882 783*  C
00883 784*  C
00884 785*  C
00885 786*  C
00886 787*  C
00887 788*  C
00888 789*  C
00889 790*  C
00890 791*  C
00891 792*  C
00892 793*  C
00893 794*  C
00894 795*  C
00895 796*  C
00896 797*  C
00897 798*  C
00898 799*  C
00899 800*  C
00900 801*  C
00901 802*  C
00902 803*  C
00903 804*  C
00904 805*  C
00905 806*  C
00906 807*  C
00907 808*  C
00908 809*  C
00909 810*  C
00910 811*  C
00911 812*  C
00912 813*  C
00913 814*  C
00914 815*  C
00915 816*  C
00916 817*  C
00917 818*  C
00918 819*  C
00919 820*  C
00920 821*  C
00921 822*  C
00922 823*  C
00923 824*  C
00924 825*  C
00925 826*  C
00926 827*  C
00927 828*  C
00928 829*  C
00929 830*  C
00930 831*  C
00931 832*  C
00932 833*  C
00933 834*  C
00934 835*  C
00935 836*  C
00936 837*  C
00937 838*  C
00938 839*  C
00939 840*  C
00940 841*  C
00941 842*  C
00942 843*  C
00943 844*  C
00944 845*  C
00945 846*  C
00946 847*  C
00947 848*  C
00948 849*  C
00949 850*  C
00950 851*  C
00951 852*  C
00952 853*  C
00953 854*  C
00954 855*  C
00955 856*  C
00956 857*  C
00957 858*  C
00958 859*  C
00959 860*  C
00960 861*  C
00961 862*  C
00962 863*  C
00963 864*  C
00964 865*  C
00965 866*  C
00966 867*  C
00967 868*  C
00968 869*  C
00969 870*  C
00970 871*  C
00971 872*  C
00972 873*  C
00973 874*  C
00974 875*  C
00975 876*  C
00976 877*  C
00977 878*  C
00978 879*  C
00979 880*  C
00980 881*  C
00981 882*  C
00982 883*  C
00983 884*  C
00984 885*  C
00985 886*  C
00986 887*  C
00987 888*  C
00988 889*  C
00989 890*  C
00990 891*  C
00991 892*  C
00992 893*  C
00993 894*  C
00994 895*  C
00995 896*  C
00996 897*  C
00997 898*  C
00998 899*  C
00999 900*  C
01000 901*  C
01001 902*  C
01002 903*  C
01003 904*  C
01004 905*  C
01005 906*  C
01006 907*  C
01007 908*  C
01008 909*  C
01009 910*  C
01010 911*  C
01011 912*  C
01012 913*  C
01013 914*  C
01014 915*  C
01015 916*  C
01016 917*  C
01017 918*  C
01018 919*  C

```

```

00100 10 CGT
00101 20 SUBROUTINE GETP,CE,ES,PL,EFZ,P,Z,PMH,SP,PI,PIF,PIST,PAS,PA,SK,VH,
00102 30 1,PI,PMI,CCI,CNI)
00103 40
00104 50 C
00105 60 C PUMPDSE YJDL AC INDUCEDBY GENERATOR
00106 70 C
00107 80 C METHOD MECHANICAL AND ELECTRICAL EFFICIENCIES ARE USED TO COMPUTE
00108 90 OUTPUT POWER. MOTOR SPEED IS COMPUTED ASSUMING MOTOR IS
00109 00 PROPORTIONAL TO SLIP.
00110 100 C
00111 110 C WRITTEN BY J. L. HARMEN VERSION 1, MARCH 16, 1977
00112 120 C
00113 130 C CALL SEQUENCE
00114 140 C
00115 150 C P2 = OUTPUT POWER, KW
00116 160 C CE = ELECTRICAL EFFICIENCY
00117 170 C ES = MOTOR SPEED, RPM
00118 180 C PL = POWER LOSS, KW
00119 190 C EFZ = OUTPUT PRODUCT EFFICIENCY
00120 200 C PMS = MAXIMUM OUTPUT POWER, KW
00121 210 C PMH = MAX. (MECHANICAL) OUTPUT POWER / RATED POWER
00122 220 C SP = TOTAL OUTPUT ENERGY, KWH
00123 230 C
00124 240 C
00125 250 C INPUTS
00126 260 C P1 = INPUT POWER, KW
00127 270 C RHP = RATED OUTPUT POWER, KW
00128 280 C RSP = SYNCHRONOUS MOTOR SPEED, RPM
00129 290 C PAS = FIELD POLE SLIP (DEFAULT = .05)
00130 300 C PA = MECHANICAL LOSS, JOULE/SEC
00131 310 C SM = STATOR RESISTANCE, OHMS
00132 320 C VD = RATED ARM VOLTAGE, VOLTS
00133 330 C EPI = INPUT PRODUCT EFFICIENCY
00134 340 C PM1 = MAXIMUM INPUT POWER, KW
00135 350 C CE1 = CAPITAL COST/TON, $
00136 360 C CNI = MAINTENANCE COST/TON, $
00137 370 C
00138 380 C COMMON /CMTPL/ IMPL,ICNT,CTIME,TIME
00139 390 C COMMON /COST/ CC,CM,CO,CCV,CSTIM,CONCOST,TIME,PMAX
00140 400 C (INITIALIZATION)
00141 410 C
00142 420 C IF IMPL.GT.0 GO TO 30
00143 430 EFF = 1.
00144 440 PMAX = IMAX+.00000
00145 450 IFIRST,EL, .00000) MSF = 1000.
00146 460 IF (PAS,LD, .00000) PAS = .05
00147 470 IF (PA, .00, .00000) PA = 0.
00148 480 IF (SM, .00, .00000) SM = 0.0/RHP
00149 490 IF (VD, .00, .00000) VD = 0.0
00150 500 IF (PM1,EL, .00000) PM1 = 1.0/EL
00151 510 PM = 0.0
00152 520 SP = 0.0
00153 530 CATE = 0.0/CONCOST/VD
00154 540 CE = RHP/(RHP + 400.00) + 0.01/PA/VD
00155 550 C
00156 560 C COMPUTE MOTOR SPEED AND OUTPUT POWER
00157 570 C
00158 580 C 10 IF (P1,CT, 0.0) GO TO 20
00159 590 P2 = 0.0
00160 600 PL = 0.0
00161 610 MS = RSP
00162 620 GO TO 10
00163 630 C
00164 640 C 20 A = RHP/(CE+PA3)
00165 650 B = RSP/(1 + RSP**2+DAN1.0900E-05)
00166 660 MS = B*(1 + P1)
00167 670 P2 = RHP*(MS/MSF + PL)/RHP
00168 680 IF (P2,CT, 0.0, 0.0, IMPL,PI,2) WRITE(CT,100)
00169 690 100 FORMATTED, NON-STRUCTURED OUTPUT ENERGY RATED POWER P)
00170 700 C
00171 710 C IF (P2,CT, 0.0, 0.0, IMPL,ES,2) ICNT=ICNT+1
00172 720 PL = P1 - P2
00173 730 EFF = P2/P1
00174 740 PMS = 3*PI*(RHP, PM1,CEP)
00175 750 C
00176 760 C STATISTICS
00177 770 C IF (IMPL,EL,1) OPTIM=
00178 780 PMH = 3*PI*(PMH, P2/RHP)
00179 790 SP = SP + P2*.01/11C
00180 800 C
00181 810 C COST COMPUTATION
00182 820 C 10 TIME,LT,TIME) METHOD
00183 830 CC = CC + CE1
00184 840 CM = CM + CNI
00185 850 C
00186 860 C RETURN
00187 870 C

```

FIGURE 6.1-2 SAMPLE COMPONENT CODE

when the model has presumably attained steady state values. Finally, the costs are added up when the simulation has reached the maximum time point. Capital costs, maintenance costs, and operating costs are stored in the first three locations of common block COST.

Figure 6.1-2 shows the code for the generator component GE. The program automatically assigns default parameters = .99999. Hence, when IMPL = 0 component dependent default values are assigned whenever the .99999 default is assumed. The code near Format statement 100 shows a typical diagnostic printout. The diagnostic is only printed if IMPL = 2 since we need only diagnose errors at the final iteration. Note that a counter ICNT is updated each time a diagnostic is printed. It is stored in the second location of common block CIMPL and is monitored to see if diagnostic print lines exceeds DLINES. If so, IMPL is set to 3 the final iteration, so that no further diagnostics are printed. The last convention observed here concerns the use of the maximum power and product efficiency variables denoted PM1, EF1, PM2, EF2. These variables are used to communicate information to the logic components PD and PA. The efficiency variable EFF is defined as the ratio of output power to input power except when P1 = 0. In this case the old EFF value is used, but in any case EFF = 0 must be avoided since this would communicate a zero efficiency to a logic device which would then generate an infinite request. It is seen that EF2 and PM2 represent the joint efficiency and maximum power at the output port as a consequence of the rated generator power and computed input/output efficiency.

Storage devices have in addition to the above, certain conventions to communicate with the logic components. An input parameter RE1 for port 1 request is used to initiate power discharge from storage. An output variable RE2 for port 2 request is used to communicate a maximum charge rate request and is usually computed by

$$RE2 = \text{MIN} (MP1, RAPI) / EF1$$

where MP1 and EF1 are the input maximum power and input product efficiency,

and RAP denotes the maximum storage charging rate. A priority interrupt INT should also be defined so that $INT = 1$, when storage is empty or at a minimum, $INT = 0$, if no interrupt is required, and $INT = -1$, at full storage capacity. The state of storage is normally a state variable so that the code computes the state derivative at each time point and lets the integrator update the state at each time point.

6.2 FILOAD PROGRAM

In addition to placing the subroutine representing the new standard component in the component library, descriptions of the inputs, outputs, and tables required by the new component must be added to the permanent file, M18. These lists are used by the Model Generation program to direct the connection of component inputs and outputs. The program FILOAD is provided to perform any of the following tasks:

1. Add new input, output, or table name lists.
2. Replace existing input, output or table name lists.
3. Remove all name lists for specified components.
4. Dump contents of M18 file onto Tape 9 in input format.

6.2.1 FILOAD Program Commands

The FILOAD program will recognize the following commands.

LIST STANDARD COMPONENTS

The LIST COMPONENTS command causes the program to print the input, output, and table lists for all components modified or added to the M18 file. If this command is not given the program will merely give a message stating the name of the new components being added to the file.

PURGE

The PURGE command can be used to remove a component from the M18 file. The PURGE command is followed by the names of the components to be purged. The command and the component names must be separated by one of the standard delimiters; i.e. [] three or more blanks, [,] comma, [=] equal sign, [{}] left or right parentheses.

Example 6.3: PURGE = CM, TB, OB

This command would remove all lists for the CM, TB, and OB components from the name list file.

SYMBOL

The SYMBOL command may be used to designate the type of symbol that is to appear for each standard component in the lineprinter drawn model schematic diagram. The SYMBOL command is followed by the names of the components each followed by a symbol number. The symbol numbers and their associated symbols are shown in Figure 6.2-1. The SYMBOL command, component names, and symbol numbers are separated by standard delimiters.

Example: SYMBOL, CO = 100, SH = 200, TU = 300, OC = 400

If a symbol number is not specified for a component the default symbol of a square box will be used.

DUMP FILE

The DUMP FILE command causes the FILOAD program to dump the contents of the M18 file onto DUMPF9, in the input format of the FILOAD program. Thus for each standard component, a list of inputs, outputs, and tables will be produced. This data will be preceded by the command NEW FILE described below. This file may be edited to modify the input, output or tables description of any existing standard component or to derive a new standard component

STANDARD SCHEMATIC SYMBOLS

4	5	6	7	8
14	<pre> ** * * * * * * * * * CO * * 15 * * * * * * * ** </pre> <p>SYMBOL = 100</p>	16	<pre> 00000000 0 0 0 OC 0 0 17 0 0 0 00000000 </pre> <p>SYMBOL = 400</p>	18
24	25	26	27	28
34	<pre> ***** * * * SH * * 35 * * * ***** </pre> <p>SYMBOL = 200</p>	36	<pre> ***** * * * ME * * 37 * * * ***** </pre> <p>SYMBOL = ANY OTHER NUMBER</p>	38
44	45	46	47	48
54	<pre> ** * * * * * * * * TU * * 55 * * * * * * ** </pre> <p>SYMBOL = 300</p>	56	57	58
64	65	66	67	68

FIGURE 6.2-1 LIST OF STANDARD COMPONENT SYMBOLS

description from an existing one. The results of such an editing would then serve as input data to a subsequent run of the FILOAD program. Unless it is intended to purge the M18 file and start anew, the NEW FILE command at the beginning of DUMPF9 should be removed before the subsequent run of the FILOAD program.

NEW FILE

The NEW FILE command instructs the FILOAD program to construct a new M18 file. This command must occur as the first card in a set of data describing a completely new M18 file. Any previous components that may have existed on the M18 file are purged by this command. It is therefore only used when installing a complete new M18 file.

FILE NAME

This command is used to load the file name to be associated with the M18 file. The current M18 file name is WINDENERGY. This command is used as:

FILE NAME = WINDENERGY

6.2.2 Input Name Lists

Input name lists are identified by the letters INPT following the component name. Thus, the input name list for a component DC would be introduced with the phrase, DCINPT. This must be followed by a phrase that contains the number of names in the input name list.

The input names are contained on the following data cards, 8 names per card. The names must be left adjusted in fields, 10 characters wide. The names are placed in Columns 1 through 3 of each field. Column 9 of each field can be used to indicate a port number which can be attached to the name to distinguish it from other quantities of the same name that occur with the given

component. Thus, to indicate that temperature, T, is an input to port 1, the input name list would be:

Column:	1	2	3	4	5	6	7	8	9	10
Item:	T								1	

This quantity would then be referred to as T1.

Example 6.4:

```
SWINPT = 3
IN.....1,IN.....2,CNT
```

(The dots are used here to indicate blank spaces and would not be included in an actual data card).

These two data cards would indicate that the component SW had 3 input quantity names. A quantity IN appears at port 1, and is to be referred to as IN1. A quantity IN appears at port 2, and is to be referred to as IN2. A third input quantity CNT has no port designation. Note that if a port number is to be attached to a quantity name, that name should contain no more than 2 characters.

The sequence of names in the input name list must match the sequence of input arguments in the component call sequence.

6.2.3 Output Name Lists

Output name lists are identified by the letters OUTP following the component name. Thus, the output name list for a component DC would be introduced with the phrase, DCOUTP. This must be followed by a phrase that contains the number of names in the output name list.

The output names are contained on the following data cards, 8 names per card. The names must be left adjusted in fields 10 characters wide. The names are placed in Columns 1 through 3 of each field. Column 9 of each field can be used to introduce a port number which can be attached to the name to distinguish it from other quantities of the same name that occur with the given component. If the output quantity is a state variable, this must be indicated by placing S in Column 10 of the field. Thus, if power P is a state variable output quantity at port 2, the output name list would be:

Column:	1	2	3	4	5	6	7	8	9	10
Item:	P								2	S

This quantity would then be referred to as P2.

Example 6.5:

```
TZOUTP = 3
X.....1SX.....2SOUT
```

(The dots are used here to indicate blank spaces, and would not be included on an actual data card).

These two data cards would indicate that the component TZ had 3 output quantity names. A quantity X appears at port 1. This is a state variable, and will be referred to as X1. A quantity X is also a state variable that appears at port 2. It will be referred to as X2. The quantity OUT is an output variable, not a state variable, and does not have a port number associated with it. Note, that if a port number is to be attached to a quantity name that name should contain no more than 2 characters. These two characters plus the port number will reach the maximum number of 3 characters in a quantity name.

The sequence of names in the output name list must match the sequence of output arguments in the component call sequence. However, whereas three argu-

ments are provided for each state in the subroutine call sequence, only one name is included in the output name list.

6.2.4 Table Name Lists

Table name lists are identified by the letters TABS, following the component name. Thus, the table name list for a component CM would be introduced with the phrase CMTABS. This must be followed by a phrase containing the number of names in the table name list. The table names are contained in the following cards, one table name per card. The name is located in the first 3 columns of the card, it must be accompanied by the maximum dimension that is to be provided for this table. This number must be given in columns 4 through 10 and should have a decimal point given. For single independent variable tables this number must be negative. For tables with two independent variables, this number must be positive.

Example:

```
CMTABS = 3
TAM      53.
TAB      43.
TCM     -27.
```

These four data cards would indicate that the component CM had 3 tables. The first two tables TAM and TAB have two independent variables each, as indicated by the positive dimension numbers. The table TCM has only one independent variable, as indicated by the negative dimension number. 53, 43, and 27 words of storage are to be provided for tables TAM, TAB, and TCM respectively. The maximum storage is related to the maximum number of primary, NX, and secondary, NZ, independent variables by:

```
MAX = 3 + NX*NZ+NX*NZ for tables with two independent variables
MAX = 3 + 2*NX for tables with one independent variable
```


7.0 LIBRARY COMPONENT DESCRIPTIONS

This section describes the mathematical algorithms and input/output structure of the SIMWEST library components. Each component writeup contains a brief textual description of the algorithms, a mathematical expression summarizing its function, a list of input and output variables, and a description of the calculation sequence and logic used in the model. A figure is provided which shows the nominal input and output connections, and the state variables of each component.

There are a number of features and conventions in the component descriptions which require some elaboration. These are briefly summarized below.

7a. INPUT/OUTPUT NAME LISTS

A potentially confusing factor is the way port numbers on input parameters and output variables are designated. On the model generation input cards the name of the physical quantity and the port number is separated by a comma. For example, the power variable with port designation 1 is denoted P,1. In defining input to the simulation program, this same variable would be denoted P1. To emphasize the distinction between the physical quantities and port numbers, they are listed separately in the name lists of the component writeups. For example, P 1 in the name list denotes the power variable (or parameter) with port designation 1 even though in other parts of the text it may simply be denoted P1.

Another convention in the name lists is that the alphabetic symbol 'O' is shown as @ to distinguish this symbol from a zero. Elsewhere in the text symbols such as V@ may be referred to as VO.

7b. INPUT PARAMETER SPECIFICATION

All input parameters are associated with default values. Many of the parameters have default values denoted in the parameter description by the letter D. For example in the Battery component the default value for terminal resistance, RT, is $D = .001$ ohms. All input parameters for which a default value is not so specified has a default value of .99999. Default values are intended to enable users to put models together quickly by specifying a minimum of input data. Users need only specify detailed parameter values for those components of current interest. One must be careful using this approach since the operating characteristics and efficiency of a 10kw rated device may for example be quite different than for a 100kw device.

Any user specified input parameter can be driven by one or two dimension table lookups using the FU and FV components. This enables the user to build more detailed models using time or other output variables to drive the tables. For example, if one needs to specify cost of peak load generation to the utility component as a function of peak load request, then one adds FU as an input to UT and specifies load request as an input connection to FU. The desired function table is then input to FU.

It may be noted that not all of the components have maintenance or operating cost inputs. Thus, whenever these costs are important, one can aggregate such costs and input lumped costs to the model. For example, the maintenance cost of the hydro storage system may include maintenance costs for the pump and turbine.

7c. COMPONENT LOGIC

In constructing SIMWEST components, we have adopted several conventions to aid communication with the logic components. All physical components distributing power are given two input parameters EF and MP (port 1) and two output

variables EF and MP (port 2). The output EF is the product efficiency of all components in the distribution subsystem up to and including the given component, and MP is the maximum power deliverable at the output of the component. Each storage component has in addition a power request input denoted RE (port 1), a power request output denoted RE (port 2), and a priority interrupt flag denoted INT.

Figure 7.0 shows the logic and physical variable connections for power flow in and out of a hydro reservoir. Power flows from the power divider to the pump at a rate not to exceed the request RE from HS. The HS request is computed by dividing the input maximum power by the input (or pump) efficiency EF. Hence, the maximum power flowing to HS cannot exceed $RE * EF = MP$. Similarly, the input request to HS is computed by the PA component so as not to exceed the maximum input power MP divided by EF (turbine efficiency). Hence, the power that flows to PA cannot exceed $RE * EF = \text{input maximum power}$.

When the hydro reservoir is empty, the interrupt flag is turned on and the priority sequence is changed to 1 so that the reservoir is given access to power flowing into the divider.

FIGURE 7.0 SAMPLE CONNECTIONS FOR LOGIC COMPONENTS

7.1 ADMITTANCE

The admittance model can be used to model transmission lines, transformers, capacitors or impedance power flows. A primary assumption is that the reactive parameters dominate the real parameters so that power transfer angle is solely based on reactive values, and power losses are based on the real admittance parameters and on power angle. The equation for power loss is based upon the following model:

FIGURE 7.1 ADMITTANCE NETWORK MODEL

$$\begin{pmatrix} I_1 \\ I_2 \end{pmatrix} = \begin{pmatrix} G_1 + jB_1 & G_M + jB_M \\ G_M + jB_M & G_2 + jB_2 \end{pmatrix} \begin{pmatrix} V_1 \\ V_2 \end{pmatrix}$$

Where the reactive parameters B_1 and B_2 do not enter into the power loss calculations.

Inputs

<u>Parameter/Port</u>		<u>Description</u>	<u>Units</u>
G1,GM,G2		Real admittance parameters *	mho
BM		Reactive admittance parameter *(#0)	mho
V0		Rated voltage magnitude	volts
P	1	Input power	kw
EF	1	Input product efficiency	-
MP	1	Maximum input power	kw
CC		Capital cost/year	\$

Outputs

<u>Variable/Port</u>			
P	2	Output power	kw
PL		Power loss	kw
PA		Power angle	deg
EF	2	Output product efficiency	-
MP	2	Maximum output power	kw

* - See next page for User Input to Model Transmission lines, Transformers and Impedances.

Transmission Line Input:

$$G1 = G2 = g * l$$

$$GM = -g * l$$

$$BM = 1 / (\omega * L * l)$$

where g = line conductance per unit length

l = length of line

ω = frequency in radians/sec = 120π

L = line inductance per unit length

Transformer Input:

$$G1 = G2 = GM = 0$$

$$BM = 1 / X * h$$

where X = reactance in ohms

h = turns ratio

(No power loss modeled with a transformer)

Impedance Input: (Includes capacitors and inductors)

$$G1 = G2 = -GM = R / (R^2 + X^2)$$

$$BM = X / (R^2 + X^2)$$

where R = resistance in ohms

X = reactance in ohms

$$= \begin{pmatrix} \omega L & \text{for an inductance } L \\ -\frac{1}{\omega C} & \text{for a capacitance } C \end{pmatrix}$$

Calculation Sequence

If $P_1 \leq 0$ $P_2 = P_L = P_A = 0$ and Return

1) Compute power angle

If $P_1 * 1000 > BM * V_0^2$, $\cos \theta = 0$ and write DIAGNOSTIC

$$\theta = -\sin^{-1}(P_1 * 1000 / BM * V_0^2)$$

$$PA = \theta * 180 / \pi$$

$$\cos \theta = \sqrt{1 - (P_1 * 1000 / BM * V_0^2)^2}$$

2) Compute power loss and output power

$$P_L = V_0^2 * (G_1 + G_2 + 2 * G_M * \cos \theta) / 1000$$

$$P_2 = P_1 - P_L$$

$$EFF = P_2 / P_1$$

If $P_2 > 0$ go to 3)

write DIAGNOSTIC

$$EFF = 1.$$

3) Efficiency and maximum output power

$$EF2 = EF1 * EFF$$

$$MP2 = \min(MP1, |BM| * V_0^2 / 1000) * EFF$$

4) Compute costs

ENTRY POINT 000212

COMMON FLORES

EXTERNAL REFERENCES (BLOCK NAME)

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

3001	000017	IM	0F01	000040	100L	0000	000004	10BF	0001	000076	200L	0001	000140	300L
3003	000027	30BF	0C01	000162	400L	0006	R 000000	CCY	0005	000000	DUM	0000	R 000035	EFF
3003	1 000031	ICNT	0003	1 000000	IMPL	0000	000054	FNJPS	0000	R 000001	RR	0000	R 000003	RPC
0000	R 000002	RR2	0000	R 000000	TIMEIA	0004	R 000000	TIME	0005	R 000007	THAX	0000	R 000000	THAX1

```

C0100      1*      C00      SUBROUTINE ADIPZ,PL,PA,EF2,MP2, B1,GM,G2,BM,VO,P1,EF1,MP1,CC)      C00000
C0101      2*      C00      C00000
C0101      3*      C      C00000
C0101      4*      C      PURPOSE      MODEL OF TRANSMISSION LINES, TRANSFORMERS,      C00000
C0101      5*      C      CAPACITORS, OR IMPEADANCE POWER LOSS      C00000
C0101      6*      C      C00000
C0101      7*      C      METHOD      OUTPUT POWER AND POWER LOSS COMPUTED FROM      C00000
C0101      8*      C      INPUT POWER      C00000
C0101      9*      C      C00000
C0101     10*      C      WRITTEN BY Y.K.CHAN      VERSION 1, JULY,1977      C00000
C0101     11*      C      C00000
C0101     12*      C      CALL SEQUENCE      C00000
C0101     13*      C      OUTPUTS      C00000
C0101     14*      C      P2 -OUTPUT POWER,KW      C00000
C0101     15*      C      PL -POWER LOSS,KW      C00000
C0101     16*      C      PA -POWER ANGLE,DEG      C00000
C0101     17*      C      EF2 -OUTPUT PRODUCT EFFICIENCY      C00000
C0101     18*      C      MP2 -MAXIMUM OUTPUT POWER,KW      C00000
C0101     19*      C      INPUTS      C00000
C0101     20*      C      B1,GM,G2 -REAL ADMITTANCE PARAMETERS,MHO      C00000
C0101     21*      C      PM -REACTIVE ADMITTANCE PARAMETERS (LINE 0.3,MHO)      C00000

```

AD

00101	22*	C	VO -RATED VOLTAGE MAGNITUDE,VOLTS	C00000
00101	23*	C	P1 -INPUT POWER,KW	C00000
00101	24*	C	EF1 -INPUT PRODUCT EFFICIENCY	000000
00101	25*	C	MP1 -MAXIMUM INPUT POWER,KW	000000
00101	26*	C	CC -CAPITAL COST/YEAR,%	000000
00101	27*	C		000000
00103	28*		COMMON /C/IMPL/INPL,ICNT/CTIME/TIME/CSIMUL/QUIN71,TMAX	CC0000
00103	29*	X	/COST/CCI	C00C00
00104	30*		REAL MP2,MP1	000000
00104	31*	C		000000
00105	32*		P2=0.	C0C000
00106	33*			000000
00106	34*		TMAX1=TMAX*.99999	000000
00107	35*		IF(P1.67.C.160 TO 10	000000
00111	36*		P2=0.	000000
00112	37*		PL=C.	000000
00113	38*		PA=C.	000010
00114	39*		MP2=MP1	C00010
00115	40*		EF2=EF1	000010
00116	41*		GO TO 400	000015
00116	42*	C		CC0015
00116	43*	C	COMPUTE POWER ANGLE	000015
00116	44*	C		CC0015
00117	45*		10 RP=P1*1000./(GM*VO*VO)	000017
00120	46*		RP2=RP*RR	000025
00121	47*		IF(RR2.LE.1.160 TO 100	000027
00123	48*		PA=-9C.	000032
00124	49*		RFC=0.	000034
00125	50*		IF(IMPL.EQ.2)WRITE(6,(08)P1,BN,VO	000035
00133	51*	100	FORMAT(10C,13H INPUT POWER ,F12.3,33H 100 HIGH RELATIVE TO ADMITT	000050
00133	52*		XNCE ,F12.3,19H AND RATED VOLTAGE ,F12.31	000050
00134	53*		IF(IMPL.EQ.2)ICNT=ICNT+1	000050
00134	54*		GO TO 200	000056
00137	55*	100	THETA=-ASIN(RR)	000060
00140	56*		PA=THETA*180./3.14159	000063
00141	57*		RRC=SQRT(1.-RR2)	000066
00141	58*	C		000066
00141	59*	C	COMPUTE POWER LOSS AND OUTPUT POWER	000066
00141	60*	C		000066
00142	61*	200	PL=VO*VO*(G1+G2+2.*GM*RRRC)/1000.	000076
00143	62*		P2=P1-PL	000110
00144	63*		EFF= P2/P1	000112
00145	64*		IF(P2.GE.0.160 TO 100	000114
00147	65*		P2=0.	000117
00150	66*		EFF=1.	000120
00151	67*		IF(IMPL.NE.2)160 TO 100	000122
00153	68*		WRITE(6,308)PL,P1	000125
00157	69*	300	FORMAT(10C,24H ADMITTANCE POWER LOSS ,F12.3,21H EXCEEDS INPUT PONE	000134
00157	70*		XR ,F12.3)	000134
00160	71*		ICNT=ICNT+1	000134
00160	72*	C		000134
00161	73*	300	EF2=EF1	000140
00162	74*		IF(P2.C1.C.1EF2=EF1+EFF	000141
00164	75*		MP2=AMIN(MP1,ABS(BN)*VO*VO/1000.)+EFF	000147
00164	76*	C		000147
00165	77*	400	IF(IMPL.LE.1)RETURN	000162
00167	78*		IF(TIME.LT.TMAX)RETURN	000170

00171
00171
00172
00173

79*
80*
81*
82*

C

CCI=CCI+CC
RETURN
END

000177
000177
000202
000263

AD

7.2 TEST FUNCTION GENERATOR

InputsParameter/PortDescription

C00 Specifies which analytic function is calculated. (See equations below for use of these inputs)

C1

C2

C3

C4

C5

OutputsVariable/Port

F0 Output variable

Calculation Sequence

```

C00 = 1  F0 = C1 + C2*SIN(C3*T + C4)
        2  F0 = C1 + C2*COS(C3*T + C4)
        3  F0 = C1 + exp(-C5*T) * SIN(C3*T + C4)
        4  F0 = C1 + exp(-C5*T) * COS(C3*T + C4)
        5  F0 = C1 + C2*T
        6  F0 = C1 + C2*exp(-C3*T)
        where: T = TIME
  
```

SUBROUTINE AF ENTRY POINT 000143

STORAGE USED CODE(1) 000166; DATA(0) 000014; BLANK COMMON(2) 000600

COMMON BLOCKS

0003 CTIME 000001

EXTERNAL REFERENCES (BLOCK, NAME)

0004 NERR25
 0005 SIN
 0006 COS
 0007 EXP
 0010 NERR35

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001	000022	10L	0001	000138	100L	0001	000035	20L	0001	000050	30L	0001	000073	40L
0001	000116	50L	0001	000123	60L	0000	000001	INJPS	0000	I	000000	NCODE	0003	R
														000000 TIME

00100	1*	CAF													000000
00101	2*		SUBROUTINE AF(F0,C0D,C1,C2,C3,C4,C5)												000000
00101	3*	C													000000
00101	4*	C	PURPOSE - TO SIMULATE ANALYTICAL FUNCTIONS												000000
00101	5*	C													000000
00101	6*	C													000000
00101	7*	C	METHOD - SEE CODING												000000
00101	8*	C													000000
00101	9*	C													000000
00101	10*	C	WRITTEN BY - ADAM LLOYD												000000
00101	11*	C													000000
00101	12*	C													000000
00101	13*	C	LIMITATIONS - NONE												000000
00101	14*	C													000000
00101	15*	C													000000
00101	16*	C	INPUT/OUTPUT LIST												000000
00101	17*	C													000000
00101	18*	C	F0	OUTPUT VARIABLE											000000
00101	19*	C	C0D	CODE IDENTIFYING ANALYTICAL FUNCTION											000000
00101	20*	C		CODE = F0 =											000000
00101	21*	C		1	$C1+C2*\sin(C3*TIME+C4)$										000000
00101	22*	C		2	$C1+C2*\cos(C3*TIME+C4)$										000000
00101	23*	C		3	$C1+C2*\exp(-C5*TIME)*\sin(C3*TIME+C4)$										000000
00101	24*	C		4	$C1+C2*\exp(-C5*TIME)*\cos(C3*TIME+C4)$										000000
00101	25*	C		5	$C1+C2*TIME$										000000
00101	26*	C		6	$C1+C2*\exp(-C5*TIME)$										000000

00101	27*	C	C1	CONSTANT INPUTS FOR ABOVE EQNS	---	INPUT	PARAM	000000
00101	28*	C	C2	CONSTANT INPUTS FOR ABOVE EQNS	---	INPUT	PARAM	000000
00101	29*	C	C3	CONSTANT INPUTS FOR ABOVE EQNS	---	INPUT	PARAM	000000
00101	30*	C	C4	CONSTANT INPUTS FOR ABOVE EQNS	---	INPUT	PARAM	000000
00101	31*	C	C5	CONSTANT INPUTS FOR ABOVE EQNS	---	INPUT	PARAM	000000
00101	32*			COMMON/CTIME/TIME				000000
00104	33*			NCODE=000				000000
00105	34*			GO TO (10,20,30,40,50,60),NCODE				000006
00106	35*	10		F0=C1+C2+SIN(C3*TIME+C4)				000022
00107	36*			GO TO 100				000033
00110	37*	20		F0=C1+C2+COS(C3*TIME+C4)				000035
00111	38*			GO TO 100				000046
00112	39*	30		F0=C1+C2*EXP(-C5*TIME)*SIN(C3*TIME+C4)				000050
00113	40*			GO TO 100				000071
00114	41*	40		F0=C1+C2*EXP(-C5*TIME)*COS(C3*TIME+C4)				000073
00115	42*			GO TO 100				000114
00116	43*	50		F0=C1+C2*TIME				000116
00117	44*			GO TO 100				000121
00120	45*	60		F0=C1+C2*EXP(-C5*TIME)				000123
00121	46*	100		RETURN				000134
00122	47*			END				000165

7.3 BATTERY

The battery model is based on the circuit diagram shown below. Current flow is determined by the output power request minus input power. Battery leakage is proportional to stored energy. Priority interrupt logic is activated when a minimum or maximum capacity level is attained.

FIGURE 7.3 BATTERY CIRCUIT DIAGRAM

Basic Equations

The output power P2, stored energy PE, terminal current I, and capacitor voltage VC is computed using the following equations:

$$P2 = RE1$$

$$PE = (VC^2 + 2*VO*VC)*CB/7.2*10^6$$

$$(P2 - P1)*1000 = (VO + VC)I - I^2*RT$$

$$PE = -(I + VC/RL)(VC + VO)/1000$$

Inputs

<u>Parameter/Port</u>		<u>Description</u>	<u>Units</u>
P	1	Input power	kw
V0		Internal voltage	volts
RT		Terminal resistance (D = 0.001)	ohms
CB ¹		Battery capacitance (D = 2.88×10^8)	farads
RL ¹		Leakage resistance (D = 0.05)	ohms
RAP		Rated input power	kw
EF	1	Input product efficiency	-
MP	1	Maximum input power	kw
E1		Maximum energy storage	kwh
RE	1	Power request	kw
EDE		Energy deadband for priority resequencing	kwh
DT		Down time for priority resequencing	h
CC		Capital cost/year	\$
CM		Maintenance cost/year	\$

Outputs

<u>Variable/Port</u>			
P	2	Output power (=RE1)	kw
PE		Stored energy (state of charge)	kwh
I		Terminal current (+out, -in)	amps
VC		Capacitor voltage	volts
VT		Terminal voltage	volts
PL		Power loss	kw
T0		Time when battery was discharged	h
MP	2	Maximum output power	kw
INT		Priority interrupt flag	-
RE	2	Maximum charging rate request	kw

D - Default values supplied

1 - Battery leakage time constant in hours = $CB \cdot RL / 3600$

Statistics

<u>Variable/Port</u>	<u>Description</u>	<u>Units</u>
MPE	Maximum stored energy	kwh
SPC	Sum of charging energy	kwh
SPD	Sum of discharging energy	kwh

Calculation Sequence

- 1) Compute VC

$$VC = \sqrt{7.2 \times 10^6 * PE / CB + V0^2} - V0$$

- 2) Solve for terminal current I

If $(P2 - P1) * 1000 \geq (VC + V0)^2 / 4 * RT$, GO TO 2'

$$I = \frac{(VC + V0) - \sqrt{(VC + V0)^2 - 4 * RT * (P2 - P1) * 1000}}{2 * RT}$$

Go to 3)

- 2)' $I = (VC + V0) / 2 * RT$ and write DIAGNOSTIC

- 3) Compute VT

$$VT = VC + V0 - I * RT$$

- 4) Potential energy balance and power loss

$$PE = -(I + VC / RL) (VC + V0) / 1000.$$

$$PL = (I^2 * RT + VC^2 / RL) / 1000.$$

- 5) Maximum charging and discharging rates

$$RE2 = \text{MIN}(MP1, RAP, (E1 - PE) / TINC) / EF1$$

$$MP2 = \text{MIN}(RAP, (VC + V0)^2 / (4000 * RT), (PE - EDE) / TINC)$$

where TINC = integration step size

Calculation Sequence Cont.

6) Priority interrupt logic

If $PE \leq EDE$ and $T0 = 10^6$, $T0 = TIME$
 If $PE \leq EDE$ and $TIME - T0 \geq DT$, $INT = 1$ and go to 7)
 $T0 = 10^6$
 If $PE > 2 \times EDE$ and $INT = 1$, $INT = 0$
 If $PE \geq E1$, $INT = -1$
 If $PE < E1 - EDE$ and $INT = -1$, $INT = 0$

7) Compute Statistics and Costs

BA

```

00101 21* C VC -CAPACITOR VOLTAGE, VOLTS 000000
00101 22* C VT -TERMINAL VOLTAGE, VOLTS 000000
00101 23* C PL -POWER LOSS, KW 000000
00101 24* C T0 TIME WHEN BATTERY WAS DISCHARGED, HR 000000
00101 25* C MP2 -MAXIMUM OUTPUT POWER, KW 000000
00101 26* C INT -PRIORITY INTERRUPT FLAG 000000
00101 27* C REZ -MAXIMUM CHARGING RATE REQUEST, KW 000000
00101 28* C STATISTICS 000000
00101 29* C SPC -SUM OF CHARGING ENERGY, KWH 000000
00101 30* C MPE -MAXIMUM STORED ENERGY, KWH 000000
00101 31* C SPD -SUM OF DISCHARGING ENERGY, KWH 000000
00101 32* C INPUTS 000000
00101 33* C P1 -INPUT POWER, KW 000000
00101 34* C V0 -INTERNAL VOLTAGE, VOLTS 000000
00101 35* C RT -TERMINAL RESISTANCE, OHMS 000000
00101 36* C CB -BATTERY CAPACITANCE, FARADS 000000
00101 37* C RL -LEAKAGE RESISTANCE, OHMS 000000
00101 38* C RAP -RATED INPUT POWER, KW 000000
00101 39* C EFI -INPUT PRODUCT EFFICIENCY 000000
00101 40* C MP1 -MAXIMUM INPUT POWER, KW 000000
00101 41* C E1 -MAXIMUM ENERGY STORAGE, KWH 000000
00101 42* C RE1 -POWER REQUEST, KW 000000
00101 43* C EOE -ENERGY DEADEND FOR PRIORITY RESEQUENCING, KWH 000000
00101 44* C DT -DOWNTIME FOR PRIORITY RESEQUENCING, HR 000000
00101 45* C CC -CAPITAL COST/YEAR, $ 000000
00101 46* C CM -MAINTENANCE COST/YEAR, $ 000000
00101 47* C
00103 48* COMMON /CINPL/INPL,TCNT/CTIME/TIME/CSIMUL/QUIN1,1,MAX/COST/CC1,CM1 000000
00104 49* REAL I,MP2,MPE,MP1,INT 000000
00105 50* DOUBLE PRECISION AA,B,RZ 000000
00106 51* TINC1=QUIN1*.5 000000
00106 52* C 000000
00107 53* IF(INPL.GT.0) GO TO 100 000000
00111 54* IF(CT.EQ..99999)RT=.001 000000
00113 55* IF(CB.EQ..99999)CB=2.0ME8 000012
00115 56* IF(PL.EQ..99999)RL=.25 000017
00117 57* T0=1000000. 000024
00120 58* INT=0 000026
00121 59* TMAX1=TMAX+.99999 000027
00122 60* 000032
00122 61* MPE=0. 000032
00123 62* SPC=0. 000033
00124 63* SPD=0. 000034
00124 64* C 000034
00124 65* C CAPACITOR VOLTAGE 000034
00124 66* C 000034
00125 67* 100 VC=50RT*(7.2E6)*PE/CB + V0*.2) -V0 000034
00125 68* C 000036
00125 69* C TERMINAL CURRENT 000036
00125 70* C 000036
00126 71* P2=PE1 000036
00127 72* AA=(P2-P1)*4000.*RT 000051
00130 73* B=VC+V0 000053
00131 74* R2=R+R 000062
00131 75* IF(AA.GT.0) GO TO 200 000066
00131 76* I= B-DSORT(82-AA) 000070
00131 77* I=I/(2.*RT) 000073
00131 78* 000103

```

```

00136 76*      GO TO 300
00136 79*      C
00137 80*      200 I=B/(2.*RT)
00140 81*      IF (IMPL.EQ.2)WRITE(6,200)P2
00144 82*      204 FORMAT(1H0,15H POWER REQUEST ,F12.3,50H EXCEEDS BATTERY CAPABILITY
00144 83*      1. CHECK VC,VO, AND RT. )
00145 84*      IF (IMPL.EQ.2)ICNT=ICNT+1
00145 85*      C
00145 86*      C      TERMINAL VOLTAGE
00145 87*      C
00147 88*      300 VT=VC+VO-I*RT
00147 89*      C
00147 90*      C      POTENTIAL ENERGY BALANCE AND ENERGY LOSS
00147 91*      C
00150 92*      IF (PE.NE.0)PEO=1-I-VC/RL)*(VC+VO)/1000.
00152 93*      PL=(1+I*RT+VC+VO/RL)/1000.
00152 94*      C
00152 95*      C      MAXIMUM CHARGING AND DISCHARGING RATES
00152 96*      C
00153 97*      AP1=AMAX1(0.,(E1-PE)/DUM17)I
00154 98*      RE2=AMIN1(IMPI,RAP,AP)I
00155 99*      RE2=RE2/EF1
00156 100*      AP2=AMAX1(0.,(PE-EDE)/DUM17)I
00157 101*      MP2=AMIN1(RAP,82/14000.*RT),AP2)
00157 102*      C
00157 103*      C      PRIORITY INTERRUPT
00157 104*      C
00160 105*      CE1=EDE
00161 106*      ED2=(DE+EDE
00162 107*      IF (PE.GT.EDF)60 TO 401
00164 108*      IF (TO.GT.999999.)TO=TIME
00166 109*      WAIT=TIME-TO
00167 110*      IF (WAIT.GT.DT)INT=)
00171 111*      GO TO 400
00172 112*      401 TO=1000000
00173 113*      IF (PE.LE.ED2)60 TO 400
00175 114*      IF (PE.GT.E1)60 TO 403
00177 115*      IF (PE.GT.C1)60 TO 400
00201 116*      INT=0
00202 117*      GO TO 400
00203 118*      403 INT=-1
00204 119*      400 CONTINUE
00204 120*      C
00205 121*      IF (IMPL.EE.1)RETURN
00205 122*      C
00205 123*      C      STATISTICS
00205 124*      C
00207 125*      MPE=AMAX1(MPE,PE)I
00210 126*      SPC=SPC+TIMEI+P)I
00211 127*      SPD=SPD+TIMEI+P2)I
00211 128*      C
00212 129*      IF (TIME.LT.TMAX)RETURN
00214 130*      CCJ=CCJ+CC
00215 131*      CMI=CMI+CM
00215 132*      C
00216 133*      RETURN
00217 134*      END

```

```

000107
000107
000111
000116
000127
000127
000127
000127
000136
000136
000136
000136
000144
000154
000154
000154
000154
000154
000165
000175
000207
000211
000221
000221
000221
000221
000241
000244
000247
000253
000261
000264
000272
000274
000275
000301
000303
000311
000312
000314
000316
000316
000316
000316
000316
000324
000332
000336
000336
000342
000351
000354
000354
000357
000356

```

BA

7.4 BURNER

The burner model computes the amount of fuel required to be burned in the inlet airstream to raise the air temperature from the given inlet temperature to the specified outlet temperature. The fuel mass flow rate when integrated over time allows calculation of the cost of burner fuel.

Basic Equation

The mass of fuel consumed, F, is computed from the equation:

$$F = \frac{M1 * CP * (T2 - T1)}{NU * HF}$$

Inputs

<u>Parameter/Port</u>		<u>Description</u>	<u>Units</u>
M	1	Inlet air mass flow rate	lb/h
CP		Air heat capacity ($D = 72 \times 10^{-6}$)	kwh/lb-°F
T	1	Inlet air temperature	°F
T	3	Outlet air temperature (specified)	°F
NU		Combustor efficiency ($D = 0.98$)	-
HF		Fuel heating value ($D = 5.56$)	kwh/lb
CF		Specific fuel cost ($D = 0.094$)	\$/lb
FDM		Maximum allowable fuel mass flow rate ($D=17800$)	lb/h
CB		Burner cost coefficient ($D = 1.683$)	\$/lb/h
LE		Burner life expectancy	years
MDM		Maximum allowable air mass flow rate ($D=27000$)	lb/h
EF	1	Input product efficiency	-
MP	1	Maximum input power	kw
P	1	Input power	kw

Outputs

<u>Variable/Port</u>			
F		Fuel mass consumed (state)	lb
EF	2	Output product efficiency	-
MP	2	Maximum output power	kw
T	2	Outlet air temperature	°F
FD		Fuel mass flow rate	lb/h
CCO		Burner capital cost/year	\$
CO		Fuel cost	\$
M	2	Outlet mass flow rate (= M1)	lb/h
P	2	Output power	kw

Statistics

FDM		Maximum fuel mass flow rate	lb/h
-----	--	-----------------------------	------

0 - Default values supplied

The calculation sequence and default values are based on a burner sized using first principles to maintain the outlet temperature at 600°F assuming an inlet temperature of 120°F and a mass flowrate of 2.7×10^4 lb/h. These conditions represent the extreme conditions expected and should satisfy all burner requirements. No. 6 fuel oil is assumed to be the fuel type. Cost and heating values were obtained from References 1 and 2. Cost estimates for the burner were estimated from the results of Reference 1.

Calculation Sequence

1) Capital Cost

$$CC0 = CB * MDM / LE$$

2) Maximum air mass flow rate allowed

If $M1 = 0$ set $EFF = 1$, $MP2 = MP1$ and go to 3)

$$M1M = \min \left\{ \frac{NU * HE * FDM}{CP * (T3 - T1)}, MDM \right\}$$

If $T1 > T3$, $M1M = MDM$

3) Efficiency and maximum discharge power

$$EFF = 1 + M1 * CP * (T2 - T1) / P1 \quad (\text{if } P1 > 0)$$

$$EF2 = EF1 * EFF$$

$$MP2 = \min \{ MP1 * EFF, P1 * M1M / M1 \} \quad (\text{if } M1 > 0)$$

$$P2 = P1 * EFF$$

1. "Preliminary Feasibility Evaluation of Compressed Air Storage Power Systems," United Technologies AER 74-00242, December 1976.
2. Steam, Its Generation and Use, Babcock and Wilcox, New York, NY, 1972.

Calculation Sequence Cont.

- 4) Fuel mass flow rate

$$F = \frac{M1 * CP * (T2 - T1)}{NU * HF}$$

$$T2 = \text{MAX}(T1, T3)$$

If $M1 > M1M$ write DIAGNOSTIC

- 5) Compute Statistics and Costs

$$C0 = CF * F$$

BZ

00150	81*		IF(M1.GT.0.) MP2=AMTM1(MP1*EFF,P1*M1M/M1)	000125
00152	82*		P2=P1*EFF	000142
00152	83*	C		000142
00152	84*	C	FUEL FLOW RATE	000142
00152	85*	C		000142
00153	86*		IF(1F.NE.0) DF= M1*CP*(T2-T1)/NU/HF	000145
00155	87*		IF(T1.GT.T2) DF=D.0	000157
00157	88*		FD=DF	000164
00157	89*	C		000164
00157	90*	C	COSTS	000164
00160	91*		CO = CF*F	000166
00161	92*		T3= AMAX1(T1,T2)	000171
00162	93*		M2=M1	000177
00162	94*	C		000177
00162	95*	C	STATISTICS	000177
00163	96*		IF(1MPL.LE.1) RETURN	000201
00163	97*	C		000201
00165	98*		FDU = AMAX1(FD,FOUR)	000210
00165	99*	C		000210
00166	100*		IF(1IME.LT.TMAX1) RETURN	000216
00170	101*		CC1= CC1 + CC	000225
00171	102*		COP= COP + CO	000230
00172	103*		RETURN	000233
00172	104*	C		000233
00173	105*		1000 IF(1MPL.EQ.2)WRITE(6,1010) M1,M1M	000237
00200	106*		1010 FORMAT(1H0,2HMBM INLET AIR MASS FLOW RATE ,F12.3,	000250
00200	107*		1 36H GREATER THAN MAXIMUM ALLOWABLE ,F12.3)	000250
00201	108*		IF(1MPL.EQ.2)ICNT=ICNT+1	000250
00203	109*		GO TO 100	000256
00204	110*		END	000260

7.5 COST MONITOR¹

This component sums the capital, operating and maintenance costs of all system components. The total yearly cost TC is then computed using a fixed charge rate factor which represents depreciation, cost of money, insurance and taxes.

¹ This component must be placed last in the model generation input file, i.e., just prior to the END OF MODEL command.

The total wind energy delivered to the loads plus surplus energy is then summed and yearly energy delivered TED computed. Cost of operation in mills is then given by

$$\text{Wind system cost/kwh} = \text{TC} * 1000./\text{TED}$$

Similarly, the value of energy delivered to the loads is summed minus the utility energy value and including the value of surplus energy, and factored to give yearly energy value delivered VED. Energy value in mills is given by

$$\text{Load value/kwh} = \text{VED} * 1000./\text{TED}.$$

Cost per value delivered is the ratio of the above two equations.

In addition to the above cost calculations, percent of total load supplied by wind storage PCW, percent of load supplied by utilities PCU and, percent of wind energy surplus to the utilities PCS is computed. The total cost in mills to meet the load is then given by

$$\text{Load cost/kwh} = (\text{wind system cost/kwh} * \text{PCW} + \text{utility cost/kwh} * \text{PCU})/100.,$$

where

$$\text{Utility cost/kwh} = \text{value of utility energy} * 1000./\text{utility energy delivered}.$$

Inputs

<u>Parameter/Port</u>	<u>Description</u>	<u>Units</u>
CR	Capital charge rate	%/year
LE	System life expectancy	years

Common Block Inputs

CC	Total yearly capital costs	\$
CM	Total yearly maintenance costs	\$
CO	Operating and fuel costs over TMAX	\$
TMAX	Simulation time interval	hr
VDE	Value of energy delivered (including surplus)	\$
TDE	Wind energy delivered (including surplus)	kwh
TLD	Total load demand	kwh
UTV	Value of utility energy	\$
UTD	Utility energy delivered	kwh
SPD	Surplus wind energy supplied	kwh

Outputs ¹

Total yearly costs (TC)	\$
Yearly energy delivered (TED)	kwh
Cost of energy per kwh	mills
Yearly value delivered (VED)	\$
Cost per value delivered	-
Percent of load supplied by	
Wind Storage (PCW)	-
Utility (PCU)	-
Surplus energy load factor (PCS)	-
Total load cost per kwh	mills

¹ Printout only occurs when simulation is completed. Thus no output variable symbol is required.

SUBROUTINE CM ENTRY POINT 000213

STORAGE USED CODE(11) 000226; DATA(10) 000330; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 COST 000011
0004 CEMPL 000001
0005 CTIME 000001
0006 CSIMUL 000010

EXTERNAL REFERENCES (BLOCK, NAME)

0007 NMMN
0010 NIN24
0011 NERRIS

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000020 150L	0000 000016 200F	0003 000034 300F	0000 000105 400F	0000 000210 500F
0003 R 000000 CC	0000 R 000003 CCY	0003 R 000001 CMA	0003 R 000002 CO	0003 R 000032 CNY
0000 R 000015 CPKWH	0000 R 000011 CPY	0006 000000 DUM	0000 R 000005 EDE	0004 000000 IMPL
0000 000013 INJPS	0000 000006 TYPE	0000 000001 LLE	0000 R 000012 PCB	0000 R 000014 PCS
0000 R 000013 PCU	0003 R 000010 SPQ	0003 R 000004 TOE	0005 R 000000 TIME	0003 R 000005 TLD
0006 R 000007 THAX	0000 R 000000 THAX1	0000 R 000004 TOY	0000 R 000007 TOYN	0003 R 000007 UTO
0003 R 000006 UTY	0003 R 000003 VDE	0000 R 000010 VDEN		

00100	1*	COST		000000
00101	2*		SUBROUTINE CM(CUMM,FCR,LE)	000000
00101	3*	C		000000
00101	4*	C	PURPOSE	000000
00101	5*	C	SUMMARIZE WIND ENERGY STORAGE COSTS AND LEVELIZED	000000
00101	6*	C	ENERGY COSTS PER KWH.	000000
00101	7*	C		000000
00101	8*	C	WRITTEN BY A.W. WARREN	000000
00101	9*	C	VERSION 1, MAY 1977	000000
00101	10*	C	INPUT PARAMETERS	000000
00101	11*	C		000000
00101	12*	C	FCR - FIXED CHARGE RATE FACTOR INCLUDING DEPRECIATION,	000000
00101	13*	C	MONEY COST, INSURANCE, AND TAXES, PER YEAR	000000
00101	14*	C	LE - SYSTEM LIFE EXPECTANCY, YEARS	000000
00101	15*	C	THAX - SIMULATION TIME, HR	000000
00101	16*	C	CC - TOTAL YEARLY CAPITAL COSTS, \$	000000
00101	17*	C	CM - TOTAL YEARLY MAINTENANCE COSTS, \$	000000
00101	18*	C	CO - TOTAL OPERATING AND FUEL COSTS OVER THAX, \$	000000
00101	19*	C	VDE - VALUE OF ENERGY DELIVERED OVER THAX, \$	000000
00101	20*	C	TOE - TOTAL ENERGY DELIVERED OVER THAX, KWH	000000
00101	21*	C	TLD - TOTAL LOAD DEMAND OVER THAX, KWH	000000

CM

00101	21*	C		UTV - VALUE OF UTILITY ENERGY SUPPLIED LESS SURPLUS VALUE, \$	000000
00101	22*	C		UTD - TOTAL UTILITY ENERGY DELIVERED, KWH	000000
00101	23*	C		SPN - TOTAL SURPLUS ENERGY SUPPLIED TO UTILITY, \$	000000
00101	24*	C			000000
00103	25*			COMMON /COST/ CC, CMA, CO, VDE, TDE, TLO, UTV, UTD, SPD	000000
00104	26*			COMMON /CIMPL/ INH, L /CTIME/ TIME /CSIMUL/ DUM(7), THAX	000000
00105	27*			REAL LE	000000
00105	28*	C		INITIALIZATION	000000
00105	29*	C			000000
00106	30*			IF (INPL.GT.0) GO TO 100	000000
00110	31*			DUMM=0.0	000000
00111	32*			CC = 0.	000000
00112	33*			CMA = 0.	000000
00113	34*			CO = 0.	000000
00114	35*			VDE = 0.	000000
00115	36*			TDE = 0.	000000
00116	37*			TLO = 0.	000000
00117	38*			UTV = 0.	000000
00120	39*			UTD = 0.	000000
00121	40*			SPD = 0.	000000
00122	41*			THAX = THAX + .99999	000000
00122	42*	C			000000
00123	43*		100	IF (TIME.LT.THAX) RETURN	000000
00125	44*			IF (IMPL.LE.1) RETURN	000000
00125	45*	C			000000
00125	46*	C		COST SUMMARY OUTPUT	000000
00125	47*	C			000000
00127	48*			LLE = LE	000000
00130	49*			WRITE(6,200)LLE	000000
00133	50*		200	FORMAT(1H,35X,33H WIND ENERGY STORAGE COST SUMMARY // 1H ,40X,12,	000000
00133	51*			1 17M YEAR LIFE CYCLE)	000000
00133	52*	C			000000
00134	53*			COY = CO * 8760. / THAX	000000
00135	54*			CCY = CC * LE * FCR * .01	000000
00136	55*			TOY = COY + CMA + CCY	000000
00137	56*			WRITE(6,300)CCY,CMA,COY,TOY	000000
00145	57*		300	FORMAT(1/// 30X,22HD YEARLY SYSTEM COSTS/ 1H*,29X,1H*/ 1H-,42X,	000000
00145	58*			1 12HCAPITAL COST,12X,F8.0,2H % / 1H ,42X, 17HINCLUDING FIXED ,	000000
00145	59*			2 8HCARGES) / 1H*,42X,16HFIXED O + M COST, BX,F8.0,2H % / 1H ,	000000
00145	60*			3 42X,21HOPERATING + FUEL COST, 3X,F8.0,2H % / 1H*,42X,5HTOTAL,	000000
00145	61*			4 19X,F8.0,2H %)	000000
00145	62*	C			000000
00146	63*			EDE = TDE * 8760. / THAX	000000
00147	64*			IVDE = VDE * 8760. / THAX	000000
00150	65*			TOYN = TOY * 1000. / EDE	000000
00151	66*			VDEH = VDE * 1000. / TDE	000000
00152	67*			CPV = TOYN / VDEH	000000
00152	68*	C			000000
00153	69*			WRITE(6,400)EDE,TOYN,IVDE,VDEH,CPV	000000
00162	70*		400	FORMAT(1/// 30X,26HD ENERGY DELIVERED / 1H*,29X,1H* / 1H-,	000000
00162	71*			1 42X,16HEMPGY DELIVERED, 7X,F9.0,4H KWH / 1H*,33X,50F1M=I /	000000
00162	72*			1 1H ,33X,1H*,48X,1H* /	000000
00162	73*			2 1H ,33X,1H*, BX,19HEMPGY COST PER KWH, 7X,F6.1,9H MILLS * /	000000
00162	74*			2 1H ,33X,1H*,48X,1H* / 1H ,	000000
00162	75*			3 33X,10(5H*****) / 1H*,42X,25HVALUE OF ENERGY DELIVERED,17,	000000
00162	76*			4 2H % / 1H ,42X,22HVALUE OF FUEL SAVED) / 1H*,42X,20HEMPGY VALUE	000000
00162	77*			5 PER KWH, 6X,F6.1,6H MILLS / 1H*,42X,24HCOST PER VALUE DELIVERED,	000000

00162	78*	4 2H,F6.2)	000144
00162	79*	C	000144
00163	81*	PCD= (TDE-SPD)*100./TLD	000144
00164	81*	PCU= UTD*100./TLD	000150
00165	82*	PCS= SPD*100./TLD	000154
00166	83*	CPKWH= (TOYN*ITDE-SPD) + UTY*1000./TLD	000160
00167	84*	WRITE(6,500)PCD,PCU,PCS,CPKWH	000167
00175	85*	500 FORMAT(/// 30X,3IND LOAD FACTOR / 1H*,29X,	000200
00175	86*	1 1H* / 1H*,42X,	000200
00175	87*	3 26HPERCENT OF LOAD SUPPLIED , F6.1, 2H / 1H ,42X,22HBY TOTAL WIN	000200
00175	88*	2ND SYSTEM / 1H,42X,24HPERCENT OF LOAD SUPPLIED,2X,F6.1 /	000200
00175	89*	2 1H ,41X,11H BY UTILITY /	000200
00175	90*	3 1H,42X,26HPERCENT OF WIND ENERGY , F6.1 /	000200
00175	91*	3 1H ,42X,9HSUPPLUSED /	000200
00175	92*	4 1H,42X,23HCOST TO MEET LOAD , 3X,F6.1,6H MILLS/	000200
00175	93*	5 1H ,42X,27HAWIND + UTILITY / 1H1)	000200
00175	94*	C	000200
00176	95*	RETURN	000200
00177	96*	END	000225

7.6 COMPRESSOR (PNEUMATIC)

The compressor model represents the off-design performance of a typical axial flow compressor. The compressor is assumed designed for a specified set of design operating conditions and performance requirements. The mass flow rate is assumed directly proportional to angular velocity and independent of the pressure ratio across the compressor. This is expected to hold for $\pm 15\%$ of the design mass flow rate. The polytropic efficiency of the compressor is assumed to be a weak function of the angular velocity. Initial calculations are made with the design polytropic efficiency, and refinements made after the off-design parameters are calculated.

Basic Equations

The expression for the angular velocity is

$$RS = P1 * \frac{RSD}{MD} \frac{EFF}{CP * (TA + 460) * [(PR2/PA)^A - 1]}$$

where:

$$EFF = ((PR2/PA)^{(A*NP)} - 1.) / ((PR2/PA)^A - 1.)$$

$$A = (GAM - 1) / GAM * NP$$

Inputs

<u>Parameter/Port</u>		<u>Description</u>	<u>Units</u>
P	1	Input power	kw
RSD		Design angular velocity (D = 3600)	rpm
MD		Design mass flow rate (D = 3000)	lb/h
CP		Heat capacity of air (D = 7.2×10^{-5})	kwh/lb ^o F
TA		Inlet air temperature (ambient) (D = 70)	^o F
PR	2	Exit pressure (D = 147)	psi
PA		Inlet pressure (ambient) (D = 14.7)	psi
GAM		Heat capacity ratio (D = 1.4)	-
EF	1	Input product efficiency	-
MP	1	Maximum input power	kw
NPD		Design polytropic efficiency (D = 0.88)	-
PID		Design inlet pressure (ambient) (D = 14.7)	psi
POD		Design outlet pressure (D = 147)	psi
TID		Design inlet air temp (ambient) (D = 70)	^o F
CK		Compressor capacity cost coefficient ¹ (D = 1.0)	-
FO		Compressor exponent for cost calculations (D = 0.75)	-

Outputs

<u>Variable/Port</u>			
NP		Polytropic efficiency	-
EF	2	Output product efficiency	-
MP	2	Maximum output power	kw
TQ		Torque	ft-lb
M		Mass flow rate	lb/h
T	2	Exit temperature	^o F
RS		Angular velocity	rpm
CCO		Cost of compressor/year	\$

¹ CK = capital cost (known unit)/(design point mass flow rate)^{FO} *
LN (outlet/inlet pressure ratio)*(life expectancy of unit)
D = Default values supplied

<u>Statistics</u>	<u>Description</u>	<u>Units</u>
MT	Maximum temperature	°F
MF	Maximum mass flow rate	lb/h

The calculation sequence and the default values are based on the assumption of an axial flow compressor, nominally rated at 125kw, and a pressure ratio of 10. The equations used relate first order effects among the various physical quantities and were derived from first principles originally to support the research work of Reference 1. Cost scaling was also developed in that reference based on cost estimates obtained from turbomachinery manufacturers.

Calculation Sequence

1) Costs (First pass only)

$$CCO = CK * (MD)^{**F0} * LN \left(\frac{P0D}{PID} \right)$$

If P1 > 0 go to 2)

$$A = (GAM-1)/(GAM*NPD)$$

$$RAT = (P0D/PID)^{**A}$$

$$EFF = 1, RS = 0, \text{ go to 3)}$$

-
1. "Closed Cycle High Temperature Central Receiver Concept for Solar Electric Power," BEC/EPR! RP 377-1, June 1976.

Calculation Sequence Cont.

2) Angular velocity iteration

$$A = (GAM-1)/(GAM*NP) \quad (\text{initially } NP = NPD)$$

$$RAT = (PR2/PA)**A$$

$$EFF = (RAT**NP-1)/(RAT-1)$$

$$\frac{RS}{RSD} = \frac{P1}{MD \cdot CP \cdot (TA+460)} \cdot \frac{EFF}{(RAT-1)}$$

Polytropic efficiency

$$NP = 1 - (1 - NPD) * [2.0 - \frac{P1D * (TA+460) * RSD}{PA * (T1D+460) * RS}]^{0.2}$$

Iterate until NP and RS are consistent

(If iteration doesn't converge, then write DIAGNOSTIC and exit)

3) Mass flow rate

$$M = MD * RS / RSD$$

4) Exit temperature

$$T2 = (TA+460)*RAT - 460$$

5) Torque

If $P1 \leq 0$, set $T0 = 0$ and go to 6)

$$T0 = P1 * 737.6 / (RS * 2 * 60)$$

6) Efficiency and maximum power

$$EF2 = EF1 * EFF$$

$$MP2 = \min(MP1 * EFF, 1.5 * MD * CP * (T2 - TA))$$

7) Compute Statistics and Costs

SUBROUTINE CO

ENTRY POINT 000420

STORAGE USED CODE(1) 000613; DATA(1) 000066; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 C1PPL 000001
 0004 C1TME 000001
 0005 C1TMDL 000010
 0006 C0ST 000021

EXTERNAL REFERENCES (BLOCK, NAME)

0007 XPRR
 0010 AL06
 0011 N1004
 0012 N1024
 0013 N10P4
 0014 NERR34

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000136 100L	0001 000375 100L	0000 000010 101CF	0001 000166 200L	0001 000271 300L
0001 000315 100L	0000 R 000005 A	0006 R 000000 CC1	0003 000000 00K	0030 R 000003 EFF
0003 I 000000 IMPL	0000 000052 IN,PS	0000 I 000002 ISP	0000 R 000000 PT	0000 R 000004 RAT
0000 R 000006 RSHO	0004 R 000000 TIME	0003 R 000007 THAX	0000 R 000001 THAX1	0000 R 000007 XMP

00100	1*	CCO		000000
00101	2*		SUBROUTINE CO(MP,EF2,MP2,TO,M,TZ,NS,CC,MT,MF,P1,RSD,MD,CP,TA,PR2,	000000
00101	3*		PR1,GAN,CF1,MP1,MPD,PID,P0D,PID,CK,F01	000000
00101	4*	C		000000
00101	5*	C	PURPOSE PERFORMANCE MODEL OF AXIAL FLOW COMPRESSOR	000000
00101	6*	C		000000
00101	7*	C	METHOD COMPRESSOR IS SIZED FROM INPUT OPERATING REQUIREMENTS.	000000
00101	8*	C		000000
00101	9*	C	MASS FLOW IS ASSUMED PROPORTIONAL TO ANGULAR VELOCITY	000000
00101	10*	C		000000
00101	11*	C	AND INDEPENDENT OF PRESSURE RATIO.	000000
00101	12*	C		000000
00101	13*	C	WRITTEN BY F.O. MAHONY	000000
00101	14*	C	VERSION 1, MARCH 22 1977	000000
00101	15*	C	CALL SEQUENCE	000000
00101	16*	C	OUTPUTS	000000
00101	17*	C	MP - POLYTROPIC EFFICIENCY	000000
00101	18*	C	EF2 - OUTPUT PRODUCT EFFICIENCY	000000
00101	19*	C	MP2 - MAXIMUM OUTPUT POWER, KW	000000
00101	20*	C	TO - TORQUE, FT-LB	000000

3


```

00152 78* C AND ANGULAR VELOCITY
00152 79* C
00153 80* ISP = 0
00153 81* C
00154 82* EFF=1.0
00155 83* IF(P1.GT.0.0160 TO 200
00155 84* C
00157 85* RATE = (P0D/PID)**((IGAM-1.0)/(IGAM+MPD))
00160 86* T0 = 0.0
00161 87* RSD=0.0
00162 88* MP=MPD
00163 89* 60 TO 300
00163 90* C
00164 91* 200 A = (IGAM-1.0)/(IGAM+MP)
00165 92* RATE = (PR2/PP1)**A
00166 93* EFF=(RATE**MP - 1.1/(RATE - 1.1)
00166 94* C
00167 95* RSN0 = EFF*P1/MD*1.0/CP/(TA+460.01/(RATE-1.0)
00167 96* C
00170 97* XMP = MP
00170 98* C
00171 99* MP = 1.0-11.0-MPD*(2.0-(PID/PR1*(TA+460.01/ITID+460.01/RSN0)
00171 100* 1 **0.2)
00171 101* C
00172 102* IF(ISP.GT.10) 60 TO 1000
00172 103* C
00174 104* ISP = ISP+1
00174 105* C
00175 106* IF(ABS((MP-XMP)/MP).GT.0.001) 60 TO 200
00177 107* RSC = RSD+RSN0
00177 108* C
00177 109* C MASS FLOW RATE
00177 110* C
00200 111* 300 M = MD*RS/RSD
00200 112* C
00200 113* C EXIT TEMPERATURE
00200 114* C
00201 115* T2 = (TA+460.01)+RATE*460.0
00202 116* IF(P1.LE.0.0160 TO 400
00202 117* C
00202 118* C TORQUE
00202 119* C
00204 120* T0 = P1*737.6/(RS*2.0*PI/60.0)
00204 121* C
00204 122* C EFFICIENCY AND MAXIMUM POWER
00204 123* C
00204 124* C
00204 125* C
00205 126* 400 EF2 = EF1*EFF
00205 127* C
00206 128* MP2 = AMIN(MP1*EFF,1.5*MD*CP*(T2-TA))
00206 129* C
00206 130* C STATISTICS AND COST SUMMATION
00206 131* C
00207 132* IF (IMPL.LE.1) RETURN
00211 132* MT = AMAX(MT, T2)
00212 133* MF = AMAX(MF, M)
00213 134* IF(TIME.LT.TMAX) RETURN

```

```

000136
000136
000136
000136
000136
000140
000140
000143
000160
000161
000162
000164
000164
000166
000173
000203
000263
000214
000214
000224
000224
000226
000226
000226
000250
000250
000254
000254
000257
000265
000265
000265
000271
000271
000271
000274
000301
000301
000301
000304
000304
000304
000304
000315
000315
000317
000317
000317
000334
000343
000351
000357

```

```

00215 135*      CCI = CCI + CC
00215 136*      C
00216 137*      RETURN
00216 138*      C
00217 139*      1000 WRITE (6,1010) NP,XNP,RS
00224 140*      1010 FORMAT (1H0,40HMAX ITERATIONS FOR COMPRESSOR EFFICIENCY,
00224 141*      1      15H - NP,XNP,RS = ,3F12.6)
00225 142*      STOP
00226 143*      END

```

```

000366
000366
000371
000371
000375
000404
000404
000404
000404
000617

```

7.7 PNEUMATIC STORAGE VESSEL (CONSTANT PRESSURE)

The pneumatic storage vessel is based on a constant pressure underground cavern design as represented in Figure 7.7. A surface pressure-compensation pond via a water shaft is assumed to maintain the vessel pressure at a constant value. This model is assumed to be used in conjunction with a heat exchanger. The energy is calculated as a function of the stored gas mass, the inlet/storage air temperature, and a leakage function proportional to the stored energy.

Basic Equation

The rate of energy storage is computed from the equation

$$\dot{E} = M1 * CP * (T1 - T0) - NU * E, \text{ charging}$$

$$\dot{E} = -M2 * CP * (T2 - T0) - NU * E, \text{ discharging}$$

where $M1$ = mass flow rate during charge

$M2$ = mass flow rate during discharge

FIGURE 7.7 CONSTANT PRESSURE AIR STORAGE

Inputs

<u>Parameter/Port</u>		<u>Description</u>	<u>Units</u>
M	1	Inlet air mass flow rate	lb/h
CP		Air heat capacity ($D = 7.2 \times 10^{-5}$)	kwh/lb ^o F
T	1	Inlet air temperature	^o F
TO		Minimum air temperature ($D = 60$)	^o F
NU		Leakage coefficient ($D = 0.0008$)	h ⁻¹
R		Gas constant ($D = 2.009 \times 10^{-5}$)	kwh/lb
VM		Maximum storage capacity ($D = 1.2 \times 10^6$)	ft ³
PR	1	Vessel pressure ($D = 147$)	psi
LE		Life expectancy of vessel	years
CV		Vessel capacity cost ($D = 0.22$)	\$/ft ³
RE	1	Load request	kw
EF	1	Input product efficiency	-
MP	1	Input maximum power	kw
MDE		Mass threshold for priority resequencing	lb
MD		Maximum charge or discharge mass flow rate	lb
TM		Maximum allowable air temperature ($D = 120$)	^o F
TEA		Maximum allowable inlet temperature	^o F
CM		Maintenance cost/year	\$

OutputsVariable/Port

E		Stored energy (state)	kwh
M	2	Outlet mass flowrate	lb/hr
T	2	Storage temperature	^o F
V		Storage volume	ft ³
COO		Cost of vessel/year	\$
MS		Mass of air in storage (state)	lb
MP	2	Maximum output power	kw
RE	2	Maximum charging rate	kw
INT		Interrupt priority flag	-
TSO		Minimum air temperature (=TO)	^o F

D - Default values supplied

Outputs Cont.Variable/Port

PR	2	Vessel pressure (=PR1)	psi
P	2	Output power (discharge)	kw
NDM		Maximum allowable mass flow rate (=ND)	

Statistics

EU		Maximum stored energy	kwh
VU		Maximum storage volume	ft ³

The pneumatic storage vessel calculation sequence and default values assume a 10atm cavern approximately 340 ft. below ground and sized for storage of 120kw for 24 hours. A maximum cavern wall temperature of 120°F is assumed. Cost estimates for the vessel were estimated from the results of Reference 1, with cost scaling by .05 to account for plant size differences.

-
1. "Preliminary Feasibility Evaluation of Compressed Air Storage Power Systems," United Technologies AER 74-00242, December 1976.

Calculation Sequence

TINC = Integration step size, hr

C1 = conversion constant = 5.43×10^{-5}

$\frac{\text{kwh/ft}^3}{\text{psi}}$

- 1) Vessel Cost

$$CC0 = CV * VM / LE$$

- 2) Storage temperature

$$T2 = \frac{E}{CP * MS} + T0$$

- 3) Storage volume

$$V = MS * \frac{R * (T2 + 460)}{PR1 * C1}$$

- 4) Maximum Mass and charging rate

$$MSM = VM * \frac{(PR1 * C1)}{R * (T2 + 460)}$$

$$MD1 = \text{MIN}(MDM, (MSM - MS) / TINC)$$

$$RE2 = \min \left[MP1, MD1 * CP * (TEM - T0) \right] / EF1$$

- 5) Mass flow out (discharge mode)

$$M2 = \frac{RE1}{CP * (T2 - T0)}$$

$$P2 = RE1$$

- 6) Maximum discharge rate

$$MD = \text{MIN}(MDM, (MS - MDE) / TINC)$$

$$MP2 = CP * (T2 - T0) * MD$$

Calculation Sequence Cont.

7) Stored energy rate

$$\dot{E} = CP*(T_1 - T_0)*M_1 - NU*E - RE_1$$

8) Stored mass rate

$$\dot{M}_S = M_1 - M_2$$

9) Priority interrupt

If $M_S \leq MDE$, $INT = 1$ If $M_S > 2*MDE$ and $INT = 1$, $INT = 0$ If $M_S \geq MSM$, $INT = -1$ If $M_S < MSM - MDE$ and $INT = -1$, $INT = 0$ If $T_2 > T_M$ write diagnostic and set $INT = -1$ If $M_S < MDE$ or $M_S > MSM$ write diagnostic

10) Compute Statistics and Costs

SUBROUTINE CS ENTRY POINT 000427

STORAGE USED CODE(1) 000437; DATA(0) 000125; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 C1MPL 000002
0004 C1TME 000001
0005 C1MUL 000010
0006 C0ST 000002

EXTERNAL REFERENCES (BLOCK, NAME)

0007 NUNUN
0010 NJ021
0011 NERP31

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000326 10L	0001 000074 100L	0000 000004 1000F	0000 000021 1010F	0000 000036 1020F
0001 000344 20L	0001 000362 200L	0006 R 000000 CCI	0006 R 000001 CMI	0000 R 000003 CI
0005 R 000000 DUM	0003 I 000001 ICNT	0003 I 000000 IMPL	0000 000072 INJP1	0000 R 000001 MD1
0000 R 000000 MSN	0004 R 000000 TIME	0005 R 000007 IMAX	0000 R 000002 IMAX1	

00100	1*	CCS		000000
00101	2*		SUBROUTINE CS(E,DE,IE,MS,DMS,IMS,M2,T2,V,CC,MP2,RE2,INT,T30,PR2	000000
00101	3*		1 ,P,MDN,EU,VU,M1,CP,T,TD,R,VH,PR1,LE,MU,CV,RE1,EF1,MP1,MOE,MD,TM	000000
00101	4*		2 ,TEM,CN)	000000
00101	5*	C		000000
00101	6*	C	PURPOSE PERFORMANCE MODEL OF CONSTANT PRESSURE STORAGE VESSEL	000000
00101	7*	C		000000
00101	8*	C	METHOD ENERGY IN STORAGE COMPUTED AS A FUNCTION MASS AND	000000
00101	9*	C		000000
00101	10*	C	INLET TEMPERATURE.	000000
00101	11*	C		000000
00101	12*	C	WRITTEN BY F.O. HANONY	000000
00101	13*	C	VERSION 1, MARCH 23 1977	000000
00101	14*	C	CALL SEQUENCE	000000
00101	15*	C		000000
00101	16*	C	OUTPUTS	000000
00101	17*	C	E - STORED ENERGY (STATE VARIABLE), KWH	000000
00101	18*	C	DE - STORED ENERGY DERIVATIVE, KW	000000
00101	19*	C	IE - STATUS INDICATOR FOR E	000000
00101	20*	C	MS - MASS OF AIR IN STORAGE (STATE VARIABLE), LB	000000
00101	21*	C	DMS - AIR FLOW RATE, LB/HR	000000
00101	22*	C	IMS - STATUS INDICATOR FOR MS	000000
00101	23*	C	M2 - OUTLET MASS FLOWRATE, LB/HR	000000

CS

```
COMMON /CINPL/INPL,ICNT/CTIME/ TIME /CSIMUL/DUM177,IMAX
COMMON /COST/ CCI,CMI
REAL NU,MS,MP2,INT,MOM,M1,LE,MP1,MDE,MD,M2,MSN,MD1
```

IF (IMPL.GT.9) GO TO 100

```
IF(CP.EQ..99999) CP = 72.0E+6
IF(TM.EQ..99999) TM= 120.0
IF(TO.EQ..99999) TO = 60.0
IF(MU.EQ..99999) MU = 0.0000
IF(R.EQ..99999) R = 2.039E-5
IF(WM.EQ..99999) WM = 1.2E+6
IF(PRI.EQ..99999) PRI = 147.0
IF(CV.EQ..99999) CV = 0.22
9E)=0.0
```

TRAIL = IMAGE01.00000

150 = 10

CC = CV*VM/LE
C1 = 5.43E-5

```
T2 - STORAGE TEMPERATURE, DEG F
V - STORAGE VOLUME, FT**3
CC - COST OF VESSEL/YEAR, $
MP2 - MAXIMUM OUTPUT POWER, KW
REZ - MAXIMUM CHARGING RATE, MW
INT - INTERRUPT PRIORITY FLAG
TSO - MINIMUM AIR TEMPERATURE, DEG F
PR2 - VESSEL PRESSURE, PSI
P - OUTPUT POWER (DISCHARGE), KW
MDM - MAXIMUM ALLOWABLE MASS FLOW RATE, LB/MR
EU - MAXIMUM STORED ENERGY, KWH
VU - MAXIMUM STORAGE VOLUME, FT**3
```

INPUTS

```

MI - INLET AIR MASS FLOW RATE, LB/HR
CP - AIR HEAT CAPACITY, KWH/LB-DEG F
T - INLET AIR TEMPERATURE, DEG F
TO - MINIMUM AID TEMPERATURE, DEG F
R - GAS CONSTANT, KWH/LB-DEG R
VM - MAXIMUM STORAGE CAPACITY, FT**3
PRI - VESSEL PRESSURE, PSI
LE - LIFE EXPECTANCY OF VESSEL, YEARS
NU - LEAKAGE COEFFICIENT . . 1/HR
CV - VESSEL CAPACITY COST , $/FT**3
REI - LOAD REQUEST, KW
EFI - INPUT PRODUCT EFFICIENCY
MPI - INPUT MAXIMUM POWER, KW
MOE - RESERVOIR THRESHOLD MASS FOR PRIORITY
      RESEQUENCING, LB
MO - MAXIMUM CHARGE / DISCHARGE MASS FLOW RATE, LB/HR
TM - MAXIMUM ALLOWABLE STORAGE TEMPERATURE, DEG F
TEM - MAXIMUM ALLOWABLE INLET TEMPERATURE, DEG F
CM - MAINTENANCE COST / YEAR . $

```

C00006
C00007
C00008
C00009
C00010
C00011
C00012
C00013
C00014
C00015
C00016
C00017
C00018
C00019
C00020
C00021
C00022
C00023
C00024
C00025
C00026
C00027
C00028
C00029
C00030
C00031
C00032
C00033
C00034
C00035
C00036
C00037
C00038
C00039
C00040
C00041
C00042
C00043
C00044
C00045
C00046
C00047
C00048
C00049
C00050
C00051
C00052
C00053
C00054
C00055
C00056
C00057
C00058
C00059
C00060

[illegible]

00214	138*	20	IFIMS.LI.NSM) 60 TO 200	000344
00216	139*		WRITE(6,1020) NS,MSH	000347
00222	140*		ICNT=ICNT+1	000356
00222	141*	C		000356
00222	142*	C	STATISTICS	000356
00222	143*	C		000356
00223	144*	200	EU = AMAX1(EU,CI	000362
00224	145*		VU = AMAX1(VU,VI	000367
00224	146*	C		000367
00225	147*		IFITIME.LI.TMAX) RETURN	000375
00227	148*		CCI = CCI+CC	000404
00230	149*		CHI=CHI+CH	000407
00233	150*	C		000407
00231	151*		RETURN	000412
00232	152*	1000	FORMAT(1ND,25HCS STORAGE TEMPERATURE,F12.3,18HGREATER THAN ALLOW,	000436
00232	153*		1 5HABLE , F12.3)	000436
00233	154*	1010	FORMAT(1ND,25HCS MASS OF AIR IN STORAGE,F12.3,	000436
00233	155*		1 26H BELOW MINIMUM ALLOWABLE,F12.3)	000436
00234	156*	1020	FORMAT(1ND,25HCS MASS OF AIR IN STORAGE,F12.3,	000436
00234	157*		1 28H EXCEEDS MAXIMUM ALLOWABLE,F12.3)	000436
00235	158*		END	000436

7.8 FLYWHEEL/CLUTCH

The flywheel model is a first order differential equation for kinetic energy which is driven by input power when charging and by a load request when discharging. Power losses include clutch losses versus shaft speed and torque, windage losses, and friction losses due to bearing and seals. Shaft speed is determined analytically from kinetic energy. Priority interrupt logic is activated if minimum or maximum capacity levels are reached.

Basic Equations

$$KE = k * \omega^2$$

$$\dot{KE} = P_{IN} - P_{OUT} - C_1 * \omega - C_2 * \omega^{2.8},$$

where k, C_1, C_2 are flywheel constants

ω = rotor speed in rad/sec

P_{IN} = input power - clutch losses

P_{OUT} = output load request

<u>Tables</u>	<u>Description</u>	<u>Units</u>
CL0	Clutch losses versus rotor speed (rpm) and torque (ft-lb), when engaged (Table dimension = 90)	kw
CL1	Clutch losses versus rotor speed (rpm) when disengaged (Table dimension = 17)	kw

Inputs

<u>Parameter/Port</u>			
PR		Pressure in vacuum housing	psi
HM		Moment of inertia ¹	slug-ft ²
RF		Radius of flywheel	ft
SR		Shaft radius	ft
WT		Flywheel weight	lb
KF		Coefficient of friction	-
ZE		Width of flywheel at tip	ft
C2		Windage loss coefficient (analytic default)	-
P	1	Input power	kw
EF	1	Input product efficiency	-
MP	1	Input maximum charging rate	kw
RAP		Rated power, charge or discharge	kw
RE	1	Discharge load request	kw
E0		Minimum allowable storage capacity	kwh
E1		Maximum allowable storage capacity	kwh
EDE		Energy deadband for priority resequencing	kwh
CM		Maintenance cost/year	\$
CC		Capital cost/year	\$

Outputs

<u>Variable/Port</u>		
RS		Rotor speed
KE		Kinetic energy (state)

¹ Includes physical drive system.

Outputs Cont.

<u>Variable/Port</u>	<u>Description</u>	<u>Units</u>
T0	Input torque (charging)	ft-lb
T1	Output torque (discharging)	ft-lb
P 2	Output power	kw
PL0	Clutch losses (charging)	kw
PL1	Clutch losses (discharging)	kw
EF 2	Output efficiency	-
MP 2	Maximum output power	kw
INT	Priority interrupt flag	-
RE 2	Maximum charging power request	kw

Statistics

ME	Maximum stored energy	kwh
MPC	Maximum charge rate	kw
MPD	Maximum discharge rate	kw
SPC	Sum of charging energy	kwh
SPD	Sum of discharging energy	kwh

Calculation Sequence

- 1) Compute flywheel constants

$$k = \frac{1}{2} * HW * 3.76616 * 10^{-7}$$

$$C_1 = KF * WT * SR * 1.3558 * 10^{-3}$$

$$C_2 = C_0 * PR^{0.8} * RF^{4.6} * (1 + 2.3 * ZE / RF) \quad (\text{DEFAULT})$$

$$C_0 = 1.0946 * 10^{-7}$$

If $KE < E0$ or $KE > E1$ write diagnostic

- 2) Compute rotor speed

$$\omega = \sqrt{KE/k}$$

$$RS = \omega * (60 / 2\pi)$$

$$P_{IN} = 0$$

- 3) Compute power losses and net power when charging

If $P1 = 0$, set $T0 = PL0 = P_{IN} = 0$ and go to 4)

$$T0 = P1 * 737.6 / \omega$$

$$PL0 = CL0(RS, T0)$$

$$P_{IN} = P1 - PL0$$

If $P_{IN} < 0$, write diagnostic

- 4) Compute power losses and output power when discharging

If $RE1 = 0$, set $T1 = PL1 = P2 = P_{OUT} = 0$ and go to 5)

$$T1 = RE1 * 737.6 / \omega$$

$$PL1 = CL0(RS, -T1)$$

Calculation Sequence

4) Cont.

$$P_2 = RE1 - PL1$$

$$P_{OUT} = RE1$$

If $P_2 < 0$, set $P_2 = 0$. and write diagnostic

5) Compute power losses when disengaged

If $P_1 > 0$ or $RE1 > 0$, go to 6)

$$P_{OUT} = CL1(RS)$$

6) Flywheel kinetic energy rate

$$\dot{KE} = P_{IN} - P_{OUT} - C_1 * \omega - C_2 * \omega^{2.8}$$

7) Maximum Input (charging power)

$$TM = MP1 * 737.6 / \omega$$

$$MP0 = MP1 - CL0(RS, TM)$$

If $MP0 \leq 0$, write diagnostic and go to 8)

$$EF0 = EF1 * MP0 / MP1$$

$$RE2 = \text{MIN}(MP0, RAP1, (E1 - KE) / TINC) / EF0$$

8) Output efficiency and maximum power

$$RAP1 = \text{MIN}(RAP, (KE - E0) / TINC), \quad TINC = \text{integration step}$$

$$TM = RAP1 * 737.6 / \omega$$

$$MP2 = RAP1 - CL0(RS, -TM)$$

Calculation Sequence

8) Cont.

If $MP2 < 0$ write diagnostic

$$EF2 = MP2/RAP1$$

If $RE1 > 0$, $EF2 = P2/RE1$

9) Priority interrupt logic

If $KE \leq E0$, $INT = 1$

If $KE > E0 + EDE$ and $INT=1$, $INT=0$

If $KE \geq E1$, $INT = -1$

If $KE < E1 - EDE$ and $INT = -1$, $INT=0$

10) Compute Statistics and Costs

SUBROUTINE FL ENTRY POINT 001325

STORAGE USED CODE(1) 001333; DATA(9) 000231; BLANK COMMON(2) 000000

COMMON BLOCKS

```

0003 CIMPL 000002
3004 CTIME 000001
0005 CSIMUL 000010
0006 COST 000002

```

EXTERNAL REFERENCES (BLOCK, NAME)

```

0007 TBLU2
0010 TBLU1
0011 XPRR
0012 NWDS
0013 NIDZS
0014 SQRT
0015 NERR35

```

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000060 10L	0000 000042 108F	0000 000024 109F	0001 000147 20L	0001 000315 200L
0000 000060 208F	0001 000412 300L	0000 000074 308F	0001 000440 400L	0001 000544 500L
0000 000110 508F	0001 000574 600L	0001 000705 700L	0000 000125 708F	0000 R 000005 AK
0000 R 000022 APC	0006 R 000000 CCI	3006 R 000001 CHI	0000 R 000006 CI	0000 R 000000 BWM
0009 R 000025 ECA	0000 R 000024 ECI	0000 R 000021 EFA	0003 I 000001 ICNT	0003 I 000000 IMPL
0000 000166 INJPS	0000 I 000012 MN4	0000 R 000001 MPA	0000 R 000002 MPB	0000 R 000000 PPA
0000 I 000011 MN	0000 I 000013 MNHRS	0000 I 000014 MNHRS4	0000 I 000007 MNR5	0000 I 000011 MNT
0000 R 000015 OMEGA	0000 R 000016 PIN	0000 R 000017 POUT	0000 R 000023 RAPT	0010 R 000000 TBLU1
0007 R 000000 TBLU2	0004 R 000000 TIME	0000 R 000003 TIME	0000 R 000020 TH	0005 R 000007 THAY
0000 R 000004 IMAT1				

```

00100 1* CFL
00101 2*
00101 14 SUBROUTINE FL(CLO,CL1,R5,NE,KED,TKE,TO,T1,P2,PLO,PL1,EF2,MP2,INT,
00101 4* 1 REZ,NE,MPC,NPD,SPC,SPD, PR,NM,RF,SR,VT,KF,ZE,C2,P1,EF1,MP1,RAP
00101 5* 2 ,REL,ZO,EL,EOE,CH,CC)
00101 5* C
00101 6* C PURPOSE MODEL OF FLYWHEEL CAPABLE OF ABSORBING POWER
00101 7* C AND OF DELIVERING POWER ON REQUEST.
00101 8* C
00101 9* C METHOD OUTPUT POWER AND KINETIC ENERGY COMPUTED FROM
00103 10* POWER REQUEST AND INPUT POWER
00103 11* C
00103 12* C
00103 13*

```

```

000007
000007
000007
000007
000007
000007
000007
000007
000007
000007
000007
000007
000007
000007
000007

```

[illegible]

```

00114 710 C
00115 720 RE=0.
00116 730 RE1=0.
00117 740 MPC=0.
00120 750 MP0=0.
00121 760 SPC=0.
00122 770 SPO=0.
00123 780 10 CONTINUE
00124 790 AK=.5*HN*.76616E-7
00125 800 C1=MPC*MT*SR*.3558E-3
00126 810 IF(1/KE.GT.T0).AND.(KE.LT.E1).OR.(IMPL.NE.2)GO TO 20
00130 820 IF(KE.LE.E0)WRITE(6,100)KE,E0
00135 830 IF(KE.GE.C1)WRITE(6,100)KE,C1
00140 840 100 FORMAT(1H0,20H FLYWHEEL KINETIC ENERGY ,F12.3,
00142 850 X 10H EXCEEDS CAPACITY ,F12.3)
00143 860 100 FORMAT(1H0,24H FLYWHEEL KINETIC ENERGY,F12.3,
00144 870 X 10H FALLS BELOW MINIMUM REQUIREMENT ,F12.3)
00145 880 ICNT=ICNT+1
00146 890 20 CONTINUE
00147 900 NARS=C(012)
00150 910 NNT=C(013)
00151 920 NN4=NN4+NARS
00152 930 NNNRS=C(112)
00153 940 NNNRS4=NNNRS+4
00154 950 T0=0.
00155 960 T1=0.
00156 970 P2=0.
00157 980 PLO=0.
00160 990 P1=0.
00161 1000 RF2=0.
00161 1010
00161 1020 C
00161 1030 C COMPUTE MOTOR SPEED
00161 1040 C
00162 1050 100 OMEGA=1.E-6
00163 1060 IF(KE.GT.C.10)OMEGA=SQRT(KE/AN)
00165 1070 RSE=OMEGA*30./3.14159
00166 1080 PIN=0.
00166 1090 C
00166 1100 C COMPUTE POWER LOSSES AND NET POWER WHEN CHARGING
00166 1110 C
00167 1120 IF(P1.EQ.0.160 TO 200
00171 1130 TO=P1*737.6/OMEGA
00172 1140 PLO=TELU2(RS,T0,CLO(MN),CLOIN1,CLO(MN),1,1,-NARS,-NNT,MNRS,NNT)
00173 1150 PIN=P1-PLO
00174 1160 POUT=0.
00174 1170 C
00175 1180 IF(PIN.GE.0.160 TO 200
00177 1190 IF(IMPL.NE.2)GO TO 200
00181 1200 WRITE(6,200)PLO,P1
00205 1210 200 FORMAT(1H0,21H FLYWHEEL POWER LOSS ,F12.3,
00206 1220 X24H EXCEEDS CHARGING POWER ,F12.3)
00206 1230 ICNT=ICNT+1
00206 1240 C
00206 1250 C COMPUTE POWER LOSSES AND OUTPUT POWER WHEN DISCHARGING
00206 1260 C
00207 1270 200 IF(RE1.EQ.0.160 TO 300

```

```

000046
000051
000052
000053
000054
000055
000056
000060
000060
000063
000070
000115
000130
000143
000143
000143
000143
000147
000147
000155
000164
000166
000170
000177
000201
000202
000203
000204
000205
000206
000206
000206
000206
000207
000211
000223
000227
000227
000227
000230
000236
000242
000272
000274
000274
000275
000277
000302
000311
000311
000311
000311
000311
000315

```

```

00211 128*      T1=RE1*737.6/OMEGA
00212 129*      P1)=TBLU2IRS,-1),CLO(MN4),CLO(4),CLO(MN4),1,1,-NNRS,-NMT,NNRS,NNTI)
00213 130*      P2=RE1-P1
00214 131*      POUT=RE1
00215 132*      IF(P2.GT.0..OR.IMPL.NE.2)GO TO 300
00217 133*      WRITE(6,300)P1,RE1
00223 134*      300 FORMAT(1NC,16H FLYWHEEL LOSS ,F12.3,
00223 135*      X27H EXCEEDS DISCHARGING POWER ,F12.3)
00224 136*      ICNT=ICNT+1
00225 137*      P2=0.
00225 138*      C
00225 139*      C      COMPUTE POWER LOSSES WHEN DISENGAGED
00225 140*      C
00226 141*
00226 142*
00226 143*      300 IF(P1.GT.0.)GO TO 400
00230 144*      IF(IRE).GT.0.)GO TO 400
00232 145*      POUT=TBLU2IRS,CL(44),CL(INNRS4),1,-NNRS1)
00232 146*      C
00232 147*      C      FLYWHEEL KINETIC ENERGY BALANCE
00232 148*      C
00233 149*      400 IF((KE.NE.0)KCO=PIN-POUT-C1*OMEGA-C2*(OMEGA**2.0)
00233 150*      C
00233 151*      C      MAXIMUM CHARGING POWER REQUEST
00233 152*      C
00235 153*      TM=MP1*737.6/OMEGA
00236 154*      MPA=TBLU2IRS,TM,CLO(MN4),CLO(4),CLO(MN4),1,1,-NNRS,-NMT,
00236 155*      X      NNRS,NNTI)
00237 156*      MPO=MP1-MPA
00240 157*      IF(MPO.GT.0.)GO TO 500
00242 158*      IF(IMPL.EQ.2)WRITE(6,500)MPA,MP1
00247 159*      500 FORMAT(1H0,22H FLYWHEEL CLUTCH LOSS ,F12.3,
00247 160*      X 31H EXCEEDS MAXIMUM INPUT POWER ,F12.3)
00250 161*      IF(IMPL.EQ.2)ICNT=ICNT+1
00252 162*      GO TO 600
00253 163*      500 EFO=E1+MPO/MP1
00254 164*      APC=ANAX1(0.,(E1-KE)/DUNIT1)
00255 165*      RE2=AMIN(MPO,RAP,APC)
00256 166*      RE2=RE2/EFO
00256 167*      C
00256 168*      C      OUTPUT EFFICIENCY AND MAXIMUM POWER
00256 169*      C
00257 170*      600 RAPT=(KE-E0)/(TINC*2.)
00260 171*      RAPT=AMIN(RAPT,RAP)
00261 172*      RAPT=ANAX1(RAPT,RAP/100.)
00262 173*      TM=RAPT*737.6/OMEGA
00263 174*      MFD=TBLU2IRS,-1M,CLO(MN4),CLO(4),CLO(MN4),1,1,-NNRS,-NMT,
00263 175*      X      NNRS,NNTI)
00264 176*      MP2=RAPT-MFD
00265 177*      IF(MP2.GT.0..OR.IMPL.NE.2)GO TO 700
00267 178*      700 FORMAT(1H0,22H FLYWHEEL CLUTCH LOSS ,F12.3,
00267 179*      X27H EXCEEDS DELIVERABLE POWER ,F12.3)
00270 180*      WRITE(6,700)MP2,RAPT
00274 181*      ICNT=ICNT+1
00275 182*      700 MP2=ANAX1(MP2,RAP/100.)
00276 183*      EF2=MP2/RAPT
00276 184*      IF(IRE).GT.0..AND.P2.GT.0.)EF2=P2/RE1

```

```

000314
000322
000357
000361
000363
000376
000405
000405
000405
000410
000410
000410
000412
000412
000414
000421
000421
000421
000421
000440
000440
000440
000440
000456
000462
000462
000516
000520
000522
000534
000534
000534
000542
000544
000547
000557
000571
000571
000571
000571
000574
000601
000606
000615
000620
000620
000635
000637
000672
000672
000672
000701
000705
000713
000715

```

FL

00277	185*	C		000715
00277	186*	C	PRIORITY INTERRUPT	000715
00277	187*	C		000715
00301	188*		EC1=EC1-EDE	000734
00302	189*		EC0=EC0+EDE	000737
00303	190*		IF (INE.GT.EC0).AND.(INT.EQ.1.)INT=0.	000742
00325	191*		IF ((KE.LT.EC1).AND.(INT.EQ.-1.))INT=0.	000760
00327	192*			000776
00327	193*		IF (INE.LE.EC0)INT=1.	000776
00311	194*		IF (INE.GT.EC1)INT=-1.	001004
00313	195*		IF ((INE.GT.EC0).AND.(INE.LT.EC1))INT=0.	001012
00315	196*		IF (IMPL.LE.1)RETURN	001031
00315	197*	C		001031
00315	198*	C	STATISTICS	001031
00315	199*	C		001031
00317	200*		ME=AMAX1(ME,KE)	001040
00320	201*		MPC=AMAX1(MPC,KE0)	001046
00321	202*		MPC=AMAX1(MPC,-ME0)	001054
00322	203*		SPC=SPC+TIME*P1	001062
00323	204*		SPD=SPD+TIME*P2	001066
00323	205*	C		001066
00324	206*		IF (TIME.LT.TMAX)RETURN	001072
00326	207*		CCI=CCI+CC	001101
00327	208*		CHI=CHI+CM	001104
00327	209*	C		001104
00330	210*			001107
00330	211*		RETURN	001107
00331	212*		END	001332

7.9 ONE DIMENSION TABLE LOOKUP

Tables

FTA

Description

Tabular values of function

Inputs

Parameter/Port

FIN

Input quantity

AN

$ABS(AN) \leq 0.5$ for equispaced interpolation
($AN < 0$ prevents extrapolation)

Outputs

Variable/Port

F0

Output quantity

Calculation Sequence

$$F0 = FTA(FIN)$$

NOTE: A maximum of 18 points is allowed in the table.

SUBROUTINE FU ENTRY POINT 000065

STORAGE USED CODE(1) 000075; DATA(2) 000014; BLANK COMMON(2) 000000

EXTERNAL REFERENCES (BLOCK, NAME)

0003 TBLU1
0004 NEPR35

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0000 000005 INJPS 0000 1 000002 N 0000 1 000000 NA 0000 1 000001 NB 0003 * 000020 TBLU1

00100	1*	CFU				000003
00101	2*		SUBROUTINE FU(FTA,FO,FIN,AN)			000003
00101	3*	C				000003
00101	4*	C	PURPOSE - TO CALCULATE OUTPUT FO AS AN ARBITRARY FUNCTION OF			000003
00101	5*	C	INPUT FIN USING TABULAR INPUT FTA GIVING FO=F(FIN)			000003
00101	6*	C				000003
00101	7*	C	METHOD - SELF EXPLANATORY			000003
00101	8*	C				000003
00101	9*	C				000003
00101	10*	C	LIMITATIONS - MAXIMUM ARRAY SIZE IS 18			000003
00101	11*	C				000003
00101	12*	C				000003
00101	13*	C	WRITTEN BY - ADAM LLOYD	LATEST REVISION	APRIL 77	000003
00101	14*	C				000003
00101	15*	C				000003
00101	16*	C	INPUT/OUTPUT LIST			000003
00101	17*	C				000003
00101	18*	C	FTA	TABULAR INPUT FO=F(FIN)	ANY	INPUT TABLE
00101	19*	C	FO	OUTPUT	ANY	OUTPUT VAR
00101	20*	C	FIN	INPUT	ANY	INPUT VAR
00101	21*	C	AN	SET ABS(AN).GT.0.5 FOR UNEQUAL SPACED TABLE DATA---INPUT		000003
00101	22*	C		SET ABS(AN).LE.0.5 FOR EQUI-SPACED TABLE DATA		000003
00101	23*	C		A NEGATIVE VALUE OF AN WILL		000003
00101	24*	C		PREVENT EXTRAPOLATION BEYOND		000003
00101	25*	C		TABLE LIMITS		000003
00101	26*	C				000003
00103	27*		DIMENSION FTA(1)			000003
00104	28*		NA= SIGNIF(12),ANI			000003
00105	29*		NP=FTA(2)+4			000015
00106	30*		N=1			000026
00107	31*		IF(ABS(AN).LE.0.5) N=0			000030
00111	32*		FO=TBLU1(FIN,FTA(4),FTA(NB),N,NA)			000035
00112	33*		RETURN			000053
00113	34*		END			000074

FU

7.10 TWO DIMENSION TABLE LOOKUP

Tables

Description

FTA Table of functional relationships (maximum number of table values = 144)

Inputs

Parameter/Port

FNA	Input quantity (primary)
FNB	Input quantity (secondary)
AN	$ABS(AN) \leq 0.5$ for equal spaced FNA data*
BN	$ABS(BN) \leq 0.5$ for equal spaced FNB data*

Outputs

Variable/Port

FQ	Output quantity
----	-----------------

Calculation Sequence

$$FQ = FTA(FNA, FNB)$$

* A negative value for AN or BN prevents extrapolation beyond the table boundaries.

00107	34*	MN=FTA(3)	000035
00110	35*	M5= SIGN(FTA(2),AN)	000044
00111	36*	M6= SIGN(FTA(3),BN)	000056
00112	37*	MAN=1	000070
00113	38*	IF (ABS(AN)).LE.D.5) MAN=0	000072
00115	39*	MPN=1	000077
00116	40*	IF (ABS(BN)).LE.D.5) MBN=0	000101
00120	41*	FO=16LU2(IFNA,FNB,FTA(1),FTA(4),FTA(2),MAN,MBN,M5,M6,M3,M4)	000114
00121	42*	RETURN	000142
00122	43*	END	000203

7.11 AC INDUCTION GENERATOR

The induction generator produces electrical power proportional to rotor slip, i.e. difference between rotor speed and synchronous speed. This relationship is used to compute rotor speed given input power and the generator parameters. Two power losses are modeled: a constant multiplicative term due to resistive heating and an additive term due to mechanical friction. Default parameters are based on a conventional squirrel-cage induction motor/generator machine. This component can also be used as a synchronous generator with $RAS \leq 0.1$.

Basic Equations

Output power P2 and rotor speed RS are computed from the following equations:

$$P2 = EE * (P1 - C * RS^2)$$

$$\frac{P2}{RAP} = \frac{(RS/RSY - 1)}{RAS} \quad (\text{Power is proportional to slip})$$

where EE = electrical efficiency

C = constant of conversion

Inputs

<u>Parameter/Port</u>		<u>Description</u>	<u>Units</u>
P	1	Input power	kw
RAP		Rated output power	kw
RSY		Synchronous rotor speed (D = 1800)	rpm
RAS		Rated power slip (D = 0.05)	-
DA		Mechanical damping (D = 0.0)	joule-sec
SR		Internal stator resistance (D = 6.4/RAP)	ohms
V0		Rated bus voltage (D = 400)	volts
EF	1	Input product efficiency	-
MP	1	Maximum input discharge rate (D = 1×10^8)	kw
CC		Capital cost/year	\$
CM		Maintenance cost/year	\$

Outputs

<u>Variable/Port</u>			
P	2	Output power	kw
EE		Electrical efficiency	-
RS		Rotor speed	rpm
PL		Power loss	kw
EF	2	Output product efficiency	-
MP	2	Maximum output discharge rate	kw

Statistics

MPN		Maximum output power/rated power	kw
SP		Total output energy	kwh

D - Default values supplied.

Calculation Sequence

- 1) First pass only

$$EFF = 1$$

$$I_{RAT} = RAP * 1000 / V\phi$$

$$EE = \frac{RAP}{RAP + SR * I_{RAT}^2 / 1000}$$

- 2) If $P1 = 0$ set $P2 = 0$, $RS = RSY$ and go to 4)

Compute rotor speed ω in rad/sec using

$$\frac{EE(P1 * 1000 - \omega^2 * DA)}{RAP * 1000} = \frac{(\omega / \omega_o - 1)}{RAS}$$

with $\omega_o = RSY * (2\pi / 60)$

- 3) Compute RS and output power

$$RS = \omega * (60 / 2\pi)$$

$$P2 = RAP(RS / RSY - 1) / RAS$$

$$P2 > RAP \Rightarrow \text{DIAGNOSTIC}$$

$$EFF = P2 / P1$$

- 4) Compute loss, efficiency terms

$$PL = P1 - P2$$

$$EF2 = EF1 * EFF$$

- 5) Compute maximum output rate

$$MP2 = \text{MIN}(RAP, MP1 * EFF)$$

- 6) Compute Statistics and Costs

T

67

00160	81*	C		
00161	82*			
00163	83*			
00164	84*			
00164	85*	C		
00165	86*			
00166	87*			

COST SUMMATION
 IF1 TIME.LT.TMAX11 RETURN
 CC = CC + CCI
 CM = CM + CMI
 RETURN
 END

C00221
 000726
 C00235
 C00240
 C00240
 C00243
 000242

7.12 FIXED RATIO TRANSMISSION

This component models a fixed gear ratio transmission. Power losses are modeled by a table lookup depending on input power. Rotor input speed is used as a feedback variable.

FIGURE 7.12: FIXED GEAR POWER LOSS

<u>Tables</u>	<u>Description</u>	<u>Units</u>
PLØ	Power loss versus input power	kw

Inputs

Parameter/Port

GR		Gear ratio	-
RS	2	Output rotor speed	rpm
P	1	Input power	kw
EF	1	Input product efficiency	-
MP	1	Maximum input power (Default = 1×10^8)	kw
CC		Capital cost/year	\$
CM		Maintenance cost/year	\$

Outputs

Variable/Port

P	2	Output power	kw
TØ		Output torque	ft-lb
PL		Power loss	kw
EF	2	Output product efficiency	-
MP	2	Maximum output power	kw
RS	1	Input rotor speed	rpm

Calculation Sequence

1) $MP2 = MP1 - PL0(MP1)$ (First Pass Only)

If $P1 \leq 0$, set $PL = P2 = 0$ and go to 2)

$$PL = PL0(P1)$$

$$P2 = P1 - PL$$

$$RS1 = RS2/GR$$

$$EF2 = EF1 * P2 / P1$$

2) $T0 = P2 * 737.6 / (RS2 * 2 \pi / 60)$

3) Compute Costs

SUBROUTINE GR ENTRY POINT 00017

STORAGE USED CODE(1) 000237; DATA(1) 000023; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 C1MPL 000001
0004 CTIME 000001
0005 FOST 000004
0006 CSIMUL 000010

EXTERNAL REFERENCES (BLOCK, NAME)

0007 TBLU1
0010 NEPR36

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001	000057	10L	0001	000113	20L	0005	R	000000	CC	0005	R	000001	CN	0005	000002	CO
0005	000003	CV	0006	000000	DUM	0003	I	000000	IMPL	0000	000007	INJP3	0000	I	000000	NP
0007	R	000000	TBLU1	0004	R	000000	TIME	0006	R	000007	TMAX	0000	R	000001	TMAX1	

00100	1*	CGR		000000
00101	2*		SUBROUTINE GR(PLO,P2,T0,PL,EF2,PM2,RS1,GR4,RS2,P1,EF1,PM1,CCI,CHI)	000000
00101	3*	C		000000
00101	4*	C	PURPOSE MODEL A FIXED GEAR RATIO TRANSMISSION	000000
00101	5*	C		000000
00101	6*	C	METHOD POWER LOSSES ARE INPUT AS A FUNCTION OF INPUT POWER P1.	000000
00101	7*	C		000000
00101	8*	C	WRITTEN BY A.W. WARREN	000000
00101	9*	C	VERSION 1, MARCH 16 1977	000000
00101	10*	C	CALL SEQUENCE	000000
00101	11*	C	TABLES	000000
00101	12*	C	PLO - POWER LOSS IN KW VERSUS INPUT POWER IN KW	000000
00101	13*	C		000000
00101	14*	C	OUTPUTS	000000
00101	15*	C	P2 - OUTPUT POWER, KW	000000
00101	16*	C	T0 - OUTPUT TORQUE, FT-LB	000000
00101	17*	C	PL - POWER LOSS, KW	000000
00101	18*	C	EF2 - OUTPUT PRODUCT EFFICIENCY	000000
00101	19*	C	PM2 - MAXIMUM OUTPUT POWER, KW	000000
00101	20*	C	RS1 - INPUT ROTOR SPEED, RPM	000000
00101	21*	C		000000
00101	22*	C	INPUTS	000000
00101	23*	C	GR4 - GEAR RATIO	000000
00101	24*	C	RS2 - OUTPUT ROTOR SPEED, RPM	000000
00101	25*	C	P1 - INPUT POWER, KW	000000

00101	26*	C	EFI - INPUT PRODUCT EFFICIENCY	000000
00102	27*	C	PM1 - MAXIMUM INPUT POWER, KW	000000
00103	28*	C	CCI - CAPITAL COST / YEAR, \$	000000
00104	29*	C	CMI - MAINTENANCE COST / YEAR, \$	000000
00105	30*	C		000000
00106	31*		DIMENSION PLO(1)	000000
00107	32*		COMMON /CINPL/INPL /CTIME/ TIME	000000
00108	33*		COMMON /COST/CC,CM,CO,CV /CSIMUL/ DUNIT1,TMAX	000000
00109	34*	C		000000
00110	35*	C	INITIALIZATION	000000
00111	36*	C		000000
00112	37*		NP = PLO(2)	000000
00113	38*		IF(INPL.GT.0) GO TO 10	000007
00114	39*		EF2 = 1.	000012
00115	40*		TMAX1 = .99999*TMAX	000014
00116	41*		RS2 = 1.	000021
00117	42*		IF(PM1.LE. .99999) PM1 = 1.E10	000023
00118	43*		PM2 = PM1	000030
00119	44*		IF(PM1.LE. PLO(3)+NP)) PM2 = PM1-TBLU(PM1,PLO(4),PLO(4+NP),1,-NP)	000032
00120	45*	C		000032
00121	46*	C	POWER LOSS AND ROTOR SPEED CALCULATIONS	000032
00122	47*		10 PL=0.	000057
00123	48*		P2=0.	000057
00124	49*		IF(P1 .EQ. 0.) GO TO 20	000060
00125	50*		PL = TBLU(P1,PLO(4),PLO(4+NP),1,-NP)	000062
00126	51*		P2 = P1 - PL	000102
00127	52*		EF2 = (EF1+P2/ P1	000104
00128	53*		RS1 = RS2/8PA	000107
00129	54*		20 IF(RS2 .GT. 0.) TO = P2*704%./RS2	000113
00130	55*	C		000113
00131	56*	C	COST SUMMATION	000113
00132	57*		IF(INPL.LE.1) RETURN	000121
00133	58*		IF(TIME.LE.TMAX) RETURN	000130
00134	59*		CC = CC + CCI	000137
00135	60*		CM = CM + CMI	000142
00136	61*		RETURN	000145
00137	62*		END	000236

7.13 HISTOGRAM

The input quantity is monitored during a SIMULATE analysis. When time reaches TMAX a plotted histogram is produced with 16 intervals that span the range from FLO to FUP.

Inputs

<u>Variable/Port</u>	<u>Description</u>
FIN	Input quantity to be monitored
FUP	Upper limit for histogram
FLO	Lower limit for histogram
F1,...,F16 ¹	Array containing histogram data
FA ¹	Measurement interval

Outputs

<u>Variable/Port</u>	
AV	Mean value (running sum during simulation)
SD	Standard deviation (running sum squared)
SAM	Number of samples

¹ These quantities do not require data input values.

ENTRY POINT 000430

COMMON BLOCKS

0003	CTIME	000001
0004	CSIMUL	000010
0005	CORLY	000004
0006	CIMPL	000001

EXTERNAL REFERENCES 18BLOCK, NAME)

0007	SGRT
0010	HMWU
0011	N1038
0012	N1028
0013	N1012
0014	N18938

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001	000023	100L	0000	000052	100CF	0000	000056	1100F	0001	000012	120G	0000	000062	1200F
0000	000067	1300F	0001	000034	134G	0000	000073	140CF	0001	000111	154G	0001	000154	166G
0001	000175	177G	0001	000203	205G	0001	000211	213G	0001	000216	220G	0001	000223	225G
0001	000224	230G	0001	000234	236G	0001	000257	243G	0001	000272	246G	0001	000306	255G
0001	000314	262G	0001	000053	300L	0001	000364	303G	0001	000364	305G	0000	000047	900F
0000	R 000010	AX1	0000	R 000030	BLANK	0000	000003	CPUSEC	0000	000000	DUM	0000	000000	DUMH
0000	R 000036	FAX	0002	R 000000	GRAPH	0000	R 000032	HORIZ	0000	R 000041	HX	0000	I 000034	I
0000	I 000043	IC	0006	I 000000	IMPL	0000	000135	INJPS	0000	I 000037	ISAMP	0000	I 000042	J
0000	I 000044	J1	0000	I 000045	J2	0000	I 000046	J3	0000	I 000035	L	0000	R 000033	POTNT
0000	R 000000	IND	0003	R 000000	TIME	0004	R 000007	IMAX	0000	R 000031	VERT	0000	R 000040	YMAX

```

00100      1*      CHG
00101      2*      SUBROUTINE H6ISAMP,AV,SD,F1,F2,F3,F4,F5,F6,F7,F8,F9,F10,F11,
00101      3*      1 F12,F13,F14,F15,F16,FA,YIN,FUP,FLO )
00101      4*      C      VERSION 2.      REVISED MARCH 1977
00101      5*      C      PURPOSE - DEVELOP A RUNNING HISTOGRAM OF AN INPUT SEQUENCE
00101      6*      C      CALL SEQUENCE
00101      7*      C      SAMP - OUTPUT NUMBER OF SAMPLES
00101      8*      C      AV - OUTPUT AVERAGE (RUNNING SUM)
00101      9*      C      SD - OUTPUT STANDARD DEVIATION (RUNNING SUM SQUARED)
00101      10*      C      F1-F16 - ARRAY WITH NUMBER OF OCCURENCES IN EACH INTERVAL
00101      11*      C      FA - OUTPUT CONTAINING MEASUREMENT INTERVAL
00101      12*      C      FUP - INPUT SPECIFYING UPPER MEASUREMENT LIMIT
00101      13*      C      FLO - INPUT SPECIFYING LOWER MEASUREMENT LIMIT
00101      14*      C      FYN - INPUT MEASUREMENT

```

000003
600003
600003
C00003
000003
000003
C00003
C00003
C00003
C00003
C00003
C00003
C00003
C00003
C00003

寄

```

00103 15*      DIMENSION F(16),T(16),AX(16)
00104 16*      DIMENSION GRAPH(114,46)
00105 17*      COMMON GRAPH
00106 18*      COMMON/CTIME/TIME/CSIMUL/DUM(7),TMAX
00107 19*      COMMON/COVRLY/OUH(13),CPUSEC /CIMPL/IMPL
00110 20*      DATA PLANK,VERT,HORIZ,POINT/IN ,INI,IN-,IN+/
00115 21*      IF(IMPL.GT.0) GO TO 100
00117 22*      DO 50 I=1,16
00122 23*      50 F(I)=0.
00124 24*      FAC=(FUP-FLO)/14.
00125 25*      SD=0.
00126 26*      AV =0.0
00127 27*      SAMP=0.0
00130 28*      100 CONTINUE
00131 29*      IF(IMPL.LT.2) RETURN
00133 30*      DO 200 I=1,16
00134 31*      L=I
00137 32*      FAX=F(0+(I-1)*FA
00140 33*      IF(FIN.LE.FAX) GO TO 300
00142 34*      200 CONTINUE
00144 35*      300 F(ILI)=F(ILI)+1.
00145 36*      SAMP=SAMP+1.
00146 37*      AV=AV+FIN
00147 38*      SD=SD+FIN*FIN
00150 39*      IF(TIME.LT.TMAX+.99999)RETURN
00152 40*      SAMP=7.
00153 41*      DO 350 I=1,16
00156 42*      350 SAMP=SAMP+F(I)
00160 43*      ISAMP=SAMP
00161 44*      ISAMP=MAX(01,ISAMP)
00162 45*      AV2=AV/ISAMP
00163 46*      SD=SQRT(SD/ISAMP-AV*AV)
00164 47*      XMAX=F(1)
00165 48*      DO 360 I=1,16
00170 49*      360 IF(F(I).GE.XMAX) XMAX=F(I)
00173 50*      IF(XMAX.EQ.0.) XMAX=10.
00175 51*      MX=XMAX/44.
00176 52*      DO 370 I=1,46
00201 53*      GRAPH(I,1)=VERT
00202 54*      370 GRAPH(114,I)=VERT
00204 55*      DO 380 I=2,113
00207 56*      GRAPH(I,1)=HORIZ
00210 57*      380 GRAPH(1,46)=HORIZ
00212 58*      DO 400 I=5,103,14
00215 59*      400 GRAPH(I,46)=VERT
00217 60*      DO 450 I=8,106,7
00222 61*      450 GRAPH(I,1)=VERT
00224 62*      DO 500 J=2,45
00227 63*      DO 500 J=2,113
00232 64*      500 GRAPH(J,1)=PLANK
00234 65*      DO 600 IC=1,16
00240 66*      J=IF IX(45.5-F(1IC)/MX)
00241 67*      DO 600 J=1,7
00244 68*      J2=(1IC-1)*7+J+1
00245 69*      DO 600 J3=J,45
00250 70*      600 GRAPH(J2,J3)=POINT
00250 71*

```

```

006003
006003
006003
006003
006003
006003
006012
006012
006013
006013
006017
006020
006021
006023
006023
006034
006034
006036
006045
006053
006053
006057
006062
006065
006071
006104
006111
006111
006114
006123
006130
006135
006147
006154
006154
006163
006167
006175
006175
006176
006203
006204
006211
006211
006216
006216
006224
006224
006224
006234
006240
006252
006257
006263
006272
006272

```

HC

```

00254 72*      DO 700 I=1,16
00257 73*
00257 74*      700 AX(I)=F1(I)/ISAMP
00261 75*      DO 800 I=1,8
00264 76*
00264 77*      800 Y01(I)=FLO+(I-1)*2.*FA-FA/2.
00266 78*
00266 79*      WRITE(6,900)((GRAPH(I,I),I=1,114)
00271 80*      900 FORMAT(1H1,9X,114A1/)
00272 81*      WRITE(6,1000)(AX(I),I=1,16)
00275 82*      1000 FORMAT(1H+,9X,1H1,16F7.5,1H1//)
00276 83*      WRITE(6,1100)
00300 84*      1100 FORMAT(1H+,9X,3H1,112X,1H1//)
00301 85*      WRITE(6,1200)((GRAPH(I,J),I=1,114),J=2,46)
00312 86*      1200 FORMAT(1H+,9X,114A1/45(13X,114A1//)
00313 87*      WRITE(6,1300)(ID(I),I=1,8)
00316 88*      1300 FORMAT(1H+,9X,8(F13.5,1X1//)
00317 89*      WRITE(6,1400) ISAMP,AV,SD
00324 90*      1400 FORMAT(1H+,10X,14H1HISTOGRAM FOR ,I7,8H SAMPLES,
00324 91*      17H MEAN= ,F13.5,18H STANDARD DEV.= ,F13.5/)
00325 92*      RETURN
00326 93*      END

```

```

000304
000306
000306
000314
000314
000314
000330
000330
000340
000340
000340
000350
000350
000355
000355
000367
000367
000377
000377
000407
000407
000407
000505

```

7.14 HYDRO STORAGE VESSEL

The hydro storage vessel is modeled as an above ground reservoir with a large and constant surface area. The change in reservoir height between maximum and minimum levels is assumed small in comparison to the height of the water above the turbine. Hence, reservoir height is assumed constant. The reservoir has specified evaporation and leakage rates. Average input flow gained by rainfall is also specified. Energy storage is calculated based on the potential energy of the water in the reservoir relative to the turbine inlet.

Basic Equation

$$MA = M1 - M2 - NE*AS - NL + MDR*AS/14052$$

Inputs

<u>Parameter/Port</u>		<u>Description</u>	<u>Units</u>
M	1	Input water mass flow rate	gal/h
NE		Evaporation coefficient (D = 0.03)	gal/ft ² -h
AS		Reservoir surface area	ft ²
NL		Leakage coefficient (D = 8.0)	gal/h
MDR		Rainfall rate	Inches/year
MDM		Maximum allowable mass flow rate (D = 4×10^5)	gal/h
MM		Maximum allowable reservoir capacity (D = 5×10^6)	gal
MO		Minimum allowable reservoir capacity	gal
H	1	Reservoir height above turbine	ft
MDE		Reservoir deadband for priority resequence	gal
RE	1	Power request (discharge)	kw
CR		Reservoir cost coefficient (D = 0.025)	\$/gal
EF	1	Input product efficiency	-
MP	1	Maximum input charging rate	kw
LE		Reservoir life expectancy	years
CM		Maintenance cost/year	\$

Outputs

<u>Variable/Port</u>			
M	2	Outlet water mass flow rate	gal/hr
E		Energy stored	kwh
H	2	Reservoir height above turbine (=H1)	ft
MA		Reservoir mass (state)	gal
CCO		Reservoir cost/year	\$
MP	2	Maximum discharge rate allowable	kw
INT		Priority interrupt flag	-
RE	2	Maximum charging rate request	kw

D - Default values supplied

<u>Statistics</u>	<u>Description</u>	<u>Units</u>
MDU	Maximum mass flow rate	gal/hr
MU	Maximum reservoir mass	gal
ML	Minimum reservoir mass	gal

The calculation sequence and default values assume a pond sized for 120kw storage for 24 hours (5×10^6 gallons of water 200 ft. above turbine inlet). The evaporation coefficient NE assures the pond drops $\frac{1}{2}$ " in height per 10 hours. To obtain a more accurate value for this parameter requires knowledge of local conditions. The leakage coefficient NL is based on the assumption of a loss of 0.1% of the maximum reservoir capacity in the rated storage time of 24 hours. The reservoir cost estimates are based on the compensation reservoir given in Reference 1.

-
1. "Preliminary Feasibility Evaluation of Compressed Air Storage Power Systems," United Technologies AER 74-00242, December 1976.

Calculation Sequence

$$C1 = \text{conversion constant} = 0.377 \times 10^{-6} \frac{\text{kwh}}{\text{ft-lb}}$$

$$C2 = \text{conversion constant} = 8.3398 \text{ lb/gal}$$

$$A = C1 * C2 * H1$$

1) Reservoir cost

$$CC = CR * MM / LE$$

2) Volume of water discharged

$$M2 = RE1 / A$$

3) Reservoir water volume

$$MA = M1 - M2 - NE * AS - NL + (MDR * AS / 14052.)$$

4) Energy stored

$$E = A * MM$$

5) Checks

$$M1 > MDM \text{ or } M2 > MDM \Rightarrow \text{DIAGNOSTIC}$$

$$M > MM, \Rightarrow \text{DIAGNOSTIC}$$

$$M < M0 \Rightarrow \text{DIAGNOSTIC}$$

6) Priority interrupt

$$\text{If } M \leq M0, \text{ INT} = 1$$

$$\text{IF } M > M0 + MDE \text{ and } \text{INT} = 1, \text{ INT} = 0$$

$$\text{If } M \geq MM, \text{ INT} = -1$$

$$\text{If } M < MM - DME \text{ and } \text{INT} = -1, \text{ INT} = 0$$

Calculation Sequence Cont.

7) Maximum charging rate request

$$MD1 = \text{MIN} (MDM, (MM-M)/TINC)$$

$$RE2 = \text{MIN} (MP1, MD1 * A) / EF1$$

Maximum discharge rate

$$MP2 = A * \text{MIN} (MDM, (M-MD)/TINC)$$

where TINC = Integration step size in hrs

8) Compute Statistics and Costs

HS

```

00101 24* C MP2 - MAXIMUM DISCHARGE RATE ALLOWABLE, KW 000000
00101 25* C INT - PRIORITY INTERRUPT FLAG 000000
00101 26* C RE2 - MAXIMUM CHARGING RATE REQUEST, MW 000000
00101 27* C MDU - MAXIMUM MASS FLOW RATE, GAL/HR 000000
00101 28* C MU - MAXIMUM RESERVOIR MASS, GAL 000000
00101 29* C ML - MINIMUM RESERVOIR MASS, GAL 000000
00101 30* C
00101 31* C INPUTS 000000
00101 32* C M1 - INPUT WATER MASS FLOW RATE, GAL/HR 000000
00101 33* C ME - EVAPORATION COEFFICIENT, GAL/FT**2-HR 000000
00101 34* C AS - RESERVOIR SURFACE AREA, FT**2 000000
00101 35* C NL - LEAKAGE COEFFICIENT 000000
00101 36* C MOR - RAINFALL RATE, INCHES/YEAR 000000
00101 37* C MDN - MAXIMUM ALLOWABLE MASS FLOW RATE, GAL/HR 000000
00101 38* C MM - MAXIMUM ALLOWABLE RESERVOIR CAPACITY, GAL 000000
00101 39* C MO - MINIMUM ALLOWABLE RESERVOIR CAPACITY, GAL 000000
00101 40* C M1 - RESERVOIR HEIGHT ABOVE TURBINE, FT 000000
00101 41* C MDE - RESERVOIR DEADEND FOR PRIORITY RESEQUENCE, GAL 000000
00101 42* C RE1 - POWER REQUEST (DISCHARGE), KW 000000
00101 43* C CR - RESERVOIR COST COEFFICIENT 000000
00101 44* C EF1 - INPUT PRODUCT EFFICIENCY 000000
00101 45* C MP1 - MAXIMUM INPUT CHARGING RATE, KW 000000
00101 46* C LE - RESERVOIR LIFE EXPECTANCY, YEARS 000000
00101 47* C CM - MAINTENANCE COST/YEAR, $ 000000
00101 48* C
00103 49* COMMON/CMPL/INPL,ICNT/CTIME/TIME/CSINUL/QUIN(T),TMAX 000000
00104 50* COMMON/COST/CCI,CN1 000000
00105 51* REAL M2,MP2,MDU,MU,ML,M1,ME,NL,MOR,MDN,MM,MO,MDE,MP1,LE,INT,M 000000
00106 52* REAL M01,M0M1 000000
00106 53* C
00107 54* IF(IINPL.GT.0)GO TO 100 000000
00107 55* C
00111 56* M1=0.0 000000
00112 57* M2=M1 000000
00113 58* TMAX=TMAX+0.99999 000000
00114 59* C1 = 1.1441E-4 000010
00114 60* C
00115 61* INT=0.0 000010
00116 62* MDU=0.0 000012
00117 63* MU =0.0 000013
00120 64* ML =1.0E10 000014
00121 65* IF(ME .EQ. .99999)ME =0.03 000015
00123 66* IF(NL .EQ. .99999)NL =8.0 000017
00125 67* IF(MDN .EQ. .99999)MDN=4.0E5 000024
00127 68* IF(MM .EQ. .99999)MM =5.0E6 000031
00131 69* IF(CR .EQ. .99999)CR =0.025 000036
00131 70* C 000043
00131 71* C 000043
00131 72* C RESERVOIR COST 000043
00131 73* C 000043
00133 74* CC =CM*MM/LE 000050
00133 75* C 000050
00133 76* C VOLUME OF WATER DISCHARGED 000050
00133 77* C 000050
00134 78* 100 A=C1*M1 000055
0013 79* M2 =RE1/A 000057
0013 80* C 000057

```

```

00135 81* C RESERVOIR MASS FLOW RATE 000057
00135 82* C 000057
00136 83* IF (IM.NC.OIDM=M1-M2-NE*AS-NL*MDR*AS/14052.0 000062
00136 84* C 000062
00136 85* C ENERGY STORED 000062
00136 86* C 000062
00140 87* E =A*M 000077
00140 88* C 000077
00141 89* MDH1=MDH/.9999 000102
00142 90* IF (M1.LT.MDH1.AND. 000105
00142 91* 1 M2.LT.MDH1) GO TO 200 000105
00142 92* C 000105
00144 93* IF (IMPL.EQ.2) WRITE (6,1010) M1,M2,MDH 000127
00152 94* IF (IMPL.EQ.2) ICNT=ICNT+1 000135
00152 95* C 000135
00154 96* 200 IF (M .LT. MM+MDE160 TO 300 000144
00154 97* C 000144
00156 98* IF (IMPL.EQ.2) WRITE (6,1020) M,MM 000150
00163 99* IF (IMPL.EQ.2) ICNT=ICNT+1 000162
00163 100* C 000162
00165 101* 300 IF (M .GT. MM) GO TO 400 000171
00165 102* C 000171
00167 103* IF (IMPL.EQ.2) WRITE (6,1030) M,MO 000174
00174 104* IF (IMPL.EQ.2) ICNT=ICNT+1 000206
00174 105* C 000206
00174 106* C PRIORITY INTERRUPT 000206
00174 107* C 000256
00176 108* *DO IF (M .LE. MO) INT=1.0 000215
00200 109* IF (M .GT. (MO+MDE1.AND. 000222
00200 110* 1 INT.EQ.1.0) INT=0.0 000222
00202 111* IF (M .GT. MM) INT=-1.0 000241
00204 112* IF (M .LT. (MM-MDE1.AND. 000247
00204 113* 1 INT.EQ.-1.0) INT=0.0 000247
00204 114* C 000247
00204 115* C MAXIMUM CHARGE RATE REQUEST AND DISCHARGE RATE 000247
00204 116* C 000247
00206 117* MD1=AMIN1(MDH,AMAX1(0.,(MM-M1)/DUM(7))) 000266
00207 118* RE2=AMIN1(MP1,MD1+A)/EF 000302
00210 119* MP2=AMIN1(MDH,AMAX1(0.,(M-MO)/DUM(7))) *A 000311
00210 120* C 000311
00211 121* IF (IMPL.LE.1) RETURN 000326
00211 122* C 000326
00211 123* C STATISTICS 000326
00211 124* C 000326
00213 125* MDU=AMAX1(0,MDU) 000335
00214 126* MU =AMAX1(M ,MU ) 000343
00215 127* ML =AMIN1(M ,ML ) 000351
00215 128* C 000351
00216 129* IF (TIME.LT.TMAX) RETURN 000357
00216 130* C 000357
00220 131* CC=CC+CC 000366
00221 132* CM=CM+CM 000371
00221 133* C 000371
00222 134* RETURN 000374
00222 135* C 000374
00223 136* 1010 FORMAT(1H0,23HH5 INLET MASS FLOW RATE,F12.3, 5H OR , 000550
00223 137* 1 21HOUTLET MASS FLOW RATE,F12.3. 000550

```

HS

```

00223 138*      2      26H  IS GREATER THAN MAXIMUM,F12.3)
00224 139*    1020 FORMAT(1H0,19HNS RESERVOIR VOLUME,F12.3,
00225 140*      1      30H  EXCEEDED MAXIMUM ALLOWABLE,F12.3)
00225 141*    1030 FORMAT(1H0,19HNS RESERVOIR VOLUME,F12.3,
00225 142*      1      24H  DROPPED BELOW MINIMUM,F12.3)
00225 143*  C
00226 144*      END

```

```

000550
000550
000550
000550
000550
000550
000550

```

7.15 HYDRAULIC TURBINE

The hydraulic turbine model is based on a constant speed design and is typical of a reaction/Francis type turbine. The turbine is assumed to be designed to a specified operating point and output power.

For off design performance the pump efficiency is assumed to be functionally related to the first power of the mass flow rate. The equations are assumed to be valid over a specified range of values for the turbine parameter.

Basic Equations

$$P = \text{EFF} * M * C1 * C2 * H$$

$$\text{EFF} = 1 - (1 - \text{EFD}) * M^D / M$$

where C1, C2 are conversion constants.

Inputs

<u>Parameter/Port</u>	<u>Description</u>	<u>Units</u>
M	Inlet mass flow rate	gal/h
H	Height of reservoir above turbine inlet	ft
EFD	Design pt. turbine efficiency (D = 0.90)	-
MD	Design pt. mass flow rate (D = 2×10^5)	gal/h
MM	Maximum mass flow rate (D = 3×10^5)	gal/h
EF	1 Input product efficiency	-
MP	1 Input maximum discharge rate	kw
CK	Turbine capacity cost coefficient ¹ (D = 0.011)	-
F0	Turbine exponent for cost calculations (D = 0.5)	-
RS	Angular velocity	rpm
X	Turbine head exponent for cost calculations (D = 0.25)	-

Outputs

<u>Variable/Port</u>		
CC0	Turbine cost/year	\$
EFF	Turbine efficiency	-
P	2 Output power	kw
EF	2 Output product efficiency	-
MP	2 Output maximum discharge rate	kw
CP	Turbine characteristic parameter	-

Statistics

CPU	Maximum CP	-
CPL	Minimum CP	-
PU	Maximum output power	kw

D - Default values

¹CK = Capital cost (known unit)/((MD*481.2)**F0*H**X*life expectancy)

The calculation sequence and default values assume a constant speed reaction type hydraulic turbine nominally rated for 120kw and located 200 ft. below the reservoir. The equations relating the various physical parameters are assumed to be valid for the indicated range of the characteristic turbine parameter, CP. The equations and cost estimates are based on the data given in Reference 1, and the cost estimates on data from Reference 2.

Calculation Sequence

$$C1 = 0.377 \times 10^{-6} \frac{\text{kwh}}{\text{ft-lb}}$$

$$C2 = 8.3398 \text{ lb/gal}$$

$$A = C1 * C2 * H$$

1) Costs

$$CC0 = CK * (ND * 481.2) ** F0 ** H ** X$$

2) Efficiency

If $M \leq 0$ set $EFF = 1$ and go to 3)

$$EFF = 1 - (1 - EFD) * ND / M$$

$$EFF = \text{MAX}(EFF, 0.6)$$

-
1. L. Marks and T. Baumeister, "Mechanical Engineers Handbook", McGraw Hill, N.Y., 1958, Section 9, p. 207.
 2. Carson and Fogleman, "Comparison of Methods for Converting Existing Power Plants to Pumped Storage Facilities", International Engineering Company, Inc., 1974.

Calculation Sequence Cont.

3) Output Power

$$P2 = EFF * A * M$$

4) Product Efficiency

$$EF2 = EF1 * EFF$$

$$EFM = MM - (1 - EFD) * MD$$

5) Maximum Discharge Rate

$$MP2 = \text{Min} \{ MP1 * EFD, EFM * A \}$$

6) Turbine Characteristic Parameter

(If $P2 \leq 0$ go to 7)

$$CP = RS * \text{SQRT} (P2 * 0.746 / H ** 1.25)$$

If $CP > 100$ write DIAGNOSTIC

If $M > MM$ write DIAGNOSTIC

7) Compute Statistics and Costs

SUBROUTINE HT ENTRY POINT 000305

STORAGE USED CODE(1) 00043D; DATA(0) 000075; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 C1MPL 000002
 0004 C1TME 000001
 0005 C5TMM 000010
 0006 C05T 000001

EXTERNAL REFERENCES (BLOCK, NAME)

0007 XPPR
 0010 SGR1
 0011 NWDUS
 0012 N1021
 0013 NC4R31

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001	000073	100L	0000	000003	1010F	0000	000016	1020F	0001	000205	200L	0101	000231	300L				
0001	000113	400L	0006	R	000000	CCL	0000	R	000001	C1	0005	000000	DUM	0000	R	000002	EFM	
0003	I	000001	ICMT	0003	T	000000	IMPL	0000	000057	IN,PS	0004	R	000000	TIME	0005	R	000007	TMAX
0003	R	000000	TMAX1															

00100	1*	CHT				000000
00101	2*		SUBROUTINE HT(CC, EFF, P, EF2, MP2, CP, CPU, CPL, PU, H, H, EFD, MO, MM			000000
00101	3*		1	EF1, MP1, CK, FO, RS, X)		000000
00101	4*	C				000000
00101	5*	C	PURPOSE	PERFORMANCE OF A HYDRAULIC TURBINE		000000
00101	6*	C				000000
00101	7*	C	METHOD	OFF DESIGN PERFORMANCE IS ASSUMED PROPORTIONAL TO		000000
00101	8*	C				000000
00101	9*	C		MASS FLOW RATE		000000
00101	10*	C				000000
00101	11*	C	WRITTEN BY F. D. MAHONY	VERSION 1, MARCH 30 1977		000000
00101	12*	C				000000
00101	13*	C	CALL SEQUENCE			000000
00101	14*	C	OUTPUTS			000000
00101	15*	C	CC	- TURBINE COST/YEAR, \$		000000
00101	16*	C	EFF	- TURBINE EFFICIENCY		000000
00101	17*	C	P	- OUTPUT POWER, KW		000000
00101	18*	C	EF2	- OUTPUT PRODUCT EFFICIENCY		000000
00101	19*	C	MP2	- OUTPUT MAXIMUM DISCHARGE RATE, KW		000000
00101	20*	C	CP	- TURBINE CHARACTERISTIC PARAMETER		000000
00101	21*	C	CPU	- MAXIMUM CP		000000

HT

00101	22*	C	CPL = MINIMUM CP	000000
00101	23*	C	PU = MAXIMUM OUTPUT POWER, KW	000000
00101	24*	C		000000
00101	25*	C	INPUTS	000000
00101	26*	C	M - INLET MASS FLOW RATE, GAL/HR	000000
00101	27*	C	H - HEIGHT OF RESERVOIR ABOVE TURBINE INLET, FT	000000
00101	28*	C	EFD - DESIGN POINT TURBINE EFFICIENCY	000000
00101	29*	C	MD - DESIGN POINT MASS FLOW RATE, GAL/HR	000000
00101	30*	C	MM - MAXIMUM MASS FLOW RATE, GAL/HR	000000
00101	31*	C	EF1 - INPUT PRODUCT EFFICIENCY	000000
00101	32*	C	MP1 - INPUT MAXIMUM DISCHARGE RATE	000000
00101	33*	C	CK - TURBINE CAPACITY COST COEFFICIENT	000000
00101	34*	C	FO - TURBINE EXPONENT FOR COST CALCULATIONS	000000
00101	35*	C	RS - ANGULAR VELOCITY, RPM	000000
00101	36*	C	X - TURBINE HEAD EXPONENT FOR COST CALCULATIONS	000000
00101	37*	C		000000
00103	38*		COMMON/CIMPL/IMPL,ICNT/CTIME/TIME /CSIMUL/DUNIT1,IMAX /COS1/ CCI	000000
00104	39*		REAL MP2,M,MD,MM,MP1	000000
00104	40*	C		000000
00105	41*		IF4TMPL.EQ.0.060 TO 100	000000
00105	42*	C		000000
00107	43*		IMAX1=IMAX+0.00000	000002
00110	44*		RS =3600.0	000005
00110	45*	C		000005
00111	46*		IF1EFD.EQ. .999999)EFD=0.9	000007
00111	47*		IF1MD .EQ. .999999)MD =2.0E5	000014
00115	48*		IF1MM .EQ. .999999)MM =3.0E5	000021
00117	49*		IF1CK .EQ. .999999)CK =0.011	000026
00121	50*		IF1FO .EQ. .999999)FO =0.5	000033
00123	51*		IF1X .EQ. .999999)X =0.25	000040
00123	52*	C		000040
00125	53*		CPL=1.0E10	000045
00126	54*		CPU=0.0	000047
00127	55*		PU =0.0	000050
00127	56*	C		000050
00130	57*		C1 = 3.1441E-6	000051
00131	58*		CC =CK*(MD+401.2)**FO*MM**X	000053
00131	59*	C		000053
00131	60*	C	EFFICIENCY	000053
00132	61*		100 EFF =1.0	000073
00132	62*	C		000073
00133	63*		IF4M.LE.0.0160 TO 400	000074
00133	64*	C		000074
00135	65*		EFF=1.0-(1.0-EFD)*MD/M	000077
00136	66*		IF1EFF.LT.0.61 EFF=0.6	000104
00136	67*	C		000104
00136	68*	C	OUTPUT POWER	000104
00136	69*	C		000104
00140	70*		400 P =EFF*M*H*CC1	000113
00140	71*	C		000113
00140	72*	C	PRODUCT EFFICIENCY	000113
00140	73*	C		000113
00141	74*		EF2=EF1*EFF	000117
00141	75*	C		000117
00141	76*	C	MAXIMUM DISCHARGE RATE	000117
0014	77*	C		000117
0014	78*		EFM =MM*.9999-11.0-EFD)*MD	000122

HT

```

00142 79* C
00143 80*      NP2=AMIN(IMP1*EFD,EFM*N*CI)
00143 81* C
00143 82*      TURNING CHARACTERISTIC PARAMETER
00143 83* C
00144 84*      IF(P.LE.D.D) GO TO 300
00146 85*      CP=RS*SQRT(P*0.7861/M**1.25
00146 86* C
00147 87*      IF(CP.LT.100.0)GO TO 200
00147 88* C
00151 89*      IF(IMP1.EQ.2)WRITE(6,1010)CP
00155 90*      IF(IMP1.EQ.2) ICNT=ICNT+1
00155 91* C
00157 92*      200 IF(M.LT.MM)GO TO 300
00157 93* C
00161 94*      IF(IMP1.EQ.2)WRITE(6,1020)M,MM
00166 95*      IF(IMP1.EQ.2) ICNT=ICNT+1
00166 96* C
00170 97*      300 IF(IMP1.LE.1)RETURN
00170 98* C
00170 99*      STATISTICS
00170 100* C
00172 101*      CPU=AMAX1(CPU,CP)
00173 102*      CPL=AMIN1(CPL,CP)
00174 103*      PU=AMAX1(PU,P)
00174 104* C
00175 105*      IF(TIME.LT.TMAX)RETURN
00175 106* C
00175 107*      COST
00175 108* C
00177 109*      CCI=CC)+CC
00177 110* C
00200 111*      RETURN
00200 112* C
00201 113*      1010 FORMAT(1H0,48HNT TURNING CHARACTERISTIC PARAMETER OUT OF RANGE
00201 114*      1 F12.3)
00202 115*      1020 FORMAT(1H0,23HNT INLET MASS FLOW RATE,F12.3
00202 116*      1 ,37H GREATER THAN MAXIMUM DESIGN VALUE,F12.3)
00202 117* C
00203 118*      END

```

```

000122
000131
000131
000131
000131
000142
000144
000144
000162
000162
000162
000165
000176
000176
000205
000205
000210
000222
000222
000231
000231
000231
000231
000237
000245
000253
000253
000261
000261
000261
000261
000270
000270
000273
000273
000427
000427
000427
000427
000427

```

7.16 ADIABATIC HEAT EXCHANGER

The purpose of the adiabatic heat exchanger is to recover a portion of the heat of compression from the high pressure, high temperature air exiting from the compressor. Figure 7.16-1 shows an adiabatic heat exchanger used in an underground, constant pressure compressed air energy storage system. The adiabatic heat exchanger operates in a manner similar to the high temperature thermal energy storage systems currently conceived for solar thermal power plants¹. In the storage charging mode, high pressure, high temperature air enters the top of the heat exchanger and deposits a portion of its thermal energy in the storage media as either sensible heat or latent heat of fusion. The exiting high pressure air is stored in an appropriate vessel, e.g., underground cavern. In the discharge cycle (HY), high pressure, low temperature air enters the bottom of the heat exchanger, recovers thermal energy from the storage media and exits to the turbine.

The adiabatic heat exchanger model is based on a two cell storage model. Given the stored energy in both cells, a linear temperature profile is computed

¹ BEC/EPR/ RP 788-1, "Advanced Thermal Energy Storage Systems," November 1976.

FIGURE 7.16-1 KOUTZ-GLENDENNING ADIABATIC COMPRESSED AIR STORAGE SCHEME (SINGLE-STAGE HEAT-OF-COMPRESSION STORAGE)

for the media mass. Based on a given inlet mass flow rate, the convective film coefficient, unit thermal conductance, and heat exchanger exit temperature are calculated.

The rate of energy deposited (or withdrawn) is calculated and integrated to yield the stored energy state. For a phase change media, the temperature profile is approximated in the following way: Average cell temperatures TS1 and TS2 are determined from the enthalpy diagram (Figure 7.16-2) using average cell entropy EC1/MA and EC2/MA, respectively. Then a linear temperature profile is constructed as shown in Figure 7.16-3.

Basic Equations

$$\dot{EC}_1 = PX - PY - NU * EC_1 - BE * (EC_1 - EC_2)$$

$$\dot{EC}_2 = (P_2 - PX) - (P_0 - PY) - NU * EC_2 + BE * (EC_1 - EC_2)$$

where

EC1, EC2 = storage power in cells 1 and 2, respectively

PX = charging power in cell 1

PY = discharging power in cell 1

P2 - PX = charging power in cell 2

P0 - PY = discharging power in cell 2

NU = storage media leakage constant

BE = storage media mixing constant

FIGURE 7.16-2: ENTHALPY-TEMPERATURE DIAGRAM FOR HX

FIGURE 7.16-3: STORAGE TEMPERATURE VERSUS TUBE LENGTH

Inputs

<u>Parameter/Port</u>		<u>Description</u>	<u>Units</u>
NU		Storage energy loss coefficient (D = 0.002)	(h) ⁻¹
ST		Rated storage time ¹	h
BE		Storage energy mixing coefficient (D = 0.0)	h ⁻¹
TQ1		Minimum allowable storage temperature (D = 60)	°F
DTD		Media temperature swing ¹ (D = 400)	°F
PD		Rated storage thermal power	kw
TEM		Maximum allowable exit temperature (D = 240)	°F
XD		Design point fraction of molten media mass (D = 0.8)	-
EF	1	Input product efficiency	-
MP	1	Maximum input charging rate	kw
CP1		Storage media heat capacity (D = 2.93X10 ⁻⁴)	kwh/lb °F
H		Storage media heat of fusion ² (D = 0.0219)	kwh/lb
TMT		Storage media melt temperature ² (D = 147)	°F
CPF		Air heat capacity (D = 7.6X10 ⁻⁵)	kwh/lb °F
KF		Air thermal conductivity (D = 1.03X10 ⁻⁴)	kw/ft °F
MU		Air viscosity (D = 0.055)	lb/ft-h
NT		Number of tubes (D = 200)	-
D		Tube diameter (D = 0.03)	ft
L		Tube length (D = 4)	ft
DEL		Tube half spacing (D = 0.085)	ft
K		Storage media thermal conductivity (D = 0.0078)	kw/ft-°F
T	1	Inlet air temperature	°F
M	1	Inlet mass flow rate	lb/h
CM		Storage device yearly maintenance cost (D = 0.6)	\$/kw
CSA		Storage device capacity cost (D = 50)	\$/kw
CSB		Storage device energy cost (D = 15.6)	\$/kwh
LE		Unit life expectancy	years

D - Default values specified

1 - Design point conditions

2 - Used for phase change media, H = 0 for sensible heat

Inputs

<u>Parameter/Port</u>		<u>Description</u>	<u>Units</u>
M	3	Discharge cycle mass flow rate from storage	lb/hr
T	3	Discharge cycle temperature from storage	°F
TA		Ambient temperature	°F
TS0		Storage vessel minimum temperature	°F

Outputs

<u>Variable/Port</u>		<u>Description</u>	<u>Units</u>
EC1		Stored energy (state) for cell 1 (hot side)	kwh
EC2		Stored energy (state) for cell 2 (cold side)	kwh
M	2	Outlet mass flow rate (=M1)	lb/hr
MP	2	Maximum discharge rate	kw
TS1,TS2		Average temperatures for cells 1 and 2	°F
T	2	Air exit temperature	°F
MA		Required storage media mass	lb
CC0		Storage device capital cost/year	\$
HF		Convective heat transfer coefficient	kwh/ft ² -°F
U		Unit thermal conductance	kwh/ft ² -°F
P	2	Charge rate into heat exchanger	kw
E1,E2		Energy stored at start and end of melt	kwh
PM		Maximum allowable charge rate	kw
EF	2	Output product efficiency	-
RT		Thermal resistance	°F/kw
P0		Discharge power taken from heat exchanger	kw
T0		Discharge cycle output temperature	°F
EF0		Discharge cycle efficiency	-

Statistics

TSU	Maximum storage temperature	°F
TSL	Minimum storage temperature	°F
ME	Maximum stored energy	kwh
MT	Maximum exit temperature	°F

The default values assume use of paraffin wax as the phase change storage medium. (In reality, paraffin wax may not be applicable to temperatures as high as 600°F. The selection of a phase change medium involves careful consideration of a number of factors [see Reference 1]). The heat exchanger geometric parameters, i.e., tube number, diameter, etc., and heat exchanger cost estimates are based on the baseline phase change storage device developed in Reference 1, but scaled down to reflect expected mass flow rates and required media mass. Although these data were developed for a different application (50 MWe, 6 hour storage, average temperature = 786°C), they can be considered representative until detail design data is available.

1. "Advanced Thermal Energy Storage," BEC/EPRI RP 788-1, July 1976.

Calculation Sequence

1) Initial Calculations

$$MA = \frac{PD*ST*0.5}{XD*H+CP1*DTD}$$

$$OC0 = (CSA+CSB*ST)*PD/LE$$

$$E1 = MA*CP1*(TMT-T01)$$

$$E2 = MA*[H+CP1*(TMT-T01)]$$

$$T3 = TS0=TA$$

$$A = (D*DEL+DEL**2)/5.$$

$$RB(1) = D/2, RB(I+1) = \sqrt{RB(I)**2+A} \quad I=1,5$$

$$RN(I) = \sqrt{RB(I+1)**2+RB(I)**2}/2$$

$$RT = \frac{D}{2*k} \sum_{i=1}^4 \ln \left(\frac{RN(I+1)}{RN(I)} \right)$$

2) Storage Temperature (see Figure 7.16-2)

$$TS = \begin{cases} T01 + \frac{E}{MA*CP1} & \text{if } E < E_1 \\ TMT & \text{if } E_1 \leq E \leq E_2 \\ T01 + \frac{\left(\frac{E}{MA} - H\right)}{CP1} & \text{if } E > E_2 \end{cases}$$

where TS = TS1 and E = EC1 for storage cell 1 and similarly for cell 2.

3) HX Exit Temperature Calculations

$$M2 = M1$$

$$P2 = 0$$

$$PX = 0$$

3) Cont.

$$T2 = TS2 - (TS1 - TS2)/2$$

$$\Delta T = TS1 - TS2$$

If M1 = 0, GO TO 7)

4) Convective Heat Transfer Coefficient¹

$$HF = \frac{KE}{D} \left[0.0215 * \left(\frac{M1}{NT} * \frac{4}{MU*PI*D} \right)^{0.8} * \left(\frac{CPF*MU}{KF} \right)^{0.6} \right]$$

5) Thermal Conductance

$$U = \left[\frac{1}{HF} + RT \right]^{-1}$$

$$UA = U*PI*D*L*NT / (CPF*M1*2)$$

6) Exit Temperature and Charge Rate (See Equation A2. in Appendix)

$$TX = T1 - \Delta T - (1. - EXP(-UA)) * (T1 - TS1 - \Delta T/2 - \Delta T/UA)$$

$$T2 = TX - \Delta T - (1. - EXP(-UA)) * (TX - (TS1 + TS2)/2 - \Delta T/UA)$$

$$P2 = M1 * CPF * (T1 - T2)$$

$$PX = M1 * CPF * (T1 - TX)$$

7) HY Exit Temperature Calculations

$$T0 = TS1 + \Delta T/2$$

$$P0 = 0.$$

$$PY = 0.$$

If M3 = 0 GO TO 11)

¹ Kays, W. M., Convective Heat and Mass Transfer, McGraw Hill, N.Y., 1966, p. 173.

8) Convective Heat Transfer Coefficient

$$HF0 = \frac{KE}{D} \left(.0215 * \left(\frac{M3}{NT} * \frac{4}{MU*PI*D} \right)^{0.8} * \left(\frac{CPF*MU}{KF} \right)^{0.6} \right)$$

9) Thermal Conductance

$$U0 = \left(\frac{1}{HF0} + RT \right)^{-1}$$

$$UA = U0*PI*D*L*NT/(CPF*M3*2)$$

10) Exit Temperature and Discharge Rate (See Equation A3. In Appendix)

$$TY = T3 + \Delta T - (1.-EXP(-UA)) * (T3 - TS2 + \Delta T/2 + \Delta T/UA)$$

$$T0 = TY + \Delta T - (1.-EXP(-UA)) * (TY - (TS1 + TS2)/2 + \Delta T/UA)$$

$$P0 = M3*CPF*(T0-T3)$$

$$PY = M3*CPF*(T0-TY)$$

11) Energy Deposited

$$\dot{EC1} = PX - NU * EC1 - PY - BE * (EC1 - EC2)$$

$$\dot{EC2} = (P2-PX) - NU * EC2 - (P0 - PY) + BE * (EC1 - EC2)$$

If T2 ≥ TEM, WRITE DIAGNOSTIC

12) Maximum Allowable Mass Flow Rate

$$MDM = PD/(CPF*DTD)$$

13) Maximum Allowable Charge Rate

$$PM = MDM * CPF * (T1-TA)$$

14) Charging and Discharging Efficiency

$$EFF = 1$$

$$\text{If } T2 > TS0 \quad EFF = \frac{T2 - TS0}{T1 - TA}$$

$$EF0 = 1$$

$$\text{If } T3 > TS0 \quad EF0 = \frac{T0 - TA}{T3 - TS0}$$

$$WP2 = \text{MIN} (WP1, PM) * EFF$$

$$EF2 = EF1 * EFF$$

15) Compute Statistics and Cost Summation

Appendix: Temperature Equations for a Media with Constant Gradient

Assumptions

- 1) Constant Gradient Media Temperature:

- 2) Working Fluid Differential Equation:

$$A1. \quad \frac{\partial T_f}{\partial X} = \frac{UA}{L} (TS - T_f) \quad 0 < x < L$$

Main Results: Exit temperature in the charging and discharging cycles are given by

$$A2. \quad T_f(L) = T_f(0) + \Delta TS - (1 - \exp(-UA)) * \left(T_f(0) - TS0 + \frac{\Delta TS}{UA} \right)$$

$$A3. \quad T_f(0) = T_f(L) - \Delta TS - (1 - \exp(-UA)) * \left(T_f(L) - TSL - \frac{\Delta TS}{UA} \right)$$

where $\Delta TS = TSL - TS0$.

Proof: Multiplying A1. by $\exp(UA \cdot X/L)$ and recombining terms yields:

$$A4. \quad \frac{\partial}{\partial X} \left(\exp(UA \cdot X/L) * T_f \right) = \frac{UA}{L} * \exp(UA \cdot X/L) * TS(X).$$

Integrating A4. and substituting $TS(X) = TS0 + \frac{\Delta TS}{L} * X$ yields

$$\begin{aligned}
 \text{A5.} \quad T_f(X) &= \exp(-UA \cdot X/L) \cdot T_f(0) + \frac{UA}{L} \int_0^X \exp(-UA(x-y)/L) \cdot TS(y) dy \\
 &= \exp(-UA \cdot X/L) T_f(0) + (1 - \exp(-UA \cdot X/L)) \cdot (TS_0 - \Delta TS/UA) \\
 &\quad + \frac{\Delta TS}{L} \cdot X
 \end{aligned}$$

Recombining terms in A5. and letting $X=L$ yields A2. Equation A3. follows from A2. by symmetry, i.e., substitute in A2:

$T_f(0)$ for $T_f(L)$

$T_f(L)$ for $T_f(0)$

TSL for TS_0

TS_0 for TSL .

SUBROUTINE HX ENTRY POINT 001065

STORAGE USED CODE(1) 001517; DATA(1) 000143; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 CIMPL 000002
 0004 CTIME 000001
 0005 CSHML 000010
 0006 COST 000002

EXTERNAL REFERENCES (BLOCK, NAME)

0007 SORT
 0010 MLOG
 0011 XPRR
 0012 EXP
 0013 MWDUS
 0014 MLOG2
 0015 MLOG3

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000301 300L	0000 000034 1010F	0001 000514 200L	0001 000234 205G	0001 000264 214G
0001 000617 300L	0001 000714 500L	0000 R 000016 A	0006 R 000000 CCI	0006 R 000001 CNI
0000 R 000020 DELT	0005 000030 DUM	0000 R 000033 EFF	0000 R 000030 HFO	0000 I 000017 I
0003 I 000001 ICNT	0003 I 000000 IMPL	0000 000107 IMJP4	0000 R 000000 MOM	0000 R 000014 PI
0000 R 000023 PX	0000 R 000027 PY	0000 R 000001 NB	0000 R 000007 RN	0000 R 000025 TCMF
0004 R 000000 TIME	0005 R 000037 TMAX	0003 R 000015 TMAX1	0000 R 000022 TSC	0000 R 000021 TSH
0000 R 000026 TX	0000 R 000032 TY	0000 R 000024 UA	0000 R 000031 UB	

00100 1* CHX
 00101 2*
 00101 3*
 00101 4*
 00101 5*
 00101 6*
 00101 7*
 00101 8*
 00101 9*
 00101 10*
 00101 11*
 00101 12*
 00101 13*
 00101 14*
 00101 15*
 00101 16*

SUBROUTINE HX(EC1,DE1,IE1,EC2,DE2,IE2,M2,MP2,TS1,TS2,T2,NA,
 ICC,HF,U,P,E1,E2,PM,EP2,P0,T0,CFO,N,TSU,TSL,NE,MT,MU,ST,BE,TOL,DT0
 2 ,PO,TEM,XD,CF1,MP1,CP1,M,TMT,CPF,MF,MU,MT,O,L,DEL,K,T1
 3 ,M1,CM,CSA,CSB,LE,M3,T1,TA,TSO1

PURPOSE PERFORMANCE OF ADIABATIC HEAT EXCHANGER DURING CHARGE
 CYCLE
 METHOD HEAT STORAGE MEDIA ASSUMED TO CONTAIN NO TEMPERATURE
 GRADIENTS. ENERGY DEPOSITED IS A FUNCTION OF TEMPERATURE
 AND THERMAL CONDUCTANCE

000000
 000000
 000000
 000000
 000000
 000000
 000000
 000000
 000000
 000000
 000000
 000000
 000000
 000000
 000000
 000000
 000000

HX

00101	17*	C	WRITTEN BY F. O. MAHONY	VERSION 2, JUNE 1977	000000
00101	18*	C			000000
00101	19*	C	CALL SEQUENCE		000000
00101	20*	C	OUTPUTS		000000
00101	21*	C	EC1 - STORED ENERGY (STATE) FOR STORAGE CELL 1, KWH		000000
00101	22*	C	DE1 - ENERGY RATE FOR EC1, KW		000000
00101	23*	C	IE1 - STATUS INDICATOR FOR EC1		000000
00101	24*	C	EC2 - STORED ENERGY STATE FOR STORAGE CELL 2, KWH		000000
00101	25*	C	DE2 - ENERGY RATE FOR EC2, KW		000000
00101	26*	C	IE2 - STATUS INDICATOR FOR EC2		000000
00101	27*	C	M2 - OUTLET MASS FLOW RATE, LB/HR		000000
00101	28*	C	MP2 - MAXIMUM DISCHARGE RATE ALLOWABLE, KW		000000
00101	29*	C	TS1 - STORAGE TEMPERATURE IN CELL 1, DEG F		000000
00101	30*	C	TS2 - STORAGE TEMPERATURE IN CELL 2, DEG F		000000
00101	31*	C	T2 - AIR EXIT TEMPERATURE, DEG F		000000
00101	32*	C	MA - REQUIRED STORAGE MEDIA MASS, LB		000000
00101	33*	C	CC - STORAGE DEVICE CAPITAL COST/YEAR, \$		000000
00101	34*	C	HF - CONVECTIVE HEAT TRANSFER COEFFICIENT, KWH/FT2-F		000000
00101	35*	C	U - UNIT THERMAL CONDUCTANCE, KWH/FT2-F		000000
00101	36*	C	P - CHARGE RATE OF HEAT EXCHANGER		000000
00101	37*	C	E1 - ENERGY STORED AT START OF MELT PHASE, KWH		000000
00101	38*	C	E2 - ENERGY STORED AT END OF MELT PHASE, KWH		000000
00101	39*	C	PM - MAXIMUM ALLOWABLE CHARGE RATE, KW		000000
00101	40*	C	CF2 - OUTPUT PRODUCT EFFICIENCY		000000
00101	41*	C	PD - DISCHARGE POWER TAKEN FROM HEAT EXCHANGER, KW		000000
00101	42*	C	TD - DISCHARGE CYCLE OUTPUT TEMPERATURE, DEG F		000000
00101	43*	C	EFO - DISCHARGE CYCLE EFFICIENCY		000000
00101	44*	C	R - THERMAL RESISTANCE, DEG F/KW		000000
00101	45*	C			000000
00101	46*	C	STATISTICS		000000
00101	47*	C	TSU - MAXIMUM STORAGE TEMPERATURE, DEG F		000000
00101	48*	C	TSL - MINIMUM STORAGE TEMPERATURE, DEG F		000000
00101	49*	C	ME - MAXIMUM STORED ENERGY, KWH		000000
00101	50*	C	MT - MAXIMUM EXIT TEMPERATURE, DEG F		000000
00101	51*	C			000000
00101	52*	C	INPUTS		000000
00101	53*	C	NU - STORAGE ENERGY LOSS COEFFICIENT		000000
00101	54*	C	ST - RATED STORAGE TIME, HR		000000
00101	55*	C	BE - STORAGE ENERGY MIXING COEFFICIENT, 1/HR		000000
00101	56*	C	TQ1 - MINIMUM ALLOWABLE STORAGE TEMPERATURE, DEG F		000000
00101	57*	C	DTD - MEDIA TEMPERATURE SWING, DEG F		000000
00101	58*	C	PD - RATED THERMAL STORAGE POWER, KW		000000
00101	59*	C	TEM - MAXIMUM ALLOWABLE EXIT TEMPERATURE, DEG F		000000
00101	60*	C	XD - DESIGN POINT FRACTION OF MOLTEN MEDIA MASS		000000
00101	61*	C	EFI - INPUT PRODUCT EFFICIENCY		000000
00101	62*	C	MP1 - MAXIMUM INPUT CHARGING RATE		000000
00101	63*	C	CPI - STORAGE MEDIA HEAT CAPACITY, KWH/LB-F		000000
00101	64*	C	H - STORAGE MEDIA HEAT OF FUSION, KWH/LB		000000
00101	65*	C	TMF - STORAGE MEDIA MELT TEMPERATURE, DEG F		000000
00101	66*	C	CPF - AIR HEAT CAPACITY, KWH/LB-F		000000
00101	67*	C	KA - AIR THERMAL CONDUCTIVITY, KWH/FT-F		000000
00101	68*	C	NU - AIR VISCOSITY, LB/FT-HR		000000
00101	69*	C	NT - NUMBER OF M/X TUBES		000000
00101	70*	C	D - TUBE DIAMETER, FT		000000
00101	71*	C	L - TUBE LENGTH, FT		000000
00101	72*	C	DEL - TUBE HALF SPACING, FT		000000
00101	73*	C	K - STORAGE MEDIA THERMAL CONDUCTIVITY		000000

HX

```
COMMON /CINPL/INPL,ICNT/CTIME/TIME /CSINUL/DUM(7),IMAX
COMMON /COSI/ CCI,CMJ
REAL M3,NDM,NU,M2,MF2,MA,ME,MT,MPI,MU,NT,M1,LE,MF,M,L
DIMENSION RR(6),RR(5)
DATA PI/3.14159/
```

```
IF(IIMPL.GT.0)GO TO 100
```

```

IFIMU .EQ. .99999IMU =0.002
IFIBC .EQ. .99999IBC = 0.0
IFTOT1.EQ. .99999TOT1=60.0
IFTOT0.EQ. .99999TOT0=400.0
IFTEN.EQ. .99999TEN=240.0
IFICP1.EQ. .99999ICP1=2.93E-4
IFIM .EQ. .99999IM =2.188E-2
IFIXD .EQ. .99999XD =0.8
IFTMT.EQ. .99999TMT=147.0
IFICPF.EQ. .99999ICPF=7.6E-5
IFINF .EQ. .99999INF =1.03E-4
IFIMU .EQ. .99999IMU =0.055
IFINT .EQ. .99999INT =207.0
IFID .EQ. .99999ID =3.0E-2
IFIL .EQ. .99999IL =4.0
IFIOEL.EQ. .99999IOEL =8.5E-2
IFIK .EQ. .99999IK =7.0E-3
IFICH .EQ. .99999ICH =5.6
IFICSA.EQ. .99999ICSA=30.0
IFICSB.EQ. .99999ICSB=15.4

```

```

TSL=1.DEB
PO=0.0
PH= 0.0
TSU=0.0
MC=0.0
MT=0.0
MJ=0.0
TJ=TA
TSO=TA
MA =PO+0.5*ST/(XD+H+CP1+0T0)
CC =ICSA+CSR*ST)*PD/LE
CH= CH+PD
E1 =HA+CP1*(TM1-T01)
E2 =HA+(H+CP1*(TM1-T01))
TMAX1=TMAX+0.99999
A = (D+DEL+DEL**2)/5.0

```

[illegible]

五

```

00202 131* C COMPUTE THERMAL RESISTANCE OF MEDIA
00202 132* C
00203 133* R0(1)=D/2.0
00203 134* C
00204 135* DO 20 I=1,5
00207 136* R0(I+1)=SQRT(R0(I)**2+A)
00210 137* 20 R0(I)=SQRT(R0(I+1)**2+R0(I)**2)/2.0)
00210 138* C
00212 139* R=0.0
00212 140* C
00213 141* DO 30 I=1,4
00216 142* 30 R=R+ALOG(R0(I+1)/R0(I))
00216 143* C
00220 144* R=R*D/2.0/K
00220 145* C
00222 146* C STORAGE TEMPERATURES
00222 147* C
00222 148* 100 TSI=TMP
00222 149* C
00222 150* IF(FC1.LT.E1) TSI= T01+ EC1/(HA*CP1)
00224 151* IF(FC1.GT.E2) TSI= T01+ (EC1/HA - M1/CP1)
00224 152* TS2=TMP
00227 153* IF(FC2.LT.E1) TS2= T01+ EC2/(HA*CP1)
00231 154* IF(FC2.GT.E2) TS2= T01+ (EC2/HA - M1/CP1)
00231 155* C
00233 156* DELT= TS1 - TS2
00234 157* TSH= TS1+ .5*DELT
00235 158* TSC= TS2 - .5*DELT
00235 159* C
00236 160* T2= TSC
00237 161* M2=M1
00240 162* P =0.0
00241 163* PX=0.0
00242 164* U=1.0/R
00242 165* C
00243 166* IF(M1.LT.0.001)GO TO 200
00243 167* C
00243 168* C CONVECTIVE HEAT TRANSFER COEFFICIENT
00243 169* C
00245 170* HF =KF/D*(0.0215*(M1/NT**4.0/MU/PI/D)**0.8*(CPF*NU/KF)**0.6)
00245 171* C
00245 172* C UNIT THERMAL CONDUCTANCE
00245 173* C
00246 174* U = 1.0/(1.0/HF+R)
00247 175* UA= U*PI*D*L*NT/(M1+CPF*2.)
00250 176* TEMP= DELT/UA
00251 177* UA= 1. - EXP(-UA)
00251 178* C
00251 179* C EXIT TEMPERATURE
00251 180* C
00252 181* TX= T1 - DELT - UA*(T1-TSH-TEMP)
00253 182* T2= TX- DELT - UA*(TX-(TS1+TS2)*.5-TEMP)
00253 183* C
00253 184* C CHARGE RATE
00253 185* C
00255 186* P =M1*CPF*(T1-T2)
00255 187* PX = M1*CPF*(T1-TX)

```

```

000221
000221
000230
000230
000236
000236
000245
000245
000260
000260
000264
000264
000264
000274
000274
000274
000274
000301
000301
000302
000315
000327
000331
000341
000341
000353
000353
000356
000362
000362
000365
000366
000370
000371
000372
000372
000375
000375
000375
000375
000400
000400
000430
000430
000432
000440
000452
000455
000455
000455
000463
000473
000473
000473
000473
000504
000507

```

HY

```

00255 188* C
00255 189* C
00255 190* C
00256 191* 200 TO = TSH
00257 192* PO = 0.0
00260 193* PY = 0.0
00261 194* IF (M3.LT. .001) GO TO 300
00261 195* C
00261 196* C
00261 197* C
00263 198* CONVECTIVE HEAT TRANSFER COEFFICIENT
00263 199* C
00263 200*  $HFO = KF/D = 10.0215 * (M3/NT) * (0.7/MU/PI/D) * (0.3 * (CPF * MU/KF) + 0.6)$ 
00263 201* C
00263 202* UNIT THERMAL CONDUCTANCE
00264 203*  $UO = 1.0 / (1.0 / HFO + R)$ 
00265 204*  $UA = UO * PI * D * (L * NT / (M3 * CPF + 2.))$ 
00266 205* TEMP = DELT / UA
00267 206*  $UA = (EXP1 - UA) - 1.$ 
00267 207* C
00267 208* C
00267 209* C
00270 210* EXIT TEMPERATURE AND DISCHARGE RATE
00271 211*  $TY = T3 + DELT * UA * (T3 - TSC + TEMP)$ 
00272 212*  $TSC = TY + DELT * UA * (TY - (T51 + T52) * 0.5 + TEMP)$ 
00273 213*  $PO = M3 * CPF * (T0 - T3)$ 
00273 214*  $PY = M3 * CPF * (T0 - TY)$ 
00273 215* C
00273 216* C
00273 217* ENERGY DEPOSITED
00276 218* 300 IF (TE1.NE.0) DE1 = PX - PY - MU*EC1 - BE*(EC1-EC2)
00276 219* IF (TE2.NE.0) DE2 = P-PX - (PO-PY) - MU*EC2 - BE*(EC1-EC2)
00276 220* C
00276 221* IF (T2.LT.TEM) GO TO 500
00300 222* C
00302 223* IF (IMPL.EQ.2) WRITE(6,1010) T2,TEM
00307 224* IF (IMPL.EQ.2) ICNT = ICNT + 1
00307 225* C
00307 226* C
00307 227* MAXIMUM ALLOWABLE CHARGE AND FLOW RATES
00311 228* 500 NDM = PO / (CPF * DT0)
00311 229* C
00312 230*  $PH = NDM * (CPF * (T) - TA)$ 
00312 231* C
00312 232* CHARGING AND DISCHARGING EFFICIENCY
00313 233* C
00313 234* EFF = 1.0
00314 235* IF (T2.GE.T50) EFF = (T2 - T50) / (T1 - TA)
00314 236* C
00316 237* EFO = 1.0
00316 238* C
00317 239* IF (T3.GT.T50) EFO = (T0 - TA) / (T3 - T50)
00317 240* C
00321 241*  $MP2 = AMIN1(MP1, PH) * EFF$ 
00321 242* C
00322 243* EF2 = EF1 * EFF
00322 244* C

```

```

000507
000507
000507
000514
000515
000516
000517
000517
000517
000517
000523
000523
000523
000523
000555
000563
000575
000600
000600
000600
000600
000606
000616
000627
000632
000632
000632
000632
000632
000637
000653
000653
000667
000667
000673
000705
000705
000705
000714
000714
000720
000720
000720
000720
000726
000726
000730
000730
000740
000740
000740
000742
000742
000754
000754
000763
000763

```

HX

00322	245*	C		000763
00322	246*	C	STATISTICS	000763
00322	247*	C		000763
00323	248*		IF(IMPL.LE.1)RETURN	000764
00323	249*	C		000766
00325	250*		TSU =AMAX1(TSU,TS1)	000775
00326	251*		TSL =AMIN1(TSL,TS2)	001003
00327	252*		ME = AMAX1(ME, (CCI*CC2)	001011
00330	253*		MT =AMAX1(MT, T2)	001020
00330	254*	C		001020
00331	255*		IF(TIME.LT.TMAX)RETURN	001026
00331	256*	C		001026
00333	257*		CCI =CCI+CC	001035
00334	258*		CPI=CMI+CM	001040
00335	259*		CM= CM/PO	001043
00335	260*	C		001043
00336	261*		RETURN	001046
00336	262*	C		001046
00337	263*		1010 FORMAT(1ND,22HXX EXIT TEMPERATURE ,F12.3	001514
00337	264*		1 ,35M GREATER THAN MAXIMUM ALLOWABLE ,F12.3)	001516
00337	265*	C		001516
00340	266*		END	001516

7.17 ADIABATIC HEAT EXCHANGER - DISCHARGING CYCLE

HY is the discharge cycle complement to HX. All the calculations to obtain the exit temperature and heat exchange power deposited or withdrawn are done in HX. The results are then passed to HY for summary.

Inputs

<u>Parameter/Port</u>		<u>Description</u>	<u>Units</u>
M	1	Air mass flow rate from storage	lb/hr
T0		Exit temperature from HX	°F
P	1	Discharge power from storage	kw
EF0		Discharge cycle efficiency from HX	-
MP	1	Maximum power from storage	kw

Outputs

<u>Variable/Port</u>			
M	2	Exit air mass flow rate (=M1)	lb/hr
T		Exit temperature (=T0)	°F
P	2	Discharge power	kw
MP	2	Maximum discharge power	kw
EF	2	Output product efficiency	-

Statistics

TL	Minimum exit temperature	°F
TU	Maximum exit temperature	°F
SP	Total energy discharged	kwh

Calculation Sequence

$$1) \quad M2 = M1$$

$$T = T0$$

$$MP2 = MP1 * EF0$$

$$EF2 = EF0$$

$$P2 = P1 * EF0$$

$$2) \quad \text{Compute Statistics}$$

SUBROUTINE HY

ENTRY POINT 000064

STORAGE USED CODE(1) 000135; DATA(9) 000017; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 CIMPL 000001

0004 CSMUL 000007

EXTERNAL REFERENCES (BLOCK, NAME)

0005 NERR34

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000007 100L 0004 000000 DUM 0003 1.000000 IMPL 0000 000002 INJPS 0004 R 000006 TINC

00100	1*	CHY		000000
00101	2*		SUBROUTINE HYIN2,T,P2,MP2,EF2,TL,TU,SP,M1,TO,P1,EFO,MP1	000000
00101	3*	C		000000
00101	4*	C	PURPOSE PERFORMANCE OF ADIABATIC HEAT EXCHANGER DURING DISCHARGE	000000
00101	5*	C		000000
00101	6*	C	CYCLE	000000
00101	7*	C		000000
00101	8*	C	METHOD COMPUTE EXIT CONDITIONS USING HEAT EXCHANGER STATE	000000
00101	9*	C		000000
00101	10*	C	DETERMINED IN HX	000000
00101	11*	C		000000
00101	12*	C	WRITTEN BY F. O. MAHONY VERSION 1, MARCH 27 1977	000000
00101	13*	C		000000
00101	14*	C	CALL SEQUENCE	000000
00101	15*	C	OUTPUTS	000000
00101	16*	C	M2 - EXIT AIR MASS FLOW RATE (M1), LB/HR	000000
00101	17*	C	T - EXIT TEMPERATURE (TO), DEG F	000000
00101	18*	C	P2 - TOTAL DISCHARGE POWER, KW	000000
00101	19*	C	MP2 - MAXIMUM DISCHARGE POWER, KW	000000
00101	20*	C	EF2 - OUTPUT PRODUCT EFFICIENCY	000000
00101	21*	C		000000
00101	22*	C	STATISTICS	000000
00101	23*	C	TL - MINIMUM EXIT TEMPERATURE, DEG F	000000
00101	24*	C	TU - MAXIMUM EXIT TEMPERATURE, DEG F	000000
00101	25*	C	SP - TOTAL ENERGY DISCHARGED, KWH	000000
00101	26*	C		000000
00101	27*	C	INPUTS	000000
00101	28*	C	M1 - AIR MASS FLOW RATE FROM STORAGE, LB/HR	000000
00101	29*	C	TO - EXIT TEMPERATURE FROM HX, DEG F	000000
00101	30*	C	P1 - DISCHARGE POWER FROM STORAGE, KW	000000

HY

CO101	3*	C	EFO - DISCHARGE CYCLE EFFICIENCY	000000
CO101	32*	C	MP1 - MAXIMUM POWER FROM STORAGE, KW	000000
CO101	33*	C		000000
CO103	34*		COMMON /CINPL/INPL /CSINUL/DUM16/,TINC	000000
CO103	35*	C		000000
CO104	36*		REAL M2,MP2,M1,MP1	000000
CO104	37*	C		000000
CO105	38*		IF1INPL.GT.0.060 TO 100	000000
CO105	39*	C		000000
CO107	40*		TU =0.0	000002
CO110	41*		SP =0.0	000003
CO110	42*	C		000003
CO111	43*		TL =1.0E10	000004
CO112	44*	100	M2 =M1	000007
CO113	45*		T =T0	000010
CO114	46*		P2 =P1+EFO	000012
CO115	47*		MP2=MP1+EFO	000015
CO116	48*		EF2=EFO	000020
CO116	49*	C		000020
CO117	50*		IF1INPL.LE.1.0RETURN	000022
CO117	51*	C		000022
CO121	52*		TL =AMIN1(TL ,T)	000031
CO122	53*		TU =AMAX1(TU ,T)	000033
CO122	54*	C		000037
CO123	55*		SP =SP +P2*TINC/2.0	000045
CO123	56*	C		000045
CO124	57*		RETURN	000052
CO125	58*		END	000134

7.18 INTEGRATOR WITH SATURATION

InputsParameter/PortDescription

FIN	Input
GKI	Integration gain
GKL	Saturation limiter gain
AMA	Upper limit of output (Default = 10^{36})
AMI	Lower limit of output (Default = -10^{36})

OutputsVariable/Port

FO	Output (state)
----	----------------

Calculation Sequence

$\dot{FO} = GKI * [FIN - GKL * (FO - AMA)]$	if $FO > AMA$
$\dot{FO} = GKI * FIN$	if $AMI \leq FO \leq AMA$
$\dot{FO} = GKI * [FIN - GKL * (FO - AMI)]$	if $FO < AMI$

SUBROUTINE IT ENTRY POINT 000051

STORAGE USED CODE(1) 000067; DATA(0) 000007; BLANK COMMON(2) 000000

EXTERNAL REFERENCES (BLOCK, NAME)

0003 NEPR3

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0000 R 000000 EPS 0000 000003 INJPS

00100	1*	CIT				000000
00101	2*		SUBROUTINE IT(F0,F0DOT,JF0,FIN,GKI,GKL,AMA,AMI)			000000
00101	3*	C	VERSION 2.	REVISED OCT 8 1976		000000
00101	4*	C				000000
00101	5*	C	PURPOSE - SIMULATION OF AN INTEGRATOR WITH SATURATION			000000
00101	6*	C				000000
00101	7*	C				000000
00101	8*	C	METHOD - SEE CODING			000000
00101	9*	C				000000
00101	10*	C				000000
00101	11*	C	LIMITATIONS - EXCESSIVELY HIGH VALUES OF GKL MAY RESULT IN POOR			000000
00101	12*	C	STEADY STATE CONVERGENCE			000000
00101	13*	C				000000
00101	14*	C				000000
00101	15*	C	WRITTEN BY - ADAM LLOYD	LATEST REVISION - NOV 75		000000
00101	16*	C				000000
00101	17*	C				000000
00101	18*	C	INPUT/OUTPUT LIST			000000
00101	19*	C				000000
00101	20*	C	F0 INTEGRATOR OUTPUT	ANY	OUTPUT STATE	000000
00101	21*	C	F0DOT OUTPUT DERIVATIVE	ANY	OUTPUT DERIV	000000
00101	22*	C	JF0 INTEGRATOR CONTROL	---	PROGRAM VAR	000000
00101	23*	C	FIN FUNCTION INPUT	ANY	INPUT VAR	000000
00101	24*	C	GKI INTEGRATOR GAIN	ANY	INPUT PARAM	000000
00101	25*	C	GKL DERIVATIVE LIMITER GAIN	ANY	INPUT PARAM	000000
00101	26*	C	AMA UPPER LIMIT OF OUTPUT	ANY	INPUT PARAM	000000
00101	27*	C	WHERE DERIV. LIMITER STARTS			000000
00101	28*	C	AMI LOWER LIMIT OF OUTPUT	ANY	INPUT PARAM	000000
00101	29*	C	WHERE DERIV. LIMITER STARTS			000000
00103	30*		EPS=FIN			000000
00103	31*	C	----- PROVIDE DEFAULTS THAT ELIMINATE SATURATION			000000
00104	32*		IF(AMA.E0..99999)AMA=1.E36			000000
00104	33*		IF(AMI.E0..99999)AMI=-1.E36			000000
00104	34*		IF(F0.GT.AMA)EPS = FIN - GKL*(F0-AMA)			000000
00104	35*		IF(F0.LT.AMI)EPS = FIN - GKL*(F0-AMI)			000000

00114 36*
00116 37*
00117 38*

IF(IFO.WC.DIF000T=GK)*EPS
RETURN
END

000035
000042
000066

7.19 DC-AC INVERTER

This component models a solid state inverter/transformer. Power losses due to resistive heating and contact potential loss are modeled. A step-up transformer may also be needed to boost output voltage up to that of the bus. Default parameter values are based on rated power = 200 kw.

FIGURE 7.19 INVERTER FUNCTIONAL DIAGRAM

Inputs

<u>Parameter/Port</u>		<u>Description</u>	<u>Units</u>
P	1	DC Input power	kw
RT		Transformer resistance (D = 0)	ohms
VDC		Rated DC voltage (D = 100)	volts
DI		Inverter contact potential (D = 0)	volts
RI		Inverter resistance (D = 0.005)	ohms
RAP		Rated input power	kw
EF	1	Input product efficiency	-
MP	1	Maximum input power	kw
CC		Inverter cost/year	\$

Outputs

<u>Variable/Port</u>			
P	2	AC output power	kw
IDC		DC input current	amps
PL		Power loss	kw
EF	2	Output product efficiency	-
MP	2	Maximum output power	kw

D - Default values supplied.

Calculation Sequence

If $P_1 \leq 0$, $P_2 = I_{DC} = P_L = 0$, $EFF = 1$ and go to 3)

1) Input and output current

$$I_{DC} = P_1 * 1000 / V_{DC}$$

$$I_{AC} = \sqrt{6} * I_{DC} / \pi$$

2) Power loss and output power

$$P_L = (I_{DC} * (D_1 + R_1 * I_{DC}) + \sqrt{3} * R_T * I_{AC}^2) / 1000$$

$$P_2 = P_1 - P_L$$

$$EFF = P_2 / P_1$$

$$P_2 \leq 0 \Rightarrow \text{Diagnostic, } EFF = 1$$

3) Efficiency and maximum power

$$EF_2 = EF_1 * EFF$$

$$MP_2 = \text{MIN}(MP_1, RAP) * EFF$$

4) Compute Costs


```

C0101 24* C RAP -RATED OUTPUT POWER, MW 000000
C0101 25* C EF1 -INPUT PRODUCT EFFICIENCY 000000
C0101 26* C MP1 -MAXIMUM INPUT POWER, MW 000000
C0101 27* C CC -INVERTER COST/YEAR 000000
C0101 28* C 000000
C0103 29* COMMON /CIMPL/IMPL,ICNT/CTIME/TIME/CSIMUL/DUM(7),TMAX/COST/CCI 000000
C0104 30* REAL IDC,MP2,MP1,IAC 000000
C0105 31* DATA P1/3.14159/ 000000
C0105 32* C 000000
C0107 33* IF (IMPL.GT.0) GO TO 100 000000
C0111 34* IF (RT.EQ..99999) RT=0. 000000
C0113 35* IF (VDC.EQ..99999) VDC=100. 000000
C0115 36* IF (IDJ.EQ..99999) IDJ=0. 000000
C0117 37* IF (RI.EQ..99999) RI=0.005 000000
C0121 38* TMAX=TMAX*.99999 000000
C0121 39* C 000000
C0121 40* C COMPUTE INPUT AND OUTPUT CURRENT 000000
C0121 41* C 000000
C0122 42* 100 IF (P1.GT.0.150 TO 200 000000
C0124 43* P2=0. 000000
C0125 44* IDC=0. 000000
C0126 45* PL=0. 000000
C0127 46* EFF2=EF1 000000
C0130 47* MP2=AMIN1(MP1,RAP) 000000
C0131 48* GO TO 400 000000
C0132 49* 200 IDC=P1*1000./VDC 000000
C0133 50* IAC=SQRT(6.)*IDC/P1 000000
C0133 51* C 000000
C0133 52* C POWER LOSS AND OUTPUT POWER 000000
C0133 53* C 000000
C0134 54* PL=(IDC+IDJ+RI*IDC)+SQRT(3.)*RT*IAC*IAC/1000. 000000
C0135 55* P2=P1-PL 000000
C0136 56* EFF=P2/P1 000000
C0136 57* C 000000
C0136 58* C EFFICIENCY AND MAXIMUM POWER 000000
C0136 59* C 000000
C0137 60* EFF2=EF1*EFF 000000
C0140 61* MP2=AMIN1(MP1,RAP) 000000
C0141 62* MP2=MP2*EFF 000000
C0142 63* IF (P2.GT.0.150 TO 400 000000
C0142 64* C 000000
C0144 65* EFF2=EF1 000000
C0145 66* MP2=AMIN1(MP1,RAP) 000000
C0146 67* IF (IMPL.EQ.2) WRITE (6,200) PL,P1 000000
C0151 68* 200 FORMAT(1H0,14MIW POWER LOSS ,F12.3,21H EXCEEDS INPUT POWER ,F12.3, 000000
C0153 69* 12FH CHECK RATED DC VOLTAGE VDC ) 000000
C0154 70* IF (IMPL.EQ.2) ICNT=ICNT+1 000000
C0156 71* P2=0. 000000
C0156 72* C 000000
C0157 73* 400 IF (IMPL.LE.1) RETURN 000000
C0161 74* IF (TIME.LT.TMAX) RETURN 000000
C0163 75* CCI=CCI*CC 000000
C0164 76* RETURN 000000
C0165 77* END 000000

```

7.20 FIRST ORDER LAG

Inputs

<u>Parameter/Port</u>	<u>Description</u>
FIN	Input quantity
GAI	Gain
TC	Time constant ¹ (hours)

F0 Output variable (state)

Calculation Sequence

$$\dot{F0} = (GAI * FIN - F0) / TC$$

NOTE: d.c. gain = GAI; time constant = TC

infinite frequency gain = 0

pole location = $\frac{1}{TC}$ rad/sec.

¹ If TC = 0, then F0 = FIN*GAI

SUBROUTINE LA ENTRY POINT 000026

STORAGE USED CODE(1) 000045; DATA(1) 000010; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 C10 000003

EXTERNAL REFERENCES (BLOCK, NAME)

0004 NERR35

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000010 10L 0003 000002 10IAG 0000 000000 10JPS 0003 000000 10EAD 0003 000001 10WRITE

00100	1*	CLA				000000
00101	2*		SUBROUTINE LA(F0,F0DOT,IFO,FIN,GAJ,TC)			000000
00101	3*	C				000000
00101	4*	C	PURPOSE - TO SIMULATE FIRST ORDER LAG	$F_0 = \frac{GAJ}{1 + TC \cdot S}$		000000
00101	5*	C				000000
00101	6*	C				000000
00101	7*	C				000000
00101	8*	C				000000
00101	9*	C	METHOD - SEE CODING			000000
00101	10*	C				000000
00101	11*	C				000000
00101	12*	C	WRITTEN BY - ADAM LLOYD	LATEST REVISION	NOV 75	000000
00101	13*	C				000000
00101	14*	C				000000
00101	15*	C	INPUT/OUTPUT LIST			000000
00101	16*	C				000000
00101	17*	C	F0	TRANSFER FUNCTION OUTPUT	ANY	OUTPUT STATE
00101	18*	C	F0DOT	TRANSFER FUNCTION OUTPUT DERIV.	ANY	OUTPUT STATE
00101	19*	C	IFO	INTEGRATOR CONTROL	---	PROGRAM VAR
00101	20*	C	FIN	TRANSFER FUNCTION INPUT	ANY	INPUT VAR
00101	21*	C	GAJ	TRANSFER FUNCTION GAIN	---	INPUT PARAM
00101	22*	C	TC	TIME CONSTANT	SECS	INPUT PARAM
00103	23*		COMMON/C10/10EAD,10WRITE,10IAG			000000
00104	24*		IF(10EAD.NE.C.) GO TO 10			000000
00104	25*		F0= GAJ*FIN			000000
00107	26*		RETURN			000000
00110	27*		10 IF(10EAD.NE.O) F0DOT=(GAJ*FIN-F0)/TC			000000
00112	28*		RETURN			000000
00113	29*		END			000000

7.21 LEAD LAG

Inputs

<u>Parameter/Port</u>	<u>Description</u>
FIN	Input quantity
TC1	Numerator time constant (hours)
TC2	Denominator time constant (hours)
GAI	Gain

Outputs

<u>Variable/Port</u>	
X1	Intermediate quantity (state)
F0	Output quantity (variable)

Calculation Sequence

$$F0 = (X1 + FIN*TC1*GAI)/TC2$$

$$\dot{X1} = GAI*FIN - F0$$

NOTE: d.c. gain = GAI

$$\text{infinite gain} = \frac{GAI*TC1}{TC2}$$

$$\text{zero location} = -\frac{1}{TC1}, \text{ rad/sec}$$

$$\text{pole location} = -\frac{1}{TC2}, \text{ rad/sec}$$

SUBROUTINE LL ENTRY POINT 000022

STORAGE USED CODE(1) 000031; DATA(0) 000004; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 CJO 000003

EXTERNAL REFERENCES (BLOCK, NAME)

0004 HERR35

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0003 000002 I0IAG 0000 000000 INJPS 0003 000000 INEAD 0003 000001 IWRITE

00100	1*	CLL				000000
00101	2*		SUBROUTINE LL(X1,X1DOT,IX1,F0,FIN,TC1,TC2,GA1)			000000
00101	3*	C				000000
00101	4*	C	PURPOSE - TO SIMULATE LEAD LAG TRANSFER FUNCTION			000000
00101	5*	C				000000
00101	6*	C	F0 GA1*(1.+TC1*S)			000000
00101	7*	C	----- = -----			000000
00101	8*	C	FIN (1.+TC2*S)			000000
00101	9*	C				000000
00101	10*	C				000000
00101	11*	C	METHOD - SELF EXPLANATORY			000000
00101	12*	C				000000
00101	13*	C				000000
00101	14*	C	LIMITATIONS - NONE			000000
00101	15*	C				000000
00101	16*	C				000000
00101	17*	C	WRITTEN BY - ADAM LLOYD LATEST REVISION NOV 75			000000
00101	18*	C				000000
00101	19*	C				000000
00101	20*	C	INPUT/OUTPUT LIST			000000
00101	21*	C				000000
00101	22*	C	X1 STATE VARIABLE ANY OUTPUT STATE			000000
00101	23*	C	X1DOT STATE VARIABLE DERIVATIVE ANY OUTPUT STATE			000000
00101	24*	C	IX1 INTEGRATOR CONTROL --- PROGRAM VAR			000000
00101	25*	C	F0 TRANSFER FUNCTION OUTPUT ANY OUTPUT VAR			000000
00101	26*	C	FIN TRANSFER FUNCTION INPUT ANY INPUT VAR			000000
00101	27*	C	TC1 TIME CONSTANT (NUMERATOR) SECS INPUT PARAM			000000
00101	28*	C	TC2 TIME CONSTANT (DENOMINATOR) SECS INPUT PARAM			000000
00101	29*	C	GA1 TRANSFER FUNCTION GAIN --- INPUT PARAM			000000
0010	30*		COMMON/CJO/INEAD,IWRITE,IDIAS			000000
0010	31*		F0=(X1+FIN*TC1+GA1)/TC2			000000

BCS 40180-2

00105	32*
00107	33*
00110	34*

IF(IX1.WE.Q:XI00Y= 5A)FJN-FO
RETURN
END

000005
000013
000030

7.22 ELECTRICAL LOAD

This component represents electrical load either by a user-specified data file time history or by a set of random numbers with user-specified daily, weekly, and yearly average profiles and user-specified random variation. It also computes the value of the power delivered to the load by the system. This value delivered is determined from a user-specified value per kwh. This value may be input in tabular form as a function of time of day, time of year, or any other system parameter.

If the user selects to have the electrical load represented by random numbers, then the load (LO2) is generated from the following equation:

Basic Equation

$$LO2 = [PD(TD) + CN(t)] * PW(DW) * PY(WY) * NC$$

where

PD, PW, PY are the daily, weekly, and yearly profiles, respectively, and TD, DW, WY are the time of day, day of the week, and week of the year, respectively. NC is a normalizing constant.

CN is a colored noise term with user-specified correlation time, standard deviation and mean.

<u>Tables</u>		<u>Description</u>	<u>Units</u>
PD		Daily profile (tabular with TD)	kw
PW		Weekly profile (tabular with DW)	arbitrary
PY		Yearly profile (tabular with WY)	arbitrary

Inputs

Parameter/Port

P	1	Power delivered	kw
MP	1	Maximum Input Power deliverable ($D = 1 \times 10^{10}$)	kw
NC		Normalizing constant	
VE		Value of Electrical Energy	\$/kwh
LØ	1	Electrical load data file input	kw
TD		Time of day	-
DW		Day of week	-
WY		Week of year	-
CT		Correlation time of random noise	hr
MN,STD		Mean ($D = 0$) and std. deviation of random noise	kw
EF	1	Input Power Efficiency	-

Outputs

Variable/Port

RE	1	Power request	kw
VDE		Value of energy delivered (state)	\$
LØ	2	Electrical load	kw
TIM		Last time a random sample was used	hr
CN		Colored noise sample	kw

Statistics

SRE		Total energy requested	kwh
SDE		Total energy delivered	kwh
PC		Percentage of load met	-

D - Default values supplied

Calculation Sequence

1) Initialize CN(0) (first pass)

2) Check for data file input

If L01 = .99999 go to 3)

L02 = L01 and go to 5)

3) Generate colored noise CN

If TIM = TIME go to 5)

$$A = \begin{cases} \exp(-\Delta/CT), & CT > 0, \Delta = \text{integration step size, hr} \\ 0, & CT = 0 \end{cases}$$

$$CN = A \times CN + W,$$

Where W is white noise generated by RN with

$$\text{Mean} = MN * (1-A) \text{ and standard deviation} = STD * \sqrt{1-A^2}$$

4) Compute L02

$$L02 = (PD(TD) + CN) * PW(DW) * PY(WY) * NC$$

$$TIM = TIME$$

5) Power request and value delivered

$$RE = \text{MIN}(MP, L02) / EF1$$

6) Statistics

$$VDE = P1 * VE$$

$$SRE = SRE + L02 * \Delta / 2$$

$$SDE = SDE + P1 * \Delta / 2$$

$$PC = 100. * SDE / SRE$$

[illegible]

10

5

```

00137 77*      IF(CT.GT.0.1) A = EXP(-TIME/CT)
00141 78*      WPM = XNM*(1.-A)
00142 79*      WSD = STD*SQRT(1.-A+A1)
00143 80*      CALL RNTW,AX,WSD,WPM)
00144 81*      WN = XN*A + N
00144 82*      C      COMPUTE ELECTRICAL LOAD DEMAND
00144 83*      C
00145 84*      DLO = TBLU1(1D,PO(4),PO(ND+4),1,-ND)
00146 85*      WLO = TBLU1(0V,PW(4),PW(NW+4),1,-NW)
00147 86*      YLO = TBLU1(VY,PY(4),PY(NY+4),1,-NY)
00150 87*      LOZ = (DLO*XN)+WLO + YLO*YNC
00151 88*      TIME = TIME
00152 89*      150 RE = AMIN1(PHAX,LOZ)/EF
00152 90*      C      PERFORMANCE STATISTICS
00152 91*      C
00153 92*      IF(JMPL,LC,2) RETURN
00155 93*      IF(TVD.NE. 0) DVD = PO+VE
00157 94*      SRE = SRE + LOZ*0.5*TIME
00160 95*      SDE = SDE + PO*0.5*TIME
00161 96*      IF(SRC.GT.0.) PC = 100.*SDE/SRE
00161 97*      C
00163 98*      IF(TIME.LT.TMAX) RETURN
00165 99*      CV = CV + VDE
00166 100*      CDE = CDE + SDE - PO*0.5*TIME
00167 101*      CRE = CRE + SRE - LOZ*0.5*TIME
00171 102*      RETURN
00171 103*      END

```

```

000073
000114
000122
000133
000141
000141
000141
000145
000164
000203
000222
000230
000233
000233
000233
000241
000250
000261
000266
000271
000271
000300
000307
000312
000316
000325
000470

```

7.23 MULTIPLY AND ADD

Inputs

<u>Parameter/Port</u>	<u>Description</u>
FIN	Input quantity
C1	Input quantity
C2	Input quantity

Outputs

<u>Variable/Port</u>	
F0	Output quantity

Calculation Sequence

$$F0 = C1 * FIN + C2$$

SUBROUTINE MA ENTRY POINT 000012

STORAGE USED CODE(1) 000016; DATA(0) 000004; BLANK COMMON(2) 000000

EXTERNAL REFERENCES (BLOCK, NAME)

0003 WERR36

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0000 000000 INJPS

00100	1*	CMA				000000
00101	2*		SUBROUTINE MA(F0,FIN,C1,C2)			000000
00101	3*	C				000000
00101	4*	C	PURPOSE - TO SIMULATE THE EQUATION	OUTPUT=C1*INPUT + C2		000000
00101	5*	C				000000
00101	6*	C				000000
00101	7*	C	METHOD - SEE CODING			000000
00101	8*	C				000000
00101	9*	C				000000
00101	10*	C	WRITTEN BY - ADAM LLOYD	LATEST REVISION	NOV 75	000000
00101	11*	C				000000
00101	12*	C				000000
00101	13*	C	LIMITATIONS - NONE			000000
00101	14*	C				000000
00101	15*	C				000000
00101	16*	C	INPUT/OUTPUT LIST			000000
00101	17*	C				000000
00101	18*	C	F0	OUTPUT VARIABLE	ANY	000000
00101	19*	C	FIN	INPUT VARIABLE	ANY	000000
00101	20*	C	C1	CONSTANT MULTIPLIER	---	000000
00101	21*	C	C2	CONSTANT ADDITION	---	000000
00103	22*		F0=C1*FIN + C2			000000
00104	23*		RETURN			000000
00105	24*		END			000015

MA

7.24 MULTIPLY, DIVIDE, AND ADD

Inputs

<u>Parameter/Port</u>	<u>Description</u>
FNA	Input quantity
FNB	Input quantity
C1	Input quantity
C2	Input quantity
C3	Input quantity
C4	Input quantity
C5	Input quantity.

Outputs

<u>Variable/Port</u>	
F0	Output quantity

Calculation Sequence

$$F0 = C1 * FNA + C2 * FNB + C3 * FNA * FNB + C4 * FNA / FNB + C5$$

SUBROUTINE MB ENTRY POINT 000051

STORAGE USED CODE(1) 000064; DATA(1) 000023; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 ERMESS 000002
0004 C10 000003

EXTERNAL REFERENCES (BLOCK, NAME)

0005 NUNUS
0006 N1024
0007 NERR35

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001	000030	30L	0000	000000	20F	0001	000042	30L	0004	000002	IDTAG	0003	000001	IERR
0003	000000	IFATAL	0000	000015	INJPN	0004	000000	IREAD	0004	I	000001	IWRITE		

```

00100      1*      CMB
00101      2*      SUBROUTINE MB(F0,FNA,FNB,C1,C2,C3,C4,C5)
00101      3*      C
00101      4*      C PURPOSE - TO SIMULATE THE EQUATION Y=C1*X1+C2*X2+C3*X3+C4*X4/C5
00101      5*      C
00101      6*      C
00101      7*      C WRITTEN BY - GEORGE DULEBA LATEST REVISION MAY 76
00101      8*      C
00101      9*      C
00101     10*      C LIMITATIONS - IF FNB=0 DURING DIVISION, FNB IS SET TO E-20.
00101     11*      C DIAGNOSTIC MESSAGE IS GIVEN.
00101     12*      C
00101     13*      C
00101     14*      C INPUT/OUTPUT LIST
00101     15*      C
00101     16*      C F0 OUTPUT VARIABLE ANY OUTPUT VAR
00101     17*      C FNA INPUT VARIABLE A ANY INPUT VAR
00101     18*      C FNB INPUT VARIABLE B ANY INPUT VA
00101     19*      C C1 MULTIPLIER 1 ANY INPUT VAR
00101     20*      C C2 MULTIPLIER 2 ANY INPUT VAR
00101     21*      C C3 MULTIPLIER 3 ANY INPUT VAR
00101     22*      C C4 MULTIPLIER 4 ANY INPUT VAR
00101     23*      C C5 ADDITIVE VARIABLE ANY INPUT VAR
00101     24*      C
00101     25*      C
00103     26*      COMMON/ERMESS/IFATAL,IERR
00104     27*      COMMON/C10/IREAD,IWRITE,IDTAG

```

MB

00105	28*		F0= C1*FNA + C2*FNB + C3*FNA*FNB + C5	000000
00106	29*		IF(C4.(0.0.99999) GO TO 30	000012
00110	30*		IF(FNB.E0.0.) GO TO 10	000015
00112	31*		F0= F0 + C4*FNA/FNB	000017
00113	32*		RETURN	000024
00114	33*	10	WRITE(3WRITE,20)	000030
00116	34*	20	FORMAT(1/.10X, 53HWARNING- DIVISOR IN NB EQUALS 0., HAS BEEN SET=1.	000034
00114	35*	2E-20)		000034
00117	36*		F0= F0 + C4*FNA*1.E+20	000034
00120	37*	30	RETURN	000042
00121	38*		END	000063

7.25 MULTIPLY AND ADD

Inputs

<u>Parameter/Port</u>	<u>Description</u>
FIN	Input quantity
FI0	Input quantity
FIP	Input quantity
C1	Input quantity
C2	Input quantity
C3	Input quantity
C4	Input quantity

Outputs

<u>Variable/Port</u>	
F0	Output quantity

Calculation Sequence

$$F0 = C1*FIN + C2*FI0 + C3*FIP + C4$$

[illegible]

W

7.26 AC INDUCTION MOTOR

The induction motor produces mechanical power and torque proportional to slip speed, i.e. power and torque approach zero as the rotor approaches synchronous speed. Two power losses are modeled: a constant multiplicative term due to resistive heating and an additive term due to mechanical friction. Default parameters are based on a conventional squirrel-cage induction motor/generator machine.

Basic Equations

$$P2 = EE * P1 + DA * RS^2 * C$$

where

$P1, P2$ = input and output power

EE = electrical efficiency

DA = mechanical damping

C = conversion constant

Inputs

<u>Parameter/Port</u>		<u>Description</u>	<u>Units</u>
P	1	Input power	kw
DA		Mechanical damping (D = 0)	joule-sec
RS		Rotor speed	rpm
RSY		Synchronous rotor speed (D = 1800)	rpm
SR		Stator resistance (D = 8/RAP)	ohms
V0		Rated input voltage (D = 400)	volts
RAP		Rated input power	kw
RAS		Rated power slip (D = 0.05)	-
EF	1	Input product efficiency	-
MP	1	Maximum input power (D = 1×10^6)	kw
CC		Capital cost/year	\$
CM		Maintenance cost/year	\$

Outputs

<u>Variable/Port</u>			
P	2	Output mechanical power	kw
EE		Electrical efficiency	-
T0		Mechanical torque	ft-lb
PL		Power loss	kw
EF	2	Output product efficiency	-
MP	2	Output maximum power	kw

Statistics

MT	Maximum torque	ft-lb
MPN	Maximum output power/rated power	-
SP	Output energy sum	kwh

D - Default values supplied.

Calculation Sequence

- 1) Compute electrical efficiency (first pass only)

$$I_{RAT} = RAP * 1000 / V_0$$

$$EE = 1 - SR * I_{RAT}^2 / RAP * 1000$$

- 2) Diagnostics

$$P_1 > RAP \Rightarrow \text{DIAGNOSTIC}$$

$$SLIP = 1 - RS / RSY > RAS \Rightarrow \text{DIAGNOSTIC}$$

- 3) Output power and power loss

$$\omega = RS * (2\pi / 60)$$

$$P_2 = EE * P_1 - DA * \omega^2 / 1000$$

$$P_L = P_1 - P_2$$

- 4) If $P_2 > 0$ go to 5)

$$P_1 > 0 \Rightarrow \text{DIAGNOSTIC}$$

$$EF2 = EF1, MP2 = \text{MIN}(MP1, RAP)$$

Go to 7)

- 5) Compute torque

$$T_0 = P_2 * 1000 / \omega * k$$

$$k = 1.3558 \text{ joules/ft-lb}$$

Calculation Sequence Cont.

6) Efficiency and maximum output power

$$EF2 = EF1 * (P2/P1)$$

$$MP2 = \text{MIN}(MP1, RAP1 * (P2/P1))$$

7) Compute Statistics and Costs

STORAGE USED CODE(1) 000434: DATA(1) 000102: BLANK COMMON(2) 000000

0003	CIMPL	000002
0004	CTIME	000001
0005	PSIMUL	000010
0006	COST	000010

0007 WUDUS
JQ10 M1021
0011 MERRIS

[illegible][illegible]

W

00173	79*	408 FORMAT(1H0,19H STATOR RESISTANCE ,F12.3,12H OH DARNING ,	000172
00173	80*	XF12.3,20H TOO HIGH FOR MOTOR)	000172
00174	81*	IF(1IMPL-0.2)ICNT=ICNT+1	000172
00174	82*	C	000172
00174	83*	C EFFICIENCY AND MAXIMUM OUTPUT POWER	000172
00174	84*	C	000172
00176	85*	409 CONTINUE	000201
00177	86*	P2=0.	000201
00200	87*	EF2=EF1	000201
00201	88*	MP2=AMIN1(MP1,RAP)	000203
00202	89*	GO TO 500	000211
00203	90*	400 EF2=EF1*P2/P1	000213
00204	91*	MP2=AMIN1(MP1,RAP)*P2/P1	000216
00205	92*	IF(RS.NC-C.1)TO=P2*737.6/OMEGA	000226
00205	93*	C	000226
00207	94*	500 IF(1IMPL.LE.1)RETURN	000235
00207	95*	C	000235
00207	96*	C STATISTICS	000235
00207	97*	C	000235
00211	98*	MT=AMAX1(ITD,MT)	000243
00212	99*	MPN=AMAX1(P2/RAP,MPN)	000251
00213	100*	SP=SP+P2*TIME	000260
00213	101*	C	000260
00214	102*	IF(TIME.LT.1MAX)RETURN	000264
00216	103*	CC1=CC1+CC	000273
00217	104*	CM1=CM1+CM	000276
00217	105*	C	000276
00220	106*	RETURN	000301
00221	107*	END	000433

7.27 POWER ACCUMULATOR

This component sums power from four input ports and allocates power requests to each port's source of power generation. If an input power request (load) exceeds the maximum power that can be delivered by the port of highest priority, then the remaining load is allocated according to weight within priority, and then allocated to the next priority ports. (See 1.2.3 for further discussion.)

Inputs¹

<u>Parameter/Port</u>	<u>Description</u>	<u>Units</u>
RE 0	Load request	kw
EF 1,2,3,4	Input efficiency from port i	-
P 1,2,3,4	Input power from port i (default = 0.)	kw
PS 1,2,3,4	Priority sequence (default=1,2,3,4)	-
F 1,2,3,4	Allocation weight (for equal priorities)	-
MP 1,2,3,4	Maximum power (default = 1×10^8)	kw

Outputs

<u>Variable/Port</u>			
MP 0	Maximum deliverable power ($\sum MP(i)$)		kw
RE 1,2,3,4	Power request for port i		kw
P 0	Power output		kw
SP	Supplemental power request to meet load (Power deficit) = $RE_0 - \sum MP_i$		kw

Statistics

SRE	Sum of energy requested	kwh
PC 1,2,3,4	Percent of cumulative load request delivered by port i	%

¹ No capital costs assigned since this is an allocation component, not a physical device.

CALCULATION LOGIC

PA FAIR SHARE ALLOCATION

SUBROUTINE PA ENTRY POINT 000520

STORAGE USED CODE(1) 000752; DATA(0) 000110; BLANK COMMON(2) 000000

COMMON BLOCKS

0001 CIMPL 000001
0004 CSIMUL 000010

EXTERNAL REFERENCES (BLOCK, NAME)

0005 NERR36

STORAGEC ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000376 1000L	0001 000401 2000L	0001 000235 2346	0001 000244 2416	0001 000264 2546
0001 000306 2736	0001 000325 3056	0001 000363 3256	0001 000104 40L	0001 000301 400L
0001 000417 500L	0001 000301 600L	0001 000316 700L	0001 000161 80L	0001 000353 800L
0001 000372 900L	0004 000000 DUM	0000 R 000030 FR	0000 R 000042 FRU	0000 I 000037 I
0001 I 000000 IMPL	0000 000055 INJPS	0000 I 000041 M	0000 R 000034 LL	0000 R 000035 LOLD
0000 R 000014 MP	0000 R 000004 PR	0000 R 000000 R	0000 R 000024 SMP	0000 R 000044 SRI
0000 R 000043 SR0	0000 R 000020 SV	0004 R 000006 TIME	0000 R 000036 TIME1	0004 C00007 THAY
0000 R 000016 W	0000 R 000040 XT			

00100	1*	CPA		000000
00101	2*		SUBROUTINE PAIMP0,	000000
00101	3*	1	R1, R2, R3, R4,	000000
00101	4*	2	PO,SP,	000000
00101	5*	3	SR,PC1,PC2,PC3,PC4,	000000
00101	6*	4	RO,	000000
00101	7*	5	EF1, EF2, EF3, EF4,	000000
00101	8*	6	P1, P2, P3, P4,	000000
00101	9*	7	PR1, PR2, PR3, PR4,	000000
00101	10*	8	W1, W2, W3, W4,	000000
00101	11*		MP1, MP2, MP3, MP4,	000000
00101	12*	C		000000
00101	13*	C	PURPOSE. MODEL POWER ACCUMULATOR	000000
00101	14*	C		000000
00101	15*	C	METHOD. PRIMARY REQUEST ALLOCATION RESULTING FROM PRIORITY	000000
00101	16*	C	ASSIGNMENTS. SECONDARY REQUEST ALLOCATION RESULTING	000000
00101	17*	C	FROM WEIGHT ASSIGNMENTS.	000000
00101	18*	C	THAT IS, REQUESTS ARE ALLOCATED ACCORDING TO	000000
00101	19*	C	* PORT PRIORITY (HIGHEST PRIORITY = 1)	000000
00101	20*	C	* PORT WEIGHTS (IN CASE OF EQUAL PRIORITIES.)	000000
00101	21*	C		000000
00101	22*	C	FORMAL ARGUMENT DEFINITION.	000000
00101	23*	C	R1,..., R4 POWER REQUESTS IN KW (OUTPUTS)	000000

PA

00101	24*	C	MP0	TOTAL MAXIMUM POWER	OUTPUT	000000
00101	25*	C	SP	SURPLUS REQUEST	OUTPUT	000000
00101	26*	C	PO	TOTAL LOAD IN KW	OUTPUT	000000
00101	27*	C	SR	SUM OF ENERGY REQUESTED, KWH	OUTPUT	000000
00101	28*	C	PC1,...,PC4	PERCENT OF CUM LOAD DELIVERED	OUTPUT	000000
00101	29*	C	RD	TOTAL POWER REQUESTED, KW	INPUT	000000
00101	30*	C	P1,..., P4	INPUT POWER IN KW	INPUT	000000
00101	31*	C	PR1,..., PR4	PORT PRIORITIES	INPUT	000000
00101	32*	C	W1,..., W4	PORT WEIGHTS	INPUT	000000
00101	33*	C	MP1, ..., MP4	MAXIMUM POWERS	INPUT	000000
00101	34*	C	EF1, ..., EF4	EFFICIENCIES	INPUT	000000
00101	35*	C	COMMON STORAGE			000000
00103	36*		COMMON / CINPL / IMPL			000000
00104	37*		COMMON / CSINUL / DUM161, TIME, THAX			000000
00105	38*		REAL MPC,MP1,MP2,MP3,MP4			000000
00105	39*	C				000000
00105	40*	C	LOCAL VARIABLES			000000
00105	41*	C				000000
00105	42*	C				000000
00105	43*	C	R(K) IS THE POWER REQUEST AT PORT K			000000
00106	44*		REAL R(4)			000000
00106	45*	C				000000
00106	46*	C	PR(K) IS THE PRIORITY ASSIGNED TO PORT K			000000
00107	47*		REAL PR(4)			000000
00107	48*	C				000000
00107	49*	C	W(K) IS THE WEIGHT ASSIGNED TO PORT K			000000
00110	50*		REAL W(4)			000000
00110	51*	C				000000
00110	52*	C	MP(K) IS MAXIMUM POWER TO BE ALLOCATED TO PORT K			000000
00111	53*		REAL MP(4)			000000
00111	54*	C				000000
00111	55*	C	SW(I) IS THE SUM OF THE WEIGHTS ASSIGNED TO PRIORITY-I PORTS			000000
00112	56*		REAL SW(4)			000000
00112	57*	C				000000
00112	58*	C	SMP(I) IS THE SUM OF THE MAXIMUM POWER AT PRIORITY-I PORTS			000000
00113	59*		REAL SMP(4)			000000
00113	60*	C				000000
00113	61*	C	FRU IS "FAIR SHARE" UNIT FOR PRIORITY-I PORTS			000000
00113	62*	C				000000
00113	63*	C	FR(K) IS THE COMPUTED "FAIR SHARE" REQUEST FOR PORT K			000000
00114	64*		REAL FR(4)			000000
00114	65*	C				000000
00114	66*	C	LL IS THE LOAD LEFT AT EACH POINT IN THE ITERATION			000000
00115	67*		REAL LL,LOLD			000000
00115	68*	C				000000
00115	69*	C	IF IMPL IS ZERO, THEN ASSIGN DEFAULT VALUES			000000
00116	70*		IF (IMPL .GT. 0) GO TO 40			000000
00120	71*		RD = 0.0			000000
00121	72*		IF (PR1 .EQ. 0.99999) PR1 = 1.0			000003
00123	73*		IF (PR2 .EQ. 0.99999) PR2 = 2.0			000010
00125	74*		IF (PR3 .EQ. 0.99999) PR3 = 3.0			000015
00127	75*		IF (PR4 .EQ. 0.99999) PR4 = 4.0			000022
00131	76*		IF (MP1 .EQ. 0.99999) MP1 = 1.0E0			000027
00133	77*		IF (MP2 .EQ. 0.99999) MP2 = 1.0E0			000034
00135	78*		IF (MP3 .EQ. 0.99999) MP3 = 1.0E0			000041
00137	79*		IF (MP4 .EQ. 0.99999) MP4 = 1.0E0			000046
00141	80*		IF (P1 .EQ. .99999) P1=0.0			000053

PA

```

00143 81*      IF(P2 .EQ. .99999) P2=0.0
00145 82*      IF(P3 .EQ. .99999) P3= 0.0
00147 83*      IF(P4 .EQ. .99999) P4=0.0
00151 84*      SR=C.
00152 85*      PC1=0.
00153 86*      PC2=0.
00154 87*      PC3=0.
00155 88*      PC4=0.
00156 89*      TIME1= 0.5*TIME
00157 90*      CONTINUE
00157 91*      C
00157 92*      C      IF THE TOTAL MAXIMUM POWER IS .LE. TOTAL POWER
00157 93*      C      REQUESTED, THEN SUBMIT REQUESTS AT MAX-POWER, SET REQUEST
00157 94*      C      SURPLUS EQUAL TO THE DIFFERENCE, AND RETURN
00160 95*      PC = P1 + P2 + P3 + P4
00161 96*      IF(PR1.LE.0.0) MP1=0.
00163 97*      IF(PR2.LE.0.0) MP2=0.
00165 98*      IF(PR3.LE.0.0) MP3=0.
00167 99*      IF(PR4.LE.0.0) MP4=C.
00171 100*      MPJ = MP1 + MP2 + MP3 + MP4
00172 101*      IF (MPC .GT. RD) GO TO 80
00174 102*      R1 = MP1/EF1
00175 103*      R2 = MP2/EF2
00176 104*      R3 = MP3/EF3
00177 105*      R4 = MP4/EF4
00200 106*      SP = RD - MPC
00201 107*      GO TO 500
00202 108*      80 CONTINUE
00202 109*      C
00202 110*      C      PROCEED WITH ALLOCATION ALGORITHM SINCE THE SUM OF
00202 111*      C      ALL MAXIMUM POWER INPUTS EXCEEDS THE TOTAL REQUEST RD
00202 112*      C
00202 113*      C      INITIALIZATION
00203 114*      LL = RD
00204 115*      R1 = 0.0
00205 116*      R2 = 0.0
00206 117*      R3 = 0.0
00207 118*      R4 = 0.0
00210 119*      SP = 0.0
00210 120*      C
00210 121*      C      IF THE TOTAL REQUEST (FOR LOAD) IS ZERO, THEN RETURN
00211 122*      IF (RC .LE. 0.0) GO TO 500
00213 123*      R11=R1
00214 124*      R12=R2
00215 125*      R13=R3
00216 126*      R14=R4
00217 127*      PR(1) = PR1
00220 128*      PR(2) = PR2
00221 129*      PR(3) = PR3
00222 130*      PR(4) = PR4
00223 131*      W(1) = W1
00224 132*      W(2) = W2
00225 133*      W(3) = W3
00226 134*      W(4) = W4
00227 135*      MP(1) = MP1
00228 136*      MP(2) = MP2
00229 137*      MP(3) = MP3

```

```

000057
000063
000067
000073
000074
000075
000076
000077
000100
000104
000104
000104
000104
000104
000110
000114
000120
000124
000130
000135
000140
000143
000146
000151
000154
000157
000161
000161
000161
000161
000162
000163
000164
000165
000166
000166
000166
000167
000171
000173
000175
000177
000201
000203
000205
000207
000211
000213
000215
000217
000221
000223
000225

```

```

00232 138*      MP(I) = MP4
00232 139*      C
00232 140*      C
00232 141*      C      ITERATE ON PRIORITY I FOR I = 1, 2, 3, 4
00232 142*      C
00233 143*      DO 1000 I = 1, 4
00233 144*      C
00236 145*      XI = I
00236 146*      C      OBTAIN SUM OF MAXIMUM POWER FOR PORTS WITH PRIORITY I
00237 147*      SMP(I) = 0.0
00240 148*      DO 100 K = 1, 4
00243 149*      IF (PRIN(I) .EQ. XI) SMP(I) = SMP(I) + MP(K)
00245 150*      100 CONTINUE
00245 151*      C
00245 152*      C      IF NO PRIORITY-I MAXIMUM POWER EXISTS, THEN PROCEED WITH
00245 153*      C      THE NEXT HIGHER PRIORITY
00247 154*      IF (SMP(I) .EQ. 0.0) GO TO 1000
00247 155*      C
00247 156*      C      IF THE SUM OF ALL PRIORITY-I MAXIMUM POWER .GT. LOAD
00247 157*      C      LEFT, THEN GO AROUND
00251 158*      IF (SMP(I) .GT. LL) GO TO 400
00251 159*      C
00251 160*      C      THE SUM OF ALL PRIORITY-I MAXIMUM POWER .LE. LOAD
00251 161*      C      LEFT, SO SUBMIT EACH PRIORITY-I REQUEST
00253 162*      DO 200 K = 1, 4
00256 163*      IF (PRIN(I) .EQ. XI) R(K) = MP(K)
00260 164*      200 CONTINUE
00260 165*      C
00263 166*      C      UPDATE LOAD LEFT
00262 167*      LL = LL - SMP(I)
00262 168*      C
00262 169*      C      IF THE REMAINING LOAD IS ZERO, THEN EXIT THE ITERATION
00263 170*      IF (LL .LE. 0.0) GO TO 2000
00263 171*      C
00263 172*      C      OTHERWISE, PROCEED WITH NEXT HIGHER PRIORITY
00265 173*      GO TO 1000
00265 174*      C
00266 175*      400 CONTINUE
00266 176*      C
00266 177*      C      THE SUM OF THE PRIORITY-I MAXIMUM POWER EXCEEDS THE
00266 178*      C      LOAD LEFT, SO COMPUTE AND SUBMIT FAIR SHARE REQUESTS
00266 179*      C      TO EACH PRIORITY-I PORT
00266 180*      C
00267 181*      600 CONTINUE
00267 182*      C
00267 183*      C      SAVE LL FOR LATER REFERENCE
00270 184*      LOLD = LL
00270 185*      C
00270 186*      C      DETERMINE FAIR SHARE UNITS FOR ALL PRIORITY-I
00270 187*      C      PORTS TO WHICH NO REQUEST HAS BEEN SUBMITTED
00271 188*      SW(I) = 0.0
00272 189*      DO 700 K = 1, 4
00275 190*      IF (R(K) .NE. 0.0) GO TO 700
00277 191*      IF (PRIN(I) .EQ. XI) SW(I) = SW(I) + U(K)
00301 192*      700 CONTINUE
00303 193*      FRU = 1.0 / SW(I)
00303 194*      C

```

[illegible]

PA

P/

00363 252*
 00364 253*
 00365 254*

PC= PC*SR0 + P*SR1
 RETURN
 END

00070
 00074
 000751

7.28 POWER DIVIDER

This component allocates power to four ports plus surplus based on priority, port requests, and allocation weights for equal priority ports. Each port is assigned a priority sequence from 1 to 4, and a weighting $F_i > 0$, $i=1,2,3,4$ for proportional allocation among equal priority ports. If power available exceeds the power requested for a set of ports of equal priority, then the remaining power is allocated to ports having the next highest priority. If power available is less than the power requested for ports of equal priority then power is allocated between them in proportion to their respective allocation weights.

The total power request is the sum of the port requests divided by input efficiency. The maximum power outputs $MP1, \dots, MP4$ are necessary for direct connections to a power accumulator PA. These variables may be used as maximum power inputs to other components, although such connections are not required. (See 1.2.3 for further discussion.)

Inputs¹

<u>Parameter/Port</u>	<u>Description</u>	<u>Units</u>
P 0	Input power	kw
RE 1,2,3,4	Power requests of output ports	kw
PS 1,2,3,4	Priority sequence (default = 1,2,3,4)	-
F 1,2,3,4	Allocation weight (for equal priorities)	-
MP	Maximum input power (default = 1×10^8)	kw
EF	Input efficiency	-

Outputs

<u>Variable/Port</u>		
P 1,2,3,4	Output power for port i	kw
RE 0	Output power request	kw
SP	Surplus power	kw
MP 1,2,3,4	Output maximum power based on MP	kw

¹ No capital costs assigned since this is an allocation component, not a physical device.

CALCULATION LOGIC

PD FAIR SHARE ALLOCATION

pr

பு

00113	83*	C		000000
00113	84*	C	IF IMPL IS ZERO, THEN ASSIGN DEFAULT VALUES	000000
00114	85*		IF (IMPL .GT. 0) GO TO 40	000000
00114	86*		R1 = 0.0	000007
00117	87*		R2 = 0.0	000003
00120	88*		R3 = 0.0	000009
00121	89*		R4 = 0.0	000005
00122	90*		IF (PH .EQ. .99999) PH = 1.E0	000006
00124	91*		IF (PR1 .EQ. 0.99999) PR1 = 1.0	000013
00126	92*		IF (PR2 .EQ. 0.99999) PR2 = 2.0	000020
00130	93*		IF (PR3 .EQ. 0.99999) PR3 = 3.0	000025
00132	94*		IF (PR4 .EQ. 0.99999) PR4 = 4.0	000032
00134	95*		IF (PH .EQ. .99999) PH = 1.E0	000037
00134	96*	C	INITIALIZATION OF F'S	000037
00134	97*	C		000037
00136	98*		NO CONTINUE	000045
00137	99*		F1 = 1.	000045
00140	100*		F2 = 1.	000046
00141	101*		F3 = 1.	000047
00142	102*		F4 = 1.	000050
00142	103*	C		000050
00142	104*	C	IF THE TOTAL POWER REQUESTED IS .LE. TOTAL POWER	000050
00142	105*	C	INPUT, THEN SATISFY REQUESTS, SET POWER SURPLUS	000050
00142	106*	C	EQUAL TO THE DIFFERENCE, AND RETURN	000050
00143	107*		IF (PR1 .LE. 0.0) R1 = 0.0	000051
0145	108*		IF (PR2 .LE. 0.0) R2 = 0.0	000055
00147	109*		IF (PR3 .LE. 0.0) R3 = 0.0	000061
00151	110*		IF (PR4 .LE. 0.0) R4 = 0.0	000065
00153	111*		RD = R1 + R2 + R3 + R4	000071
00154	112*		IF (RD .GT. PD) GO TO 40	000074
00156	113*		P1 = R1	000101
00157	114*		P2 = R2	000103
00160	115*		P3 = R3	000105
00161	116*		P4 = R4	000107
00162	117*		SP = PD - RD	000111
00163	118*		GO TO 3000	000114
00164	119*		NO CONTINUE	000116
00164	120*	C		000116
00164	121*	C	PROCEED WITH ALLOCATION ALGORITHM SINCE THE SUM OF	000116
00164	122*	C	ALL REQUESTS EXCEEDS THE TOTAL AVAILABLE POWER PD	000116
00164	123*	C		000116
00164	124*	C	INITIALIZATION	000116
00165	125*		PL = PD	000116
00166	126*		P1 = 0.0	000117
00167	127*		P2 = 0.0	000120
00170	128*		P3 = 0.0	000121
00171	129*		P4 = 0.0	000122
00172	130*		SP = 0.0	000123
00172	131*	C		000123
00172	132*	C	IF THE TOTAL POWER IS ZERO, THEN RETURN	000123
00173	133*		IF (PS .EQ. 0.0) GO TO 3000	000124
00175	134*		P11 = P1	000126
00176	135*		P12 = P2	000130
00177	136*		P13 = P3	000132
00201	137*		P14 = P4	000134
0020	138*		R11 = R1	000136
0020	139*		R12 = R2	000140

PC

00203	140*	R(3) = R3	000142
00204	141*	R(4) = R4	000144
00205	142*	PR(1) = PR1	000146
00206	143*	PR(2) = PR2	000150
00207	144*	PR(3) = PR3	000152
00210	145*	PR(4) = PR4	000154
00211	146*	W(1) = W1	000156
00212	147*	W(2) = W2	000160
00213	148*	W(3) = W3	000162
00214	149*	W(4) = W4	000164
00214	150*		000166
00214	151*		000168
00214	152*	ITERATE ON PRIORITY I FOR I = 1, 2, 3, 4	000169
00214	153*		000172
00215	154*	DO 1000 I = 1, 4	000172
00215	155*		000172
00220	156*	XI = I	000172
00220	157*	OBTAIN SUM OF REQUESTS FROM PORTS WITH PRIORITY I	000175
00221	158*	SR(1) = 0.0	000176
00222	159*	WT=0.0	000202
00223	160*	DO 100 K = 1, 4	000202
00226	161*	IF (PR(K) .EQ. XI) SR(1) = SR(1) + R(K)	000207
00230	162*	IF (PR(K) .EQ. XI) WT = WT + W(K)	000217
00232	163*	100 CONTINUE	000217
00232	164*		000225
00234	165*	IF (PR1 .EQ. XI) F1 = W1/WT	000233
00236	166*	IF (PR2 .EQ. XI) F2 = W2/WT	000241
00240	167*	IF (PR3 .EQ. XI) F3 = W3/WT	000247
00242	168*	IF (PR4 .EQ. XI) F4 = W4/WT	000247
00244	169*	IF (PL .LE. 0.0) GO TO 1000	000252
00244	170*		000252
00244	171*	IF NO PRIORITY-I REQUESTS EXIST, THEN PROCEED WITH	000252
00244	172*	THE NEXT HIGHER PRIORITY	000254
00244	173*	IF (SR(1) .EQ. 0.0) GO TO 1000	000254
00244	174*		000263
00246	175*	IF THE SUM OF ALL PRIORITY-I REQUESTS .GT. POWER	000271
00246	176*	AVAILABLE, THEN GO AROUND	000271
00250	177*	IF (SR(1) .GT. PL) GO TO 400	000271
00250	178*		000274
00250	179*	THE SUM OF ALL PRIORITY-I REQUESTS .LE. POWER	000276
00250	180*	AVAILABLE, SO FULFILL EACH PRIORITY-I REQUEST	000276
00252	181*	DO 200 K = 1, 4	000276
00255	182*	IF (PR(K) .EQ. XI) P(K) = R(K)	000276
00257	183*	200 CONTINUE	000276
00257	184*		000276
00257	185*	UPDATE POWER AVAILABLE	000276
00261	186*	PL = PL - SR(1)	000276
00262	187*	GO TO 1000	000276
00262	188*		000276
00263	189*	400 CONTINUE	000276
00263	190*		000276
00263	191*	THE SUM OF THE PRIORITY-I REQUESTS EXCEEDS THE	000276
00263	192*	POWER AVAILABLE, SO COMPUTE AND ALLOCATE FAIR	000276
00263	193*	SHARE TO EACH PRIORITY-I PORT	000276
00263	194*		000276
00264	195*	600 CONTINUE	000276
00264	196*		000276

PD

५

```

00341 254*      PQ = P(R1
00342 255*      SP = PL
00343 256*      3000 PH1=PH*F1
00344 257*      PH2=PH*F2
00345 258*      PH3=PH*F3
00346 259*      PH4=PH*F4
00347 260*      RG= ANIN1(RG,PH)/EF
00350 261*      RETURN
00351 262*      END

```

```

000401
000403
000406
000410
000413
000416
000421
000430
000441

```

PD

7.29 PRIORITY INTERRUPT

This component is used by the storage components to change priority of the power requests when minimum or maximum capacity is approached.

Inputs

<u>Parameter/Port</u>		<u>Description</u>
PS	1	Input priority for PS4 output
PS	3	Input priority for PS2 output (default=PS1)
INT		Interrupt flag

Outputs

<u>Variable/Port</u>		
PS	2	Output priority for charge cycle
PS	4	Output priority for discharge cycle

Equations

$$\begin{aligned}
 PS2 &= PS1 && \text{if } INT=0 \\
 PS2 &= 1 && \text{if } INT > 0 \\
 PS2 &= 0 && \text{if } INT < 0 \\
 PS4 &= PS3 && \text{if } INT \leq 0 \\
 PS4 &= 0 && \text{if } INT > 0
 \end{aligned}$$

SUBROUTINE PI ENTRY POINT 000040

STORAGE USED CODE(1) 000061; DATA(0) 000010; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 CIMPL 000001

EXTERNAL REFERENCES INLOCN, NAME1

0004 NERR34

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000010 10L 0003 1 000000 IMPL 0000 000002 INJPS

```

00100 1* CPI 000000
00101 2* SUBROUTINE PI(PS2,PS4,PS1,PS3,INT) 000020
00101 3* C 000000
00101 4* C PURPOSE CHANGE PRIORITY OF POWER ALLOCATION TO STORAGE COMPONENTS 000000
00101 5* C 000000
00101 6* C WRITTEN BY A.W.WARREN VERSION 1, APRIL 14 1977 000000
00101 7* C 000000
00101 8* C CALL SEQUENCE 000000
00101 9* C PS2 - OUTPUT PRIORITY (0 TO 4) 000000
00101 10* C PS4 - OUTPUT PRIORITY (COMPLEMENT TO PS2) 000000
00101 11* C PS1 - INPUT PRIORITY FOR PS2 000000
00101 12* C PS3 - INPUT PRIORITY FOR PS4 000000
00101 13* C INT - INTERRUPT FLAG 000000
00101 14* C 0= NO INTERRUPT 000000
00101 15* C 1= INCREASE ALLOCATION PRIORITY 000000
00101 16* C -1= DECREASE ALLOCATION PRIORITY 000000
00101 17* C 000000
00103 18* REAL INT 000000
00104 19* COMMON /CIMPL/IMPL 000000
00105 20* IF(IMPL.GT.0) GO TO 10 000000
00107 21* IF(PS3.EQ. .99999) PS3=PS1 000002
00107 22* C 000002
00111 23* 10 PS2=PS1 000010
00112 24* PS4=PS3 000011
00113 25* IF(INT.GT.0.) PS2=1. 000013
00115 26* IF(INT.LT.0.) PS2=0. 000020
00117 27* IF(INT.GT.0) PS4= 0. 000024
00121 28* RETURN 000030
00122 29* END 000060

```

PI

7.30 HYDRAULIC PUMP

The hydraulic pump model is based on a constant speed design. The pump is assumed to be designed to a nominal operating point and input power. For off-design performance the pump efficiency is assumed to be functionally related to the square root of the mass flow rate.

Basic Equations

The output mass flow rate is based on the equations

$$M = P1 * EFF / (C1 * C2 * H1)$$

$$EFF = 1 - (1 - EFD1) * SQRT(MD / M)$$

where C1, C2 are conversion constants

Inputs

<u>Parameter/Port</u>		<u>Description</u>	<u>Units</u>
P	1	Input power	kw
H	1	Height of water above inlet	ft
EFD		Pump efficiency at design pt. (D = 0.90)	-
MD		Mass flow rate at design pt. (D = 2×10^5)	gal/h
EF	1	Input product efficiency	-
MP	1	Input maximum charging rate	kw
MA		Maximum allowable mass flow rate (D = 3×10^5)	gal/h
CK		Pump capacity cost coefficient ¹ (D = 0.011)	-
F0		Pump exponent for cost calculations (D = 0.5)	-
Y		Pumphead exponent for cost calculations (D=0.25)	-

Outputs

<u>Variable/Port</u>			
M		Output mass flow rate	gal/h
EFF		Pump efficiency	-
CO0		Pump cost/year	\$
EF	2	Output product efficiency	-
MP	2	Maximum output power	kw

Statistics

M2U		Maximum output mass flow rate	gal/h
-----	--	-------------------------------	-------

D - default values

¹CK = capital cost (known unit)/((MD*481.2)**F0*H1 **Y * expected life time)

1

The calculation sequence and default values assume a constant speed hydraulic pump nominally rated for 120KW and located 200 ft. below a reservoir. The equations relating the various physical quantities and the cost estimates are based on first principles and the data presented in Reference 1, and the cost estimates on Reference 2.

Calculation Sequence

$$C1 = 0.377 * 10^{-6} \frac{\text{kwh}}{\text{ft-lb}}$$

$$C2 = 8.3398 \text{ lb/gal}$$

1) Costs (first pass only)

$$CC = CK * (MD * 481.2)^{F0} * H1 * Y$$

-
1. L. Marks and T. Baumeister, "Mechanical Engineers Handbook", McGraw Hill, N.Y., 1958, Section 14, p. 19.
 2. Carson and Fogleman, "Comparison of Methods for Converting Existing Power Plants to Pumped Storage Facilities", International Engineering Company, Inc., 1974.

Calculation Sequence Cont.

2) Mass flow rate and pump efficiency

If $P1 \leq 0$, set $EFF = 1$, $M = 0$ and go to 3)

Solve the basic equations for M and EFF using:

$$X^3 - XA + B = 0$$

where

$$A = P1 / (C1 * C2 * H1)$$

$$B = A * (1 - EFD) * \sqrt{MD}$$

$$M = X^2$$

$$EFF = 1 - (1 - EFD) * \sqrt{MD} / X$$

3) Product efficiency and maximum charge rate

$$EF2 = EF1 * EFF$$

$$MP2 = \text{MIN}(MP1 * EFF, \text{MIN}(C1 * C2 * H1))$$

4) Compute Statistics and Costs

५

```

00101 24* C W1 - HEIGHT OF WATER ABOVE INLET, FT 000000
00101 25* C EFD - PUMP EFFICIENCY AT DESIGN POINT 000000
00101 26* C MD - MASS FLOW RATE AT DESIGN POINT, GAL/HR 000000
00101 27* C EF1 - INPUT PRODUCT EFFICIENCY 000000
00101 28* C MP1 - INPUT MAXIMUM CHARGING RATE, KW 000000
00101 29* C MM - MAXIMUM ALLOWABLE MASS FLOW RATE, GAL/HR 000000
00101 30* C CK - PUMP CAPACITY COST COEFFICIENT 000000
00101 31* C FO - PUMP EXPONENT FOR COST CALCULATIONS 000000
00101 32* C Y - PUMP HEAD EXPONENT FOR COST CALCULATIONS 000000
00101 33* C 000000
00103 34* COMMON /CINPL/IMPL /CTIME/TIME/CSIMUL/DUNIT/,TMAX /COST/CCI 000000
00103 35* C 000000
00104 36* REAL N,MP2,M2U,MD,MP1,MM 000000
00104 37* C 000000
00105 38* IF(IMPL.GT.0)GO TO 100 000000
00105 39* C 000000
00107 40* TMAX1=TMAX+.99999 000002
00107 41* C 000002
00110 42* C1= 3.1441E-6 000005
00110 43* C 000005
00111 44* IF(EFF.EQ..99999)EFD=0.9 000007
00113 45* IF(MD.EQ..99999)MD=2.0E5 000014
00115 46* IF(MP1.EQ..99999)MP1=1.E0 000021
00117 47* IF(MM.EQ..99999)MM=3.0E5 000026
00121 48* IF(CK.EQ..99999)CK=0.011 000033
00123 49* IF(FO.EQ..99999)FO=0.5 000040
00125 50* IF(Y.EQ..99999)Y=0.25 000045
00127 51* CC=CK*(MD+.81+.21*FO*MM)*Y 000052
00127 52* C 000052
00133 53* M2U=C.C 000071
00131 54* 100 EFF=1.0 000073
00132 55* M=C.0 000074
00133 56* IF(MP1.LC.0.01 GO TO 200 000075
00133 57* C 000075
00133 58* C SOLVE CUBIC EQUATION FOR M AND EFF 000075
00133 59* C 000075
00135 60* A3=-P1/(C1*MM1) 000100
00136 61* A4=-A3*(1.0-EFD)*SQRT(MD) 000105
00136 62* C 000105
00137 63* CALL CUSTC(A3,A4,ANS) 000115
00140 64* IF(ANS.LC.0.1 GO TO 200 000122
00140 65* C 000122
00142 66* M=ANS**2 000125
00143 67* EFF=1.0-(1.0-EFD)*SQRT(MD)/ANS 000130
00143 68* C 000130
00143 69* C PRODUCT EFFICIENCY AND CHARGE RATE 000130
00143 70* C 000130
00144 71* 200 EF2=EF1*EFF 000142
00145 72* MP2=MM*(MP1*EFF,MM*MM)*C1) 000144
00145 73* C 000144
00146 74* IF(IMPL.LE.1)RETURN 000156
00146 75* C 000156
00146 76* C STATISTICS 000156
00146 77* C 000156
00150 78* M2U=AMAX1(M2U,M) 000165
00150 79* C 000165
00151 80* IF(TIME.LT.TMAX)RETURN 000173

```

PU

272

00151	01*	C	
00153	02*		CCI=CCI+CC
00153	03*	C	
00154	04*		RETURN
00155	05*		END

000173
000202
000202
000205
000312

BCS 40180-2

PL'

7.31 AC-DC RECTIFIER

This component models a solid-state rectifier/transformer. Power losses due to resistive heating and contact potential loss are modeled. Default parameter values determining power losses are based on 200 kw rated power.

FIGURE 7.31: RECTIFIER FUNCTIONAL DIAGRAM

Inputs

<u>Parameter/Port</u>		<u>Description</u>	<u>Units</u>
P	1	AC input power	kw
RT		Transformer resistance (D = 0)	ohms
XT		Transformer reactance (D = 0.03)	ohms
VAC		Rated AC voltage (D = 440)	volts
DR		Rectifier contact potential (D = 0)	volts
RR		Rectifier resistance (D = 0.02)	ohms
RAP		Rated input power	kw
EF	1	Input product efficiency	-
MP	1	Maximum input power (D = 1×10^8)	kw
CC		Rectifier cost/year	\$

Outputs

<u>Variable/Port</u>			
P	2	DC output power	kw
IAC		AC input current	amps
PL		Power loss	kw
EF	2	Output product efficiency	-
MP	2	Maximum output power	kw

D - Default values supplied.

Calculation Sequence

- 1) Compute transformer power angles

$$Y = \sin(\theta) = \sqrt{3} * X_T * P_1 * 1000 / VAC^2$$

$$ABS(Y) > 1 \Rightarrow \text{DIAGNOSTIC}$$

- 2) Input and output current

If $P_1 \leq 0$ set $P_2 = IAC = PL = 0.$, $EFF = 1$ and go to 4)

$$IAC = VAC \sqrt{2-2\cos(\theta)} / (\sqrt{3} * X_T)$$

$$" = VAC \sqrt{2-2 * \sqrt{1-Y^2}} / (\sqrt{3} * X_T)$$

$$IDC = \pi * IAC / \sqrt{6}$$

- 3) Power loss and output power

$$PL = (\sqrt{3} * R_T * IAC^2 + IDC * (DR + IDC * RR)) / 1000$$

$$P_2 = P_1 - PL$$

$$EFF = P_2 / P_1$$

$$P_2 \leq 0 \Rightarrow \text{DIAGNOSTIC, } EFF = 1$$

- 4) Efficiency and maximum power

$$EF2 = EF1 * EFF$$

$$MP2 = \min(MP1, RAP) * EFF$$

- 5) Compute Costs

SUBROUTINE RE ENTRY POINT 000257

STORAGE USED CODE(1) 000352; DATA(0) 000073; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 CIMPL 000002
0004 CTIME 000001
0005 CSIMUL 000010
0006 COST 000001

EXTERNAL REFERENCES (BLOCK, NAME)

0007 MWDUS
0010 NI028
0011 SQR7
0012 NEPR35

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000042 100L	0000 000007 108F	0001 000100 200L	0001 000117 300L	0000 000034 308F
0001 000214 400L	0006 R 000000 CCI	0005 000000 DUM	0000 R 000006 EFF	0001 000001 ICNT
0000 R 000000 J0C	0003 1 000000 IMPL	0000 000061 INJP8	0000 R 000001 PI	0000 R 000002 ROOT3
0004 R 000000 TIME	0005 R 000007 TMAX	0000 R 000003 TMAX1	0000 R 000004 Y	0000 R 000005 YF

00100	1*	C	CRE	000000
00101	2*	C	SUBROUTINE RE(P2,IAC,PL,EF2,MP2,P1,RT,XT,VAC,DR,RR,RAP,EF1,MP1,CC)	000000
00101	3*	C		000000
00101	4*	C	PURPOSE SOLID STATE RECTIFIER/TRANSFORMER MODEL	000000
00101	5*	C		000000
00101	6*	C	METHOD COMPUTE OUTPUT DC POWER AS A FUNCTION	000000
00101	7*	C	OF INPUT AC POWER	000000
00101	8*	C		000000
00101	9*	C	WRITTEN BY Y.K.CHAN VERSION 1, JUNE 1, 1977	000000
00101	10*	C		000000
00101	11*	C	CALL SEQUENCE	000000
00101	12*	C	OUTPUTS	000000
00101	13*	C	P2 -DC OUTPUT POWER, KW	000000
00101	14*	C	IAC -AC INPUT CURRENT, AMPS	000000
00101	15*	C	PL -POWER LOSS, KW	000000
00101	16*	C	EF2 -OUTPUT PRODUCT EFFICIENCY	000000
00101	17*	C	MP2 -MAXIMUM OUTPUT POWER, KW	000000
00101	18*	C	INPUTS	000000
00101	19*	C	P1 -AC INPUT POWER, KW	000000
00101	20*	C	RT -TRANSFORMER RESISTANCE, OHMS	000000
00101	21*	C	XT -TRANSFORMER REACTANCE, OHMS	000000
00101	22*	C	VAC -RATED AC VOLTAGE, VOLTS	000000

RL

```

00101 23* C      RR -RECTIFIER CONTACT POTENTIAL, VOLTS      000000
00101 24* C      RR -RECTIFIER RESISTANCE, OHMS           000000
00101 25* C      RAP -RATED INPUT POWER, MW              000000
00101 26* C      EFF -INPUT PRODUCT EFFICIENCY            000000
00101 27* C      MP1 -MAXIMUM INPUT POWER, KW             000000
00101 28* C      CC -RECTIFIER COST/YEAR, $               000000
00101 29* C
00103 30*      COMMON /CIMPL/IMPL,ICNT/CTIME/TIME/CSINUL/DUNIT/,TMAX/COST/CC3 000000
00104 31*      REAL IAC,MP2,MP1,IOC                        000000
00105 32*      DATA PI/3.14159/                          000000
00107 33*      DATA ROOT3/1.73205/                      000000
00107 34* C
00111 35*      IF(IMPL.GT.0) GO TO 100                      000000
00113 36*      IF(MP1.EQ..99999)MP1=1.E0                  000002
00115 37*      IF(RT.EQ..99999) RT=0.                    000007
00117 38*      IF(XT.EQ..99999) XT=0.3                    000013
00121 39*      IF(VAC.EQ..99999) VAC=440.                000020
00123 40*      IF(DR.EQ..99999) DR=0.                    000025
00125 41*      IF(RR.EQ..99999) RR=0.2                  000031
00127 42*      TMAX1=TMAX+.99999                          000036
00127 43* C
00127 44* C      COMPUTE TRANSFORMER POWER ANGLES      000036
00127 45* C
00130 46*      100 Y=ROOT3*X1*PI+1000./(VAC+VAC)          000042
00131 47*      YY=Y*Y                                        000051
00132 48*      IF(YY.LE.1.160 TO 200                      000053
00134 49*      IF(IMPL.EQ.2)WRITE(6,100)PI,XT,VAC        000056
00142 50*      100 FORMAT(1H0,19HRC, AC INPUT POWER ,F12.3,49H 100 LARGE IN RELATION 000071
00142 51*      110 TRANSFORMER REACTANCE ,F12.3,22H AND RATED AC VOLTAGE ,F12.3) 000071
00143 52*      IF(IMPL.(0.2)ICNT=ICNT+1                  000071
00145 53*      200 YU=AMIN(1.-YY)                          000100
00145 54* C      INPUT AND OUTPUT CURRENT              000100
00145 55* C
00145 56* C
00146 57*      IF(PI.GT.0.160 TO 300                      000105
00150 58*      P2=0.                                        000110
00151 59*      IAC=0.                                       000111
00152 60*      PL=0.                                        000112
00153 61*      EFF=1.                                       000113
00154 62*      GO TO 400                                    000115
00154 63* C
00155 64*      300 IAC=VAC*SQRT(2.-2.*SQRT(1.-YY))/(ROOT3*XT) 000115
00156 65*      IOC=PI*IAC/SQRT(6.)                        000117
00156 66* C
00156 67* C      POWER LOSS AND OUTPUT POWER          000140
00156 68* C
00157 69*      PL=1000*RT*IAC*IAC+IOC*IOC*(DR+IOC*RR)/1000. 000140
00160 70*      P2=P1-PL                                       000147
00161 71*      EFF=P2/P1                                       000147
00162 72*      IF(P2.GT.0.) GO TO 400                      000147
00164 73*      IF(IMPL.EQ.2)WRITE(6,300)PL,P3              000147
00171 74*      300 FORMAT(1H0,19HRE POWER LOSS ,F12.3,21H EXCEEDS INPUT POWER ,F12.3) 000170
00172 75*      IF(IMPL.EQ.2)ICNT=ICNT+1                  000202
00174 76*      P2=0.                                        000210
00175 77*      EFF=1.                                       000211
00175 78* C
00175 79* C      EFFICIENCY AND MAXIMUM POWER          000211

```

RE

```

00175      80*      C
00176      81*      400 EF2=CF1*EFF
00177      82*      NP2=AFIN1(NP1,RAP)
00200      83*      NP2=NP2*EFF
00201      84*      IF(IMPL.LE.1)RETURN
00203      85*      IF(TIME.LT.TMAX)RETURN
00205      86*      CCI=CCI+CC
00206      87*      C
00207      88*      RETURN
          89*      END

```

```

000211
000214
000216
000224
000226
000235
000244
000244
000247
000351

```

7.32 RANDOM NUMBERS

This component generates an uncorrelated sequence of normally distributed random numbers with a specified mean and standard deviation.

Inputs

<u>Parameter/Port</u>	<u>Description</u>
MN	Mean value of sequence
SIG	Standard deviation of sequence
NST ¹	Start parameter. (Use any odd integer greater than 1). Default supplied.

Outputs

<u>Variable/Port</u>	
F0	Random number output

¹ If RESET parameter > 0 then succeeding simulations use NST to start random sequence.

SUBROUTINE RN ENTRY POINT 000065

STORAGE USED CODE(1) 000105; DATA(1) 000033; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 C1MPL 000003

EXTERNAL REFERENCES (BLOCK, NAME)

0004 NEPR35

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001	000026	1236	0001	000021	SL	0000	R	000006	AX0	0000	I	000007	I	0003	000001	ICNT		
0003	I	000000	IMPL	0005	000024	INJPS	0003	I	000002	ITEST	0000	D	000004	SUM	0000	D	000000	K
0000	D	000002	Y															

00100	1*	CRN		000000
00101	2*		SUBROUTINE RN(U,AX,SIG,AMN)	000000
00101	3*	C	VERSION 2.	000000
00101	4*	C	REVISD MAY 1977	000000
00101	5*	C	PURPOSE - GENERATES A NORMALLY DISTRIBUTED RANDOM NUMBER	000000
00101	6*	C	CALL SEQUENCE	000000
00101	7*	C	U - RANDOM NUMBER OUTPUT	000000
00101	8*	C	AX - A START PARAMETER WHICH CONTROLS THE BEGINNING POINT	000000
00101	9*	C	OF THE OUTPUT SEQUENCE. AX SHOULD BE ANY ODD INTEGER	000000
00101	10*	C	GREATER THAN ONE. THE DEFAULT VALUE OF AX IS 431469.	000000
00101	11*	C	AX IS UPDATED FOR NEW CALLS TO THE SUBROUTINE.	000000
00101	12*	C	SIG - THE DESIRED STANDARD DEVIATION OF THE SEQUENCE	000000
00101	13*	C	AMN - THE DESIRED MEAN OF THE SEQUENCE	000000
00101	14*	C	DESIGNED BY ROGER W. CALL	000000
00101	15*	C	SEPT 1976	000000
00103	16*		COMMON /C1MPL/IMPL,ICNT,ITEST	000000
00104	17*		DOUBLE PRECISION X,Y,SUM	000000
00105	18*		DATA Y /253967.00/ AX0/D./	000000
00110	19*		IF IMPL.GT.C160 TO 5	000000
00112	20*		IF (AX.EQ..999999) AX=431469.	000002
00114	21*		IF (AX0.EQ.0.) AX0= AX	000007
00116	22*		IF (ITEST.EQ.1) AX= AX0	000013
00120	23*	5	X = AX	000021
00121	24*		SUM=D.D0	000022
00122	25*		DO 1 I=1,12	000026
00125	26*		X= C1MPL*X+Y,16777216.D01	000026
00126	27*	1	SUM= SUM+ X/16777215.D0	000036
00130	28*		AX= X	000043
00131	29*		U=(SUM-6.0)*SIG+AMN	000045
00132	30*		RETURN	000054
00133	31*		END	000104

RN

7.33 SATURATION FUNCTION

Inputs

Parameter/Port	Description
FIN	Input quantity
C1	Slope $0 < FIN < C3$
C2	Slope $FIN > C3$
C3	Positive saturation Intercept
C4	Slope $0 > FIN > C6$
C5	Slope $FIN < C6$
C6	Negative saturation Intercept

Outputs

Variable/Port	Description
F0	Output quantity

Calculation Sequence

$$\begin{aligned}
 F0 &= C1 * C3 + C2 * (FIN - C3) && \text{if } FIN > C3 \\
 F0 &= C1 * FIN && \text{if } 0 < FIN < C3 \\
 F0 &= C4 * FIN && \text{if } 0 > FIN > C6 \\
 F0 &= C4 * C6 + C5 * (FIN - C6) && \text{if } FIN < C6
 \end{aligned}$$

SUBROUTINE SA ENTRY POINT 000051

STORAGE USED CODE(1) 000076; DATA(1) 000013; BLANK COMMON(2) 000000

EXTERNAL REFERENCES (BLOCK, NAME)

0003 MERRSS

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000017 10L 0001 000042 130L 0001 000027 20L 0001 000037 30L 0100 000000 INJPS

```

00100      14      CSA
00101      24      SUBROUTINE SA(F0,FIN,C1,C2,C3,C4,C5,C6)
00101      34      C
00101      44      C PURPOSE - TO SIMULATE SATURATION
00101      54      C
00101      64      C
00101      74      C METHOD - SEE CODING. C3 AND C6 ARE VALUES OF THE INPUT AT WHICH
00101      84      C SATURATION OCCURS. C3 IS GREATER THAN C6. THE ROUTINE
00101      94      C CAN SIMULATE A CHANGE OF SLOPE AT THE ORIGIN (C3,MC,C4)
00101     104      C PROVIDED C6 IS LESS THAN ZERO. SIMILARLY THE SLOPES
00101     114      C IN THE SATURATION REGION (C2 AND C5) CAN DIFFER.
00101     124      C THE SLOPES CAN BE POSITIVE OR NEGATIVE
00101     134      C
00101     144      C
00101     154      C WRITTEN BY - ADAM LLOYD LATEST REVISION - NOV 75
00101     164      C
00101     174      C LIMITATIONS - USE OF ZERO SLOPES (C2=0 OR C5=0) IN THE SATURATION
00101     184      C REGION SHOULD BE AVOIDED. IT IS DESIRABLE THAT THE
00101     194      C SLOPE RATIOS C1/C2 AND C4/C5 SHOULD NOT EXCEED 100.
00101     204      C EXCESSIVE SLOPE RATIOS MAY RESULT IN VERY SLOW
00101     214      C CONVERGENCE
00101     224      C
00101     234      C
00101     244      C INPUT/OUTPUT LIST
00101     254      C
00101     264      C F0 OUTPUT VARIABLE ANY OUTPUT VAR
00101     274      C FIN INPUT VARIABLE ANY INPUT VAR
00101     284      C C1 SLOPE ) FIRST ANY INPUT PARAM
00101     294      C C2 SATURATION SLOPE ) SLOPE ANY INPUT PARAM
00101     304      C C3 SATURATION INTERCEPT ) ANY INPUT PARAM
00101     314      C C4 SLOPE ) SECOND ANY INPUT PARAM
00101     324      C C5 SATURATION SLOPE ) SLOPE ANY INPUT PARAM
00101     334      C C6 SATURATION INTERCEPT ) ANY INPUT PARAM
00101     344      C
00101     354      IF (FIN.GT.C3) GO TO 10

```

51

```

00105 36*      IF(FIN.LT.C6)GO TO 20
00107 37*      IF(FIN.LT.0.150)GO TO 30
00111 38*      FO=C1*FIN
00112 39*      GO TO 100
00112 40*      C POSITIVE SATURATION
00113 41*      10 FO=C1+C3+C2*(FIN-C3)
00114 42*      GO TO 100
00114 43*      C NEGATIVE SATURATION
00115 44*      20 FO=C4+C6+C5*(FIN-C6)
00116 45*      GO TO 100
00116 46*      C NEGATIVE UNSATURATED
00117 47*      30 FO=C4*FIN
00120 48*      100 RETURN
00121 49*      END

```

```

000001
000007
000012
000015
000015
000017
000025
000025
000027
000035
000035
000037
000042
000075

```

SA

7.34 SINGLE POLE SWITCH

THE SWITCHING OPERATION MAY BE CONTROLLED BY EITHER TIME OR THE INPUT PARAMETER SW1. THE TIME DEPENDENCE MAY BE ELIMINATED BY SETTING $TC1 = 10^{36}$

Inputs

Parameter/Port	Description
VA1	Input to switch
VB1	Input to switch
SW1	Switch control parameter
TC1	Time for first switching (hours)
TC2	Time for second switching (hours)

Outputs

Variable/Port	
V01	Switch output

Calculation Sequence

If $SW1 = 0$ then

$$V01 = \begin{cases} VA1 & TC1 < TIME < TC2 \\ VB1 & \text{otherwise} \end{cases}$$

If $SW1 = 1$ then

$$V01 = \begin{cases} VB1 & TC1 < TIME < TC2 \\ VA1 & \text{otherwise} \end{cases}$$

SUBROUTINE SW ENTRY POINT 000042

STORAGE USED CODE(1) 000047; DATA(0) 000007; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 CTIME 000003
0004 CIO 000003

EXTERNAL REFERENCES (BLOCK, NAME)

0005 NEPRIS

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0004 000002 IDIAS 0000 000003 INJPS 0004 000000 IREAD 0004 000001 INRITE 0000 R 000005 SX
0001 P 000000 TIME

00100	1*	CSW					000000
00101	2*		SUBROUTINE SW(V01,VA1,V01,SW1,TC1,TC2)				000000
00101	3*	C					000000
00101	4*	C	PURPOSE - TO PROVIDE SWITCH CONTROL FOR ONE VARIABLE				000000
00101	5*	C					000000
00101	6*	C					000000
00101	7*	C	METHOD - SEE CODING				000000
00101	8*	C					000000
00101	9*	C					000000
00101	10*	C	WRITTEN BY - ADAM LLOYD	LATEST REVISION	NOV 75		000000
00101	11*	C					000000
00101	12*	C					000000
00101	13*	C	LIMITATIONS - NOT MORE THAN TWO SWITCHINGS AT TIMES TC1 AND TC2				000000
00101	14*	C					000000
00101	15*	C					000000
00101	16*	C	INPUT/OUTPUT LIST				000000
00101	17*	C					000000
00101	18*	C	V01	OUTPUT VARIABLE NO 1	ANY	OUTPUT VAR	000000
00101	19*	C	VA1	INPUT VARIABLE NO A1	ANY	INPUT VAR	000000
00101	20*	C	V01	INPUT VARIABLE NO P1	ANY	INPUT VAR	000000
00101	21*	C	SW1	SWITCH CONTROL INITIAL VALUE	---	INPUT PARAM	000000
00101	22*	C		F1. VO=VR			000000
00101	23*	C		F2. VO=VA			000000
00101	24*	C	TC1	TIME FOR FIRST SWITCH	SECS	INPUT PARAM	000000
00101	25*	C	TC2	TIME FOR SECOND SWITCH	SFCS	INPUT PARAM	000000
00101	26*	C		(TC2,67,TC1)			000000
00103	27*		COMMON/CTIME/TIME				000000
00104	28*		COMMON/CIO/IREAD,INRITE,IDIAS				000000
00105	29*		SX=SW1				000000

MS

```

00106      35*      IF (TIME.GT.TC1.AND.TIME.LT.TC2) SX=ABS(SW)-1.1
00110      31*      VOI=VA1
00111      32*      IF (SX.GT.0.5) VO1=V01
00113      33*      RETURN
00114      34*      END

```

```

C00001
C00023
000025
000033
C00046

```

7.35 TWO POLE SWITCH

SEE SW FOR SWITCH CONTROL LOGIC

Inputs

<u>Parameter/Port</u>	<u>Description</u>
VA1	Input to switch 1
VA2	Input to switch 2
VB1	Input to switch 1
VB2	Input to switch 2
SW1	Switch control parameter
TC1	Time for first switching (hours)
TC2	Time for second switching (hours)

Outputs

<u>Variable/Port</u>	
V01	Output from switch 1
V02	Output from switch 2

SUBROUTINE SX ENTRY POINT 000052

STORAGE USED CODE(1) 000062; DATA(0) 000007; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 CTIME 000001
0004 CIO 000003

EXTERNAL REFERENCES (BLOCK, NAME)

0005 NEPRIS

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0004 000002 IDIAG 0000 000003 INJPS 0004 000000 INCRD 0004 000001 INRITE 0000 R 000000 SV
0003 R 000000 TIME

00100	1*	CSX						000000
00101	2*		SUBROUTINE SX(V01,V02,V01,V02,V01,V02,SW1,TC1,TC2)					000000
00101	3*	C						000000
00101	4*	C	PURPOSE - TO PROVIDE A SWITCH COMPONENT FOR TWO VARIABLES					000000
00101	5*	C						000000
00101	6*	C						000000
00101	7*	C	METHOD - SEE CODING					000000
00101	8*	C						000000
00101	9*	C						000000
00101	10*	C	WRITTEN BY - ADAM LLOYD	LATEST REVISION	NOV 75			000000
00101	11*	C						000000
00101	12*	C						000000
00101	13*	C	LIMITATIONS - NOT MORE THAN TWO SWITCHINGS AT TIMES TC1 AND TC2					000000
00101	14*	C						000000
00101	15*	C						000000
00101	16*	C	INPUT/OUTPUT LIST					000000
00101	17*	C						000000
00101	18*	C	V01 OUTPUT VARIABLE NO 1	ANY	OUTPUT	VAR		000000
00101	19*	C	V02 OUTPUT VARIABLE NO 2	ANY	OUTPUT	VAR		000000
00101	20*	C	V01 INPUT VARIABLE NO A1	ANY	INPUT	VAR		000000
00101	21*	C	V02 INPUT VARIABLE NO A2	ANY	INPUT	VAR		000000
00101	22*	C	V01 INPUT VARIABLE NO B1	ANY	INPUT	VAR		000000
00101	23*	C	V02 INPUT VARIABLE NO B2	ANY	INPUT	VAR		000000
00101	24*	C	SW1 SWITCH CONTROL INITIAL VALUE	---	INPUT	PARAM		000000
00101	25*	C	1. V0=VB					000000
00101	26*	C	2. V0=VA					000000
00101	27*	C	TC1 TIME FOR FIRST SWITCH	SECS	INPUT	PARAM		000000
00101	28*	C	TC2 TIME FOR SECOND SWITCH	SECS	INPUT	PARAM		000000
00101	29*	C	(TC2,GT,TC1)					000000

SX

00103	30*	COMMON/CTIME/TIME	C00000
00104	31*	COMMON/CTO/ITCAD,ITWRITE,IDIAB	C00000
00105	32*	SM=SM1	C00000
00106	33*	IF (TIME.GT.TC1.AND.TIME.LT.TC2) SM=ABS(SM1-1.)	C00001
00110	34*	V01=VA1	C00023
00111	35*	V02=VA2	C00025
00112	36*	IF (SM.GT.0.5) V01=VB1	C00027
00114	37*	IF (SM.GT.0.5) V02=VB2	C00035
00116	38*	RETURN	C00043
00117	39*	END	C00061

SX

7.36 THREE POLE SWITCH

SEE SW FOR SWITCH CONTROL LOGIC

InputsParameter/PortDescription

VA1	Input to switch 1
VA2	Input to switch 2
VA3	Input to switch 3
VB1	Input to switch 1
VB2	Input to switch 2
VB3	Input to switch 3
SW1	Switch control parameter
TC1	Time for first switching (hours)
TC2	Time for second switching (hours)

OutputsVariable/Port

V01	Output from switch 1
V02	Output from switch 2
V03	Output from switch 3

SY

					SECS	INPUT	PARAM	
00101	30*	C	TC1	TIME FOR FIRST SWITCH	SECS	INPUT	PARAM	000000
00101	31*	C	TC2	TIME FOR SECOND SWITCH	SECS	INPUT	PARAM	000000
00101	32*	C		(TC2.GT.TC1)				000000
00103	33*			COMMON/CTIME/TIME				000000
00104	34*			COMMON/CIO/IREAD,IWRITE,IDIAS				000000
00105	35*			SW=SW1				000000
00106	36*			V01=VA1				000001
00107	37*			V02=VA2				000003
00110	38*			V03=VA3				000005
00111	39*			IF (TIME.GT.TC1).AND.(TIME.LT.TC2) SW=ABS(SW1-1.)				000007
00113	40*			IF (SW.GT.0.5) V01=V01				000031
00115	41*			IF (SW.GT.0.5) V02=V02				000037
00117	42*			IF (SW.GT.0.5) V03=V03				000045
00121	43*			RETURN				000053
00122	44*			END				000076

7.37 FOUR POLE SWITCH

SEE SW FOR SWITCH CONTROL LOGIC

InputsParameter/PortDescription

VA1	Input to switch 1
VA2	Input to switch 2
VA3	Input to switch 3
VA4	Input to switch 4
VB1	Input to switch 1
VB2	Input to switch 2
VB3	Input to switch 3
VB4	Input to switch 4
SW1	Switch control parameter
TC1	Time for first switching (hours)
TC2	Time for second switching (hours)

OutputsVariable/Port

V01	Output from switch 1
V02	Output from switch 2
V03	Output from switch 3
V04	Output from switch 4

SUBROUTINE SZ ENTRY POINT 000073

STORAGE USED CLOC(1) 000112; DATA(1) 000011; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 CTIME 000001
0004 CIO 000003

EXTERNAL REFERENCES (BLOCK, NAME)

0005 NERR38

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0004 000002 INIAG 0000 000003 INJPS 0004 000000 IREAD 0004 000001 IWRITE 0000 N 000000 SM
0003 N 000000 TIME

00100	1*	CSZ				000000
00101	2*		SUBROUTINE SZ(V01,V02,V03,V04,VA1,VA2,VA3,VA4,V01,V02,V03,V04,			000000
00101	3*		1 SM1,TC1,TC2)			000000
00101	4*	C				000000
00101	5*	C	PURPOSE - TO PROVIDE A SWITCH COMPONENT FOR FOUR VARIABLES			000000
00101	6*	C				000000
00101	7*	C				000000
00101	8*	C	METHOD - SEE CODING			000000
00101	9*	C				000000
00101	10*	C				000000
00101	11*	C	WRITTEN BY - ADAM LLOYD	LATEST REVISION	NOV 75	000000
00101	12*	C				000000
00101	13*	C				000000
00101	14*	C	LIMITATIONS - NOT MORE THAN TWO SWITCHINGS AT TIMES TC1 AND TC2			000000
00101	15*	C				000000
00101	16*	C				000000
00101	17*	C	INPUT/OUTPUT LIST			000000
00101	18*	C				000000
00101	19*	C	V01	OUTPUT VARIABLE NO 1	ANY	OUTPUT VAR
00101	20*	C	V02	OUTPUT VARIABLE NO 2	ANY	OUTPUT VAR
00101	21*	C	V03	OUTPUT VARIABLE NO 3	ANY	OUTPUT VAR
00101	22*	C	V04	OUTPUT VARIABLE NO 4	ANY	OUTPUT VAR
00101	23*	C	VA1	INPUT VARIABLE NO A1	ANY	INPUT VAR
00101	24*	C	VA2	INPUT VARIABLE NO A2	ANY	INPUT VAR
00101	25*	C	VA3	INPUT VARIABLE NO A3	ANY	INPUT VAR
00101	26*	C	VA4	INPUT VARIABLE NO A4	ANY	INPUT VAR
00101	27*	C	VB1	INPUT VARIABLE NO B1	ANY	INPUT VAR
00101	28*	C	VB2	INPUT VARIABLE NO B2	ANY	INPUT VAR
00101	29*	C	VB3	INPUT VARIABLE NO B3	ANY	INPUT VAR

00101	30*	C	VB*	INPUT VARIABLE NO B*	ANY	INPUT	VAR	000000
00101	31*	C	SW1	SWITCH CONTROL INITIAL VALUE	---	INPUT	PARAM	000000
00101	32*	C		=1. VO=VB				000000
00101	33*	C		=0. VO=VA				000000
00101	34*	C	TC1	TIME FOR FIRST SWITCH	SECS	INPUT	PARAM	000000
00101	35*	C	TC2	TIME FOR SECOND SWITCH	SECS	INPUT	PARAM	000000
00101	36*	C		(TC2.GT.TC1)				000000
00103	37*			COMMON/CTIME/TIME				000000
00104	38*			COMMON/C10/IRAD,IRITE,JD)AB				000000
00105	39*			SW=SW1				000000
00106	40*			IF(TIME.GT.TC1.AND.TIME.LT.TC2)SW=ABS(SW1-1.)				000001
00110	41*			VO1=VA1				000021
00111	42*			VO2=VA2				000025
00112	43*			VO3=VA3				000027
00113	44*			VO4=VA4				000031
00114	45*			IF(SW.GT.0.5)VO1=VB1				000033
00116	46*			IF(SW.GT.0.5)VO2=VB2				000041
00120	47*			IF(SW.GT.0.5)VO3=VB3				000047
00122	48*			IF(SW.GT.0.5)VO4=VB4				000055
00124	49*			RETURN				000063
00125	50*			END				000111

7.38 TAPE/FILE READ

This component reads a data file containing a single output variable time history. The file structure is specified below, and assumes equal increment data. Linear interpolation is used to obtain the output value. No more than eight TA components are allowed per model. The component TI is used to supply the time input T. The data files must be catalogued using the names F1, F2, or the JCL procedure file XQTANALYSIS modified appropriately. If tape data is used, it must be read into permanent storage files in a prior job step.

Inputs

<u>Parameter/Port</u>	<u>Description</u>	<u>Units</u>
NST	Number of records to skip at start ¹	
T	Simulation time referenced to start of year	hr
ITF	Indicator function: 0 = no read J = read data into Jth array ($J \leq 8$)	

¹ NST = (start time of simulation - start time of file) / (data increment x 446)

Outputs

<u>Variable/Port</u>	<u>Description</u>	<u>Units</u>
VAR	Output variable at time T	
N	Length of file record	
T0	Time of first data value in current record	hr
H	Time increment between data values	hr

Record Structure

1st record = identification header, 28 words

Nth record (N>1): 448 words

first word = time of first data value in hours (T0)

second word = time increment between data values (H)

J+2 word = data value at time = $T0 + (J-1)H$

(J=1,...,446)

If the useful data ends in the middle of a record, then an end value = .99999 is used to signal end of information.

Calculation Sequence

If ITF = 0 RETURN

1) Initialization. (First Pass Only)

o Read 1st record and write out identification header data and unit number. (Error exit to 6))

o Read past NST data records. Go to 4)

Calculation Sequence Cont.

2) File location incorrect; use initial value

```
o If  $T \geq T_0$  GO TO 3)
  Write diagnostic
  VAR = A(1,J) (J=1TF)
  Return
```

3) Table Interpolation for Output

```
o If  $T \geq T_0 + N * H$  GO TO 4)
  XT(1) = T0
  XT(2) = T0+H
  VAR = TBLU1(T,XT,A(1,J),O,N)
  Return
```

4) Read Next Data Record

```
If end has been encountered previously go to 5)
Read next record into array A(*,J). If end encountered, set
N = last value. (Error exit to 6))
  T0 = A(1,J)
  H = A(2,J)
  GO TO 3)
```

5) End of File. Use last value.

```
VAR = A(N,J)
Write Diagnostic
Return
```

6) Read Error Encountered

```
Write Diagnostic and STOP
```

SUBROUTINE TA ENTRY POINT 000393

STORAGE USED CODE(1) 000403; DATA(2) 007132; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 CIMPL 000002
0004 0ATARD 000700

EXTERNAL REFERENCES (BLOCK, NAME)

0005 NTRAN
0006 TALUI
0007 NUNUS
0010 TIO34
0011 N1024
0012 KSTOP4
0013 NEPR34

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000321 10L	0001 000102 100L	0001 000231 1676	0000 007035 20F	0001 000155 200L
0000 007046 30F	0001 000174 300L	0001 000242 400L	0001 000244 60L	0000 007042 80F
0000 007076 90F	0000 R 000000 A	0000 R 006760 8	0000 I 007034 I	0003 1 000001 ICMY
0003 I 000000 IMPL	0000 I 007014 IN0	0000 007121 IN4PS	0000 I 007230 14W	0000 I 007027 J
0000 I 007031 L1	0000 I 007033 N	0000 I 007032 NO	0006 R 000000 TALUI	0004 R 000000 TEMP
0000 R 007026 X	0000 R 007024 XT			

00100	1*	CTAPE		000000
00101	2*		SUBROUTINE TAIVAR, NOUT, TD, M, NST, T, ITF)	000000
00101	3*	C		004000
00101	4*	C	PURPOSE READ TIME HISTORY DATA ON STORAGE DEVICE	000000
00101	5*	C		000000
00101	6*	C		000000
00101	7*	C	METHOD USE NTRAN SEQUENTIAL BLOCK READ FROM MASS STORAGE	000000
00101	8*	C	INTO CORE. LINEAR INTERPOLATION IS USED TO OBTAIN	000000
00101	9*	C	VALUE AT SPECIFIED TIME T.	000000
00101	10*	C		000000
00101	11*	C	WRITTEN BY A.W. WARREN	000000
00101	12*	C	JUNE 1977	000000
00101	13*	C	CALL SEQUENCE	000000
00101	14*	C	OUTPUTS	000000
00101	15*	C	VAR - OUTPUT VARIABLE AT TIME T	000000
00101	16*	C	NOUT - LENGTH OF DATA BLOCK LAST READ	000000
00101	17*	C	TD - TIME OF FIRST DATA VALUE IN CORE, HR	000000
00101	18*	C	N - TIME INCREMENT BETWEEN DATA VALUES, HR	000000
00101	19*	C	INPUTS	000000

TA

00171	77*	IF(TEMP(1+21).EQ.X160 TO 60	000231
00173	78*	50 A(1,J)= TEMP(1+2)	000234
00175	79*	I=I+1	000240
00176	80*	60 NCUT= I-1	000244
00177	81*	IF(I7.LT.447)IND(IJ)=0	000250
00201	82*	60 TO 200	000260
00201	83*	C	000260
00201	84*	C	000260
00201	85*	C	000260
00201	86*	C	000260
00202	87*	400 N= NCUT	000262
00203	88*	VAR=A(1N,J)	000270
00204	89*	IF(1MPL.EQ.2)WRITE(6,80)VAR,IUN	000275
00211	90*	80 FORMAT(1H0,4XTIME POINT PAST TABLE RANGE. LAST VALUE=	000307
00211	91*	1 F12.5,5H ON UNIT,I4)	000307
00212	92*	IF(1MPL.EQ.2)ICNT=ICNT+1	000307
00214	93*	RETURN	000315
00214	94*	C	000315
00214	95*	C	000315
00214	96*	C	000315
00215	97*	10 WRITE(6,90)IUN	000321
00220	98*	90 FORMAT(1H0,33HREAD ERROR OR END OF FILE ON UNIT,I4)	000326
00221	99*	STOP	000326
00222	100*	END A SUBROUTINE TA	000402

7.39 SECOND ORDER TRANSFER FUNCTION

Inputs

<u>Parameter/Port</u>	<u>Description</u>
FIN	Input quantity
Z0	Numerator coefficient
Z1	Numerator coefficient
P0	Denominator coefficient
P1	Denominator coefficient

Outputs

<u>Variable/Port</u>	
X1	Intermediate state
F0	Output quantity (state)

Calculation Sequence

$$\begin{aligned} \dot{X1} &= Z0*FIN - P0*F0 \\ \dot{F0} &= X1 + Z1*FIN - P1*F0 \end{aligned}$$

NOTE: d.c. gain = $\frac{Z0}{P0}$; infinite frequency gain = 0.

SUBROUTINE TF ENTRY POINT D00027

STORAGE USED CODE(1) 000035; DATA(10) 000008; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 C10 000003

EXTERNAL REFERENCES (BLOCK, NAME)

0004 HERR36

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0003 000002 IDIAG 0000 000000 INJPX 0003 000000 IREAD 0003 000001 INRITE

00100	1*	CTF				000000
00101	2*		SUBROUTINE TF(X1,XIDOT,IX1,F0,F0DOT,IFO,FTM,Z0,Z1,PO,PI)			000000
00101	3*	C				000000
00101	4*	C	PURPOSE - TO SIMULATE A SECOND ORDER TRANSFER FUNCTION WITH			000000
00101	5*	C	FIRST ORDER NUMERATOR			000000
00101	6*	C				000000
00101	7*	C				000000
00101	8*	C				000000
00101	9*	C				000000
00101	10*	C				000000
00101	11*	C				000000
00101	12*	C				000000
00101	13*	C				000000
00101	14*	C				000000
00101	15*	C				000000
00101	16*	C				000000
00101	17*	C				000000
00101	18*	C				000000
00101	19*	C				000000
00101	20*	C				000000
00101	21*	C				000000
00101	22*	C				000000
00101	23*	C				000000
00101	24*	C				000000
00101	25*	C				000000
00101	26*	C				000000
00101	27*	C				000000
00101	28*	C				000000
00101	29*	C				000000
00101	30*	C				000000
00101	31*	C				000000

00101	32*	C	PD	DENOMINATOR COEFFICIENT	1/SEC2	INPUT	VAR	000000
00102	33*	C	P1	DENOMINATOR COEFFICIENT	1/SEC	INPUT	VAR	000000
00103	34*			COMMON/CIO/IREAD,IWRITE,IOIAS				000000
00104	35*			IF(IK1.NE.0)X1001=20*FIN-P0*F0				000000
00106	36*			IF(IIF0.NE.0)F0001=X1+71*FIN-P1*F0				000007
00110	37*			RETURN				000020
00111	38*			END				000039

7.40 TIME CONVERSION

Converts simulation running time in hours to time referenced to start of day and start of week, and computes number of days and weeks elapsed since start of year.

Inputs

<u>Parameter/Port</u>	<u>Description</u>	<u>Units</u>
TO	Initial time of simulation from start of year	hrs
TIME	Running time (input via common/CTIME)	hrs

Outputs

<u>Variable/Port</u>		
TW	Time since start of week	hrs
TD	Time since start of day	hrs
WY	Number of weeks	-
DY	Number of days	-
MY	Number of months (approx.)	-
T	Running time from start of year	hrs
DW	Day of week	-

Calculation Sequence

$T = \text{AMOD}(TO + \text{TIME}, 8760)$	$TW = \text{AMOD}(T, 168)$
$WY = T / 168 + 1$	$TD = \text{AMOD}(T, 24)$
$DY = T / 24 + 1$	$DW = TW / 24 + 1$
$MY = T / 730 + 1$	

SUBROUTINE TI ENTRY POINT 000122

STORAGE USED CODE(1) 000155; DATA(1) 000031; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 CTIME 000001

EXTERNAL REFERENCES (BLOCK, NAME)

0004 HERRIS

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0000 000013 IN/PS 0000 I 000002 NY 0000 I 000000 NO 0000 I 000001 NW 0003 * 000000 TIME

00100	1*	CTI		000000
00101	2*		SUBROUTINE TI(T,TD,TV,DW,OY,NY,NM,TOT	000000
00101	3*	C		000000
00101	4*	C	PURPOSE CONVERT SIMULATION TIME TO DAILY, WEEKLY, MONTHLY UNITS	000000
00101	5*	C		000000
00101	6*	C	WRITTEN BY A.W. WARREN	000000
00101	7*	C	VERSION 1, MARCH 3 1977	000000
00101	8*	C		000000
00101	9*	C	CALL SEQUENCE	000000
00101	10*	C	T - SIMULATION TIME FROM START OF YEAR, HR	000000
00101	11*	C	TD - TIME OF DAY, HR	000000
00101	12*	C	TV - TIME SINCE START OF WEEK, HR	000000
00101	13*	C	DW - DAY OF WEEK	000000
00101	14*	C	OY - DAY OF YEAR	000000
00101	15*	C	NY - WEEK OF YEAR	000000
00101	16*	C	NM - MONTH OF YEAR (APPROX.)	000000
00101	17*	C	TOT - SIMULATION INITIAL TIME FROM START OF YEAR, HR	000000
00101	18*	C		000000
00101	19*	C	COMMON / CTIME / TIME	000000
00101	20*	C	T = AMOD(TD + TIME, 8760.1)	000000
00101	21*	C	TD = AMOD(T, 24.1)	000000
00101	22*	C	TV = AMOD(T, 168.1)	000000
00101	23*	C	DW = TV / 24.1 + 1.001	000000
00101	24*	C	OY = DW	000000
00101	25*	C	NY = T / 24.1 + 1.001	000000
00101	26*	C	NM = NY / 168.1 + 1.001	000000
00101	27*	C	TOT = NY	000000
00101	28*	C	NY = T / 730.1 + 1.001	000000
00101	29*	C	NM = NY	000000
00101	30*	C		000000
00101	31*	C	RETURN	000000

11

000154

END

32*

00120

BCS 40180-2

307

7.41 THERMAL LOAD

Thermal load is computed as a user specified function of ambient temperature and time of day. The actual load delivered is either the load requested or the maximum discharge rate of the thermal storage chamber. The value of the thermal energy delivered and % of total load actually delivered are also computed.

Basic Equation

$$RE = TL\theta(TA) * TWT(TD) * NC$$

where

$TL\theta$ = Thermal load versus temperature table

TWT = Daily profile weighting function

NC = Normalizing constant

<u>Tables</u>	<u>Description</u>	<u>Units</u>
TL0	Thermal load versus ambient temperature	kw
TWT	Daily profile weighting function (tabular with time of day)	

Inputs

Parameter/Port

TA	Ambient temperature	°F
LD	Power delivered	kw
TD	Time of day (0-24)	h
VE	Value of thermal energy	\$/kwh
NC	Normalizing constant	

Outputs

Variable/Port

RE	Load request	kw
VDE	Total value of energy delivered (state)	\$

Statistics

PC	Cumulative percent of load delivered	-
SLD	Total energy delivered	kwh
SRE	Total energy requested	kwh

Calculation Sequence

1) Compute load request

$$RE = TLQ(TA)*TWT(TD)*NC$$

2) Value of energy dynamics

$$VDE = LD*VE$$

3) Statistics

$$SLD = SLD+LD*\Delta/2$$

$$SRE = SRE+RE*\Delta/2$$

$$PC = 100.* SLD/SRE$$

where Δ = integration step size

ENTRY POINT 000173

STORAGE USED CODE(1) 000251; DATA(0) 000033; BLANK COMMON(2) 000000

COMMON BLOCKS

```

0003      CTIME  000001
0004      CSIMUL  000010
0005      CTIME  000001
0006      COST   000004

```

EXTERNAL REFERENCES (BLOCK, NAME)

DDOT TELU
ODIO MERRIS

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001	000014	100L	0006	000000	CC	0006	R	000004	CLD	0006	000001	CM	0006	000002	CO				
0006	R	000005	CWE	0506	R	000003	CV	0004	000000	DDW	0003	I	000000	IMPL	0300	000012	INJPS		
0000	I	000000	ITL	0C00	I	000001	ITW	0307	R	000000	TOLU1	0005	R	000000	TINC	0CC4	R	000006	TINC
0000	R	000001	TINC1	0C20	R	000000	TLO	0004	R	000007	TMAX	0000	R	000002	TMAX1	0000	R	000005	TW

```

00100      1*      CTL
00101      2*      SUBROUTINE TLT(TLO,TWT,VDE,DVD,IVD,RE,PC,SLO,SRE,TA,LO,TD,VE,WC1
00101      3*      C
00101      4*      C PURPOSE      COMPUTE ENERGY RESPONSE FROM A THERMAL LOAD REQUEST
00101      5*      C
00101      6*      C METHOD      ENERGY DELIVERED IS EQUAL TO THE LOAD REQUESTED ON
00101      7*      C
00101      8*      C
00101      9*      C THE MAXIMUM DISCHARGE RATE.
00101     10*      C
00101     11*      C WRITTEN BY F. D. HANOHY
00101     12*      C
00101     13*      C CALL SEQUENCE
00101     14*      C
00101     15*      C TABLES
00101     16*      C TLO - THERMAL LOAD AS FUNCTION OF AMBIENT TEMPERATURE
00101     17*      C TWT - DAILY PROFILE WEIGHTING FUNCTION VS TIME OF DAY
00101     18*      C
00101     19*      C
00101     20*      C
00101     21*      C
00101     22*      C
00101     23*      C
00101     24*      C
00101     25*      C
00101     26*      C
00101     27*      C
00101     28*      C
00101     29*      C
00101     30*      C
00101     31*      C
00101     32*      C
00101     33*      C
00101     34*      C
00101     35*      C
00101     36*      C
00101     37*      C
00101     38*      C
00101     39*      C
00101     40*      C
00101     41*      C
00101     42*      C
00101     43*      C
00101     44*      C
00101     45*      C
00101     46*      C
00101     47*      C
00101     48*      C
00101     49*      C
00101     50*      C
00101     51*      C
00101     52*      C
00101     53*      C
00101     54*      C
00101     55*      C
00101     56*      C
00101     57*      C
00101     58*      C
00101     59*      C
00101     60*      C
00101     61*      C
00101     62*      C
00101     63*      C
00101     64*      C
00101     65*      C
00101     66*      C
00101     67*      C
00101     68*      C
00101     69*      C
00101     70*      C
00101     71*      C
00101     72*      C
00101     73*      C
00101     74*      C
00101     75*      C
00101     76*      C
00101     77*      C
00101     78*      C
00101     79*      C
00101     80*      C
00101     81*      C
00101     82*      C
00101     83*      C
00101     84*      C
00101     85*      C
00101     86*      C
00101     87*      C
00101     88*      C
00101     89*      C
00101     90*      C
00101     91*      C
00101     92*      C
00101     93*      C
00101     94*      C
00101     95*      C
00101     96*      C
00101     97*      C
00101     98*      C
00101     99*      C
00101    100*      C
00101    101*      C
00101    102*      C
00101    103*      C
00101    104*      C
00101    105*      C
00101    106*      C
00101    107*      C
00101    108*      C
00101    109*      C
00101    110*      C
00101    111*      C
00101    112*      C
00101    113*      C
00101    114*      C
00101    115*      C
00101    116*      C
00101    117*      C
00101    118*      C
00101    119*      C
00101    120*      C
00101    121*      C
00101    122*      C
00101    123*      C
00101    124*      C
00101    125*      C
00101    126*      C
00101    127*      C
00101    128*      C
00101    129*      C
00101    130*      C
00101    131*      C
00101    132*      C
00101    133*      C
00101    134*      C
00101    135*      C
00101    136*      C
00101    137*      C
00101    138*      C
00101    139*      C
00101    140*      C
00101    141*      C
00101    142*      C
00101    143*      C
00101    144*      C
00101    145*      C
00101    146*      C
00101    147*      C
00101    148*      C
00101    149*      C
00101    150*      C
00101    151*      C
00101    152*      C
00101    153*      C
00101    154*      C
00101    155*      C
00101    156*      C
00101    157*      C
00101    158*      C
00101    159*      C
00101    160*      C
00101    161*      C
00101    162*      C
00101    163*      C
00101    164*      C
00101    165*      C
00101    166*      C
00101    167*      C
00101    168*      C
00101    169*      C
00101    170*      C
00101    171*      C
00101    172*      C
00101    173*      C
00101    174*      C
00101    175*      C
00101    176*      C
00101    177*      C
00101    178*      C
00101    179*      C
00101    180*      C
00101    181*      C
00101    182*      C
00101    183*      C
00101    184*      C
00101    185*      C
00101    186*      C
00101    187*      C
00101    188*      C
00101    189*      C
00101    190*      C
00101    191*      C
00101    192*      C
00101    193*      C
00101    194*      C
00101    195*      C
00101    196*      C
00101    197*      C
00101    198*      C
00101    199*      C
00101    200*      C
00101    201*      C
00101    202*      C
00101    203*      C
00101    204*      C
00101    205*      C
00101    206*      C
00101    207*      C
00101    208*      C
00101    209*      C
00101    210*      C
00101    211*      C
00101    212*      C
00101    213*      C
00101    214*      C
00101    215*      C
00101    216*      C
00101    217*      C
00101    218*      C
00101    219*      C
00101    220*      C
00101    221*      C
00101    222*      C
00101    223*      C
00101    224*      C
00101    225*      C
00101    226*      C
00101    227*      C
00101    228*      C
00101    229*      C
00101    230*      C
00101    231*      C
00101    232*      C
00101    233*      C
00101    234*      C
00101    235*      C
00101    236*      C
00101    237*      C
00101    238*      C
00101    239*      C
00101    240*      C
00101    241*      C
00101    242*      C
00101    243*      C
00101    244*      C
00101    245*      C
00101    246*      C
00101    247*      C
00101    248*      C
00101    249*      C
00101    250*      C
00101    251*      C
00101    252*      C
00101    253*      C
00101    254*      C
00101    255*      C
00101    256*      C
00101    257*      C
00101    258*      C
00101    259*      C
00101    260*      C
00101    261*      C
00101    262*      C
00101    263*      C
00101    264*      C
00101    265*      C
00101    266*      C
00101    267*      C
00101    268*      C
00101    269*      C
00101    270*      C
00101    271*      C
00101    272*      C
00101    273*      C
00101    274*      C
00101    275*      C
00101    276*      C
00101    277*      C
00101    278*      C
00101    279*      C
00101    280*      C
00101    281*      C
00101    282*      C
00101    283*      C
00101    284*      C
00101    285*      C
00101    286*      C
00101    287*      C
00101    288*      C
00101    289*      C
00101    290*      C
00101    291*      C
00101    292*      C
0
```

[illegible]

—

7.42 AMBIENT TEMPERATURE

This component is very similar to the wind component. Ambient temperature is output either from user supplied time histories on storage files or by generating a set of random numbers with user specified random variations. If user supplied profiles are available, then the temperatures are generated from the following equation:

$$TA2 = [PD(TD) + CN(t)] * PY(WY) / MO$$

where PD and PY are the user supplied daily and weekly profiles, TD and WY are the time of the day and week of the year, CN is a colored noise term and MO is the average value of PY:

$$MO = \frac{1}{J} \sum_{j=1}^J PY(j)$$

<u>Tables</u>	<u>Description</u>	<u>Units</u>
PD	Daily profile versus TD	$^{\circ}\text{F}$
PY	Yearly profile versus WY	arbitrary

Inputs

Parameter/Port

TA	1	Ambient temperature data file	$^{\circ}\text{F}$
TD		Time of day	hr
WY		Week of the year	-
CT		Correlation time of colored noise	hr
MN		Mean temperature of colored noise	$^{\circ}\text{F}$
STD		Standard deviation of colored noise	$^{\circ}\text{F}$

Outputs

Variable/Port

CN		Colored noise sample	$^{\circ}\text{F}$
TA	2	Ambient temperature	$^{\circ}\text{F}$
AV		Mean of daily temperature	$^{\circ}\text{F}$
MO		Mean of yearly profile	
TIM		Last time a random sample was generated	hr

Calculation Sequence

- 1) Initialization (first pass only)
Compute AV, MO, and initial CN

$$AV = MN + \frac{1}{N} \sum_{j=1}^N PD(j)$$

- 2) Check for data file input
If TA1 = .99999 go to 3)
TA2 = TA1
Return

- 3) Generate colored noise sample CN

If TIME = TIM RETURN

$$A = \begin{cases} \text{EXP}(-TINC/CT) & CT > 0 \\ 0. & CT = 0 \end{cases}$$

where TINC = integration step size

$$CN = CN * A + W$$

Where W is white noise with mean = MN*(1-A) and

$$\text{standard deviation} = \text{STD} * \sqrt{1-A^2}$$

$$TIM = TIME$$

- 4) Compute Temperature

$$TA2 = (PD(TD) + CN) * PY (WY) / MO$$

```

STORAGE USED  CODE(1) 000350; DATA(0) 000044; BLANK COMMON(2) 000000

```

```
0003      CIPPL      000001
0004      CSIMUL      000007
0005      CTIME       000001
```

0006	RN
0007	TPLU1
0010	EXP
0011	SORT
0012	NEPORS

0001	000121	10L	0001	000127	100L	0001	000057	1176	0001	000102	1276	0001	000247	150L	
0000	R	C00005	A	0000	R	000000	4X	0000	R	000011	DTP	0004	000000	DUM	
0003	I	000000	IMPL	0000	R	000016	IMJPS	0000	I	000004	L	0000	I	000001	ND
0007	R	000000	TRLU1	0005	R	000000	TIME	0004	R	000006	TINC	0000	R	000010	W
0007	R	000000	WSD	0000	R	000012	YIP					0000	R	000006	WNN

[illegible]

一

॥

7.43 VARIABLE RATIO TRANSMISSION

This component models a transmission which couples a fixed speed rotor input (or output) to a variable speed rotor output (or input) component. Power losses are modeled as a table lookup depending on gear ratio and input power.

FIGURE 7.43 TRANSMISSION MODEL - LOOKUP TABLE

<u>Tables</u>	<u>Description</u>	<u>Units</u>
PLØ	Power loss versus input power and gear ratio (TABLE DIMENSION = 66)	kw

Inputs

Parameter/Port

RS	1	Input rotor speed	rpm
RS	2	Output rotor speed	rpm
P	1	Input power	kw
EF	1	Input product efficiency	-
MP	1	Maximum input power	kw
CC		Capital cost/year	\$
CM		Maintenance cost/year	\$

Outputs

Variable/Port

P	2	Output power	kw
TØ		Output torque	ft-lb
PL		Power loss	kw
EF	2	Output product efficiency	-
MP	2	Maximum power output	kw

Calculation Sequence

If $P_1 \leq 0$ or $RS_1 \leq 0$ set $P_2 = T_0 = PL = 0$ and go to 4)

- 1) Determine gear ratio and power terms

$$GR = RS_2/RS_1$$

$$PL = PL_0(P_1, GR)$$

$$P_2 = P_1 - PL$$

- 2) Determine output torque

$$T_0 = P_2 * 737.6 / (RS_2 * (2\pi/60))$$

- 3) Efficiency and maximum power

$$EF_2 = EF_1 * (P_2/P_1)$$

If $P_2 \leq 0$, set $EF_2 = EF_1$ and write Diagnostic

$$MP_2 = MP_1 - PL_0(MP_1, GR)$$

$$MP_2 \leq 0 \Rightarrow \text{DIAGNOSTIC}$$

- 4) Compute Costs

一

[illegible]

一

```

00163 79* 508 FORMAT(ND,25H TRANSMISSION POWER LOSS ,F12.3,
00163 80* 29H EXCEEDS MAXIMUM INPUT POWER ,F12.3)
00164 81* IF(IMPL.EQ.2)ICNT=ICNT+1
00164 82* C
00166 83* 900 IF(IMPL.LE.1)RETURN
00170 84* IF(IME.LT.1)RETURN
00172 85* CCI=CCI+CC
00173 86* CM=CM+CM
00173 87* C
00174 88* RETURN
00175 89* END

```

```

000236
000234
000236
000236
000246
000253
000262
000265
000265
000270
000273

```

7.44 THERMAL STORAGE CHAMBER

The thermal storage chamber is modeled by a "lumped" parameter approach. The entire storage media mass is characterized by a single temperature (no temperature gradient). The storage media is either a sensible heat or a phase change media. Energy is input via electrical resistance heaters and withdrawn by a heat exchanger. Energy is deposited in the media at a rate equal to the available electrical power up to a maximum charging power. The discharge heat exchanger fluid mass flow rate is adjusted to provide the desired heat load demand. The maximum mass flow rate condition determines the maximum thermal load. The maximum energy limit represents the point where the maximum media temperature is reached.

The model initially calculates the required storage media mass to provide the rated thermal energy storage (design point). Cost calculations are also made on the design point conditions. Initial checks on charge and discharge power and initial stored energy level are made. The storage temperature is determined based on the energy level.

Basic Equation

$$E = P - LD - NUXE$$

<u>Tables</u>		<u>Description</u>	<u>Units</u>
HT		Media temperature versus enthalpy in KWH/LB ¹	°F
<u>Inputs</u>			
<u>Parameter/Port</u>			
P		Input power	kw
RE	1	Demand thermal load	kw
NU		Stored energy loss coefficient (D = 0.02)	(h) ⁻¹
TS		Rated storage time ²	h
V0		Rated input voltage ²	v
TM1		Maximum allowable storage temperature (D = 212)	°F
T01		Minimum allowable storage temperature (D = 60)	°F
DH		Design point enthalpy	kwh/lb
PD		Rated storage thermal power ²	kw
PM		Maximum charge rate (D = 2*PD)	kw
MFM		Maximum working fluid mass flow rate (D = 9000)	lb/h
TDE		Temperature deadband for priority resequence (D = 4)	°F
EF	1	Input product efficiency	-
MP	1	Maximum input charging rate (D = 1.X10 ⁸)	kw
CP2		Working fluid heat capacity (D = 2.93X10 ⁻⁴)	kwh/lb-°F
T02		Working fluid return temperature (D = 40)	°F
TM2		Maximum allowable working fluid temperature (D = 212)	°F
R		Effective heat exchanger thermal resistance (D = 3.08X10 ⁻⁴)	°F/kw
CM		Storage device yearly maintenance cost (D = 0.6)	\$/kw
CSA		Storage device capacity cost (D = 50)	\$/kw
CSB		Storage device energy cost (D = 15.2)	\$/kwh
LE		Unit life expectancy	years

D - Default values specified

1 - See Figure 7.44

2 - Design point conditions

Outputs

Variable/Port	Description	Units
E	Stored energy (state)	kwh
I	Input current	amps
MP 2	Maximum discharge rate allowable	kw
INT	Priority interrupt flag	-
T	Storage temperature	°F
M	Required storage media mass	lb
COO	Storage device capital cost/year	\$
RE 2	Maximum charging rate request	kw
MF	Working fluid mass flow rate	lb/h
LD	Power Delivered	kw

Statistics

TSU	Maximum storage temperature	°F
TSL	Minimum storage temperature	°F
ME	Maximum stored energy:	kwh
MFU	Maximum working fluid mass flow rate	lb/h

FIGURE 7.44: TEMPERATURE - ENTHALPY DIAGRAM

The calculation sequence and default values assume a thermal storage device sized to provide 10kw for 24 hours. A paraffin wax phase change storage medium is assumed. Water is assumed as the thermal transport fluid. Costs are assumed to be given by data for the phase change storage device given in Reference 1. The thermal resistance value, R, is assumed equal to that determined for the device of Reference 1. The value for the maximum charging rate, PM, reflects the acceptance of twice the design charge rate. The actual numbers which should be used will depend on specific design and performance requirements obtained from a desired application.

Calculation Sequence

- 1) Media mass, capital cost, maintenance cost (first pass)

$$M = \frac{PD \cdot TS}{OH}$$

$$CC = (CSA + CSB \cdot TS) \cdot PD / LE$$

$$CM = CM \cdot PD$$

- 2) Storage Temperature and Working Fluid Temperature

$$T = HT(E/M)$$

$$TF = \min(TM2, \max[T02, T - RE1 \cdot R])$$

$$E2 = M \cdot HT^{-1}(TM1)$$

1. "Advanced Thermal Energy Storage," BEC/EPRI RP 788-1, July 1976.

Calculation Sequence Cont.

3) Discharge Rate and Thermal Load

$$\begin{aligned} E1 &= M*HT^{-1}(T01) \\ MP2 &= MFM*CP2*(TF-T02) \\ LD &= \min(RE1, MP2, (E-E1)/TINC) \\ MF &= LD/(CP2*(TF-T02)) \end{aligned}$$

4) Diagnostic Checks

$$\begin{aligned} MF &\leq MFM \\ P &\leq PM \\ T01 &\leq T \leq TM1 \end{aligned}$$

5) Current calculations

$$I = \frac{P*1000}{VB}$$

6) Energy dynamics

$$\dot{E} = P - LD - NU*E$$

7) Maximum Charging Rates

$$RE2 = \min(PM, MP1, (E2-E1)/TINC)/EF1$$

where TINC = integration step size

Calculation Sequence Cont.

8) Priority resequencing

if $T \leq T01$, $INT = 1$
if $T \geq T01 + TDE$ and $INT=1$, $INT=0$
if $T \geq TM1$, $INT=-1$
if $T < TM1 - TDE$ and $INT=-1$, $INT=0$

9) Compute Statistics and Costs

5

00101	22*	C	I	- INPUT ELECTRIC CURRENT, KW	000000
00101	23*	C	MPZ	- MAXIMUM DISCHARGE RATE ALLOWABLE, KW	000000
00101	24*	C	INT	- PRIORITY FLAG INTERRUPT	000000
00101	25*	C	T	- STORAGE TEMPERATURE, DEG F	000000
00101	26*	C	M	- REQUIRED STORAGE MEDIA MASS, LB	000000
00101	27*	C	CCO	- STORAGE DEVICE CAPITAL COST/YEAR, \$	000000
00101	28*	C	RE	- MAXIMUM CHARGING RATE REQUEST, KW	000000
00101	29*	C	MF	- WORKING FLUID MASS FLOW RATE, LB/HR	000000
00101	30*	C	LO	- THERMAL LOAD DELIVERED, KW	000000
00101	31*	C	TSU	- MAXIMUM STORAGE TEMPERATURE, DEG F	000000
00101	32*	C	TSL	- MINIMUM STORAGE TEMPERATURE, DEG F	000000
00101	33*	C	ME	- MAXIMUM STORED ENERGY, KWH	000000
00101	34*	C	MFU	- MAXIMUM WORKING FLUID MASS FLOW RATE, LB/HR	000000
00101	35*	C	INPUTS		000000
00101	36*	C	P	- INPUT POWER, KW	000000
00101	37*	C	REI	- THERMAL DISCHARGE REQUEST, KW	000000
00101	38*	C	NU	- STORAGE ENERGY LOSS COEFFICIENT, 1/HR	000000
00101	39*	C	TSO	- RATED STORAGE TIME, HR	000000
00101	40*	C	VO	- RATED INPUT VOLTAGE, VOLTS	000000
00101	41*	C	TSI	- MAXIMUM ALLOWABLE STORAGE TEMPERATURE, DEG F	000000
00101	42*	C	TSI	- MINIMUM ALLOWABLE STORAGE TEMPERATURE, DEG F	000000
00101	43*	C	DN	- DESIGN POINT ENTHALPY, KWH/LB	000000
00101	44*	C	PO	- RATED STORAGE THERMAL POWER, KW	000000
00101	45*	C	PH	- MAXIMUM CHARGE RATE, KW	000000
00101	46*	C	MFH	- MAXIMUM WORKING FLUID MASS FLOW RATE, LB/HR	000000
00101	47*	C	TDE	- TEMPERATURE DEADBAND FOR PRIORITY RESEQUENCE, DEG F	000000
00101	48*	C	EPI	- INPUT PRODUCT EFFICIENCY	000000
00101	49*	C	MP1	- MAXIMUM INPUT CHARGING RATE, KW	000000
00101	50*	C	CP2	- WORKING FLUID HEAT CAPACITY, KWH/LB-F	000000
00101	51*	C	TO2	- WORKING FLUID RETURN TEMPERATURE, DEG F	000000
00101	52*	C	TM2	- MAXIMUM ALLOWABLE WORKING FLUID TEMPERATURE, DEG F	000000
00101	53*	C	R	- EFFECTIVE HEAT EXCHANGER THERMAL RESISTANCE, F/KW	000000
00101	54*	C	CM	- STORAGE DEVICE YEARLY MAINTENANCE COST, \$/KW	000000
00101	55*	C	CSA	- STORAGE DEVICE CAPACITY COST, \$/KW	000000
00101	56*	C	CSB	- STORAGE DEVICE ENERGY COST, \$/KWH	000000
00101	57*	C	LE	- UNIT/LIFE FILE EXPECTANCY, YEARS	000000
00101	58*	C			000000
00103	59*		COMMON/CIPL/IMPL,ICN/CTIME/TIME /CSIMUL/DUMI71,TMAX /COST/CCI,CN1	000000	
00104	60*		PEAL 'U,I,MPZ,INT,MF,LO,ME,MFU,MFH,MP1,LC,M	000000	
00105	61*		DIMENSION HT(1)	000000	
00105	62*	C			000000
00106	63*		IF(IMPL,GT,C160 TO 100	000000	
00110	64*		TMAX1=TMAX+.99999	000000	
00111	65*		TINC= DUMI7)	000000	
00111	66*	C			000000
00111	67*	C			000000
00112	68*		IF(NU,EQ,.99999)NU=0.02	000000	
00114	69*		IF(TMI,LT,.99999)TMI=212.0	000000	
00116	70*		IF(TOI,LT,.99999)TOI=60.0	000000	
00120	71*		IF(PH,LT,.99999)PH=2.0*PO	000000	
00122	72*		IF(MFH,LT,.99999)MFH=9000.0	000000	
00124	73*		IF(TDE,LT,.99999)TDE=4.0	000000	
00126	74*		IF(CP2,LT,.99999)CP2=2.93E-4	000000	
00130	75*		IF(TO2,LT,.99999)TO2=40.0	000000	
00132	76*		IF(TM2,LT,.99999)TM2=212.0	000000	
00134	77*		IF(R,LT,.99999)R=3.08E-4	000000	
00134	78*		IF(CM,LT,.99999)CM=0.6	000000	

5

15

00223	136*	C				000412
00223	137*	C				200412
00223	138*	C				000412
00225	139*		A=AMAX1(C1-C,D)/TYINC			000422
00226	140*		RE=AMIN1(PM,WP1,A)/EF1			000432
00226	141*	C				000432
00226	142*	C				000432
00226	143*	C				000432
00226	144*	C				000432
00227	145*		IF(T.LE.101)INT=1.0			000445
00227	146*	C				000445
00231	147*		IF(T.GE.(Y01+YDE).AND.			000453
00231	148*		1 INT.(Q.1.)INT=0.0			000453
00231	149*	C				000453
00233	150*		IF(T.GE.YH1)INT=-1.0			000472
00233	151*	C				000472
00235	152*		IF(T.LY.4TH1-YDE).AND.			000500
00235	153*		1 INT.(Q.-1.)INT=0.0			000500
00235	154*	C				000500
00237	155*		IF(1MPL.LE.1)RETURN			000517
00237	156*	C				000517
00241	157*		TSU=AMAX1(TSU,T)			000526
00242	158*		TSL=AMIN1(TSL,T)			000534
00243	159*		MF=AMAX1(MF,C)			000542
00244	160*		MFU=AMAX1(MFU,MF)			000550
00245	161*		IF(TIME.LY.THAX1)RETURN			000556
00246	162*	C				000556
00246	163*	C				000556
00246	164*	C				000556
00247	165*		CHI=CHI+CM			000563
00250	166*		CCI=CCI+CC0			000570
00251	167*		CM = CM/PD			000573
00251	168*	C				000573
00252	169*		RETURN			000576
00253	170*		1010 FORMAT(10,26HTS WORKING FLUID FLOW RATE,F12.3			001070
00253	171*		1 ,32H GREATER THAN MAXIMUM ALLOWED,F12.3)			001070
00254	172*		1020 FORMAT(10,14HTS INPUT POWER,F12.3			001070
00254	173*		1 ,44H GREATER THAN MAXIMUM ALLOWED CHARGE RATE,F12.3)			001070
00255	174*		1030 FORMAT(10,23HTS STORAGE TEMPERATURE ,F12.3			001070
00255	175*		1 ,20H OUTSIDE MINIMUM ,F12.3			001070
00255	176*		2 ,15H AND MAXIMUM,F12.3)			001070
00255	177*	C				001070
00256	178*		END			001070

7.45 TURBINE (PNEUMATIC)

The turbine model is based on a high pressure ratio, constant angular velocity design. The turbine is assumed to be designed to a set of of operating conditions defined in terms of user specified parameters. The polytropic efficiency is only weakly related to angular velocity. Initial calculations are made with the design polytropic efficiency, and refinements are then computed after off-design parameters are calculated.

Basic Equation

The equation for output power P2 is

$$P2 = M * CP * (T1 - TA)$$

Inputs

Parameter/Port	Description	Units
M	Inlet mass flow rate	lb/h
CP	Air heat capacity (D = 7.2×10^{-5})	kwh/lb/°F
T 1	Input air temperature	°F
TA	Ambient air temperature	°F
MD	Design mass flow rate (D = 4800)	lb/h
TID	Design inlet air temperature (D = 600)	°F
PID	Design inlet pressure (D = 117.6)	psi
P2D	Design exit pressure (ambient) (D = 14.7)	psi
T2D	Design exit temperature (ambient) (D = 70)	°F
PS	Storage vessel pressure	psi
RS	Angular velocity	rpm
EF 1	Input product efficiency	-
MP 1	Maximum input power	kw
P 1	Input power	kw
CK	Capacity cost coefficient ¹ (D = 0.015)	-
FO	Turbine mass flow exponent for capital cost (D = 0.75)	-
G	Turbine temperature exponent for capital cost (D = 0.5)	-
NPD	Design Polytropic Efficiency (D = 0.88)	-

Outputs

Variable/Port

P 2	Output power	kw
CC0	Turbine cost/year	\$
PR	Back pressure	psi
T0	Torque	ft-lb

D - Default values supplied

¹ CK = Capital cost (known unit)/[(design point mass flow rate)^{FO} * (design point temperature + 460)^G * LN (inlet/outlet pressure ratio)*LE], where LE = life expectancy in years.

Outputs Cont.

<u>Variable/Port</u>		<u>Description</u>	<u>Units</u>
EF	2	Output product efficiency	-
MP	2	Maximum discharge power	kw

Statistics

MP		Maximum power observed	kw
----	--	------------------------	----

The calculation sequence and the default values are based on the assumption of a high pressure ratio, constant angular velocity turbine, rated at 150 kw and a pressure ratio of 8. The equations used relate first order effects among the various physical quantities and were derived from first principles originally in support of the research work of Reference 1. Cost scaling was also developed in that reference based on cost estimates from turbomachinery manufacturers.

-
1. "Closed Cycle High Temperature Control Receiver Concept for Solar Electric Power," BEC/EPRI RP377-1, June 1976.

Calculation Sequence

1) Costs

$$CC = CK*(MD)**F0*(TID+460)**G*LN(PID/P2D)$$

2) Back Pressure PR determined by

$$PR = (M/MD)*PID* \sqrt{(T1+460)/(TID+460)}$$

If PR > PS write DIAGNOSTIC

3) Efficiency

$$RAT = (PID/P2D)**(2/7)$$

$$EFF = (RAT-1)/(RAT**(1/NPD)-1)$$

4) Power Out

$$P2 = M*CP*(T1-TA)*EFF$$

5) Torque

If RS = 0, set T0 = 0 and go to 6)

$$T0 = P2*(737.6)/(RS*2\pi/60)$$

6) Efficiency and maximum power

$$EF2 = EF1*EFF$$

$$MP2 = \min(MP1*EFF, MD*CP*(T1-TA))$$

7) Compute Statistics and Costs

U

```

00101 21* C EF2 - OUTPUT PRODUCT EFFICIENCY 000000
00101 22* C MP2 - MAXIMUM DISCHARGE POWER, KW 000000
00101 23* C MOP - MAXIMUM POWER OBSERVED, KW 000000
00101 24* C INPUTS 000000
00101 25* C M - INLET MASS FLOW RATE, LB/HR 000000
00101 26* C CP - AIR HEAT CAPACITY, KWH/LB/DEG F 000000
00101 27* C T1 - INPUT AIR TEMPERATURE, DEG F 000000
00101 28* C TA - AMBIENT AIR TEMPERATURE, DEG F 000000
00101 29* C MD - DESIGN MASS FLOW RATE, LB/HR 000000
00101 30* C TID - DESIGN INLET AIR TEMPERATURE, DEG F 000000
00101 31* C PID - DESIGN INLET PRESSURE, PSI 000000
00101 32* C P2D - DESIGN EXIT PRESSURE (AMBIENT), PSI 000000
00101 33* C T2D - DESIGN EXIT TEMPERATURE (AMBIENT), PSI 000000
00101 34* C PS - STORAGE VESSEL PRESSURE, PSI 000000
00101 35* C RS - ANGULAR VELOCITY, RPM 000000
00101 36* C EF1 - INPUT PRODUCT EFFICIENCY 000000
00101 37* C MPI - MAXIMUM INPUT POWER, KW 000000
00101 38* C PI - INPUT POWER, KW 000000
00101 39* C CK - CAPACITY COST COEFFICIENT 000000
00101 40* C F - TURBINE MASS FLOW EXPONENT FOR CAPITAL COST 000000
00101 41* C G - TURBINE TEMPERATURE EXPONENT FOR CAPITAL COST 000000
00101 42* C MPD - DESIGN POLYTROPIC EFFICIENCY 000000
00101 43* C
00103 44* COMMON /CIMPL/IMPL,ICHT/CTIME/ TIME /CSINUL/DUM(T),THAX /COST/CCI
00104 45* REAL MP2,MOP, M,MD, MPI,MPD 000000
00105 46* DATA PI /3.14159/ 000000
00105 47* C 000000
00107 48* IF(IMPL.EQ.C) GO TO 100 000000
00111 49* IF(CP.EQ..99999) CP = 72.0E-6 000002
00113 50* IF(TA.EQ..99999) TA = 70.0 000007
00115 51* IF(MD.EQ..99999) MD = 4000. 000014
00117 52* IF(TID.EQ..99999) TID=800.0 000021
00121 53* IF(PID.EQ..99999) PID=117.6 000026
00123 54* IF(P2D.EQ..99999) P2D=14.7 000033
00125 55* IF(T2D.EQ..99999) T2D=70.0 000040
00127 56* IF(CK.EQ..99999) CK =0.015 000045
00131 57* IF(F.EQ..99999) F =0.75 000052
00133 58* IF(G.EQ..99999) G =0.5 000057
00135 59* IF(MPD.EQ..99999)MPD= .88 000064
00137 60* MOP = C. 000071
00140 61* RS= AMAX(10.0,AMIN(1RS, 4000.)) 000072
00141 62* THAX=.99999*THAX 000104
00142 63* CC = CK*MD**F*(TID+460.)***G*ALOG(PID/P2D) 000107
00142 64* C DETERMINE BACK PRESSURE 000107
00142 65* C 000107
00143 66* 100 RAT= (PID/P2D)**.2857 000137
00144 67* EFF= (RAT-1.0)/(RAT** (1./MPD) - 1.0) 000146
00145 68* PR = M/MD*PID*507*((T1 + 460.0)/(TID+460.)) 000162
00145 69* C 000162
00146 70* IF(PA.GT.PS) GO TO 1000 000200
00146 71* C 000200
00146 72* C POWER OUTPUT 000200
00146 73* C 000200
00146 74* C 000200
00150 75* 200 P2= M*CP*(T1-TA)*EFF 000204
00151 76* TO = 0. 000212
00152 77* IF(RS.EQ.C. .OR. P1.CO. 0.) GO TO 300 000213

```

TU

00152	78*	C	TORQUE	000213
00154	79*	C	$TQ = P2 * 737.6 / (RS * 2.0 * P1 / 60.0)$	000223
00154	80*	C		000223
00154	81*	C	EFFICIENCY AND MAXIMUM POWER	000223
00154	82*	C		000223
00155	83*	C	300 EF2 = EF1*EFF	000233
00156	84*	C	NP2 = AMIN(NP1*EFF,ND*CP*AT1-TA1)	000235
00157	85*	C	IF(IMPL.LE.1) RETURN	000241
00161	86*	C	NOP = AMAX(NOP,P2)	000260
00162	87*	C	IF(TIME.LT.TMAX1) RETURN	000266
00164	88*	C	CCI = CCI + CC	000275
00164	89*	C		000275
00165	90*	C	RETURN	000300
00165	91*	C		000300
00166	92*	C	1000 IF(IMPL.EQ.2)WRITE(14,10)0) PR,P3	000304
00173	93*	C	2010 FORMAT (1H0,21HTURBINE BACK PRESSURE,F12.3,	000315
00173	94*	C	1 39H GREATER THAN STORAGE VESSEL PRESSURE ,F12.3)	000315
00174	95*	C	IF(IMPL.EQ.2)ICHT=ICHT+1	000315
00174	96*	C		000315
00176	97*	C	50 TO 200	000323
00176	98*	C		000323
00177	99*	C	END	000470

7.46 UTILITY

The utility model has two power outputs corresponding to baseload and peak generation, with corresponding generation cost inputs. A surplus power input is also provided with cost credit depending on whether baseload or peak power is reduced. Total energy cost, total output power and total peak load requests are monitored.

Inputs

<u>Parameter/Port</u>	<u>Description</u>	<u>Units</u>
BS	Baseload generation (default = 0.1)	kw
CB	Cost of baseload generation/kwh	\$
MP	1 Maximum power capacity (default = 1×10^8)	kw
P	0 Surplus power returned to utility	kw
RE	Peak generation request	kw
CP	Cost of peak load generation/kwh	\$
CC	Capital cost/year	\$
CM	Maintenance cost/year	\$

Outputs

<u>Variable/Port</u>	<u>Description</u>	<u>Units</u>
P	1 Baseload generation (= BS)	kw
MP	2 Maximum power capacity (= MP1)	kw
P	2 Peak load generation	kw
CO	Cost of energy used (state)	\$

Statistics

SR	Sum of requested peak generation	kwh
SP0	Sum of output energy	kwh
SP	Sum of surplus energy	kwh
VSP	Value of surplus energy	\$

Calculation Sequence

1) Power outputs

If $BS > MP1$, write diagnostic

$$P1 = BS, MP2 = MP1$$

$$P2 = \text{MIN} (MP1 - BS, RE)$$

2) Energy cost dynamics

$$C0 = BS * CB + (P2 - P0) * CX$$

$$CX = \begin{cases} CP & \text{if } P2 - P0 > 0 \\ 0 & \text{if } P2 - P0 < 0 \end{cases}$$

3) Statistics

$$SR = SR + RE * TINC$$

$$DEL = \begin{cases} 0 & \text{if } P2 > P0 \\ (P0 - P2) * TINC & \text{if } P0 > P2 \end{cases}$$

$$SP0 = SP0 + (P1 + P2 - P0) * TINC + DEL$$

$$SP = SP + DEL$$

$$VSP = VSP + DEL * CB$$

Where $TINC = \text{integration step size}/2$

4) Compute Costs

SUBROUTINE UT ENTRY POINT 000217

STORAGE USED CODE(1) 000341; DATA(0) 000045; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 CEMPL 000002
0004 CTIME 000001
0005 CSMPL 000010
0006 COST 000011

EXTERNAL REFERENCES (BLOCK, NAME)

0007 NUDMS
0010 N1025
0011 NERR35

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000053 100L	0001 000146 200L	0000 000004 208F	0006 R 000000 CCI	0006 R 000001 CMI
0004 000002 C0P	0000 R 000002 CX	0005 R 000000 DUM	0003 I 000001 ICMT	0003 I 000000 IMPL
0000 000023 INJP5	0006 R 000010 SPD	0006 R 000004 TDE	0000 R 000003 TERM	0004 R 000000 TIME
0000 R 000001 TIMC1	0006 000005 TLD	0005 R 000007 THAX	0000 R 000000 THAX1	0006 R 000007 UTD
0006 R 000006 UTV	0006 R 000003 VOC			

00100	1*	CUT		000000
00101	2*		SUBROUTINE UT(P1,MP2,P2,CO,COD,ICO,SR,SP0,SP,VSP	000000
00101	3*	1	,BS,CB,MPI,PO,RE,CP,CC,CN)	000000
00101	4*	C		000000
00101	5*	C	PURPOSE MODEL OF UTILITY CAPABLE OF PRODUCING	000000
00101	6*	C	BASELOAD AND PEAKLOAD POWER, AND OF	000000
00101	7*	C	ABSORBING SURPLUS POWER	000000
00101	8*	C		000000
00101	9*	C	METHOD COMPUTE PEAKLOAD GENERATION AND ENERGY COST	000000
00101	10*	C		000000
00101	11*	C	WRITTEN BY Y.H.CHAN VERSION 1, JUNE 8, 1977	000000
00101	12*	C		000000
00101	13*	C	CALL SEQUENCE	000000
00101	14*	C	OUTPUT	000000
00101	15*	C	P1 -BASELOAD GENERATION,KW	000000
00101	16*	C	MP2 -MAXIMUM POWER CAPACITY,KW	000000
00101	17*	C	P2 -PEAKLOAD GENERATION,KW	000000
00101	18*	C	CO -COST OF ENERGY USED (STATE), \$	000000
00101	19*	C	COD -ENERGY COST RATE, \$/HR	000000
00101	20*	C	ICO -INTEGRATION CONTROL FOR CO	000000
00101	21*	C	STATISTICS	000000
00101	22*	C	SR -SUM OF REQUESTED PEAK GENERATION,KW	000000

00101	23*	C	SPD -SUM OF OUTPUT ENERGY, KWH	000000
00101	24*	C	SP -SUM OF SURPLUS ENERGY, KWH	000000
00101	25*	C	VSP -VALUE OF SURPLUS ENERGY, \$	000000
00101	26*	C	INPUTS	000000
00101	27*	C	PS -BASELOAD GENERATION (DEFAULT=0.),KW	000000
00101	28*	C	CB -COST OF BASELOAD GENERATION/KWH, \$	000000
00101	29*	C	MP1 -MAXIMUM POWER CAPACITY,KW	000000
00101	30*	C	PD -SURPLUS POWER RETURNED TO UTILITY,KW	000000
00101	31*	C	RE -PEAK GENERATION REQUEST, KW	000000
00101	32*	C	CP -COST OF PEAKLOAD GENERATION/KWH, \$	000000
00101	33*	C	CC -CAPITAL COST/YEAR, \$	000000
00101	34*	C	CM -MAINTENANCE COST/YEAR, \$	000000
00101	35*	C	COMMON /CINPL/INPL,ICHT/CIME/TIME/CSIMUL/DUNIT1,TMAX	000000
00103	36*		X /COST/CCT,CMI,COP,VDE,TDE,ILD,UTV,UTD,SPD	000000
00103	37*		REAL MP2,MP1	000000
00104	38*			000000
00104	39*	C	IF (INPL.GI.D)60 TO 100	000000
00105	40*		IF (IS.EQ..99999)BS=0.	000000
00107	41*		IF (MP1.EQ..99999)MP1=1.E8	000002
00111	42*		TMAX)=TMAX+.99999	000006
00123	43*		SR=0.	000013
00114	44*		SP=0.	000014
00115	45*		SPO=0.	000017
00116	46*		VSP=0.	000020
00117	47*		RE=0.	000021
00120	48*		PD=0.	000022
00121	49*			000023
00121	50*	C	COMPUTE POWER OUTPUTS	000023
00121	51*	C		000023
00121	52*	C		000023
00122	53*		TINC1=DUNIT1*.5	000024
00123	54*		IF (BS.LE.MP1)60 TO 100	000027
00125	55*		WRITE (6,208185,MP1	000033
00131	56*		208 FORMAT(10,10N BASELOAD ,F12.3,32N EXCEEDS MAXIMUM POWER CAPACITY,	000042
00133	57*		1 F12.3)	000042
00132	58*		IF (INPL.EQ.2)ICHT=ICHT+1	000042
00134	59*		BS=MP1	000050
00134	60*	C		000050
00135	61*		100 P1=BS	000053
00136	62*		MP2=MP1	000054
00137	63*		P2=AMIN(MP1-BS,RE)	000054
00137	64*	C		000054
00137	65*	C	COMPUTE ENERGY COST	000054
00137	66*	C		000054
00140	67*		CX=0.	000056
00141	68*		IF (P2.GT.PD)CX=CP	000065
00143	69*		IF (ICHT.NE.0)CDD=BS*CB+(P2-PD)*CX	000073
00145	70*		IF (INPL.LE.1)RETURN	000104
00145	71*	C		000104
00145	72*	C	STATISTICS	000104
00145	73*	C		000104
00147	74*		SR=SR+RE*TINC1	000113
00150	75*		SPD=SPD+(P1+P2-PD)*TINC1	000117
00151	76*		IF (P2.GT.PD) GO TO 200	000125
00153	77*		TERM=(PD-P2)*TINC1	000131
00154	78*		SPD= SPD+TERM	000135
00155	79*		SP= SP+ TERM	000137

UT

```

00154 80*      WSP= WSP+ CB*TERN
00157 81*
00157 82*      C
00157 83*      ZDD IF (TIME.C7.TMAX) RETURN
00161 84*      CCI=CCI+CC
00162 85*      CHI=CHI+CM
00163 86*      VDE=VDE-CB+WSP
00164 87*      TDE=TDE-SPD+SP
00165 88*      UTV=UTV+CO
00166 89*      UTD=UTD+SPD
00167 90*      SPD= SPD+SP
00170 91*      RETURN
00171 92*      END

```

```

000141
000146
000146
000146
000154
000157
000162
000166
000172
000175
000200
000203
000340

```

7.47 WIND

This model computes wind speed either from user supplied time histories (data tape) or by generating a set of random numbers with user supplied daily and yearly average profiles and user specified random variation. If user supplied profiles are available then the wind speeds are generated from the following equation:

Basic Equation

$$WV = [PD(TD) + N(T)] * PY(WY) / M$$

where PD is the user supplied daily mean profile
 TD is the time of the day
 PY is the user supplied yearly profile
 WY is the week of the year
 N is white noise with user specified probability distribution

$$M = \frac{1}{J} \sum_{i=1}^J PY(i)$$

<u>Tables</u>	<u>Description</u>	<u>Units</u>
PD	Daily profile versus TD (default = 0)	miles/hr
PY	Yearly profile versus WY	arbitrary
DF	Density function for white noise terms (tabular with speed WV)	arbitrary

Inputs

Parameter/Port

WV	1	Wind speed data file input	miles/hr
TD		Time of day	hr
WY		Week of the year	-

Outputs

Variable/Port

WV	2	Wind speed	miles/hr
M		Mean of yearly profile	
TIM		Last time a random sample was generated	hr

Statistics

MV	Maximum speed	miles/hr
AV	Average speed (expected daily wind)	miles/hr

Calculation Sequence

- 1) Compute distribution function and mean M (first pass only)

$$F(WV) = \frac{(\sum DF(V_i) : V_i \leq WV)}{\sum DF(V_i)}$$

- 2) Check for data file input

If WV1 = .99999 go to 3)

WV2 = WV1

Go to 5)

- 3) Generate white noise input N

If TIME = TIM go to 5)

U = random noise sample, uniformly distributed [0,1]

Interpolate to find $N = F^{-1}(U)$

TIM = TIME

- 4) Compute wind speed

WV2 = [PD(TD) + N] * PY(WY)/M

- 5) Compute Statistics

W

[illegible]

MD

00176	81*	VVO = MVI	000247
00177	82*	GO TO 150	000251
00177	83*	C GENERATE WHITE NOISE WITH DIST. WF	000251
00177	84*	C	000251
00200	85*	100 IF1 TIME.EO.TIME0 GO TO 150	000253
00202	86*	CALL UNIF(U,IX)	000255
00203	87*	NP1=NP+1	000261
00204	88*	WN = TBLU1(U,WF(4+NP),WF(3),1,-NP1)	000264
00204	89*	C	000264
00204	90*	C GENERATE WIND SPEED USING DAILY AND YEARLY PROFILES	000264
00205	91*	DNV = TBLU1(TO,PD(4),PD(4+ND),1,-ND)	000303
00206	92*	YMW = TBLU1(WY,PY(4),PY(4+NY),1,-NY)	000323
00207	93*	MV0 = (DNV + WN)* YMW / XM	000343
00210	94*	TIME=TIME	000350
00210	95*	C	000350
00210	96*	C MAX. OBSERVED WIND SPEED	000350
00211	97*	150 IF(IMPL.LC.1) RETURN	000353
00213	98*	APV = AMAX1(APV,MV0)	000361
00214	99*	RETURN	000367
00215	100*	END	000472

7.48 TURBINE/GENERATOR

This component uses a power curve relationship with wind velocity to model the wind turbine and generator. It may be used in place of the more detailed wind turbine-transmission-generator components where a simplified analysis is desirable, or where a nonstandard wind generator model is desired. The model may be used for either A.C. or D.C. power generation.

Basic Equations

$$P_2 = 0 \quad \begin{cases} WV < WVO \\ WV > WV1 \end{cases}$$

$$P_2 = V \cdot I / 1000 \quad WVO \leq WV \leq WV1$$

FIGURE 7.48: OUTPUT POWER VERSUS WIND VELOCITY

<u>Tables</u>	<u>Description</u>	<u>Units</u>
PW	Wind generation power versus wind velocity ¹	kw

Inputs

Parameter/Port

V	Bus voltage (Rated)	volts
WVO	Power cutin velocity	mph
WV1	Power cutout velocity	mph
WV	Wind velocity	mph
CC	Capital cost/year	\$
CM	Maintenance cost/year	\$
EC	Control Energy Rate	\$/hr

Outputs

Variable/Port

I	Bus current	amps
P	2 Real power output	kw

Statistics

MI	Maximum current	amps
MP0	Maximum power	kw
SP	Total output energy	kwh
CO	Total operating costs	\$

¹ Output power including mechanical and electrical efficiencies

Calculation Sequence

1) Initialize statistics

2) Compute P2 and I

$$P2 = \begin{pmatrix} PW(WV) & WVO \leq WV \leq WV1 \\ 0 & \text{otherwise} \end{pmatrix}$$

$$I = P2 * 1000 / V$$

3) Compute Statistics and Costs

SUBROUTINE WP ENTRY POINT 000153

STORAGE USED CODE(1) 000232; DATA(01 000024; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 CEMPL 000001
0004 COST 000003
0005 CTIME 000001
0006 CSTMUL 000010

EXTERNAL REFERENCES (BLOCK, NAME)

0007 TBLU1
0010 NERR15

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000006 IOL	0001 000006 ZOL	0004 R 000000 CC	0004 R 000001 CM	0004 R 000002 COP
0006 000000 DUM	0003 I 000000 IMPL	0000 000006 INJPS	0000 I 000000 N	0007 R 000000 TBLU1
0005 R 000000 TIME	0006 R 000006 TINC	0006 R 000007 THAX	0000 R 000001 THAX1	

00100	1*	CMP		000000
00101	2*		SUBROUTINE WP I PW,PI,PO,AMI,AMP,SP,CO,VO,WVC,MV1,MV,CCI,CMI,EC)	000000
00101	3*	C		000000
00101	4*	C	PURPOSE MODEL THE WIND TURBINE AND GENERATOR USING A POWER CURVE	000000
00101	5*	C		000000
00101	6*	C	WRITTEN BY A.W. VARRÉN	000000
00101	7*	C	VERSION 1, MARCH 3 1977	000000
00101	8*	C	CALL SEQUENCE	000000
00101	9*	C	TABLES	000000
00101	10*	C	PW - WIND GENERATION POWER IN KW VERSUS WIND VELOCITY IN MPH	000000
00101	11*	C		000000
00101	12*	C	OUTPUTS	000000
00101	13*	C	BI - OUTPUT BUS CURRENT, AMPS	000000
00101	14*	C	PO - POWER OUTPUT, KW	000000
00101	15*	C	AMI - MAX. OBSERVED CURRENT, AMPS	000000
00101	16*	C	AMP - MAX. OBSERVED POWER, KW	000000
00101	17*	C	SP - TOTAL OUTPUT ENERGY, MWH	000000
00101	18*	C	CO - OPERATING COST, \$	000000
00101	19*	C		000000
00101	20*	C	INPUTS	000000
00101	21*	C	VO - RATED BUS VOLTAGE, VOLTS	000000
00101	22*	C	WVO - POWER CUTIN VELOCITY, MPH	000000
00	23*	C	MV1 - POWER CUTOFF VELOCITY, MPH	000000
00	24*	C	MV - WIND VELOCITY, MPH	000000
00	25*	C	CCI - CAPITAL COST / YEAR, \$	000000

W/P

00101	26*	C	CNI = MAINTENANCE COST / YEAR, \$	000000
00101	27*	C	EC = CONTROL ENERGY RATE, \$/HR	000000
00101	28*	C		000000
00103	29*		DIMENSION PW(1)	000000
00104	30*		COMMON / CINPL / INPL	000000
00105	31*		COMMON/COST/ CC,CN,COP /CTIME/ TIME /CSTIMUL/ DUM(6),TIME,TMAX	000000
00105	32*	C		000000
00105	33*	C	POWER OUTPUT CALCULATIONS	000000
00105	34*	C		000000
00106	35*		PO = 0.	000000
00107	36*		IF(MV.LT.MVO .OR. MV.GT.MV1) GO TO 10	000001
00111	37*		N = PW(2)	000017
00112	38*		PO = TBLV(MV,PW(4),PW(4+N),1,-N)	000026
00113	39*		10 BI = PO*1000/V0	000046
00113	40*	C	STATISTICS	000046
00113	41*	C		000046
00114	42*		IF(INPL.GT.0) GO TO 20	000051
00116	43*		CO = 0.	000054
00117	44*		API = 0.	000055
00120	45*		AMP = 0.	000056
00121	46*		SP = 0.	000057
00122	47*		TMAX1=TMAX+.99999	000060
00123	48*	20	IF(INPL.LE.1) RETURN	000064
00125	49*		AVI = AMAX1(API,BI)	000072
00126	50*		AMP = AMAX1(AMP,PO)	000100
00127	51*		SP = SP + PO*.5*TIME	000106
00130	52*		CO = CO + EC*.5*TIME	000113
00130	53*	C	COST SUMMATION	000113
00131	54*		IF(TIME.L.TMAX1) RETURN	000120
00133	55*		CC = CC + CCI	000127
00134	56*		CN = CN + CNI	000132
00135	57*		COP = COP + CO	000135
00136	58*		RETURN	000140
00137	59*		END	000231

W/P

7.49 WIND TURBINE

This component models the wind turbine in terms of physical properties such as blade radius, power coefficient, and design tip speed ratio.¹ The step-up gear ratio is computed based on design rotor speed.

Basic Equations

Output power is given by

$$P2 = CP * 1/2 * AD * A * (WV * C)^3 * k$$

where:

CP = effective power coefficient (tabular with WV)

A = $\pi * (BR)^2$

C = 1.4667 (mph to ft/sec. conversion)

k = 1.3558×10^{-3} (ft-lb to kw-sec. conversion)

¹ NASA CR 134937 "Design Study of Wind Turbines - 50kw to 3000 kw - For Electric Utility Applications", Kaman Aerospace Corporation, February 1976.

Inputs

<u>Parameter/Port</u>	<u>Description</u>	<u>Units</u>
WV	Wind speed	mph
V0	Mean wind speed (yearly)	mph
VR	Rated wind speed (default = $1.35 \times V0$)	mph
RS	Rotor speed	rpm
RSG	Generator shaft speed (design)(default = 1800)	rpm
BR	Blade radius	ft
EC	Cost to operate controls	\$/h
AD	Air density (default = 0.0023)	slugs/ft ³
LAM ¹	Design tip speed ratio (default = 9.4)	-
CPM ²	Maximum power coefficient at V0 (default = 0.4)	-
CP	Effective power coefficient (default table versus V0/WV)	-
CC	Capital cost/year	\$
CM	Maintenance cost/year	\$

Outputs

<u>Variable/Port</u>			
P	2	Output mechanical power	kw
T0		Mechanical torque	ft-lb
C0		Total operating cost	\$
GR		Step-up gear ratio	-
RAP		Rated output power	kw

Statistics

MT	Maximum torque	ft-lb
MP0	Maximum power	kw
SP	Total energy delivered	kwh

¹ LAM may be computed using the design equation:

$$LAM = \sqrt{(8/13 * \text{solidity constant} * \text{design lift coefficient})}$$

² If default CP table not used then set CPM = CP(rated wind speed)

Calculation Sequence

- 1) First pass - Compute Gear Ratio and Rated Power

$$RS = (LAM*V0*C/BR)*(60/2\pi)$$

$$GR = RSG/RS$$

$$RAP = .5*CP1*AD*A*(VR*C)^3$$

where

$$CP1 = \begin{cases} CPM*F(V0/VR) & \text{If CP default used} \\ CPM & \text{otherwise} \end{cases}$$

- 2) Compute power coefficient CP

If $WV = 0$ set $P2=TO=0$ and go to 4)

If CP default used, then

$$CP = CPM*F(V0/WV)$$

where F is shown in Figure 7.49

- 3) Power and torque

$$A = \pi*BR^2$$

$$P = .5*CP*AD*A*(WV*C)^3 \quad (C = 1.4667)$$

$$TO = P/(RS*2\pi/60)$$

$$P2 = P/k \quad (k = 1.3558 \times 10^{-3})$$

- 4) Compute Statistics and Costs

FIGURE 7.49 GENERALIZED MACHINE POWER OUTPUT PERFORMANCE

WT

SUBROUTINE WT ENTRY POINT 000270

STORAGE USED CODE(1) 000413; DATA(0) 000070; BLANK COMMON(2) 000000

COMMON BLOCKS

0003 C1MPL 000003
0004 CTIME 000001
0005 CSIMUL 000010
0006 COST 000003

EXTERNAL REFERENCES (BLOCK, NAME)

0007 TELUI
0010 NEPR38

STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001 000122 100L	0001 000211 200L	0000 R 000034 A	0006 R 000000 CCI	0006 R 000001 CNI
0006 R 000002 C01	0000 R 000033 CP1	0000 R 000026 C1	0000 R 000027 C2	0005 000000 DUM
0000 R 000000 F	0003 I 000000 IMPL	0000 000051 INJPS	0000 R 000035 P	0000 R 000030 PJ
0007 R 000000 TPLUI	0004 R 000000 TIME	0005 R 000006 TINC	0000 R 000032 TINC2	0005 R 000007 TMAX
0000 R 000031 TMAX1				

```

00100 1+ CNT1
00101 2+ SUBROUTINE WT (P2,TD,CD,GR,RAP,MT,NPO,SP,WV,VO,VR,RS, RSG,BR,EC,
00101 3+ 1 AD,LAN,CPM, CP,CC,CN)
00101 4+ C
00101 5+ C PURPOSE MODEL WIND TURBINE POWER OUTPUT
00101 6+ C
00101 7+ C METHOD COMPUTE POWER COEFFICIENT AND ROTOR SPEED FROM PHYSICAL
00101 8+ C DESIGN PARAMETERS. RATED POWER COEFF. IS 3/4 OF CPM.
00101 9+ C
00101 10+ C WRITTEN BY A. W. WARREN VERSION 2, APRIL 6 1977
00101 11+ C
00101 12+ C CALL SEQUENCE
00101 13+ C OUTPUTS
00101 14+ C P2 - OUTPUT MECHANICAL POWER, KW
00101 15+ C TD - OUTPUT MECHANICAL TORQUE, FT-LB
00101 16+ C CD - OPERATING COST SUM, $
00101 17+ C GR - TURBINE/GENERATOR GEAR RATIO
00101 18+ C RAP - RATED OUTPUT POWER, KW
00101 19+ C MT - MAXIMUM TORQUE STATISTIC, FT-LB
00101 20+ C NPO - MAXIMUM POWER STATISTIC, KW
00101 21+ C SP - TOTAL OUTPUT ENERGY DELIVERED, KWH
00101 22+ C
001 23+ C INPUTS

```

000000
000000
000000
000000
000000
000000
000000
000000
000000
000000
000000
000000
000000
000000
000000
000000
000000
000000
000000
000000
000000
000000
000000
000000

WT

5

```

00161      81*      T0=P/(RS*PI/30.)
00162      82*      P2 = P4C2
00162      83*      C
00162      84*      C                      STATISTICS AND COSTS
00162      85*      C
00163      86*      200 IF(I)NPL.(LE.1) RETURN
00163      87*      C
00165      88*      CO = CO + EC*TIME2
00166      89*      MT = MAX(1, MT, T0)
00167      90*      MPO= MAX(1,MPO,P2)
00170      91*      SP = SP + P2*TIME2
00170      92*      C
00171      93*      IF(TIME.LT.1)RETURN
00173      94*      CC1 = CC1 + CC
00174      95*      CM1 = CM1 + CM
00175      96*      CMI = CMI + C0
00175      97*      C
00176      98*      RETURN
00177      99*      END

```

```

000177
000205
000205
000205
000205
000211
000211
000217
000223
000231
000237
000237
000243
000252
000255
000260
000260
000263
000412

```

8.0 EXAMPLES

This section gives five simple example simulations using the SIMWEST program. These examples exercise all physical components of the SIMWEST library and many of the model features such as Fortran code insertion and the file read capability. Each example contains the input data for model generation and analysis, selected printer output generated by the programs and a discussion of the results obtained. It is recommended that a user work through and understand the model connections for these examples before attempting to build more complex models such as that of Figure 1-7.

8.1 WIND TURBINE AND FILE READ MODEL

Figure 8.1-1 shows a simplified schematic of the wind turbine and file read model. In this example a wind turbine is used to feed power directly to a load. Wind and load time histories are read from a mass storage file and then used to drive the simulation. A histogram for power output is also included. Figure 8.1-2 shows the input data to build the model and print out some of the load file data. The order of the component definitions is such that information passes down the list, i.e. no component is defined before components in the INPUTS list. This assures that the component subroutines in the Fortran model will be called in the right order. The first Fortran statement is inserted in the model prior to the Fortran which sets up the iteration loop, while the second set of Fortran statements is within the loop and writes out the load file data from array TEMP the first pass through the model. Figure 8.1-3 shows the model schematic generated. In addition to showing the component connections, the names of the input connections are printed out. Notice that information passes not only from WT to GR but also vice-versa. The input RSIGR to WT is a feedback variable and so is the input RS GE to GR. It is the presence of the feedback variables which require several iterations to attain steady state.

FIGURE 8.1-1: WIND TURBINE AND FILE READ EXAMPLE

```

MODEL DESCRIPTION      TAPE READ TEST
  FORTRAN STATEMENTS
    COMMON /DATARD/ TEMP(448)
LOCATION=15      TI
LOCATION=3       TAW      INPUTS=TI
LOCATION=11      WD      INPUTS=TI,TAW(VAR=WW)
LOCATION=31      WT      INPUTS=WD
LOCATION=35      GR      INPUTS=WT
LOCATION=51      HG      INPUTS=WT(P =FIN)
LOCATION=39      GE      INPUTS=GR
LOCATION= 7      TAL     INPUTS=TI
LOCATION=19      LO      INPUTS=GE, TI, TAL (VAR=LO)
  FORTRAN STATEMENTS
    IF (IMPL.GT.0) GO TO 2
    WRITE (6,100) (TEMP(I),I=1,448)
100 FORMAT(1H ,12F10.3)
    2 CONTINUE
END OF MODEL
LIST STANDARD COMPONENTS
PRINT

```

FIGURE 8.1-2 INPUT DATA FOR FILE READ MODEL

The input data for several simulations using this model is shown in Figure 8.1-4. The component parameter values are first specified, those inputs not specified taking default values. A number of tables are then specified. The WD and LO tables are not really needed here. The wind and load file data was originally generated from an earlier run using these tables. Following the tables are the printer plot input commands, and the simulation values and print commands for a one week run. The parameter values for a second simulation which reads to the end of the file are then given. The last simulation attempts to read past the end of file.

Some of the output for the first simulation are shown in Figures 8.1-5 to 8.1-8. Figure 8.1-5 shows the output resulting from the FORTRAN STATEMENTS code. This data is formatted to output the time of the initial load value (0.0), the data increment step (0.25), and 446 load values at successive time increments. Figure 8.1-6 shows the power output histogram from the wind turbine. Almost 40% of the time the turbine reaches rated power (800 kw). Figure 8.1-7 shows a crossplot of wind turbine output versus wind velocity. The cutin velocity is about 12 mph and rated power is attained at about 28

```

TITLE= TAPE READ TEST
PARAMETER VALUES
CYCLES=2.01,10 TI=0,FUPHG=1000,FLOHG=50,NSTTAW=0,ITFTAW=1
-2 WT=21,ER WT=80,EC WT=1,CPMWT=.41,CC WT=16000,CM WT=2000
PAGE=.01, CC GR=1000,CM GR=200,NSTTAL=0,ITFTAL=2
PAGE=750,PSYGE=1800,DA GE=0.2,SR GE=.005,UO GE=400,CC GE=1000,CM GE=200
NO LO=.00574,CT LO=4,MN LO=0,STOLO=6,UE LO=.023
TABLE,PY WD=13
1.,4.33,8.27,13.,17.33,21.67,26.,30.33,34.67,39.,43.33,47.67,52.
25.27,28.25,31.56,34.49,39.52,56.61,65
TABLE,PD WD=7
0,4,8,12,16,20,24
0,0,0,6,6,6,0
TABLE,IF WD=16
0,3,6,9,12,15,18,21,24,27,30,33,36,39,42,45
5,44,160,380,480,512,440,376,307,270,148,76,40,22,9,3
TABLE, PLOGP =7
0,100,200,300,400,500,600
1,4,9,18,32,50,72
TABLE,PD LO=17
0,1.5,3,4.5,6,7.5,9,10.5,12
13.5,15,16.5,18,19.5,21,22.5,24
450,560,672,730,850,960,810,798,804
690,702,699,702,750,708,570,450
TABLE,FW LO=7
1,2,3,4,5,6,7
1,1,1,2,2,3,3,4,4,5,5,6,6,7
TABLE,PY LO=6
0,10,20,30,40,50
12,124,120,174,194,226
%1:TOP PLOTS,DISPLAY1
%END,US,TIME
P2 WT,US,TIME
P2 GR,US,P2 WT
LO2LO,US,TIME
P2 LO,US,TIME
DISPLAY2,P2 GE,US,TIME
TIME=.25,TMAX=168.,PRATE=24,PRINT CONTROL=3,INT NODE=3,OUTRATE=2
SIMULATE
PARAMETER VALUES,T0 TI=4200,NSTTAW=38,NSTTAL=39,TMAX=220
SIMULATE
PARAMETER VALUES,NSTTAL=40,

```

FIGURE 8.1-4 INPUT DATA FOR ANALYSIS PROGRAM

mph. The load-time profile for the simulation is shown in Figure 8.1-9. Daily peaks and troughs are clearly indicated. The lower levels of the last two days reflect week-end load modeling.

 PRINT NAME 20 DISPLAY NAME 2 MODE 3 TINC 25000 TMAX 168.00

TAPE READ TEST

CASE NO. 1

IDENTIFICATION HEADER FOR UNIT 11
 NO GENERATED MIND DATA

IDENTIFICATION HEADER FOR UNIT 12
 NO GENERATED LOAD DATA

1000	250	572,020	576,450	541,273	542,303	504,160	490,027	458,850	465,241	471,429	461,310
472,202	469,504	464,207	478,548	457,784	462,708	428,157	417,025	435,024	420,501	477,098	477,597
524,078	529,255	540,527	574,953	655,125	684,704	751,090	774,505	849,244	842,653	919,063	906,349
478,191	971,344	1035,062	1030,017	1019,400	1025,785	1025,351	1015,478	1029,836	1014,403	1021,731	1030,062
1036,085	1020,645	1024,203	1026,614	982,779	972,742	929,110	935,981	880,547	879,323	888,269	897,320
840,753	914,397	906,599	905,685	846,960	897,141	895,837	899,829	899,173	897,042	906,923	887,126
902,818	898,822	848,457	898,273	910,522	920,504	951,416	932,214	966,988	941,958	927,407	940,644
906,246	909,801	914,014	904,625	848,249	849,443	783,334	709,167	737,314	728,981	668,681	668,212
822,550	633,111	506,890	571,579	529,372	544,890	509,011	501,604	464,107	453,340	459,528	467,007
460,501	477,917	462,666	483,800	462,589	457,579	444,235	439,874	419,515	429,792	483,195	466,143
526,520	526,181	572,035	570,249	660,377	652,042	770,330	756,319	843,668	857,838	846,782	901,220
957,964	976,595	1026,770	1039,277	1021,489	1024,710	1030,602	1024,092	1031,524	1028,026	1013,419	1024,933
1010,140	1036,326	1032,017	1025,948	991,639	974,386	937,970	924,588	874,771	866,031	886,719	888,582
910,831	849,333	847,415	907,327	909,429	902,392	897,479	895,145	901,262	885,588	900,631	888,768
897,689	893,247	886,110	907,132	909,001	926,617	950,342	937,912	958,230	960,450	963,353	935,515
929,145	918,661	845,342	903,105	844,841	804,314	785,423	780,048	725,415	734,232	684,312	683,844
620,583	621,210	527,518	515,674	487,255	479,243	454,255	434,638	413,880	415,578	421,951	409,586
445,041	425,911	429,801	427,555	417,808	404,358	389,100	388,022	376,195	378,948	417,267	427,101
409,653	461,186	516,310	539,500	542,569	600,906	670,397	672,622	760,028	771,088	805,759	820,013
612,908	654,862	632,067	640,076	917,971	932,658	923,327	932,503	929,850	921,527	920,069	942,602
917,149	934,573	925,140	924,464	878,517	881,672	810,214	842,894	801,363	796,343	790,602	795,509
806,655	804,783	818,895	811,978	807,374	814,032	812,055	807,109	803,271	810,696	812,847	801,771
806,953	818,793	749,760	808,956	826,075	820,398	835,254	839,504	854,904	865,872	850,213	857,347
822,271	837,615	832,466	814,272	797,037	770,703	702,264	691,332	654,351	653,748	608,418	607,997
566,499	567,083	524,901	523,648	498,076	490,064	489,639	451,553	411,712	436,143	423,429	408,227
424,876	436,731	431,681	429,435	413,593	408,683	405,618	402,090	378,074	386,922	421,993	438,323
874,379	466,112	506,845	528,789	594,047	596,289	680,816	677,508	750,476	775,412	807,615	820,550
888,372	872,930	942,887	935,459	925,594	915,451	925,206	921,792	913,044	913,864	912,204	922,549
915,479	911,673	926,217	919,446	883,648	883,150	834,940	841,525	791,053	794,974	805,072	799,813
801,172	794,474	811,932	814,259	824,899	812,262	807,640	811,835	801,591	806,079	804,181	800,402
814,525	820,271	800,858	814,777	827,553	837,715	861,511	838,135	850,327	864,102	858,187	844,919
838,785	823,656	812,413	809,656	756,079	757,145	709,836	711,495	659,479	661,320	609,896	615,971
571,225	562,447	516,686	512,937	481,270	488,695	448,270	450,184	410,343	404,747	403,375	418,444
418,680	429,468	417,722	425,220	421,165	413,810	392,056	388,131	376,368	385,152	329,967	327,812
472,604	473,684	513,171	521,324	542,678	532,074	679,849	682,274	761,291	758,607	810,835	812,685
863,755	874,428	935,924	940,586	939,612	929,921	920,590	930,167	918,926	918,596	923,024	908,991
923,153	916,399	928,498	927,420	878,628	885,029	827,477	834,062	804,720	746,452	809,396	801,713
746,556	789,857	809,862	803,548								

FIGURE 8.1-5 TAPE READ FORMATTED LOAD DATA

HISTOGRAM FOR 1349 SAMPLES MEAN 900.77955 STANDARD DEV. 290.39880

TIME = 100.0
U = 1696.7

STATES

FIGURE 8.1-6 WIND TURBINE POWER HISTOGRAM

FIGURE 8.1-7 WIND POWER OUTPUT VERSUS WIND VELOCITY

374

TAPE READ TEST

BCS 40180-2

BCS 40180-2

FIGURE 8.1-B WEEKLY LOAD PROFILE

8.2 BATTERY STORAGE MODEL

A simplified schematic of the battery storage model is shown in Figure 8.2-1. In this model, wind power supplemented by a utility generation source is supplied to a power divider, which delivers power to the load as a first priority, and battery storage as second priority. Similarly, if the load cannot be met from the wind or storage, then the utility is requested to supply peaking power to meet the load. This model exercises the logic components including the priority interrupt.

Figure 8.2-2 shows the model generation input data for the model. The components are generally defined in the order of power flow shown in Figure 8.2-1. Ordering the component definition in this way is recommended to avoid convergence problems in the iteration loop. Thus, it would be somewhat better for consistency to define UT after WP rather than after LO in the model. All three types of model connections are illustrated in this example. For example, WP has the general input connection WD, MAB has the specific input connection WP (P,2 = FIN), and PD has the port to port connection PA (1,1). The port connections are especially useful for connecting up the multiport logic components PA and PD. The connection PA (1,1), for example, connects an input request of PD to PA and a power and maximum power input of PA to PD. It may be observed that the utility is connected up to the surplus port of PD. Thus the baseload power sent to MAB in effect is reduced whenever the load and battery cannot absorb all the power generated. The last component defined is the cost monitor CM, which receives cost input data from other components through a common block rather than by model connections. Figure 8.2-3 shows the model schematic generated by the program. Most of the connection inputs are shown but occasionally a model connection will be overprinted. For example, the input RE1PA to PD is not shown in 8.2-3. In cases like this it is necessary to check the Fortran model (EQNO) in order to verify the model connections.

FIGURE 8.2-1: BATTERY STORAGE EXAMPLE

MODEL DESCRIPTION		BATTERY TEST CASE
LOCATION=74	TI	
LOCATION=61	WD	INPUTS=TI
LOCATION=21	WP	INPUTS=WD
LOCATION=42	MAB	INPUTS=WP (P,2=FIN) ,UT (P,1=C2)
LOCATION=33	PD	INPUTS=MAB (FO=P) ,MAB (FO=WP) ,PA (1,1) ,PIB (2,2) , BA (RE=RE,2)
LOCATION=15	RE	INPUTS=PD (2,1)
LOCATION=17	BA	INPUTS=RE,PA (RE,2=RE)
LOCATION=45	PIB	INPUTS=BA
LOCATION=19	IV	INPUTS=BA
LOCATION=69	PA	INPUTS=IV (2,2) ,LO (1,0) ,PIB (4,2) ,UT (2,3)
LOCATION=76	LO	INPUTS=TI
LOCATION=62	UT	INPUTS=PD (SP=P,0)
LOCATION=1	OM	
END OF MODEL		
LIST STANDARD COMPONENTS		
PRINT		

FIGURE 8.2-2 BATTERY MODEL INPUT DATA

The input data for two simulations is shown in Figure 8.2-4. In the first simulation the battery is nearly full at time = 0 and the load is chosen larger on the average than the wind and utility power supplied. In the second simulation the reverse is true, i.e., the load is less than that supplied by the wind system, and the battery storage is fairly low. Figures 8.2-5 to 8.2-8 show results from the first simulation. The cost monitor output is shown in Figure 8.2-5. The energy cost of the wind system is low because the wind profile delivers high energy winds during most of the simulation. The average wind velocity in Figure 8.2-6 is about 22 mph. Figure 8.2-7 shows the wind power output supplied directly to the load. The median power output is seen to exceed 450 kw and occasionally output reaches 800 kw or rated power. Figure 8.2-8 shows the usage of battery energy to meet the load during the week, and the increase in storage capacity during the weekends. Since the battery subsystem was limited to 180 kw maximum discharge, the utility was frequently called to meet peak load. Thus about 10% of the load was satisfied by the utility.

BCS 40180-2

```

TITLED BATTERY MODEL TEST
PARAMETER VALUES
CN CM15.,LE CM30,
BS UTM20.,CB UTM.014,CP UTM.03,CC UTM1000.,CM UTM1000,
C1 MABM1,
CYCLEB=4,01,T0 T1=0,V WPM400,WVONPM8,WV1WPM60,DLINES=100,
CC WPM16000,CM WPM1200,PB1P10=2,
EC WPM.2
NC LOM.005,CY LOM4,MN LOM0,STDLOM6,VE LOM.023
RAPBAM200.,E1 BAM2000.,EDEBAM100,
VQ BAM100
DT BAM10.,CC BAM2000.,CM BAM100,
RAPRE=200.,CC RE=200.,RAPIV=200.,CC IV=0,
TABLE,PW NPM10
8,10,12,14,16,18,20,21,53,25,30
25,6,50,1,88,5,137,4,205,1,292.,400,6,500,782,8,800
TABLE,PV NDM13
0.,4,33,6,67,13.,17,33,21,67,26.,30,33,34,67,39.,43,33,47,67,52,
63,67,68,65,61,56,51,49,49,52,56,61,65
TABLE,PD NDM7
0,4,8,12,16,20,24
10,12,14,16,14,12,10
TABLE,DF NDM16
0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15
5,44,160,380,480,512,440,376,307,270,148,76,40,22,9,3
TABLE,PD LOM17
0,1,5,3,4,5,6,7,5,9,10,5,12
13,5,15,16,5,18,19,5,21,22,5,24
450,360,372,330,450,660,810,798,804
690,708,699,702,750,708,570,450
TABLE,PW LOM7
1,2,3,4,5,6,7
1,1,9,9,9,6,5
TABLE,PV LOM6
0,10,20,30,40,52
226,194,180,174,194,226
INITIAL CONDITIONS, PE BA M1990,
PRINTER PLOTS,DISPLAY1
WV2ND,V8,TIME
P1 PD,V8,TIME
P2 PD,V8,TIME
PE BA,V8,TIME
DISPLAY2
P2 IV,V8,TIME
NE2BA,V8,TIME
RE1LO,V8,TIME
TING=25,THAX=336.,PRATE=8,PRINT CONTROL=3,INT MODE=3,OUTRATE=8
SIMULATE
PARAMETER VALUES,BS UTM0.,NC LOM.005
E1 BAM1000.,EDEBAM200,
CC IV=1000,
INITIAL CONDITIONS,PE BAM250,
SIMULATE

```

WIND ENERGY STORAGE COST SUMMARY

- 30 YEAR LIFE CYCLE

YEARLY SYSTEM COSTS

CAPITAL COST (INCLUDING FIXED CHARGES)	86400, \$
FIXED O + M COST	2300, \$
OPERATING + FUEL COST	1753, \$
TOTAL	90453, \$

* ENERGY DELIVERED

ENERGY DELIVERED 8682165, KWH

ENERGY COST PER KWH	19.3 MILLS
---------------------	------------

VALUE OF ENERGY DELIVERED (VALUE OF FUEL SAVED)	105440 \$
--	-----------

ENERGY VALUE PER KWH 22.5 MILLS

COST PER VALUE DELIVERED .86

* LOAD FACTOR

PERCENT OF LOAD SUPPLIED BY TOTAL WIND SYSTEM	90.4
--	------

PERCENT OF LOAD SUPPLIED BY UTILITY	9.6
--	-----

PERCENT OF WIND ENERGY SURPLUSSED	9.0
--------------------------------------	-----

COST TO MEET LOAD (WIND + UTILITY)	19.8 MILLS
---------------------------------------	------------

FIGURE 8.2-5 COST MONITOR OUTPUT FOR BATTERY MODEL

BCS 40180-2

W
V
D

581

BATTERY MODEL TEST

FIGURE 8.2-6 WIND PROFILE FOR BATTERY SIMULATION

BATTERY MODEL TEST

FIGURE 8.2-7 WIND POWER SUPPLIED TO LOAD

FIGURE B.2-8 BATTERY POTENTIAL ENERGY STORAGE

8.3 FLYWHEEL STORAGE MODEL

Figure 8.3-1 shows a simplified schematic of the flywheel storage model. This model is very similar to that of 8.2 except that flywheel storage replaces battery storage, and a power line loss is included in the model. The input data for this model is shown in Figure 8.3-2. Observe that the components are defined in the order of information flow shown in 8.3-1. The admittance component AD is used to model transmission line power losses. The model schematic is shown in Figure 8.3-3.

MODEL DESCRIPTION	FLYWHEEL TEST CASE	
LOCATION=74	TI	
LOCATION=61	WD	INPUTS=TI
LOCATION=21	WP	INPUTS=WD
LOCATION=42	MAB	INPUTS=WP(P,2=FIN),UT(P,1=C2)
LOCATION=33	PD	INPUTS=MAB(FO=P,0),MAB(FO=WP),PA(1,1),PIB(2,2) FL(RE=RE,2)
LOCATION=13	MO	INPUTS=PD(2,1)
LOCATION=4	TRI	INPUTS=MO(2,1),FL(RS=RS,2)
LOCATION=6	FL	INPUTS=TRI,PA(2,1)
LOCATION=8	TRO	INPUTS=FL,GE(RS=RS,2)
LOCATION=19	GE	INPUTS=TRO
LOCATION=45	PIB	INPUTS=FL
LOCATION=69	PA	INPUTS=GE(2,2),LO(RE,1=RE,0),PIB(4,2) UT(2,3)
LOCATION=78	AD	INPUTS=PA
LOCATION=76	LO	INPUTS=TI,AD
LOCATION=62	UT	INPUTS=PD(1SP=P,0)
LOCATION=1	CM	
END OF MODEL		
LIST STANDARD COMPONENTS		
PRINT		

FIGURE 8.3-2 FLYWHEEL MODEL INPUT DATA

The simulation input data shown in Figure 8.3-4 uses the same wind and load data as Example 8.2. However, the storage component is rated at 400 kw with one hour storage, simulating a system used for temporary storage and discharge during peak power generation and load demand periods. It may be noted that the transmission power loss table is input for both TRI and TRO. Figures 8.3-5 to 8.3-7 show results from the simulation. Charging power to the

FIGURE 8.3-1: FLYWHEEL STORAGE EXAMPLE

FIGURE 8.3-3 FLYWHEEL MODEL SCHEMATIC

flywheel in excess of that needed for the load is shown in Figure 8.3-5. Even with average load demand exceeding wind generation, the flywheel is charged at rated power fairly often. The kinetic energy stored by the flywheel over a two week period is shown in Figure 8.3-6. During the week energy is frequently withdrawn and storage is generally not much above the dead-band (80 kwh), whereas during the weekend the reverse is true. Output from the cost monitor is shown in Figure 8.3-7. The capital costs are probably low since nominal values were used for component costs. The utility supplied nearly 20% of the load in this case, since flywheel storage capacity is quite low.

```

TITLE= FLYWHEEL MODEL TEST
PARAMETER VALUES
VO AD=100,G1 AD=8,,G2 AD=8,,G4 AD=8,,G5 AD=200
SR GE=,008,C2 FL=3,E=8
PR FL=,02,MH FL=3372,RF FL=3,5,SR FL=,4,WT FL=24000,KF FL=1,SE=5
ZE FL=1,NAPFL=400,EO FL=40,E1 FL=400,EDFL=20,CM FL=600,CC FL=500
RS MD=1750,RAPMD=1000,MP1MD=1,,E8,CC MD=500,CM MD=0,
R91TR=1750,CC TR=500,CM TR=0,CC TRD=500,CM TRD=0,
RAPGE=1000,CC GE=1000,CM GE=100,
C2 CM=15,,LE CM=30,
G9 UT=20,,CB UT=,016,CP UT=,03,CC UT=1000,,CM UT=1000,
C1 NAB=1,
CYCLES=4,01,TS T1=0,V WP=400,MVWP=8,MV1WP=40,OLINES=100,
CC MP=1000,CM MP=1200,PSIP1=2,
EC MP=2
NC LO=,005,CT LO=4,MH LO=0,STOLO=6,VE LO=,023
TABLE,PLDTR1=5,4
0,5,1,1,5,1,72
0,400,900,1100,1300
0,10,10,10,5,20
0,10,11,11,5,12
0,10,10,10,5,11
0,6,6,5,7,10
TABLE,PLDTRD=5,4
.5,1,1,5,1,72
0,400,900,1100,1300
0,10,10,10,5,20
0,10,11,11,5,12
0,10,10,10,5,11
0,6,6,5,7,10
TABLE,CLOFL=3,3
-1000, 0, 1000
2000,4000,7000
2,8, 7,4, 15
.9, 2,5, 5
2,6, 7,2, 15
TABLE,CLIFL=3
2000,4000,7000
.6, 2,4, 0
TABLE,PM WP=10
8,10,12,14,16,18,20,21,53,25,30
25,8,50,1,86,5,137,4,205,1,292,,400,6,500,782,8,600
TABLE,PY MD=13
0,,4,33,8,67,13,,17,33,21,67,24,,30,33,34,67,30,,43,33,47,67,52,
65,67,68,65,61,56,51,49,49,52,56,61,65
TABLE,PD MD=7
0,4,8,12,16,20,24
10,12,14,16,14,12,10
TABLE,DF MD=16
0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15
5,44,160,380,480,512,440,376,307,270,148,76,40,22,9,3
TABLE,PD LD=17
0,1,5,3,4,5,6,7,5,0,10,5,12
13,5,15,16,5,18,19,5,21,22,5,24
450,360,372,330,450,660,810,798,804
600,708,699,702,750,708,570,450
TABLE,PM LD=7
1,2,3,4,5,6,7
1,1,,9,,9,,9,,6,,5
TABLE,PY LD=6
0,10,20,30,40,52
226,194,180,170,194,226
INITIAL CONDITIONS, KE FL=500,
PRINTER PLOTS,DISPLAY1
MV2=0,V3,TIME
P1 PD,V3,TIME
P2 PD,V3,TIME
KE FL,V3,TIME
DISPLAY2
P2 GE,V3,TIME
RE2FL,V3,TIME
RE1LO,V3,TIME
TINC=,25,TMAX=330,,PRATE=8,PRINT CONTROL=3,INT MODE=3,OUTRATE=8
SIMULATE

```

FIGURE 8.3-4 FLYWHEEL SIMULATION DATA

FIGURE 8.3-5 WIND POWER SUPPLIED TO FLYWHEEL STORAGE

FLYWHEEL MODEL TEST

FIGURE 8.3-6 FLYWHEEL KINETIC ENERGY STORAGE

WIND ENERGY STORAGE COST SUMMARY

30 YEAR LIFE CYCLE

YEARLY SYSTEM COSTS

CAPITAL COST
(INCLUDING FIXED CHARGES) 89104.8

FIXED O + M COST 3100.8

OPERATING + FUEL COST 1753.8

TOTAL 93957.8

ENERGY DELIVERED

ENERGY DELIVERED 4440974. KWH

*
* ENERGY COST PER KWH 21.2 MILLS *
*
*****VALUE OF ENERGY DELIVERED 95217.8
(VALUE OF FUEL SAVED)

ENERGY VALUE PER KWH 21.4 MILLS

COST PER VALUE DELIVERED .99

LOAD FACTOR

PERCENT OF LOAD SUPPLIED 61.3
BY TOTAL WIND SYSTEMPERCENT OF LOAD SUPPLIED 19.7
BY UTILITYPERCENT OF WIND ENERGY 8.3
SUPPLIEDCOST TO MEET LOAD 22.3 MILLS
(WIND + UTILITY)

FIGURE 8.3-7 FLYWHEEL MODEL COST MONITOR OUTPUT

8.4 HYDRO AND THERMAL STORAGE MODEL

Figure 8.4-1 shows the basic model schematic for a model with both thermal and electrical loads. Wind power is supplied first to meet the electrical load, with excess power going into hydro and thermal storage. The thermal load is driven by an ambient temperature component and electrical load energy value is supplied by a time dependent look-up table. Figure 8.4-2 shows the model input data. The components are ordered according to the flow of information in 8.4-1. Observe that the maximum power input of the power divider is connected up to the wind power output P. The model schematic is shown in Figure 8.4-3.

LIST STANDARD COMPONENTS

MODEL DESCRIPTION	HYDRO AND THERMAL TEST CASE
LOCATION=77 TI	
LOCATION=51 WD	INPUTS=TI
LOCATION=21 WP	INPUTS=WD
LOCATION=33 PD	INPUTS=WP, WP(P=WP), PA(1,1), PIH(2,2), HS(RE=RE,2) INPUTS=TS(2,3), PIT(2,3)
LOCATION=13 MO	INPUTS=PD(2,1)
LOCATION=15 PU	INPUTS=MO
LOCATION=17 HS	INPUTS=PU, PA(RE,2=RE)
LOCATION=45 PIH	INPUTS=HS
LOCATION=19 HT	INPUTS=HS
LOCATION=40 GE	INPUTS=HT
LOCATION=59 PA	INPUTS=GE(2,2), LO(1,0), PIH(4,2)
LOCATION=78 FU	INPUTS=TI(TD=FIN)
LOCATION=80 LO	INPUTS=TI, FU(FO=VE)
LOCATION=63 TS	INPUTS=TL
LOCATION=52 PIT	INPUTS=TS
LOCATION=67 TP	INPUTS=TI
LOCATION=65 TL	INPUTS=TI, TP
LOCATION= 1 CM	
END OF MODEL PRINT	

FIGURE 8.4-2 HYDRO AND THERMAL MODEL INPUT DATA

The input data for a two week simulation with this model is shown in Figure 8.4-4. CYCLES is equal to 6 in this model for sufficient iterations to attain steady state in the hydro storage subsystem. The hydro system has

FIGURE 8.4-1: HYDRO AND THERMAL STORAGE EXAMPLE

394

BCS 40180-2

FIGURE B.4-3 HYDRO AND THERMAL MODEL SCHEMATIC

```

PARAMETER VALUES
CYLCES=50,TS=10,TF=0.4V WP=400,WPBP=5,WPBP=50,3LINES=100.
C WP=16000,CM WP=1200,PSIPJM=7.5,EC WP=.7,CR CM=13,LE CM=30
41 PU=200,AS HS=3500,40RHS=80,40 HS=80000,CM HS=1000
41 HS=200,40HS=400000,LE HS=30,P2=PD=.5,P3 PD=.5
2APGC=200,45TSE=3600,SR GE=.0333,CC GE=1000,CM GE=120
V LO=.004,CT LO=4,MN LO=0,STLJ=4,AV FU=-1.
2SL=10,TS=10,VO=110,PD=100,LE TS=30,4F4YS=10000
34 TS=.01453,10EFS=2,RS MD=1750,2AP40=200,CC MD=500,CM MD=100
VE TL=.023,VC TL=40,CT TP=12,MN TP=0,ST97=5.
F4)LE,VP WP=10
3,10,12,14,15,19,20,21,53,25,30
23,6,50,1,85,5,137,4,205,1,292,4,400,6,500,782,8,900
F4)LE,PT MD=13
2,4,33,6,57,13,17,33,21,67,26,30,33,34,67,39,43,33,47,67,52.
65,67,58,53,51,55,51,49,49,52,55,61,65
F4)LE,PD WJ=7
2,1,8,12,14,20,24
10,12,14,15,14,12,10
F4)LE,DP MD=15
2,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15
5,14,150,383,480,512,440,376,307,270,148,76,40,22,9,3
F4)LE,4F TS=4
.37879,.025331,.047371,.064072
23,147,147,204
F4)LE,PO LO=17
0,1,5,3,4,5,6,7,8,9,10,5,12
13,5,15,15,5,18,19,5,21,22,5,24
45,1360,772,330,450,660,210,798,504
53,709,599,702,750,708,570,450
F4)LE,PM LJ=7
1,2,3,4,5,6,7
1,1,9,9,9,6,5
F4)LE,2Y LJ=5
2,10,20,10,10,10,10
225,174,190,174,194,226
F4)LE,4TAFJ = 5
2,5,18,18,22,24
.019,.019,.029,.028,.019,.019
F4)LE,TLDTL=4
2,12,58,103
4,2,1,5,14
F4)LE,TJTT=4
0,5,13,24
4,1,1,1,4
F4)LE,PD TP=9
2,5,5,9,12,13,15,21,24
45,45,45,55,52,54,56,48,46
F4)LE,PY TP=3
0,13,25,39,32
40,50,75,65,40
INITIAL CONDITIONS, MA HS=1600000+E TS=600
PRINT9 PL912,DISPLAY1
J2)MD,VS,TIME
21 PD,VS,TIME
4 PU,VS,TIME
DISPLAY2
2 HS,VS,TIME

42 HS,VS,TIME
22)VS,VS,TIME
DISPLAY3
2C LO,VS,TIME
23 PD,VS,TIME
2 TS,VS,TIME
2CLD,VS,TIME
DISPLAY4
L2 TS,VS,TIME
2C TL,VS,TIME
F4)TP,VS,TIME
23 FU,VS,TI FI
F4)VC=-50,194X=336,PRATE=6,PRINT CONTROL=3,INT 403E=3,OUTRATE=4
TITLE =HYDRO AND THERMAL TEST
SIMULATE

```

FIGURE 8.4-4 HYDRO AND THERMAL SIMULATION DATA

much larger capacity and supplies a bigger load than the thermal system in this run. Figures 8.4-5 to 8.4-9 show results of the simulation. Hydro energy storage is shown in 8.4-5. During the week most of the wind energy goes directly to the load except at night. The reservoir builds up to capacity during the week-ends. The cumulative percent load delivered by wind and hydro storage is shown in 8.4-6, and averages about 91%. Similarly, thermal energy stored and percent thermal load delivered are shown in 8.4-7 and 8.4-8. The ambient temperature profile for a similar, one week simulation is shown in 8.4.9.

BCS 40180-2

HYDRO AND THERMAL TEST

77/P6/10. 12.22.50.

SIMULATION DISPLAY 2

CASE NO. 1

FIGURE B.4-5 HYDRO RESERVOIR ENERGY STORAGE

HYDRO AND THERMAL TEST

FIGURE 8.4-6 PERCENT CUMULATIVE LOAD DELIVERED

HYDRO AND THERMAL TEST

BCS 40180-2

77708704. 11:00:57.

SIMULATION DISPLAY 3

CASE NO. 1

598

FIGURE 8.4-7 THERMAL ENERGY STORAGE

FIGURE B.4-8 PERCENT CUMULATIVE THERMAL LOAD DELIVERED

FIGURE 8.4-9 AMBIENT TEMPERATURE SIMULATION OVER ONE WEEK

8.5 PNEUMATIC STORAGE MODEL

Figure 8.5-1 shows the simplified schematic for the pneumatic storage model. For simplicity the motor and generator have been omitted from the pneumatic storage subsystem. A burner is used if needed to heat the exiting air to the turbine. The heat exchanger has a phase change medium. Figure 8.5-2 shows the input data for this model.

MODEL DESCRIPTION	PNEUMATIC STORAGE TEST CASE
LOCATION= 1 TI	
LOCATION=21 WD	INPUTS=TI
LOCATION=51 WP	INPUTS=WD
LOCATION= 5 TP	INPUTS=TI
LOCATION=43 PD	INPUTS=WP, WP(P=WP), PA(1,1), PI(2,2), CS(RE=RE,2)
LOCATION=64 UT	INPUTS=PD(SP=PI)
LOCATION=15 CO	INPUTS=PD(2,1), TP
LOCATION=17 HX	INPUTS=CO, TP, CS
LOCATION=47 CS	INPUTS=HX, PA(RE,2=RE)
LOCATION=36 PI	INPUTS=CS
LOCATION=49 HY	INPUTS=CS, HX
LOCATION=59 BN	INPUTS=HY
LOCATION=80 TU	INPUTS=BN, TP, CS(PR=PS)
LOCATION=76 PA	INPUTS=TU(2,2), LO(1,0), PI(4,2), UT(2,3)
LOCATION=72 LO	INPUTS=TI
LOCATION=71 CM	
END OF MODEL	
LIST STANDARD COMPONENTS	
PRINT	

FIGURE 8.5-2 PNEUMATIC STORAGE MODEL INPUT DATA

The input data for a two week simulation is shown in 8.5-3. In order to keep the air entering the storage cavern from overheating, a fairly large leakage coefficient ($\text{NU} = 0.01$) is assumed. Hence the storage cavern loses about 2/3 of its heat energy every four days. The load constant NC LO can be adjusted to balance wind energy to the load so that weekly air mass flow in and out of the cavern is balanced. The initial values for the CS and HX states were chosen on the basis of an earlier one week simulation. Figures 8.5-4 to 8.5-8 show results of this simulation. Figure 8.5-4 shows the average temperature of the heat exchanger storage medium for the 'cool' cell. The initial

FIGURE 8.5-1: PNEUMATIC STORAGE EXAMPLE

```

TITLE= PNEUMATIC STORAGE TEST CASE2
PARAMETER VALUES
CYCLES=4,01,10,TIME,CT,TP=12,MN,TP=0,STDTP=5,DLINES=100
V WP=400,HVOWP=8,HVWP=60,CC WP=16000,CM WP=1200,PS1PI=2,EC WP=,2
LE CS=30,MDECS=10000,TEMCS=350,NU CS=,010,TM CS=125,BE HX=,001
MD CM=1500,IS,BN=600,LE,BN=30,HDMBN=3000
ST HX=24,LE HX=30,PD HX=150,TMTHX=250,TEMHX=350,L HX=8
MD CS=1500,TIDTU=600,RS TU=3600,CR CM=15,LE CM=30,CM CS=400
NC LO=,0043,CT LO=,MN LO=,STDLO=,VE LO=,023
CB UT=,019,HP1UT=1,EB,CP UT=,023,CC UT=,CM UT=
TABLE,PW WP=10
0,10,12,14,16,18,20,21,23,24,30
29,6,50,1,86,5,137,4,205,1,292,,400,6,500,,880,,880,
TABLE,PY WD=13
0,,4,33,8,67,13,,17,33,21,67,26,,30,33,36,67,39,,43,33,47,67,52,
65,67,68,65,61,56,51,49,49,52,56,61,65
TABLE,PD WD=7
0,4,8,12,16,20,24
10,12,14,16,18,12,10
TABLE,DF WD=16
0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15
5,44,160,380,480,512,440,376,307,270,148,76,40,22,9,3
TABLE,PD TP=9
0,3,6,9,12,15,18,21,24
46,45,48,55,62,62,56,48,46
TABLE,PY TP=5
0,13,26,39,52
40,50,75,65,40
TABLE,PD LO=17
0,1,5,3,4,5,6,7,8,9,10,5,12
13,5,15,16,5,18,19,5,21,22,5,24
450,360,372,330,450,660,810,798,804
690,708,699,702,750,708,570,450
TABLE,PW LO=7
1,2,3,4,5,6,7
1,1,,9,,9,,9,,6,,6
TABLE,PY LO=6
0,10,20,30,40,52
226,194,180,174,194,226
INITIAL CONDITIONS,E CS=1250,MS CS=5,ES,EC1HX=1300,EC2HX=800
PRINTER PLOTS,DISPLAY1
M CO,V8,TIME
T2 CO,V8,TIME
T2 HX,V8,TIME
TS1HX,V8,TIME
P2 UT,V8,TIME
DISPLAY2
E CS,V8,TIME
M8 CS,V8,TIME
T2 CS,V8,TIME
M2 HY,V8,TIME
T HY,V8,TIME
DISPLAY3,P2 TU,V8,TIME,TS2HX,V8,TIME
TINC=,5,TMAX=336,,PRATE=6,PRINT CONTROL=3,INT MODE=3,OUTRATE=4
SIMULATE

```

FIGURE 8.5-3 PNEUMATIC STORAGE SIMULATION DATA

PNEUMATIC STORAGE TEST CASE2

FIGURE 8.5-4 AVERAGE TEMPERATURE IN HEAT EXCHANGER CELL 2

FIGURE 8.5-5 HEAT EXCHANGER OUTLET TEMPERATURE (CHARGING)

BCS 40180-2

PNEUMATIC STORAGE TEST CASE2

FIGURE 8.5-6 AIR MASS IN PNEUMATIC STORAGE

PNEUMATIC STORAGE TEST CASE2

FIGURE 8.5-7 AIR MASS TEMPERATURE IN PNEUMATIC STORAGE VESSEL

PNEUMATIC STORAGE TEST CASE2

FIGURE B.5-8 HEAT EXCHANGER OUTLET TEMPERATURE (DISCHARGING)

temperature at the beginning of the week is a little too cool since the temperature rises to about 400° during the weekends. Phase change in this medium is indicated by the constant temperature intervals at 250° . Figure 8.5-5 shows the air temperature exiting from the heat exchanger into the cavern. During the week this temperature is generally held below 200° but may exceed 350° during the weekend. Figure 8.5-6 shows the air mass stored in the cavern. In this simulation wind power generated exceeded that of the load and thus there is a gradual buildup of air mass in the cavern. The temperature of the stored air mass is shown in Figure 8.5-7. There is about a 10° fluctuation in temperature each week in this case. The last figure, 8.5-8 shows the air temperature exiting from the heat exchanger to the burner. Neglecting the influence of the initial conditions, the average temperature is about 550° and thus a burner is probably not required for this system.

*USGPO: 1978 - 757-139/6320 Region 5-1)