DOE/NASA/0042-79/2
NASA CR 159602
BCS 40180-2 Rev.

(NASA-CR-159602) AN EXPANDED SYSTEM SIMULATION MODEL FOR SOLAR ENERGY STORAGE (UNIVAC OPERATION MANUAL REVISIONS), VOLUME 2 Final Report (Boeing Computer Services, Inc., Seattle, Wash.) 198 p HC A09/MF A01 G3/61 N79-33882 Unclas 35948

AN EXPANDED SYSTEM SIMULATION MODEL FOR SOLAR ENERGY STORAGE (UNIVAC Operation Manual Revisions)

Volume II

A. W. Warren
Energy Technology Applications Division
Boeing Computer Services Company

August 1979

Prepared for
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Lewis Research Center
Under Contract DEN3-42

for
U.S. DEPARTMENT OF ENERGY
Division of Energy Storage Systems

DOE/NASA/0042-79/2
NASA CR 159602
BCS 40180-2 Rev.

AN EXPANDED SYSTEM SIMULATION MODEL FOR SOLAR ENERGY STORAGE (UNIVAC Operation Manual Revisions)

Volume II

A. W. Warren
Energy Technology Applications Division
Boeing Computer Services Company
Seattle, Washington 98124

August 1979

Prepared for
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Lewis Research Center
Cleveland, Ohio 44135
Under Contract DEN3-42

for
U.S. DEPARTMENT OF ENERGY
Division of Energy Storage Systems
Washington, D.C. 20545
Under Interagency Agreement EX-76-A-31-1026

OPERATIONS MANUAL

Revision Pages

FOREWORD to the Second Edition
Table of Contents
List of Figures
List of Tables
List of References

The revision pages numbered iii - xvi replaces pages iii - xii of the original document.

FOREWORD to the Second Edition

This document is the second edition of the SIMWEST operating manual. The SIMWEST program described in the first edition was capable of modeling total wind energy storage systems. This edition also includes a description of recent enhancements to the program which give it the capability to model solar photovoltaic systems. These enhancements were developed under NASA contract DEN3-42 "An Expanded System Simulation Model for Solar Energy Storage." The principal investigator for this contract was Dr. A. W. Warren. Co-investigators were Dr. Y. K. Chan and Dr. M. H. Dwarakanath. This program was conducted under the sponsorship of the Division of Energy Storage Systems, DOE, 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 and Mr. R. H. Beach as project managers.

TABLE OF CONTENTS

Table of Contents (Continued)

* Added or revised components (1979)

Table of Contents (Continued)

* Added or revised components (1979)

Table of Contents (Continued)

* Added or revised components (1979)

LIST OF FIGURES

LIST OF FIGURES (continued)

LIST OF FIGURES (continued)

LIST OF TABLES

PRECEDING PAGE BLANK NOT FILMED

LIST OF REFERENCES

[1] R. W. Edsinger, Y, K. Chan, A. W. Warren, "A Simulation Model for Wind Energy Storage Systems," Volume I: Technical Report, NASA CR-135283, prepared by Boeing Computer Services Company, August 1977.

[2] A. W. Warren, R. W. Edsinger, J. D. Burroughs, "A Simulation Model for Wind Energy Storage Systems," Volume II: Operation Manual, NASA CR-135284, prepared by Boeing Computer Services Company, August 1977.

[3] SOLMET User's Manual, TD-9724, National Climatic Center, Ashville, North Carolina, December 1977.

[4] J. K. Linn, "Photovoltaic System Analysis Program--SOLCEL," Sandia Laboratories Report SAND77-1268, August 1977.

[5] AFFDL-TR-75-(D180-18575), "ECS Transient Analysis"; Vol. I - Component Mathematical Models; Vol. II - Component Transient Computer Programs; Vol. III - Integrated Transient Computer Program; and Vol. IV - Optimal Control and Design Synthesis; The Boeing Company, prepared under Air Force Contract F33615-74-C-3041, October 1975.

[6] A. W. Warren, "An Expanded System Simulation Model for Solar Energy Storage," Volume I: Technical Report, NASA CR-159601 prepared by Boeing Computer Services Company, August 1979.

Revision Pages

Sections 1.0 - 1.5

Revision pages 1 - 16C replaces pages 1 - 16 of the original document.

1.0 INTRODUCTION

Energy storage systems for the utilization of intermittent power sources have received increased study over the past few years. The analysis of storage requirements for optimal utilization of solar-derived energy systems and the total cost of the resulting generator/storage system are often evaluated in such studies. 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 a wind or solar-photovoltaic energy system for a given application and to allow the resulting system to be evaluated and verified through 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 version described in this document is for use on the UNIVAC 1100 series of computers.

The simulation program has proven to be efficient and versatile for performing parametric studies. It has a unique capability for simulating total wind/solar systems containing any one or combination of the above types of storage and at the same time has the flexibility and depth required to perform thorough and meaningful parameter studies.

1.1 SIMWEST OVERVIEW

SIMWEST consists of two basic programs, and a library of generation, storage, environmental, and load components. The first program, the Model Generation Program, is a precompiler which generates computer models (in FORTRAN) of complex energy generation/storage systems, from user specifications using SIMWEST library components. The second program utilizes the resulting computer model to perform cost and power utilization analysis. It handles input, output, integration of system dynamics, and iterates to obtain convergence of implicit variables. The combination of these two programs provides a powerful tool for analyzing alternate generation and storage system designs.

Figure 1.1-1 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 as a number of subprograms, it can be executed as a single batch program by supplying the model description cards and the control cards describing the desired analysis to be performed and the desired tabular and/or plotted output.

The SIMWEST model generation and simulation programs have a number of user oriented features which greatly enhance the value of the codes. Some of the more prominent features are shown in Table 1.1-1. These features and the supplemental components described in 1.2 enable the user to quickly build, debug, simulate and interpret alternative system designs.

1.2 SIMWEST LIBRARY

The SIMWEST library is listed in Table 1.2-1. It is made up of six types of components: environmental, generation, load, logical, storage and supplemental. The two character mnemonic names are used to identify components in the users model.

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 system cost effectiveness. Second, the models should be designed to minimize computer time and required user specification. It is assumed that a SIMWEST simulation might cover a time span of one year. Thus, from a computer run time and economic impact point of view a simulation step size of between 15 minutes and one hour was established as a design goal.

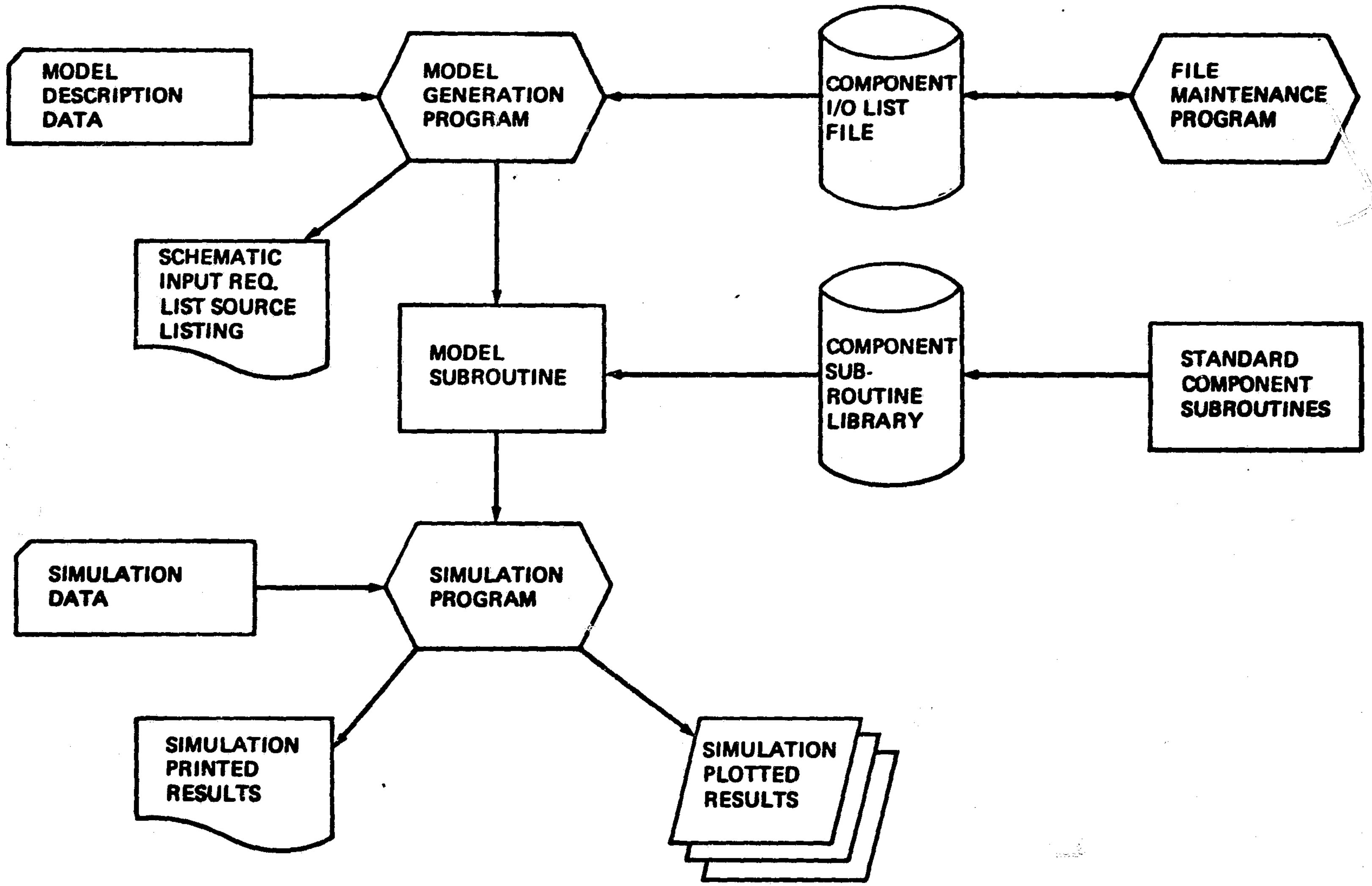


Figure 1.1-1 SIMWEST Program Organization

Table 1.1-1 SIMWEST User Oriented Features

MODEL GENERATION PROGRAM

- Simplified Component Connections
- Availability of all Input Parameters for Connection
- Fortran Insertion Capability Between Components
- Line Printer Schematic of User's Model Provided
- Automated Naming of Parameters and Variables
- Built-in Diagnostic Capabilities

SIMULATION PROGRAM

- Free Field Data Inputs, Including Tables
- Diagnostics on Data Inputs
- Default Values Assigned to Unspecified Parameters
- Optional Levels of Line Printer and Diagnostic Output
- Multiple, Back-to-back Simulation Capability
- Printer Plotter Output of Time Histories and Crossplots

As a result of the above design criteria, many physical components, such as the electrical components, were modeled mainly in terms of power flow and steady state response. This level of detail is consistent with a 15 minute time step and with the concept that 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 time transients were to be modeled, additional detail would be required in 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 and computer runs would be much costlier.

The environmental components listed in Table 1.2-1 simulate environmental conditions. In the present SIMWEST library a user can generate wind speed and ambient temperatures, or can use selected inputs from the recorded weather and insolation data on the Typical Meterological Year (TMY) tapes for one of 26 U.S. locations. These variables are generally used as inputs to physical components.

The generation components consist of wind generation, solar-photovoltaic and utility routines. The wind turbine-generation components are fairly simple models for computing the power output of a conventional, horizontal axis wind machine given basic machine parameters. The solar-photovoltaic components are somewhat more sophisticated, especially in the collector thermal analysis, and have a number of modeling options which a user may employ, e.g., active or passive cooling.

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

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

Table 1.2-1 SIMWEST Library Components

ENVIRONMENTAL	
WIND	WD
AMBIENT TEMP	TP
TMY WEATHER TAPE	ED
WIND POWER GENERATION	
TURBINE/GENERATOR	WP
WIND TURBINE	WT
FIXED RATIO TRANSMISSION	GR
AC GENERATOR	GE
SOLAR POWER GENERATION	
SOLAR ORIENTATION (TRACKING)	SO
FLAT PLATE COLLECTOR	FP
FOCUSING LENS COLLECTOR	FO
PHOTOVOLTAIC ARRAY	PV
UTILITY GENERATION	
UTILITY	UT
LOGIC	
POWER DIVIDER	PD
POWER ACCUMULATOR	PA
PRIORITY INTERRUPT	PI
SWITCHES	SW,SX SY,SZ
LOAD	
ELECTRICAL LOAD	LO
THERMAL LOAD	TL
BATTERY STORAGE	
INVERTER	IV
RECTIFIER	RE
BATTERY	BA
ADMITTANCE	AD
FLYWHEEL STORAGE	
AC MOTOR	MO
VARIABLE RATIO TRANSMISSION	TR
FLYWHEEL/CLUTCH	FL
HYDRO STORAGE	
HYDRO PUMP	PU
HYDRO TURBINE	HT
HYDRO STORAGE	HS
PNEUMATIC STORAGE	
COMPRESSOR	CO
TURBINE	TU
ADIABATIC HEAT EXCHANGER	HX,HY
BURNER	BN
PNEUMATIC STORAGE	CS
THERMAL STORAGE	
STORAGE VESSEL	TS
SUPPLEMENTAL	
SATURATION	SA
RANDOM NUMBER GENERATOR	RN
TEST FUNCTIONS	AF
TABLE LOOKUPS	FU,FV
TRANSFER FUNCTIONS	IT,LA,LL,TF
ARITHMETIC ELEMENTS	MA,MB,MC
COST MONITOR	CM
HISTOGRAM	HG
TAPE READ	TA
TIME CONVERSION	TI

simulated demand and compute the value of the energy delivered 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 or logic devices could be built to serve the function of the logical components, they are not meant to represent any particular existing hardware. Instead, they are idealized components that allow the user flexibility in modeling a wide variety of system and control logic for operational evaluation of energy storage systems. In practice, the control function might be performed by a control room operator using a predefined control strategy or by use of a process computer.

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

1.2.1 Storage Subsystems

Figures 1.2-1 through 1.2-5 give example configurations of the five types of storage subsystems which can be modeled with the present SIMWEST 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 1.2-2.

A total energy system will generally be made up of elements from a number of different subsystems (see Figure 1.2-6). In addition, the SIMWEST program can be used for models which include networks of storage subsystems of the same type or a network of wind or solar generators.

1.2.2 Logic Components

The capability for modeling complex system control logic is provided by the power divider, power accumulator and priority interrupt components. Both

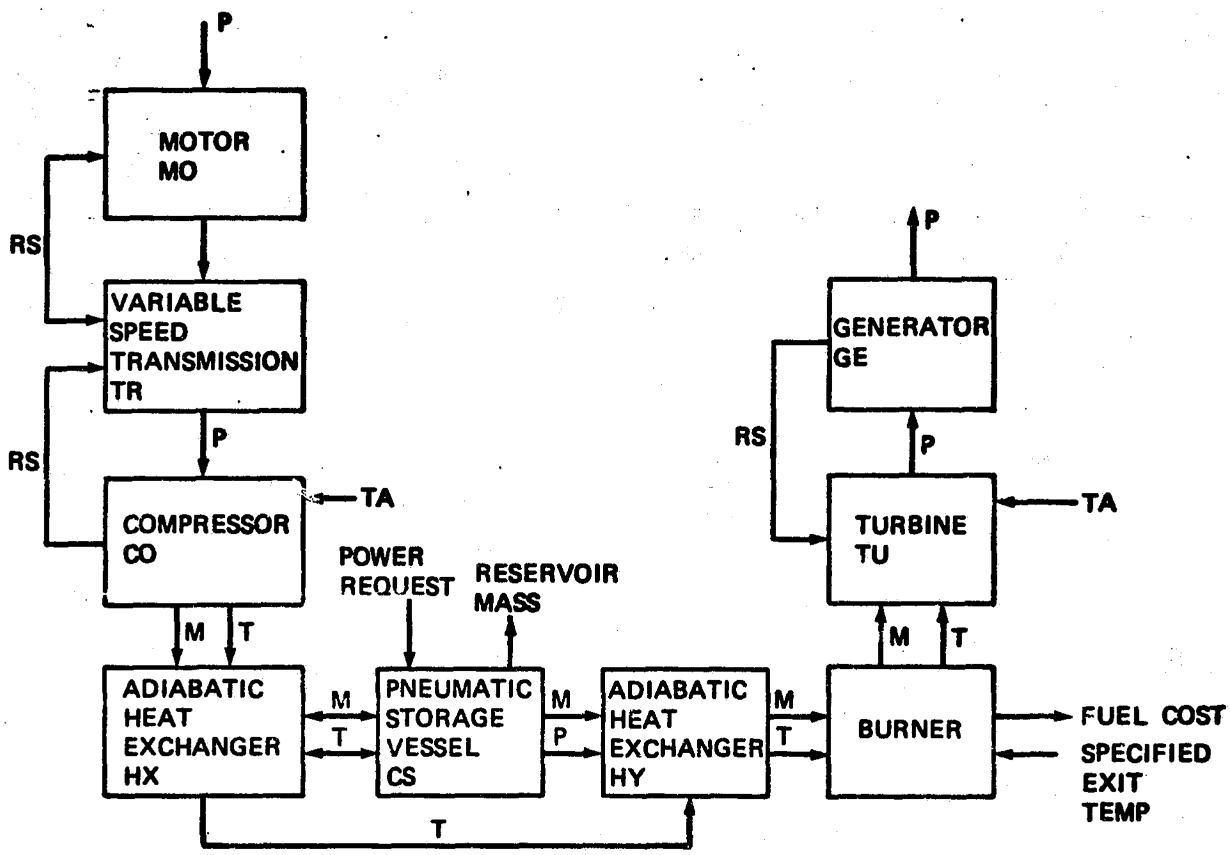


Figure 1.2-1 Pneumatic Storage Subsystem

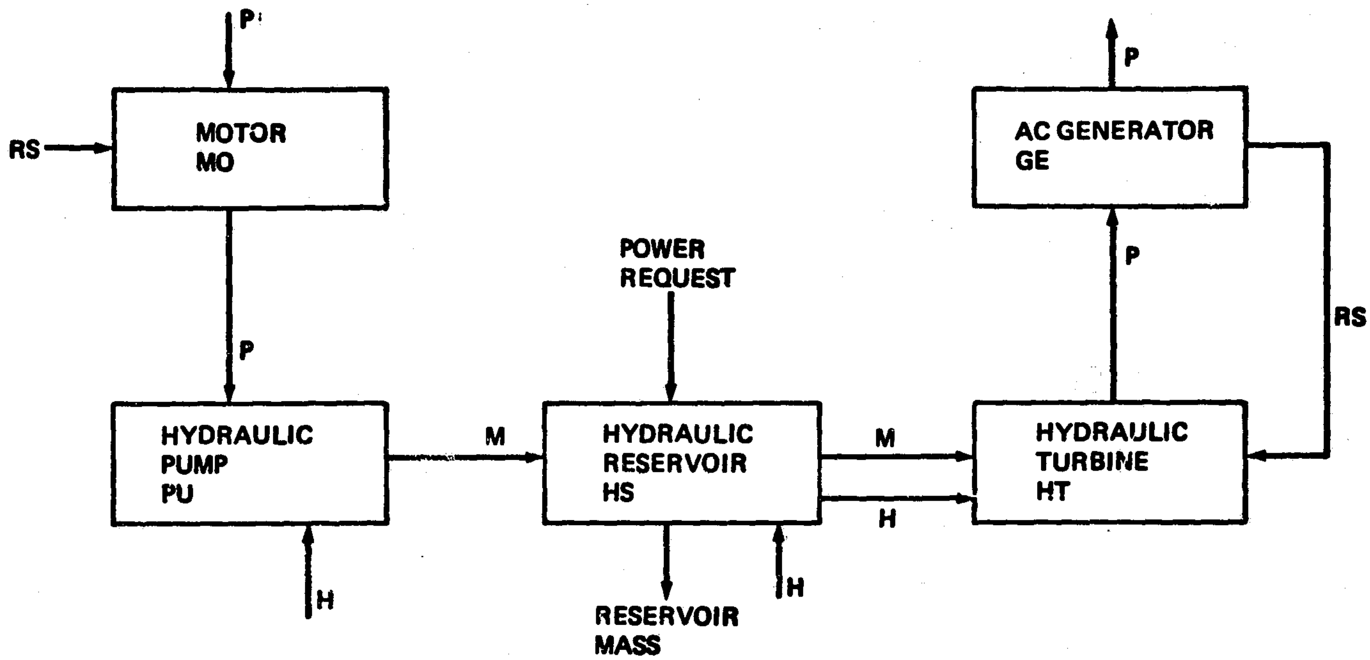


Figure 1.2-2 Pumped Hydro Storage Subsystem

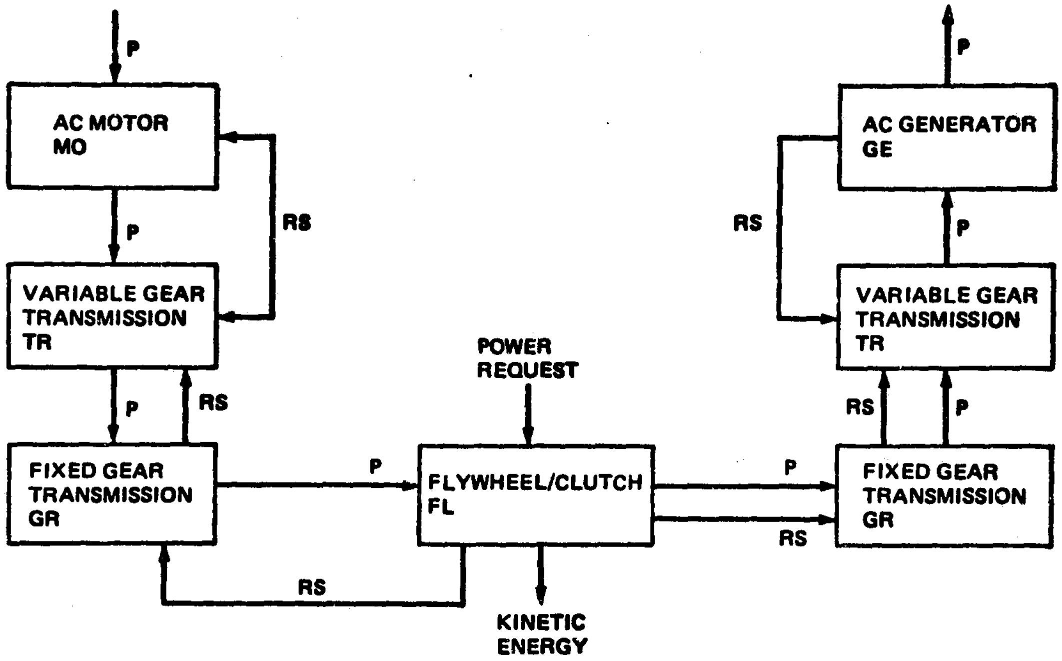


Figure 1.2-3 Flywheel Storage

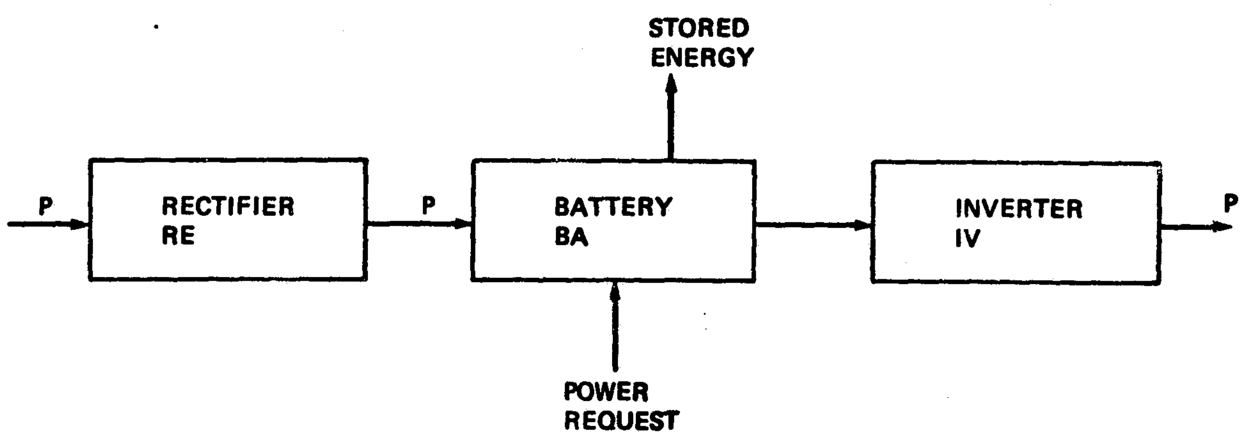


Figure 1.2-4 Battery Storage

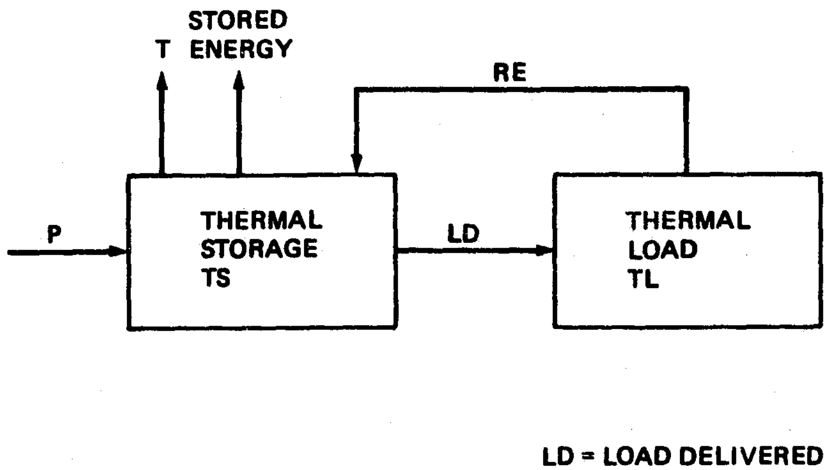


Figure 1.2-5 Thermal Storage

Table 1.2-2 Partial List of Component Inputs and Outputs

	SYMBOLS
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
PR	PRESSURE
PS	PRIORITY SEQUENCE
WY	WEEK OF YEAR
DW	DAY OF WEEK
TD	TIME OF DAY
SP	SURPLUS POWER

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 an input request associated with each of its output ports. A power request originates with a component which is directly or indirectly connected to an output port. 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 exceeds that requested of the port with highest priority (priority 1) then the excess power goes to the port with the next 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 does not receive power. Such ports are included to model backup or switch operated components. In these situations, the connected component would 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 among four output ports, it accumulates power from four input ports and sends it out through a single output port. The power accumulator accepts power requests from the downstream component and allocates requests to each of its input ports in order to service the downstream component.

An example illustration of the use of power dividers and power accumulators is given in Figure 1.2-6. It is seen that power from the turbine/generator is distributed with highest priority (priority 1) going to the power accumulator that services load 1. Since the power accumulator servicing load 1 has its priority 1 input port connected to the power divider, it will try first to satisfy load 1 from the turbine/generator and then from the utility. If the power divider satisfies load 1 and there is power left

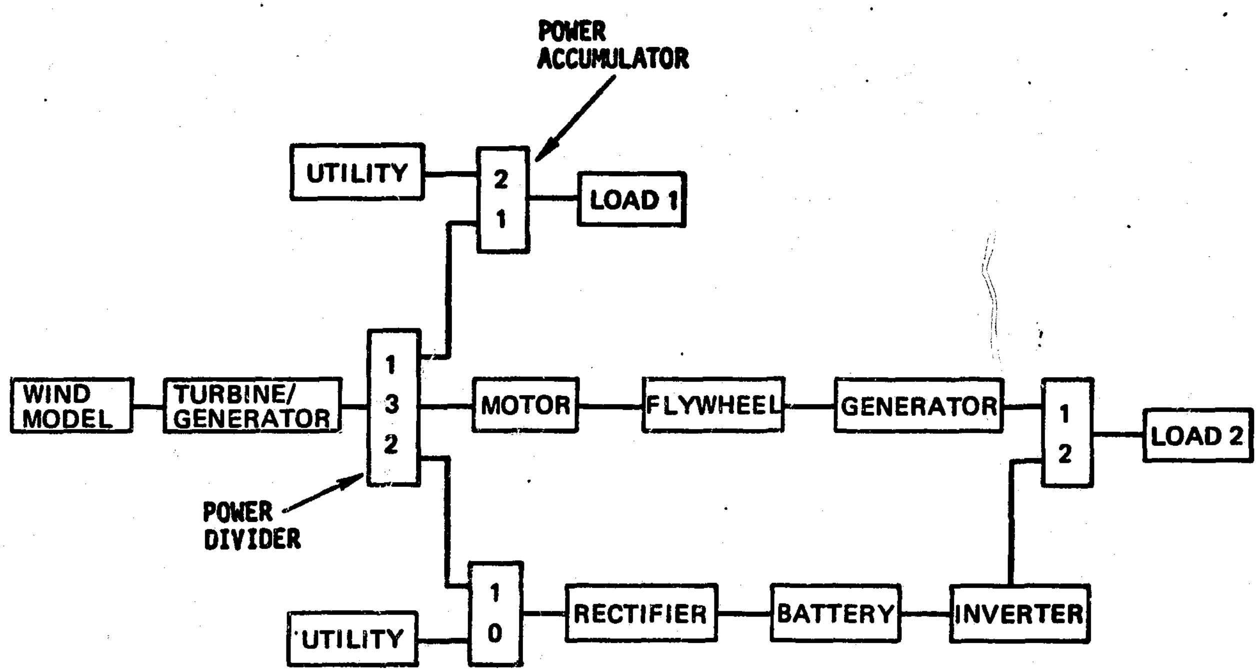


Figure 1.2-6 Example of Power Divider and Accumulator Use

over, it will be used 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 also has a priority zero connection to the utility. 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 needed power.

Also in Figure 1.2-6, 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.

1.3 SIMWEST OUTPUT

There are three basic forms of SIMWEST output to facilitate the analysis of wind and solar energy storage systems; line printer plots, histograms of system variables and time sequenced output of variable values. Each SIMWEST library component is associated with a number of output variables. Prior to simulating a given system the user may select any of these outputs for plotting or tabular output. For example, he may want to plot the energy of pneumatic storage as a function of time and/or as a function of temperature. If the user wants a time sequenced listing of all variable values, he may specify the time step between printouts. The listing of all variables has proven to be a useful tool in understanding the performance of the system under consideration and a valuable aid in validating the system design.

SIMWEST also provides a special output which computes life cycle and levelized energy costs per kwh. This output is produced by the cost monitor component and is illustrated in Figure 1.3-1. The levelized energy costs are based on energy delivered to the loads during a simulation and forecasted to a full years' system operation. This output permits direct comparison of capital and energy costs for alternative system configurations, enabling a user to perform economic trade studies and system sizing.

SOLAR/WIND ENERGY STORAGE COST SUMMARY

20 YEAR LIFE CYCLE

- YEARLY SYSTEM COSTS

CAPITAL COST (INCLUDING FIXED CHARGES)	526. $
FIXED O + M COST	107. $
OPERATING + FUEL COST	14. $
TOTAL	646. $

- ENERGY DELIVERED

ENERGY DELIVERED	7445. KWH
ENERGY COST PER KWH	86.8 MILLS
VALUE OF ENERGY DELIVERED (VALUE OF FUEL SAVED)	372 $
ENERGY VALUE PER KWH	50.0 MILLS
COST PER VALUE DELIVERED	1.74

- LOAD FACTOR

PERCENT OF LOAD SUPPLIED BY TOTAL SOLAR SYSTEM	100.0
PERCENT OF LOAD SUPPLIED BY UTILITY	0.0
PERCENT OF SOLAR ENERGY SURPLUSED	0.0
COST TO MEET LOAD (SOLAR + UTILITY)	86.8 MILLS

Figure 1.3-1 Cost Monitor Output For Fresnel Lens Model

1.4 TESTING

Reference [1] describes two simulation studies which were used to test the original SIMWEST program. Reference [6] describes the NASA-Lewis approved simulation studies for the expanded SIMWEST program. These studies provide an excellent test and illustration of the program's capability to model complex wind/solar energy systems.

Prior to performing the simulation studies and throughout its development the SIMWEST program was systematically tested. First components were grouped into simple systems and simulations were performed. During these simulations system parameters were driven so as to force the individual components through every normal program path and to assure that all component outputs assume a wide range of values. The number of components and the number of ways they can be connected makes it impossible to exercise every combination. However, the subsystem groupings that were used were representative of the expected program usage. Sections 8 and 9 describe some of the test cases for the wind and solar-photovoltaic generation components.

In terms of computer efficiency, it was found during the testing that the program exceeded original expectations. Even on very complex systems, such as represented by the NASA-Lewis test case, convergence of logic variables was quite rapid. Convergence generally took place in less than six iterations per simulation time step. As an example, the year simulations used in the NASA defined parameter study of reference [1] took less than 420 CPU seconds on the CDC 6600. For comparison, the CPU time on the UNIVAC 1100/40 is approximately two to three times as great as that on the 6600, and CPU time on the Cyber 175 is a factor of two to three times smaller than that of the 6600.

1.5 PROGRAM USAGE

While the user need not be a SIMWEST expert or software specialist to make efficient use of the program, he should thoroughly think through and be familiar with the characteristics of the system he wants to simulate.

Component models, if not carefully specified, may perform in unexpected ways. If the systems logic is not well thought out, the resulting system may be significantly out of balance and subsystems may not be fully utilized. The test case described in reference [6] illustrates the process of sizing and logic adjustment to satisfy system performance objectives.

A number of useful procedures were developed during the simulation studies. First it was found that when simulating a complex system, it is best to separately develop and test subsystem portions of the model. This allows problems or unexpected results to be isolated and understood prior to the introduction of the more complex characteristics associated with the total system.

It was found during the simulations that the use of Fortran statements in the model definition is very useful for creating special input to system components and for defining special outputs to be plotted or statistics to be printed. For example, Fortran statements enable the user to generate and interpret trade study data by computing component input parameters from user specified system parameters. The use of Fortran statements is simple and should be encouraged early in SIMWEST applications.

Computer simulation costs may be minimized by appropriate tradeoffs between run time and simulation accuracy. Run time is most directly affected by the integration step size, the total simulation length, and the average number of iterations through the model at each time step. For long duration runs, an hour step size is usually acceptable. Models having smaller time constants than the step size may be approximated by implicit steady state conditions and solved by iteration through the model. If a model requires many iterations for convergence then it may be useful to isolate the source of instability in order to modify or simplify that portion of the system model. It has been generally found in the simulation studies that use of a few seasonal weekly simulations is adequate to predict long term performance for system trade studies and design optimization. Based the results of [6], four to six week long simulations are recommended this purpose.

When making a year simulation run, it is best to break it into twelve monthly simulations. Thus, measures of performance such as plots, histograms and performance statistics are available on a monthly basis. In addition to giving better visibility of the system performance, this helps limit the job core size. The twelve monthly simulations can be submitted as a single run with the results of a given month acting as initial conditions for the next month. The user only needs to submit new data cards for data which changes from one month to the next.

Revision Pages

Section 3.6

Replaces pages 49 - 52 of the original document.

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 maximum number of iterations through the model is equal to CYCLES + 1. If cycles is positive, but not an integer, then the maximum 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 models tested require no more than six iterations per time step to attain steady state. A complex model with cascaded logic components may require more.

Each of the model output variables are monitored each pass for convergence. If all of the outputs are converged within 3% of their previous values, then one final pass is made through the model. Otherwise, all variables exceeding 5% of their previous value are printed out after the last iteration.

Since output statistics are only updated the last iteration, some 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 component. 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 = 50 is sufficient to catch most simulation errors per run.

```
TS STORAGE TEMPERATURE        59,899     OUTSIDE MINIMUM       60,000    AND MAXIMUM     212,000

TS STORAGE TEMPERATURE        59,731     OUTSIDE MINIMUM       60,000    AND MAXIMUM     212,000

         TIME=    88,50
         P2 HT   NONCONVERGENCE, OLD VALUE=      31,913   NEW VALUE=      30,309
         P2 GE   NONCONVERGENCE, OLD VALUE=      30,638   NEW VALUE=      29,098
         PL GE   NONCONVERGENCE, OLD VALUE=       1,275   NEW VALUE=       1,211

HS RESERVOIR VOLUME   77210,404    DROPED BELOW MINIMUM   80000,000

TS STORAGE TEMPERATURE        58,664     OUTSIDE MINIMUM       60,000    AND MAXIMUM     212,000

TS STORAGE TEMPERATURE        58,964     OUTSIDE MINIMUM       60,000    AND MAXIMUM     212,000

TS STORAGE TEMPERATURE        59,936     OUTSIDE MINIMUM       60,000    AND MAXIMUM     212,000
```

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 names 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: [,], equals [=], left parenthesis [(], 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

Revision Pages

Section 4.1

Replaces pages 55 - 58 of the original document.

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.

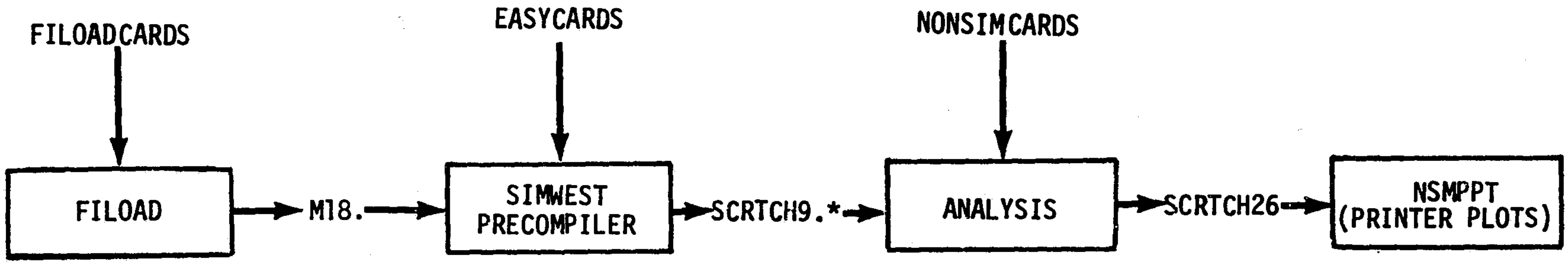
4.1 MODEL GENERATION AND ANALYSIS EXECUTION

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.
```


PRECEDING PAGE BLANK NOT FILMED



*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
          .
          .
          .
ASG,T 2.U9B., Reel No.*
@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-

*Used whenever TMY environmental tape data is to be input.

```
@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

Revision Pages

Section 5.2

Replaces pages 67 and 68 of the original document.

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

ED: INPUT ERROR, DAY OF YEAR DY IS OUT OF RANGE
TAPE INPUT ERROR OR EOF

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

PV: WARNING: INSOLATION OR TEMPERATURE AT CELL EXCEED RANGE

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

Revision Pages

Section 7.0

Replaces pages 89 - 92 of the original document.

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, a description of the calculation sequence and logic used in the model, and the model code. 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 are separated by a comma. For example, the power variable with port designation 1 is denoted P,1. 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 Ø to distinguish this symbol from a zero. Elsewhere in the text symbols such as VØ 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

PRECEDING PAGE BLANK NOT FILMED

a default value is not so specified have 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 for FU is specified in the simulation input.

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 = input maximum power.

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

7d. UNITS

Most of the SIMWEST components are coded in English units. However, SI or metric units were used to code the solar-photovoltaic components: ED, SO, FP, FO, and PV. This is generally not a problem since there are at most only a few interconnection variables between the solar-photovoltaic generation components and other SIMWEST components, and units conversions are easily handled using an MA arithmetic component. (See for example the Fresnel Lens Model, section 9.3.)

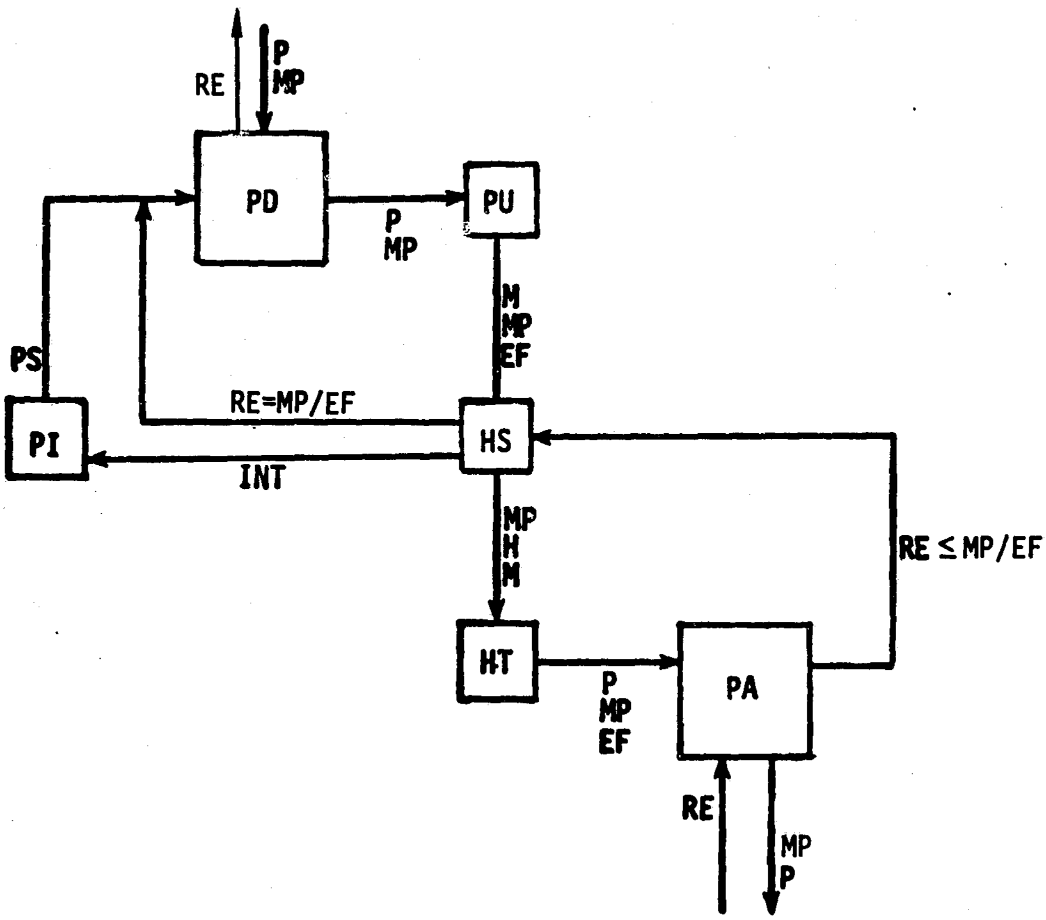


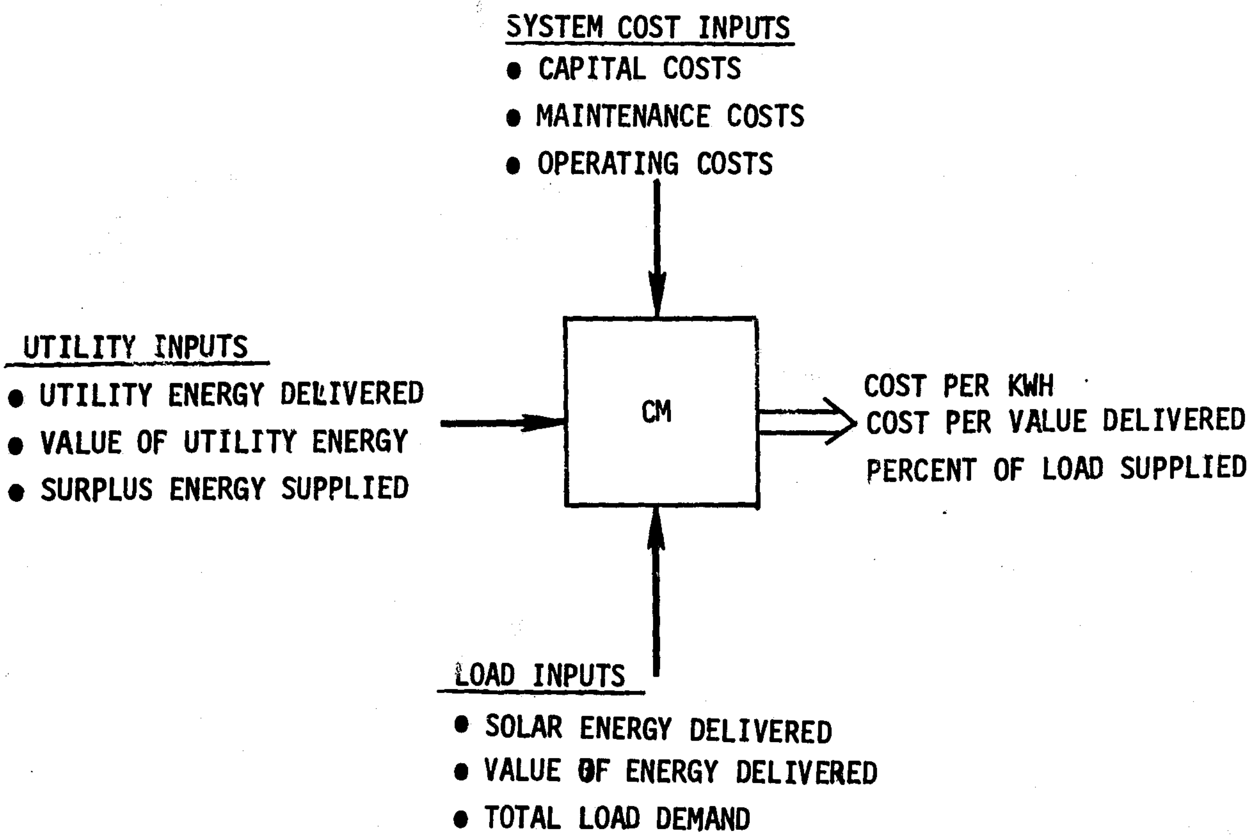
FIGURE 7.0 SAMPLE CONNECTIONS FOR LOGIC COMPONENTS

Revision Pages

Section 7.5

Replaces pages 117 - 122 of the original document.

7.5 COST MONITOR[1]



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.

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

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

then given by

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

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 storage PCW, percent of load supplied by utilities PCU, and percent of energy surplused to the utilities PCS is computed. The total cost in mills to meet the load is then given by

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

where

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

Inputs

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

Common Block Inputs	Description	Units
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	Solar energy delivered (including surplus)	kwh
TLD	Total load demand	kwh
UTV	Value of utility energy	$
UTD	Utility energy supplied	kwh
SPD	Surplus 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	
	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.

ORIGINAL PAGE IS
OF POOR QUALITY

```
SUBROUTINE CM          ENTRY POINT 000213

STORAGE USED: CODE(1) 000226; DATA(0) 000332; BLANK COMMON(2) 000000

 COMMON BLOCKS:

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

EXTERNAL REFERENCES (BLOCK, NAME)

 0007   NWDU$
 0010   NI02$
 0011   NERR3$

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

 0001   000020 100L      0000   000016 200F      0000   000035 300F      0000   000106 400F      0000   000211 500F
 0003 R 000000 CC        0000 R 000003 CCY       0003 R 000001 CMA       0003 R 000002 CO        0000 R 000002 COY
 0000 R 000015 CPKWH     0000 R 000011 CPV       0006   000000 DUM       0000 R 000005 EDE       0004 I 000000 IMPL
 0000   000315 INJP$     0000 I 000006 IVDE      0000 I 000001 LLE       0000 R 000012 PCD       0000 R 000014 PCS
 0000 R 000013 PCU       0003 R 000010 SPD       0003 R 000004 TDE       0005 R 000000 TIME      0003 R 000005 TLD
 0006 R 000007 TMAX      0000 R 000000 TMAX1     0000 R 000004 TOY       0000 R 000007 TOYN      0003 R 000007 UTD
 0003 R 000006 UTV       0003 R 000003 VDE       0000 R 000010 VDEN

00100        COST                                                                  000000
00101              SUBROUTINE CM(DUMM,FCR,LE)                                      000000
00101        C                                                                     000000
00101        C    PURPOSE    SUMMARIZE WIND ENERGE STORAGE COSTS AND LEVELIZED     000000
00101        C               ENERGY COSTS PER KWH.                                 000000
00101        C                                                                     000000
00101        C    WRITTEN BY A.W. WARREN                    VERSION 1, MAY 1977    000000
00101        C                                                                     000000
00101        C    INPUT PARAMETERS                                                 000000
00101        C                                                                     000000
00101        C               FCR  - FIXED CHARGE RATE FACTOR INCLUDING DEPRECIATION,  000000
00101        C                      MONEY COST, INSURANCE, AND TAXES,  PER YEAR    000000
00101        C               LE   - SYSTEM LIFE EXPECTANCY , YEARS                 000000
00101        C               TMAX - SIMULATION TIME, HR                            000000
00101        C               CC   - TOTAL YEARLY CAPITAL COSTS, $                  000000
00101        C               CM   - TOTAL YEARLY MAINTENANCE COSTS, $              000000
00101        C               CO   - TOTAL OPERATING AND FUEL COSTS OVER TMAX, $    000000
00101        C               VDE  - VALUE OF ENERGY DELIVERED OVER TMAX, $         000000
00101        C               TDE  - TOTAL ENERGY DELIVERED OVER TMAX, KWH          000000
00101        C               TLD  - TOTAL LOAD DEMAND OVER TMAX, KWH               000000
```

CM

```
00101      C               UTV  - VALUE OF UTILITY ENERGY SUPPLIED LESS SURPLUS VALUE, $      000000
00101      C               UTD  - TOTAL UTILITY ENERGY DELIVERED, KWH                         000000
00101      C               SPD  - TOTAL SURPLUS ENERGY SUPPLIED TO UTILITY, $                 000000
00101      C                                                                                  000000
00103            COMMON /COST/ CC, CMA,CO,VDE,TDE,TLD,UTV,UTD ,SPD                            000000
00104            COMMON /CIMPL/IMPL  /CTIME/ TIME /CSIMUL/ DUM(7),TMAX                        000000
00105            REAL LE                                                                      000000
00105      C                             INITIALIZATION                                       000000
00105      C                                                                                  000000
00106            IF(IMPL.GT.0)GO TO 100                                                       000000
00110            DUMM=0.0                                                                     000002
00111            CC = 0 .                                                                     000003
00112            CMA = 0.                                                                     000004
00113            CO = 0.                                                                      000005
00114            VDE= 0.                                                                      000006
00115            TDE= 0.                                                                      000007
00116            TLD= 0.                                                                      000010
00117            UTV=0.                                                                       000011
00120            UTD=0.                                                                       000012
00121            SPD=0.                                                                       000013
00122            TMAX1= TMAX*.99999                                                           000014
00122      C                                                                                  000014
00123        100 IF(TIME.LT.TMAX1)RETURN                                                      000020
00125            IF(IMPL.LE.1)RETURN                                                          000026
00125      C                                                                                  000026
00125      C                             COST SUMMARY OUTPUT                                  000026
00125      C                                                                                  000026
00127            LLE = LE                                                                     000040
00130            WRITE(6,200)LLE                                                              000047
00133        200 FORMAT(1H1,35X,39H SOLAR/WIND ENERGY STORAGE COST SUMMARY //                 000055
00133          1 1H ,40X,I2,17H  YEAR LIFE CYCLE )                                            000055
00133      C                                                                                  000055
00134            COY = CO*8760./TMAX                                                          000055
00135            CCY = CC*LE*FCR*.01                                                          000061
00136            TOY = COY + CMA + CCY                                                        000066
00137            WRITE(6,300)CCY,CMA,COY,TOY                                                  000071
00145        300 FORMAT(//// 30X,22H0  YEARLY SYSTEM COSTS/ 1H+,29X,1H+/ 1H-,42X,             000102
00145          1 12HCAPITAL COST,12X,F8.0,2H $ / 1H ,42X, 17H(INCLUDING FIXED ,               000102
00145          2  8HCHARGES) / 1H0,42X,16HFIXED O + M COST, 8X,F8.0,2H $ /1H0 ,               000102
00145          3 42X,21HOPERATING + FUEL COST, 3X,F8.0,2H $ / 1H0,42X,5HTOTAL,                000102
00145          4 19X,F8.0,2H $ )                                                              000102
00145      C                                                                                  000102
00146            EDE  = TDE * 8760./TMAX                                                      000102
00147            IVDE = VDE * 8760./TMAX                                                      000106
00150            TOYN = TOY*1000./ EDE                                                        000120
00151            VDEN = VDE*1000./ TDE                                                        000124
00152            CPV  = TOYN / VDEN                                                           000130
00152      C                                                                                  000130
00153            WRITE(6,400)EDE,TOYN,IVDE,VDEN,CPV                                           000132
00162        400 FORMAT(//// 30X,26H0  ENERGY DELIVERED          / 1H+,29X,1H+ / 1H-,         000144
00162          1 42X,16HENERGY DELIVERED, 7X,F9.3,4H KWH / 1H0,33X,50(1H*) /                  000144
00162          1  1H ,33X,1H*,48X,1H* /                                                       000144
00162          2 1H ,33X,1H*, 8X,19HENERGY COST PER KWH, 7X,F6.1,9H MILLS  * /                000144
00162          2  1H ,33X,1H*,48X,1H* / 1H ,                                                  000144
00162          3 33X,10(5H*****) / 1H0,42X,25HVALUE OF ENERGY DELIVERED,I7,                   000144
00162          4 2H $ / 1H ,42X,22H(VALUE OF FUEL SAVED) / 1H0,42X,20HENERGY VALUE            000144
00162          5 PER KWH, 6X,F6.1,6H MILLS / 1H0,42X,24HCOST PER VALUE DELIVERED,             000144
```


CM

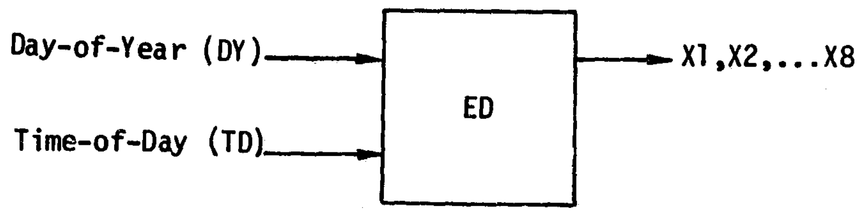
```
10162         6 2X,F6.2)                                                          000144
10162   C                                                                          000144
10163         PCD= (TDE-SPD)*100./TLD                                              000144
10164         PCU= UTD*100./TLD                                                    000150
10165         PCS= SPD*100./TLD                                                    000154
10166         CPKWH= (TOYN*(TDE-SPD) + UTV*1000.)/TLD                              000160
10167         WRITE(6,500)PCD ,PCU,PCS,CPKWH                                       000167
10175     500 FORMAT(//// 30X,31H0  LOAD FACTOR                    / 1H+,29X,      000200
10175         1  1H+ / 1H-,42X,                                                    000200
10175         1 26HPERCENT OF LOAD SUPPLIED  , F6.1, 2H  / 1H ,42X, 28HBY TOTAL S  000200
10175         2OLAR SYSTEM       / 1H0,42X,24HPERCENT OF LOAD SUPPLIED,2X,F6.1 /   000200
10175         2 1H ,41X,11H BY UTILITY  /                                          000200
10175         3  1H0,42X,26HPERCENT OF SOLAR ENERGY    , F6.1 /                    000200
10175         3 1H ,42X,9HSURPLUSED  /                                             000200
10175         4  1H0,42X,23HCOST TO MEET LOAD       , 3X,F6.1,6H MILLS/            000200
10175         5 1H ,42X,29H(SOLAR + UTILITY)           / 1H1)                      000200
10175   C                                                                          000200
10176         RETURN                                                               000200
10177         END                                                                  000225
```

Revision Pages

Section 7.7A - ED

Insert revision pages 140A - 140N between pages 140 and 141 of the original document.

7.7A ENVIRONMENTAL DATA (TMY TAPE)



This component reads data values from the Typical Meteorological Year (TMY) tapes or data with a similar format structure such as the University of Wisconsin insolation and environmental data tape or the SOLMET tapes. Only one ED component is allowed per model. (Unit 2 is reserved for the input tape.) The file structure assumes hourly recorded data with one record or card image per hour of data. Twenty-four hourly records are read into core at a time and linear interpolation is used to obtain the output values at the current simulation time. The component TI is used to supply the time inputs DY and TD. Standard outputs with the TMY tape are direct and global solar insolation, dry bulb temperature, and wind speed. For non-standard outputs or non-TMY format tapes the user may specify the input format to read one to eight data variables. The following limitations apply in this case:

1) Time information is decoded in integer month (1-12), day (1-31), and hour (0-24) format.

2) Output variables are decoded in F or E format, even if recorded in integer format.

3) Where data is missing, fill in with 9's is assumed. The code checks for certain 9 fill values, namely 99., 999., 9999., and 99999. If any one of these values is read, then the corresponding data input is replaced with 0. or the previous value, depending on the sign of IND. (However, one must use FN.0 format N=2,3,4,5 for this option and a scale multiplier if necessary to obtain the desired exponent.)

Inputs/Port	Description	Units
NST	Number of tape blocks to skip at start[1]	-
NX	Number of output variables (default = 4, max = 8)	-
IND	Indicator function: 0 = no read ±1 = standard format and units (default) ±2 = user-specified format and units IND>0 sets missing data = 0 IND<0 sets missing data = previous value	

---

[1] For the TMY tapes we may compute NST from the station number (NSTAT) shown in table 7.7A and the start day (DSTART):

NST = (NFILE-1)*365 + DSTART-1

where NFILE = $\begin{cases} \text{NSTAT if NSTAT} \le 13 \\ \text{NSTAT-13} \quad \text{otherwise} \end{cases}$

Inputs/Port (cont'd)	Description	Units
TS*	Time shift of data (default = -0.5)	hours
TD	Time of day (0-24)	hours
DY	Day of year (1-365)	-
M1	Units multiplier for X1 (default = 1)	-
.	.	
.	.	
.	.	
M8	Units multiplier for X8 (default = 1)	-
A1	Units addition factor for X1 (default = 0)	-
.	.	
.	.	
.	.	
A8	Units addition factor for X8 (default = 0)	-

*Compensation term since solar radiation data is an integrated total over the observation interval.

ED

Outputs/Port	Description	Units
X1	1st output variable (IND = ±1: beam radiation in w/m^2	-
X2	2nd output variable (IND = ±1: global horizontal radiation in w/m^2)	-
X3	3rd output variable (IND = ±1: dry bulb temperature in ^oC)	-
X4	4th output variable (IND = ±1: wind speed in m/s)	-
.		
.		
.		
X8	8th output variable	-

Format Specification

A user-specified format may be input in order to select non-standard environmental outputs or to read a tape other than the TMY insolation tape. The following sequence of data cards is recommended for insertion in the model generation input following the MODEL DESCRIPTION command:

```
      FORTRAN STATEMENTS
      DIMENSION FMT(12)
      COMMON/READER/N,FMT
      DATA FMT/72H...)
     1            /N/NN/
```

where the format specification contains up to 71 characters inserted after '72H' and followed by ')', and NN = the number of characters per data record.

The format specification must conform to the following rules:

1) The first two words read are station and year identifying information. These words must be either A format or nH format with up to six characters for station and two characters NN for year 19NN.

2) The next three words are two-digit integers containing month (1-12), day (1-31), and hour (0-24) information.

3) The next one to eight words specify the location of the output variables X1...X8 and must be given in F or E format.

NOTE: The tab or column spacing control T may be used to read data from files which are not ordered as in 1) to 3), e.g., T71, A5, T1, A2,...).

For example, the standard TMY tape format specification is

Station	Yr-Mo-Dy-Hr	Beam Rad.	Global Rad.	Temp	Wind
A5,	A2,3I2,11X,	F4.0 ,26X,	F4.0,45X,	F4.1,7X,	F4.1)

and N = 132.

The general format for variables on the TMY tape is summarized in Figure 7.7A.

ED

		SOLAR TIME					SOLAR RADIATION OBSERVATION										
	WBAN STN #	YR	MO	DY	HRMN	LST TIME	ETR KJ/m²	RADIATION VALUES KJ/m² DIRECT	DIFFUSE	NET	TILTED	GLOBAL OBS	GLOBAL ENG COR	GLOBAL STD YR COR	A	B	SUNSHINE MIN
	XXXXX	XX	XX	XX	XXXX	XXXX	XXXX	1XXXX	1XXXX	1XXXX	1XXXX	1XXXX	1XXXX	1XXXX	1XXXX	1XXXX	XX
FIELD NUMBER	002	003				004	101	102	103	104	105	106	107	108	109	110	111

SURFACE METEOROLOGICAL OBSERVATION													
OBS TIME LST	CEILING dam	SKY COND	VSBY hm	WEATHER	PRESSURE kPa SEA LEVEL	PRESSURE kPa STATION	TEMP °C DRY BULB	TEMP °C DEW-PT.	WIND DIR deg	WIND SPD m/s	TOTAL	OPAQUE	SNOW COVER
XX	XXXX	1XXXX	XXXX	XXXXXXXX	XXXXX	XXXXX	XXXX	XXXX	XXX	XXXX	XX	XX	X
201	202	203	204	205	206		207		208		209		210

TAPE FIELD NUMBER	RECORD POSITIONS	DESCRIPTION
002	01-05	WBAN STATION NUMBER
003	06-15	SOLAR TIME (YR,MO,DAY,HOUR,MINUTE)
004	16-19	LOCAL STANDARD TIME (HR AND MINUTE)
101	20-23	EXTRATERRESTRIAL RADIATION
102	24-28	DIRECT RADIATION
103	29-33	DIFFUSE RADIATION
104	34-38	NET RADIATION
105	39-43	GLOBAL RADIATION ON A TILTED SURFACE
106	44-48	GLOBAL RADIATION ON A HORIZONTAL SURFACE-OBSERVED DATA
107	49-53	GLOBAL RADIATION ON A HORIZONTAL SURFACE-ENGINEERING CORRECTED DATA
108	54-58	GLOBAL RADIATION ON A HORIZONTAL SURFACE-STANDARD YEAR CORRECTED DATA
109,110	59-68	ADDITIONAL RADIATION MEASUREMENTS
111	69-70	MINUTES OF SUNSHINE
201	71-72	TIME OF COLLATERAL SURFACE OBSERVATION (LST)
202	73-76	CEILING HEIGHT (DEKAMETERS)
203	77-81	SKY CONDITION
204	82-85	VISIBILITY (HECTOMETERS)
205	86-93	WEATHER
206	94-103	PRESSURE (KILOPASCALS)
207	104-111	TEMPERATURE (DEGREES CELSIUS TO TENTHS)
208	112-118	WIND (SPEED IN METERS PER SECOND TO TENTHS)
209	119-122	CLOUDS
210	123	SNOW COVER INDICATOR

Figure 7.7A TMY Tape Format

A complete description of the available data, and the meaning of the recorded outputs, is contained in the SOLMET user's manual [3]. The TMY tape was derived from SOLMET tapes of the 26 stations with rehabilitated solar radiation data, and has the same format as the SOLMET tapes except that tape deck number and detailed cloud data have been omitted. Table 7.7A shows the identity and location of the 26 stations on the TMY tape.

Calculation Sequence

If IND = 0 Return

1) INITIALIZATION (first pass only)

- Set defaults and initialize LTD = -1
- Skip NST blocks to position the file
- Read first data block and write out identification information. (Error exit to 6))
- Go to 4)

2) Table Interpolation for Output (DY = DYF)

- If DY > DYF go to 3)
- If DYF > DY go to 5)
- If LTD = TD return (LTD = last time C(I,J) was accessed)
- X(I) = TBLU1 (TD, TO, C(1,I),0,24)*M(I)+A(I) I = 1,...NX
- LTD = TD
- Return

3) Read One or More Data Blocks (DY > DYF)

- Read DY-DYF data blocks. (Error exit or EOF exit to 6))

4) Decode Using Specified Format

- Decode day-of-year (DYF) and time information (TO) and put output variables in array C(I,J) I=1,24 and J=1,NX. Check for missing data values in C(I,J).
- Go to 2)

5) Backspace the File (DYF > DY)

- Backspace and read first data block
- Decode day-of-year (DYF)
- Go to 4) if DYF ≤ DY. Otherwise print diagnostic and stop.

6) Read Error or EOF Encountered

- Print diagnostic and stop.

ED

TABLE 7.7A TMY TAPE STATIONS AND LOCATION

STATION NUMBER	WBAN IDENTIFIER	STATION	LATITUDE	LONGITUDE
1	3927	Fort Worth, Texas	32°50'	97°03'
2	3937	Lake Charles, Louisiana	30°07'	93°13'
3	3945	Columbia, Missouri	38°49'	92°13'
4	12832	Apalachicola, Florida	29°44'	84°59'
5	12839	Miami, Florida	25°48'	80°16'
6	12919	Brownsville, Texas	25°54'	97°26'
7	13880	Charleston, South Carolina	32°54'	80°02'
8	13897	Nashville, Tennessee	36°07'	86°41'
9	13985	Dodge City, Kansas	37°46'	99°58'
10	14607	Caribou, Maine	46°52'	68°01'
11	14837	Madison, Wisconsin	43°08'	89°20'
12	23044	El Paso, Texas	31°48'	106°24'
13	23050	Albuquerque, New Mexico	35°03'	106°37'
14	23154	Ely, Nevada	39°17'	114°51'
15	23183	Phoenix, Arizona	33°26'	112°01'
16	23273	Santa Maria	34°54'	120°27'
17	24011	Bismarck, North Dakota	46°46'	100°45'
18	24143	Great Falls, Montana	47°29'	111°22'
19	24225	Medford, Oregon	42°22'	122°52'
20	24233	Seattle-Tacoma, Washington	47°27'	122°18'
21	93193	Fresno, California	36°46'	119°43'
22	93729	Cape Hatteras, North Carolina	35°16'	75°33'
23	93734	Washington, D.C.	38°59'	77°28'
24	94701	Boston, Massachusetts	42°22'	71°03'
25	94728	New York, New York	40°47'	73°58'
26	94918	North Omaha, Nebraska	41°22'	96°01'

```
SUBROUTINE ED        ENTRY POINT 000644

STORAGE USED: CODE(1) 000704; DATA(0) 001624; BLANK COMMON(2) 000000

 COMMON BLOCKS:

  0003    READER 000015
  0004    CIMPL  000003

EXTERNAL REFERENCES (BLOCK, NAME)

  0005    NTRAN
  0006    NDCODS
  0007    TBLU1
  0010    NI03$
  0011    NI02$
  0012    NWDU$
  0013    NSTOP$
  0014    NERR3$

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

  0001   000240 100L     0001   000101 147G     0001   000125 163G     0001   000141 173G     0001   000132 2L
  0001   000240 200L     0001   000150 201G     0001   000270 240G     0001   000365 256G     0001   000424 270G
  0001   000312 300L     0000   001506 306F     0001   000345 400L     0001   000512 500L     0001   000605 507L
  0000   001516 508F     0001   000614 600L     0000   001530 608F     0000 R 000300 A        0000 R 000324 AA
  0000 R 001351 B        0000 R 000024 C        0000 R 001504 CIJ      0000 R 001361 CL       0000 R 001421 DM
  0000 R 001500 DYF      0003 R 000001 FMT      0000 R 001435 FMTP     0000 R 000310 FRMT     0000 I 001472 I
  0000 I 001344 IB       0004   000001 ICNT     0000 I 001501 ID       0004 I 000000 IMPL     0000   001571 INJP$
  0000 I 001471 INX      0000 I 001475 IREWIN   0004   000002 ITEST    0000 I 001503 I1       0000 I 001473 J
  0000 I 001502 J1       0000 I 001505 L        0000 R 001470 LTD      0000 I 001477 L1       0000 R 001460 M
  0003 I 000000 N        0000 I 001474 NO       0000 I 001476 N1       0000 R 001452 OFFSET   0007 R 000000 TBLU1
  0000 R 001371 TO


00100      1*     CED                                                                      000000
00101      2*           SUBROUTINE ED(X,X2,X3,X4,X5,X6,X7,X8,NST,NX,IND,TS,TD,DY,           000000
00101      3*          1M1,M2,M3,M4,M5,M6,M7,M8,A1,A2,A3,A4,A5,A6,A7,A8)                   000000
00101      4*     C                                                                        000000
00101      5*     C        PURPOSE    THIS COMPONENT READS THE TYPICAL METEOROLOGICAL          000000
00101      6*     C                   YEAR TAPE WITH A STRUCTURE SIMILAR TO THE                000000
00101      7*     C                   SOLMET DATA TAPE. USER MAY SPECIFY FORMAT FOR NON-       000000
00101      8*     C                   STANDARD TAPES                                           000000
00101      9*     C                                                                        000000
00101     10*     C        WRITTEN BY Y.K.CHAN, 10-5-78, VERSION 1                            000000
00101     11*     C                                                                        000000
00101     12*     C        METHOD     TWENTY FOUR HOURLY RECORDS ARE READ INTO CORE            000000
00101     13*     C                   AT A TIME AND LINEAR INTERPOLATION IS USED TO            000000
```

BCS 40180-2 Rev.

ED

BCS 40180-2 Rev.

```
00101   C                    OBTAIN THE OUTPUT AT CURRENT SIMULATION TIME.             000000
00101   C                                                                               000000
00101   C         CALL SEQUENCE                                                         000000
00101   C              OUTPUTS                                                          000000
00101   C                   X1,....,X8  -OUTPUT VARIABLES AT CURRENT TIME               000000
00101   C                   X1  -BEAM RADIATION IF IND=+-1, W/M2                        000000
00101   C                   X2  -GLOBAL RADIATION IF IND=+-1, W/M2                      000000
00101   C                   X3  -DRY BULB TEMPERATURE IF IND=+-1,  C                    000000
00101   C                   X4  -WIND SPEED IF IND=+-1,M/S                              000000
00101   C              INPUTS                                                           000000
00101   C                   NST     -NUMBER OF BLOCKS TO SKIP AT START                  000000
00101   C                   NX      -NUMBER OF OUTPUT VARIABLES(DEFAULT=4,MAX=8)        000000
00101   C                   IND     -INDICATOR FUNCTION                                 000000
00101   C                            0=NO READ                                          000000
00101   C                           +-1=STANDARD FORMAT AND UNITS(DEFAULT)              000000
00101   C                           +-2=USER SPECIFIED FORMAT AND UNITS                 000000
00101   C                            0,SETS MISSING DATA TO 0                           000000
00101   C                           ?0,SETS MISSING DATA TO PREVIOUS VALUE              000000
00101   C                   TS      -TIME SHIFT OF DATA(DEFAULT=-0.5)                   000000
00101   C                            (COMPENSATION TERM SINCE SOLAR RADIATION           000000
00101   C                            DATA IS AN INTEGRATED TOTAL, USUALLY OVER 1 HOUR)  000000
00101   C                   TD      -CURRENT TIME OF DAY(0-24)                          000000
00101   C                   DY      -CURRENT DAY OF YEAR(1-365)                         000000
00101   C                   M1,....,M8  -UNITS MULTIPLIERS FOR X1,....,X8               000000
00101   C                            DEFAULT M1=...=M8=1                                000000
00101   C                   A1,....,A8  -ADDITION FACTOR FOR X1,....,X8                 000000
00101   C                            DEFAULT A1=...=A8=0                                000000
00101   C                                                                               000000
00103         DIMENSION X(8),A(8),FRMT(12),FMT(12),C(24,6),AA(528),IB(5),B(8),         000000
00103        1 CL(8),TO(24),DM(12),FMTP(13),OFFSET(6)                                  000000
00104         COMMON /READER/N,FMT                                                      000000
00105         COMMON /CIMPL/IMPL,ICNT,ITEST                                             000000
00106         REAL NX,IND,M1,M2,M3,M4,M5,M6,M7,M8,M(8),LTD,NST                          000000
00107         DATA FRMT/72HA5,A2,3I2,11X,F4.0,26X,F4.0,45X,F4.1,7X,F4.1)               000000
00107        1                                                         /                000000
00111         DATA DM/0.,31.,59.,90.,120.,151.,181.,212.,243.,                          000000
00111        1 273.,304.,334./                                                          000000
00113         DATA OFFSET/6H(      ,6H(1X,  ,6H(2X,  ,6H(3X,  ,6H(4X,  ,6H(5X,  /      000000
00113   C                                                                               000000
00115         IF(ABS(IND).LE..1)RETURN                                                  000000
00115   C                                                                               000000
00115   C              INITIALIZATION                                                   000000
00115   C                                                                               000000
00117         IF(IMPL.GT.0)GO TO 100                                                    000006
00121         IF(NX.EQ..99999)NX=4                                                      000011
00123         IF(TS.EQ..99999)TS=-.5                                                    000016
00125         INX=NX+.1                                                                 000023
00126         M(1)=M1                                                                   000034
00127         M(2)=M2                                                                   000036
00130         M(3)=M3                                                                   000040
00131         M(4)=M4                                                                   000042
00132         M(5)=M5                                                                   000044
00133         M(6)=M6                                                                   000046
00134         M(7)=M7                                                                   000050
00135         M(8)=M8                                                                   000052
00136         A(1)=A1                                                                   000054
00137         A(2)=A2                                                                   000056
```

ORIGINAL PAGE IS
OF POOR QUALITY

140K

ED

140L

```
00140            A(3)=A3                                                000060
00141            A(4)=A4                                                000062
00142            A(5)=A5                                                000064
00143            A(6)=A6                                                000066
00144            A(7)=A7                                                000070
00145            A(8)=A8                                                000072
00146            DO 11 I=1,INX                                          000101
00151            IF(M(I).EQ..99999)M(I)=1.                              000101
00153         11 IF(A(I).EQ..99999)A(I)=0.                              000105
00153      C                                                            000105
00153      C                SET DEFAULT TAPE RECORD FORMAT TO STANDARD  000105
00156            IF(ABS(IND).GT.1.01)GO TO 2                            000113
00160            M(1)=1./3.6                                            000117
00161            M(2)=1./3.6                                            000121
00162            DO 3 I=1,12                                            000125
00165          3 FMT(I)=FRMT(I)                                         000125
00167            N=132                                                  000127
00170          2 CONTINUE                                               000132
00171            FMTP(1)=OFFSET(1)                                      000132
00172            DO 5 I=1,12                                            000141
00175          5 FMTP(I+1)=FMT(I)                                       000141
00177            LTD=-1.                                                000143
00200            DO 10 J=1,INX                                          000150
00203         10 CL(J)=0.                                               000150
00203      C                                                            000150
00203      C      POSITION THE FILE                                     000150
00203      C                                                            000150
00205            CALL NTRAN(2,10)                                       000151
00206            NO=NST*.001                                            000155
00207            CALL NTRAN(2,7,NO,22)                                  000166
00207      C                                                            000166
00207      C             READ FIRST DATA BLOCK                          000166
00207      C                                                            000166
00210            IREWIN=0                                               000174
00211            N1=4*N                                                 000175
00212            CALL NTRAN(2,2,N1,AA,L1,22)                            000200
00213            IF(L1.LT.0)GO TO 600                                   000210
00215            DECODE(N,FMTP,AA(1))IB,B                               000213
00221            WRITE(6,308)IB(1),IB(2)                                000227
00225        308 FORMAT(1H0,3X,15HED& STATION ID=,A5,10X,7HYEAR 19,A2)  000236
00226            GO TO 400                                              000236
00226      C                                                            000236
00226      C                                                            000236
00227        100 CONTINUE                                               000240
00227      C                                                            000240
00230        200 CONTINUE                                               000240
00230      C                                                            000240
00230      C               INTERPOLATION FOR OUTPUT IF CURRENT DAY OF YEAR HAS   000240
00230      C               BEEN LOCATED                                 000240
00230      C                                                            000240
00231            IF(DY.GT.(DYF+.1))GO TO 300                            000240
00233            IF(DY.LT.(DYF-.1))GO TO 500                            000244
00235            IF(LTD.EQ.TD)RETURN                                    000254
00237            DO 201 I=1,INX                                         000270
00242        201 X(I)=TBLU1(TD,TO,C(1,I),0,24)*M(I)+A(I)                000270
00244            LTD=TD                                                 000304
00245            RETURN                                                 000336
```

BCS 40180-2 Rev.

ED

```
00245      C                                                              000306
00246       300 CONTINUE                                                  000312
00246      C                                                              000312
00246      C          IF CURRENT DAY OF YEAR HAS NOT BEEN LOCATED, READ MORE TAPE   000312
00246      C                                                              000312
00247            ID = DY -DYF -.9                                         000312
00250            CALL NTRAN( 2, 7, ID, 22)                                000323
00251            CALL NTRAN( 2, 2, N1, AA, L1, 22)                        000331
00252            IF( L1 .LT. 0)   GO TO 600                               000341
00252      C                                                              000341
00252      C           DECODE DATA AND TIME OF DAY                        000341
00252      C                                                              000341
00254       400 CONTINUE                                                  000345
00255            DO 402 I=1,24                                            000345
00260            J1 = ((I-1)*N)/6                                         000371
00261            I1=(I-1)*N-6*J1                                          000374
00262            FMTP(1)=OFFSET(I1+1)                                     000377
00263            DECODE(N,FMTP,AA(J1+1))IB,B                              000402
00267            DO 401 J=1,INX                                           000424
00272            C(I,J)=B(J)                                              000424
00273            CIJ=C(I,J)                                               000425
00274            IF((CIJ.EQ.99.).OR.(CIJ.EQ.999.).OR.(CIJ.EQ.9999.).OR.   000426
00274           1(CIJ.EQ.99999.))C(I,J)=CL(J)                             000426
00276            IF(IND.LT.0.)CL(J)=C(I,J)                                000451
00300       401 CONTINUE                                                  000470
00302            TO(I)=IB(5)+TS                                           000470
00303       402 CONTINUE                                                  000500
00305            L = IB(3)                                                000500
00306            DYF = IB(4) +DM(L)                                       000502
00307            GO TO 200                                                000510
00307      C                                                              000510
00310       500 CONTINUE                                                  000512
00310      C                                                              000512
00310      C            IF DAY OF YEAR ON TAPE IS PAST CURRENT DAY OF YEAR,   000512
00310      C            BACKSPACE TAPE.                                   000512
00311            IF(IREWIN.GT.0)GO TO 507                                 000512
00313            ID=DY-DYF-1.1                                            000514
00314            CALL NTRAN(2,7,ID,22)                                    000526
00315            IREWIN=1                                                 000534
00316            CALL NTRAN(2,2,N1,AA,L1,22)                              000536
00317            IF(L1.LT.0)GO TO 600                                     000546
00321            FMTP(1)=OFFSET(1)                                        000551
00322            DECODE(N,FMTP,AA(1))IB,B                                 000553
00326            L = IB(3)                                                000567
00327            DYF = IB(4) +DM(L)                                       000571
00330            IF(DYF.LT.(DY+.1))GO TO 400                              000577
00332       507 WRITE(6,508)                                              000605
00334       508 FORMAT(1H0,47HEDE INPUT ERROR, DAY OF YEAR DY IS OUT OF RANGE)   000611
00335            STOP                                                     000611
00335      C                                                              000611
00335      C             IF ERROR IN READ, PRINT DIAGNOSTICS              000611
00336       600 WRITE(6,608)                                              000614
00340       608 FORMAT(1H0,27HEDE TAPE INPUT ERROR OR EOF)                000620
00341            STOP                                                     000620
00342            END                                                      000703
```

Revision Pages

Section 7.8A - FO and 7.8B - FP

Delete pages 151 and 152 of the original document and insert revision pages 151 - 152N between pages 150 and 153.

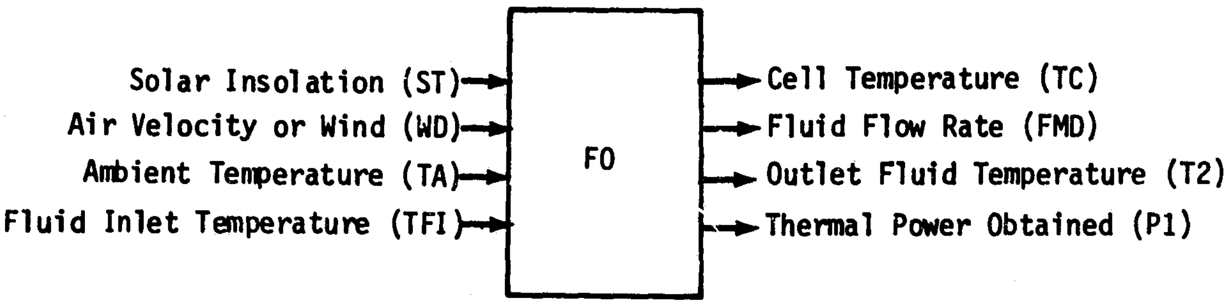
PRECEDING PAGE BLANK NOT FILMED

```
00277    C                                                000715
00277    C            PRIORITY INTERRUPT                  000715
00277    C                                                000715
00301          EC1=E1-EDE                                 000734
00302          EC0=E0+EDE                                 000737
00303          IF((KE.GT.EC0).AND.(INT.EQ.1.))INT=0.      000742
00305          IF((KE.LT.EC1).AND.(INT.EQ.-1.))INT=0.     000760
00307                                                     000776
00307          IF(KE.LE.E0)INT=1.                         000776
00311          IF(KE.GT.E1)INT=-1.                        001004
00313          IF((KE.GT.EC0).AND.(KE.LT.EC1))INT=0.      001012
00315          IF(IMPL.LE.1)RETURN                        001031
00315    C                                                001031
00315    C            STATISTICS                          001031
00315    C                                                001031
00317          ME=AMAX1(ME,KE)                            001040
00320          MPC=AMAX1(MPC,KED)                         001046
00321          MPD=AMAX1(MPD,-KED)                        001054
00322          SPC=SPC+TINC*P1                            001062
00323          SPD=SPD+TINC*P2                            001066
00323    C                                                001066
00324          IF(TIME.LT.TMAX1)RETURN                    001072
00326          CCI=CCI+CC                                 001101
00327          CMI=CMI+CM                                 001104
00327    C                                                001104
00330                                                     001107
00330          RETURN                                     001107
00331          END                                        001332
```

PRECEDING PAGE BLANK NOT FILMED

FL

7.8A FRESNEL LENS SOLAR COLLECTOR



The Fresnel lens collector model performs a thermal analysis for a concentrating photovoltaic array which tracks the sun. The array may be cooled passively or by forced air or fluid. Fins may be used on the back to increase convective heat transfer to the environment. Figures 7.8A-1 and 7.8A-2 show the physical construction of the array and the equivalent thermal network for the focusing collector. The purpose of the model is to compute the cell temperature TC, and the fluid pump rate FMD when fluid cooling is used. The analysis is based on a similar thermal model in SOLCEL [4].

FO

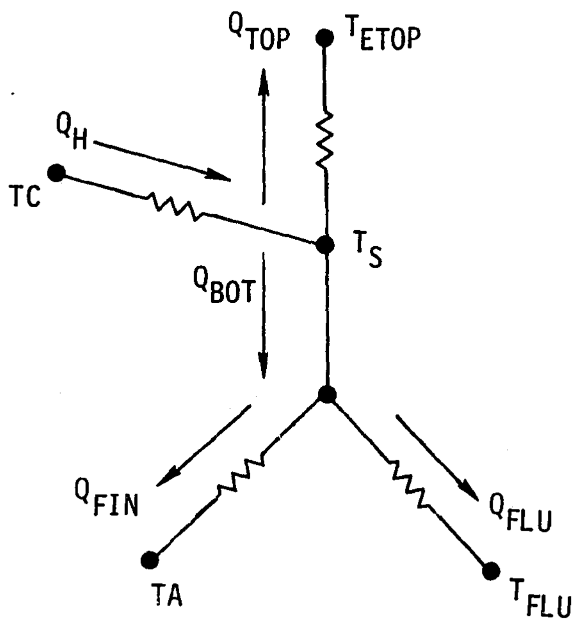


Figure 7.8A-1 Equivalent Thermal Network for Fresnel Lens Collector

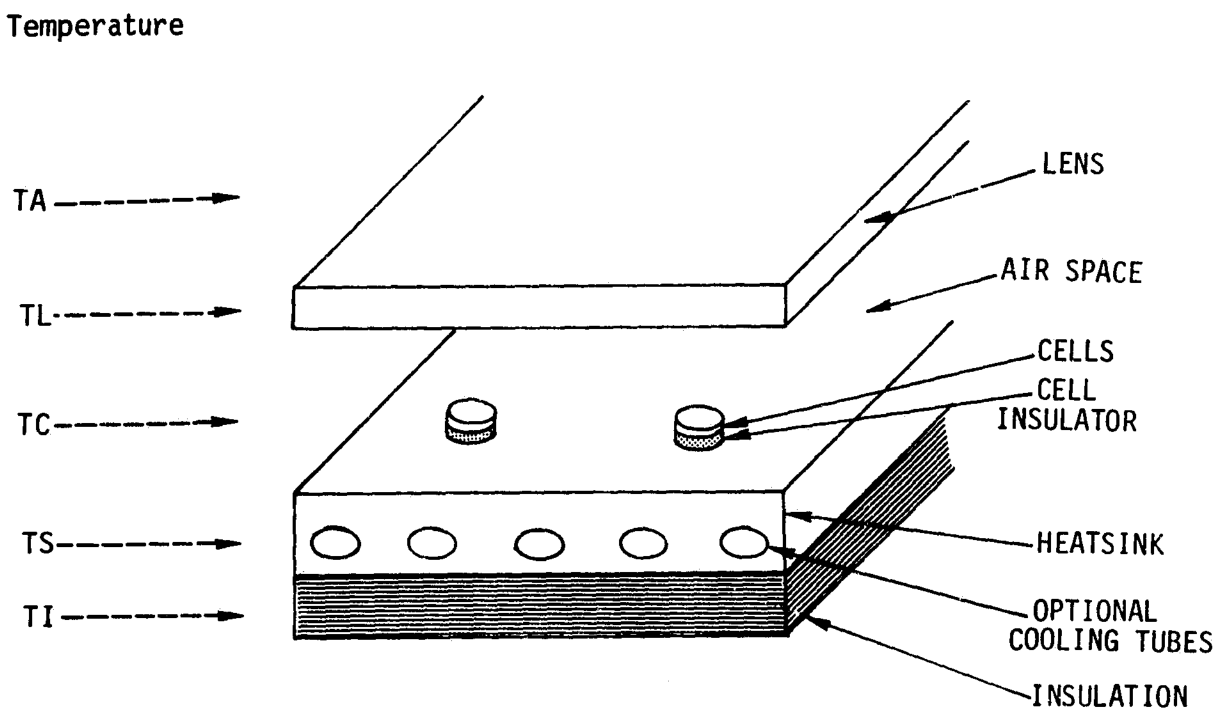


Figure 7.8A-2 Fresnel Lens Thermal Model

BASIC EQUATIONS

1) Energy absorbed by the collector per unit area

QH = ST*TAU*(ABC-EFF)

where

ST = direct beam solar insolation

TAU = lens transmittance

ABC = cell absorptance

EFF = nominal cell efficiency

2) Heat balance equations for the thermal network of 7.8A-1:

$$Q_h = Q_{TOP} + Q_{BOT}$$

$$Q_{TOP} = H_{TOP}(TS-T_{ETOP}) = H_L(TS-TL)$$

$$Q_{BOT} = H_{BOT}(TS-T_{EBOT}) = Q_{FIN} + Q_{FLU}$$

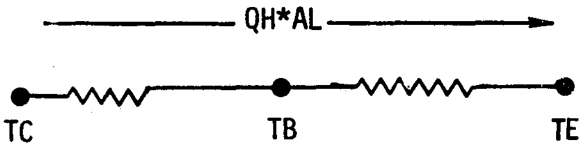
$$Q_{FIN} = H_{FIN}(TS-TA) = H_I(TS-TI)$$

$$Q_{FLU} = H_{FLU}(TS-T_{FLU})$$

3) The temperature variation in the insulating bond between the cell and the heat sink is given by a radial conduction equation for $r > a$:

$$r^2 \frac{\partial^2 T_B}{\partial r^2} + \frac{\partial T_B}{\partial r} - \alpha r^2 T_B = 0,$$

with $\frac{\partial T_B}{\partial r}$ specified at the cell radius r=a and at the equivalent lens radius r=b. This equation may be solved using modified Bessel functions to compute T_B at r=a given the overall heat transfer coefficient and equivalent temperature of the collector minus bonding. Thus the cell, bonding, and collector thermal diagram reduces to



where

AL = lens area = πb^2

$TE = (H_{TOP}*T_{ETOP}+H_{BOT}*T_{EBOT})/(H_{TOP}+H_{BOT})$

Input Specification Notes

Minimum input parameters to specify FO are

CMØ	=	Cooling mode option
TFØ	=	Outlet fluid temperature (CMØ=2)
NT	=	Number of cooling tubes (CMØ=2)
HI	=	Thermal conductivity/thickness of back insulation (CMØ=2)
AL	=	Area of lens
NL	=	Number of lenses
CL	=	Collector length
CW	=	Collector width
RC	=	Radius of solar cell
FIR	=	Cooling fin/collector area ratio (CMØ=0)

The user should check inputs for consistency with those used in the photovoltaic model PV. For example

FO collector area = $CL*CW \overset{?}{\geq} AL*NL \overset{?}{\geq}$ PV array area

FO concentration ratio = $AL/(\pi*RC^2) \overset{?}{=}$ PV concentration ratio

FO cell area = $\pi*RC^2 \overset{?}{\geq}$ PV array area/number of cells

Inputs/Port	Description	Units
ST	Direct beam solar insolation	w/m^2
WD	Air or wind velocity (default = 0.)	m/s
TA	Ambient temperature	^oC
TFI	Inlet fluid temperature	^oC
TFØ	Specified outlet fluid temperature	^oC
CMØ	Cooling mode (default = 0.) 0 = natural air cooling 1 = forced air cooling 2 = fluid cooling	-
AL	Lens area	m^2
TAU	Lens transmittance (default = 1.)	-
ABC	Cell absorptance (default = .95)	-
EFF	Nominal cell efficiency (default = .12)	-
SPA	Lens to heatsink space (default = .025)	m
EL	Emittance of lens (default = .9)	-
ES	Heatsink emittance (default = .5)	-
EI	Emittance of the back surface (default = .5)	-
CW	Collector width	m
CL	Collector length	m
NL	Number of lenses on collector	-
RC	Radius of solar cells (default = .025)	m
ABL	Absorptance of the lens (default = .05)	-
SPT	Specific heat of coolant (default = 4184)	j/kg-K
HI	Conductivity/thickness of the back insulation (default = 10^9 for no insulation)	w/m^2-K

Inputs/Port (cont'd)		Description	Units
FIR		Cooling fin to flat plate area ratio (default = 1 for no fin)	-
NT		Number of cooling tubes	-
MFM		Maximum fluid flow rate	kg/s
DT		Diameter of cooling tubes (default = .015)	m
CØS		Conductivity of heatsink (default = 202)	w/m-K
THS		Heatsink plate thickness (default = .003)	m
DEN		Coolant density (default = 980.)	kg/m^3
CØC		Conductivity of the coolant (default = .657)	w/m
HC		Conductivity/thickness of the cell insulator (default = 10^9 for no insulation)	w/m^2-K
CC		Capital cost per unit collector area per year	$\$/m^2$
CM		Maintenance cost per year	$
CØP		Cost of operating power	$/kwh

Outputs/Port		Description	Units
TC		Cell temperature	^oC
TS		Heatsink temperature	^oC
FMD		Fluid flow rate	kg/s
T	1	Inlet fluid temperature	^oC
T	2	Outlet fluid temperature	^oC
PH		Collector energy absorbed	kw
P	1	Thermal energy collected	kw

Outputs/Port (cont'd)	Description	Units
REA	Reynolds number (air cooling)	-
REF	Reynolds number (fluid cooling)	-
LTI	Last time at which the collector calculations were performed	hr
ØP	Operating Power used (state)	kwh

CALCULATION SEQUENCE

$$RL = (AL/\pi)^{.5}$$

1) Solar Power Absorbed by the Collector

QH = ST*TAU(ABC-EFF)

PH = QH*AL*NL/1000.

If $QH \leq 0.1$ set TC = TA, FMD = P1 = $\dot{Ø}$P = 0 and return

If LTI = TIME and $|TFI - T1| < .1$, return

LTI = TIME

2) Convert TA,TFO,TFI to ^{o}K

3) Initial Temperature and Flow Rate Estimates

TS = TA + QH/20

TL = (TS + TØ)*.5

TF = (TFI + TFO)*.5

TI = TL

FMD = IFLU = 0

If CMØ = 2 and TFØ >TFI, IFLU = 1

If IFLU = 1,

CALCULATION SEQUENCE (cont'd)

RO = NT*SPT*(TFØ-TFI)/(AL*NL)

FMD = MIN(0.5*QH/RO,MFM)

o Iterate 4) to 8) three times:

4) HTOP Heat Transfer Coefficient and TETOP

$T_{SKY} = .0552*TA^{1.5}$

$\begin{pmatrix} HC1 \\ REA \end{pmatrix}$ = CNVC(TL,TA,WD,CL) Appendix (2)-(3)

HR1 = RADC(TL,TSKY,EL,1.)*$\frac{(TL-TSKY)}{(TL-TA)}$ Ibid,(8)

H1 = HC1 + HR1

TM = .5*(TL+TS)

HC2 = $(7.25x10^{-5}*TM+4.325x10^{-3})/SPA$

HR2 = RADC(TS,TL,ES,EL) Ibid,(8)

HL = HC2 + HR2

HTOP = $(1/H1 + 1/HL)^{-1}$

TETOP = TA+ST*(ABL+(1-TAU)*TAU*ABC)/H1

5) Fin Factor and HFIN Heat Transfer Coefficient

HC = CNVC(TI,TA,WD,CL) Ibid,(2)-(3)

HR = RADC(TI,TA,EI,1.) Ibid,(8)

FAC = 4.318 - 4.3375*EXP(-.26795*FIR) (First pass)

HFIN = $(1/HI + 1/(HC*FAC+HR))^{-1}$

6) HFLU Heat Transfer Coefficient to Fluid and REF

HFLU = 0.

CALCULATION SEQUENCE (cont'd)

If IFLU = 0 go to (7)

$$\begin{pmatrix} HFLU \\ REF \end{pmatrix} = FLUC(NT,DT,CW,CØS,THS, FMD,DEN,TF,CØC) \qquad Ibid,(5)-(6)$$

7) HBOT Heat Transfer Coefficient and Temperature TEBOT

HBOT = HFIN + HFLU

TEBOT = (HFIN*TA+HFLU*TF)/HBOT

8) Temperature and Flow Rate Updates

H = HTOP + HBOT

TE = (HTOP*TETOP+HBOT*TEBOT)/H

TS = TE + QH/H

TL = TS - HTOP*(TS-TETOP)/HL

TI = TS - HFIN(TS-TA)/HI

QFLU = HFLU(TS-TF)

FMD = 0.

If QFLU > 0, FMD = QFLU/RO

If QFLU >MFM*RO,

FMD = MFM

RA = QFLU/MFM

TF = TFI+RA*AL*NL*.5/(SPT*NT)

9) Check for QFLU < 0

If QFLU < 0 set IFLU = 0 and repeat (4)-(8) once

10) Cell Temperature

ALPH = H/(CØS*THS)

X = SQRT(ALPH)*RC

CALCULATION SEQUENCE (cont'd)

Y = SQRT(ALPH)*RL

BETA = QH*AL/(2π*CØS*THS*X)

A = BETA*I1(Y)/(K1(X)*I1(Y)-K1(Y)*I1(X))

B = BETA*K1(Y)/(K1(X)*I1(Y)-K1(Y)*I1(X))

TB = A*K0(X)+B*I0(X)+TE

TC = TB+QH*AL/(π*RC^2*HC)

where I0,I1,K0,K1 are modified Bessel functions.

11) Output Calculation

T2 = 2*TF-TFI

Convert TC,TS,T1,T2,TA,TFI,TFØ to ^{o}C

P1 = QFLU*AL*NL/1000.

TKP = 5.E-4*CL*CW

$$\dot{Ø}P = TKP + \begin{cases} 0. & \text{if } CMØ = 0 \\ .0742*(CW*CL)^{.2835}*WD^{.567} & \text{if } CMØ = 1 \text{ and } WD > 0 \\ 7.85\times10^{-11}*FMD^{2.855}*DT^{(-4.702)}*NT*CL & \text{if } CMØ = 2 \text{ and } FMD > 0 \end{cases}$$

REFERENCES FOR FO

1. J. K. Linn, "Photovoltaic System Analysis Program-SOLCEL," Sandia Laboratories Report SAND77-1268, 1977.

2. E. L. Burgess and M. W. Edenburn, "One Kilowatt Photovoltaic Subsystem Using Fresnel Lens Concentrators," Paper 11.6, IEEE Photovoltaic Specialists Conference, Baton Rouge, November 1976.

```
00101   C              DT   -DIAMETER OF COOLING TUBES,M,(DEFAULT=.015)                    000000
00101   C              COS  -CONDUCTIVITY OF HEAT SINK,W/M-K,(DEFAULT=202)                 000000
00101   C              THS  -HEATSINK PLATE THICKNESS,M,(DEFAULT=.003)                     000000
00101   C              DEN  -COOLANT DENSITY,KG/M3,(DEFAULT=980)                           000000
00101   C              COC  -CONDUCTIVITY OF THE COOLANT,W/M-K,(DEFAULT=.657)              000000
00101   C              HC   -CONDUCTIVITY/THICKNESS OF THE CELL INSULATOR,                 000000
00101   C                    W/M2-K,(DEFAULT=10**9 FOR NO INSULATION)                      000000
00101   C              CC   -CAPITAL COST PER UNIT COLLECTOR AREA PER YEAR,$/M2            000000
00101   C              CM   -MAINTENANCE COST PER YEAR                                     000000
00101   C              COP  -COST OF OPERATING POWER,$/KWH                                 000000
00101   C                                                                                  000000
00103         COMMON /CIMPL/IMPL,ICNT,ITEST                                                000000
00104         COMMON /CTIME/TIME /CSIMUL/DUM(7),TMAX                                       000000
00105         COMMON /COST/CCAP,CMA,CPO                                                    000000
00106         REAL NL,NT,MFM,LTI                                                           000000
00107         DOUBLE PRECISION  MMBSI0,MMBSI1,MMBSK0,MMBSK1                                000000
00110         DOUBLE PRECISION X,Y                                                         000000
00110   C                                                                                  000000
00110   C          INITIALIZATION                                                          000000
00110   C                                                                                  000000
00111         IF(IMPL.GT.0)GO TO 100                                                       000000
00113         IF(WD.EQ..99999)WD=0.                                                        000002
00115         IF(CMO.EQ..99999)CMO=0.                                                      000006
00117         IF(AL.EQ..99999)AL=.09                                                       000012
00121         IF(TAU.EQ..99999)TAU=1.                                                      000017
00123         IF(ABC.EQ..99999)ABC=.95                                                     000024
00125         IF(EFF.EQ..99999)EFF=.12                                                     000031
00127         IF(SPA.EQ..99999)SPA=.025                                                    000036
00131         IF(EL.EQ..99999)EL=.9                                                        000043
00133         IF(ES.EQ..99999)ES=.5                                                        000050
00135         IF(EI.EQ..99999)EI=.5                                                        000055
00137         IF(RC.EQ..99999)RC=.025                                                      000062
00141         IF(ABL.EQ..99999)ABL=.05                                                     000067
00143         IF(SPT.EQ..99999)SPT=4184                                                    000074
00145         IF(HI.EQ..99999)HI=1.E9                                                      000101
00147         IF(FIR.EQ..99999)FIR=1.                                                      000106
00151         IF(DT.EQ..99999)DT=.015                                                      000113
00153         IF(COS.EQ..99999)COS=202                                                     000120
00155         IF(THS.EQ..99999)THS=.003                                                    000125
00157         IF(DEN.EQ..99999)DEN=980                                                     000132
00161         IF(COC.EQ..99999)COC=.657                                                    000137
00163         IF(HC.EQ..99999)HC=1.E9                                                      000144
00165         RL=SQRT(AL/3.1415926)                                                        000151
00166         FAC=4.318-4.3375*EXP(-.26795*FIR)                                            000160
00167         TMAX1=TMAX*.99999                                                            000171
00167   C                                                                                  000171
00170     100 CONTINUE                                                                     000175
00170   C                                                                                  000175
00170   C     SOLAR POWER ABSORBED BY THE COLLECTOR                                        000175
00170   C                                                                                  000175
00171         QH=ST*TAU*(ABC-EFF)                                                          000175
00172         PH=QH*AL*NL/1000.                                                            000202
00173         IF( QH .GT. 0.1)   GO TO 201                                                 000206
00175         TS=TA                                                                        000212
00176         TC=TA                                                                        000214
00177         OPD=0.                                                                       000215
00200         FMD=0.                                                                       000216
```

```
00201              PI=0.                                                                000217
00202              GO TO 920                                                            000220
00203          201 IF((LTI.EQ.TIME).AND.(ABS(TFI-TI).LT..1))GO TO 920                   000222
00205              LTI=TIME                                                             000240
00205        C                                                                          000240
00205        C             CONVERT TA,TFO,TFI  FROM CENTIGRADE TO KELVIN                000240
00205        C                                                                          000240
00206              TA=TA+273                                                            000242
00207              TFO=TFO+273                                                          000245
00210              TFI=TFI+273                                                          000250
00210        C                                                                          000250
00210        C           INITIAL TEMPERATURE AND FLOW RATE ESTIMATES                    000250
00210        C                                                                          000250
00211              TS=TA+QH/20.                                                         000253
00212              TL=(TS+TA)*.5                                                        000257
00213              TF=(TFI+TFO)*.5                                                      000262
00214              TI=TL                                                                000265
00215              IFLU=0.                                                              000266
00216              FMD=0.                                                               000267
00217              IF((ABS(CMO-2.).LT..1).AND.(TFO.GT.TFI))IFLU=1                       000270
00221              IF(IFLU.NE.1)GO TO 301                                               000312
00223              RO=NT*SPT*(TFO-TFI)/(AL*NL)                                          000315
00224              FMD=MFM                                                              000326
00225              IF(RO.GT.0.)FMD=AMIN1(.5*QH/RO,MFM)                                  000330
00227          301 CONTINUE                                                             000344
00227        C                                                                          000344
00227        C           ITERATE HEAT COEFFICIENT CALCULATION THREE TIMES               000344
00227        C                                                                          000344
00230              LOOP=0                                                               000344
00231          400 CONTINUE                                                             000345
00231        C                                                                          000345
00231        C    HTOP, HEAT TRANSFER COEFFICIENT, AND TETOP,TOP EQUIVALENT TEMPERATURE 000345
00232              TSKY=.0552*(TA**1.5)                                                 000345
00233              CALL CNVC(HC1,REA,TL,TA,WD,CL)                                       000352
00234              CALL RADC(HR1,TL,TSKY,EL,1.)                                         000362
00235              HR1=HR1*(TL-TSKY)/(TL-TA)                                            000371
00236              H1=HC1+HR1                                                           000400
00237              TM=.5*(TL+TS)                                                        000402
00240              HC2=(7.25*1.E-5*TM+4.325E-3)/SPA                                     000405
00241              CALL RADC(HR2,TS,TL,ES,EL)                                           000412
00242              HL=HC2+HR2                                                           000421
00243              HTOP=1./(1./H1+1./HL)                                                000424
00244              TETOP=TA+ST*(ABL+(1-TAU)*TAU*ABC)/H1                                 000434
00244        C                                                                          000434
00244        C            HEAT TRANSFER COEFFICIENT HFIN                                000434
00245              CALL CNVC(HC2,RE,TI,TA,WD,CL)                                        000445
00246              CALL RADC(HR,TI,TA,EI,1.)                                            000455
00247              HFIN=1./(1./HI+1./(HC2*FAC+HR))                                      000464
00247        C                                                                          000464
00247        C     FLUID HEAT TRANSFER COEFFICIENT HFLU AND REYNOLDS NUMBER REF         000464
00250              HFLU=0.                                                              000477
00251              IF(IFLU.EQ.0)GO TO 700                                               000500
00253              CALL FLUC(HFLU,REF,NT,DT,CW,COS,THS,FMD,DEN,TF,COC)                  000502
00253        C                                                                          000502
00253        C     EQUIVALENT BOTTOM TEMPERATURE AND HEAT TRANSFER COEFFICIENT          000502
00254          700 CONTINUE                                                             000520
00255              HBOT=HFIN+HFLU                                                       000520
```

151N

F0

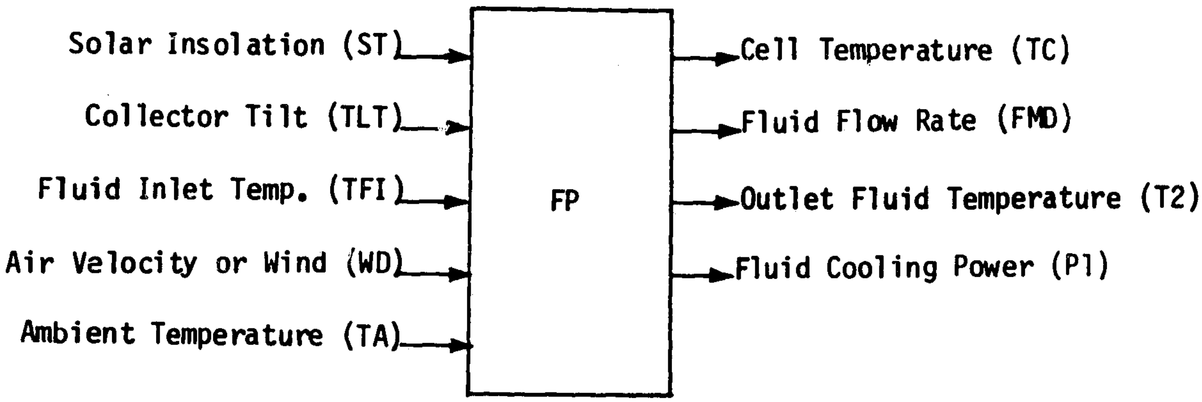
```
00256          TEBOT=(HFIN*TA+HFLU*TF)/HBOT                                 000522
00256     C                                                                 000522
00256     C        UPDATE TEMPERATURE AND FLOW RATE                         000522
00257          H=HTOP+HBOT                                                  000531
00260          TE=(HTOP*TETOP+HBOT*TEBOT)/H                                 000533
00261          TS=TE+QH/H                                                   000541
00262          TL=TS-HTOP*(TS-TETOP)/HL                                     000545
00263          TI=TS-HFIN*(TS-TA)/HI                                        000552
00264          QFLU=HFLU*(TS-TF)                                            000560
00264     C    WRITE(6,108)HFIN,HBOT,TEBOT,HTOP,TETOP,H,TE,TS,TL,TI,QFLU,RO 000560
00264     C 108 FORMAT(1H ,*FOC *,8E10.2,/,5X,8E10.2)                       000560
00265          FMD=0.                                                       000564
00266          IF(QFLU.LE.0.)GO TO 800                                      000565
00270          IF(QFLU.GT.(MFM*RO))GO TO 799                                000567
00272          FMD=QFLU/RO                                                  000574
00273          GO TO 800                                                    000577
00274     799  FMD=MFM                                                      000601
00275          RA=QFLU/MFM                                                  000602
00276          TF=TFI+RA*AL*NL*.5/(SPT*NT)                                  000605
00277     800  CONTINUE                                                     000616
00277     C                                                                 000616
00300          LOOP=LOOP+1                                                  000616
00301          IF(LOOP.LE.2)GO TO 400                                       000620
00301     C                                                                 000620
00301     C     CHECK FOR EFFECTIVE FLUID COOLING                           000620
00301     C                                                                 000620
00303          IF(QFLU.GE.0.)GO TO 900                                      000623
00305          IFLU=0.                                                      000626
00306          GO TO 400                                                    000627
00307     900  CONTINUE                                                     000631
00307     C                                                                 000631
00307     C              CELL TEMPERATURE                                   000631
00307     C                                                                 000631
00310           ALPH=H/(COS*THS)                                            000631
00311          X=SQRT(ALPH)*RC                                              000635
00312          Y=SQRT(ALPH)*RL                                              000645
00313          BETA=QH*AL/(2.*3.14159*COS*THS*X)                            000650
00314          BI1Y=MMBSI1(1,Y,IER)                                         000663
00315          BK1X=MMBSK1(1,X,IER)                                         000672
00316          BI1X=MMBSI1(1,X,IER)                                         000701
00317          BK1Y=MMBSK1(1,Y,IER)                                         000710
00320          BK0X=MMBSK0(1,X,IER)                                         000717
00321          BI0X=MMBSI0(1,X,IER)                                         000726
00322          A=BETA*BI1Y/(BK1X*BI1Y-BK1Y*BI1X)                            000735
00323          B=BETA*BK1Y/(BK1X*BI1Y-BK1Y*BI1X)                            000746
00324          TB=A*BK0X+B*BI0X+TE                                          000752
00325          TC=TB+QH*AL/(3.14159*RC*RC*HC)                               000757
00325     C                                                                 000757
00325     C     OUTPUT CALCULATION                                          000757
00325     C                                                                 000757
00326          TC=TC-273                                                    000770
00327          TS=TS-273                                                    000772
00330          T1=TFI-273                                                   000775
00331          T2=2.*TF-TFI-273                                             001000
00332          TA=TA-273                                                    001005
00333          TFI=TFI-273                                                  001010
00334          TFO=TFO-273                                                  001011
```


F0

```
00335        P1=OFLU*AL*NL/1000.                                     001014
00336        RE1=0.                                                  001021
00337        IF(ABS(CMO-1.).LE..1)RE1=.0742*((CW*CL)**.2835)*WD**.567 001022
00341        IF(FMD.LE.0.)GO TO 909                                  001047
00343        IF(CMO.GT.1.1)RE1=7.65E-11*(FMD**2.855)*(DT**(-4.702))*NT*CL 001052
00345    909 CONTINUE                                                001075
00346        TKP=5.E-4*CL*CW                                         001075
00347        IF(IOP.NE.0)OPD=TKP+RE1                                 001100
00351    920 IF(TIME.LT.TMAX1)RETURN                                 001106
00353        IF(IMPL.LT.2)RETURN                                     001114
00355        CCAP=CCAP+CC*AL*NL                                      001123
00356        CMA=CMA+CM                                              001130
00357        CPO=CPO+COP*OP                                          001133
00360        RETURN                                                  001137
00361        END                                                     001515
```


FO

7.8B FLAT PLATE SOLAR COLLECTOR



The flat plate component performs a thermal analysis on a nonconcentrating photovoltaic array. Three types of cooling may be used:

- Front surface cooling using natural or forced air.
- Back surface cooling using natural or forced air with or without a finned back surface.
- Fluid cooling using tubes on the back and N glass covers (N = 0,1,2,3).

Figures 7.8B-1 and 7.8B-2 show the physical construction of the array and the equivalent thermal network for the flat plate component. The purpose of the analysis is to compute the cell temperature TC and the fluid pump rate FMD when fluid cooling is used. The analysis is based on the flat plate thermal model in SOLCEL [4], except that an empirical equation due to Klein is used to compute the top loss coefficient for 1 to 3 glass covers.

TEMPERATURES

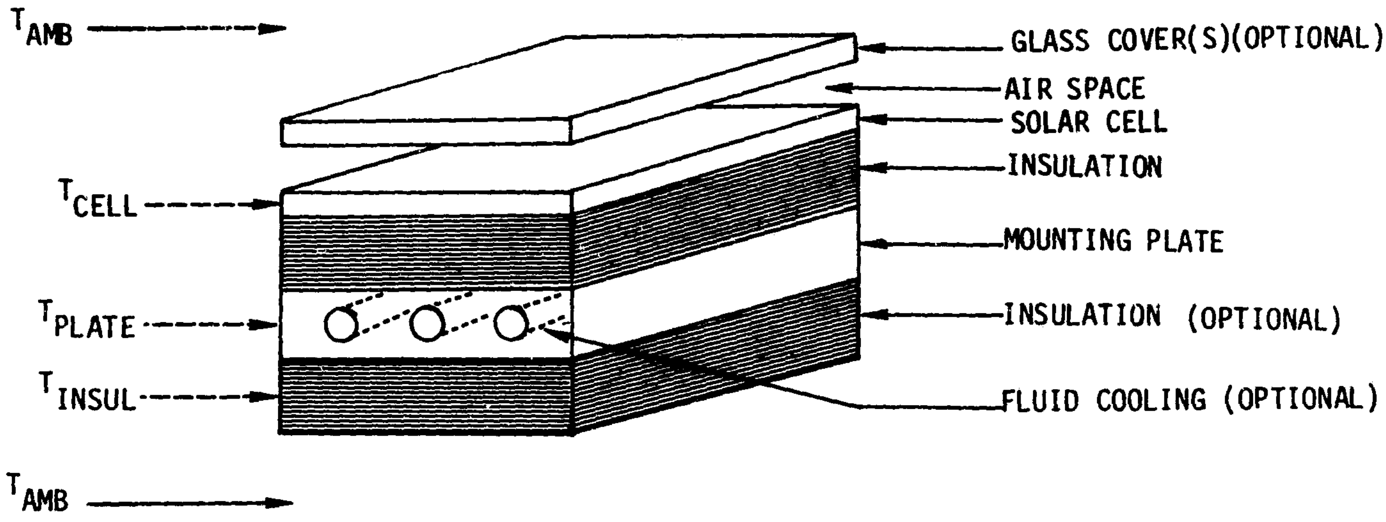


Figure 7.8B-1 Physical Diagram of Flat Plate Collector

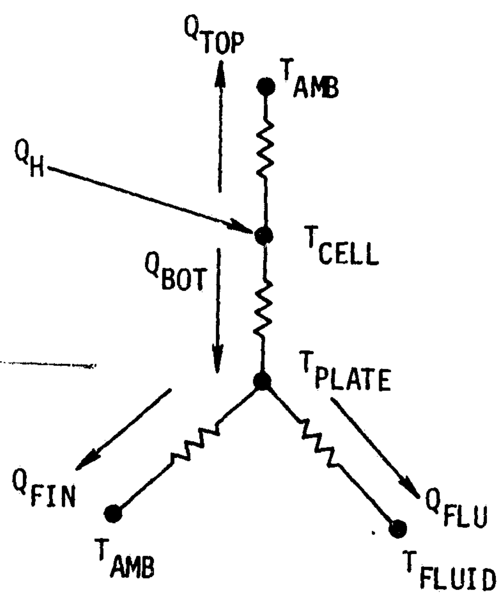


Figure 7.8B-2 Equivalent Thermal Network for Flat Plate Collector

BASIC EQUATIONS

The basic thermal equations for the model are the heat balance equations for the network of Figure 7.8B-2.

$$Q_H = ST*T_N(AB - EFF) = Q_{TOP} + Q_{BOT}$$

$$Q_{TOP} = H_{TOP}(T_{CELL} - T_{AMB})$$

$$Q_{BOT} = H_{BOT}(T_{CELL} - T_{EBOT}) = H_C\ (T_{CELL} - T_{PLATE})$$

$$= Q_{FIN} + Q_{FLU}$$

$$Q_{FIN} = H_{FIN}(T_{PLATE} - T_{AMB}) = H_I(T_{PLATE} - T_{INSUL})$$

$$Q_{FLU} = FMD*P(TFØ - TFI) = H_{FLU}(T_{PLATE} - T_{FLUID})$$

where H_{TOP}, H_{BOT}, H_C... denote heat transfer coefficients, and

T_N = transmittance of the N-covers

AB = collector cell absorptance

EFF = nominal cell efficiency

T_{EBOT} = equivalent bottom temp. (= T_{AMB} with no fluid cooling)

P = fluid specific heat/unit cell area * No. of cooling tubes

T_{FLUID} = average fluid temperature = (TFØ + TFI)/2.

Input Specification Notes

Minimum input parameters to specify FO are

CMØ	=	Cooling mode option
TFØ	=	Outlet fluid temperature (CMØ=2)
NG	=	Number of glass covers
HI	=	Conductivity/thickness of the back insulation
CW	=	Collector width
CL	=	Collector length
NT	=	Number of cooling tubes (CMØ=2)
FIR	=	Cooling fin/collector area ratio (CMØ=0)

The user should check the consistency of these inputs (e.g., collector area) with those of the tracking component SO and the photovoltaic component PV.

Inputs/Port	Description	Units
ST	Global solar insolation	w/m^2
TLT	Collector tilt	Deg
WD	Air or wind velocity (default = 0.)	m/s
TA	Ambient drybulb temperature	^oC
TFI	Inlet fluid temperature	^oC
TFØ	Specified outlet fluid temperature	^oC
MFM	Maximum fluid flow rate	kg/s
RE	Tracking power request	kw
CMØ	Cooling mode (default = 0.) 0 = natural air cooling 1 = forced air cooling 2 = fluid cooling	-
NG	Number of glass covers (default = 0.)	-
TN	Transmittance of the N-covers	-
AB	Collector cell absorptance (default = .9)	-
EFF	Nominal cell efficiency (default = .12)	-
EC	Emittance of cell (default = 0.5)	-
EG	Emittance of the glass covers (default = .9)	-
EP	Emittance of the back surface (default = .9)	-
CW	Collector width	m
CL	Collector length	m
SPT	Specific heat of coolant (default = 4184.)	j/kg-K
HI	Conductivity/thickness of the back insulation (default = 10^9 for no insulation)	w/m^2K

Inputs/Port (cont'd)	Description	Units
FIR	Cooling fin to flat plate area ratio (default = 1. for no fin)	-
NT	Number of cooling tubes (default = 1)	-
DT	Diameter of cooling tubes (default = .015)	m
CØP	Conductivity of mounting plate (default = 202.)	w/m-K
THP	Mounting plate thickness (default = .003)	m
DEN	Coolant density (default = 980.)	kg/m^3
CØC	Conductivity of the coolant (default = .657)	w/m-K
HC	Conductivity/thickness for cell insulation (default = 10^9 for no insulation)	w/m^2-K
CC	Capital cost per unit area per year	$\$/m^2$
CM	Maintenance cost per year	\$
CPØ	Cost of operating power	\$/kwh

Outputs/Port	Description	Units
TC	Cell temperature	^oC
TP	Mounting plate temperature	^oC
FMD	Fluid flow rate	kg/s
T1	Inlet fluid temperature	^oC
T2	Outlet fluid temperature	^oC
PH	Collector energy absorbed	kw
P1	Thermal energy collected	kw
ØP	Operating power used (state)	kwh
REA	Reynolds number (air cooling)	-
REF	Reynolds number (fluid cooling)	-
LTI	Last time at which the flat plate array calculations were performed (used internally)	hr

CALCULATION SEQUENCE

1) Solar power absorbed by the collector

QH = ST*TN*(AB - EFF)

PH = QH*CL*CW/1000

If QH $\leq$ 0.1 set TC = TA, FMD = P1 = $\dot{O}$P = 0 and return

If LTI = TIME and |TFI - T1| < .1, return

LTI = TIME

2) Convert TA, TFØ, TFI to ^{o}K

3) Initial temperature and flow rate estimates

TC = TA + QH/20

TI = (TC + TA)*.5

TF = (TFI+TFØ)*.5

TP = TI

FMD = 0

IFLU = 0

If CMØ = 2 and TFØ >TFI, IFLU = 1

If IFLU = 1,

RO = NT*SPT*(TFØ - TFI)/CW*CL

FMD = MIN(MFM,0.8*QH/RO)

o Iterate 4) to 8) three times:

4) HTOP heat transfer coefficient and REA

TSKY = .0552*TA$^{1.5}$

$\begin{pmatrix} HC1 \\ REA \end{pmatrix}$ = CNVC(TC, TA, WD, CL) See (2)-(3) in Appendix

If NG = 0,

CALLCULATIONS (cont'd)

$$HR1 = RADC(TC,TSKY,EC,1.)*\frac{(TC-TSKY)}{(TC-TA)}$$ Ibid,(8)

$$HTOP = HC1 + HR1$$

If $NG > 0$,

$$HTOP = HTGLAS(N,TA,TC,HC1,EC,EG,TLT)$$ Ibid,(7)

5) Fin factor FAC and HFIN heat transfer coefficient

$$HC2 = CNVC(TI, TA, WD, CL)$$ Ibid,(3)

$$HR2 = RADC(TI, TA, EP, 1.)$$ Ibid,(8)

$$FAC = 4.318 - 4.3375*\exp(- .26795*FIR)$$ (first pass)

$$HFIN = (1/HI + 1/(HC2*FAC + HR2))^{-1}$$

6) HFLU heat transfer coefficient to fluid and REF

$$HFLU = 0.$$

If IFLU = 0 go to 7)

$$\begin{pmatrix} HFLU \\ REF \end{pmatrix} = FLUC(NT,DT,DW,C\emptyset P,THP,FMD,DEN,TF,C\emptyset C)$$ Ibid,(5)-(6)

7) HBOT heat transfer coefficient and equivalent temperature TEBOT

$$HBOT = (1/HC + 1/(HFIN + HFLU))^{-1}$$

$$TEBOT = (HFIN*TA + HFLU*TF)/(HFIN + HFLU)$$

8) Temperature and flow rate updates

$$TC = (QH + HTOP*TA + HBOT*TEBOT)/(HTOP + HBOT)$$

$$TP = TC - HBOT*(TC-TEBOT)/HC$$

$$TI = TP - HFIN*(TP - TA)/HI$$

$$QFLU = HFLU*(TP - TF)$$

CALCULATIONS (cont'd)

$$FMD = \begin{cases} 0. & \text{if } QFLU \leq 0. \\ QFLU/RO & \text{if } QFLU > 0. \end{cases}$$

If QFLU > MFM*RO,

FMD = MFM

RA = QFLU/MFM

TF = TFI + RA*CL*CW*.5/(SPT*NT)

9) Check for QFLU < 0

If QFLU < 0 set IFLU = 0 and repeat 4) to 8) once

10) Output calculations

T2 = 2*TF - TFI

Convert TC,TP,T1,T2,TA,TFI,TFØ to ^{o}C

P1 = QFLU*CL*CW/1000

If CMØ = 0

$\dot{Ø}P$ = RE

If CMØ = 1 and WD>0

$\dot{Ø}P = RE+.0742*(CW*CL)^{.2835}*WD^{.567}$

If CMØ = 2 and FMD>0

$\dot{Ø}P = RE + 7.85 \times 10^{-11}*FMD^{2.855}*DT^{(-4.702)}*NT*CL$

REFERENCES FOR FP

1. S. A. Klein, M.S. Thesis, "The Effects of Thermal Capacitance Upon the Performance of Flat Plate Solar Collectors," University of Wisconsin, 1973.

2. J. A. Duffie and W. A. Beckman, Solar Energy Thermal Processes. Wiley, 1974.

3. F. Kreith, Principles of Heat Transfer, 3rd Edition, International Textbook Company, 1973.

```
SUBROUTINE FP        ENTRY POINT 000741

STORAGE USED: CODE(1) 001270; DATA(0) 000111; BLANK COMMON(2) 000000

 COMMON BLOCKS:

 0003   CIMPL  000001
 0004   CTIME  000001
 0005   CSIMUL 000310
 0006   COST   000003

EXTERNAL REFERENCES (BLOCK, NAME)

 0007   CNVC
 0010   RADC
 0011   HTGLAS
 0012   FLUC
 0013   EXP
 0014   XPRR
 0015   NERR3$

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

 0001   000146 100L     0001   000173 201L     0001   000315 301L     0001   000316 400L     0001   000360 401L
 0001   000372 402L     0001   000444 700L     0001   000530 799L     0001   000545 800L     0001   000560 900L
 0001   000667 909L     0001   000674 920L     0006 R 000000 CCAP     0006 R 000001 CMA      0006 R 000002 CPOS
 0005   000000 DUM      0000 R 000001 FAC      0000 R 000021 HBOT     0000 R 000011 HC1      0000 R 000014 HC2
 0000 R 000017 HFIN     0000 R 000020 HFLU     0000 R 000012 HR1      0000 R 000016 HR2      0011 R 000000 HTGLAS
 0000 R 000013 HTOP     0000 I 000005 IFLU     0003 I 000000 IMPL     0000   000063 INJPS    0000 I 000007 LOOP
 0000 R 000023 QFLU     0000 R 000002 QH       0000 R 000024 RA       0000 R 000315 REN1     0000 R 000025 RE1
 0000 R 000006 RO       0000 R 000022 TEBOT    0000 R 000004 TF       0000 R 000003 TI       0004 R 000000 TIME
 0005 R 000007 TMAX     0000 R 000000 TMAX1    0000 R 000010 TSKY


00100        CFP                                                                       000000
00101              SUBROUTINE FP(TC,TP,FMD,T1,T2,PH,P1,OP,OPD,IOP,REA,REF,LTI,        000000
00101             1 ST,TLT,WD,TA,TFI,TFO,MFM,RE,CMO,NG,TN,AB,                         000000
00101             2 EFF,EC,EG,EP,CW,CL,SPT,HI,FIR,NT,DT,COP,THP,DEN,                  000000
00101             3 COC,HC,CC,CM,CPO)                                                 000000
00101        C                                                                         000000
00101        C      PURPOSE     THIS COMPONENT PERFORMS A THERMAL ANALYSIS             000000
00101        C                  ON A NONCONCENTRATING PHOTOVOLTAIC ARRAY.              000000
00101        C                  THREE TYPES OF COOLING MAY BE USED:                    000000
00101        C                      FRONT SURFACE COOLING USING NATURAL OR FORCED AIR  000000
00101        C                      BACK SURFACE COOLING USING NATURAL OR FORCED AIR   000000
00101        C                           WITH OR WITHOUT FINS.                         000000
00101        C                      FLUID COOLING USING TUBES ON THE BACK AND NG       000000
00101        C                           GLASS COVERS (NG=0,1,2,3).                    000000
```

FP

```
C
C        WRITTEN BY Y.K.CHAN, 11-6-78, VERSION 1
C
C
C        METHOD    BASED ON THE FLAT PLATE THERMAL MODEL IN SOLCEL,
C                     EXCEPT THAT AN EMPIRICAL EQUATION DUE TO KLEIN IS USED
C                     TO COMPUTE THE TOP LOSS COEFFICIENT FOR 1 TO 3
C                     GLASS COVERS
C        CALLING SEQUENCE
C         OUTPUTS
C           ST    -GLOBAL SOLAR INSOLATION,W/M2
C           TC    -CELL TEMPERATURE,C
C           FMD   -FLUID FLOW RATE,KG/S
C           T1    -INLET FLUID TEMPERATURE,C
C           T2    -OUTLET FLUID TEMPERATURE,C
C           PH    -COLLECTOR ENERGY ABSORBED,KW
C           P1    -THERMAL ENERGY COLLECTED,KW
C           OP    -OPERATING POWER USED(STATE),KWH
C           REA   -REYNOLDS NUMBER(AIR COOLING)
C           REF   -REYNOLDS NUMBER(FLUID COOLING)
C           LTI   -LAST TIME AT WHICH THE FLAT PLATE ARRAY
C                  CALCULATIONS WERE PERFORMED(USED INTERNALLY)
C         INPUTS
C           TLT   -COLLECTOR TILT,DEGREES
C           WD    -AIR OR WIND VELOCITY,M/S,(DEFAULT=0.)
C           TA    -AMBIENT DRYBULB TEMPERATURE,C
C           TFI   -SPECIFIED INLET FLUID TEMPERATURE,C
C           TFO   -SPECIFIED OUTLET FLUID TEMPERATURE,C
C           MFM   -MAXIMUM FLUID FLOW RATE,KG/S
C           RE    -TRACKING POWER RECUEST,KW
C           CMO   -COOLING MODE(DEFAULT=0)
C                     0=NATURAL AIR COOLING
C                     1=FORCED AIR COOLING
C                     2=FLUID COOLING
C           NG    -NUMBER OF GLASS COVERS(DEFAULT=0)
C           TN    -TRANSMITTANCE OF THE NG GLASS COVERS
C           AB    -COLLECTOR CELL ABSORPTANCE(DEFAULT=.9)
C           EFF   -NOMINAL CELL EFFICIENCY(DEFAULT=.12)
C           EC    -EMITTANCE OF CELL(DEFAULT=.5)
C           EG    -EMITTANCE OF GLASS COVERS(DEFAULT=.9)
C           EP    -EMITTANCE OF THE BACK SURFACE(DEFAULT=.9)
C           CW    -COLLECTOR WIDTH,M
C           CL    -COLLECTOR LENGTH,M
C           SPT   -SPECIFIC HEAT OF COOLANT,J/KG-K,(DEFAULT=4184)
C           HI    -CONDUCTIVITY/THICKNESS OF THE BACK INSULATION,W/M2-K,
C                  (DEFAULT=1.E9 FOR NO INSULATION)
C           FIR   -COOLING FIN TO FLAT PLATE AREA RATIO(DEFAULT=1. FOR NO FIN)
C           NT    -NUMBER OF COOLING TUBES(DEFAULT=1)
C           DT    -DIAMETER OF COOLING TUBES,M,(DEFAULT=.015)
C           COP   -CONDUCTIVITY OF MOUNTING PLATE,W/M-K,(DEFAULT=202)
C           THP   -MOUNTING PLATE THICKNESS,M,(DEFAULT=.003)
C           DEN   -COOLANT DENSITY,KG/M3,(DEFAULT=983)
C           COC   -CONDUCTIVITY OF COOLANT,W/M-K,(DEFAULT=.657
C           HC    -CONDUCTIVITY/THICKNESS FOR CELL INSULATION,W/M2-K,
C                  (DEFAULT=1.E9 FOR NO INSULATION)
C           CC    -CAPITAL COST PER UNIT AREA PER YEAR,$/M2
C           CM    -MAINTENANCE COST PER YEAR,$
```


FP

152J

```
00101   C          CPO  -COST OF OPERATING POWER,$/KWH                     000000
00101   C                                                                  000000
00103        COMMON /CIMPL/IMPL                                            000000
00104        COMMON /CTIME/TIME /CSIMUL/DUM(7),TMAX                        000000
00105        COMMON /COST/CCAP,CMA,CPOS                                    000000
00106        REAL LTI,MFM,NG,NT                                            000000
00106   C                                                                  000000
00106   C           INITIALIZATION                                         000000
00106   C                                                                  000000
00107        IF(IMPL.GT.0)GO TO 100                                        000000
00111        IF(WD.EQ..99999)WD=3.                                         000002
00113        IF(CMO.EQ..99999)CMO=0.                                       000006
00115        IF(NG.EQ..99999)NG=0.                                         000012
00117        IF(AB.EQ..99999)AB=.9                                         000016
00121        IF(EFF.EQ..99999)EFF=.12                                      000023
00123        IF(EC.EQ..99999)EC=.5                                         000030
00125        IF(EG.EQ..99999)EG=.9                                         000035
00127        IF(EP.EQ..99999)EP=.9                                         000042
00131        IF(SPT.EQ..99999)SPT=4184                                     000047
00133        IF(HI.EQ..99999) HI=1.E9                                      000054
00135        IF(FIR.EQ..99999)FIR=1.                                       000061
00137        IF(NT.EQ..99999)NT=1                                          000066
00141        IF(DT.EQ..99999)DT=.015                                       000073
00143        IF(COP.EQ..99999)COP=202.                                     000100
00145         IF(THP.EQ..99999)THP=.003                                    000105
00147        IF(DEN.EQ..99999)DEN=980                                      000112
00151        IF(COC.EQ..99999)COC=.657                                     000117
00153        IF(HC.EQ..99999)HC=1.E9                                       000124
00155        TMAX1=TMAX*.99999                                             000131
00155        FAC=4.318-4.3375*EXP(-.26795*FIR)                             000134
00157    100 CONTINUE                                                      000146
00157   C                                                                  000146
00157   C         SOLAR POWER ABSORBED BY COLLECTOR                        000146
00157   C                                                                  000146
00160        QH=ST*TN*(AB-EFF)                                             000146
00161        PH=QH*CL*CW/1000.                                             000153
00162        IF( QH .GT. 0.1)   GO TO 201                                  000157
00164        TP=TA                                                         000163
00165        QPD=0.                                                        000165
00166        TC=TA                                                         000166
00167        FMD=0.                                                        000167
00170        P1=0.                                                         000170
00171        GO TO 920                                                     000171
00172    201 IF((LTI.EQ.TIME).AND.(ABS(TFI-T1).LT..1))GO TO 920            000173
00174        LTI=TIME                                                      000211
00174   C                                                                  000211
00174   C         CONVERT TA,TFO,TFI TO KELVIN                             000211
00174   C                                                                  000211
00175        TA=TA+273                                                     000213
00176        TFO=TFO+273                                                   000216
00177        TFI=TFI+273                                                   000221
00177   C                                                                  000221
00177   C          INITIAL TEMPERATURE AND FLOW RATE ESTIMATES             000221
00177   C                                                                  000221
00200        TC=TA+QH/20.                                                  000224
00201        TI=(TC+TA)*.5                                                 000230
00202        TF=(TFI+TFO)*.5                                               000233
```

C-2

BCS 40180-2 Rev.

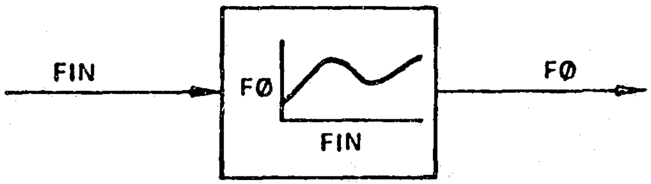
FP

```
00203              TP=TI                                                    000236
00204              FMD=0.                                                   000237
00205              IFLU=0                                                   000240
00206              IF((ABS(CMO-2.).LT..1).AND.(TFO.GT.TFI))IFLU=1           000241
00210              IF(IFLU.NE.1)GO TO 301                                   000263
00212              RO=NT*SPT*(TFO-TFI)/(CW*CL)                              000266
00213              FMD=MFM                                                  000277
00214              IF(PO.GT.0.)FMD=AMIN1(MFM,.8*QH/RO)                      000301
00216          301 CONTINUE                                                 000315
00216        C                                                              000315
00216        C      ITERATE HEAT TRANSFER COEFFICIENT CALCULATION THREE TIMES  000315
00216        C                                                              000315
00217              LOOP=0                                                   000315
00220          400 CONTINUE                                                 000316
00220        C                                                              000316
00220        C      HTOP, HEAT TRANSFER COEFFICIENT AND REA, REYNOLDS NUMBER  000316
00220        C                                                              000316
00220        C                                                              000316
00221              TSKY=.0552*(TA**1.5)                                     000316
00222              CALL CNVC(HC1,REA,TC,TA,WD,CL)                           000323
00223              IF(NG.GT.0.)GO TO 401                                    000333
00225              CALL RADC(HR1,TC,TSKY,EC,1.)                             000336
00226              HR1=HR1*(TC-TSKY)/(TC-TA)                                000345
00227              HTOP=HC1+HR1                                             000354
00230              GO TO 402                                                000356
00231          401 HTOP=HTGLAS(NG,TA,TC,HC1,EC,EG,TLT)                      000360
00232          402 CONTINUE                                                 000372
00232        C                                                              000372
00232        C          HFIN  HEAT TRANSFER COEFFICIENT                     000372
00232        C                                                              000372
00233              CALL CNVC(HC2,REN1,TI,TA,WD,CL)                          000372
00234              CALL RADC(HR2,TI,TA,EP,1.)                               000401
00235              HFIN=1./(1./HI+1./(HC2*FAC+HR2))                         000410
00235        C                                                              000410
00235        C     HFLU, HEAT TRANSFER COEFFICIENT TO FLUID AND REF,REYNOLDS NUMBER  000410
00235        C                                                              000410
00236              HFLU=0.                                                  000423
00237              IF(IFLU.EQ.0)GO TO 700                                   000424
00241              CALL FLUC(HFLU,REF,NT,DT,CW,COP,THP,FMD,DEN,TF,COC)      000426
00241        C                                                              000426
00241        C      EQUIVALENT BOTTOM TEMPERATURE TEBOT AND HEAT TRANSFER COEFFICIENT HBOT  000426
00241        C                                                              000426
00242          700 CONTINUE                                                 000444
00243              HBOT=1./(1./HC+1./(HFIN+HFLU))                           000444
00244              TEBOT=(HFIN*TA+HFLU*TF)/(HFIN+HFLU)                      000455
00244        C                                                              000455
00244        C     UPDATE TEMPERATURE AND FLOW RATE                         000455
00244        C                                                              000455
00245              TC=(QH+HTOP*TA+HBOT*TEBOT)/(HTOP+HBOT)                   000464
00246              TP=TC-HBOT*(TC-TEBOT)/HC                                 000475
00247              TI=TP-HFIN*(TP-TA)/HI                                    000502
00250              QFLU=HFLU*(TP-TF)                                        000507
00251              FMD=0.                                                   000513
00252              IF(QFLU.LE.0.)GO TO 800                                  000514
00254              IF(QFLU.GT.(MFM*RO))GO TO 799                            000516
00256              FMD=QFLU/RO                                              000523
00257              GO TO 800                                                000526
```


FP

```
00260         799 FMD=MFM                                              000530
00261             RA=QFLU/MFM                                          000531
00262             TF=TFI+RA*CL*CW*.5/(SPT*NT)                          000534
00263         800 CONTINUE                                             000545
00263       C                                                          000545
00264             LOOP=LOOP+1                                          000545
00265             IF(LOOP.LE.2)GO TO 400                               000547
00265       C                                                          000547
00265       C       CHECK FOR EFFECTIVE FLUID COOLING                  000547
00265       C                                                          000547
00267             IF(QFLU.GE.0.)GO TO 900                              000552
00271             IFLU=0                                               000555
00272             GO TO 400                                            000556
00273         900 CONTINUE                                             000560
00273       C                                                          000560
00273       C       OUTPUT CALCULATION                                 000560
00273       C                                                          000560
00274             TC=TC-273.                                           000560
00275             TP=TP-273.                                           000562
00276             T1=TFI-273.                                          000565
00277             T2=2.*TF-TFI-273.                                    000570
00300             TA=TA-273.                                           000575
00301             TFI=TFI-273.                                         000600
00302             TFO=TFO-273.                                         000601
00303             P1=QFLU*CL*CW/1000.                                  000604
00304             RE1=0.                                               000611
00305             IF(ABS(CMO-1.).LE..1)RE1=.0742*((CW*CL)**.2835)*(VD**.567)   000612
00307             IF(FMD.LE.0.)GO TO 909                               000637
00311             IF(ABS(CMO-2.).LE..1)RE1=7.85E-11*(FMD**2.855)*(DT**(-4.792))   000642
00311        1     *NT*CL                                              000642
00313         909 CONTINUE                                             000667
00314             IF(IOP.NE.0)OPD=RE+RE1                               000667
00316         920 IF(TIME.LT.TMAX1)RETURN                              000674
00320             IF(IMPL.LT.2)RETURN                                  000702
00322             CCAP=CCAP+CC*CL*CW                                   000711
00323             CMA=CMA+CM                                           000716
00324             CPOS=CPOS+CPO*OP                                     000721
00325             RETURN                                               000725
00326             END                                                  001267
```

7.9 ONE DIMENSION TABLE LOOKUP



Tables	Description
FTA	Tabular values of function

Inputs

Parameter/Port	
FIN	Input quantity
AN	ABS(AN) ≤ 0.5 for equispaced interpolation (AN < 0 prevents extrapolation)

Outputs

Variable/Port	
FØ	Output quantity

Calculation Sequence

FØ = FTA(FIN)

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

PRECEDING PAGE BLANK NOT FILMED

Revision Pages

Sections 7.27 - 7.29

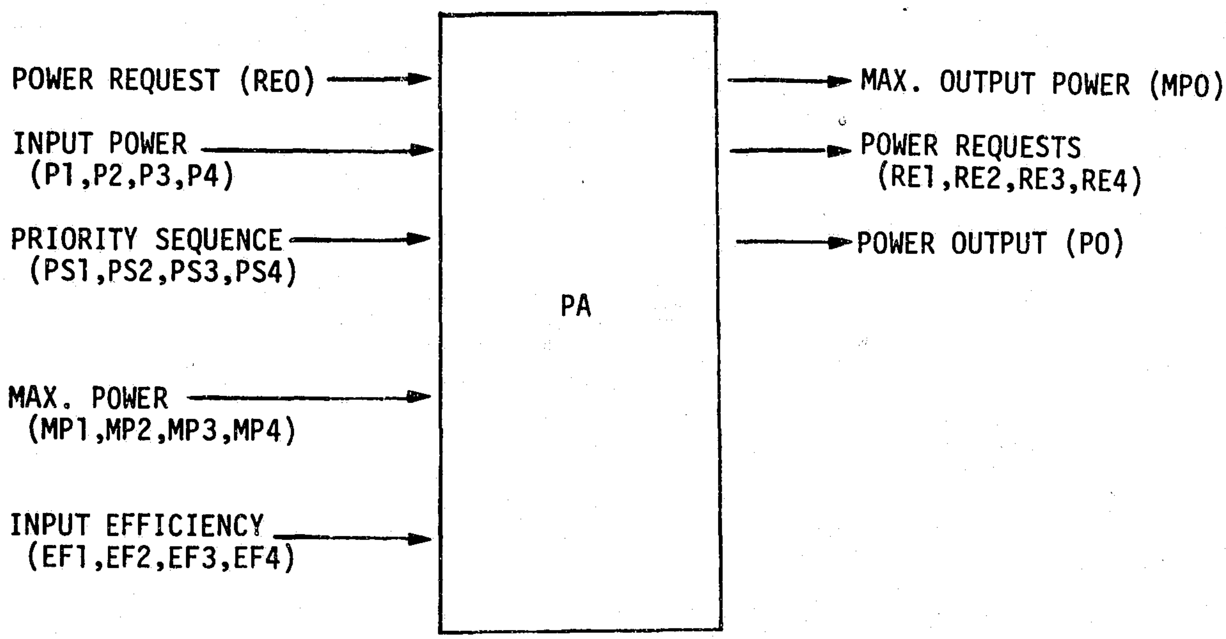
Replaces pages 243 to 266 of the original document.

PRECEDING PAGE BLANK NOT FILMED

```
C0173    408 FORMAT(1H0,19H STATOR RESISTANCE  ,F12.3,12H OR DAMPING ,     000172
00173      XF12.3,20H TOO HIGH FOR MOTOR  )                                 000172
00174       IF(IMPL.EQ.2)ICNT=ICNT+1                                        000172
00174   C                                                                   000172
00174   C            EFFICIENCY AND MAXIMUM OUTPUT POWER                    000172
00174   C                                                                   000172
00176    409 CONTINUE                                                       000201
00177       P2=0.                                                           000201
00200       EF2=EF1                                                         000201
00201       MP2=AMIN1(MP1,RAP)                                              000203
00202       GO TO 500                                                       000211
00203    400 EF2=EF1*P2/P1                                                  000213
00204       MP2=AMIN1(MP1,RAP)*P2/P1                                        000216
00205       IF(RS.NE.0.)TO=P2*737.6/OMEGA                                   000226
00205   C                                                                   000226
00207    500 IF(IMPL.LE.1)RETURN                                            000235
00207   C                                                                   000235
00207   C            STATISTICS                                             000235
00207   C                                                                   000235
00211       MT=AMAX1(TO,MT)                                                 000243
00212       MPN=AMAX1(P2/RAP,MPN)                                           000251
00213       SP=SP+P2*TINC                                                   000260
00213   C                                                                   000260
00214       IF(TIME.LT.TMAX1)RETURN                                         000264
00216       CCI=CCI+CC                                                      000273
00217       CMI=CMI+CM                                                      000276
00217   C                                                                   000276
00220       RETURN                                                          000301
00221       END                                                             000433
```

PRECEDING PAGE BLANK NOT FILMED

7.27 POWER ACCUMULATOR



This component sums power from four input ports and allocates power requests to each port's source of power generation. An input power request is allocated according to user-supplied weights within the ports of highest priority. If an input power request (load) exceeds the maximum power that can be delivered by the ports of highest priority, then the remaining load is allocated to the next priority ports. (See 1.2.2 and 7c for further discussion.)

PA

Inputs[1]

Parameter/Port		Description	Units
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 = 0.)	kw

Outputs

Variable/Port			
MP	0	Maximum deliverable power (ΣMP(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 = $RE_0 - \Sigma MP_i$)	kw

Statistics

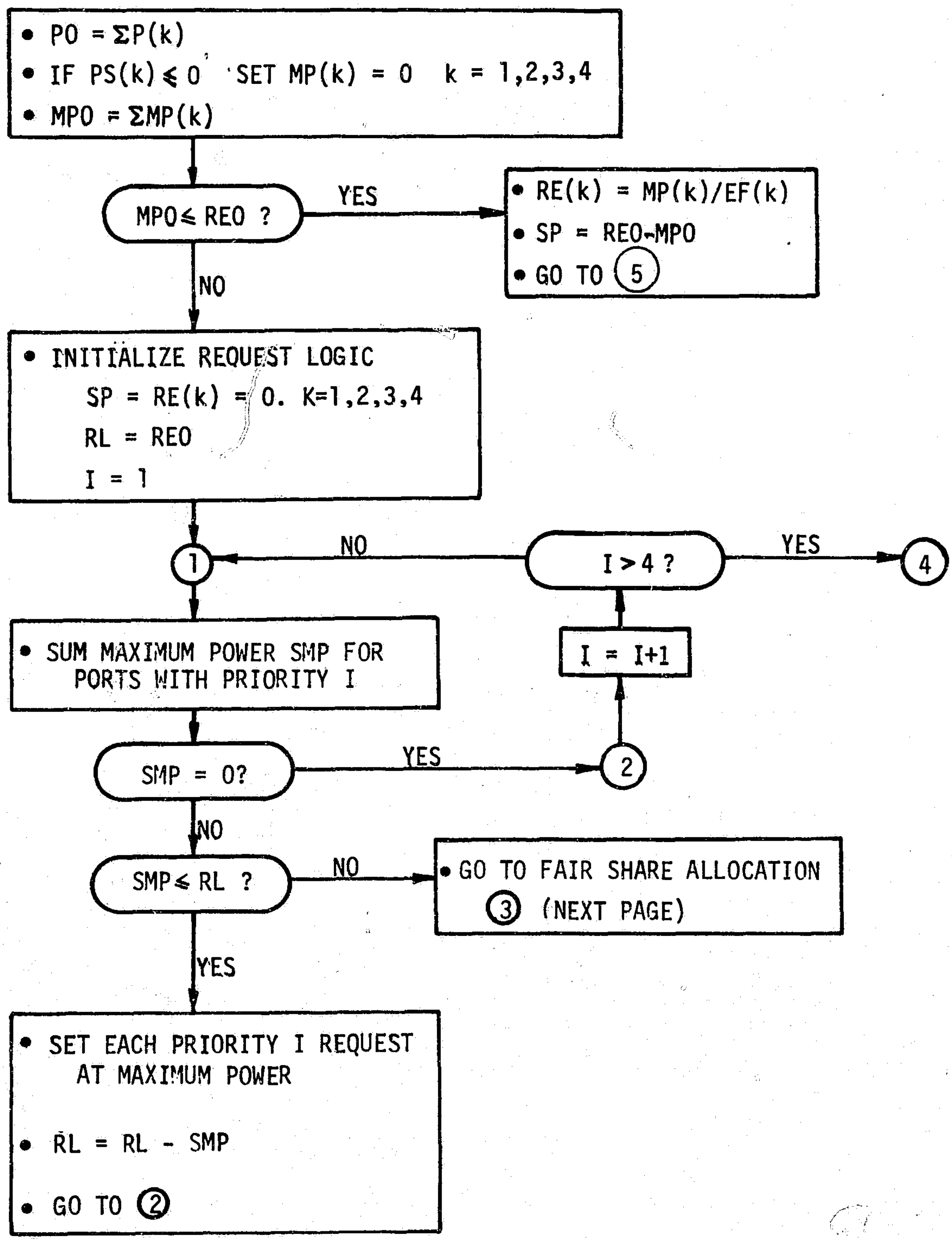
SRE		Sum of energy requested	kwh
PC	1,2,3,4	Percent of cumulative load request delivered by port i	%

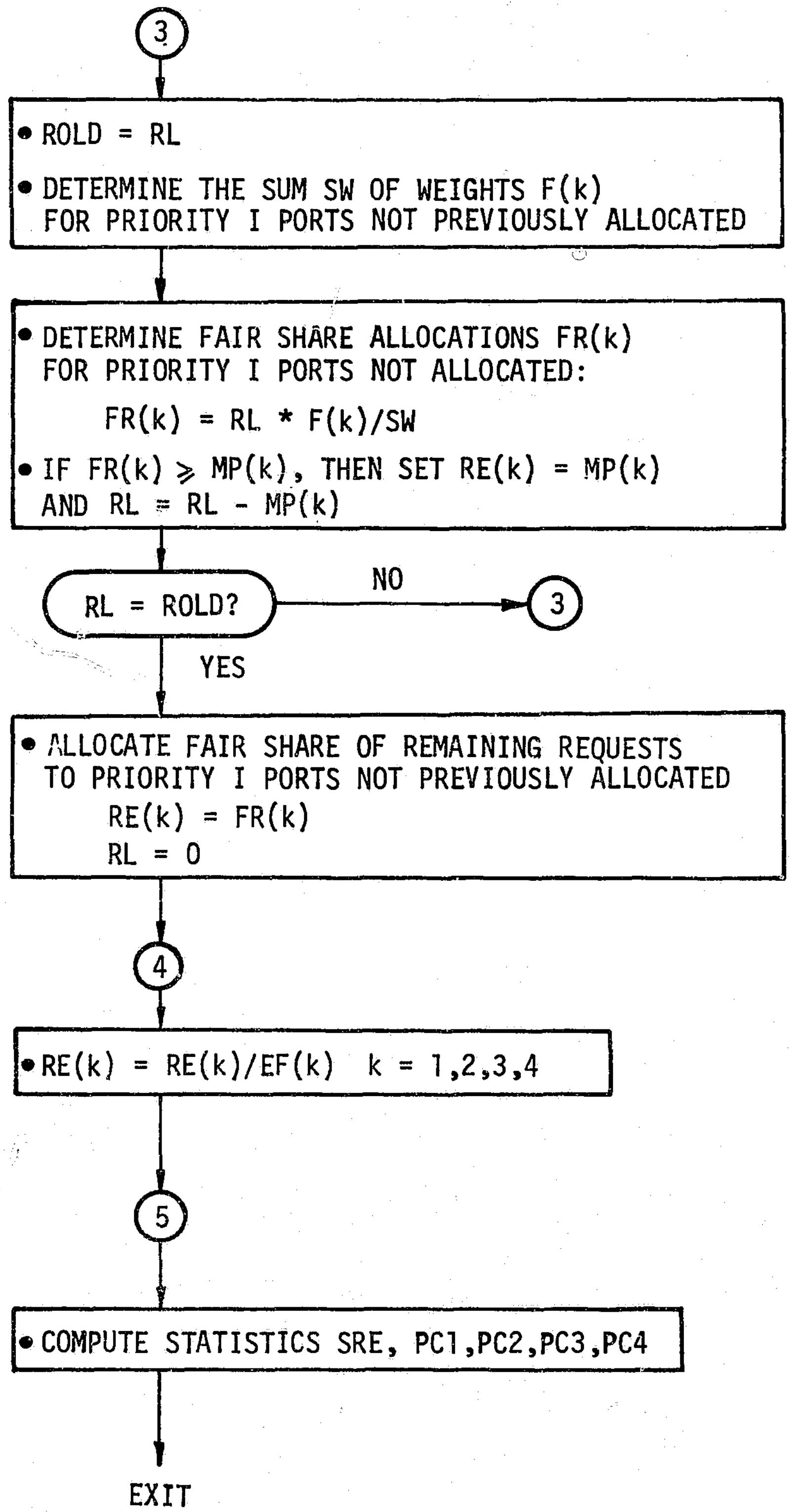
---

[1] No capital costs assigned since this is an allocation component, not a physical device.

PA

CALCULATION LOGIC




3
• ROLD = RL
• DETERMINE THE SUM SW OF WEIGHTS F(k) FOR PRIORITY I PORTS NOT PREVIOUSLY ALLOCATED
• DETERMINE FAIR SHARE ALLOCATIONS FR(k) FOR PRIORITY I PORTS NOT ALLOCATED:
FR(k) = RL * F(k)/SW
• IF FR(k) ⩾ MP(k), THEN SET RE(k) = MP(k) AND RL = RL - MP(k)
RL = ROLD?
NO
3
YES
• ALLOCATE FAIR SHARE OF REMAINING REQUESTS TO PRIORITY I PORTS NOT PREVIOUSLY ALLOCATED
RE(k) = FR(k)
RL = 0
4
• RE(k) = RE(k)/EF(k)   k = 1,2,3,4
5
• COMPUTE STATISTICS SRE, PC1,PC2,PC3,PC4
EXIT

ORIGINAL PAGE IS OF POOR QUALITY

```
SUBROUTINE PA          ENTRY POINT 000514

STORAGE USED: CODE(1) 000746; DATA(0) 000107; BLANK COMMON(2) 000000

 COMMON BLOCKS:

 0003   CIMPL  000001
 0004   CSIMUL 000010

EXTERNAL REFERENCES (BLOCK, NAME)

 0005   NERR3$

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

 0001     000372 1000L     0001     000375 2000L     0001     000231 234G      0001     000240 241G      0001     000260 255G
 0001     000302 273G      0001     000321 305G      0001     000357 325G      0001     000100 40L       0001     000275 400L
 0001     000413 500L      0001     000275 600L      0001     000312 700L      0001     000155 80L       0001     000347 800L
 0001     000366 900L      0004     000000 DUM       0000 R 000030 FR        0000 R 000042 FRU       0000 I 000037 I
 0003 I 000000 IMPL      0000     000054 INJP$     0000 I 000041 K         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 SRO       0000 R 000020 SW        0004 R 000006 TINC      0000 R 000036 TINC1     0004     000007 TMAX
 0000 R 000010 W         0000 R 000040 XI


00100        CPA                                                                               000000
00101                SUBROUTINE PA(MPD,                                                        000000
00101              1       R1, R2, R3, R4,                                                     000000
00101              2       PO,SP,                                                              000000
00101              3       SR,PC1,PC2,PC3,PC4,                                                 000000
00101              4       RO,                                                                 000000
00101              4       EF1, EF2, EF3, EF4,                                                 000000
00101              5       P1, P2, P3, P4,                                                     000000
00101              6       PR1, PR2, PR3, PR4,                                                 000000
00101              7       W1, W2, W3, W4,                                                     000000
00101              8       MP1, MP2, MP3, MP4)                                                 000000
00101        C                                                                                 000000
00101        C      PURPOSE.   MODEL POWER ACCUMULATOR                                         000000
00101        C                                                                                 000000
00101        C      METHOD.  PRIMARY REQUEST ALLOCATION RESULTING FROM PRIORITY                000000
00101        C      ASSIGNMENTS.  SECONDARY REQUEST ALLOCATION RESULTING                       000000
00101        C      FROM WEIGHT ASSIGNMENTS.                                                   000000
00101        C      THAT IS, REQUESTS ARE ALLOCATED ACCORDING TO%                              000000
00101        C        *  PORT PRIORITY (HIGHEST PRIORITY = 1)                                  000000
00101        C        *  PORT WEIGHTS (IN CASE OF EQUAL PRIORTIES.  )                          000000
00101        C                                                                                 000000
00101        C      FORMAL ARGUMENT DEFINITION.                                                000000
00101        C      R1,...., R4 %   POWER REQUESTS IN KW      (OUTPUTS)                        000000
```

PA

```
00101        C      MPO %          TOTAL MAXIMUM POWER        (OUTPUT)            000000
00101        C      SP  %          SURPLUS REQUEST            (OUTPUT)            000000
00101        C      PO  %          TOTAL LOAD IN KW           (OUTPUT)            000000
00101        C      SR  %          SUM OF ENERGY REQUESTED, KWH (OUTPUT)          000000
00101        C      PC1,....,PC4   PERCENT OF CUM LOAD DELIVERED (OUTPUT)         000000
00101        C      RO %           TOTAL POWER REQUESTED,KW (INPUT)               000000
00101        C      P1,...., P4 %  INPUT POWER IN KW          (INPUTS)            000000
00101        C      PR1,...., PR4 % PORT PRIORITIES           (INPUTS)            000000
00101        C      W1,...., W4 %  PORT WEIGHTS               (INPUTS)            000000
00101        C      MP1, ...., MP4 %  MAXIMUM POWERS          (INPUTS)            000000
00101        C      EF1, ...., EF4 %  EFFICIENCIES            (INPUTS)            000000
00101        C       COMMON STORAGE                                               000000
00103                COMMON/ CIMPL / IMPL                                         000000
00104                COMMON / CSIMUL / DUM(6), TINC, TMAX                         000000
00105               REAL MPO,MP1,MP2,MP3,MP4                                      000000
00105        C                                                                    000000
00105        C      LOCAL VARIABLES                                               000000
00105        C                                                                    000000
00105        C                                                                    000000
00105        C      R(K) IS THE POWER REQUEST AT PORT K                           000000
00106               REAL R(4)                                                     000000
00106        C                                                                    000000
00106        C      PR(K) IS THE PRIORITY ASSIGNED TO PORT K                      000000
00107               REAL PR(4)                                                    000000
00107        C                                                                    000000
00107        C      W(K) IS THE WEIGHT ASSIGNED TO PORT K                         000000
00110               REAL W(4)                                                     000000
00110        C                                                                    000000
00110        C      MP(K) IS MAXIMUM POWER TO BE ALLOCATED TO PORT K              000000
00111               REAL MP(4)                                                    000000
00111        C                                                                    000000
00111        C      SW(I) IS THE SUM OF THE WEIGHTS ASSIGNED TO PRIORITY-I PORTS  000000
00112               REAL SW(4)                                                    000000
00112        C                                                                    000000
00112        C      SMP(I) IS THE SUM OF THE MAXIMUM POWER AT PRIORITY-I PORTS    000000
00113               REAL SMP(4)                                                   000000
00113        C                                                                    000000
00113        C      FPU IS  FAIR SHARE  UNIT FOR PRIORITY-I PORTS                 000000
00113        C                                                                    000000
00113        C      FR(K) IS THE COMPUTED  FAIR SHARE  REQUEST FOR PORT K         000000
00114               REAL FR(4)                                                    000000
00114        C                                                                    000000
00114        C      LL IS THE LOAD LEFT AT EACH POINT IN THE ITERATION            000000
00115               REAL LL,LOLD                                                  000000
00115        C                                                                    000000
00115        C      IF IMPL IS ZERO, THEN ASSIGN DEFAULT VALUES                   000000
00116               IF (IMPL .GT. 0) GO TO 40                                     000000
00120               RO = 0.0                                                      000002
00121               IF (PR1 .EQ. 0.99999) PR1 = 1.0                               000003
00123               IF (PR2 .EQ. 0.99999) PR2 = 2.0                               000010
00125               IF (PR3 .EQ. 0.99999) PR3 = 3.0                               000015
00127               IF (PR4 .EQ. 0.99999) PR4 = 4.0                               000022
00131               IF (MP1 .EQ. 0.99999) MP1 = 0                                 000027
00133               IF (MP2 .EQ. 0.99999) MP2 = 0                                 000033
00135               IF (MP3 .EQ. 0.99999) MP3 = 0                                 000037
00137               IF (MP4 .EQ. 0.99999) MP4 = 0                                 000043
00141               IF(P1  .EQ. .99999) P1=0.0                                    000047
```


PA

```
00143        IF(P2  .EQ. .99999) P2=0.0                                   000053
00145        IF(P3  .EQ. .99999) P3= 0.0                                  000057
00147        IF(P4  .EQ. .99999) P4=0.0                                   000063
00151        SR=0.                                                        000067
00152        PC1=0.                                                       000070
00153        PC2=0.                                                       000071
00154        PC3=0.                                                       000072
00155        PC4=0.                                                       000073
00156        TINC1= 0.5*TINC                                              000074
00157     40 CONTINUE                                                     000100
00157  C                                                                  000100
00157  C     IF THE TOTAL MAXIMUM POWER IS .LE. TOTAL POWER               000100
00157  C     REQUESTED, THEN SUBMIT REQUESTS AT MAX-POWER, SET REQUEST    000100
00157  C     SURPLUS EQUAL TO THE DIFFERENCE, AND RETURN                  000100
00160        PD = P1 + P2 + P3 + P4                                       000100
00161        IF(PR1.LE.0.0) MP1=0.                                        000104
00163        IF(PR2.LE.0.0) MP2=0.                                        000110
00165        IF(PR3.LE.0.0) MP3=0.                                        000114
00167        IF(PR4.LE.0.0) MP4=0.                                        000120
00171        MPD = MP1 + MP2 + MP3 + MP4                                  000124
00172        IF (MPD .GT. RD) GO TO 80                                    000131
00174        R1 = MP1/EF1                                                 000134
00175        R2 = MP2/EF2                                                 000137
00176        R3 = MP3/EF3                                                 000142
00177        R4 = MP4/EF4                                                 000145
00200        SP = RD - MPD                                                000150
00201        GO TO 500                                                    000153
00202     80 CONTINUE                                                     000155
00202  C                                                                  000155
00202  C     PROCEED WITH ALLOCATION ALGORITHM SINCE THE SUM OF           000155
00202  C     ALL MAXIMUM POWER INPUTS EXCEEDS THE TOTAL REQUEST RD         000155
00202  C                                                                  000155
00202  C     INITIALIZATION                                               000155
00203        LL = RD                                                      000155
00204        R1 = 0.0                                                     000156
00205        R2 = 0.0                                                     000157
00206        R3 = 0.0                                                     000160
00207        R4 = 0.0                                                     000161
00210        SP = 0.0                                                     000162
00210  C                                                                  000162
00210  C     IF THE TOTAL REQUEST (OR LOAD) IS ZERO, THEN RETURN           000162
00211        IF (RD .LE. 0.0) GO TO 500                                   000163
00213        R(1)=R1                                                      000165
00214        R(2)=R2                                                      000167
00215        R(3)=R3                                                      000171
00216        R(4)=R4                                                      000173
00217        PR(1) = PR1                                                  000175
00220        PR(2) = PR2                                                  000177
00221        PR(3) = PR3                                                  000201
00222        PR(4) = PR4                                                  000203
00223        W(1) = W1                                                    000205
00224        W(2) = W2                                                    000207
00225        W(3) = W3                                                    000211
00226        W(4) = W4                                                    000213
00227        MP(1) = MP1                                                  000215
00230        MP(2) = MP2                                                  000217
00231        MP(3) = MP3                                                  000221
```

PA

```
00232              MP(4) = MP4                                          000223
00232        C                                                          000223
00232        C                                                          000223
00232        C     ITERATE ON PRIORITY I FOR I = 1, 2, 3, 4             000223
00232        C                                                          000223
00233              DO 1000 I = 1, 4                                     000231
00233        C                                                          000231
00236              XI = I                                               000231
00236        C     OBTAIN SUM OF MAXIMUM POWER FOR PORTS WITH PRIORITY I  000231
00237              SMP(I) = 0.0                                         000234
00240              DO 100 K = 1, 4                                      000240
00243              IF (PR(K) .EQ. XI) SMP(I) = SMP(I) + MP(K)           000240
00245         100  CONTINUE                                             000247
00245        C                                                          000247
00245        C     IF NO PRIORITY-I MAXIMUM POWER EXISTS, THEN PROCEED WITH  000247
00245        C     THE NEXT HIGHER PRIORITY                             000247
00247              IF (SMP(I) .EQ. 0.0) GO TO 1000                      000247
00247        C                                                          000247
00247        C     IF THE SUM OF ALL PRIORITY-I MAXIMUM POWER .GT. LOAD  000247
00247        C     LEFT, THEN GO AROUND                                 000247
00251              IF (SMP(I) .GT. LL) GO TO 400                        000251
00251        C                                                          000251
00251        C     THE SUM OF ALL PRIORITY-I MAXIMUM POWER .LE. LOAD    000251
00251        C     LEFT, SO SUBMIT EACH PRIORITY-I REQUEST              000251
00253              DO 200 K = 1, 4                                      000260
00256              IF (PR(K) .EQ. XI) R(K) = MP(K)                      000260
00260         200  CONTINUE                                             000266
00260        C                                                          000266
00260        C     UPDATE LOAD LEFT                                     000266
00262              LL = LL - SMP(I)                                     000266
00262        C                                                          000266
00262        C     IF THE REMAINING LOAD IS ZERO, THEN EXIT THE ITERATION  000266
00263              IF (LL .LE. 0.0) GO TO 2000                          000271
00263        C                                                          000271
00263        C     OTHERWISE, PROCEED WITH NEXT HIGHER PRIORITY         000271
00265              GO TO 1000                                           000273
00265        C                                                          000273
00266         400  CONTINUE                                             000275
00266        C                                                          000275
00266        C     THE SUM OF THE PRIORITY-I MAXIMUM POWER EXCEEDS THE  000275
00266        C     LOAD LEFT, SO COMPUTE AND SUBMIT FAIR SHARE REQUESTS  000275
00266        C      TO EACH PRIORITY-I PORT                             000275
00266        C                                                          000275
00267         600  CONTINUE                                             000275
00267        C                                                          000275
00267        C     SAVE LL FOR LATER REFERENCE                          000275
00270              LOLD = LL                                            000275
00270        C                                                          000275
00270        C     DETERMINE FAIR SHARE UNITS FOR ALL PRIORITY-I        000275
00270        C     PORTS TO WHICH NO REQUEST HAS BEEN SUBMITTED         000275
00271              SW(I) = 0.0                                          000276
00272              DO 700 K = 1, 4                                      000302
00275              IF (R(K) .NE. 0.0) GO TO 700                         000302
00277              IF (PR(K) .EQ. XI) SW(I) = SW(I) + W(K)              000303
00301         700  CONTINUE                                             000313
00303              FRU = 1.0 / SW(I)                                    000313
00303        C                                                          000313
```

PA

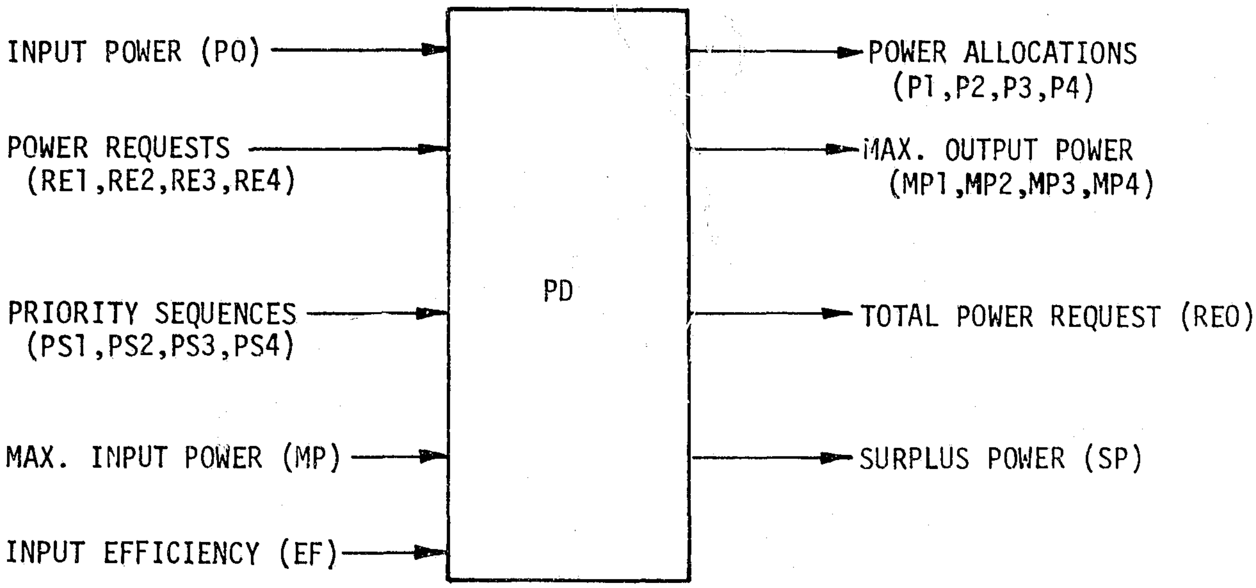
```
00303      C     FIRST, SUBMIT FAIR SHARE REQUESTS TO PORTS FOR WHICH THE           000313
00303      C     FAIR SHARE REQUEST EXCEEDS THE MAXIMUM POWER.  CONSIDER ONLY       000313
00303      C     PORTS TO WHICH NO REQUEST HAS BEEN SUBMITTED                       000313
00304            DO 800 K = 1, 4                                                    000321
00307            IF (R(K) .NE. 0.0) GO TO 800                                       000321
00311            IF (PR(K) .NE. XI) GO TO 800                                       000322
00311      C                                                                        000322
00311      C     COMPUTE FAIR SHARE                                                 000322
00313            FR(K) = (W(K) * FRU) * LL                                          000325
00313      C                                                                        000325
00313      C     IF FAIR SHARE EXCEEDS MAXIMUM POWER, THEN SUBMIT REQUEST           000325
00314            IF (FR(K) .GE. MP(K)) R(K) = MP(K)                                 000331
00314      C     - - - AND REDUCE LOAD LEFT TALLY                                   000331
00316            IF (FR(K) .GE. MP(K)) LL = LL - MP(K)                              000337
00320        800 CONTINUE                                                           000350
00320      C                                                                        000350
00320      C     IF LL .NE. LOLD, THEN LL WAS REDUCED DURING THE                    000350
00320      C     PROCESSING IN THE  DO 800  LOOP ABOVE.  THIS CHANGES               000350
00320      C     THE FAIR SHARE COMPUTATION.  IT IS THEREFORE                       000350
00320      C     NECESSARY TO GO BACK THROUGH THE  DO 800  LOOP IN                  000350
00320      C     ORDER TO RECONSIDER ANY PORT WHICH MAY NOW                         000350
00320      C     SATISFY THE REQUIREMENT THAT FR(K) .GE. MP(K).  ONLY               000350
00320      C     PRIORITY-I PORTS TO WHICH NO REQUEST HAS BEEN                      000350
00320      C     MADE ARE ELIGIBLE FOR RECONSIDERATION                              000350
00322            IF (LL .LT. LOLD) GO TO 600                                        000350
00322      C                                                                        000350
00322      C     FINALLY, SUBMIT REQUESTS TO THOSE PORTS FOR WHICH THE FAIR SHARE   000350
00322      C     .LT. THAN THEIR MAXIMUM POWER.  CONSIDER ONLY                      000350
00322      C     PRIORITY-I PORTS TO WHICH NO REQUEST HAS BEEN SUBMITTED            000350
00324            DO 900 K = 1, 4                                                    000357
00327            IF (R(K) .NE. 0.0) GO TO 900                                       000357
00331            IF (PR(K) .NE. XI) GO TO 900                                       000360
00333            R(K) = FR(K)                                                       000363
00334        900 CONTINUE                                                           000367
00336            LL=0.0                                                             000367
00337            GO TO 2000                                                         000370
00337      C                                                                        000370
00340       1000 CONTINUE                                                           000375
00340      C                                                                        000375
00342       2000 CONTINUE                                                           000375
00342      C                                                                        000375
00342      C     FINALLY, ASSIGN OUTPUTS TO NON-SUBSCRIPTED FORMAL PARAMETERS.      000375
00342      C     ALSO, MODIFY ALL REQUESTS ACCORDING TO THE INPUT EFFICIENCIES      000375
00343            R1 = R(1) / EF1                                                    000375
00344            R2 = R(2) / EF2                                                    000377
00345            R3 = R(3) / EF3                                                    000402
00346            R4 = R(4) / EF4                                                    000405
00347            SP = LL                                                            000410
00350        500 IF(IMPL.LE.1) RETURN                                               000413
00352            SRO= SR                                                            000421
00353            SR=SR+ PL*TINC1                                                    000423
00354            IF(SR.LE.0.) RETURN                                                000427
00356            SRO=SRO/SR                                                         000435
00357            SRI= TINC1*100./SR                                                 000440
00360            PC1= PC1*SRO + P1*SRI                                              000444
00361            PC2= PC2*SRO + P2*SRI                                              000450
00362            PC3= PC3*SRO + P3*SRI                                              000456
```


PA

```
00363    252*        PC4= PC4*SRO + P4*SRI                              000464
00364    253*        RETURN                                             000472
00365    254*        END                                                000745
```


PA

7.28 POWER DIVIDER



This component allocates power to four ports plus surplus based on priority 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 the ports of highest 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 among 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,...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 1.2.2 and 7c for further discussion.)

Inputs[1]

Parameter/Port		Description	Units
P	0	Input power	kw
RE	1,2,3,4	Power request of output ports	kw
PS	1,2,3,4	Priority sequence (default = 1,2,3,4)	kw
F	1,2,3,4	Allocation weight (for equal priorities)	-
MP		Maximum input power (default = PO)	kw
EF		Input efficiency	-

Outputs

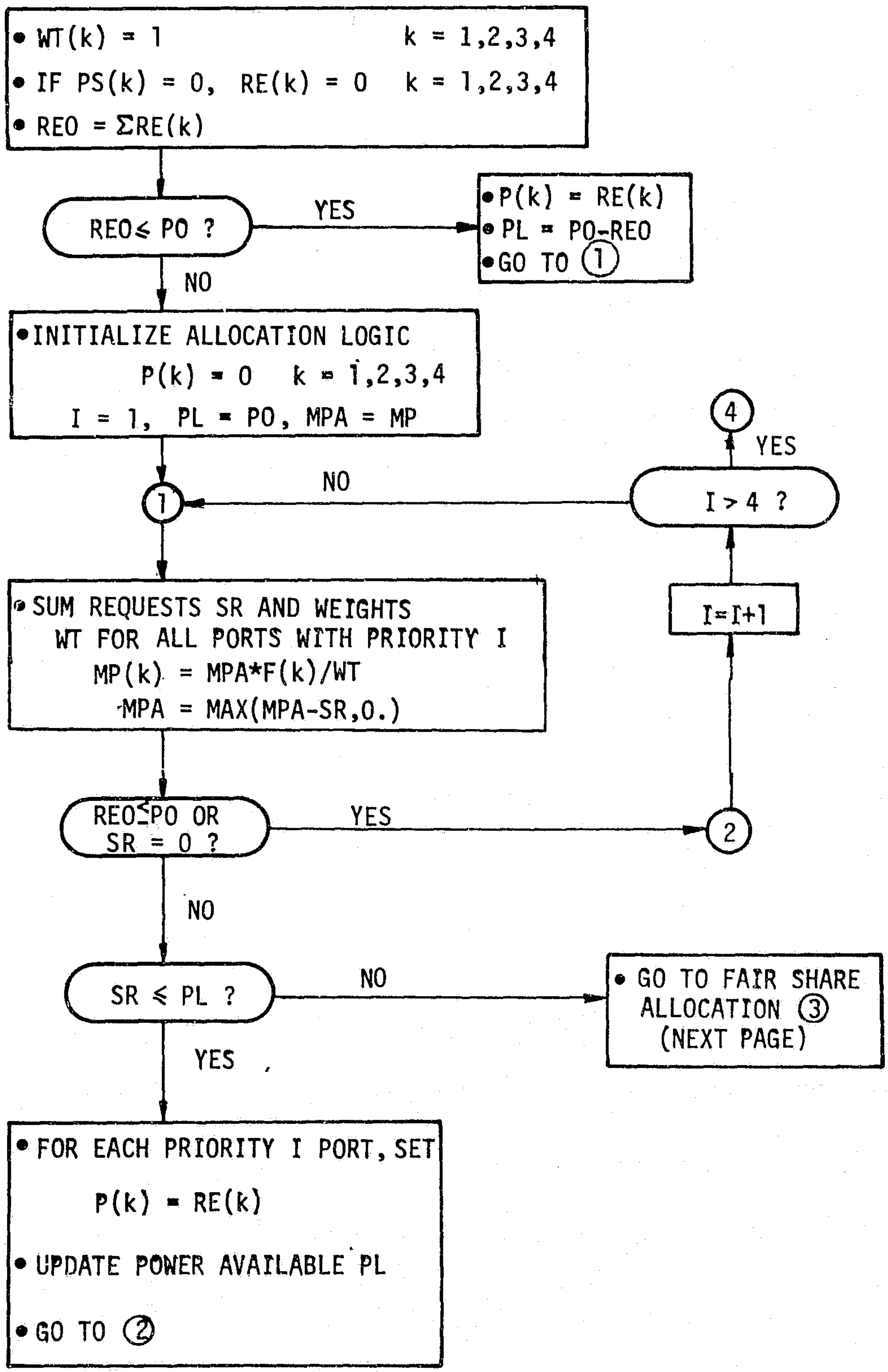
Variable/Port			
P	1,2,3,4	Output power for port i	kw
RE	0	Output power request	kw
MP	1,2,3,4	Output maximum power based on MP	kw

Statistics

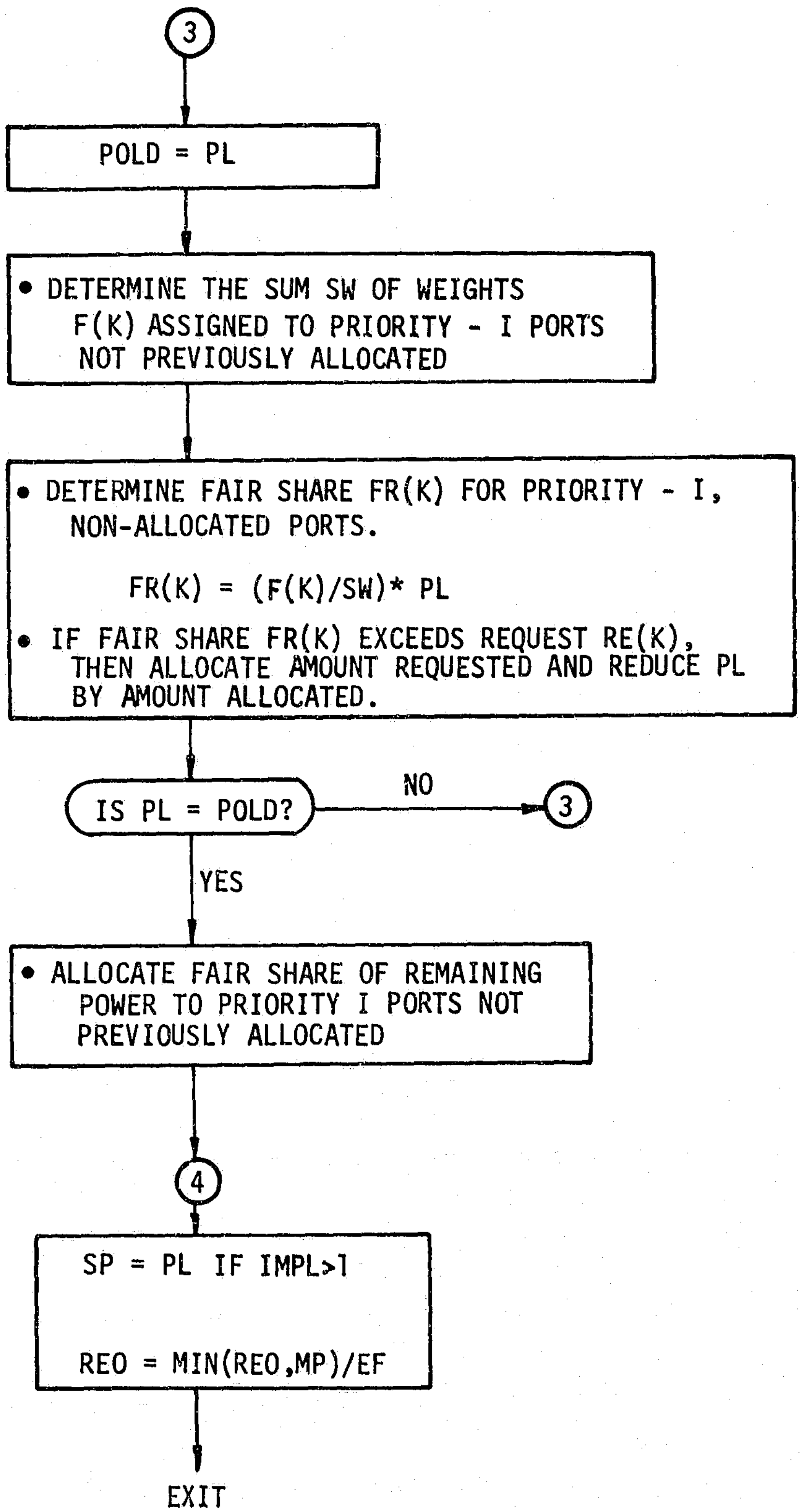
SP		Surplus power	kw

[1] No capital costs assigned since this is an allocation component, not a physical device.

CALCULATION LOGIC



PD FAIR SHARE ALLOCATION



```
SUBROUTINE PD        ENTRY POINT 000453

STORAGE USED: CODE(1) 000631; DATA(0) 000105; BLANK COMMON(2) 000000

 COMMON BLOCKS:

 0003   CIMPL  000001

EXTERNAL REFERENCES (BLOCK, NAME)

 0004   NERR3$

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

 0001     000402 1000L      0001     000166 206G      0001     000176 214G      0001     000274 246G      0001     000314 263G
 0001     000333 275G       0001     000370 315G      0001     000033 40L       0001     000307 400L      0001     000105 60L
 0001     000307 600L       0001     000324 700L      0001     000077 80L       0001     000361 800L      0001     000377 900L
 0000 R 000030 FR           0000 R 000043 FRU         0000 I 000036 I           0003 I 000000 IMPL        0000     000052 INJP$
 0000 I 000041 K            0000 R 000000 P           0000 R 000034 PL          0000 R 000035 PMA         0000 R 000042 POLD
 0000 R 000010 PR           0000 R 000004 R           0000 R 000024 SR          0000 R 000020 SW          0000 R 000014 W
 0000 R 000040 WT           0000 R 000037 XI


00100       CPD                                                                  000000
00101              SUBROUTINE PD(                                                000000
00101            1      P1, P2, P3, P4,                                          000000
00101            2      R0,                                                      000000
00101            3      SP,PM1,PM2,PM3,PM4,                                      000000
00101            4      P0,                                                      000000
00101            5      R1, R2, R3, R4,                                          000000
00101            6      PR1, PR2, PR3, PR4,                                      000000
00101            7      W1, W2, W3, W4, PM,EF)                                   000000
00101       C                                                                    000000
00101       C     PURPOSE.   MODEL POWER DIVIDER                                 000000
00101       C                                                                    000000
00101       C     METHOD.   PRIMARY FLOW ALLOCATION RESULTING FROM PRIORITY      000000
00101       C     ASSIGNMENTS.   SECONDARY FLOW ALLOCATION RESULTING             000000
00101       C     FROM WEIGHT ASSIGNMENTS.                                       000000
00101       C     THAT IS, TOTAL AVAILABLE POWER IS ALLOCATED                    000000
00101       C     ACCORDING TO:                                                  000000
00101       C       *  PORT REQUESTS                                             000000
00101       C       *  PORT PRIORITY (HIGHEST PRIORITY = 1)                      000000
00101       C       *  PORT WEIGHTS (IN CASE OF EQUAL PRIORTIES)                 000000
00101       C                                                                    000000
00101       C     ALLOCATION SCHEME.                                             000000
00101       C     IS SUM OF ALL REQUESTS .LT. POWER AVAILABLE P0\                000000
00101       C     YES.                                                           000000
00101       C       FULFILL EACH REQUEST                                         000000
```

PD

```
00101   C        UPDATE POWER AVAILABLE                                                000000
00101   C        EXIT                                                                  000000
00101   C      NO.                                                                     000000
00101   C        IS SUM OF ALL PRIORITY-1 REQUESTS .LT. PD\                            000000
00101   C        YES.                                                                  000000
00101   C          FULFILL EACH PRIORITY-1 REQUEST                                     000000
00101   C          UPDATE POWER AVAILABLE (TO PL)                                      000000
00101   C          GO ON TO PRIORITY-2 REQUESTS                                        000000
00101   C        NO.                                                                   000000
00101   C          ALLOCATE  FAIR SHARE  TO EACH PRIORITY-1 PORT                       000000
00101   C          EXIT.                                                               000000
00101   C          IS SUM OF ALL PRIORITY-2 REQUESTS .LT. PL                           000000
00101   C                                                                              000000
00101   C       AND SO ON AND SO FORTH                                                 000000
00101   C                                                                              000000
00101   C      FORMAL ARGUMENT DEFINITION.                                             000000
00101   C      P1,...., P4 %   POWER ALLOCATIONS IN KW  (OUTPUTS)                      000000
00101   C      RC %           TOTAL POWER REQUESTED    (OUTPUT)                        000000
00101   C      SP  %           SURPLUS POWER            (OUTPUT)                       000000
00101   C      PM1,....,PM4%  PORT MAXIMUM OUTPUT POWER IN KW       (OUTPUT)           000000
00101   C      PD  %           TOTAL POWER INPUT IN KW  (INPUT)                        000000
00101   C      PM  %           MAXIMUM INPUT POWER IN KW  (INPUT)                      000000
00101   C       EF  %          INPUT EFFICIENCY  (INPUT)                               000000
00101   C      R1,...., R4 %   PORT REQUESTS IN KW      (INPUTS)                       000000
00101   C      PR1,...., PR4 % PORT PRIORITIES          (INPUTS)                       000000
00101   C      W1,...., W4 %   PORT WEIGHTS            (INPUTS)                        000000
00101   C                                                                              000000
00101   C       COMMON STORAGE                                                         000000
00103           COMMON/ CIMPL / IMPL                                                   000000
00103   C                                                                              000000
00103   C      LOCAL VARIABLES                                                         000000
00103   C                                                                              000000
00103   C      P(K) IS THE POWER ALLOCATED TO PORT K                                   000000
00104          REAL P(4)                                                               000000
00104   C                                                                              000000
00104   C      R(K) IS THE POWER REQUEST AT PORT K                                     000000
00105          REAL R(4)                                                               000000
00105   C                                                                              000000
00105   C      PR(K) IS THE PRIORITY ASSIGNED TO PORT K                                000000
00106          REAL PR(4)                                                              000000
00106   C                                                                              000000
00106   C      W(K) IS THE WEIGHT ASSIGNED TO PORT K                                   000000
00107          REAL W(4)                                                               000000
00107   C                                                                              000000
00107   C      SW(I) IS THE SUM OF THE WEIGHTS ASSIGNED TO PRIORITY-I PORTS            000000
00110          REAL SW(4)                                                              000000
00110   C                                                                              000000
00110   C      SR(I) IS THE SUM OF THE REQUESTS AT PRIORITY-I PORTS                    000000
00111          REAL SR(4)                                                              000000
00111   C                                                                              000000
00111   C      FRU IS  FAIR SHARE  UNIT FOR PRIORITY-I PORTS                           000000
00111   C                                                                              000000
00111   C      FR(K) IS THE COMPUTED  FAIR SHARE  ALLOCATION TO PORT K                 000000
00112          REAL FR(4)                                                              000000
00112   C                                                                              000000
00112   C      PL IS THE POWER LEFT AT EACH POINT IN THE ITERATION                     000000
00113          REAL PL                                                                 000000
```

PD

```
00113      C                                                              000000
00113      C       IF IMPL IS ZERO, THEN ASSIGN DEFAULT VALUES            000000
00114              IF (IMPL .GT. 0) GO TO 40                              000000
00116              R1 = 0.0                                               000002
00117              R2 = 0.0                                               000003
00120              R3 = 0.0                                               000004
00121              R4 = 0.0                                               000005
00122              IF (PR1 .EQ. 0.99999) PR1 = 1.0                        000006
00124              IF (PR2 .EQ. 0.99999) PR2 = 2.0                        000013
00126              IF (PR3 .EQ. 0.99999) PR3 = 3.0                        000020
00130              IF (PR4 .EQ. 0.99999) PR4 = 4.0                        000025
00130      C                                                              000025
00132           40 CONTINUE                                               000033
00132      C                                                              000033
00132      C       IF THE TOTAL POWER REQUESTED IS .LE. TOTAL POWER       000033
00132      C       INPUT, THEN SATISFY REQUESTS, SET POWER SURPLUS        000033
00132      C       EQUAL TO THE DIFFERENCE,                               000033
00133              IF(PR1.LE.0.0) R1=0.0                                  000033
00135              IF(PR2.LE.0.0) R2=0.0                                  000036
00137              IF(PR3.LE.0.0) R3=0.0                                  000042
00141              IF(PR4.LE.0.0) R4=0.0                                  000046
00143              RO = R1 + R2 + R3 + R4                                 000052
00144              IF (RO .GT. PO) GO TO 80                               000057
00146              P1 = R1                                                000062
00147              P2 = R2                                                000064
00150              P3 = R3                                                000066
00151              P4 = R4                                                000070
00152              PL = PO -RO                                            000072
00153              GO TO 60                                               000075
00154           80 CONTINUE                                               000077
00154      C                                                              000077
00154      C       PROCEED WITH ALLOCATION ALGORITHM SINCE THE SUM OF     000077
00154      C       ALL REQUESTS EXCEEDS THE TOTAL AVAILABLE POWER PO      000077
00154      C                                                              000077
00154      C       INITIALIZATION                                         000077
00155              PL = PO                                                000077
00156              P1 = 0.0                                               000100
00157              P2 = 0.0                                               000101
00160              P3 = 0.0                                               000102
00161              P4 = 0.0                                               000103
00161      C                                                              000103
00162           60 PMA= PM                                                000105
00163              IF( PM .EQ. 0.99999)   PMA = PO                        000106
00165              P(1) = P1                                              000122
00166              P(2) = P2                                              000124
00167              P(3) = P3                                              000126
00170              P(4) = P4                                              000130
00171              R(1) = R1                                              000132
00172              R(2) = R2                                              000134
00173              R(3) = R3                                              000136
00174              R(4) = R4                                              000140
00175              PR(1) = PR1                                            000142
00176              PR(2) = PR2                                            000144
00177              PP(3) = PR3                                            000146
00200              PR(4) = PR4                                            000150
00201              W(1) = W1                                              000152
00202              W(2) = W2                                              000154
```


PD

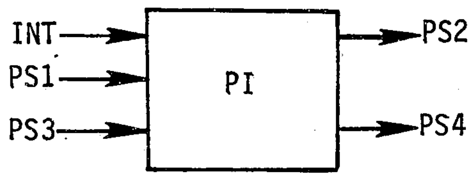
```
00203            W(3) = W3                                                 000156
00204            W(4) = W4                                                 000160
00204      C                                                               000160
00204      C                                                               000160
00204      C     ITERATE ON PRIORITY I FOR I = 1, 2, 3, 4                  000160
00204      C                                                               000160
00205            DO 1000 I = 1, 4                                          000166
00205      C                                                               000166
00210            XI = I                                                    000166
00210      C     OBTAIN SUM OF REQUESTS FROM PORTS WITH PRIORITY I         000166
00211            SR(I) = 0.0                                               000171
00212            WT=0.0                                                    000172
00213            DO 100 K = 1, 4                                           000175
00216            IF (PR(K) .EQ. XI) SR(I) = SR(I) + R(K)                   000176
00220            IF(PR(K) .EQ. XI) WT= WT+ W(K)                            000203
00222        100 CONTINUE                                                  000213
00222      C                                                               000213
00224            IF(PR1 .EQ. XI) PM1= PMA*W1/WT                            000213
00226            IF( PR2.EQ. XI) PM2= PMA*W2/WT                            000222
00230            IF(PR3 .EQ. XI) PM3= PMA*W3/WT                            000231
00232            IF(PR4 .EQ. XI) PM4= PMA*W4/WT                            000240
00234            PMA= AMAX1( PMA- SR(I),0.)                                000247
00235            IF(PL.LE.0.)GO TO 1000                                    000256
00235      C                                                               000256
00235      C     IF NO PRIORITY-I REQUESTS EXIST, THEN PROCEED WITH        000256
00235      C     THE NEXT HIGHER PRIORITY                                  000256
00237            IF (SR(I) .EQ. 0.0) GO TO 1000                            000261
00241            IF(RU.LE.PU) GO TO 1000                                   000263
00241      C                                                               000263
00241      C     IF THE SUM OF ALL PRIORITY-I REQUESTS .GT. POWER          000263
00241      C     AVAILABLE, THEN GO AROUND                                 000263
00243            IF (SR(I) .GT. PL) GO TO 400                              000265
00243      C                                                               000265
00243      C     THE SUM OF ALL PRIORITY-I REQUESTS .LE. POWER             000265
00243      C     AVAILABLE, SO FULFILL EACH PRIORITY-I REQUEST             000265
00245            DO 200 K = 1, 4                                           000274
00250            IF (PR(K) .EQ. XI) P(K) = R(K)                            000274
00252        200 CONTINUE                                                  000302
00252      C                                                               000302
00252      C     UPDATE POWER AVAILABLE                                    000302
00254            PL = PL - SR(I)                                           000302
00254      C                                                               000302
00255            GO TO 1000                                                000305
00255      C                                                               000305
00256        400 CONTINUE                                                  000307
00256      C                                                               000307
00256      C     THE SUM OF THE PRIORITY-I REQUESTS EXCEEDS THE            000307
00256      C     POWER AVAILABLE, SO COMPUTE AND ALLOCATE FAIR             000307
00256      C     SHARE TO EACH PRIORITY-I PORT                             000307
00256      C                                                               000307
00257        600 CONTINUE                                                  000307
00257      C                                                               000307
00257      C     SAVE PL FOR LATER REFERENCE                               000307
00260            POLD = PL                                                 000307
00260      C                                                               000307
00260      C     DETERMINE FAIR SHARE UNITS FOR ALL PRIORITY-I             000307
00260      C     PORTS FOR WHICH NO ALLOCATION HAS BEEN MADE               000307
```


PD

```
00261          SW(I) = 0.0                                              000310
00262          DO 700 K = 1, 4                                          000314
00265          IF (P(K) .NE. 0.0) GO TO 700                             000314
00267          IF (PR(K) .EQ. XI) SW(I) = SW(I) + W(K)                  000315
00271      700 CONTINUE                                                 000325
00273          FRU = 1.0 / SW(I)                                        000325
00273    C                                                              000325
00273    C     FIRST, ALLOCATE FAIR SHARE TO PORTS FOR WHICH THE        000325
00273    C     FAIR SHARE EXCEEDS THE REQUEST.  CONSIDER ONLY PRIORITY-I 000325
00273    C     PORTS, AND CONSIDER ONLY PORTS TO WHICH NO ALLOCATION    000325
00273    C     HAS YET BEEN MADE                                        000325
00274          DO 800 K = 1, 4                                          000333
00277          IF (P(K) .NE. 0.0) GO TO 800                             000333
00301          IF (PR(K) .NE. XI) GO TO 800                             000334
00301    C                                                              000334
00301    C     COMPUTE FAIR SHARE                                       000334
00303          FR(K) = (W(K) * FRU) * PL                                000337
00303    C                                                              000337
00303    C     IF FAIR SHARE EXCEEDS REQUEST, THEN FULFILL REQUEST      000337
00304          IF (FR(K) .GE. R(K)) P(K) = R(K)                         000343
00304    C     - - - AND REDUCE AVAILABLE POWER                         000343
00306          IF (FR(K) .GE. R(K)) PL = PL - P(K)                      000351
00310      800 CONTINUE                                                 000362
00310    C                                                              000362
00310    C     IF PL .NE. POLD, THEN PL WAS REDUCED DURING THE          000362
00310    C     PROCESSING IN THE  DO 800  LOOP ABOVE.  THIS CHANGES     000362
00310    C     THE FAIR SHARE COMPUTATION.  IT IS THEREFORE             000362
00310    C     NECESSARY TO GO BACK THROUGH THE  DO 800  LOOP IN        000362
00310    C     ORDER TO RECONSIDER ANY PORT WHICH MAY NOW               000362
00310    C     SATISFY THE REQUIREMENT THAT FR(K) .GE. R(K).  ONLY      000362
00310    C     PRIORITY-I PORTS FOR WHICH NO ALLOCATION HAS BEEN        000362
00310    C     MADE ARE ELIGIBLE FOR RECONSIDERATION                    000362
00312          IF (PL .NE. POLD) GO TO 600                              000362
00312    C                                                              000362
00312    C     FINALLY, ALLOCATE POWER TO THOSE PORTS REQUESTING        000362
00312    C     MORE THAN THEIR FAIR SHARE.  CONSIDER ONLY               000362
00312    C     PRIORITY-I PORTS FOR WHICH NO ALLOCATION HAS BEEN MADE   000362
00314          DO 900 K = 1, 4                                          000370
00317          IF (P(K) .NE. 0.0) GO TO 900                             000370
00321          IF (PR(K) .NE. XI) GO TO 900                             000371
00323          P(K) = FR(K)                                             000374
00324      900 CONTINUE                                                 000400
00326          PL = 0.0                                                 000400
00326    C                                                              000400
00327     1000 CONTINUE                                                 000404
00327    C                                                              000404
00327    C                                                              000404
00327    C     FINALLY, ASSIGN OUTPUTS TO NON-SUBSCRIPTED               000404
00327    C     FORMAL PARAMETERS                                        000404
00331          P1 = P(1)                                                000404
00332          P2 = P(2)                                                000406
00333          P3 = P(3)                                                000410
00334          P4 = P(4)                                                000412
00335          IF( IMPL .GT. 1)   SP = PL                               000414
00337          RO=AMIN1(RO,PM)/EF                                       000422
00340          RETURN                                                   000431
00341          END                                                      000630
```

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

Parameter/Port		Description
PS	1	Input priority for PS2 output (0 to 4)
PS	3	Input priority for PS4 output (default=PS1)
PMX		Maximum priority for PS2 (default = 1)
INT		Interrupt flag (0,-1,1)

Outputs

Variable/Port		
PS	2	Output priority for charge cycle
PS	4	Output priority for discharge cycle

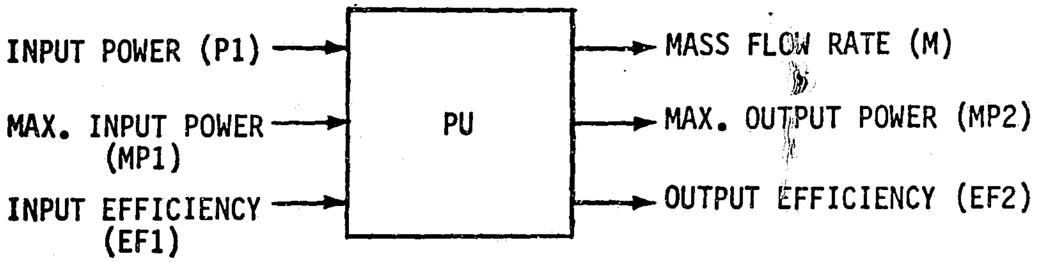
Equations

$$PS2 = PS1 \quad \text{if } INT=0$$
$$PS2 = PMX \quad \text{if } INT>0$$
$$PS2 = 0 \quad \text{if } INT<0$$
$$PS4 = PS3 \quad \text{if } INT\leq 0$$
$$PS4 = 0 \quad \text{if } INT>0$$

```
SUBROUTINE PI          ENTRY POINT 000045

STORAGE USED: CODE(1) 000072; DATA(0) 000010; BLANK COMMON(2) 000000

 COMMON BLOCKS:

 0003   CIMPL  000001

EXTERNAL REFERENCES (BLOCK, NAME)

 0004   NERR3S

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

 0001   000015 10L       0003 I 000000 IMPL      0000   000002 INJPS


00100      1*      CPI                                                                      000000
00101      2*            SUBROUTINE PI(PS2,PS4,PS1,PS3,PMX,INT)                             000000
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              PMX - MAXIMUM PRIORITY FOR PS2                            000000
00101     14*      C              INT - INTERRUPT FLAG                                      000000
00101     15*      C                     0= NO INTERRUPT                                    000000
00101     16*      C                     1= INCREASE ALLOCATION PRIORITY                    000000
00101     17*      C                    -1= DECREASE ALLOCATION PRIORITY                    000000
00101     18*      C                                                                        000000
00103     19*            REAL INT                                                           000000
00104     20*            COMMON /CIMPL/IMPL                                                 000000
00105     21*            IF(IMPL.GT.0) GO TO 10                                             000000
00107     22*            IF(PS3.EQ. .99999) PS3=PS1                                         000002
00111     23*            IF(PMX.EQ..99999)PMX=1.                                            000007
00111     24*      C                                                                        000007
00113     25*         10 PS2=PS1                                                            000015
00114     26*            PS4=PS3                                                            000016
00115     27*            IF(INT.GT.0.) PS2=PMX                                              000020
00117     28*            IF(INT.LT.0.) PS2=0.                                               000025
00121     29*            IF(INT.GT.0) PS4= 0.                                               000031
00123     30*            RETURN                                                             000035
00124     31*            END                                                                000071
```

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-EFD)*SQRT(MD/M)

where C1, C2 are conversion constants

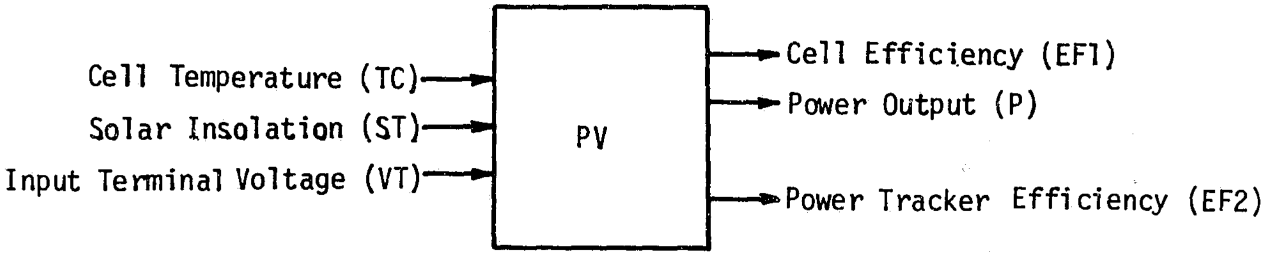
Revision Pages

Section 7.30A - PV

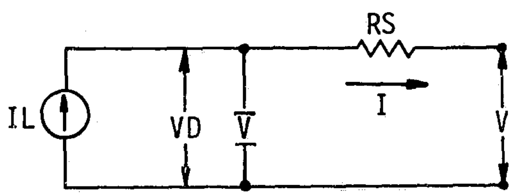
Insert revision pages 272A - 272N between pages 272 and 273 of the original document.

PRECEDING PAGE BLANK NOT FILMED

7.30A SOLAR-PHOTOVOLTAIC ARRAY



The photovoltaic cell is modeled by the circuit below. Power is delivered at terminal voltage V and is dependent on the cell temperature and insolation. Default for V is the maximum power point. A square array of solar cells is assumed with both parallel and series connections.



BASIC SOLAR CELL MODEL

Basic Equations

Output current I as a function of terminal voltage V is given by the implicit relation

I = IL + IØ*(1-EXP((V+I*RS)*QBK/(T+273))) (1)

where

IL = light current (amps)

IØ = diode reverse saturation current (amps)

T = temperature (^{o}C)

RS = internal resistance (ohms)

QBK = device constant (default = electron charge/Boltzmann's constant)

The light current IL is computed by a bivariate expansion of insolation and cell temperature. It has been reported that this model fits observed solar cell characteristics within 5% at high temperatures and insolations and within less than 1% under more moderate conditions (ref. 2). The reverse saturation current IØ is given by

```
IØ(T) = KD*AO*((T+273)**3)EXP(-EGO/(T+273))
```
(2)

where

KD = a device constant

AO = a material constant

EGO = band gap at $0^{o}K$/Boltzmann's constant

Tables	Description	Units
EFF	Efficiency of maximum power tracker versus fractional load (default table provided)	-
ØP	Optimum cell power versus insolation and temperature (computed table)	kw
ØV	Optimum cell voltage versus insolation and temperature (computed table)	volts

Inputs/Port	Description	Units
VT	Array terminal voltage (default = maximum power voltage)	volts
TC	Cell temperature	^oC
TL*	Low temperature value (default = 28)	^oC
TH*	High temperature value (default = 120)	^oC
TR	Temperature range (default = TH)	^oC
ST	Collector solar insolation	w/m^2
SL*	Low insolation value (default = 1000)	w/m^2
SH*	High insolation value (default = 25000)	w/m^2
SR	Insolation range (default = SH)	w/m^2
RC	Concentration ratio (default = 25)	-
AA	Total illuminated cell area (default = .00015*NS*NP)	m^2
NS	Number of cells in series (default = 300)	-
NP	Number of cells in parallel (default = 500)	-
I1*	Cell short circuit current at TL,SL (default = .06)	Amps
I2*	Cell short circuit current at TL,SH (default = 1.5)	Amps
I3*	Cell short circuit current at TH,SL (default = .06)	Amps

*These inputs may be ignored if IL1,DS,DT,DST,KD coefficients are supplied.

Inputs/Port (cont'd)	Description	Units
I4*	Cell short circuit current at TH,SH (default = 1.56)	Amps
V1*	Cell open circuit voltage at TL,SL (default = .6)	Volts
RS	Cell internal resistance (default = .055)	Ohms
AO	Material constant (default = 1.54E33 for silicon)	-
EGO	Band-gap at $0^{o}K$ normalized by Boltzmann's constant (default = 1.4E4 for silicon)	^{o}K
IL1	Coefficients in bivariate expansion for the	m^2V^{-1}
DS	light current IL. If not provided, they	m^2W^{-1}
DT	will be computed from the inputs I1,...,I4,	$1/^{o}C$
DST		$m^2/w^{o}C$
KD	Device constant, if not provided will be computed from I1,V1	-
CF	Lens radiation transmission coefficient	-
QBK	Device constant (default = 1.161E4)	$^{o}K/V$
RAP	Rated power of maximum power point tracker (default computed)	kw
CC	Capital cost/year/unit cell area	$\$/m^2$
CM	Maintenance cost/year	$

Note: Minimum input parameters to specify PV are cell area AA, number of cells in series NS and in parallel NP, concentration ratio RC, and rated power RAP. These parameters must be consistent with those for the collector model FO or FP.

*These inputs may be ignored if IL1,DS,DT,DST,KD coefficients are supplied.

Output/Port	Description	Units
V	Array terminal voltage	Volts
P	Array output power	kw
I	Array output current	Amps
EF1	Solar cell efficiency	-
EF2	Maximum power tracker efficiency	-
Statistics		
SP	Sum of energy delivered	kwh

Calculation Sequence

First Pass

1) Compute parameter KD (if not input)

KD = I1/[AO*((TL+273)**3)*EXP(-EGO/(TL+273))*(EXP(QBK*V1/(TL+273))-EXP(QBK*I1*RS/(TL+273)))]

2) Compute coefficients IL1,DS,DT,DST (if not input) in the light current bivariate expansion in temperature T and insolation S:

IL = IL1*S*(1+DS*(S-SL)+DT*(T-TL)+DST*(S-SL)*(T-TL)) (3)

Define

FIL(I,T) = I-IØ(T)*(1-EXP(QBK*I*RS/(T+273))).

Then

IL1 = FIL (I1,TL)/SL

DS = (FIL(I2,TL) - IL1*SH)/(IL1*SH*(SH-SL))

DT = (FIL(I3,TH) - IL1*SL)/(IL1*SL*(TH-TL))

DST = (FIL(I4,TH) - IL1*SH-IL1*SH*DS*(SH-SL) - IL1*SH*DT*(TH-TL))/(IL1*SH*(SH-SL)*(TH-TL)).

3) If a terminal voltage VT is not input, calculate the optimal cell voltage V=ØV(S,T) with S ranging through 10 values equally spaced between 0 and SR, and with T ranging through 10 values equally spaced between 0 and TR, resulting in a 10 x 10 matrix ØV(S,T). The calculation is as follows: Given S and T, the open circuit voltage VOC is given by

VOC = (T+273)*ALOG(1+IL/IØ)/QBK,

where IL and IØ are computed from (2) and (3).

A binary search is performed in the range from 0 to VOC. For a value V in this range, Newton-Raphson iterations are used to solve for the terminal current I satisfying (1). The corresponding power P (in kw) is

P = I*V/1000 .

The iterative search process to maximize P is given by

(i) Take the initial interval [VL,VH] to be [0,VOC].

(ii) Compute a numerical derivative of P at the midpoint VM of [VL,VH]:

P' = (P(VM+1E-5)-P(VM))/1E-5

(iii) If P' ≥ 0, set VL = VM.

If P' < 0, set VH = VM.

(iv) IfVH-VL > 2E-5 and the number of iterations performed is 10, go to (ii). Otherwise P is maximized and

ØV(S,T) = VM

ØP(S,T) = P

The 10 x 10 matrices ØV(S,T) (optimal cell voltage) and ØP(S,T) (maximal cell power) are stored for use in subsequent passes.

Subsequent Passes

4) Compute insolation S at the cells

S = ST*RC*CF

5) If terminal voltage VT is not input, the cell terminal voltage V and power P are obtained by interpolation from the arrays ØV(S,T) and ØP(S,T). (A diagnostic is printed if S > SR or TC > TR).

6) If VT is used as an input voltage, then the cell voltage and power are determined using

V = VT/NS

I = IL(S,TC) + IØ(TC)*(1-EXP(QBK*(V+I*RS)/(TC+273)))

P = I*V/1000

7) Array outputs prior to maximum power tracker:

V = V*NS

P = P*NS*NP

I = P*1000/V

EF1 = P*1000/(S*AA)   if S > 0

EF2 = 1.

8) If the maximum power tracker is used,

EF2 = EFF(P/RAP)

P = P*EF2

REFERENCES FOR PV

1. J. K. Linn, "Photovoltaic System Analysis Program-SOLCEL," Sandia Laboratories Report SAND77-1268, 1977.

2. L. H. Goldstein and G. R. Case, "PVSS-A Photovoltaic System Simulation Program," Sandia Laboratories, 1976.

BCS 40180-2 Rev.

```
SUBROUTINE PV        ENTRY POINT 001350

STORAGE USED: CODE(1) 001715; DATA(0) 000175; BLANK COMMON(2) 000000

 COMMON BLOCKS:

 0003   CIMPL  000003
 0004   CTIME  000001
 0005   CSIMUL 000010
 0006   COST   000003

EXTERNAL REFERENCES (BLOCK, NAME)

 0007   AINR
 0010   TBLU2
 0011   TBLU1
 0012   EXP
 0013   ALOG
 0014   NWDU$
 0015   NIO2$
 0016   NERR3$

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

 0001   000772 1COL     0001   000042 11L      0001   000037 126G     0001   000517 223G     0001   000544 232G
 0001   000567 236G     0001   000644 246G     0001   000724 6L       0000   000047 808F     0001   001050 809L
 0001   001121 900L     0001   001202 901L     0001   001273 904L     0000 R 000030 AIL      0007 R 000000 AINR
 0000 R 000031 BIO      0006 R 000000 CCAP     0006 R 000001 CMA      0006   000002 COP      0005 R 000000 DUM
 0000 R 000002 EFF1     0003 I 000001 ICNT     0000 I 000022 II       0000 I 000043 IKJ      0000 I 000044 IKJ0
 0000 R 000000 IM       0000 R 000001 IME      0003 I 000000 IMPL     0000   000123 INJP$    0003   000002 ITEST
 0000 I 000024 J        0000 I 000025 JC       0000 I 000026 K        0000 I 000035 M        0000 I 000046 NEF
 0000 R 000040 PM       0000 R 000041 PME      0000 R 000042 PMP      0000 R 000045 PRAT     0000 R 000023 S
 0000 R 000027 T        0011 R 000000 TBLU1    0010 R 000000 TBLU2    0004 R 000000 TIME     0000 R 000021 TINC1
 0005 R 000007 TMAX     0000 R 000020 TMAX1    0000 R 000034 VH       0000 R 000033 VL       0000 R 000036 VM
 0000 R 000037 VME      0000 R 000032 VOC


00100        CPV                                                                       000013
00101              SUBROUTINE PV(EFF,OP,OV,V,P,I,EF1,EF2,SP,                            000013
00101             1VT,TC,TL,TH,TR,ST,SL,SH,SR,RC,AA,NS,NP,                              000013
00101             2I1,I2,I3,I4,V1,RS,AO,EGC,IL1,DS,DT,DST,KD,                          000013
00101             3CF,QBK,RAP,CC,CM)                                                    000013
00101        C                                                                          000013
00101        C       PURPOSE     THIS COMPONENT COMPUTES THE POWER AND VOLTAGE          000013
00101        C                   OUTPUT OF A PHOTO-VOLTAIC CELL ARRAY GIVEN THE         000013
00101        C                   TEMPERATUPE AND INSOLATION                             000013
00101        C       WRITTEN BY Y.K.CHAN, 10-21-78, VERSION 1                           000013
00101        C                                                                          000013
```

2721

PV

272J

```
00101   C      METHOD    NEWTON RALPHSON METHOD IS USED TO CALCULATE CELL            000013
00101   C                CURRENT AS FUNCTION OF INSOLATION, TEMPERATURE, AND          000013
00101   C                TERMINAL VOLTAGE.  IF TERMINAL VOLTAGE IS NOT INPUT,         000013
00101   C                POWER IS COMPUTED AT OPTIMAL VOLTAGE.  THIS IS DONE          000013
00101   C                FOR A RANGE OF 10 VALUES OF TEMPERATURE AND 10               000013
00101   C                VALUES OF INSOLATION IN THE FIRST PASS.                      000013
00101   C                  AT SUBSEQUENT PASSES,                                      000013
00101   C                INTERPOLATION IS USED.                                       000013
00101   C                                                                             000013
00101   C       CALL SEQUENCE                                                         000013
00101   C           TABLES                                                            000013
00101   C              EFF   -EFFICIENCY OF MAXIMUM POWER TRACKER                     000013
00101   C                     VS FRACTIONAL LOAD (DEFAULT TABLE)                      000013
00101   C              OP    -OPTIMAL POWER,KW, VS INSOLATION,W/M2, AND               000013
00101   C                     TEMPERATURE,C                                           000013
00101   C              OV    -OPTIMAL TERMINAL VOLTAGE,V, VS INSOLATION,W/M2, AND     000013
00101   C                     TEMPERATURE,C                                           000013
00101   C           OUTPUTS                                                           000013
00101   C              V     -ARRAY TERMINAL VOLTAGE,VOLTS                            000013
00101   C              P     -ARRAY OUTPUT POWER,KW                                   000013
00101   C              I     -ARRAY OUTPUT CURRENT,AMPS                               000013
00101   C              EF1   -SOLAR CELL EFFICIENCY                                   000013
00101   C              EF2   -MAXIMUM POWER TRACKER EFFICIENCY                        000013
00101   C           STATISTICS                                                        000013
00101   C              SP    -SUM OF POWER DELIVERED,KW                               000013
00101   C           INPUTS                                                            000013
00101   C               VT     -ARRAY TERMINAL VOLTAGE,VOLTS,(DEFAULT=MAXIMUM         000013
00101   C                     POWER VOLTAGE)                                          000013
00101   C               TC     -CELL TEMPERATURE,C                                    000013
00101   C               TL     -LOW TEMPERATURE VALUE,C,(DEFAULT=28)                  000013
00101   C               TH     -HIGH TEMPERATURE VALUE,C,(DEFAULT=120)                000013
00101   C               TR     -TEMPERATURE RANGE,C,(DEFAULT=TH)                      000013
00101   C               ST     -COLLECTOR SOLAR INSOLATION,W/M2                       000013
00101   C               SL     -LOW INSOLATION VALUE,W/M2,(DEFAULT=1000)              000013
00101   C               SH     -HIGH INSOLATION VALUE,W/M2,(DEFAULT=25000)            000013
00101   C               SR     -INSOLATION RANGE,W/M2,(DEFAULT=SH)                    000013
00101   C               RC     -CONCENTRATION RATIO(DEFAULT=25)                       000013
00101   C               AA     -TOTAL COLLECTOR CELL AREA,M2,(DEFAULT=2.5E-3)         000013
00101   C               NS     -NUMBER OF CELLS IN SERIES(DEFAULT=300)                000013
00101   C               NP     -NUMBER OF CELLS INPARALLEL(DEFAULT=500)               000013
00101   C               I1     -CELL SHORT CIRCUIT CURRENT AT TL,SL, AMPS             000013
00101   C                       (DEFAULT=.06)                                         000013
00101   C               I2     -CELL SHORT CIRCUIT CURRENT AT TL,SH, AMPS             000013
00101   C                       (DEFAULT=1.5)                                         000013
00101   C               I3     -CELL SHORT CIRCUIT CURRENT AT TH,SL, AMPS             000013
00101   C                      (DEFAULT=.06)                                          000013
00101   C               I4     -CELL SHORT CIRCUIT CURRENT AT TH,SH, AMPS             000013
00101   C                       (DEFAULT=1.56)                                        000013
00101   C               V1     -CELL OPEN CIRCUIT VLOTAGE AT TL,SL, VOLTS             000013
00101   C                       (DEFAULT=.6)                                          000013
00101   C               RS     -CELL INTERNAL RESISTANCE, OHMS,(DEFAULT=.055)         000013
00101   C               A0     -MATERIAL CONSTANT(DEFAULT=1.54E33 FOR SILICON)        000013
00101   C               EG0    -BAND GAP AT 0K NORMALIZED BY BOLTZMANN'S              000013
00101   C                       CONSTANT(DEFAULT=1.4E4 FOR SILICON)                   000013
00101   C               IL1,DS,DT,DST                                                 000013
00101   C                      -COEFFICIENTS IN BIVARIATE EXPANSION FOR THE           000013
00101   C                       LIGHT CURRENT IL.  IF NOT PROVIDED, THEY WILL         000013
```

ORIGINAL PAGE IS OF POOR QUALITY

PV

BCS 40180-2 Rev.

BCS 40180-2 Rev.

```
00101      C                     BE COMPUTED FROM THE INPUTS I1,...,I4.                000013
00101      C                     THE UNITS FOR IL1,DS,DT,DST ARE RESPECTIVELY          000013
00101      C                     M2V-1,M2W-1,C-1,M2(WC)-1                              000013
00101      C             KD     -DEVICE CONSTANT. IF NOT PROVIDED, IT WILL BE          000013
00101      C                     COMPUTED FROM I1,V1                                   000013
00101      C             QBK    -DEVICE CONSTANT,K/V,(DEFAULT=ELECTRON CHARGE/         000013
00101      C                     BOLTZMANN'S CONSTANT=1.161E4)                         000013
00101      C             CF     -LENS RADIATION TRANSMITTANCE COEFFICIENT              000013
00101      C             RAP    -RATED POWER OF MAXIMUM POWER POINT TRACKER,KW         000013
00101      C                     (DEFAULT=LARGEST OPTIMAL POWER FOR THE RANGE          000013
00101      C                     OF TC AND ST)                                         000013
00101      C             CC     -CAPITAL COST/YEAR/UNIT CELL AREA, $/M2                000013
00101      C             CM     -MAINTENANCE COST/YEAR, $                              000013
00101      C                                                                           000013
00103            REAL I,NS,NP,I1,I2,I3,I4,IL1,KD,IL,IO,IM,IME                          000013
00104            DIMENSION EFF(1),EFF1(14),OP(1),OV(1)                                 000013
00105            COMMON /CIMPL/IMPL,ICNT,ITEST                                         000013
00106            COMMON /CTIME/TIME /CSIMUL/DUM(7),TMAX                                000013
00107            COMMON /COST/CCAP,CMA,COP                                             000013
00110            DATA EFF1/0.,.1,.2,.3,.4,.5,1.,.338,.44,.53,.61,.70,.75,.9/           000013
00112            IL(S,T)=IL1*S*(1.+DS*(S-SL)+DT*(T-TL)+DST*(S-SL)*(T-TL))              000013
00113            IO(T)=KD*AO*((T+273)**3)*EXP(-EGO/(T+273))                            000013
00114            FIL(I,T)=I-IO(T)*(1.-EXP(QBK*I*RS/(T+273)))                           000013
00115            IF(IMPL.GT.0)GO TO 100                                                000013
00117            SP=0.                                                                 000016
00120            TMAX1=TMAX*.99999                                                     000017
00121            TINC1 = DUM(7)*0.5                                                    000022
00121      C                                                                           000022
00121      C           INITIALIZATION                                                  000022
00121      C                                                                           000022
00122            IF(EFF(2).NE.1.99999)GO TO 11                                         000025
00124            EFF(2)=7                                                              000030
00125            DO 12 II=4,17                                                         000037
00130         12 EFF(II)=EFF1(II-3)                                                    000037
00132         11 CONTINUE                                                              000042
00133            OP(2)=10.                                                             000042
00134            OP(3)=10.                                                             000043
00135            OV(2)=10.                                                             000044
00136            OV(3)=10.                                                             000045
00137            IF(TL.EQ..99999)TL=28                                                 000046
00141            IF(TH.EQ..99999)TH=120                                                000053
00143            IF(TR.EQ..99999)TR=TH                                                 000063
00145            IF(SL.EQ..99999)SL=1000                                               000076
00147            IF(SH.EQ..99999)SH=25000                                              000106
00151            IF(SR.EQ..99999)SR=SH                                                 000113
00153            IF(RC.EQ..99999)RC=25                                                 000123
00155            IF(NS.EQ..99999)NS=300                                                000133
00157            IF(NP.EQ..99999)NP=500                                                000140
00161            IF(AA.EQ..99999)AA=1.5E-4*NS*NP                                       000145
00163            IF(I1.EQ..99999)I1=.06                                                000154
00165            IF(I2.EQ..99999)I2=1.5                                                000161
00167            IF(I3.EQ..99999)I3=.06                                                000166
00171            IF(I4.EQ..99999)I4=1.56                                               000173
00173            IF(V1.EQ..99999)V1=.6                                                 000200
00175            IF(RS.EQ..99999)RS=.055                                               000205
00177            IF(AO.EQ..99999)AO=1.54E33                                            000212
00201            IF(EGO.EQ..99999)EGO=1.4E4                                            000217
```

272K

PV

```
00203          IF(QBK.EQ..99999)QBK=1.161E4                                000224
00203    C                                                                 000224
00205          IF(KD.EQ..99999)KD=I1/(((TL+273)**3)*EXP(-EGO/             000231
00205         1 (TL+273))*(EXP(QBK*V1/(TL+273))-                          000231
00205         2 EXP(QBK*I1*RS/(TL+273))))/AD                              000231
00207          IF(IL1.EQ..99999)IL1=FIL(I1,TL)/SL                         000275
00211          IF(DS.EQ..99999)DS=(FIL(I2,TL)-IL1*SH)/(IL1*SH*(SH-SL))    000334
00213          IF(DT.EQ..99999)DT=(FIL(I3,TH)-IL1*SL)/(IL1*SL*(TH-TL))    000401
00215          IF(DST.EQ..99999)DST=(FIL(I4,TH)-IL1*SH-                   000440
00215         1 IL1*SH*DS*(SH-SL)-IL1*SH*DT*(TH-TL))/                     000440
00215         2 (IL1*SH*(SH-SL)*(TH-TL))                                  000440
00215    C                                                                 000440
00215    C      CALCULATE OPTIMAL POWER OP AND CELL VOLTAGE                000440
00215    C                                                                 000440
00215    C      IF TERMINAL VOLTAGE IS NOT INPUT                           000440
00215    C                                                                 000440
00217          IF(VT.NE..99999)GO TO 100                                  000510
00221          S=0.                                                       000513
00222          DO 33 J=1,10                                               000517
00225          J0=J+3                                                     000517
00226          OP(J0)=(J-1)*TR/9.                                         000526
00227       33 OV(J0)=OP(J0)                                              000536
00231          DO 3 K=1,10                                                000544
00234          T=0.                                                       000563
00235          DO 4 J=1,10                                                000567
00240          AIL=IL(S,T)                                                000575
00241          BIO=IO(T)                                                  000605
00242          VOC=(T+273)*ALOG(1.+AIL/BIO)/QBK                           000622
00243          VL=0.                                                      000637
00244          VH=VOC                                                     000640
00244    C                                                                 000640
00244    C              BINARY SEARCH FOR MAX POWER POINT                  000640
00244    C                                                                 000640
00245          DO 5 M=1,10                                                000644
00250          VM=(VL+VH)*.5                                              000644
00251          VME=VM+1.E-5                                               000650
00252          IM=AINR(AIL,BIO,QBK,VM,RS,T)                               000652
00253          PM=IM*VM                                                   000663
00254          IME=AINR(AIL,BIO,QBK,VME,RS,T)                             000665
00255          PME=IME*VME                                                000676
00256          PMP=PME-PM                                                 000700
00257          IF(PMP.GE.0.)VL=VM                                         000702
00261          IF(PMP.LT.0.)VH=VM                                         000707
00263          IF((VH-VL).LE.2.E-5)GO TO 6                                000714
00265        5 CONTINUE                                                   000724
00265    C                                                                 000724
00267        6 CONTINUE                                                   000724
00270          IKJ=13+K+J*10                                              000724
00271          OV(IKJ)=VM                                                 000727
00272          OP(IKJ)=PM/1000.                                           000733
00273          T=T+TR/9.                                                  000740
00274        4 CONTINUE                                                   000745
00276          IKJ0=13+K                                                  000745
00277          OP(IKJ0)=S                                                 000753
00300          OV(IKJ0)=S                                                 000755
00301          S=S+SR/9.                                                  000756
00302        3 CONTINUE                                                   000762
```

```
00302      C                                                                  000762
00304            IF(RAP.EQ..99999)RAP=OP(33)*NS*NP                            000762
00304      C     WRITE(6,101)(OP(IK),IK=24,123)                               000762
00304      C     WRITE(6,101)(OV(IK),IK=24,123)                               000762
00304      C 101 FORMAT(1H0,3HPV:,/,(5X,10E10.2))                             000762
00304      C                                                                  000762
00306        100 CONTINUE                                                     000772
00306      C                                                                  000772
00306      C          COMPUTE INSOLATION AT THE CELLS                         000772
00306      C                                                                  000772
00307            S=ST*RC*CF                                                   000772
00307      C                                                                  000772
00307      C          COMPUTE CELL VOLTAGE AND POWER                          000772
00310            IF(VT.NE..99999)GO TO 900                                    000775
00312            IF(IMPL.NE.2)GO TO 809                                       001000
00314            IF((S.GT.SR).OR.(TC.GT.TR))WRITE(6,808)                      001003
00317        808 FORMAT(1H0,62HPV: WARNING: INSOLATION OR TEMPERATURE AT CELL EXCEE   001026
00317           1D RANGE        )                                             001026
00320            IF((S.GT.SR).OR.(TC.GT.TR))ICNT=ICNT+1                       001026
00322        809 CONTINUE                                                     001050
00323            V=TBLU2(S,TC,OV(14),OV(4),OV(24),1,1,10,10,10,10)            001050
00324            P=TBLU2(S,TC,OP(14),OP(4),OP(24),1,1,10,10,10,10)            001073
00325            GO TO 901                                                    001117
00326        900 CONTINUE                                                     001121
00327            V=VT/NS                                                      001121
00330            AIL=IL(S,TC)                                                 001123
00331            BIO=IO(TC)                                                   001145
00332            I=AINR(AIL,BIO,QBK,V,RS,TC)                                  001165
00333            P=I*V/1000.                                                  001176
00334        901 CONTINUE                                                     001202
00334      C                                                                  001202
00334      C     COMPUTE ARRAY VOLTAGE AND POWER                              001202
00334      C                                                                  001202
00335            V=V*NS                                                       001202
00336            P=P*NS*NP                                                    001204
00337            I=0.                                                         001210
00340            IF(V.GT.0.)I=P*1000./V                                       001211
00342            EF1=1.                                                       001221
00343            EF2=1.                                                       001223
00344            IF(S.GT.0.)EF1=P*1000/(S*AA)                                 001224
00346            IF(VT.NE..99999)GO TO 904                                    001234
00350            PRAT=P/RAP                                                   001237
00351            NEF=EFF(2)                                                   001242
00352            EF2=TBLU1(PRAT,EFF(4),EFF(4+NEF),1,-NEF)                     001251
00353            P=P*EF2                                                      001270
00354        904 CONTINUE                                                     001273
00355            IF(IMPL.LE.1)RETURN                                          001273
00357            SP=SP+P*TINC1                                                001301
00360            IF(TIME.LT.THAX1)RETURN                                      001305
00362            CCAP=CCAP+CC*AA                                              001314
00363            CMA=CMA+CM                                                   001320
00364            RETURN                                                       001323
00365            END                                                          001714
```

PV

Revision Pages

Section 7.33A - SO

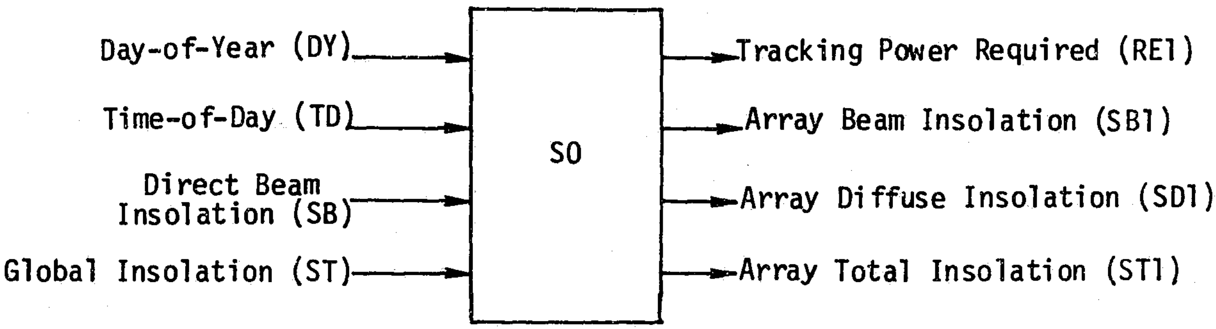
Delete pages 283 and 284 of the original document. Insert revision pages 283 - 283J and 284 between pages 282 and 285 of the original document.

PRECEDING PAGE BLANK NOT FILMED

```
00105          IF(FIN.LT.C6)GO TO 20                    000003
00107          IF(FIN.LT.0.)GO TO 30                    000007
00111          FO=C1*FIN                                000012
00112          GO TO 100                                000015
00112    C     POSITIVE SATURATION                      000015
00113     10   FO=C1*C3+C2*(FIN-C3)                     000017
00114          GO TO 100                                000025
00114    C     NEGATIVE SATURATION                      000025
00115     20   FO=C4*C6+C5*(FIN-C6)                     000027
00116          GO TO 100                                000035
00116    C     NEGATIVE UNSATURATED                     000035
00117     30   FO=C4*FIN                                000037
00120    100   RETURN                                   000042
00121          END                                      000075
```


SA

7.33A SOLAR ARRAY ORIENTATION



The Solar Orientation model computes flat plate collector insolation for five types of solar tracking:

- Tilted orientation, facing south
- Tracking about a horizontal EW axis
- Tracking about a horizontal NS axis
- Tilted, tracking about a vertical axis
- Two axis tracking

Array insolation is the sum of beam and diffuse components. The beam component is the product of normal incidence radiation and a geometry-dependent incidence factor. The diffuse component is approximated as the product of horizontal diffuse insolation times a geometry factor plus ground reflectance.

BASIC EQUATIONS

```
ST1 = SB1 + SD1 + SR1
    = SB*IF + SD*RD + ST*RR
 SD = ST - SB*SIN(EL)
 RD = .5*(1 + COS(TLT))
 RR = .5*PR*(1 - COS(TLT)),
```

where

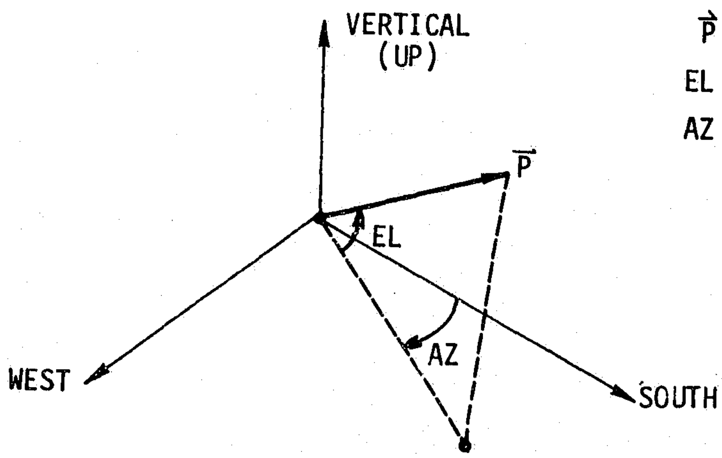
```
 IF = solar incidence factor (incidence angle cosine)
TLT = collector tilt angle from horizontal
 PR = ground reflectance
```

Inputs/Port	Description	Units
LA	Collector latitude*	Deg
DY	Day-of-the-year (1-365)	-
TD	Time-of-day (0-24)	hr
MØ	Tracking mode 1 = fixed orientation and tilt (default) 2 = horizontal EW axis tracking 3 = horizontal NS axis tracking 4 = tilted, vertical axis tracking 5 = two axis tracking	-
TL	Collector tilt (MØ = 1, 4 inputs)	Deg
SB	Direct normal beam insolation	w/m^2
ST	Global insolation on a horizontal surface	w/m^2
PR	Ground reflectance (default = 0.2)	-

*For TMY stations, see Table 7.7A of the Environmental Data Component ED.

Inputs/Port (cont'd)		Description	Units
AA		Collector array area	m^2
SBT		Insolation threshold for tracking (default = 100.)	w/m^2

Outputs/Port		Description	Units
SE		SIN (Solar Elevation Angle)*	-
SA		SIN (Solar Azimuth Angle)*	-
IF		COS (Solar Incidence Angle)	-
RE	1	Tracking power required	kw
SB	1	Collector beam insolation	w/m^2
SD	1	Collector diffuse insolation	w/m^2
SR	1	Collector reflected insolation	w/m^2
ST	1	Collector total insolation	w/m^2
TLT		Collector tilt angle	Deg



$\vec{P}$ = Solar Orientation Vector

EL = Solar Elevation Angle

AZ = Solar Azimuth Angle

* Figure 7.33A Solar Orientation Angles

CALCULATION SEQUENCE

$$RPD = \pi/180$$

If SB ≤ 0 and MØ > 1 return

1) Solar azimuth and elevation

$$W = 15*(12 - TD)*RPD$$

$$\delta = 23.45*SIN(2\pi*(284 + DY)/365)*RPD$$

$$LA' = LA*RPD$$

$$SE = SIN\ \delta *SIN\ LA' + COS\ \delta *COS\ W*COS\ LA'$$

$$CE = (1. - SE*SE)^{1/2}$$

$$TAN(AZ) = COS\ \delta *SIN\ W/(COS\ W*SIN\ LA'*COS\ \delta - SIN\ \delta *COS\ LA')$$

$$CA = 1/(1 + TAN^2(AZ))^{1/2}$$

$$SA = TAN(AZ)*CA$$

2) Horizontal diffuse insolation

$$SD = ST - SB*SE$$

3) Array geometry and tracking power

$$RE1 = 0$$

If MØ = 1 then

$$TLT' = TL*RPD$$

$$IF = SIN\ TLT'*CE*CA + COS\ TLT'*SE$$

If MØ = 2 then

$$IF = \sqrt{1. - (CE*SA)^2}$$

$$TLT' = MIN(COS^{-1}(SE/IF),\ \pi/2)$$

$$RE1 = 3.75\ E\text{-}4*AA \qquad \text{if } SB > SBT$$

CALCULATIONS (contd)

If MØ = 3 then

$$IF = \sqrt{1. - (CE*CA)^2}$$

$$TLT' = MIN(COS^{-1}(SE/IF),\ \pi/2)$$

$$RE1 = 3.75\ E\text{-}4*AA \qquad \text{if } SB > SBT$$

If MØ = 4 then

$$TLT' = TL*RPD$$

$$IF = SIN\ TLT'*CE + COS\ TLT'*SE$$

$$RE1 = 3.75\ E\text{-}4*AA \qquad \text{if } SB > SBT$$

If MØ = 5 then

$$IF = 1$$

$$TLT' = MIN(COS^{-1}(SE),\ \pi/2)$$

$$RE1 = 5.E\text{-}4*AA \qquad \text{if } SB > SBT$$

4) Insolation components

$$SB1 = SB*IF$$

$$SD1 = SD*.5*(1 + COS(TLT'))$$

$$SR1 = ST*.5*PR*(1 - COS(TLT'))$$

$$ST1 = SB1 + SD1 + SR1$$

5) Tilt

$$TLT = TLT'/RPD$$

REFERENCES FOR SO

1. J. K. Linn, "Photovoltaic System Analysis Program-SOLCEL," Sandia Laboratories Report SAND77-1268, August 1977.

2. B. Y. Liu and R. C. Jordan, "The Interrelationship and Characteristic Distribution of Direct, Diffuse and Total Solar Radiation," Solar Energy, Vol. IV, July 1960, pp. 1-19.

3. J. A. Duffie and W. A. Beckman, Solar Thermal Processes (Chapter 2), Wiley, 1974.

```
SUBROUTINE SO          ENTRY POINT 000465

STORAGE USED: CODE(1) 000605; DATA(0) 000060; BLANK COMMON(2) 000000

 COMMON BLOCKS:

 0003   CIMPL  000001

EXTERNAL REFERENCES (BLOCK, NAME)

 0004   SIN
 0005   COS
 0006   SQRT
 0007   NERR2$
 0010   ACOS
 0011   NERR3$

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

 0001     000023 100L      0001     000052 109L      0001     000175 200L      0001     000224 301L      0001     000244 302L
 0001     000306 303L      0001     000350 304L      0001     000376 305L      0001     000425 309L      0000 R 000002 ADEL
 0000 R 000020 BIF       0000 R 000013 CA        0000 R 000006 CADEL     0000 R 000011 CE        0000 R 000004 CLAP
 0000 R 000010 COSW      0000 R 000012 F         0000 I 000016 IMO       0003 I 000000 IMPL      0000   000043 INJP$
 0000 R 000003 PLA       0000 R 000000 RPD       0000 R 000005 SADEL     0000 R 000015 SD        0000 R 000021 SE1
 0000 R 000007 SINPLA    0000 R 000014 TAZ       0000 R 000017 TLTP      0000 R 000001 W


00100     CSO                                                                                000000
00101           SUBROUTINE SO(SE,SA,IF,RE1,SB1,SD1,SR1,ST1,TLT,                               000000
00101          1  LA,DY,TD,MO,TL,SB,ST,PR,AA,SBT)                                            000000
00101     C                                                                                  000000
00101     C        PURPOSE   THIS COMPONENT COMPUTES FLAT PLATE COLLECTOR                        000000
00101     C                  INSOLATION FOR FIVE MODES OF SOLAR TRACKING&                        000000
00101     C                     TILTED ORIENTATION, FACING SOUTH                                 000000
00101     C                     TRACKING ABOUT A HORIZONTAL EW AXIS                              000000
00101     C                     TRACKING ABOUT A HORIZONTAL NS AXIS                              000000
00101     C                     TILTED, TRACKING ABOUT THE VERTICAL AXIS                         000000
00101     C                     TWO AXIS TRACKING                                                000000
00101     C                                                                                  000000
00101     C        WRITTEN BY Y.K.CHAN, 11-6-78, VERSION 1                                      000000
00101     C                                                                                  000000
00101     C        METHOD   ARRAY INSOLATION IS SUM OF BEAM AND DIFFUSE                          000000
00101     C                 COMPONENTS. THE BEAM COMPONENT IS THE PRODUCT OF                     000000
00101     C                 NORMAL INCIDENCE INSOLATION AND A GEOMETRY DEPENDENT                 000000
00101     C                 INCIDENCE FACTOR.  THE  DIFFUSE COMPONENT IS                         000000
00101     C                 APPROXIMATED AS THE PRODUCT OF HORIZONTAL DIFFUSE                    000000
00101     C                 INSOLATION TIMES A GEOMETRY FACTOR PLUS GROUND REFLECTANCE.          000000
00101     C                                                                                  000000
```

SO

BCS 40180-2 Rev.

```
00101      C         CALLING SEQUENCE                                                    000000
00101      C           OUTPUTS                                                           000000
00101      C             SE    -SINE OF SOLAR ELEVATION ANGLE                            000000
00101      C             SA    -SINE OF SOLAR AZIMUTH ANGLE                              000000
00101      C             IF    -COSINE OF SOLAR INCIDENCE ANGLE                          000000
00101      C             RE1   -TRACKING POWER REQUIRED,KW                               000000
00101      C             SB1   -COLLECTOR BEAM INSOLATION,W/M2                           000000
00101      C             SD1   -COLLECTOR DIFFUSE INSOLATION,W/M2                        000000
00101      C             SR1   -COLLECTOR REFLECTED INSOLATION,W/M2                      000000
00101      C             ST1   -COLLECTOR TOTAL INSOLATION,W/M2                          000000
00101      C             TLT   -COLLECTOR TILT ANGLE,DEGREES                             000000
00101      C           INPUTS                                                            000000
00101      C             LA    -COLLECTOR LATITUDE,DEGREES                               000000
00101      C             DY    -DAY OF YEAR(1-365)                                       000000
00101      C             TD    -TIME OF DAY(0-24),HOUR                                   000000
00101      C             MO    -TRACKING MODE                                            000000
00101      C                        1=FIXED ORIENTATION AND TILT (DEFAULT)               000000
00101      C                        2=HORIZONTAL EW AXIS TRACKING                        000000
00101      C                        3=HORIZONTAL NS AXIS TRACKING                        000000
00101      C                        4=TILTED,VERTICAL AXIS TRACKING                      000000
00101      C                        5=TWO AXIS TRACKIGN                                  000000
00101      C             TL    -COLLECTOR TILT (MO=1,4  INPUTS),DEGREES                  000000
00101      C             SB    -DIRECT NORMAL BEAM INSOLATION,W/M2                       000000
00101      C             ST    -GLOBAL INSOLATION ON A HORIZONTAL SURFACE,W/M2           000000
00101      C             PR    -GROUND REFLECTANCE (DEFAULT=0.2)                         000000
00101      C             AA    -COLLECTOR ARRAY AREA,M2                                  000000
00101      C             SBT   -INSOLATION THRESHOLD FOR TRACKING,W/M2                   000000
00101      C                    (DEFAULT=100)                                            000000
00101      C                                                                             000000
00103            COMMON /CIMPL/IMPL                                                      000000
00104            REAL IF,LA,MO                                                           000000
00105            IF(IMPL.NE.0)GO TO 100                                                  000000
00107            IF(MO.EQ..99999)MO=1.                                                   000001
00111            IF(PR.EQ..99999)PR=.2                                                   000006
00113            IF(SBT.EQ..99999)SBT=100                                                000013
00115            RPD=3.1415926/180.                                                      000023
00116        100 CONTINUE                                                                000023
00117            IF((SB.GT.0.).OR.(MO.LT.2.))GO TO 109                                   000023
00121            SA=0.                                                                   000037
00122            IF=0.                                                                   000040
00123            RE1=0.                                                                  000041
00124            SB1=0.                                                                  000042
00125            SD1=0.                                                                  000043
00126            SR1=0.                                                                  000044
00127            ST1=0.                                                                  000045
00130            RETURN                                                                  000046
00131        109 CONTINUE                                                                000052
00132            RE1=0.                                                                  000052
00132      C                                                                             000052
00132      C       SOLAR AZIMUTH AND ELEVATION                                           000052
00132      C                                                                             000052
00133            W=15.*(12.-TD)*RPD                                                      000052
00134            ADEL=23.45*SIN(.0172142*(284+DY))*RPD                                   000057
00135            PLA=LA*RPD                                                              000071
00136            CLAP=COS(PLA)                                                           000074
00137            SADEL=SIN(ADEL)                                                         000100
00140            CADEL=COS(ADEL)                                                         000104
```

283H

SO

283I

```
00141          SINPLA=SIN(PLA)                                  000110
00142          COSW=COS(W)                                      000114
00143          SE=SADEL*SINPLA+CADEL*COSW*CLAP                  000120
00144          CE=SQRT(1.-SE*SE)                                000127
00145          F=CADEL*COSW*SINPLA-SADEL*CLAP                   000136
00146          CA=0.                                            000144
00147          SA=1.                                            000145
00150          IF(ABS(F).LE.1.E-5)GO TO 200                     000147
00152          TAZ=CADEL*SIN(W)/F                               000153
00153          CA=1./SQRT(1.+TAZ*TAZ)                           000161
00154          SA=TAZ*CA                                        000172
00155      200 CONTINUE                                         000175
00155    C                                                      000175
00155    C       HORIZONTAL DIFFUSE INSOLATION                  000175
00155    C                                                      000175
00156          SD=ST-SB*SE                                      000175
00156    C                                                      000175
00156    C       ARRAY GEOMETRY AND TRACKING POWER              000175
00156    C                                                      000175
00157          IMO=MO+.1                                        000200
00160          GO TO(301,302,303,304,305),IMO                   000211
00161      301 TLTP=TL*RPD                                      000224
00162          IF=SIN(TLTP)*CE*CA+COS(TLTP)*SE                  000226
00163          GO TO 309                                        000242
00164      302 IF=SQRT(1.-CE*CE*SA*SA)                          000244
00165          BIF=AMIN1(1.,SE/IF)                              000255
00166          TLTP=1.5708                                      000264
00167          IF(BIF.GT.0.)TLTP=ACOS(BIF)                      000266
00171          IF(SE.GT.SBT)RE1=3.75E-4*AA                      000275
00173          GO TO 309                                        000304
00174      303 IF=SQRT(1.-CE*CE*CA*CA)                          000306
00175          BIF=AMIN1(1.,SE/IF)                              000317
00176          TLTP=1.5708                                      000326
00177          IF(BIF.GT.0.)TLTP=ACOS(BIF)                      000330
00201          IF(SB.GT.SBT)RE1=3.75E-4*AA                      000337
00203          GO TO 309                                        000346
00204      304 TLTP=TL*RPD                                      000350
00205          IF=SIN(TLTP)*CE+COS(TLTP)*SE                     000352
00206          IF(SB.GT.SBT)RE1=3.75E-4*AA                      000365
00210          GO TO 309                                        000374
00211      305 IF=1.                                            000376
00212          SE1=AMIN1(SE,1.)                                 000377
00213          TLTP=1.5708                                      000404
00214          IF(SE1.GT.0.)TLTP=ACOS(SE1)                      000406
00216          IF(SB.GT.SBT)RE1=5.E-4*AA                        000415
00216    C                                                      000415
00220      309 CONTINUE                                         000415
00220    C                                                      000425
00220    C             INSOLATION COMPONENTS                    000425
00220    C                                                      000425
00221          SB1=SB*IF                                        000425
00222          SD1=SD*.5*(1.+COS(TLTP))                         000427
00223          SR1=ST*.5*PR*(1.-COS(TLTP))                      000440
00224          ST1=SB1+SD1+SR1                                  000447
00224    C                                                      000447
00224    C               TILT                                   000447
00224    C                                                      000447
```

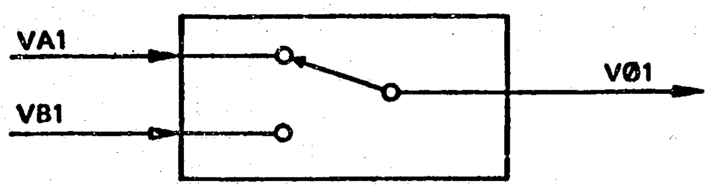
BCS 40180-2 Rev.

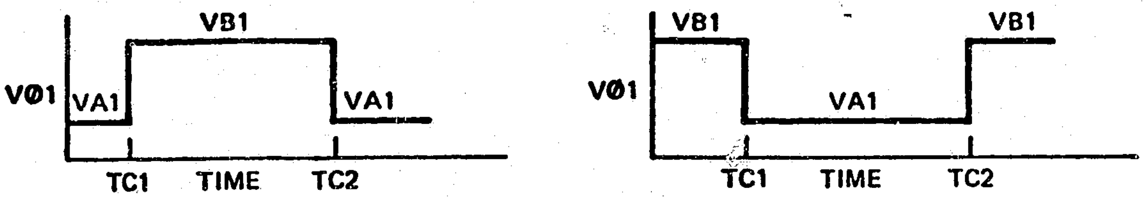
SO

```
00225    136*        TLT=TLTP/RPD                                    000452
00226    137*        RETURN                                          000455
00227    138*        END                                             000604


@FIN
```


S0

7.34 SINGLE POLE SWITCH





SW1=1 SW1=0

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	
VØ1	Switch output

Calculation Sequence

If SW1 = 0 then

$$VØ1 = \left(\begin{array}{ll} VA1 & TC1 < TIME < TC2 \\ VB1 & \text{otherwise} \end{array} \right)$$

If SW1 = 1 then

$$VØ1 = \left(\begin{array}{ll} VB1 & TC1 < TIME < TC2 \\ VA1 & \text{otherwise} \end{array} \right)$$

Revision Pages

Sections 9.0 - 9.3 and Appendix

Insert revision pages 411 - 450 following page 410 of the original document.

PRECEDING PAGE BLANK NOT FILMED

9.0 SOLAR PHOTOVOLTAIC EXAMPLES

The solar photovoltaic component models added to the SIMWEST library are briefly described and test case results illustrating their use are summarized in this section.

Table 9.0-1 summarizes the characteristics of the solar-photovoltaic components. The environmental data component is designed to read Typical Meteorological Year (TMY) data tapes containing hourly insolation and weather data at 26 U.S. locations. This component can also be used to read other hourly data tapes such as the SOLMET tapes by inputing a user specified format to the model generation program. The solar orientation or tracking component computes the sum of direct beam and global insolation on a flat plate array for fixed orientation and four different beam tracking options. The flat plate and focusing lens collector components provide detailed thermal analyses for determining average solar cell temperature. The collector models, and that of the solar array are based on similar models developed at Sandia Laboratories for the SOLCEL program. (Reference [4]). The array component model is a simplified model based on scaling the characteristics of a single solar cell. Array voltage can either be user specified or determined by a maximum power tracker. It should be observed that the above components are coded in SI (metric) units, whereas most of the SIMWEST components are coded in English units. This is generally not a problem since there are at most only a few interconnection variables between the solar-photovoltaic generation components and other SIMWEST components, and these are easily converted using arithmetic components.

The TMY data tapes are currently the best environmental data sources available for simulating typical yearly solar energy system performance. These tapes were extracted from SOLMET data tapes containing rehabilitated hourly solar and meteorological observation data over a period of many years at each observation site. Each Typical Meteorological Year was created by statistical selection of a typical meteorological month for each calendar month in the long term data base and catenating the 12 months to form a TMY. All of the TMY data files are available for use by a SIMWEST user. He thus has access to a high quality environmental data base for solar energy simulations and system analyses.

TABLE 9.0-1 SOLAR-PHOTOVOLTAIC COMPONENTS

COMPONENT	SYMBOL	PURPOSE
• ENVIRONMENTAL DATA (TAPE)	ED	READ DOE SOLAR INSOLATION AND WEATHER DATA TYPICAL METEOROLOGICAL YEAR TAPE
• SOLAR ORIENTATION (TRACKING)	SO	SOLAR INSOLATION ON TILTED FLAT PLATE ARRAY (FIVE OPTIONS)
• FLAT PLATE COLLECTOR	FP	FLAT PLATE THERMAL MODEL WITH FLUID AND PASSIVE COOLING OPTIONS
• FOCUSING LENS COLLECTOR	FO	FRESNEL LENS THERMAL MODEL WITH FLUID AND PASSIVE COOLING OPTIONS
• PHOTOVOLTAIC ARRAY	PV	CONVERTS SOLAR INSOLATION TO D.C. ELECTRICAL POWER. MAXIMUM POWER TRACKER OR USER SPECIFIED VOLTAGE

9.1 PHOTOVOLTAIC MODEL TEST CASE

The input data for the photovoltaic model test case is shown in Figure 9.1-1. The purpose of this model is to obtain characteristic current voltage curves for the default solar array parameters. Fortran statements are used in the model generation data to let the terminal voltage range between 0 and 204 volts for solar insolation values of 5, 20, and 50 suns (1 sun = 1000 w/m^2). Cell temperature is specified at 25^oC for the first simulation and 55^oC for the second. Figure 9.1-2 shows the current voltage curves and Figure 9.1-3 shows power voltage cross plots at the lower cell temperature and for the three solar insolation levels. These curves verify the physical characteristics of the solar cell model. It may be noted in these figures that current and output power become negative when the specified voltage exceeds the array open circuit voltage. Individual cell characteristics may be obtained by dividing voltage by 300 (default number of cells in series) and by dividing current by 500 (default number of cells in parallel).

9.2 FLAT PLATE COLLECTOR MODEL

The input data for the flat plate model test case is shown in Figure 9.2-1. The purpose of this model is to illustrate water and wind cooling of the collector and to test the tracking options of the orientation component **SO**. There are six 1-1/2 day simulation runs. The first run uses water cooling (CMOFP=2), a single glass cover over the front plate and insolation on the back. The second run uses passive cooling (CMOFP=0), no plate insolation and fins on the back to cool the collector. In the first two runs, the collector is tilted and has a fixed, southward facing orientation (MO SO=1). The last four runs are similar to run 2 except different tracking options are utilized.

```
MODEL DESCRIPTION   PHOTO-VOLTAIC CURRENT VOLTAGE CURVES
LOCATION=11     TI
FORTRAN STATEMENTS
        ST PV=5000
        IF(DY TI,GT,1,5)ST PV=20000
        IF(DY TI,GT,2,5)ST PV=50000
        VT PV=8,5*TD TI
LOCATION=53       PV
END OF MODEL
PRINT
```

a) Model Generation Input Data

```
PARAMETER VALUES
CYCLES=0,TO TI=0
DLINES=50
TC PV=25
RC PV=1
PRINTER PLOTS,DISPLAY1
V  PV,VS,TIME
I  PV,VS,V  PV
P  PV,VS,V  PV
P  PV,VS,TIME
TINC=,5,TMAX=72,PRATE=24,PRINT CONTROL=3,INT MODE=3,OUTRATE=1
TITLE=PHOTO-VOLTAIC CELL CURRENT VOLTAGE CURVES
SIMULATE
PARAMETER VALUES
TC PV=55
SIMULATE
```

b) Simulation Program Input Data

Figure 9.1-1 PV Test Case Input Data

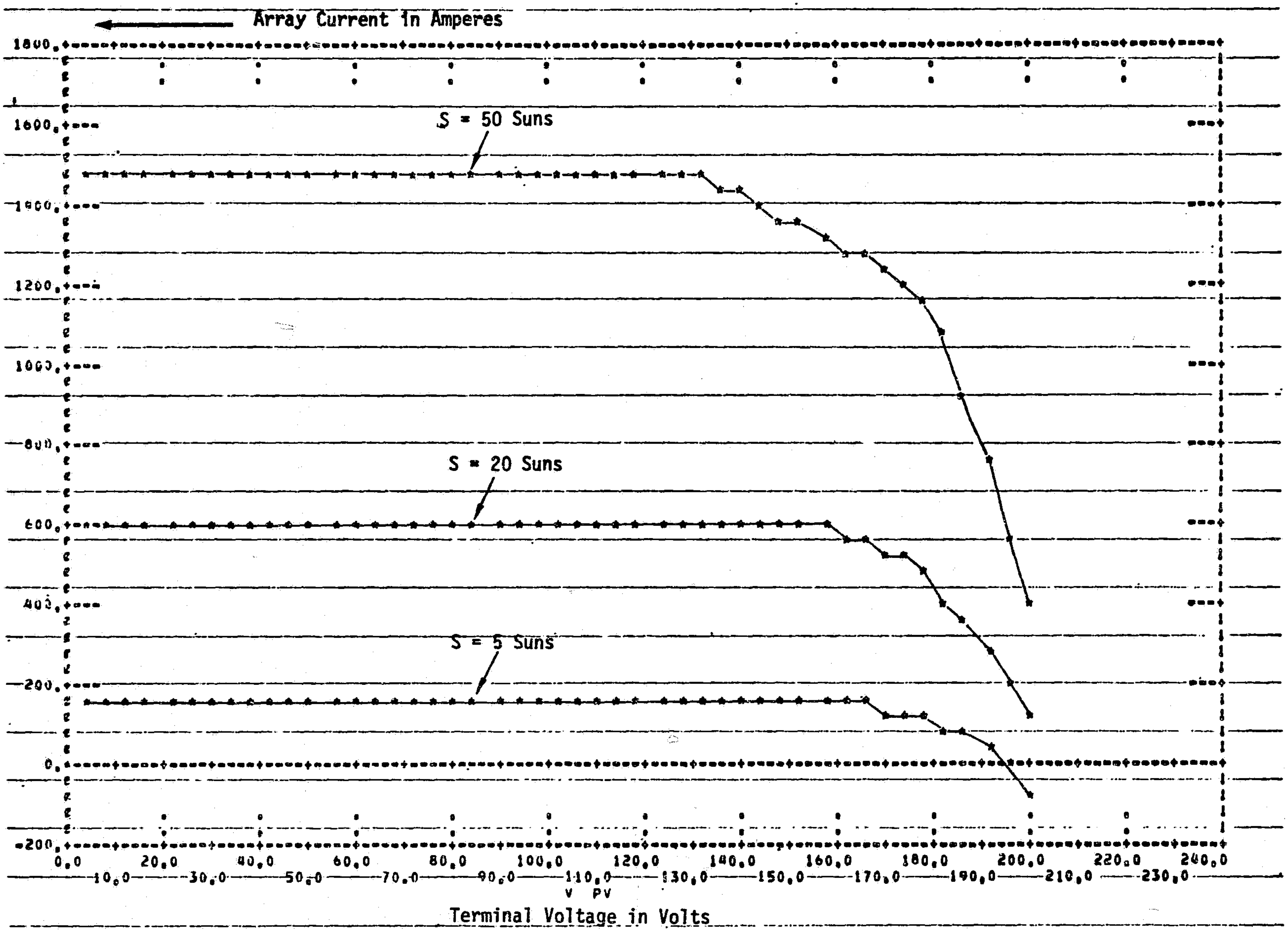


Figure 9.1-2 Solar Array Characteristic Current - Voltage Curves

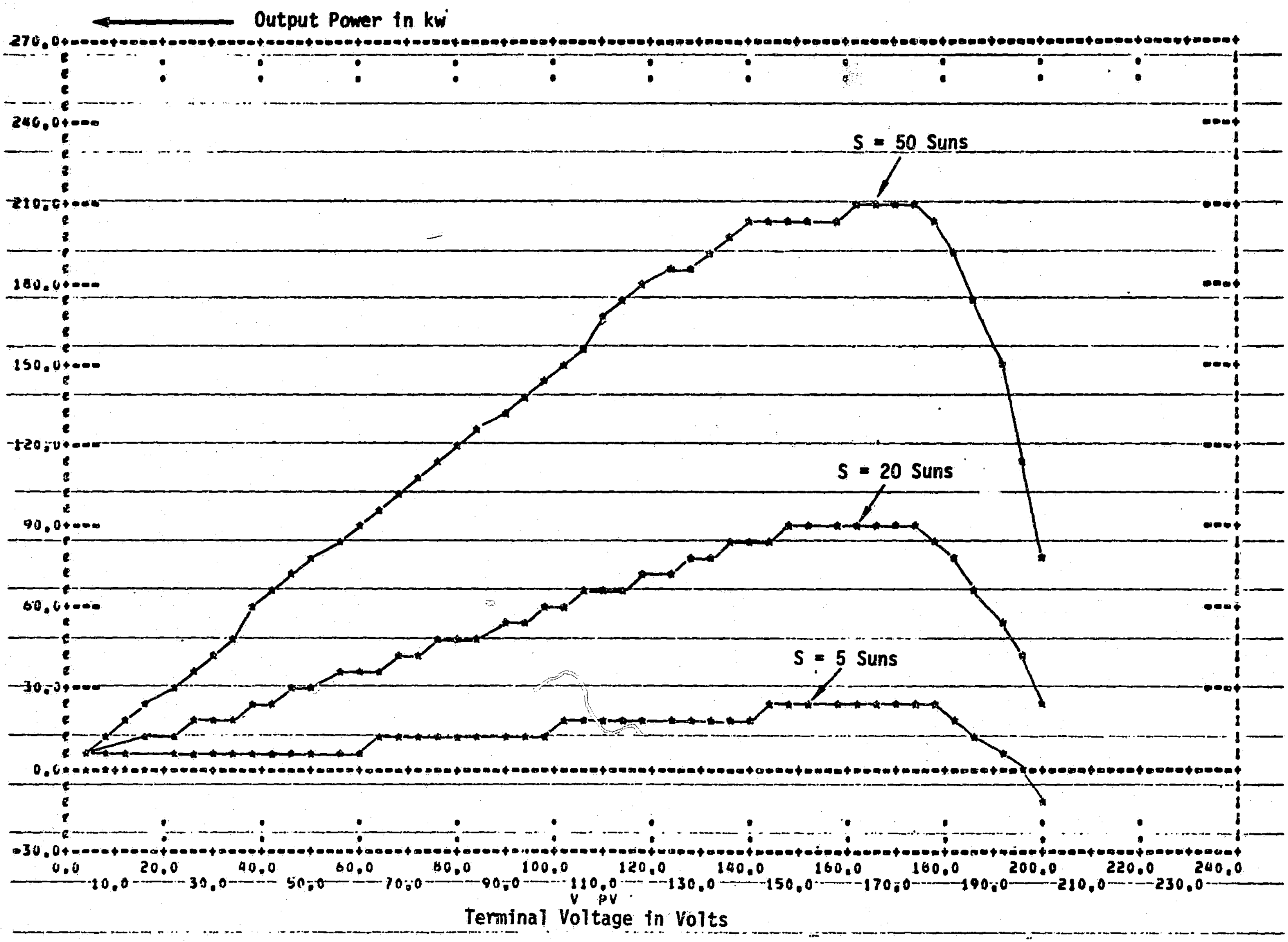


Figure 9.1-3 Solar Array Output Power Versus Voltage

```
MODEL DESCRIPTION          FLAT PLATE TEST CASE
LOCATION=11       TI
LOCATION=35       ED       INPUTS=TI
LOCATION=53       SO       INPUTS=TI,ED(X1=SB,X2=ST)
LOCATION=57       FP       INPUTS=SO,ED(X4=WD,X3=TA)
END OF MODEL
PRINT
```

a) Model Generation Input Data

```
PARAMETER VALUES
CYCLES=2,01,TO TI=36,TFIFP=10,TFOFP=30,MFMFP=,02,CMOFP=2,NG FP=1,
DLINES=50
HI FP=,01
CW FP=1,CL FP=2,NT FP=10,CC FP=1000,CM FP=10,CPOFP=,01,LA SO=29,733,
TL SO=29,733,AA SO=2
PRINTER PLOTS, DISPLAY1
TLTSO,VS,TIME
TC FP,VS,TIME
X2 ED,VS,TIME
P1 FP,VS,TIME
TINC=,5,TMAX=36,PRATE=6,PRINT CONTROL=3,INT MODE=3,OUTRATE=1
TITLE=FLAT PLATE  COLLECTOR TEST CASE
SIMULATE
PARAMETER VALUES
CMOFP=0,HI FP=1,E9,FIRFP=4
SIMULATE
PARAMETER VALUES
MO SO=2
SIMULATE
PARAMETER VALUES
MO SO=3
SIMULATE
PARAMETER VALUES
MO SO=4
SIMULATE
PARAMETER VALUES
MO SO=5
SIMULATE
```

b) Simulation Program Input Data

Figure 9.2-1 Flat Plate Collector Model Input Data

The model schematic produced by the model generation program is shown in Figure 9.2-2. The component **TI** is used to furnish time of day and day of year information to **SO** and to the TMY read component **ED**. **ED** supplies direct beam and global insolation to **SO**, and ambient temperature and wind speed to the collector component **FP**. Based on collector orientation, **SO** supplies solar insolation incident to the array, collector tilt angle, and tracking power to **FP**.

Typical results of the flat plate model runs are shown in Figures 9.2-3 through 9.2-5. Figure 9.2-3 shows the global horizontal insolation obtained from **ED** during the 36 hour simulation period. The data was for mid-winter and the daily peak levels are thus low to moderate. The array tilt angle daily pattern for horizontal E-W axis tracking is shown in Figure 9.2-4. At noon the array is oriented normal to the sun's incident rays and thus maximizes the insolation gathered during the mid-day peak. The tilt angle approaches 90^{o} as the sun approaches the horizon, and remains fixed at 90^{o} overnight. Comparison of the solar insolation peaks with the various tracking options showed that horizontal E-W axis tracking gave the best results of the single axis tracking systems, and was only slightly inferior to two-axis beam tracking. Solar cell temperature for this case is shown in Figure 9.2-5. The cell temperature is within a few degrees of ambient most of the day and rises in mid-day proportional to the solar insolation received. The results with water cooling are quite similar.

9.3 FRESNEL LENS COLLECTOR MODEL AND INCREMENTAL COSTS

The input data for the Fresnel Lens test case is shown in Figure 9.3-1. The purpose of this model is to illustrate a Fresnel Lens collector model with thermal fluid loops for collector cooling and for solar heating. Three week-long simulations are used to demonstrate incremental cost calculations for subsystem economic design. A variable speed pump is assumed for the collector fluid loop with the flow rate adjusted so that the outlet temperature is $5^{o}C$ greater than the inlet. The collector consists of a rectangular grid of 120 Fresnel lenses each of which focuses solar radiation on a 5 x 5 array of solar cells. Excess thermal energy is conducted to a heat sink surface and then dissipated by natural convection, radiation

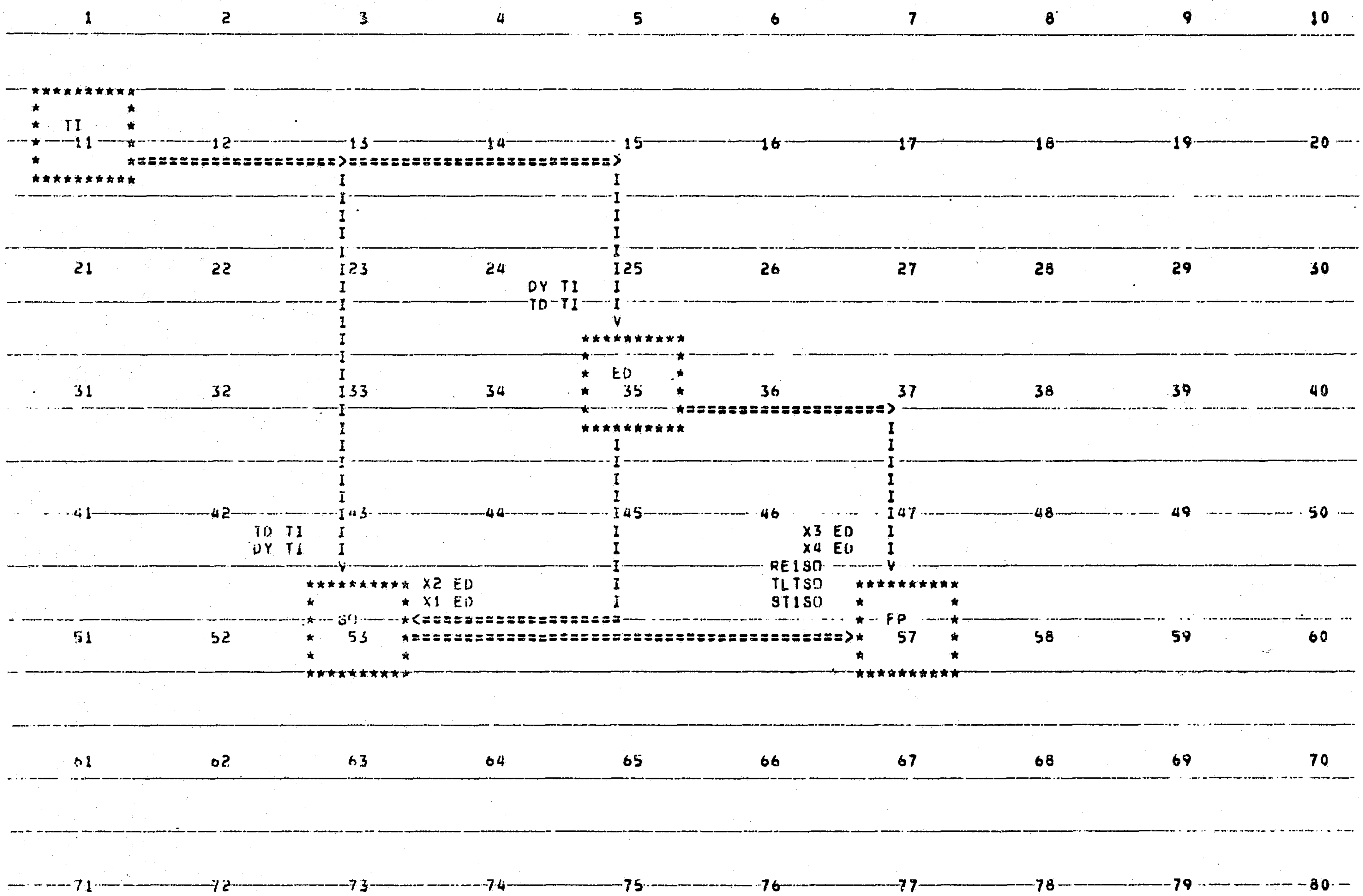


Figure 9.2-2 Flat Plate Model Schematic

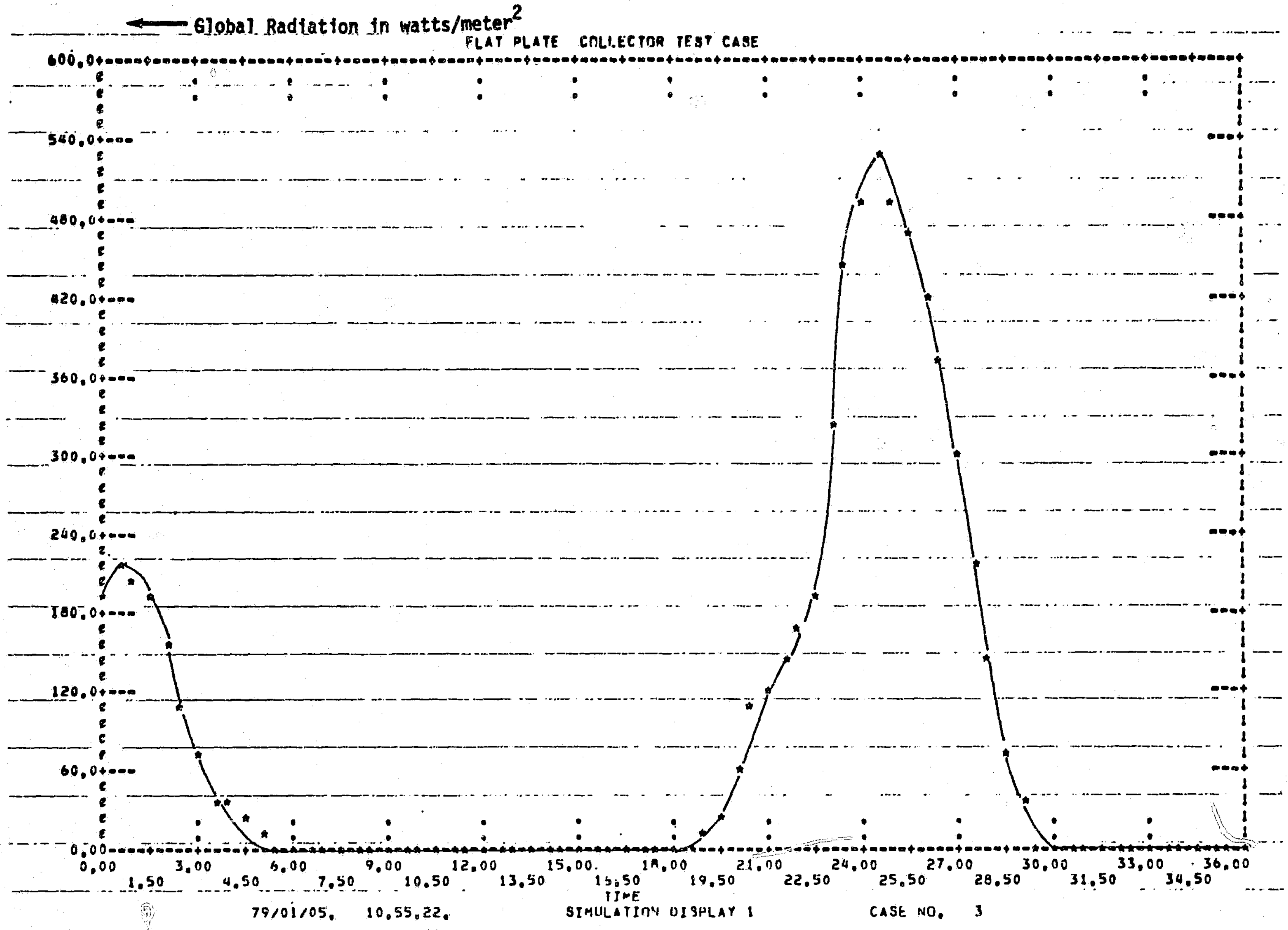


Figure 9.2-3 Global Horizontal Radiation Versus Time

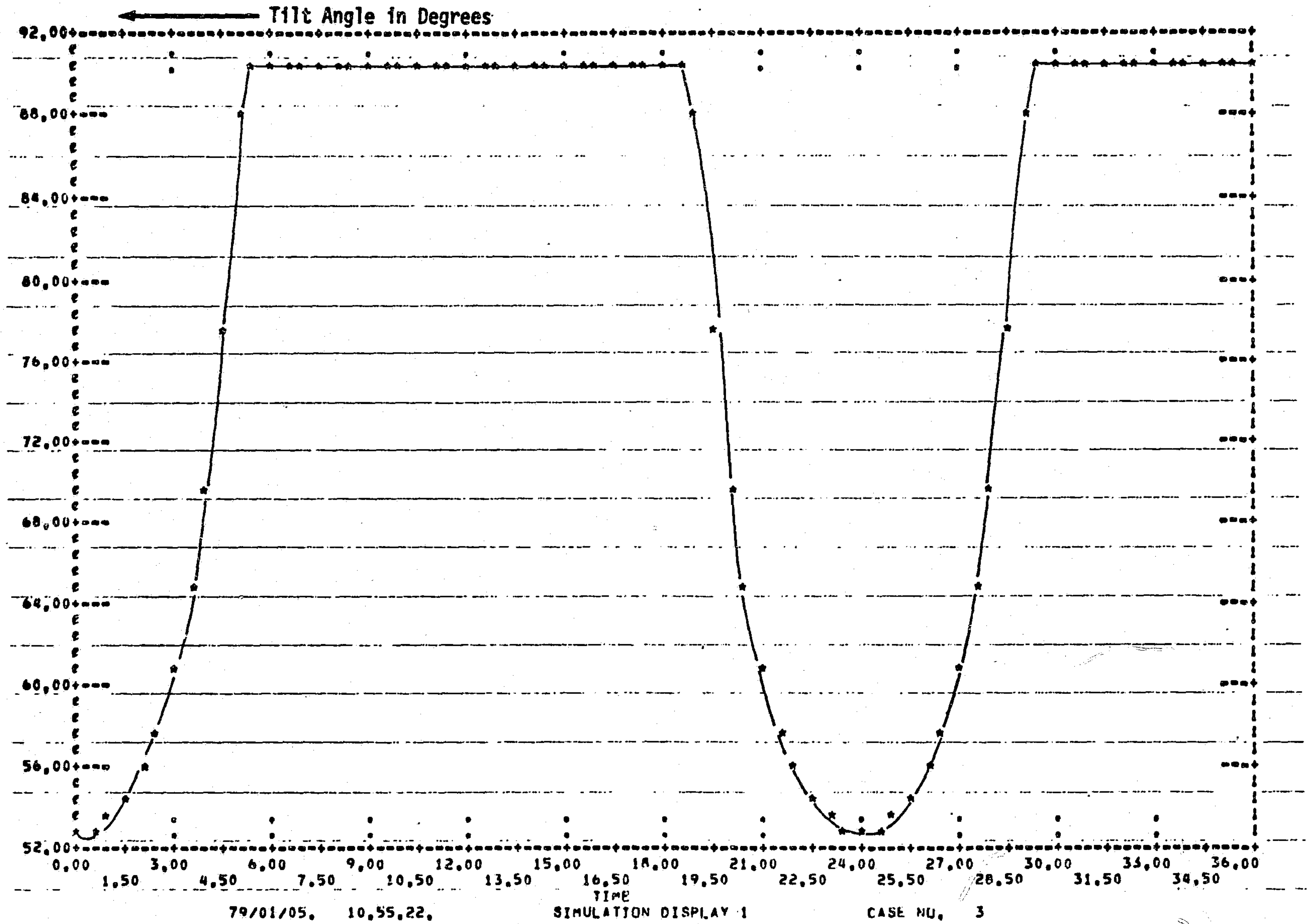


Figure 9.2-4 Tilt Angle Versus Time for Horizontal E-W Axis Tracking

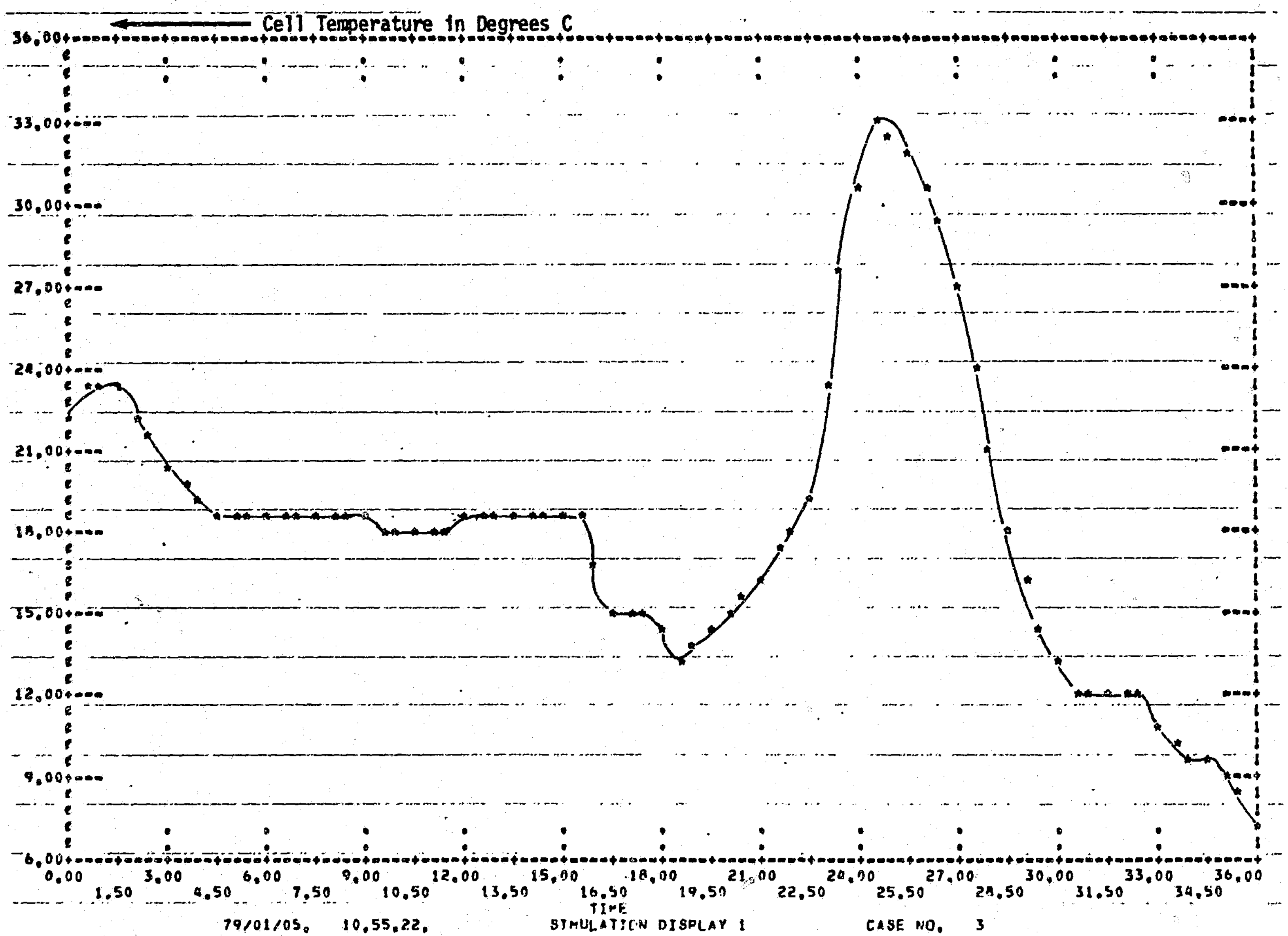


Figure 9.2-5 Solar Cell Temperature Versus Time

```
MODEL DESCRIPTION     FRESNEL LENS COLLECTOR WITH THERMAL STORAGE AND LOAD
LOCATION=11     TI
LOCATION=71     ED      INPUTS=TI
LOCATION=45     MA      INPUTS=TS(T=FIN)
FORTRAN STATEMENTS
      TFOFO = FO MA+5.
LOCATION=33     FO      INPUTS=ED(X1=ST,X3=TA,X4=WD),MA(FO=TFI)
LOCATION=73     PV      INPUTS=ED(X1=ST),FO
LOCATION=47     TS      INPUTS=FO(P,1=P),TL
LOCATION=27     TL      INPUTS=TI,ED(X3=TA)
LOCATION=77     LO      INPUTS=PV(P=P,1,P=LO,1)
LOCATION=79     CM
END OF MODEL
PRINT
```

a) Model Generation Input Data

```
TITLE=FRESNEL LENS COLLECTOR  (INCREMENTAL COST COMPUTATION)
PARAMETER VALUES
CYCLES=4.01,TO TI=0,CMOFO=2,CW FO=3.75,CL FO=3.9, DLINES=50
NL FO=120,NT FO=24,MFMFO=0.5,CC FO=6.,CM FO=50,HI FO=.01,RC FO=.06
TS TS=5,DH TS=.00879,PD TS=12,LE TS=30,NU TS=.01,NC TL=0.2
C1 MA=.55556,C2 MA=-17.7778, COPFO=0.5
CC PV=100,CM PV=50,LE TS=30,CR CM=15,LE CM=20
AA PV=0.6,NS PV=600,NP PV=5,RAPPV=1.3
VE LO=.05,VE TL=.05
TABLE,HT TS=4
.00879,.025491,.047371,.064072
90,147,147,204
TABLE,TLOTL=4
-10,0,10,25
4,2,1.5,1
TABLE,TWTTL=4
0,6,18,24
.4,1,1,.4
PRINTER PLOTS,DISPLAY1
RE TL,VS,TIME
E  TS,VS,TIME
P1 FO,VS,TIME
FMDFO,VS,TIME
DISPLAY2
TC FO,VS,TIME
P  PV,VS,TIME
FO MA,VS,TIME
INITIAL CONDITIONS=E  TS=80
TINC=.5,TMAX=168,PRATE=12,PRINT CONTROL=3,INT MODE=3,OUTRATE=1
SIMULATE
PARAMETER VALUES, TS TS=5.5
SIMULATE
PARAMETER VALUES
TS TS=5.,NL FO=126,CW FO=3.94,AA PV=0.63,NS PV=630
SIMULATE
```

b) Simulation Program Input Data

Figure 9.3-1 Fresnel Lens Model Input Data and heat exchange to the coolant fluid. The collector parameters are chosen for a lens concentration ratio of 25 and series connection of the output from each array. At maximum output the array collects about 10kw of solar radiation and produces about 1.7kw of electrical power. The user should be especially careful in specifying the input parameters to the collector and array components **FO** and **PV**, since inadvertant parameter errors can lead to physically inconsistent configurations, e.g., collector area smaller than the total lens area.

The model schematic produced by the model generation program is shown in Figure 9.3-2. The collector thermal loop is formed by the connections between the collector **FO** the thermal storage **TS** and the multiply and add component MA. The MA component is used to convert the thermal storage outlet temperature from degrees fahrenheit to degrees centigrade. The output temperature from MA is supplied as the inlet temperature to **FO**. The total thermal power gathered by the coolant fluid is computed in **FO** and supplied to **TS**. Similarly, the thermal load fluid loop is represented by a power request from the load component **TL** to **TS** and by thermal power delivered from **TS** to **TL**. The electrical output of the array is computed by **PV** and supplied to a load component **LO** which monitors the electrical energy collected.

Results of the first week simulation run are summarized in Figures 9.3-3 through 9.3-6. The weather was fairly constant during this run and solar insolation was fairly strong all week. Figure 9.3-3 shows that with water cooling cell temperature was held to less than 70^{o}C at peak insolation. In fact, about 60% of the solar energy incident on the array is exchanged to the coolant fluid during peak insolation. The electrical output of the array is shown in Figure 9.3-4. The fluid flow rate of the pump and thermal energy collected exhibit very similar daily patterns. The thermal load for this week is shown in Figure 9.3-5. This load is dependent on both time of day and ambient temperature which yields the complex load pattern shown. Figure 9.3-6 shows the temperature of the thermal storage vessel resulting from the collector and load thermal loops. The daily

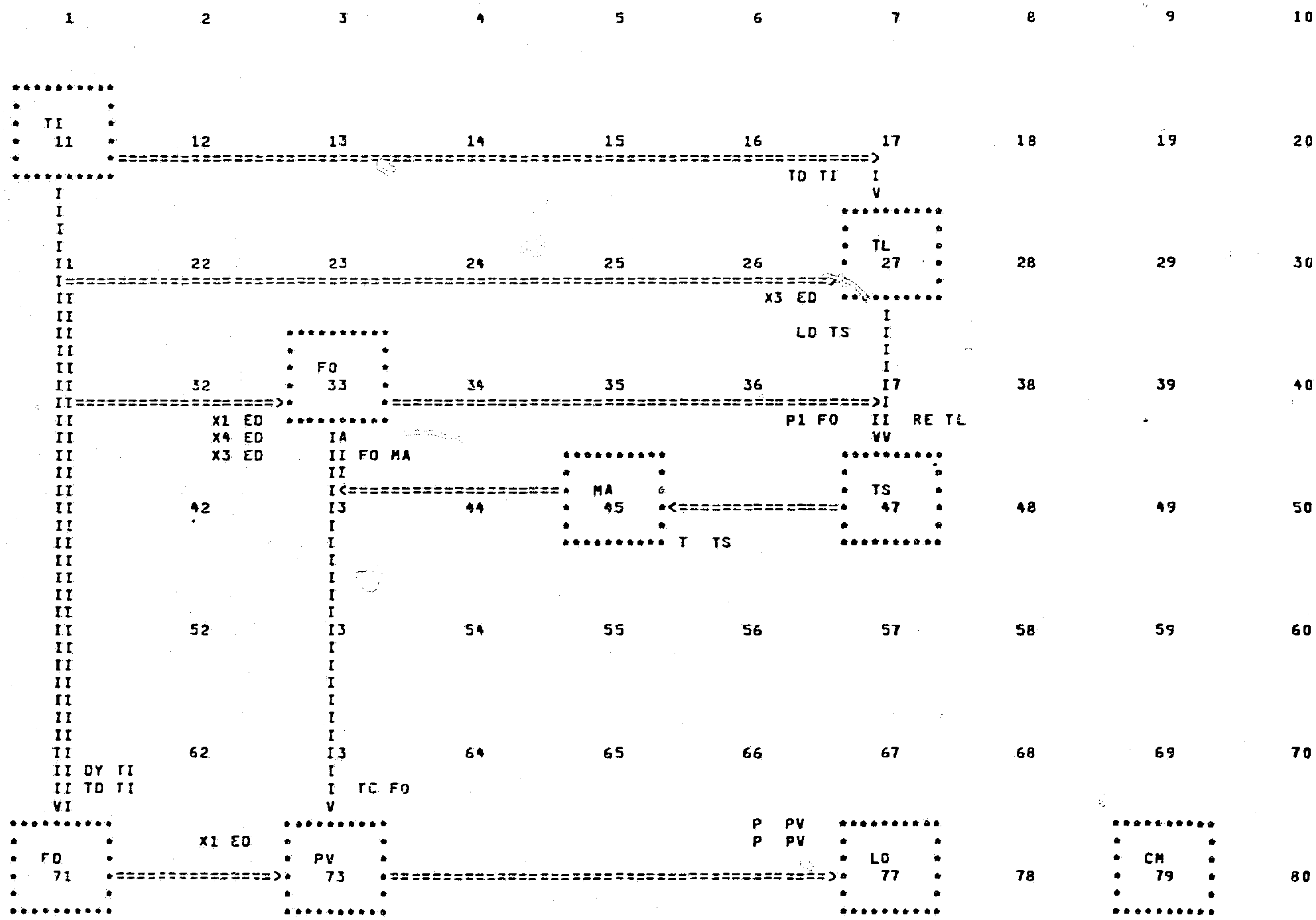


Figure 9.3-2 Fresnel Lens Model Schematic

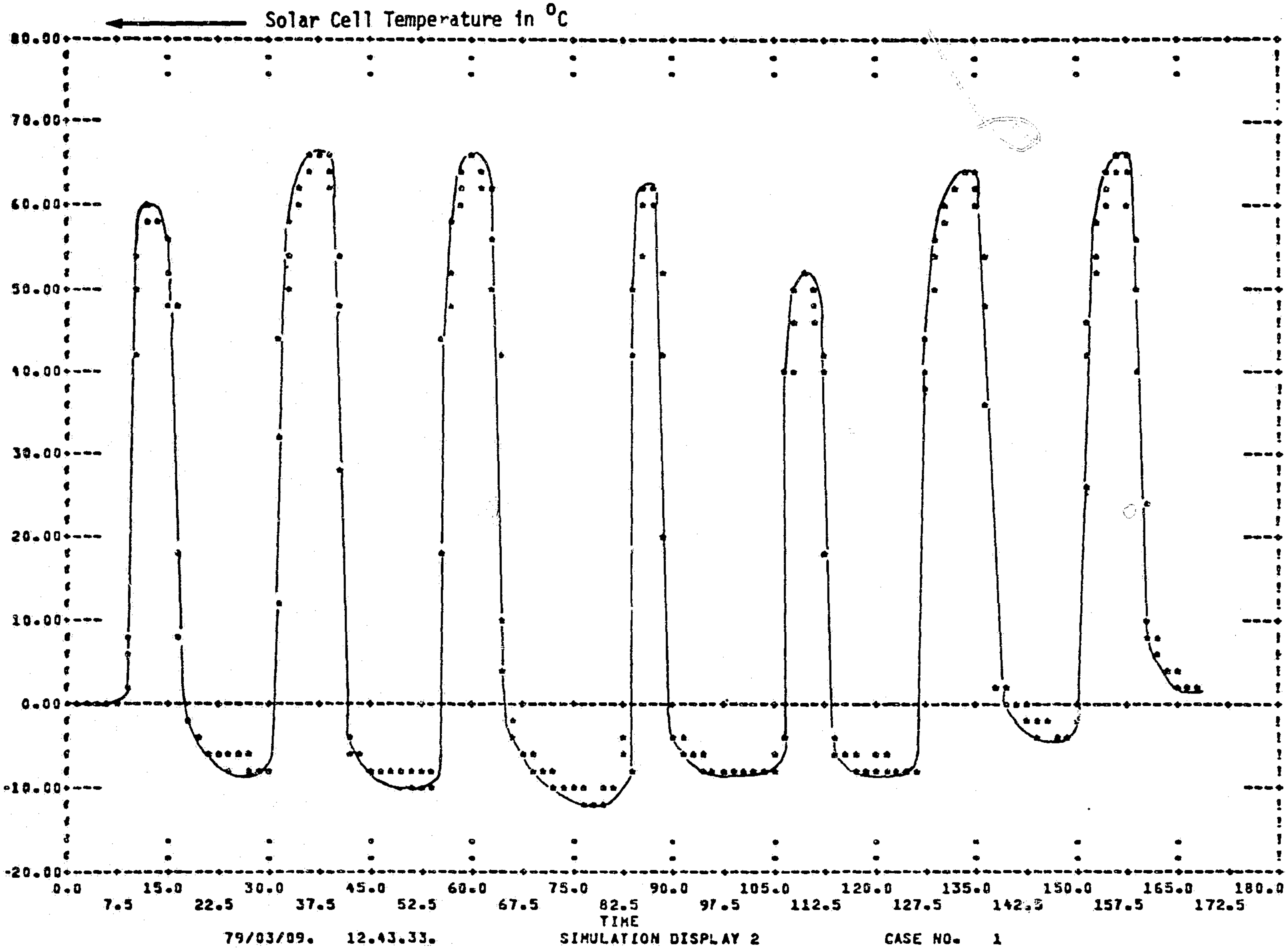


Figure 9.3-3 Solar Cell Temperature for One Week Simulation

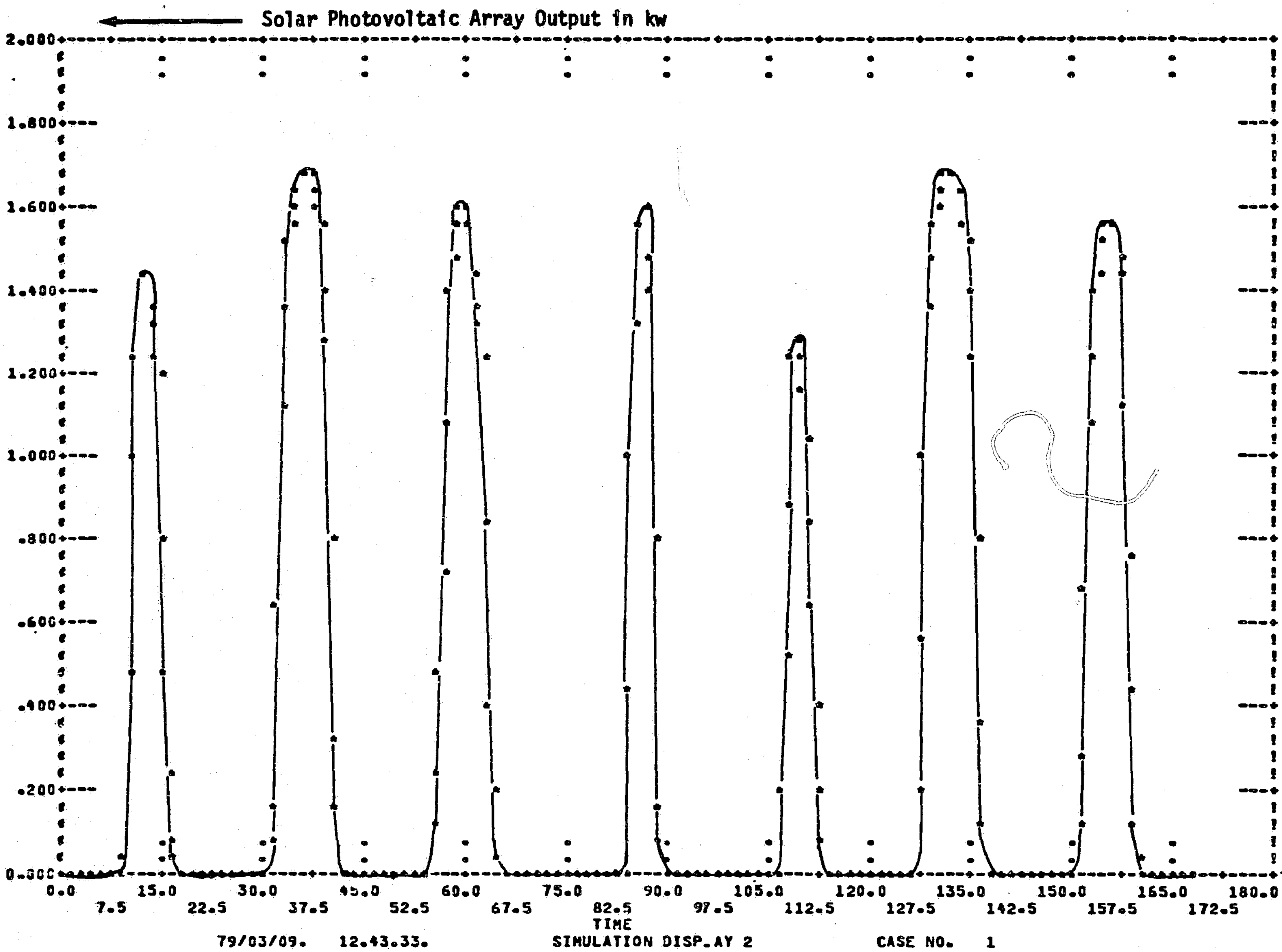


Figure 9.3-4 Photovoltaic Array Output for One Week Simulation

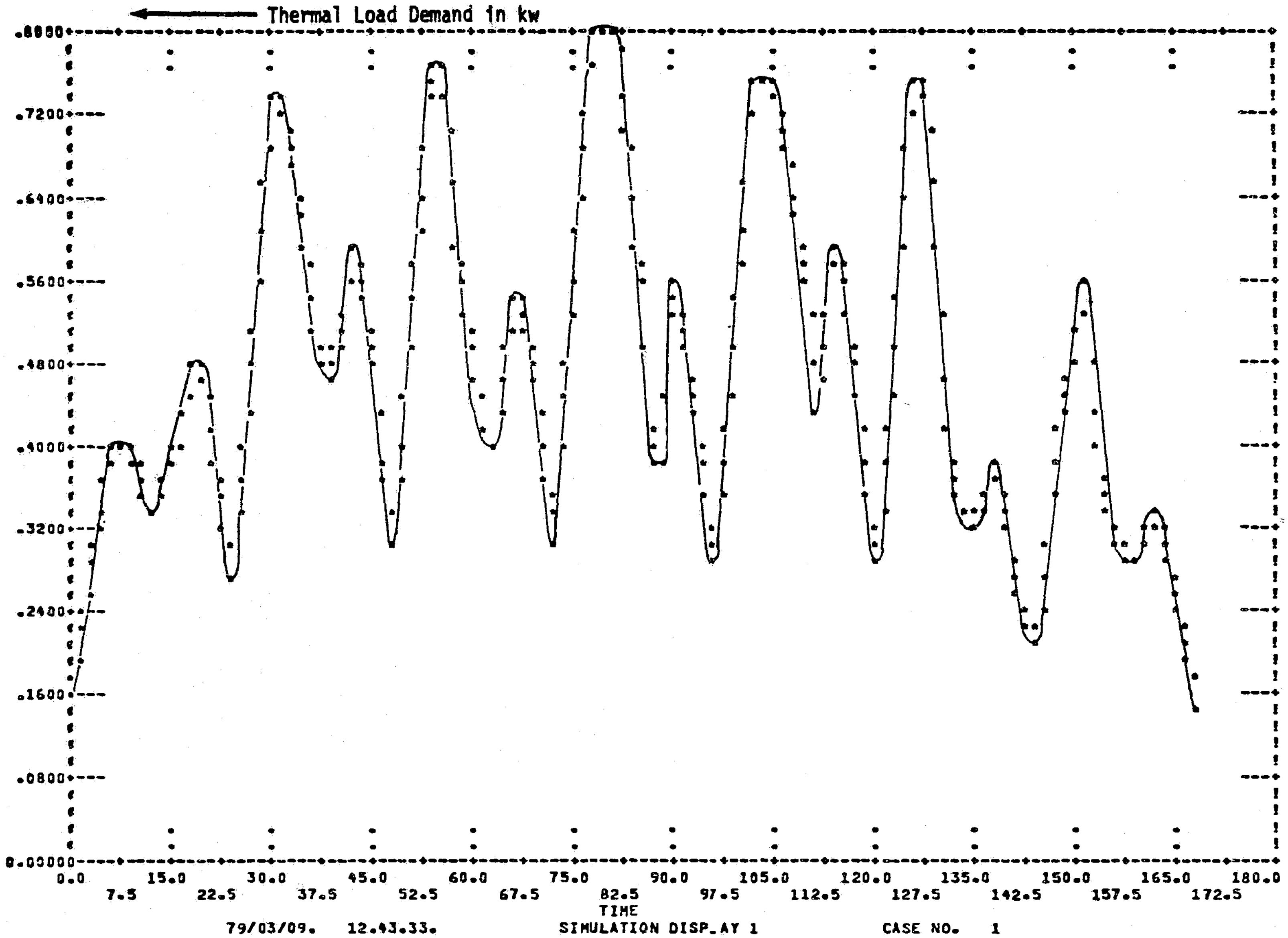


Figure 9.3-5 Thermal Load Demand for One Week Simulation

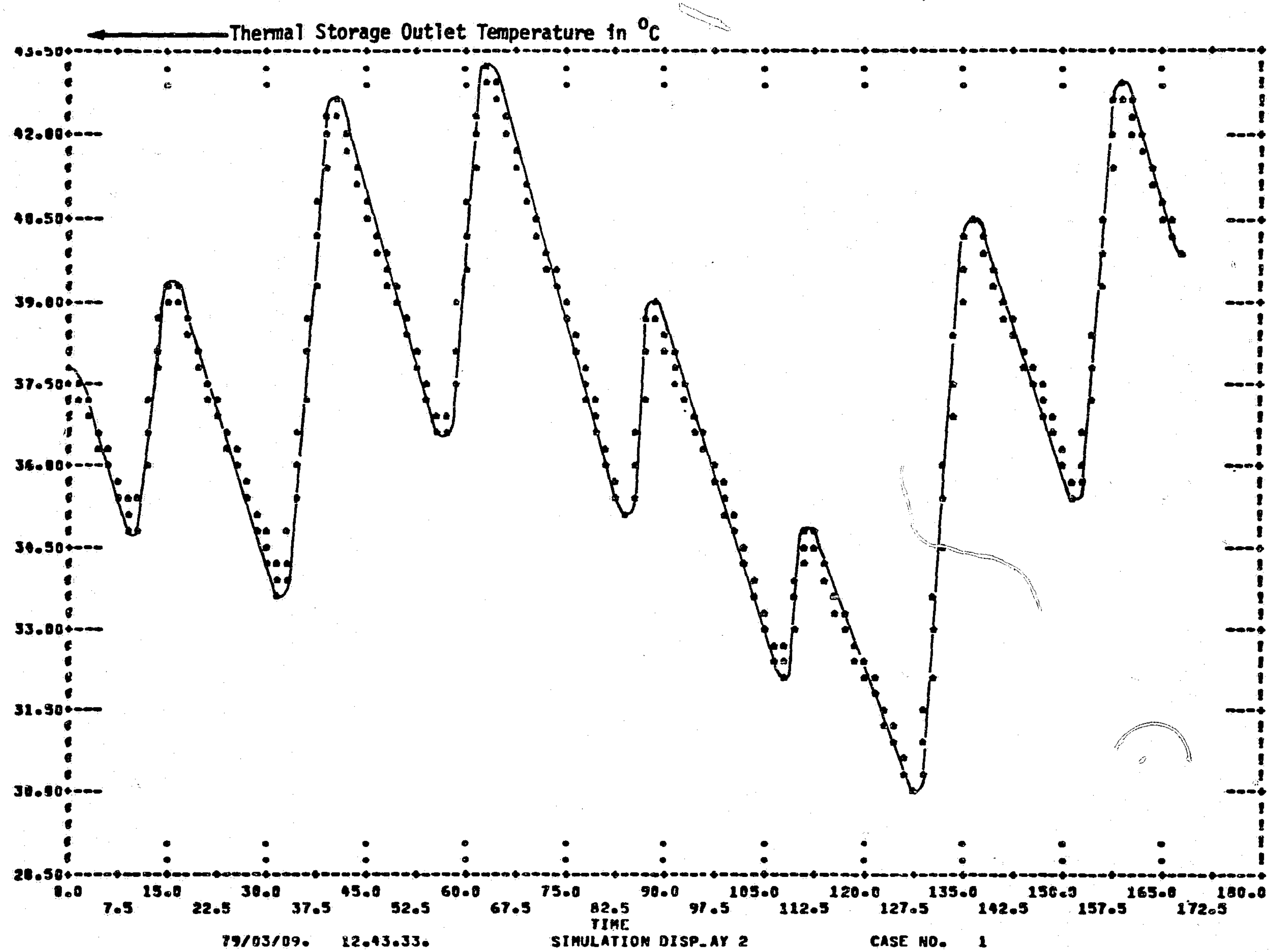


Figure 9.3-6 Thermal Storage Temperature for One Week Simulation

cycles are predominant with the periods of strong insolation providing sufficient energy to satisfy the load and compensate for thermal losses. Average load is fairly well matched to solar generation during the week since the temperature remains within a 15° channel and does not have an apparent trend away from this range.

One of the most important measures of performance for a solar energy system is the levelized cost of energy, i.e., the life cycle cost to produce one unit of usable energy including generation, storage, transmission and conversion subsystems. Energy cost may be used to size components and select most promising system alternatives, i.e., minimum energy cost is used as a selection or optimization principle. Although SIMWEST does not provide user optimization capability, optimal sizing of a few key parameters, such as the ratio of solar to utility generation and the size of storage relative to generation, is possible and may be accomplished quickly using the concept of incremental energy cost. The idea is to compute the incremental change in levelized energy cost per incremental change in capital cost, for the system parameters of interest. Given an initial system configuration and M sizing parameters to be selected, optimization proceeds as follows:

1) Perform M+1 back to back simulations to compute the cost and energy performance of the baseline configuration and M incremental configurations from the baseline.

2) Calculate the incremental energy costs for each parameter variation. Then select a new baseline configuration. Since the incremental costs are equal at the minimum cost point, increase or decrease the sizing parameters so as to equalize the new baseline incremental costs.

3) Go to 1) and continue adjusting subsystem parameters until either a performance limit is reached or until the incremental costs of the remaining parameters are equalized. (If two incremental costs are unequal, one can always lower the system energy cost by increasing the

subsystem with the smallest incremental cost at the expense of the other subsystem.)

This procedure is recommended as more efficient and economical than using a series of parametric trade studies for subsystem optimization.

The process of computing incremental costs is illustrated for the Fresnel Lens model. In the first simulation the baseline system performance and costs are computed. The second simulation differs from the first in that thermal storage capacity has been increased by 10%, and the third simulation differs from the first in that the solar collector and photovoltaic array area have been increased by 5%. Table 9.3-1 summarizes the incremental cost and simulation results for these runs. Column 1 shows the initial capital cost of the baseline system and the incremental capital costs for the thermal storage and solar array increases. (These costs are meant to be illustrative rather than representative.) Column 2 shows the results of a 20 year levelized cost analysis of the three systems, including maintenance and operating costs, e.g., the change in thermal storage increases costs by $9.10 per year. Column 3 shows the energy delivered to the loads in a year as estimated from the one week simulations. (Note: the change in storage capacity lowers the average coolant temperature, thus increasing output power.) Column 4 shows the levelized energy costs of the baseline system and of the increments in storage and generation. This column shows that the levelized energy cost will decrease as thermal storage or generation are increased, and that thermal storage is undersized relative to generation since a fixed $ increase in storage will lower the system energy cost more than the same $ increase in array area. Column 5 shows the % change in levelized energy cost given a 1% increase in capital investment. This column contains the same basic information as column 4 but provides a better quantitative measure of the economic value of increased storage capacity.

Table 9.3-1 Incremental Cost Calculations

	CC	LC	ED	EC	NIC
Baseline	7392.	1272.	7829.	16.2	----
10% Inc. in Thermal	61.	9.10	110.5	8.2	-.84
5% Inc. in Solar	319.	47.90	365.0	13.1	-.21

NOMENCLATURE:

CC = Initial Capital Cost in $

LC = Levelized Total Cost/Yr. in $
= Capital Cost*Life Cycle*Charge Rate + Maintenance Cost + Operating Cost

ED = Useful Energy Delivered/Yr. in KWH
= Electrical Load + Thermal Load + Net Change in Thermal Storage

EC = Levelized Energy Cost in ¢/KWH
= LC*100/ED

NIC = Normalized Incremental Costs
= % Change in EC Per % Change in CC
$\cong$ (ΔLC/LC - ΔED/ED)/(ΔCC/CC)

APPENDIX: UTILITY SUBROUTINES

This section provides a short description and source code for the utility subroutines called by the SIMWEST library components. These routines are also available to the user and may be called by FORTRAN statements in the user's manual. (See also page 26 of section 2.1.2 on the use of subroutines TBLU1 and TBLU2.)

- FUNCTION AINR

AINR computes the current of a photovoltaic cell given light current AIL, cell voltage V, and temperature T. Newton-Raphson iterations are used to solve the implicit equation (1) for current I:

I = AIL + BIO (1. - EXP((V+I*RS)*QBK/(T+273))) (1)

- SUBROUTINE CNVC

CNVC computes the convection coefficient HC and Reynolds number RE for air blown over a flat plate (ref. 1).

Inputs: T_A = air temperature in ^oK

T_P = plate temperature in ^oK

CL = length of plate in m

V = velocity of air in m/s

Equations:

$T_M = (T_A + T_P)/2$ (mean temp.)

$VI = 9.0 \times 10^{-8} * T_M - 1.115 \times 10^{-5}$ (viscosity)

$GR = 1.386 \times 10^{3} - 2.91 * T_M$ (Grashof's no.)

$CO = 7.25 \times 10^{-5} * T_M + 4.325 \times 10^{-3}$ (conductivity)

RE = V*CL/VI (2)

$$H_{FREE} = .116*CO*GR*|T_A - T_P|^{.333}$$

$$H_{WIND} = \begin{cases} .597*CO*REE^{.5}/CL & RE \leq 5 \times 10^5 \\ .032*CO*(RE^{.8} - 23000)/CL & \text{otherwise} \end{cases}$$

$$HC = H_{FREE} + H_{WIND} \tag{3}$$

- SUBROUTINE CUBIC

CUBIC finds the roots of the cubic equation

$$x^3 + AAx + BB = 0 \tag{4}$$

and selects the real root $\bar{x}$ with largest value.

- SUBROUTINE FLUC

FLUC computes the heat transfer coefficient HF from a collector plate into a fluid coolant. The empirical equations used are for water cooling (ref. 1).

Inputs:
- NT = number of cooling tubes
- DT = diameter of cooling tubes in m
- CW = collector width in m
- COP = conductivity of mounting plate in w/m-K
- THP = mounting plate thickness in m
- FMD = coolant mass flow rate in kg/s
- DEN = coolant density in kg/m^3
- TF = mean coolant temperature in K
- COC = coolant conductivity in w/m-K

Equations:

$$NT1 = NT/CW$$

$$HF1 = 12*NT1^2*COP*THP \quad \text{(conduction coeff.)}$$

$$VI = (21.7*(TF - 256)^{-0.8} - .185) \times 10^{-6} \quad \text{(fluid viscosity)}$$

$$PR = (.00518*TF - 1.25)**(-1.49) \quad \text{(Prandtl no.)}$$

$$RE = 4.*FMD/(\pi*DT*NT*DEN*VI) \quad \text{(Reynolds no.)} \qquad (5)$$

If $RE < 2100$,

$$HF2 = 4.36*COC*\pi*NT1$$

If $RE > 10000$

$$HF2 = .023*COC*RE^{.8}*PR^{.333}*\pi*NT1$$

If $2100 \leq RE < 10000$

$$X2 = 36.5*PR^{.33}$$

$$D2 = .0029*PR^{.33}$$

$$A = (4.36-X2)*1.6 \times 10^{-8} + D2*1.266 \times 10^{-4}$$

$$B = D2 - A*2.x10^4$$

$$C = X2 + A*10^8 - D2*10^4$$

$$HF2 = (A*RE^2+B*RE+C)*COC*\pi*NT1$$

$$HF = (1/HF1 + 1/HF2)^{-1} \qquad (6)$$

- FUNCTION HTGLAS

HTGLAS computes the top surface heat loss coefficient H_t for a collector with 1 to 3 glass covers (ref. 2).

Inputs:

N = number of glass covers (1,2,3)

T_A = ambient temperature in ^oK

T_C = mean cell temperature in ^oK

H_C = convection coefficient for air blowing over a heated flat plate in w/m^2-k

e_c, e_g = emittance of cell and glass covers

TLT = collector tilt from horizontal in degrees

Equations:

$$H_t = (N(T_C/C)/((T_C-T_A)/(N+f))^{0.33} + 1/H_C)^{-1}$$
$$+ \sigma (T_C^2+T_A^2)(T_C+T_A)/(A+(2N+f-1)/e_g-N) \qquad (7)$$

with

$$\sigma = 5.688 \times 10^{-8} \text{ w/m}^2\text{-K}^4$$
$$C = 365.9\ (1.-.00883*TLT+.0001298*TLT^2)$$
$$f = (1.-.04*H_C+.0005*H_C^2)(1.+.091*N)$$
$$A = 1/(e_C+.05*N(1-e_C))$$

- SUBROUTINE IMPLIC

IMPLIC controls the iteration logic which determines convergence of implicit variables in the user's system model, and prints convergence diagnostics. (See section 3.6 for a discussion of the iteration and diagnostic control logic.)

- SUBROUTINE RADC

RADC computes the infrared radiation coefficient HR between two bodies with surface temperatures T_1 and T_2. (See section 7.4 of Duffie and Beckman, ref. 3.)

Inputs: T_1, T_2 = surface temperatures in ^{o}K

e_1, e_2 = emittances for surfaces corresponding to T_1, T_2

$$H_R = 5.688 \times 10^{-8}\ (T_1^2+T_2^2)(T_1+T_2)/(e_1^{-1} + e_2^{-1} - 1) \qquad (8)$$

- FUNCTIONS TBLU1, TBLU2

TBLU1 and TBLU2 perform one- and two-dimension linear interpolation. A binary search is used to locate the nearest grid points for unequally spaced data. See section 2.1.2 for subroutine usage within model generation FORTRAN statements.

- SUBROUTINE UNIF

UNIF generates uniformly distributed, pseudo-random number sequences in the range [0,1]. This routine may be used to obtain random number sequences with a specified distribution function. (See for example the coding for WD in section 7.47.)

REFERENCES

1. F. Kreith, Principles of Heat Transfer, 3rd Edition, International Textbook Co., 1973.

2. S. A. Klein, M. S. Thesis, "The effects of Thermal Capacitance Upon the Performance of Flat Plate Solar Collectors", University of Wisconsin, 1973.

3. J. A. Duffie and W. A. Beckman, Solar Energy Thermal Processes, Wiley, 1974.

```
CAINR
      FUNCTION AINR(AIL,BIO,QBK,V,RS,T)
C
C       NEWTON-RALPHSON TO COMPUTE PHOTO-VOLTAIC CELL CURRENT
C
      F(A)=A-AIL-BIO*(1.-EXP(QBK*(V+A*RS)/(T+273)))
      FP(A)=1.+BIO*EXP(QBK*(V+A*RS)/(T+273))*QBK*RS/(T+273)
      A=0.
      DO 1 J=1,10
      ANEW=A-F(A)/FP(A)
      IF((ANEW-A).LE..00001)GO TO 2
    1 A=ANEW
    2 AINR=ANEW
       RETURN
      END
```

CNVC

```
      SUBROUTINE CNVC(HC,RE,TP,TA,V,CL)
C
C              COMPUTES CONVECTION COEFFICIENT HC AND REYNOLDS
C              NUMBER RE FOR AIR BLOWING OVER A FLAT PLATE.
C              CALLED BY COMPONENT FO.
C              INPUTS     TA   -AIR TEMPERATURE,K
C                         TP   -PLATE TEMPERATURE,K
C                         V    -VELOCITY OF AIR,M/S
C                         CL   -LENGTH OF PLATE,M
C
      TM=(TA+TP)*.5
      VI=9.E-8*TM-1.115E-5
      GR=1386.-2.91*TM
      CO=7.25E-5*TM+4.325E-3
      RE=V*CL/VI
      HFREE=.116*CO*GR*((ABS(TA-TP))**(.333))
      HWIND=.597*CO*SQRT(RE)/CL
      IF(RE.GT.5.E5)HWIND=.032*CO*(RE**(.8)-23000.)/CL
      HC=HFREE+HWIND
      RETURN
      END
```

CUBIC

```
      SUBROUTINE CUBIC(AA,BB,ANS)
      TER=AA**3/27.
      TERM=BB**2/4.+TER
      IF(ABS(TERM).GT..0001)GO TO 10
C     ****************************************************************
C     THREE REAL ROOTS, TWO EQUAL
C     ****************************************************************
      AB=2.*CBRT(-BB/2.)
      ABB=-AB/2.
C     ****************************************************************
C     SELECT POSITIVE ROOT
C     ****************************************************************
      ANS=AMAX1(AB,ABB)
      RETURN
   10 IF(TERM.LT.0.)GO TO 20
C     ****************************************************************
C     ONE REAL ROOT, TWO CONJUGATE IMAGINARY ROOTS
C     ****************************************************************
      STERM=SQRT(TERM)
      AAA=CBRT(-BB/2.+STERM)
      BBB=CBRT(-BB/2.-STERM)
C     ****************************************************************
C     SELECT REAL ROOT
C     ****************************************************************
      ANS=AAA+BBB
      RETURN
C     ****************************************************************
C     THREE REAL, UNEQUAL ROOTS
C     ****************************************************************
   20 STER=SQRT(-TER)
      THETA=ACOS(-BB/2./STER)
      TE=2.*SQRT(-AA/3.)
      THETA3=THETA/3.
      X1=TE*COS(THETA3)
      X2=TE*COS(THETA3+2.09439)
      X3=TE*COS(THETA3+4.18879)
C     ****************************************************************
C     SELECT SMALLEST POSITIVE ROOT
C     ****************************************************************
      ANS=AMAX1(X1,X2,X3)
      RETURN
      END
```

```
CFLUC
      SUBROUTINE FLUC(HF,RE,NT,DT,CW,CCS,THS,FMD,DEN,TF,COC)
C
C           COMPUTES HEAT TRANSFER COEFFICIENT HF TO FLUID
C           AND REYNOLDS NUMBER.
C           CALLED BY COMPONENT FO
C           INPUTS    NT   -NUMBER OF COOLING TUBES
C                     DT   -DIAMETER OF COOLING TUBES
C                     CW   -COLLECTOR WIDTH,M
C                     CCS  -CONDUCTIVITY OF MOUNTING PLATE,W/M-K
C                     THS  -MOUNTING PLATE THICKNESS,M
C                     FMD  -COOLANT MASS FLOW RATE,KG/S
C                     DEN  -COOLANT DENSITY,KG/M3
C                     TF   -MEAN COOLANT TEMPERATURE,K
C                     COC  -COOLANT CONDUCTIVITY,W/M-K
C
      REAL NT,NT1
C     WRITE(6,108)FMD,DEN,TF,COC
C 108 FORMAT(1H0,5X,*FLUC INPUTS *,4F10.2)
      PR=(.00518*TF-1.25)**(-1.49)
      NT1=NT/CW
      HF1=12.*NT1*NT1*CCS*THS
      VI=(21.7*(TF-256.)**(-.8)-.165)*1.E-6
      RE=4.*FMD/(3.1416*DT*NT*DEN*VI)
      IF(RE.GT.2100.)GO TO 1
      HF2=4.36*COC*3.1416*NT1
      GO TO 5
    1 IF(RE.GT.10000.)GO TO 2
      X2=36.5*(PR**(.33))
      D2=.0029*(PR**(.33))
      A=(4.36-X2)*1.6E-8+D2*1.266E-4
      B=D2-A*2.E4
      C=X2+A*1.E8-D2*1.E4
      HF2=(A*RE*RE+B*RE+C)*COC*3.1416*NT1
      GO TO 5
    2 CONTINUE
      HF2=.023*COC*(RE**(.8))*(PR**(.333))*3.1416*NT1
    5 CONTINUE
      HF=1./(1./HF1+1./HF2)
C     WRITE(6,109)HF,RE
C 109 FORMAT(1H0,5X,*FLUC OUTPUTS *,2F10.2)
      RETURN
      END
```

```
CHTGLAS
      FUNCTION HTGLAS(NG,TA,TC,HC1,EC,EG,TLT)
C
C        TOP HEAT LOSS COEFFICIENT HT FOR GLAS COVERS, CALLED BY FP
C
C
C        INPUTS
C             NG=NUMBER OF GLASS COVERS (1,2,3)
C             TA=AMBIENT TEMPERATURE,K
C             TC=MEAN CELL TEMPERATURE,K
C             HC1=CONVECTION COEFFICIENT FOR AIR BLOWING OVER
C                 A HEATED FLAT PLATE, W/M2-K
C             EC,EG=EMITTANCE OF CELL AND GLASS COVERS
C             TLT=COLLECTOR TILT FROM HORIZONTAL IN DEGREES
C
      REAL NG
      SIGMA=5.668E-8
      C=365.9*(1.-.00883*TLT+.0001298*TLT*TLT)
      F=(1.-.04*HC1+.0005*HC1*HC1)*(1.+.091*NG)
      A=1./(EC+.05*NG*(1.-EC))
      G=NG*(TC/C)/(((TC-TA)/(NG+F))**0.33) + 1./HC1
      B=SIGMA*(TC*TC+TA*TA)*(TC+TA)/(A+(2.*NG+F-1.)/EG-NG)
      HTGLAS=1./G+B
      RETURN
      END
```

```
CIMPLIC
      SUBROUTINE IMPLIC(CYCLES,DLINES)
      COMMON/CIMPL/IMPL,ICNT /CORDER/ NOX,NOV /COLD/VOLD
      COMMON /CV/ V /CNAMEV/ NAMEV /CTIME/ TIME
      DIMENSION V(1),NAMEV(1),VOLD(1)
C ******         UNIVAC VERSION CODE ONLY
C     IF(CYCLES.LE.0.) GO TO 40
C ******
      IF(IMPL.GT.0)GO TO 10
      SP=0
      ITERS=CYCLES
      ITERS= MAXO(1,MINO(ITERS,20))
      ILINES= ABS(DLINES)
      ITNO= 0
      IMPL=1
      DO 5 I=1,NOV
    5 VOLD(I) = 0.
   10 CONTINUE
C ******            CDC VERSION CODE ONLY
      IF(CYCLES.GE.1.) GO TO 15
      IMPL=2
      IF(ICNT.GE.ILINES) IMPL=3
      RETURN
C ******
   15 IF(IMPL.GT.1) GO TO 20
      ITNO= ITNO+1
      IF(ITNO.GE.ITERS) IMPL=2
      ICON=1
      DO 30 I=1,NOV
      IF(ABS(V(I)).LT. 1.E-6) GO TO 30
      IF( ABS(VOLD(I)-V(I)) .GT. 0.03*ABS(V(I)) )ICON=0
      VOLD(I)= V(I)
   30 CONTINUE
      IF(ICON.EQ.1) IMPL=2
      IF(IMPL.EQ.2 .AND. ICNT.GE.ILINES) IMPL=3
      RETURN
C
C
   20 ITNO=0
      IF(IMPL.GT.2) GO TO 40
      IF(ICON.EQ.1) GO TO 40
      IF(DLINES.LT.0.) GO TO 40
      ICK=0
      DO 50 I=1,NOV
      IF( ABS(V(I)).LT.1.0E-6) GO TO 50
      IF( ABS(VOLD(I)-V(I)) .LT. 0.05*ABS(V(I)) )GO TO 50
      IF(ICK.EQ.0) WRITE (6,100) TIME
  100 FORMAT(1H0,10X,5HTIME=,F9.2)
      WRITE(6,200) NAMEV(I),VOLD(I),V(I)
  200 FORMAT(1H ,10X,A6,28H  NONCONVERGENCE. OLD VALUE=,F12.3,
     1  13H   NEW VALUE=,F12.3)
      ICK=1
   50 CONTINUE
      IF(ICK.EQ.1) ICNT=ICNT+1
   40 IMPL=4
      RETURN
      END
```

```
CRADC
      SUBROUTINE RADC(HR,T1,T2,E1,E2)
C
C         COMPUTES INFRARED RADIATION COEFFICIENT HR
C         CALLED BY COMPONENT FG
C          INPUTS    T1,T2  -SURFACE TEMPERATURES,K
C                   E1,E2  -CORRESPONDING SURFACE EMITTANCES
C
      HR=5.686E-8*(T1*T1+T2*T2)*(T1+T2)/(1./E1+1./E2-1.)
      RETURN
      END
```

```
CTBLU1
      FUNCTION TBLU1(X,XT,FT,NDX,NX)
C
C     PURPOSE      ONE DIMENSION LINEAR INTERPOLATION
C
C     CALL SEQUENCE
C
C                 X   - VALUE OF INDEPENDENT VARIABLE
C                 XT  - ARRAY OF LENGTH ABS(NX) CONTAINING X VALUES
C                 FT  - ARRAY OF TABLE VALUES CORRESPONDING TO XT
C                 NDX- INDICATOR FOR STEP SPACING
C                         IF NDX.EQ.0 THEN XT CONTAINS EQUAL SPACED DATA
C                         IF NDX.NE.0 THEN XT CONTAINS UNEQUAL SPACED DATA
C                 NX  - ABS(NX) IS THE ARRAY LENGTH
C                         IF NX.LT.0 THEN TRUNCATE OUTSIDE TABLE RANGE
C                         IF NX.GE.0 THEN EXTRAPOLATE OUTSIDE TABLE RANGE
C
C     WRITTEN BY A.W.WARREN                               VERSION 1, APRIL 1977
C
      DIMENSION XT(1),FT(1)
      NA=IABS(NX)
      IF(NA.GT.1)GO TO 5
      TBLU1=FT(1)
      RETURN
    5 IF(NDX.NE.0) GO TO 100
C
C                                     EQUI-SPACED TABLE INTERPOLATION
C
      X0= XT(1)
      H= XT(2)-XT(1)
      XI= (X-X0)/H +1.
      I=XI
      IF(I.GT.0) GO TO 10
      TBLU1= FT(1)
      IF(NX.GE.0)TBLU1= FT(1) + (XI-1.)*(FT(2)-FT(1))
      RETURN
   10 IF(I.LT.NA) GO TO 20
      TBLU1=FT(NA)
      IF(NX.GE.0) TBLU1= FT(NA) + (XI-NA)*(FT(NA)-FT(NA-1))
      RETURN
   20 TBLU1= FT(I) + (XI-I)*(FT(I+1)-FT(I))
      RETURN
C
C                                     UNEQUAL SPACED TABLE INTERPOLATION
C
  100 IF(X.GE.XT(1)) GO TO 30
      TBLU1=FT(1)
      IF(NX.GE.0) TBLU1= FT(1) + (X-XT(1))*(FT(2)-FT(1))/(XT(2)-XT(1))
      RETURN
   30 IF(X.LT.XT(NA)) GO TO 40
      TBLU1= FT(NA)
      IF(NX.GE.0) TBLU1=FT(NA)+(X-XT(NA))*(FT(NA)-FT(NA-1))/(XT(NA)
     1    - XT(NA-1))
      RETURN
   40 I=1
      IGE= NA
   50 II=(IGE+I)/2
      IF(X.LT.XT(II)) GO TO 60
```

```
      I= II
      GO TO 70
   60 IGE= II
   70 IF(I+1.LT.IGE) GO TO 50
      TBLU1= FT(I) + (FT(I+1)-FT(I))*(X - XT(I))/(XT(I+1)-XT(I))
      RETURN
      END
```

```
CTBLU2
      FUNCTION TBLU2(X,Y,XT,YT,FT,IX,IY,NX,NY,MX,MY)
C
C   PURPOSE     TWO DIMENSION LINEAR INTERPOLATION
C
C   METHOD      BINARY SEARCH TO FIND NEAREST GRID POINTS.
C               TBLU1 IS USED TO REDUCE THE INTERPOLATION DIMENSION.
C
C   CALL SEQUENCE
C
C               X,Y   - POINT AT WHICH INTERPOLATION IS DESIRED
C               XT,YT- ARRAYS CONTAINING INDEPENDENT VARIABLE GRID POINTS
C               FT    - TWO DIEMSNION ARRAY OF VALUES SUCH THAT FT(I,J)
C                       CORRESPONDS TO XT(I),YT(J).
C               IX,IY- INDICATORS FOR GRID SPACING
C                          IF IX=0  THEN XT CONTAINS EQUAL SPACED VALUES
C                          IF IX.NE.0 THEN XT CONTAINS UNEQUAL SPACED VALUES
C               NX,NY- ABS(NX),ABS(NY) ARE THE ARRAY DIMENSIONS FOR XT,YT
C                          IF NX.LT.0 THEN TRUNCATE OUTSIDE XT RANGE
C                          IF NX.GT.0 THEN EXTRAPOLATE OUTSIDE XT RANGE
C                          LIKEWISE FOR NY AND YT VALUES.
C               MX,MY- DUMMY ARGUMENTS.SET EQUAL TO ABS(NX), ABS(NY).
C
C   WRITTEN BY A.W. WARREN                          VERSION 1, JUNE 1977
C
      DIMENSION XT(1),YT(1),FT(1)
      NA = IABS(NX)
      MX = NA
      NB = IABS(NY)
      MY = NB
      IF(NA.GT.1)GO TO 10
      TBLU2 =  TBLU1(Y,YT,FT,IY,NY)
      RETURN
   10 IF(NB.GT.1)GO TO 20
      TBLU2 =  TBLU1(X,XT,FT,IX,NX)
      RETURN
C                                  Y OUTSIDE YT TABLE RANGE
C
   20 IF( Y.GT. YT(1))GO TO 100
      E = (Y-YT(1))/(YT(2)- YT(1))
      FF1 = TBLU1(X,XT,FT(1),IX,NX)
      TBLU2 =FF1
      IF(NY.GT.0)TBLU2 =FF1+ E*( TBLU1(X,XT,FT(NA+1),IX,NX) -FF1)
      RETURN
C
  100 IF( Y.LT. YT(NB))GO TO 200
      E = (YT(NB)-Y)/(YT(NB)-YT(NB-1))
      NB1 = NA*(NB-1)+1
      FF1 = TBLU1(X,XT,FT(NB1),IX,NX)
      TBLU2 = FF1
      IF(NY.GT.0)TBLU2 = FF1+ E*(TBLU1(X,XT,FT(NB1-NA),IX,NX) -FF1)
      RETURN
C
C                                  YT GRID SEARCH AND INTERPOLATION
C
  200 IF(IY.NE.0)GO TO 240
      I = (Y - YT(1))/(YT(2)-YT(1)) + 1.
      GO TO 300
```

```
 240 I=1
     IGE = NB
 250 II = (IGE+I)/2
     IF(Y.LT. YT(II))GO TO 260
     I= II
     GO TO 270
 260 IGE = II
 270 IF(I+1 .LT. IGE)GO TO 250
C
 300 E = (Y-YT(I))/(YT(I+1)-YT(I))
     I1= NA*(I-1)+1
     FF1 = TBLU1(X,XT,FT(I1),IX,NX)
     TBLU2 = FF1 + E*(TBLU1(X,XT,FT(I1+NA),IX,NX) -FF1)
     RETURN
     END
```

```
CUNIF
      SUBROUTINE UNIF(U,IX)
      COMMON /CIMPL/ IMPL,ICNT,ITEST
      DATA Y/253967./
      IF(IMPL.EQ.0 .AND. ITEST.EQ.1) IX=431469
      IF (IX.EQ.1) IX = 431469
      X= AMOD( IX*Y,16777216.)
      U= X/16777215.
      IX=X
      RETURN
      END
```