MODERN COMPUTING METHODS IN SOLAR ENERGY ANALYSIS

W. A. Beckman

Professor of Mechanical Engineering and
Director of the Solar Energy Laboratory
The University of Wisconsin – Madison
Madison, WI 53706 USA
Tel.: +1 608 263 1590, Fax.: +1 608 262 8464
E-mail: beckman@engr.wisc.edu

ABSTRACT. The development of solar energy system analysis and simulation tools is traced from the classical work of Hottel and Woertz in 1942 to the present day. A discussion of early calculation methods, such as the utilizabiltiy method and the f-chart method are discussed. With the advent of modern computing machines, many hour-by-hour simulation programs were developed in the 1970s and 80s. By the 1990s the TRNSYS program emerged as preeminent. The modern TRNSYS program, with its graphical interface, is discussed in detail. Planned developments to TRNSYS are also discussed.

1. INTRODUCTION

In 1942 Professor Hoyt Hottel and his graduate student Byron Woertz of the Massachusetts Institute of Technology (MIT) laid the foundation for the analysis of active solar energy systems with their classic paper on solar collectors [1]. The research at MIT started in 1938 and led to the construction of a series of solar houses, culminating in the early 1950s with solar house IV. That house was the basis of a Ph.D. thesis by Austin Whillier [2] and another classic paper by professor Hottel co-authored with Whillier [3] in 1958. That paper developed the following well-known Hottel-Whillier collector model that is still the basis for predicting the performance of active solar collectors.

$$Q_u = A_c F_R (I_T (\tau \alpha) - U_L (T_{in} - T_{amb}))^+ \qquad 1$$

where:
Q_u is the useful rate of energy gain of the collector
A_c is the collector area
F_R is the collector heat removal factor (collector effectiveness)
I_T is the solar radiation incident on the tilted collector
$(\tau \alpha)$ is the effective transmittance-absroptance product
U_L is the collector loss coefficient
T_i is the inlet fluid temperature

T_{amb} is the ambient temperature
+ implies a controller is present to prohibit negative values of Q_u

Equation 1 has stood the test of time. About the only modification of this equation that has been deemed necessary is to include the temperature dependence on U_L. Although equation 1 is an excellent tool for understanding the implications of design changes to the solar collector, in this form it is not particularly useful in estimating the long-term performance of a solar energy system.

The utilizabiltiy concept, introduced by Whillier [3], was the first attempt to predict the long-term performance of a solar system. Figure 1 illustrates the basic idea showing solar radiation for three days: a good day, a poor day and an average day. The horizontal line is the "critical level" defined as the radiation above which the collector produces useful energy. From equation 1, this critical radiation level occurs when the solar input exceeds the thermal losses, as:

$$I_{T,\, critical} = F_R\ U_L\,(T_{in} - T_{amb})/F_R\,(\tau\,\alpha) \qquad 2$$

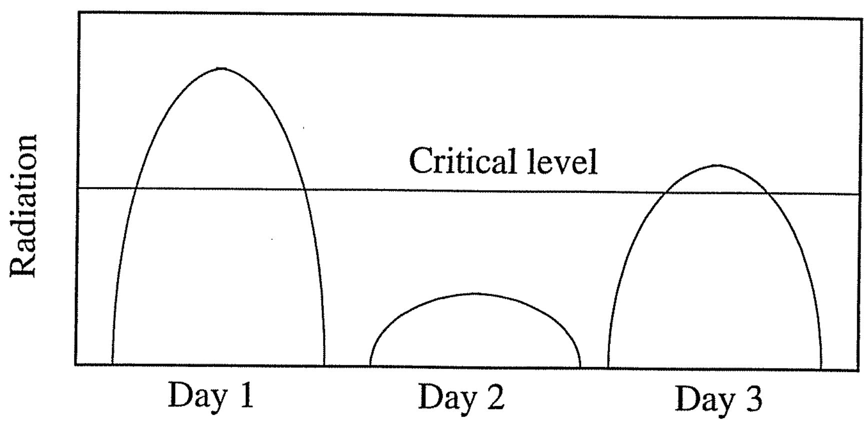

Figure 1: A sequence of good, poor and average days.

Thus, when the critical level is a constant, as illustrated by the horizontal line of Figure 1, the radiation above this line is "utilizable" by a solar system. Whillier used known statistics about the Boston MA area and was able to develop relationships called hourly utilizability. Liu and Jordan, in a series of papers [4, 5] generalized Whillier's work so that the only radiation information needed was the clearness index, K_T (the ratio of monthly average daily radiation to extraterrestrial radiation). Klein [6] extended the hourly utilizability to daily utilizability, thus reducing the calculations from 4 or 5 per month (8 to 10 hours of sunshine per day) to one calculation. Utilizability, either monthly average hourly or monthly average daily is a radiation statistic. For example, monthly average daily utilizability is defined as:

$$\bar{\phi} = \frac{\sum_{days}\sum_{hours}(I_T - I_c)^+}{N\bar{H}_T} \qquad 3$$

where $N\overline{H}_T$ is the number of days in the month times the monthly average daily radiation on the tilted surface, I_T is the radiation on the tilted surface and I_c is the critical radiation.

The utilizabiltiy concept has been extended to other types of solar energy systems. We can view Figure 1 as representing the electrical output of a photovoltaic system and the critical level as the load. Consequently, the energy above the critical level is the energy that must be stored, discarded or sold back to the utility. If we view Figure 1 as representing the radiation through a window that is absorbed by a building and the critical level as the building load, then the radiation above the critical level is the energy that must either be stored in the building or removed to maintain the comfort level. Klein and Beckman [7] review these and other utilizability concepts.

The utilizability concept, although very powerful, requires that the critical level be a constant for each month (monthly utilizability) or for each hour of a month (hourly utilizability). Some solar systems closely follow this requirement: pool heating systems, solar heating systems using pebble bed storage*, PV systems with a constant load like a radio and others. However, many solar systems have a variable critical level. In a solar system with a water storage tank the temperature to the collector depends upon the state of charge of the thermal store. Klein and Beckman [8] used utilizability to predict the maximum performance of water-storage solar systems and applied empirical correlations to estimate the difference between maximum possible performance and actual performance due to the non-constant critical level. The method is called the $\overline{\phi}$, f-chart method and has replaced the earlier f-chart method [9].

Correlation methods, such as the $\overline{\phi}$, f-chart method, are restricted to the systems for which the correlations were developed. Developing these correlations is a very time-consuming job, requiring many hour-by-hour simulations. Consequently, only a few systems have been studied in sufficient detail to develop the necessary correlations. New systems or systems of limited interest require hour-by-hour simulations to develop design guidelines and if warranted, then the detailed correlations can be developed. However, with the ever-increasing speed of desktop computers, hour-by-hour simulations are becoming more common for the normal thermal design process.

One of the earliest solar system simulation studies was by Sheridan, Bullock and Duffie [10] who, in 1967 used an analog computer. Analog computers were soon replaced by digital computers and a large number of digital simulation computer programs were developed. Merriam and Ranctore [11] reviewed the available energy system simulation programs in 1982. The only program in that review that is still in general use is TRNSYS ('tran-sis').

---

* A pebble bed is very well thermally stratified so that the inlet temperature to the collector is very nearly equal to the house temperature except when the system is oversized, as it is in the spring or fall.

2. THE TRNSYS PROGRAM

TRNSYS, originally developed in 1972-73 and available since 1975, is designed to simulate the transient performance of thermal energy systems. TRNSYS relies on a modular approach to solve large systems of equations described by FORTRAN subroutines. Each FORTRAN subroutine contains a model for a system component. For example, Subroutine Type 32 contains a model of a cooling coil. The inlet flow rates and temperatures for the air and water are inputs to the model, while the total and latent cooling rates are among the outputs of the model. By creating an input file, the user directs TRNSYS to connect the various subroutines to form a system. The TRNSYS engine calls the system components based on the input file and iterates at each timestep until the system of equations is solved.

Unlike many other programs, TRNSYS allows users to completely describe and monitor all interactions between system components. For example, the user determines the connections between the output of the pump and other system components. The modularity of the program allows the user to have as many pumps, chillers, cooling coils and solar panels as necessary, in any desired configuration. Because the components are written in FORTRAN, a user can easily generate a TRNSYS component to model any new technology that is created. Historically, TRNSYS has been used for simulating solar thermal systems, modern renewable energy systems including PV and wind power, more general HVAC systems, and buildings.

TRNSYS 14.2 for Windows, first released in the spring of 1996, culminates a tremendous international effort among universities, research facilities, distributors, and users to move the popular TRNSYS 14.1 for DOS software to the Windows 95 and Windows NT platforms. TRNSYS 14.2 features many improvements to the existing graphical user-interface programs in the TRNSYS package and the addition of an alternative front-end for TRNSYS, IISiBat.

The TRNSYS 14.2 executable program has changed significantly from previous versions. The executable is now an independent Windows program that controls the Windows display during the simulation. The executable calls a Dynamic Link Library (DLL) containing the actual TRNSYS FORTRAN source code. This combination of Windows program and DLL allows TRNSYS to take advantage of the Windows display with few modifications to the original TRNSYS source code.

During a TRNSYS simulation, up to 20 system variables can be plotted on-line. The on-line plot is a powerful diagnostic tool for the simulation formulation and allows verification that the simulation is progressing as desired. The user initializes the format of the On-line Display and chooses variables to be plotted in the TRNSYS input file; the On-line Display is the component called Type 65. But even when the simulation is running, the user may pause the TRNSYS simulation, change the scale of the plot, hide one or more variables on the plot, check the precise numerical value of any of the variables, or zoom in on a portion of the display window. If an On-line Display is not desired for a simulation, TRNSYS displays a simple window indicating the progress of the simulation.

As it has been since its conception more than 25 years ago, the TRNSYS source code is written in FORTRAN. In order to modify component models in the standard TRNSYS library or add new components to TRNSYS, a modern FORTRAN compiler, capable of producing 32-bit Dynamic Link Libraries, is required. However, no compiler is required in order to use the components in the standard library. TRNSYS may still be used outside the Windows operating system (under DOS or UNIX). However, the utility programs in the TRNSYS package and the TRNSYS executable, which handles the On-line Display, are designed for use only with Windows 95 or Windows NT. A FORTRAN compiler is required for using TRNSYS outside Windows.

The standard TRNSYS package comes with many common component models. Additional models are available from the TRNSYS web site (see appendix).

3. TRNSHELL

One of the cornerstones of the today's use of TRNSYS is the TRNSHELL program. With TRNSYS version 13.2 and earlier, the only items distributed to TRNSYS users were the FORTRAN source code and the manual containing a description of each FORTRAN component. This left the user to provide the FORTRAN compiler, an editor to work with input files and view output files, a method for keeping track of all the necessary files, a plotting package for analyzing output, etc. The user also had to type in all the information contained in the input file. With TRNSYS 14.1 for DOS, several of these problems and inconveniences were addressed by including several utility programs with the TRNSYS package. TRNSHELL brings together in one program all of the activities associated with building, running, and analyzing a TRNSYS simulation. TRNSHELL features a full-featured editor for writing and viewing TRNSYS input files, FORTRAN subroutines, and output files. Compiling and linking of FORTRAN subroutines can be executed from TRNSHELL with a single click of the mouse that executes the proper commands for your compiler. TRNSHELL can perform parametric TRNSYS simulations, varying the values of one or more system variables between runs, and store all of the results.

TRNSHELL offers extensive plotting capabilities for TRNSYS simulation results. Following a simulation, plots may be generated of any variable over time or two variables as functions of one another. An example is shown in Figure 2. Complete plotting options include overlaying of plots from one or more output files, plot proportion adjustment, colors, fonts, and more. As always, any variable or parameter in the TRNSYS simulation may be written to a file with any desired frequency. These text files may then be imported into any external analysis software.

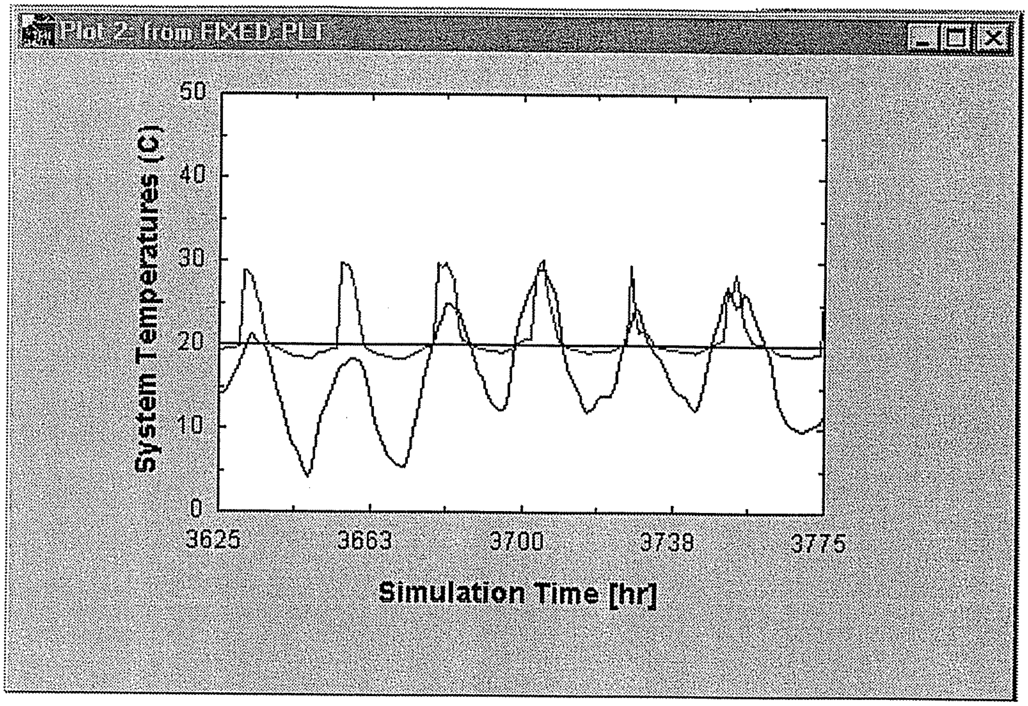

Figure 2: TRNSYS simulation output results plotted in TRNSHELL

4. TRNSED

TRNSYS users often share simulations with non-TRNSYS users. TRNSED makes this exchange simple. For the TRNSYS programmer, the standard TRNSYS input file is relatively easy to read and edit. However, manipulation of the file might be confusing for those not familiar with its syntax. With the addition of a few commands to the TRNSYS input file by the programmer, TRNSED, as shown in Figure 3, eliminates unwanted details and generates a professional, customized display of selected parameters and inputs. TRNSED provides a refined interface in which selected inputs may be viewed and/or changed and simulations run. Like TRNSHELL, TRNSED offers a parametric simulation capability as well as post-simulation plotting. Detailed help, unit conversion and input checking are also included at the discretion of the TRNSED programmer.

5 THE TWO TRNSYS PACKAGES

From the efforts of several research facilities, two separate TRNSYS packages have been developed. The packages differ only in the graphical front-end program included for automating the process of composing a TRNSYS input file and some other functions. The Solar Energy Research Center (SERC) in Borlange, Sweden has moved the PRESIM program from DOS to Windows. Independently, the Center for Scientific Research in Buildings (CSTB) in Sophia Antipolis, France has developed a general front-end program, IISiBat, designed for use with any simulation program.

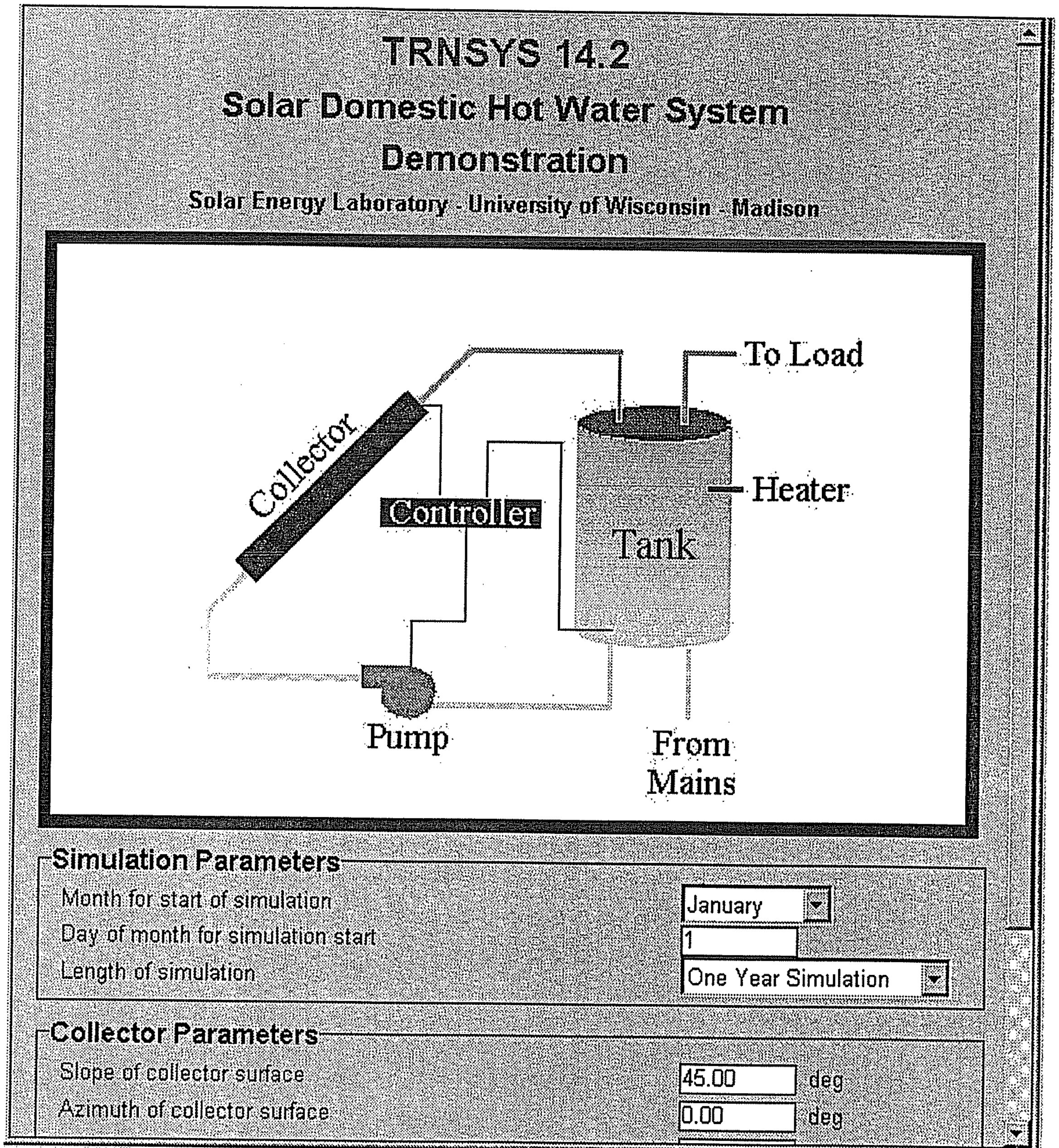

Figure 3: A TRNSED input window allows easy changes to the TRNSYS input file

CSTB has adapted IISiBat specifically for use with TRNSYS. The IISiBat environment replaces the functionality of TRNSHELL and PRESIM in the other version of TRNSYS (although TRNSHELL is included with both versions for backward compatibility). PRESIM and IISiBat had their origin in the perceived difficulty of creating a complete TRNSYS input file to describe the interconnection of components and other simulation parameters. Each TRNSYS component has inputs and outputs that represent the actual pipes, ducts, and control signals of their physical counterparts. The TRNSYS input file tells TRNSYS which parameters and initial values to use for each system component and how the various system components are linked together. For each component, a number of parameters (time-independent quantities), inputs (time-dependent quantities that may be outputs of other components), and initial values must be entered into the input file. For example, below is a section of an input file that indicates to TRNSYS that one of the system components is a solar collector of 6.5 m2 area that receives its solar radiation information from a weather file.

```
UNIT 1 TYPE 1 SOLAR COLLECTOR
PARAMETERS 14
1 1 6.5 4.19 1 50 .7 15 0 -1 0 1 0.1 0
INPUTS 10
3,1 3,2 0,0 5,5 6,6 6,4 6,5 0,0 6,9 6,10
20 200 0 10 0 0 0 0.2 20 45
```

TRNSYS input files can be difficult to interpret visually for the novice user. PRESIM and IISiBat were devised to simplify the authoring of input files and to provide other features that speed the TRNSYS simulation process. Both packages share the same FORTRAN source code, many of the same manuals, and many of the same interface components. The decision between the two packages should be based on the user's personal preference.

6 USING TRNSYS WITH PRESIM

The PRESIM program simplifies the creation of TRNSYS input files. A user initiates the simulation process through the Utility menu of the TRNSHELL program or by launching PRESIM independently. With the PRESIM program, users can create, store, retrieve and change system drawings and input files for TRNSYS. When the graphical specification of the thermal energy system is complete, PRESIM generates a complete TRNSYS input file based on this information. PRESIM checks for formal errors, such as unconnected inputs, and problem areas are highlighted on the screen. The generated TRNSYS input file is fully commented in case questions arise later.

PRESIM divides the standard TRNSYS components into several libraries that are displayed on the left side of the screen. The user drags the icons representing the desired system components onto the working area. The components are easily connected by successive clicks of the mouse. The inputs and outputs of the components are displayed on their icons. Drawing a line from the output of one component to the input of another component makes a connection. PRESIM individually displays all of the connections between the components (i.e., all the outputs that are connected to inputs are displayed as individual connections). Double clicking on a component icon allows easy modification of the initial values for the component inputs via a dialog box. Using the icons, an entire TRNSYS input file can be specified within PRESIM. PRESIM will then write the TRNSYS input file based on the connections between icons and the information entered (or default information) about the initial values of inputs. The final TRNSYS input file that PRESIM generates has many more comments than even a diligent user would normally enter into a typical TRNSYS file. The PRESIM files and drawings created with the older DOS version of PRESIM can be used in the latest Windows version of PRESIM as well. This feature makes it easy for users of previous versions to convert their work to the Windows version.

7 USING TRNSYS WITH IISiBat

IISiBat, which can be roughly translated from French as "Intelligent Interface for the Simulation of Buildings," is a general simulation environment program that has been adapted to house the TRNSYS simulation software. By its flexible nature, many powerful tools and utility programs can be controlled from the IISiBat shell. In this way, a complete simulation package can be incorporated into one environment program, from simulation engine and graphical connection programs to plotting and spreadsheet software. These functions include editing FORTRAN and input files, displaying listing and output files, plotting results, offering online help, running parametric simulations, and providing shortcuts for several repetitive tasks such as FORTRAN compiling and linking. Similar, in theory, to PRESIM, IISiBat has an integrated pre-processing utility that allows the TRNSYS user to graphically create TRNSYS input files by connecting inputs and outputs of icons that represent TRNSYS components.

The main IISiBat window contains many descriptive icons with lines connecting them. Each icon represents a different system component (e.g., pump, solar collector, etc.). The user drags the necessary icons into the Assembly Window. The user creates links between the components that share information. The lines connecting the icons represent the pipes and wires that connect the physical components. While several connections are possible between two components, just one link is displayed. To view or change the input and output connections, the user clicks on the link itself to view a detailed window. A series of Tools on the left side of the window allow the user to place icons onto the working area, connect the icons as necessary, run the simulation, access the editor, access the spreadsheet and plotting packages, and perform many other functions.

In IISiBat, all essential TRNSYS activities are controlled directly from the Assembly Window. For example, to plot the results of a simulation run, the user clicks the Spreadsheet Tool and then clicks on the Printer Icon. The output file generated by this TRNSYS component is automatically loaded into the spreadsheet to be manipulated and plotted.

8 TRNSYS 15

Planning is underway for the next TRNSYS update. The developers meet yearly to coordinate efforts and discuss plans. Some of the planned changes are:

1. Add more HVAC components
2. Add TRNSED commands to permit more complex TRNSED applications.
3. Multiple Online (graphical) display

4. Use of mathematical notation for writing TRNSYS types in addition to FORTRAN
5. Improvements in the building type (Type 56) including:
   a) solar exchange from zone to zone through internal windows
   b) thermal stratification within a zone
   c) comfort calculations
   d) floor heating and cooling
   e) room radiation view factor calculations
   f) integration with programs like AutoCad
6. Upgrade of user interface for both PRESIM and IISiBat

REFERENCES

[1] Hottel, H.C. and Woertz , B.B. (1942) Performance of flat-plate solar collectors, *Trans ASME* 64:91

[2] Hottel, H.C., and Whillier, A. (1958) Evaluation of flat plate collector performance, Transactions of the Conference On the Use of Solar Energy, 2

[3] Whillier, A., (1953) Solar energy collection and its utilization for house heating, Ph.D. thesis in Mechanical Engineering, MIT

[4] Liu, B.Y.H. and Jordan, R.C. (1960) The interrelationships and characteristic distribution of direct, diffuse and total radiation, *Solar Energy* 4

[5] Liu, B.Y.H. and Jordan, R.C. (1963) The long-term average performance of flat-plate solar energy collectors, *Solar Energy* 7

[6] Klein, S.A. (1978) Calculation of flat-plate collector utilizability, *Solar Energy* 21

[7] Klein, S.A. and Beckman, W.A. (1989) Review of solar radiation utilizability, *Journal of Solar Energy Engineering*, 106

[8] Klein, S.A. and Beckman, W.A. (1979) A general design method for closed-loop solar energy systems, *Solar Energy* 22

[9] Klein, S.A., Beckman, W.A. and Duffie, J.A. (1976) A design procedure for solar heating systems, *Solar Energy* 18

[10] Sheridan, N.R., Bullock, K.J., and Duffie, J.A., (1967) Study of solar processes by analog computer, *Solar Energy* 11

[11] Merriam, R.L. and Ranctore, R.J. (1982) Evaluation of existing programs for simulation of residential building energy use. Report EA 2572 to the Electric Power Research Institute, Palo Alto, CA.

Appendix: TRNSYS World Distributors

For more information about the TRNSYS program or for ordering information, contact one of the distributors listed below or (http://sel.me.wisc.edu.trnsys)

Solar Energy Laboratory
University of Wisconsin-Madison
1500 Engineering Drive
Madison, WI 53706
USA
Telephone: (608) 263-1589
FAX: (608) 262-8464
Email: trnsys@sel.me.wisc.edu

Thermal Energy System Specialists (TESS)
5610 Medical Circle - Suite 31
Madison, WI 53719
USA
Telephone: (608) 274- 2577
FAX: (608) 278-1475
Email: tess@bestware.net

TRANSSOLAR Energietechnik GmbH
Nobelstr. 15
70569 Stuttgart
GERMANY
Telephone: +49 711 679 760
FAX: +49 711 679 7611
Email: transsolar@transsolar.com

Solar Energy Research Center
Hogskolan Dalarna
S-781 88 Borlange
SWEDEN
Telephone: +46 23 77 87 11
FAX: +46 23 77 87 01
Email: presim@blg.du.se

Centre Scientifique et Technique du Batiment
290 Route des Lucioles, BP 209
06 904 Sophia Antipolis
FRANCE
Telephone: +33 (0)4 93 95 67 46
FAX: +33 (0)4 93 95 67 33
Email: iisibat@cstb.fr

Laboratoire de Thermodynamique
Universite de Liege
Campus du SART-TILMAN
Parking P 33 - Batiment B 49
B - 4000 Liege
BELGIUM
Telephone: +32 (0)4 366 48 00
FAX: +32 (0)4 366 48 12
Email: thermoap@vm1.ulg.ac.be