United States Army
Aeromedical Research Laboratory
Fort Rucker, Alabama 36362-5292

User's Manual
for BRNSIM/BURNSIM:
A Burn Hazard Assessment Model

By
F. S. Knox, III
Dena Bonetti
Chris Perry

Escape and Impact Branch
Biodynamics and Biocommunications Division

February 1993

Approved for public release; distribution unlimited.
Notice

Qualified requesters

Qualified requesters may obtain copies from the Defense Technical Information Center (DTIC), Cameron Station, Alexandria, Virginia 22314. Orders will be expedited if placed through the librarian or other person designated to request documents from DTIC.

Change of address

Organizations receiving reports from the U.S. Army Aeromedical Research Laboratory on automatic mailing lists should confirm correct address when corresponding about laboratory reports.

Animal use

In conducting the research described in this report, the investigators adhered to the Guide for Laboratory Animal Facilities and Care, as promulgated by the Committee on the Guide for Laboratory Animal Resources, National Academy of Sciences-National Research Council.

Disposition

Destroy this document when it is no longer needed. Do not return it to the originator.

Disclaimer

The views, opinions, and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other official documentation. Citation of trade names in this report does not constitute an official Department of the Army endorsement or approval of the use of such commercial items.

Reviewed:

Charles A. Salter
CHARLES A. SALTER
LTC, MC
Director, Biomedical Applications Research Division

Released for publication:

Roger W. Wiley, O. D., Ph.D.
Chairman, Scientific Review Committee

David H. Karney
Colonel, MC, SPS Commanding
BURNSIM is an interactive computer model which runs on DEC minicomputers (PDP 11 and VAX), Macintosh and IBM compatible PCs. The model is based on the work of Moritz and Henriques at Harvard, the Surgery Department at University of Rochester; Alice Stoll at Naval Air Development Center and Knox et al. at the U.S. Army Aeromedical Research Laboratory. Its development has been funded by the U.S. Army, U.S. Air Force, and Dr. Knox. The model predicts time to pain and burn depth when bare skin is exposed to any arbitrary time history of heat flux. It predicts burn depth with reasonable accuracy for pig and human skin. A software module to include clothing between the thermal source and the skin has been developed but not integrated with BURNSIM and has not been validated. By using sensors to measure heat flux behind fabric it has been possible to use BURNSIM to evaluate the insulating effect of clothing. BURNSIM has been used in the last several years to assess the burn hazard associated with rocket plumes in side-by-side ejection seats, shoulder launched weapons, nuclear flash and live fire. This manual provides information on model development, its installation and use on a PC.

13. ABSTRACT (Maximum 200 words)

Approved for public release; distribution is unlimited.

14. SUBJECT TERMS

Burns
Computer model
Thermal protection
Burn hazard prediction

15. NUMBER OF PAGES

70

16. PRICE CODE

Unlimited

17. SECURITY CLASSIFICATION OF REPORT

Unclassified

18. SECURITY CLASSIFICATION OF THIS PAGE

Unclassified

19. SECURITY CLASSIFICATION OF ABSTRACT

Unclassified

20. LIMITATION OF ABSTRACT

Unlimited
Acknowledgments

The senior author wishes to express his gratitude to the many colleagues who have contributed to the development of BURNSIM. Drs. Stanley C. Knapp, Thomas L. Wachtel, and Chap McCahan have worked on the project from the beginning in 1970 when we were all stationed at the U.S. Army Aeromedical Research Laboratory. Dr. Knapp had started the project to study postcrash fires with funding from various sources including the U.S. Army Medical Research and Development Command and the USAF Life Support Systems Program Office at Wright-Patterson Air Force Base. Dr. Cal Lum helped during the data collection phase in 1972. Dr. Daniel D. Reneau and his graduate student, Nelson O'Young, contributed the basic section of code to calculate the conductive heat transfer. Randy Nockton and Chet Ellis contributed additional software development including tape reading routines, statistical analysis programs and a database to manage the data. Dr. Charles Yuell, a pathologist at the Rochester School of Medicine, helped clarify the burn depth grading scheme and even found some of the original porcine skin samples from earlier burn studies at Rochester. Mr. Chris Perry and CPT Dena Bonetti have worked with BURNSIM for the past 3 years at the Armstrong Laboratory and have contributed to the current effort supported by the Defense Nuclear Agency. Part of this effort is the PC version of BURNSIM for which CPT Dena Bonetti is largely responsible. Funding for this effort has come from the U.S. Army Medical R&D Command, Defense Nuclear Agency, and the U.S. Air Force without which BURNSIM would not have been developed.
Table of contents

Background ................................................. 1
Model description .......................................... 3
  Getting started ......................................... 3
  Helpful hints ........................................... 19
References .................................................. 21
Bibliography ............................................... 23
Appendix A .................................................. 33
Appendix B .................................................. 46
Appendix C .................................................. 66

List of illustrations

Figure 1. Tissue damage integral indicative of the blister endpoint . 2
Figure 2. Skin representation .................................. 4
Figure 3. Skin temperatures at first six nodes calculated
  with Burnsim for Stoll’s data ............................ 19
Background

BF.NSIM (or BURNSIM as it is now called) is a computer model which allows the user to convert heat flux incident to bare skin to a predicted burn depth. The requirement for such a model arose when there was a need to quantify the thermal protective properties of new flight suits. Techniques employed in the 1960s and very early 1970s did not predict the full range of burns from no burn to full thickness and failed to take into account both initial conditions of the skin and its adaptive behavior when heated.

Since the late 1960s, the U.S. Army Aeromedical Research Laboratory (USAARL) at Fort Rucker, Alabama, has been involved in quantifying the burn hazard associated with post crash fires and the protective capability of flight clothing. USAARL staff (including the author) conducted a number of field studies using burning helicopters to establish the severity and time course of post crash fires (Knapp and Knox, 1982). They also 1) built and used two fire simulators to study the effect of simulated postcrash fires on pigs as an analog for man (Knox et al., 1978b), 2) placed fabrics between the fire and the pigs to study their protective capability (Knox et al., 1980), 3) assembled a large porcine (pig) burn database using this bioassay method (Knox, 1979a), and 4) developed the model, BRNSIM, to decrease the workload associated with using the bioassay method to assess fabric protective capabilities (Knox, 1979b).

The starting point for building BRNSIM (short for burn simulation) was the work of Alice Stoll who based her model on Moritz and Henriques' damage integral (Henriques, 1947). She had collected data from human volunteers on the time/heat flux relationships resulting in threshold transpidermal necrosis. This burn is represented by minor blister formation. To explain her results she added a consideration of damage occurring during cooling as well as during the heating phase (Figure 1). Stoll chose the constants (Stoll and Greene, 1959) in her model to fit her human data on threshold burns; more severe burns were not at first considered. Later Weaver and Stoll (1969) proposed an extension of Stoll's first model to include more severe burns without experimental basis.

The first model to come out of the USAARL program was that of Art Takata of IITRI (Takata, 1974) who worked for USAARL as a contractor. He started with Stoll's approach and added water boiling as a way of handling blister formation. He then adjusted the constants (P,ΔE) (see equation (7) in Appendix A) to more accurately predict USAARL's data on more severe porcine burns.

The development of this model and the work upon which it is based has been funded by U.S. Army Medical Research & Development Command, Fort Detrick; U.S. Air Force Life Support Systems Program Office and Armstrong Laboratory, Wright-Patterson Air Force Base, Ohio; Defense Nuclear Agency, Washington D.C., and as a personal project by the primary author.
Figure 1. Tissue damage integral indicative of the blister endpoint (Stoll and Chianta, 1971)
The current BURNSEM model builds on these earlier efforts (Henriques, 1947; Weaver and Stoll, 1969; Mehta and Wong, 1973; Morse et al., 1973; and Takata, 1974). It is an interactive model written in both FORTRAN and ZBASIC and runs on PDP 11/40, 11/03, 11/24, VAX 11/780, Macintosh, and IBM compatible PCs.

**Model description**

BURNSEM considers the skin to be represented as 12 chunks or nodes (See Figure 2). Seven additional nodes can be inserted between the first and second nodes when exposures are mild and burn damage is likely to be shallow (Figure 2). BURNSEM solves the Fourier heat conduction equation to find temperature as a function of time at each node. Then total damage at each node is found by computing the damage integral at each depth. The transition between normal and damaged skin is defined as that depth where the damage integral is equal to one. For a more detailed description of the mathematics of BURNSEM consult Appendix A. BURNSEM source code (FORTRAN version) can be found in Appendix B.

**Getting started**

BURNSEM has been supplied to you in either source or compiled form. The following instructions are intended to help you use the model. The instructions and comments are based, in part, on feedback received from several users who have attempted to get started without the benefit of this manual. If you have problems using BURNSEM please do not hesitate to call the author at DSN 785-3931 or (513) 255-3931. Future versions of this manual will incorporate your comments and suggestions so that we may continue to improve BURNSEM and to distribute updated versions to the users.

The first step is to load the code for the model into your computer from the medium provided. This step has many versions. Only one example is given because it is assumed that if you are using this model you are sufficiently computer literate to load and compile the source code on your system.

PC Example: BURNSEM.FOR, REN12.DAT on floppy disk.
To run off hard disk:
Set default disk drive to a:
Put diskette in a:
Type DIR (rtn)
BURNSEM.FOR REN12.DAT FLUX.DAT BURNSEM.EXE
A>CD C:
C>MD C:\BURNSIM
C>CD C:\BURNSIM
C>COPY a:\*.* c:
C:\BURNSIM>DIR
BURNSEM.FOR REN12.DAT FLUX.DAT BURNSEM.EXE
C:\BURNSIM>
Figure 2. Skin representation
At this point you have made a directory on your hard disk for BURNSIM and copied the files from the floppy to the new directory. The file REN12.DAT contains the initial values of all the variables which are changeable within BURNSIM. Some of these values such as the conductivity and heat capacity for each node can only be changed by creating a new REN12.DAT with an editor or word processing program (see Appendix C for the layout of REN12.DAT). The model expects a flat ASCII file, so if you use a word processor, save the file as an ASCII text file and not a document. Other values such as exposure time (ETIME) can be changed interactively as described below.

To run BURNSIM invoke the command for your system, e.g. RUN BURNSIM or BURNSIM. You will next see the following on the screen:

BURNSIM <CR>

The first screen that you see is shown below:

WELCOME TO BURNSIM. TO BEGIN RUNNING THE PROGRAM, BURNSIM FIRST NEEDS TO KNOW THE NAME OF THE FILE THAT YOU WANT TO STORE THE OUTPUT DATA IN. THIS FILE WILL CONTAIN ALL OF THE INPUT PARAMETERS AS WELL AS THE OUTPUT FOR EACH ITERATION THE MODEL PERFORMS. THIS FILE CAN BE CALLED ANYTHING UP TO EIGHT CHARACTERS LONG.

PLEASE ENTER A NAME FOR THE OUTPUT FILE: OUTFILE <CR>

The next screen is shown below:

NEXT BURNSIM WILL SHOW YOU THE PRESENT INPUT PARAMETERS. UNDER THE LIST OF PARAMETERS YOU WILL SEE A QUESTION ASKING IF YOU WISH TO CONTINUE. IF YOU WANT TO EXIT THE PROGRAM AT THAT POINT, TYPE N. OTHERWISE TYPE Y.

TO CONTINUE ON TO THE LIST OF PARAMETERS TYPE A <CR>.

The following screen will appear:

SKIN DIFFUSION DATA
INPUT PARAMETER LIST

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>TEMPO</td>
<td>32.5000</td>
</tr>
<tr>
<td>BL</td>
<td>.22000</td>
</tr>
<tr>
<td>TEMPB</td>
<td>4.5000</td>
</tr>
<tr>
<td>PL1</td>
<td>1.46000</td>
</tr>
<tr>
<td>PL2</td>
<td>2.24000</td>
</tr>
<tr>
<td>ETIME</td>
<td>3.02000</td>
</tr>
<tr>
<td>BLOOD</td>
<td>.00100</td>
</tr>
<tr>
<td>APL1</td>
<td>.78000</td>
</tr>
<tr>
<td>APL2</td>
<td>.60000</td>
</tr>
<tr>
<td>DENS</td>
<td>1.00000</td>
</tr>
<tr>
<td>AK</td>
<td>.01000</td>
</tr>
<tr>
<td>ABSORB</td>
<td>.61300</td>
</tr>
<tr>
<td>PLN1</td>
<td>147.37000</td>
</tr>
<tr>
<td>PLN2</td>
<td>239.47000</td>
</tr>
<tr>
<td>ITIME</td>
<td>80.00000</td>
</tr>
<tr>
<td>NXTRA</td>
<td>0</td>
</tr>
<tr>
<td>Q1</td>
<td>3.54000</td>
</tr>
<tr>
<td>JINC</td>
<td>12</td>
</tr>
<tr>
<td>BOIL</td>
<td>100.15000</td>
</tr>
<tr>
<td>DE1</td>
<td>50000.0</td>
</tr>
<tr>
<td>DE2</td>
<td>80000.0</td>
</tr>
<tr>
<td>ADE1</td>
<td>93534.9</td>
</tr>
<tr>
<td>ADE2</td>
<td>39109.8</td>
</tr>
</tbody>
</table>
DO YOU WISH TO CONTINUE? TYPE Y OR N Y <CR>

Answer yes (Y) to continue and you will be presented with the following choices:

TYPE THE NUMBER OF THE FUNCTION BELOW THAT YOU WISH TO PERFORM.

CHOOSE A FUNCTION NUMBER:

1 - CHANGE SELECTED INITIAL VALUES
2 - NO CHANGES--CONTINUE RUNNING THE PROGRAM
3 - EXIT

PLEASE ENTER THE FUNCTION NUMBER: 1 <CR>

Choose #1 to change the set up values. This will give you the following screen:

SKIN DIFFUSION DATA
INPUT PARAMETER LIST

TEMPI0 = 32.5000  DENS = 1.00000  Q1 = 3.54000
BL = .22000      AK = .01000      JINC = 12
TEMPB = 4.5000   ABSORB = .61300   BOIL = 100.15000

PL1 = 1.46000    PLN1 = 147.37000  DE1 = 50000.0
PL2 = 2.24000    PLN2 = 239.47000  DE2 = 80000.0
ETIME = 3.02000  ITIME = 80.00000  NXTRA = 7
BLOOD = .00100

APL1 = .78000    APLN1 = 285.52000  ADE1 = 93534.9
APL2 = .60000    APLN2 = 117.43000  ADE2 = 39109.8

DO YOU WANT TO MAKE ANY CHANGES? Y OR N Y

Answer yes (Y) here to change the input values using the following screen:

PICK A NUMBER

1=TEMPI0  8=ETIME
2=DENS     9=PL1
3=Q1      10=PLN1
4=BL      11=PL2
5=AK      12=PLN2
6=JINC     13=DE1
7=TEMPB  14=DE2
Choose the number representing the parameter you wish to change. The definition of these parameters is in Table I. For example, set up one of Stoll’s published cases (Weaver and Stoll, 1969). In this case, the human skin was exposed for 5.6 seconds at 0.4 cal/cm²-sec. The skin was blackened with India ink to set the absorbivity at 94 percent. Start by choosing #3 to set the incident flux level, Q1. The model responds with:

ENTER THE FLUX FILE NAME (TYPE A <CR> IF NO FILE IS TO BE USED): <CR>

Since Stoll’s case has a constant flux value, type a (CR) and the following will appear on the screen:

CONSTANT Q-VALUE = 3.5400000 INPUT NEW VALUE: 0.4 <CR>

The old value was 3.54000 and the new value entered was 0.4 cal/cm²-sec.

***Q*********************************************************************

If you do later simulations where you wish to read in a file of varying flux values instead of using a constant flux value, type the name of the flux file in response to the following statement:

ENTER THE FLUX FILE NAME (TYPE A <CR> IF NO FILE IS TO BE USED): FLUX.DAT

The file FLUX.DAT is the example flux file given on the disk. When creating flux files to be read into BURNSIM, remember that the file name can be no more than eight characters in length including the .DAT ending. Also the file must contain only one column of data, the flux data, in units of cal/cm²-sec. The number of points in the flux file and the sample interval between points must be known, too.

Continue to input responses to the following statements concerning the flux file as they appear on the screen:

ENTER FLUX ID (UP TO 8 CHARACTERS): IDFLUX (CR)

The FLUX ID can be any combination of 8 characters.

ENTER THE NUMBER OF POINTS IN THE FLUX PROFILE: 100 (CR)
The maximum number of points that can be read in is 600.

ENTER THE SAMPLE INTERVAL IN SECONDS: 0.1 (CR)

*** Note: When using a flux file for the incident flux, the exposure time (ETIME) variable must be set equal to the number of points in the flux file minus one times the sample interval in seconds. For this example, ETIME = (100 - 1) x (0.1) = 9.9 seconds.

The next screen is shown below:

SKIN DIFFUSION DATA
INPUT PARAMETER LIST

TEMPIO = 32.5000  DENS = 1.00000  Q1 = .40000
BL = .22000  AK = .01000  JINC = 12
TEMPB = 4.5000  ABSORB = .61300  BOIL = 100.15000

PL1 = 1.46000  PLN1 = 147.37000  DE1 = 50000.0
PL2 = 2.24000  PLN2 = 239.47000  DE2 = 80000.0
ETIME = 3.02000  ITIME = 80.00000  NXTRA = 7
BLOOD = .00100

APL1 = .78000  APLN1 = 285.52000  ADE1 = 93534.9
APL2 = .60000  APLN2 = 117.43000  ADE2 = 39109.8

DO YOU WANT TO MAKE ANY CHANGES? Y OR N Y

Answer yes (Y) here to change the next input value using the following screen:

PICK A NUMBER

1=TEMPIO  8=ETIME
2=DENS  9=PL1
3=Q1  10=PLN1
4=BL  11=PL2
5=AK  12=PLN2
6=JINC  13=DE1
7=TEMPB  14=DE2
15=ITIME  16=ABSORBTIVITY
17=BOIL  18=EXTRA NODES
19=BLOOD  20=APL1
21=APLN1  22=APL2
THE NUMBER OF EXTRA NODES IS: 0 INPUT NEW VALUE: 7

ENTER NEW VALUES SEPARATED BY A COMMAS, OR A <CR>
IF THE PROGRAM IS TO CALCULATE VALUES. 25,.50,.75,.100,.125,.150,.175.

The next screen is shown below:

SKIN DIFFUSION DATA
INPUT PARAMETER LIST

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>TEMPIO</td>
<td>32.5000</td>
</tr>
<tr>
<td>DENS</td>
<td>1.00000</td>
</tr>
<tr>
<td>Q1</td>
<td>0.40000</td>
</tr>
<tr>
<td>BL</td>
<td>0.22000</td>
</tr>
<tr>
<td>AK</td>
<td>0.01000</td>
</tr>
<tr>
<td>JINC</td>
<td>12</td>
</tr>
<tr>
<td>TEMPB</td>
<td>4.50000</td>
</tr>
<tr>
<td>ABSORB</td>
<td>0.61300</td>
</tr>
<tr>
<td>BOIL</td>
<td>100.15000</td>
</tr>
<tr>
<td>PL1</td>
<td>1.46000</td>
</tr>
<tr>
<td>PLN1</td>
<td>147.37000</td>
</tr>
<tr>
<td>DE1</td>
<td>50000.0</td>
</tr>
<tr>
<td>PL2</td>
<td>2.24000</td>
</tr>
<tr>
<td>PLN2</td>
<td>239.47000</td>
</tr>
<tr>
<td>DE2</td>
<td>80000.0</td>
</tr>
<tr>
<td>ETIME</td>
<td>3.02000</td>
</tr>
<tr>
<td>ITIME</td>
<td>80.00000</td>
</tr>
<tr>
<td>NXTRA</td>
<td>7</td>
</tr>
<tr>
<td>BLOOD</td>
<td>0.00100</td>
</tr>
<tr>
<td>APL1</td>
<td>0.98000</td>
</tr>
<tr>
<td>APLN1</td>
<td>285.52000</td>
</tr>
<tr>
<td>ADE1</td>
<td>93534.9</td>
</tr>
<tr>
<td>APL2</td>
<td>0.60000</td>
</tr>
<tr>
<td>APLN2</td>
<td>119.43000</td>
</tr>
<tr>
<td>ADE2</td>
<td>39109.8</td>
</tr>
</tbody>
</table>
| THE EXTRA NODES ARE: 25.0 50.0 75.0 100.0 125.0 150.0 175.0

DO YOU WANT TO MAKE ANY CHANGES? Y OR N Y

Answer yes (Y) here to change the next input value using the following screen:

PICK A NUMBER

1=TEMPIO  8=ETIME
2=DENS    9=PL1
3=Q1      10=PLN1
4=BL      11=PL2
5=AK      12=PLN2
6=JINC    13=DE1
7=TEMPB   14=DE2
15=ITIME  16=ABSORBTIVITY
17=BOIL   18=EXTRA NODES
19=BLOOD  20=APL1
21=APLN1  22=APL2
THE VALUE FOR BLOOD IS: 0.0100  INPUT NEW VALUE: 0.0007

The next screen is shown below:

SKIN DIFFUSION DATA
INPUT PARAMETER LIST

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>TEMPIO</td>
<td>32.5000</td>
</tr>
<tr>
<td>DENS</td>
<td>1.00000</td>
</tr>
<tr>
<td>Q1</td>
<td>.40000</td>
</tr>
<tr>
<td>BL</td>
<td>.22000</td>
</tr>
<tr>
<td>AK</td>
<td>.01000</td>
</tr>
<tr>
<td>JINC</td>
<td>12</td>
</tr>
<tr>
<td>TEMPE</td>
<td>4.50000</td>
</tr>
<tr>
<td>ABSORB</td>
<td>.61300</td>
</tr>
<tr>
<td>BOIL</td>
<td>100.1500</td>
</tr>
<tr>
<td>PL1</td>
<td>1.46000</td>
</tr>
<tr>
<td>PLN1</td>
<td>147.37000</td>
</tr>
<tr>
<td>DE1</td>
<td>50000.0</td>
</tr>
<tr>
<td>PL2</td>
<td>2.24000</td>
</tr>
<tr>
<td>PLN2</td>
<td>239.47000</td>
</tr>
<tr>
<td>DE2</td>
<td>80000.0</td>
</tr>
<tr>
<td>ETIME</td>
<td>3.02000</td>
</tr>
<tr>
<td>ITIME</td>
<td>80.00000</td>
</tr>
<tr>
<td>NXTRA</td>
<td>7</td>
</tr>
<tr>
<td>BLOOD</td>
<td>.00070</td>
</tr>
<tr>
<td>APL1</td>
<td>.78000</td>
</tr>
<tr>
<td>APLN1</td>
<td>285.52000</td>
</tr>
<tr>
<td>ADE1</td>
<td>93534.9</td>
</tr>
<tr>
<td>APL2</td>
<td>.60000</td>
</tr>
<tr>
<td>APLN2</td>
<td>117.43000</td>
</tr>
<tr>
<td>ADE2</td>
<td>39109.8</td>
</tr>
</tbody>
</table>

THE EXTRA NODES ARE: 25.0 50.0 75.0 100.0 125.0 150.0 175.0

DO YOU WANT TO MAKE ANY CHANGES? Y OR N Y

Answer yes (Y) here to change the next input value using the following screen:

PICK A NUMBER

1=TEMPIO  8=ETIME
2=DENS    9=PL1
3=Q1      10=PLN1
4=BL      11=PL2
5=AK      12=PLN2
6=JINC    13=DE1
7=TEMPB   14=DE2
15=ITIME  16=ABSORBTIVITY
17=BOIL   18=EXTRA NODES
19=BLOOD  20=APL1
21=APLN1  22=APL2
23=APLN2  24=ADE1
25=ADE2

8 <CR>
THE VALUE FOR ETIME IS:  3.02000  INPUT NEW VALUE:  5.6 <CR>

The next screen is shown below:

```
SKIN DIFFUSION DATA
INPUT PARAMETER LIST

TEMPIO = 32.5000  DENS = 1.00000  Q1 = .40000
BL = .22000  AK = .01000  JINC = 12
TEMPB = 4.50000  ABSORB = .61300  BOIL = 100.15000
PL1 = 1.46000  PLN1 = 147.37000  DE1 = 50000.0
PL2 = 2.24000  PLN2 = 239.47000  DE2 = 80000.0
ETIME = 5.60000  ITIME = 80.00000  NXTRA = 7
BLOOD = .00070

APL1 = .78000  APLN1 = 285.52000  ADE1 = 93534.9
APL2 = .60000  APLN2 = 117.43000  ADE2 = 39109.8

THE EXTRA NODES ARE:  25.0  50.0  75.0  100.0  125.0  150.0  175.0

DO YOU WANT TO MAKE ANY CHANGES? Y OR N Y
```

Answer yes (Y) here to change the next input value using the following screen:

```
PICK A NUMBER

1=TEMPIO  8=ETIME
2=DENS   9=PL1
3=Q1     10=PLN1
4=BL     11=PL2
5=AK     12=PLN2
6=JINC   13=DE1
7=TEMPB  14=DE2
15=ITIME 16=ABSORBTIVITY
17=BOIL  18=EXTRA NODES
19=BLOOD 20=APL1
21=APL1  22=APL2
23=APL2  24=ADE1  25=ADE2

16 <CR>
```

THE VALUE FOR ABSORB IS:  0.61300  INPUT NEW VALUE:  0.94 <CR>

The next screen is shown below:
SKIN DIFFUSION DATA
INPUT PARAMETER LIST

TEMPIO = 32.5000  
BL = 0.22000   
TEMPB = 4.50000  
PL1 = 1.46000  
PL2 = 2.24000  
ETIME = 5.60000  
BLOOD = 0.00070

DENS = 1.00000  
AK = 0.01000   
ABSORB = 0.94000  
DE1 = 50000.0  
DE2 = 80000.0  
PLN1 = 147.37000  
PLN2 = 239.47000  
ITIME = 80.00000  
NXTRA = 7

APL1 = 0.78000  
APL2 = 0.60000  
APLN1 = 285.52000  
APLN2 = 117.43000

THE EXTRA NODES ARE: 25.0 50.0 75.0 100.0 125.0 150.0 175.0

DO YOU WANT TO MAKE ANY CHANGES? Y OR N N

At this point all of the input values for Stoll's example case have been set, so the answer here is no (N).

*********************************************************************************************************
NOTE: If you inadvertently answer yes (Y) to make changes, and then decide not to make any, type a <CR> to exit the "PICK A NUMBER" menu, and the following question will appear:

DO YOU WISH TO CONTINUE? TYPE Y OR N Y <CR>

Type yes (Y) to continue on with the present run.

*********************************************************************************************************

Now that the correct parameters are set up, select #2 to proceed:

TYPE THE NUMBER OF THE FUNCTION BELOW THAT YOU WISH TO PERFORM.

CHOOSE A FUNCTION NUMBER:
1 - CHANGE SELECTED INITIAL VALUES
2 - NO CHANGES--CONTINUE RUNNING THE PROGRAM
3 - EXIT

PLEASE ENTER THE FUNCTION NUMBER: 2 <CR>

You are now ready to run the program. BURNSIM will ask you for some file names in which to store the data and summaries.

ENTER THE MODEL NAME OR DESCRIPTION (UP TO 80 CHARACTERS). THIS INFORMATION WILL BE USED AS A TITLE ON THE SUMMARY PAGE.
The following screen appears:

NOW ENTER THE SUMMARY FILENAME (UP TO 8 CHARACTERS). THIS FILE WILL CONTAIN A SUMMARY OF THE SIMULATION. SUM1 (CR)

Any name up to 8 characters can be used.

The next screen then appears:

NOW ENTER THE TEMPERATURE FILE (UP TO 8 CHARACTERS). THIS FILE WILL CONTAIN A LIST OF THE TEMPERATURES AT THE VARIOUS NODES DURING THE SIMULATION. TFILE1 (CR)

Any name up to 8 characters can be used.

While calculating, the model prints the following on the screen:

<table>
<thead>
<tr>
<th>T</th>
<th>XTIME=</th>
<th>TIME=</th>
</tr>
</thead>
<tbody>
<tr>
<td>32.50</td>
<td>0.0000E+00</td>
<td>0.000000</td>
</tr>
<tr>
<td>32.91</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>33.32</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>33.73</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>34.14</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>34.55</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>34.95</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>35.36</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>35.77</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>36.18</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>36.59</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>37.00</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>BLUD = .00000</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

T= 32.96  XTIME= 0.0000E+00  TIME= 0.010000
<p>| 32.92  | 0.0000E+00 |
| 33.32  | 0.0000E+00 |
| 33.73  | 0.0000E+00 |
| 34.14  | 0.0000E+00 |
| 34.54  | 0.0000E+00 |
| 34.95  | 0.0000E+00 |
| 35.36  | 0.0000E+00 |
| 35.77  | 0.0000E+00 |
| 36.18  | 0.0000E+00 |
| 36.59  | 0.0000E+00 |
| 36.99  | 0.0000E+00 |
| BLUD = .00000 |</p>
<table>
<thead>
<tr>
<th>T</th>
<th>XTIME</th>
<th>TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>44.40</td>
<td>0.0000E+00</td>
<td>1.000000</td>
</tr>
<tr>
<td>40.02</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>37.25</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>35.71</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>35.03</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>34.89</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>35.04</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>35.33</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>35.67</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>36.01</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>36.32</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>36.64</td>
<td>0.0000E+00</td>
<td></td>
</tr>
</tbody>
</table>

BLUD = .00003

<table>
<thead>
<tr>
<th>T</th>
<th>XTIME</th>
<th>TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>49.19</td>
<td>0.0000E+00</td>
<td>2.000000</td>
</tr>
<tr>
<td>44.62</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>41.24</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>38.86</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>37.29</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>36.36</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>35.90</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>35.76</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>35.83</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>36.02</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>36.27</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>36.60</td>
<td>0.0000E+00</td>
<td></td>
</tr>
</tbody>
</table>

BLUD = .00007

part of the sequence omitted to save space.

<table>
<thead>
<tr>
<th>T</th>
<th>XTIME</th>
<th>TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>44.50</td>
<td>0.0000E+00</td>
<td>13.000000</td>
</tr>
<tr>
<td>44.44</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>44.32</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>44.13</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>43.81</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>43.37</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>42.81</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>42.13</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>41.36</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>40.50</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>39.59</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>38.43</td>
<td>0.0000E+00</td>
<td></td>
</tr>
</tbody>
</table>

BLUD = .00045

<table>
<thead>
<tr>
<th>T</th>
<th>XTIME</th>
<th>TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>44.02</td>
<td>0.0000E+00</td>
<td>14.000000</td>
</tr>
<tr>
<td>43.97</td>
<td>0.0000E+00</td>
<td></td>
</tr>
<tr>
<td>43.86</td>
<td>0.0000E+00</td>
<td></td>
</tr>
</tbody>
</table>
At the conclusion of the calculations, the following information appears on the screen:

W=1 LIES ABOVE NODE 2. INTERPOLATING VALUES OF D AND W COMPUTED FROM INTERPOLATED VALUES OF D AND TEMPERATURE.

MAXIMUM TEMPERATURE = 60.056

THRESHOLD DEPTH = 104.6

FINAL TIME = 14.04

TIME TO PAIN = 1.59

TYPE A <CR> TO CONTINUE. <CR>

The next screen asks if you want to reformat the file so that it can be brought into the HARVARD GRAPHICS shell to make a plot.

DO YOU WANT TO PLOT THE TEMPERATURE VS. TIME IN HARVARD GRAPHICS? Y OR N  Y (CR)

If you answer yes (Y) then you must type in a new file name for the HARVARD GRAPHICS temperature file.
THE TEMPERATURE DATA IS STORED IN FILE: TFILE1

ENTER THE FILE TO STORE THE HARVARD GRAPHICS TEMPERATURES USING UP TO 12 CHARACTERS INCLUDING THE ENDING .DAT HGTFILE1.DAT

The following will then appear on the next screen:

THE MODEL OUTPUT IS IN FILE: OUTFILE
USE "PRINT" OR "TYPE " AFTER YOU EXIT THE PROGRAM TO SEE IT.

THE TEMPERATURES AT EACH NODE ARE IN FILE: TFILE1
USE "PRINT" OR "TYPE " AFTER YOU EXIT THE PROGRAM TO SEE IT.

THE TEMPERATURES FOR THE HARVARD GRAPHICS PLOTS ARE IN FILE: HGTFILE1.DAT USE "PRINT" OR "TYPE " AFTER YOU EXIT THE PROGRAM TO SEE IT.

THE SUMMARY PRINTOUT IS IN FILE: SUM1
USE "PRINT" OR "TYPE " AFTER YOU EXIT THE PROGRAM TO SEE IT.

TYPE A <CR> TO CONTINUE. <CR>

The following question will appear next on the screen:

DO YOU WANT TO CONTINUE? Y OR N

At this point choosing yes (Y) takes you back to the following screen:

TYPE THE NUMBER OF THE FUNCTION BELOW THAT YOU WISH TO PERFORM.

CHOOSE A FUNCTION NUMBER:
1 - CHANGE SELECTED INITIAL VALUES
2 - NO CHANGES--CONTINUE RUNNING THE PROGRAM
3 - EXIT

PLEASE ENTER THE FUNCTION NUMBER:

If you choose no (N) at the "DO YOU WANT TO CONTINUE?" you will see the following question:

DO YOU WANT TO DO ANOTHER RUN? Y OR N

If you answer yes (Y) you will be taken back to the following screen to change any desired input parameters:
SKIN DIFFUSION DATA
INPUT PARAMETER LIST

TEMPIO = 32.5000  DENS = 1.00000  Q1 = .40000
BL = .22000  AK = .01000  JINC = 12
TEMPB = 4.50000  ABSORB = .94000  BOIL = 100.15000

PL1 = 1.46000  PLN1 = 147.37000  DE1 = 50000.0
PL2 = 2.24000  PLN2 = 239.47000  DE2 = 80000.0
ETIME = 5.60000  ITIME = 80.00000  NXTRA = 7
BLOOD = .00070

APL1 = .78000  APLN1 = 285.52000  ADE1 = 93534.9
APL2 = .60000  APLN2 = 117.43000  ADE2 = 39109.8

THE EXTRA NODES ARE: 25.0 50.0 75.0 100.0 125.0 150.0 175.0

DO YOU WANT TO MAKE ANY CHANGES? Y OR N

If you answer no (N) to "DO YOU WANT TO DO ANOTHER RUN?", you will exit the BURNSIM program.

If you type the file SUM1 the following appears on the screen:

MODEL NAME OR DESCRIPTION: TEST OF A. STOLL .4CAL 5.6SEC CASE

SKIN DIFFUSION DATA
INPUT PARAMETER LIST

TEMPIO = 32.50000  DENS = 1.00000  Q1 = .40000
BL = .22000  AK = .01000  JINC = 12
TEMPB = 4.50000  ABSORB = .94000  BOIL = 100.15000

PL1 = 1.46000  PLN1 = 147.37000  DE1 = 50000.0
PL2 = 2.24000  PLN2 = 239.47000  DE2 = 80000.0
ETIME = 5.60000  ITIME = 80.00000  NXTRA = 7
BLOOD = .00070

APL1 = .78000  APLN1 = 285.52000  ADE1 = 93534.9
APL2 = .60000  APLN2 = 117.43000  ADE2 = 39109.8

THE EXTRA NODES ARE: 25.0 50.0 75.0 100.0 125.0 150.0 175.0

FLUX FILE I.D.: .00 2

FLUX(I)=
1 .400 2 .400

W= .21973E+01
W= .12061E+00
W= .14088E-01
D= -.16000E+02
\begin{align*}
D &= 0.52983E+01 \\
D &= 0.59915E+01 \\
\end{align*}

W=1 LIES ABOVE NODE 2. INTERCOLLATING VALUES OF D AND W COMPUTED FROM INTERPOLATED VALUES OF D AND TEMPERATURE.

\begin{align*}
W &= 0.10360E+01 \\
W &= 0.86923E+00 \\
W &= 0.73320E+00 \\
D &= 0.46052E+01 \\
D &= 0.46283E+01 \\
D &= 0.50106E+01 \\
W &= 0.21973E+01 \quad \text{AT DEPTH (IN MICRONS)} = 0.112535E-06 \\
W &= 0.18217E+01 \quad \text{AT DEPTH (IN MICRONS)} = 25.0000 \\
W &= 0.14992E+01 \quad \text{AT DEPTH (IN MICRONS)} = 50.0000 \\
W &= 0.12423E+01 \quad \text{AT DEPTH (IN MICRONS)} = 75.0000 \\
W &= 0.10360E+01 \quad \text{AT DEPTH (IN MICRONS)} = 100.000 \\
W &= 0.86923E+00 \quad \text{AT DEPTH (IN MICRONS)} = 125.000 \\
W &= 0.73320E+00 \quad \text{AT DEPTH (IN MICRONS)} = 150.000 \\
W &= 0.62140E+00 \quad \text{AT DEPTH (IN MICRONS)} = 175.000 \\
W &= 0.12061E+00 \quad \text{AT DEPTH (IN MICRONS)} = 200.000 \\
W &= 0.14088E-01 \quad \text{AT DEPTH (IN MICRONS)} = 400.000 \\
W &= 0.47704E-02 \quad \text{AT DEPTH (IN MICRONS)} = 600.000 \\
W &= 0.21844E-02 \quad \text{AT DEPTH (IN MICRONS)} = 800.000 \\
\end{align*}

MAXIMUM TEMPERATURE = 60.056

THRESHOLD DEPTH = 104.6

FINAL TIME = 14.04

TIME TO PAIN IS 1.59 SECONDS.
If you plot the data saved in FILE1 and overlay Stoll's measured data, we get the following:

![Graph showing skin temperature over time](image)

**Figure 3. Skin temperatures at first six nodes calculated with Burnsim for Stoll's Data**

Notice that there is reasonable fit between the computed temperature profiles and the recorded temperature. The predicted depth is 104.9 microns. Stoll observed a threshold blister, hence the damage should be between 80μm and 120μm.

**Helpful hints**

This section is devoted to explaining the inputs to the model and some hints about how to set up the model for special cases. The inputs are summarized in Table A-1.

There are nine special cases which have been found by previous users. First, for short exposures of less than 1 second, change the calculation interval (AK) from its normal value of .01 second to some value which is at least 100 times less than the duration. Thus, for an
exposure of 0.1 sec use AK = 0.001 second. Second, if the skin has been blackened, e.g., with India ink, use an absorptivity of about 0.92 to 0.94. Third, the default value of 0.613 for absorptivity assumes that 100 percent of the convective energy is absorbed, only 60 percent of the radiative energy is absorbed, and 5 percent of incident radiation is intercepted by hair. Thus, assuming

1) $Q_{incident} = 0.1 q_i \text{ (convective)} + 0.9 q_i \text{ (radiative)}$
2) 5 percent radiative is not absorbed because of hair stubble,

then $Q = 0.1 q_i + 0.6(0.9)(0.95)q_i = 0.613 q_i$

Fourth, use NTRA 7 especially for mild exposures so that shallow burn depths are calculated more accurately. Fifth, a value of 0.0007 for Blood works best for shallow human burns.

Sixth, new values for DE1, DE2, PL1, PLN1, PL2, and PLN2 can be calculated if you wish to try rate constants published by other authors (see model derivation in Appendix A).

Seventh, it is possible to calculate new thermal properties based on humidity changes. Read the paper on thermal properties published in the journal Burns (Knox et al. 1986).

Eighth, the model currently assumes that ambient temperature is 23.9°C. Thus, during cool down the surface loses heat to a 23.9°C environment. This number can be changed only in the source code in the following line:

$$\text{If } (\text{TIME} \geq \text{ETIME}) Q_l = -5 \cdot 10^{-4} (t(1) - 23.9)$$

Nine, for very severe exposures, where water boils in more than the first node, the thermal property recalculations routine causes an instability in the cool down phase. This can be seen if the data are plotted and can be avoided by setting the boil temperature to a much higher value. A permanent fix for this bug will appear in the next version of BURNSIM.
References


21


Bibliography


Bustad, L. K; 1966, Pigs in the laboratory. Scientific American. 214:94.


Hardy, J. D. 1962. Physiological effects of high intensity infrared heating. American society heating and refrigeration engineers journal.


------. 1972b. Realistic evaluation of fabrics for thermal protective clothing. Presented to the Survival and Flight Equipment Association 10th Annual Symposium; October 2-5; Phoenix, AZ.

Knox, F. S., III, McCahan, G. R., Jr., and Wachtel, T. L. 1972. The use of the pig as a bioassay substrate for evaluation of thermal protective clothing and physical sensor calibration. Presented at the Eighth Scientific Session of the Joint Committee on Aviation Pathology; October 8-12; Colorado Springs, CO. Also presented at the American Burn Association 5th Annual Meeting; 1973 April 6; Dallas, TX (16mm color, sound film, 20 min.). Published in Journal of aerospace medicine. 45:933, 1974.


Knox, F. S., III. 1979b. Predictability of burn depth: Data analysis and mathematical modeling based on USAARL's experimental porcine burn data. Shreveport, LA: Louisiana State University School of Medicine, Department of Physiology and Biophysics. Contract DAMD17-77-7004.


McClellan, P. O. 1968. Application of swine in biomedical research. Laboratory animal care. 18:120.


Appendix A

ANALYTICAL MODEL

Several years ago Weaver and Stoll (1969) proposed an extension of Stoll's earlier model (Stoll and Greene, 1959) to heat fluxes higher than those used in obtaining the experimental data upon which the earlier model had been based. They also found that the effective conductivity changed during the exposure and subsequent cooldown period. Takata (1974), using preliminary data from USAARL's Thermal Project (the uncorrected version of the current data base), formulated a model which not only predicted threshold burns but deep burns and tissue water boiling as well. Building on the work of Henriques (1947), Stoll and Greene (1959), Weaver and Stoll (1969), Mehta and Wong (1973) and Takata (1974), an analytical model was formulated as follows:

For the thermal exposure of interest, skin is essentially opaque to thermal radiation and can be considered to transfer energy internally by conduction only, since exposure durations are no longer than the minimum response times reported for increased thermoregulatory system activity (1954). Consequently, thermal energy transfer in skin can be described by the heat conduction or Fourier equation be written as follows:

$$\rho \ C_p \ \frac{\partial T}{\partial t} = \frac{\partial}{\partial x} \left( K \ \frac{\partial T}{\partial x} \right) + q$$

where,

$\rho$ = density, gm/cm$^3$

$C_p$ = heat capacity, cal/gm$^\circ$C

$K$ = thermal conductivity, cal/cm$^2$-sec$^\circ$C

$T$ = temperature, $^\circ$C

$x$ = distance, cm

$q$ = energy source, for the first nodal volume, cal/cm$^2$-sec

*Simplifying assumption base on the predominance of the radiate mode of heating. May be less valid with fabrics. In actuality a correction is made to $q$ to account for convective heating, surface absorptivity, and attenuation of radiant heating by hair.
Since skin is considered to be opaque to radiant energy from a post crash fire, and since the source term is due only to radiant energy, equation (1) applies only to the surface of the skin. For all conditions in which \( x > 0 \), equation (1) reduces to the following:

\[
\rho C_p \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}[k \frac{\partial T}{\partial x}] + q \tag{2}
\]

Solution of equation (1) and (2) requires two boundary conditions for \( x \), preferably at \( x = 0 \) and \( x = L \), and initial conditions at \( t = 0 \) for all positions \( 0 < x < L \). If one assumes that there is no backward flux of thermal energy at \( x = 0 \) (all conduction is into the skin), then the energy flux at \( x = 0 \) is zero and, consequently, \( \partial T/\partial x = 0 \). Similarly, if the problem assumes that an adiabatic backwell condition prevails at \( x = L \), the fatty tissue, then the net flux out of the system at \( x = L \) is 0, or \( \partial T/\partial x = 0 \). These two boundary conditions indicate that the system is closed and that all thermal energy added to the system, \( 0 \leq x \leq L \), is distributed within the system and cannot escape.

Initial conditions are established by specifying a uniform temperature for all locations, \( 0 \leq x \leq L \) at time \( t = 0 \).

Consequently, the system may be defined by the following mathematical model:

\[
\rho C_p \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}[k \frac{\partial T}{\partial x}] + q \quad \text{\( \forall \; x = 0 \): surface} \tag{3a}
\]

\[
\rho C_p \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}[k \frac{\partial T}{\partial x}] \quad \text{\( \forall \; 0 < x < L \)} \tag{3b}
\]

\( T_x = \text{CORE TEMPERATURE} = \text{TEMPIO} + \text{TEMPB} \)

\( T = T_o, \; 0 \leq x \leq L, \; t = 0 \) \quad \text{Initial Conditions} \tag{4}

\( \frac{\partial T}{\partial x} = 0, \; x = 0, \; 0 \leq t \leq x \) \quad \text{Boundary Conditions 1} \tag{5}

\( \frac{\partial T}{\partial x} = 0, \; x = L, \; 0 \leq t \leq x \) \quad \text{Boundary Conditions 2} \tag{6}
Solution of mathematical model (Reneau and O'Young, 1976, 1977, 1978)

An analytical solution to equation set (3) was not considered feasible due to the variable nature of q, Cp and K, so explicit differencing methods of numerical analysis were employed to solve the equations. Several investigators working with linear systems have found that the Crank-Nicholson six point implicit differencing method provided an excellent numerical solution (Crank and Nicholson, 1947). For the solution of equation set (3) of the mathematical model, it was decided to apply the Crank-Nicholson method to the second order partial derivatives and corresponding explicit methods to the first order partials.

The grid work in Figure A-1 is a representative of the differenced system from x = 0 to x = L (j's) and t = 0 to t = τ (i's).

The Crank-Nicholson technique involves averaging the value of the dependent variable over the i and i + 1 row at a constant j position. The second order derivative is then evaluated at the (j, i + 1/2) position. A forward difference formulation is applied to the term to match the same position.

The above described implicit differencing method is noted for the characteristics of stability and convergence. Correct increment sizes yield reliable convergence. The model was implemented in FORTRAN IV using solution techniques of Thomas as described by Bruce et al. (1953).
This initial model was revised to allow energy flux across the surface, \( x = 0 \), during heating, convective heat loss at the skin surface during cooling and heat transfer into deep tissues including conduction into fat, convective cooling via the blood, tissue water boiling, a temperature gradient from surface to fat and a gradient of thermal properties based on measured tissue water. The model, BURNsim, is a run interactively with the following variables changeable for each run:

### Table A-1
Model parameters changeable interactively

**INPUTS**

- **TEMP10** = Initial surface temperature, °C; nominally 32.5 °C for man
- **DENs** = Density of skin, 1.0 gm/cm³
- **Q1** = Incident heat flux either constant or as a file of fluxes, cal/cm²·sec
- **BL** = Skin thickness, 2200µm (The last 200µm is considered to be fat)
- **AK** = Calculation interval, nominally .01 sec. For short exposures, the calculation interval must be at least one hundred times less than the exposure duration.
- **JINC** = Number of nodes, nominally 12
- **TEMPB** = Differences between TEMPI0 and backwall (fat/core) temperature, °C. Note: TEMPI0 + TEMPB = core temperature
- **Absorb** = Absorptivity usually 0.613 assuming 10 percent convective, 90 percent radiative heating, and 5 percent of radiation intercepted by hair
- **Boil** = Temperature when water boiling occurs, 100.15 °C
- **ETIME** = Exposure time, seconds
- **ITIME** = Maximum calculation time, usually 80-100 seconds
- **Nxtra** = Number of extra nodes between the surface and node #2 at 200µm, initially set at seven, used for superficial burns
  Note: The seventh node must be at 175µm for an accurate time to pain prediction.
- **Blood** = Factor to adjust amount of convective cooling by blood usually set at 0.001
DE1 & DE2 = ΔE/R from Arrhenius relationship for tissue temperatures from 44°C to 50°C, or over 50°C, respectively

PL1, PLN1, or PL2 and PLN2 => log P = logN + ylog10 = PL + PLN
for tissue temperatures from 44°C to 50°C, or over 50°C, respectively

Damage Rate Constants: PL1, PLN1, PL2, PLN2, DE1, DE2 (for Nodes 2-12)
APL1, APLN1, APL2, APLN2, ADE1, ADE2 (for Nodes 1 and Xtra Nodes)

Cp(J) = Heat capacity as a function of depth, (J)
BK(J) = Thermal conductivity as a function of depth, (J)
PCWATER = Percent water at a skin depth of 10μm at 60 percent relative humidity
WATER(J) = Percent water at each node based on 60 percent relative humidity

OUTPUTS

Flux (I) - tabulated heat flux as a function of time
DAMAGE, W, at each depth (Node)
Maximum temperature
Threshold depth in μm (microns)
Final time - total calculation time
Time to pain

File of calculated temperatures for later plotting by HARVARD GRAPHICS

File summarizing simulation
File of temperature as printed each second on the terminal

From the relationship for first order kinetics assumed to apply in damaging tissue protein we have:

\[
damage \ rate = \frac{dQ}{dt} = Pe^{-ΔE/kT};
\]
\[
\text{total damage} = \int_0^{\text{ETIME}} \frac{dn}{dt} \, dt + \int_{\text{ETIME}}^\infty \frac{dn}{dt} \, dt
\]  

if \( P = N \times 10^7 \) and \( \Delta E/R = DE \)

then:

\[
\ln \frac{dw}{dt} = \ln N + y \ln 10 - \frac{\Delta E}{R} \cdot \frac{1}{T} = PL + PLN - DE \cdot \frac{1}{(T+273)}
\]

Thus for damage calculations the following constants are entered:

\[
\begin{align*}
PL_1 \ (44^\circ C - 50^\circ C) &= 1.46 & PL_2 \ (50^\circ C - 100^\circ C) &= 2.24 \\
PLN_1 \ (44^\circ C - 50^\circ C) &= 147.37 & PLN_2 \ (50^\circ C - 100^\circ C) &= 239.47 \\
DE_1 \ (44^\circ C - 50^\circ C) &= 50,000 & DE_2 \ (50^\circ C - 100^\circ C) &= 80,000
\end{align*}
\]

The program outputs \( \frac{dn}{dt} \), for each node at each time step, total is damage for each node and a threshold depth, where \( N = 1 \). This depth, found using inverse interpolation on two or three \( N \)s nearest 1 using either \( y \) or \( \log(y) \).

Since the first presentations (Knox, Wachtel, and Knapp, 1978a, 1978c) BURNSIM has under gone further development.

Thermal properties of skin

Measurements of the water content of pig skin as a function of thickness were made on split thickness skin samples from several pigs.

Give a table of measured values of water content as a function of skin thickness, a least-square cubic polynomial was fit to the data and water content as a function of depth was computed from the formula:

\[
W(T\text{-}d) = \frac{T}{d} (W_T - W_{T\text{-}d}) + W_{T\text{-}d}
\]

where \( T \) is the total thickness of a slab, \( W_T \) is the fraction of water computed from the cubic equation, \( d \) is the thickness of a thin slab at a depth \( T\text{-}d \), and \( W_{T\text{-}d} \) is the fraction of water above the thin slab.

Thermal properties of the tissue were computed from the equations (Cooper and Trezek, 1971):

\[
1) \text{density: } \gamma = \left[ \frac{W_v}{\gamma_v} + \frac{W_f}{\gamma_f} + \frac{W_p}{\gamma_p} \right]^{-1}
\]
2) heat capacity: \[ C_p = W_w C_{p_w} + W_f C_{p_f} + W_p C_{p_p} \]  

3) thermal conductivity: \[ K = \gamma \left[ \frac{k_w W_w}{\gamma_w} + \frac{k_f W_f}{\gamma_f} + \frac{k_p W_p}{\gamma_p} \right] \]

where the subscripts \( w, f, \) and \( p \) refer to water, fat, and protein, respectively. \( W_w \) is the mass fraction, \( \gamma_w \) the density, \( C_{p_w} \) the heat capacity, and \( k_w \) the thermal conductivity of the respective components. Values of the various terms used were:

- \( \gamma_w = 1 \) gm/cm\(^3\)  
  \( C_{p_w} = 1 \) cal/gm\(^{\circ}\)C  
  \( k_w = 1.5 \times 10^3 \) cal/cm-sec\(^{\circ}\)C
- \( \gamma_f = 0.815 \) gm/cm\(^3\)  
  \( C_{p_f} = 0.55 \) cal/gm\(^{\circ}\)C  
  \( k_f = 4.5 \times 10^4 \) cal/cm-sec\(^{\circ}\)C
- \( \gamma_p = 1.54 \) gm/cm\(^3\)  
  \( C_{p_p} = 0.26 \) cal/gm\(^{\circ}\)C  
  \( k_p = 4.3 \times 10^4 \) cal/cm-sec\(^{\circ}\)C

Fat and protein were assumed to be present in equal amounts so that:

\[ W_f = W_p = (1-W_w)/2, \]  

and the resultant equations were:

\[ \gamma = (6.18277 \times 10^3 W_w + .938172)^{1/3} \]  

\[ K = \gamma (1.08432 \times 10^4 W_w + 4.15684 \times 10^4) \]  

\[ C_p = .595 W_w + .405 \]

Using the equations above, the profile of thermal properties was calculated for skin depths of from 80 to 2000\( \mu \)m. A linear extrapolation of tissue water content from a depth of 80\( \mu \)m to the skin surface was made using a stratum corneum water content calculated from Rushmer et al. (1966) and the ambient percent humidity during the experimental phase of the project. This calculated water profile was used to complete the calculation of thermal properties profile from 80\( \mu \)m to the skin surface. The thermal properties of the skin at 2200\( \mu \)m were assumed to be those of fat. These new thermal properties replaced those chosen by Morse et al. (1973) and used during previously reported simulations (Knox, Wachtel, and Knapp, 1978a, 1978c). See the paper entitled "Thermal properties calculated from measured water content as a function of depth in porcine skin" (Knox et al., 1986).

Intraskin temperatures

In earlier simulations (Knox et al., 1978a, 1978c) it became apparent that unless the temperature calculations reasonably represented what actually occurred in the skin, adjustment of the values for PL, PLN and DE in the damage equation to match a few data points would not be likely to result in a model which works well for all cases. Fortunately
11 intraskin temperature profiles were recorded on FM magnetic tape. These voltage records were digitized and converted to tables of temperatures at 100 samples per second. Figure A-2 presents the one page summary report from a simulation of the exposure of Pig 294RF to a 3.47 cal/cm²-sec fire for 3.02 seconds. Note that boiling occurred (confirmed by blister formation, Figure A-3) and that the surface reached a maximum of 128.724°C. Predicted threshold depth was 1520µm. Three observed temperature profiles are overlayed on the calculated temperature profiles (for nodal depths of 0, 200, 400,...2200µm) in Figures A-4, A-5, and A-6. The oscillations in the observed temperature profile are most probably due to a "hunting" in the autoregulation of tissue perfusion by blood. The frequency, for example, is similar to that seen in studies of microcirculation.
MODEL NAME OR DESCRIPTION: PIG 294RF ABS 0.613

SKIN DIFFUSION DATA
INPUT PARAMETER LIST

TEMPIO= 34.9700  DENS= 1.00000  Q1= 3.47000
BL= .220000  AK= .100000E-01  JINC= 12
TEMPB= 3.3600  ABSORB= .613000  BOIL= 100.150

APL1= .780000  APLN1= 285.520  ADE1= 93534.9
APL2= .600000  APLN2= 117.430  ADE2= 39109.8

PL1= 1.46  PLN1= 147.37  DE1= 50000.00
PL2= 2.24  PLN2= 239.47  DE2= 80000.00

ETIME= 3.02  ITIME= 80.00  NXTRA= 8

EXTRA NODES: 22.2 44.4 66.7 88.9 111.1 133.3 155.6 177.8

FLUX FILE I.D.: .00 2

FLUX(I)=
1 3.470 2 3.470

W= .39950E+01
W= .40733E+00
W= .45290E-01

D= .72442E+01
D= .73778E+01
D= .74955E+01

W = .19755E+19  AT DEPTH (IN MICRONS)= .112535E-06
W = .82482E+12  AT DEPTH (IN MICRONS)= 200.000
W = .26532E+09  AT DEPTH (IN MICRONS)= 400.000
W = .57713E+06  AT DEPTH (IN MICRONS)= 600.000
W = .84775E+04  AT DEPTH (IN MICRONS)= 800.000
W = .44473E+03  AT DEPTH (IN MICRONS)= 1000.00
W = .39319E+02  AT DEPTH (IN MICRONS)= 1200.00
W = .39950E+01  AT DEPTH (IN MICRONS)= 1400.00
W = .40733E+00  AT DEPTH (IN MICRONS)= 1600.00
W = .45290E-01  AT DEPTH (IN MICRONS)= 1800.00
W = .89902E-02  AT DEPTH (IN MICRONS)= 2000.00
W = .00000E+00  AT DEPTH (IN MICRONS)= 2200.00

MAXIMUM TEMPERATURE = 128.724

THRESHOLD DEPTH = 1528.

FINAL TIME = 80.00

Figure A-2. Summary report for simulation of Pig 294RF to a 3.47 cal/cm²-sec fire for 3.02 seconds.
Figure A-3. Intraskin thermocouple (0.003", "located superficially") shown prior to burn (left) and subsequent to exposure to 3.47 cal·cm$^2$·sec$^{-1}$ for 3.02 seconds (right).

Gross grade = 13  
New micro grade = 8  
Threshold depth = 1465 μm
Figure A-4. Predicted skin temperature at each node (solid lines) and measured intraskin temperature in pig 284 location #27 when exposed to 3.47 cal.cm⁻².sec⁻¹ for 3.02 seconds
Figure A-5. Predicted skin temperature at each node (solid lines) and measured intraskin temperature in pig 284 location #28 when exposed to 3.47 cal·cm$^{-2}$·sec$^{-1}$ for 3.02 seconds
Figure A-6. Predicated skin temperature at each node (solid lines) and measured intraskin temperature in pig 284 location #29 when exposed to 3.47 cal·cm\(^{-2}\)·sec\(^{-1}\) for 3.02 seconds.
APPENDIX B

Last Edited March 5, 1992

******************************************************************
12-POINT BURN PREDICTION MODEL***********************

PROGRAM BURNSIM

BURN PREDICTION MODEL WITH WATER BOILING
AND USE OF EITHER CONSTANT OR TABULATED FLUX
AND VARIABLE COOLING BY BLOOD FROM NODES 2
AND 3 BEGINNING AT .01 SEC AND LINEARLY
INCREASING TO 20 SEC AND THEN REMAINING
CONSTANT......

CHANGED TO DO INTEGRATION OF DAMAGE W & XW
WITHIN PROGRAM AND NOT OUT TO DISK AND BACK

CHANGED TO INCORPORATE THE CHANGES IN MODEL
7 NAMELY DIFFERENT RATE CONSTANTS ETC FOR
SUPERFICIAL NODES AND VARIABLE AK IN BLUD

THIS MODEL WAS DEVELOPED UNDER CONTRACT FOR THE U.S. ARMY
MEDICAL RESEARCH AND DEVELOPMENT COMMAND, AND THE U.S. ARMY
AEROMEDICAL RESEARCH LABORATORY, FORT RUCKER AL. 36362,
STANLEY C. KNAPP, COL, MC, COMMANDING, BY FRANCIS S. KNOX, III,
PH.D. WITH THE ASSISTANCE OF DANIEL D. RENEAU, PH.D., NELSON
O'YOUNG, AND CHET ELLIS, M.S.

ADDITIONAL DEVELOPMENT CONDUCTED UNDER ILIR FUNDING AT USAARL
AND ON OWN TIME BY FRANCIS S. KNOX, III, PH.D.

QUESTIONS AND COMMENTS SHOULD BE ADDRESSED TO:
FRANCIS S. KNOX, III, PH.D
CHIEF, ESCAPE AND IMPACT PROTECTION BRANCH
BIODYNAMICS AND BIOCOMMUNICATIONS DIVISION
ARMSTRONG LABORATORY
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433
COM. 513-255-3931 AV 785-3931

This copy current working copy of Dr. Knox. Other copies
are not to be further distributed without permission from
Dr. Knox.

INCLUDE 'FGRAPH.FI'
INCLUDE 'FGRAPH.FD'
REAL*4 ITIME,NOFIL,TP,SUM(13),DW(13)
INTEGER CHANGE,PTS,AGAIN,PROCED
INTEGER*4 DUMMY4
INTEGER*2 DUMMY2
DIMENSION T(12),F(12),G(12),H(12),W(12),Z(12),SV(12),U(12)
DIMENSION CP(12,2),BK(12,2),D(12),DSCRPT(20)
DIMENSION ID(4),FLUX(600),Q(12)

46
DIMENSION XTIME(12),ZTIME(12),IFLAG(12),JFLAG(12)
DIMENSION WATER(12,2),ROCON(2),THCON(2),CPCON(2)
DIMENSION XW(8),XTRA(8),XTRALG(8),XTMP(8),XDW(8),XSUM(8)
CHARACTER RESP*1,FILNAM*8,SUMFILE*8,PROFILE*8,TFILE*8
EQUIVALENCE (NOFIL,IBLNK)
  DATA NOFIL/ ' ' /
  DATA MAXDIM/12/,D2/200. /
  DATA THCON/1.084316E-3,4.1568401E-4/
  DATA ROCON/6.18277433-2,0.93817226/
  DATA CPCON/0.595,0.405/

LOGICAL UNIT 1 INPUT : 'REN12.DAT'; INITIAL VALUES OF PARAMETERS
LOGICAL UNIT 1 SCRATCH: VALUES OF XW (IF COMPUTED)
LOGICAL UNIT 2 SCRATCH: VALUES OF W (COMPUTED)
LOGICAL UNIT 3 OUTPUT : PROFILE; TEMPERATURE PROFILES
LOGICAL UNIT 4 INPUT : FILNAM; NAME OF FLUX FILE
LOGICAL UNIT 4 OUTPUT : TFILE; DATA FOR PLOTTING TEMPERATURE PROFILES
LOGICAL UNIT 7 OUTPUT : SUMMARY PRINTOUT

C*****Introduction to BURNSIM
CALL COLORS
  DUMMY4=SETBKCOLOR( $BLUE )
CALL WELCOME(PROFILE)

C*****Read REN12.DAT input file
CALL READDATA(TEMPIO,DENS,FLUX,BL,AK,BOIL,ABSORB,JINC,TEMPB,
  + ITIME,ETIME,PCWATR,BL2,CP,BK,PL2,PLN2,PL1,PLN1,DE2,DE1,
  + APL1,APLNL2,APL2,APLNV2,APL1,DE1,DE2,WATER)
OPEN(UNIT=1,FORM='UNFORMATTED',STATUS='SCRATCH')
OPEN(UNIT=2,FORM='UNFORMATTED',STATUS='SCRATCH')
OPEN(UNIT=3,FILE=PROFILE,FORM='FORMATTED',STATUS='UNKNOWN')
FLUX(1) = QO
FLUX(2) = QO
NFLX = 2
FILNAM = NOFIL
PPL1 = PL1
PPLN1 = PLN1
DEE1 = DE1
APPL1 = APL1
APPLN1 = APLN1
ADDE1 = DE1
NXTRA = 0
NXTRA0 = NXTRA

C*****Display input values on screen
CALL SHOWVALUE(TEMPIO,DENS,FLUX,BL,AK,JOINC,TEMPB,
  + ABSORB,BOIL,PL1,PLN1,DE1,PL2,PLN2,DE2,ETIME,ITIME,NXTRA,
  + BLOOD,APL1,APLNL2,APL2,APLNV2,APL1,DE1,APL2,APLV,DE2,K,XTRA)
DO WHILE (AGAIN .EQ. 0)
  CALL PROCEED(RESP,PROCED,AGAIN)
  IF(PROCED.EQ.0) THEN
    PTS=1
    TIME=0.
CHANGE=0
MN=0
CALL clearscreen( $GCLEARSCREEN )
WRITE(*,10)
10 FORMAT(/,15X,'TYPE THE NUMBER OF THE FUNCTION BELOW THAT YOU',/ + ,15X,'WISH TO PERFORM.',///,20X,'CHOOSE A FUNCTION NUMBER:',///, + 25X,'1 - CHANGE SELECTED INITIAL VALUES',/25X,'2 - NO CHANGES', + '--CONTINUE RUNNING THE PROGRAM',//,25X,'3 - EXIT',///,20X,'PLEASE' + ' ENTER THE FUNCTION NUMBER: $)
READ(*,20)IANSR
20 FORMAT(I2)
CALL clearscreen( $GCLEARSCREEN )
IF(IANSR.EQ.1) THEN
  DO WHILE(CHANGE.EQ.0)
    CALL SHOWVALUE(TEMP10,DENS,FLUX,BL,AK,JINC,TEMPB, + ABSORB,BOIL,PL1,PLN1,DE1,PL2,PLN2,ETIME,ITIME,NXTRA, + BLOOD,APL1,APLN1,ADE1,APL2,APLN2,ADE2,K,XTRA)
    WRITE(*,30)
  30 FORMAT(/,15X,'DO YOU WANT TO MAKE ANY CHANGES? TYPE Y/N $)
    READ(*,40)RESP
  40 FORMAT(A1)
    IF(RESP.EQ.'Y'.OR.RESP.EQ.'y') THEN
      CALL CLEARESCREEN( $GCLEARSCREEN )
      WRITE(*,50)
      50 FORMAT(/,15X,'DO YOU WANT TO MAKE ANY CHANGES? TYPE Y/N $)
      READ(*,30)RESP
      IF(INUM.EQ.1) THEN
        WRITE(*,60)TEMP10
        60 FORMAT(/,9X,'THE VALUE FOR TEMPIO IS: ',F10.5,' INPUT NEW VALU' + 'E: $)
        READ(*,70)TEMP10
      70 FORMAT(G10.5)
      ELSEIF(INUM.EQ.2) THEN
        WRITE(*,80)DENS
        80 FORMAT(/,9X,'THE VALUE FOR DENS IS: ',F10.5,' INPUT NEW VALUE: ' + 'E: $)
        READ(*,90)DENS
      ELSEIF(INUM.EQ.3) THEN
        WRITE(*,90)
        90 FORMAT(/,9X,'ENTER THE FLUX FILE NAME (TYPE A <CR> IF NO FILE ' + ,9X,'IS TO BE USED): $)
        READ(*,100)FILNAM
      ELSEIF(INUM.EQ.4) THEN
        WRITE(*,100)FILNAM
        ELSE
          WRITE(*,110)
          110 FORMAT(A1)
          READ(*,120)
      END IF
  END IF
IF (FILNAM.NE.NOFIL) THEN
WRITE(*,110)
110 FORMAT(/,9X,'ENTER FLUX ID (UP TO 8 CHARACTERS): ',$)
READ(*,120)
120 FORMAT(4A2)
WRITE(*,130)
130 FORMAT(/,9X,'ENTER THE NUMBER OF POINTS IN FLUX FILE: ',$)
READ(*,*) NFLX
DO WHILE (NFLX.GT.600)
WRITE(*,140)
140 FORMAT(/,9X,'THE FLUX FILE MUST CONTAIN NO MORE THAN 600 DATA'
  + ',/,9X,' POINTS. REENTER A NUMBER LESS THAN OR EQUAL TO 600. ',$)
READ(*,*) NFLX
END DO
WRITE(*,150)
150 FORMAT(/,9X,'ENTER THE SAMPLE INTERVAL IN SECONDS: ',$)
READ(*,*) TDELT
OPEN(UNIT=4,FILE=FILNAM,FORM='FORMATTED',STATUS=
  + 'OLD')
READ(4,*)(FLUX(I),I=1,NFLX)
CLOSE (4)
IF(NFLX.LE.0) STOP 'ERROR----TOO FEW FLUX POINTS.'
WRITE(*,430)ID,TDELT,NFLX
ELSE
WRITE(*,160)FLUX(1)
160 FORMAT('/,9X,'CONSTANT Q-VALUE = ',F10.5,' INPUT NEW VALUE: '
  + $)
READ(*,70)FLUX(1)
FLUX(2)=FLUX(1)
NFLX = 2
DO I=1,4
  ID(I) = IBLNK
END DO
END IF
ELSEIF(INUM.EQ.4) THEN
WRITE(*,170)BL
170 FORMAT('/,9X,'THE VALUE FOR BL IS: ',F10.5,' INPUT NEW VALUE: '
  + $)
READ(*,70)BL
ELSEIF(INUM.EQ.5) THEN
WRITE(*,180)AK
180 FORMAT('/,9X,'THE VALUE FOR AK IS: ',F10.5,' INPUT NEW VALUE: '
  + $)
READ(*,70)AK
ELSEIF(INUM.EQ.6) THEN
WRITE(*,190)JINC
190 FORMAT('/,9X,'THE VALUE FOR JINC IS: ',I30,' INPUT NEW VALUE: '
  + $)
READ(*,20)JINC
IF (JINC.GT.MAXDIM) JINC=MAXDIM
ELSEIF(INUM.EQ.7) THEN
WRITE(*,200)TEMPB
200 FORMAT ('THE VALUE FOR TEMP IS: ',F10.5, ' INPUT NEW VALUE')
READ(*,70)TEMP
ELSEIF(INUM.EQ.8) THEN
WRITE(*,210)ETIME
210 FORMAT('THE VALUE FOR ETIME IS: ',F10.5, ' INPUT NEW VALUE')
READ(*,70)ETIME
ELSEIF(INUM.EQ.9) THEN
WRITE(*,220)PL1
220 FORMAT('THE VALUE FOR PL1 IS: ',F10.5, ' INPUT NEW VALUE')
READ(*,70)PL1
PPL1=PL1
ELSEIF(INUM.EQ.10) THEN
WRITE(*,230)PLN1
230 FORMAT('THE VALUE FOR PLN1 IS: ',F10.5, ' INPUT NEW VALUE')
READ(*,70)PLN1
PPLN1=PLN1
ELSEIF(INUM.EQ.11) THEN
WRITE(*,240)PL2
240 FORMAT('THE VALUE FOR PL2 IS: ',F10.5, ' INPUT NEW VALUE')
READ(*,70)PL2
ELSEIF(INUM.EQ.12) THEN
WRITE(*,250)PLN2
250 FORMAT('THE VALUE FOR PLN2 IS: ',F10.5, ' INPUT NEW VALUE')
READ(*,70)PLN2
ELSEIF(INUM.EQ.13) THEN
WRITE(*,260)DE1
260 FORMAT('THE VALUE FOR DE1 IS: ',F10.1, ' INPUT NEW VALUE')
READ(*,70)DE1
DDE1=DE1
ELSEIF(INUM.EQ.14) THEN
WRITE(*,270)DE2
270 FORMAT('THE VALUE FOR DE2 IS: ',F10.1, ' INPUT NEW VALUE')
READ(*,70)DE2
ELSEIF(INUM.EQ.15) THEN
WRITE(*,280)ITIME
280 FORMAT('THE VALUE FOR ITIME IS: ',F10.5, ' INPUT NEW VALUE')
READ(*,70)ITIME
ELSEIF(INUM.EQ.16) THEN
WRITE(*,290)ABSORB
290 FORMAT('THE VALUE FOR ABSORB IS: ',F10.5, ' INPUT NEW VALUE')
READ(*,70)ABSORB
ELSEIF(INUM.EQ.17) THEN

50
WRITE(*,300) BOIL

300 FORMAT(//,9X,'THE VALUE FOR BOIL IS: ',F10.5, ' INPUT NEW VALUE'
+ 'E: ')$
READ(*,70) BOIL
ELSEIF(INUM.EQ.18) THEN
NXTRA0 = NXTRA
WRITE(*,310) NXTRA

310 FORMAT(//,9X,'THE NUMBER OF EXTRA NODES IS: ',I4, ' INPUT NEW'
+ ' VALUE: ')$
READ(*,20) NXTRA
IF (NXTRA.NE.0) THEN
  IF (NXTRA.GT.8) NXTRA = 8
  IF (NXTRA0.NE.0) WRITE(*,320) (XTRA(I),I=1,NXTRA0)

320 FORMAT(//,9X,'CURRENT EXTRA NODES: ',8F6.1)
WRITE(*,330)

330 FORMAT(//,9X,'ENTER NEW VALUES SEPARATED BY COMMAS, OR A <CR>
+ ',/9X,'IF THE PROGRAM IS TO CALCULATE VALUES. ')$
READ(*,340) XTRA

340 FORMAT(866.1)

IF (XTRA(1).LE.0) THEN
  C***** Numerator in next statement is specific for n-point model
  DXTRA = D2/(NXTRA+1)
  DO I=1,NXTRA
     XTRA(I) = DXTRA*I
  END DO
  END IF
  DO I=1,NXTRA
    XTRA(I) = ALOG(XTRA(I))
  END DO
  END IF
  NXTRAO = NXTRA
ELSEIF(INUM.EQ.19) THEN
WRITE(*,350) BLOOD

350 FORMAT(//,9X,'THE VALUE FOR BLOOD IS: ',F10.5, ' INPUT NEW VALU'
+ 'E: ')$
READ(*,70) BLOOD
ELSEIF(INUM.EQ.20) THEN
WRITE(*,360) APL1

360 FORMAT(//,9X,'THE VALUE FOR APL1 IS: ',F10.5, ' INPUT NEW VALUE'
+ 'E: ')$
READ(*,70) APL1
ELSEIF(INUM.EQ.21) THEN
WRITE(*,370) APLN1

370 FORMAT(//,9X,'THE VALUE FOR APLN1 IS: ',F10.5, ' INPUT NEW VALU'
+ 'E: ')$
READ(*,70) APLN1
ELSEIF(INUM.EQ.22) THEN
WRITE(*,380) APL2

380 FORMAT(//,9X,'THE VALUE FOR APL2 IS: ',F10.5, ' INPUT NEW VALUE'
+ 'E: ')$
READ(*,70) APL2
ELSEIF(INUM.EQ.23) THEN
WRITE(*,390)APLN2
390 FORMAT(//,9X,'THE VALUE FOR APLN2 IS: ',F10.5,' INPUT NEW VALU'
      + 'E: '$)
      READ(*,70)APLN2
ELSEIF(INUM.EQ.24) THEN
WRITE(*,400)ADE1
400 FORMAT(//,9X,'THE VALUE FOR ADE1 IS: ',F10.1,' INPUT NEW VALUE'
      + ': '$)
      READ(*,70)ADE1
ELSEIF(INUM.EQ.25) THEN
WRITE(*,410)ADE2
410 FORMAT(//,9X,'THE VALUE FOR ADE2 IS: ',F10.1,' INPUT NEW VALUE'
      + ': '$)
      READ(*,70)ADE2
ELSE
CHANGE=1
ENDIF
ELSE
CHANGE=1
ENDIF
END DO
REWIND 1
ELSEIF(IANSR.EQ.2) THEN
CALL CLEARSCREEN(SGCLEARSCREEN)
CALL DESCRIPT(DSCRPT,SUMFILE,TFILE)
TP=999.
AJ=JINC
Q1 = FLUX(1)
H1=BL/(AJ-1.0)
C*******Initialize depth nodes D(J)
D(1) = -16.
DO I=2,JINC
D(I) = H1*(I-1)*1.E4
D(I) = ALOG(D(I))
END DO
DTJ = TEMPB/(JINC-1)
DO J=1,JINC
WATER(J,1) = WATER(J,2)
CP(J,1) = CP(J,2)
BK(J,1) = BK(J,2)
XTIME(J) =0.
ZTIME(J) =0.
IFLAG(J) =0.
JFLAG(J) =0.
T(J) = DTJ*(J-1)+TEMPIO
END DO
WRITE(4,420)TIME,T(1),T(2),T(3),T(4),T(5),T(6),T(7),T(8)
+ ,T(9),T(10),T(11),T(12)
420 FORMAT(13(F9.5,2X))
K=1
CALL SHOWVALUE(TEMPIO,DENS,FLUX,BL,AK,JINC,TEMPB,ABSORB,
+ BOIL,PL1,PLN1,DE1,PL2,PLN2,DE2,ETIME,ITIME,NXTRA,BLOOD,APL1,
\begin{verbatim}
+ APLN1, ADE1, APL2, APLN2, ADE2, K, XTRA)
WRITE(3, 430) ID, TDELT, NFLX, (I, FLUX(I), I=1, NFLX)
WRITE(7, 430) ID, TDELT, NFLX, (I, FLUX(I), I=1, NFLX)
430 FORMAT(/, SX, 'FLUX FILE I.D.: ', 4A2, F7.2, 14//' FLUX(I)='
+ (' ', L0(I5, F8.3)))
JJJJ=0
F(1)=-BK(2,1)/(2.0*H1*H1)-BK(1,1)/(2.0*H1*H1)
G(1) = (BK(1,1)+BK(2,1))/(2.0*H1*H1)+DENS*CP(1,1)/AK
H(1)=0.0
ITTR = 0
IFLX = 1
EITIM1 = ITIME+1.
IF (FILNAM.EQ.NOFIL) TDELT=AK
FFDG = TDELT/AK
KFDG = FFDG+.0001
TMPMAX = 0.
QCONST = ABSORB*60.892
BLUD = 0.
M = -1
TIME = 0.
ITFLG = 0
CALL SUBI2(TIME, T, XTIME, JINC, BLUD, CP, BK, NXTRA, XTMP, M, EM)
DOWHILE(TIME.LT.ITIME.AND. ITFLG.EQ.0.OR. TIME.LT.ETIME)
C*****The next program statement automatically chooses the proper
C   interval in the flux table for the computation of QT and Q1 for
C   either constant or variable flux.
C
   KFDG (=FFDG) = 1 for constant flux
C   = integer ratio of the tabular time step to
C   to model the time step for the tabulated flux
   IF (MOD(ITTR,KFDG).EQ.0.AND. IFLX.LT.NFLX) IFLX=IFLX+1
ITTR = ITTR+1
P = (ITTR-KFDG*(IFLX-2))/FFDG
QT = (1.-P)*FLUX(IFLX-1)+P*FLUX(IFLX)
Q1* = QT*QCONST
JJJJ = JJJJ+1
TIME=JJJJ*AK
IF (TIME.GE..01.AND. TIME.LE.20.) BLUD=(TIME-AK)/(20.-AK)*
+ BLOOD
IF (TIME.GE.ETIME) Q1=-5.E-4*(T(1)-23.9)
    Z(1)=-F(1)*T(2)-((BK(1,1)+BK(2,1))/(2.0*H1*H1)-(DENS*
+ CP(1,1))/AK)*T(1)+Q1
N=JINC-1
DO J=2,N
    F(J)=-BK(J+1,1)/(2.0*H1*H1)
    G(J)=(BK(J,1)+BK(J+1,1))/(2.0*H1*H1)+DENS*CP(J,1)/AK
    H(J)=BK(J,1)/(2.0*H1*H1)
    Z(J)=-F(J)*T(J-1)-((BK(J,1)+BK(J+1,1))/(2.*H1*H1)-DENS*
+ CP(J,1)/AK)*T(J)-H(J)*T(J-1)
    IF (J.LE.3) Z(J) = Z(J)-1.675*H1/BK(J,1)*BLUD*(T(J)
+ TEMPI0+TEMPB)
END DO
F(JINC)=0.0
\end{verbatim}
\[ G(JINC) = \frac{(BK(JINC,1) + BK(JINC-1,1))}{(2.0 \times H1 \times H1)} + DENS \times CP(JINC,1) / AK \]

\[ H(JINC) = -\frac{(BK(JINC,1) + BK(JINC-1,1))}{(2.0 \times H1 \times H1)} \]

\[ DT = T(JINC-1) - (TEMP0 + TEMPB) \]

\[ Z(JINC) = (H(JINC) + (DENS \times CP(JINC,1) / AK) \times T(JINC) - H(JINC)) \]

\[ W(1) = G(1) \]

\[ U(1) = \frac{Z(1)}{W(1)} \]

DO J = 2, JINC

\[ JM1 = J - 1 \]

\[ SV(JM1) = F(JM1) / W(JM1) \]

\[ W(J) = G(J) - H(J) \times SV(JM1) \]

\[ U(J) = \frac{(Z(J) - H(J) \times U(JM1))}{W(J)} \]

END DO

\[ T(JINC) = U(JINC) \]

\[ KK = JINC - 1 \]

DO J = 1, KK

\[ KMJ = JINC - J \]

IF (IFLAG(KMJ).NE.1) THEN

\[ T(KMJ) = U(KMJ) - SV(KMJ) \times T(KMJ+1) \]

IF (JFLAG(KMJ).NE.1) THEN

IF (T(KMJ).GE.BOIL) THEN

\[ Q(KMJ) = \frac{BK(KMJ,1) \times (T(KMJ) - T(KMJ+1))}{H1} \]

ELSEIF (K MJ.EQ.1) THEN

\[ Q(KMJ) = QT \]

END IF

\[ XTIME(KMJ) = 539. \times H1 \times WATER(KMJ,1) \]

\[ ZTIME(KMJ) = XTIME(KMJ) + TIME \]

\[ IFLAG(KMJ) = 1 \]

ENDIF

ELSEIF (IFLAG(KMJ).EQ.1) THEN

IF (TIME.GE.ZTIME(KMJ)) THEN

\[ WATER(KMJ,1) = PCWATR \]

\[ CP(KMJ,1) = (CPCON(1) \times WATER(KMJ,1) + CPCON(2)) / (ROCON(1) + WATER(KMJ,1) + ROCON(2)) \]

\[ BK(KMJ,1) = THCON(1) \times WATER(KMJ,1) + THCON(2) / (ROCON(1) + WATER(KMJ,1) + ROCON(2)) \]

\[ IFLAG(KMJ) = 0 \]

\[ XTIME(KMJ) = 0. \]

\[ JFLAG(KMJ) = 1 \]

ENDIF

ENDIF

END DO

C******* Interpolate extra temperatures between the surface and second node

IF (NXTRA.NE.0) THEN

IF (T(2).EQ.T(1)) THEN

C******* For constant temperature

DO I = 1, NXTRA

\[ XTMP(I) = T(2) \]

ENDIF

END IF

END
ELSEIF (T(2).EQ.T(3)) THEN

C*******Linear interpolation
DO I=1,NXTRA
  P = XTRA(I)/D2
  XTMP(I) = (1.-P)*T(1)+P*T(2)
END DO
ELSE

C*******3-point Lagrange interpolation for equally spaced abscissae
DO I=1,NXTRA
  P = (XTRA(I)-D2)/D2
  XTMP(I) = .5*P*(P-1.)*T(1)+(1.-P**2)*T(2)+.5*P*(P+1.)
END DO
END IF
ENDIF
IF (ABS(ETIME-TIME) .LE. 0.5*AK) THEN
DO I=1,JINC
  IF (IFLAG(I).NE.0) THEN
    W = (ZTIME(I)-TIME)/XTIME(I)*(WATER(I,l)-
        (l)*WATER(I,l)+CPCON(2))/(ROCON(l)*
        (l)*WATER(I,l)+THCON(2))/(ROCON(l)*
        PCWATR)+PCWATR
    CP(I,l) = (CPCON
    BK(I,l) = (THCON
    END IF
END DO
DO I=1,JINC
  XTIME(I) = 0.
  IFLAG(I) = 0
  JFLAG(I) =1
END DO
ENDIF
IF (T(l).GT.TMPMAX) TMPMAX=T(l)
ITFLG = -1 !ITFLG SET TO 0 IF ANY TEMPERATURE .GE. 44 DEGREES
DO J=1,JINC
  IF (T(J).LT.44.) THEN
    DW(J) = 0.
  ELSE
    ITFLG = 0
    IF(T(J).LT.50.) THEN
      PL1 = PPL1
      PLN1 = PPLN1
      DE1 = DDE1
      AP1 = APPL1
      APLN1 = APPLN1
      ADE1 = ADDE1
      DWLN=PL1+PLN1-DE1/(T(J)+273.)
      IF(DWLN.GE.87.0) DWLN = 87.0
      DW(J) = EXP(DWLN)
    ELSE
      PL1=PL2
  ELSE
    ITFLG = 0
    IF(T(J).LT.50.) THEN
      PL1 = PPL1
      PLN1 = PPLN1
      DE1 = DDE1
      AP1 = APPL1
      APLN1 = APPLN1
      ADE1 = ADDE1
      DWLN=PL1+PLN1-DE1/(T(J)+273.)
      IF(DWLN.GE.87.0) DWLN = 87.0
      DW(J) = EXP(DWLN)
    ELSE
      PL1=PL2
  END IF
  END IF
END
PLN1=PLN2
DE1=DE2
APL1 = APL2
APLN1 = APLN2
ADE1 = ADE2
IF(J.LE.1) THEN
   DWLN = APL1 + APLN1- ADE1/(T(J)+273.)
   IF(DWLN.GE.87.0) DWLN = 87.0
   DW(J) = EXP(DWLN)
ELSE
   DWLN=PL1+PLN1-DE1/(T(J)+273.)
   IF(DWLN.GE.87.0) DWLN = 87.0
   DW(J) = EXP(DWLN)
END IF
END IF
END IF
END DO
IF (JJJJ.LT.2) THEN
   DO I=1,JINC
      SUM(I) = .5*DW(I)
   END DO
ELSE
   DO I=1,JINC
      IF (SUM(I).LT.(1.0E38)) SUM(I)=SUM(I)+DW(I)
   END DO
END IF
ENDIF
IF (NXTRA.NE.0) THEN
   DO J=1,NXTRA
      IF (XTMP(J).LT.44.) THEN
         XDW(J) = 0.
      ELSE
         IF (XTMP(J).LT.50.) THEN
            APL1 = APPL1
            APLN1 = APPLN1
            ADE1 = ADDE1
         ELSE
            APL1 = APL2
            APLN1 = APLN2
            ADE1 = ADE2
         END IF
         IF (TP.EQ.999..AND.XTMP(4).GE.45.)
            DWLN = APL1+APLN1-ADE1/(XTMP(J)+273.)
            IF(DWLN.GE.87.0) DWLN=87.0
            XDW(J) = EXP(DWLN)
         END IF
      END IF
   END DO
ENDIF
END IF
IF (JJJJ.LT.2) THEN
   DO J=1,NXTRA
      XSUM(J) = 0.5*XDW(J)
   END DO
ELSE
   56
DO J=1,NXTRA
   IF (XSUM(J).LT.1.0E38) XSUM(J)=XSUM(J)+XDW(J)
   END DO
END IF
END IF

EMTIME = AINT(1000.*(TIME+.00001))/100.
IF (TIME.LT.10..AND.AMOD(EMTIME,1.).EQ.0..OR.TIME.GE.10. + .AND.AMOD(EMTIME,10.).EQ.0.) THEN
   WRITE(4,420)TIME,T(1),T(2),T(3),T(4),T(5),T(6),T(7), + T(8),T(9),T(10),T(11),T(12)
   PTS=PTS+1
END IF
IF (ITFLG.NE.0..AND.TIME.GE.ETIME.OR.JJJJ.EQ.M*100..OR.JJJJ + .EQ.1) CALL SUB12(TIME,T,XTIME,JINC,BLUD,CP,BK,NXTRA,XTMP,M,EM)
END DO
REWIND(4)
CLOSE(4)
DO I=1,JINC
   W(I) = (SUM(I)-.5*DW(I))*AK
END DO
IF (NXTRA.NE.0) THEN
   DO J=1,NXTRA
      XW(J) = (XSUM(J)-.5*XDW(J))*AK
   END DO
END IF
C*****Select W(J) and D(J) near W(J) =1
470  NN = 3
J=1
DO WHILE(J.LE.JINC)
   JLT1 = J
   IF(W(J).GT.1.) THEN
      IF(J.EQ.JINC) THEN
         NN=2
         WRITE(3,440)(W(K),K=JLT1-1,JLT1+1)
         WRITE(7,440)(W(K),K=JLT1-1,JLT1+1)
      END IF
      J=J+1
   ELSEIF(W(J).EQ.1.) THEN
      MN=1
      J=JINC+1
   ELSEIF(W(J).LT.1.) THEN
      IF (J.EQ.1) JLT1=2
      IF (J.EQ.JINC) JLT1=JINC-1
      WRITE(3,440)(W(K),K=JLT1-1,JLT1+1)
      WRITE(7,440)(W(K),K=JLT1-1,JLT1+1)
      WRITE(3,450)(D(K),K=JLT1-1,JLT1+1)
      WRITE(7,450)(D(K),K=JLT1-1,JLT1+1)
      IF (NXTRA.NE.0.AND.JLT1.LE.2) THEN
         57
      END IF
      J=J+1
   ELSEIF(W(J).EQ.1.) THEN
      MN=1
      J=JINC+1
   ELSEIF(W(J).LT.1.) THEN
      IF (J.EQ.1) JLT1=2
      IF (J.EQ.JINC) JLT1=JINC-1
      WRITE(3,440)(W(K),K=JLT1-1,JLT1+1)
      WRITE(7,440)(W(K),K=JLT1-1,JLT1+1)
      WRITE(3,450)(D(K),K=JLT1-1,JLT1+1)
      WRITE(7,450)(D(K),K=JLT1-1,JLT1+1)
      IF (NXTRA.NE.0.AND.JLT1.LE.2) THEN
      END IF
   END IF
END DO
WRITE(*,460)
460   FORMAT(/,9X,'W=l LIES ABOVE NODE 2. INTERCOLLATING VALUES OF D'/
+ 9X,'AND W COMPUTED FROM INTERPOLATED VALUES OF D AND',/9X,
+ 'TEMPERATURE. '/)
WRITE(3,460)
WRITE(7,460)
WRITE(1)D(1)
WRITE(2)W(1)
DO J=1,NXTRA
   WRITE(1)XTRALG(J)
   WRITE(2)XW(J)
END DO
DO J=2,JINC
   WRITE(1)D(J)
   WRITE(2)W(J)
END DO
REWIND 1
REWIND 2
DO J=1,JINC
   READ(1)D(J)
   READ(2)W(J)
END DO
REWIND 1
REWIND 2
NXTRA0 = 0
GO TO 470
END IF
J=JINC+1
END IF
END DO
IF(MN.EQ.0) THEN
   NXTRAO = NXTRA
   IF (W(JLT1+1).EQ.0..AND.NN.EQ.3) NN=2
   IF(W(JLT1-1).LT.1.0) THEN
      TD=0.0
      IERR=0
   ELSE
      CALL DEPTH(D(JLT1-1),W(JLT1-1),NN,TD,IERR)
      CALL DEPTH(D(JLT1),W(JLT1),NN,TD,IERR)
   ENDIF
   CALL DEPTH(D(JLT1-1),W(JLT1-1),NN,TD,IERR)
ENDIF
ELSE
   CALL DEPTH(D(JLT1-1),W(JLT1-1),NN,TD,IERR)
ENDIF
ENDIF
ENDIF
ENDIF
58
IF (IERR.NE.0) THEN
  WRITE(*,480)
  WRITE(3,480)
  WRITE(7,480)
480 FORMAT(9X,'ERROR IN SUBROUTINE "DEPTH". EXITING./')
  CALL ANOTHER(AGAIN)
ENDIF
IF (NN.EQ.2.AND.JLT1.EQ.JINC) THEN
  WRITE(3,490)MAXDIM
  WRITE(7,490)MAXDIM
490 FORMAT(/IX,'THE MODEL BLEW UP: DAMAGE > 1 AT NODE ',I2/)
  CALL SUB1020(W,JINC,D,TMPMAX,TD,TIME,TP)
ELSE
  CALL SUB1020(W,JINC,D,TMPMAX,TD,TIME,TP)
ENDIF
ELSE
  TD=EXP(D(J))
  CALL SUB1020(W,JINC,D,TMPMAX,TD,TIME,TP)
ENDIF
CALL HARVARD(PROFILE,TFILE,SUMFILE,PTS)
END IF
ELSEIF(PROCED.EQ.1) THEN
  IF(AGAIN.EQ.0) THEN
    CALL SHOWVALUE(TEMPIO,DENS,FLUX,BL,AK,JINC,TEMPB,
+ ABSORB,BLUL,PL1,PLN1,DE1,PL2,PLN2,DE2,ETIME,TIME,NXTRA,
+ BLOOD,APL1,APLN1,AD1,APL2,APLN2,AD2,K,XTRA)
  ENDIF
ENDIF
END DO
CLOSE(1)
CLOSE(2)
CLOSE(3)
CLOSE(4)
CLOSE(7)
CALL COLORS
DUMMY2=SETVIDEOMODE( $DEFAULTMODE )
STOP
END

SUBROUTINE COLORS
  INCLUDE 'FGRAPH.FD'
  INTEGER*2 LOOP,LOOP1,DUMMY2
  REAL RND1,RND2
  DUMMY2=SETVIDEOMODE( $RES256COLOR )
  DO LOOP1=1,10
    WRITE(*,10)
10 FORMAT(/1X,'BU~SIM' ,'B~SIM',/2X,'B~SIM')
    DUMMY2=SETCOLOR(MOD( getcolor()+1, 16)) ! Set next color
    DO loop=1,3200
      CALL RANDOM( RND1 )
      WRITE(*,59)
      C*****Set a random pixel, normalized to be on the screen
      CALL RANDOM( RND1 )
      WRITE(*,59)
  END DO
STOP
END
CALL RANDOM( RND2 )
DUMMY2=SETPIXEL( INT2( RND1*320 ),INT2( RND2*200 ) )
END DO
END DO
DUMMY2=SETVIDEOMODE( $MAXRESMODE )
END

SUBROUTINE WELCOME(PROFILE)
CHARACTER PROFILE*8
CALL CLEARSCREEN( $GCLEARSCREEN )
WRITE(*,10)
10 FORMAT(//,9X,'WELCOME TO BURNSIM. TO BEGIN RUNNING THE PROGRAM,,'
+ 'BURNSIM','/9X,'FIRST NEEDS TO KNOW THE NAME OF THE FILE THAT','
+ 'YOU WANT TO','/9X,'STORE THE OUTPUT DATA IN. THIS FILE WILL','
+ 'CONTAIN ALL OF THE','/9X,'INPUT PARAMETERS AS WELL AS THE','
+ 'OUTPUT FOR EACH ITERATION THE','/9X,'MODEL PERFORMS. THIS','
+ 'FILE CAN BE CALLED ANYTHING UP TO EIGHT','/9X,'CHARACTERS','
+ 'LONG.),'/'15X,'PLEASE ENTER A NAME FOR THE OUTPUT FILE: ')$
READ(*,20)PROFILE
20 FORMAT(A8)
C*******Set up parameters for this run
CALL CLEARSCREEN( $GCLEARSCREEN )
WRITE(*,30)
30 FORMAT(//,9X,'NEXT BURNSIM WILL SHOW YOU THE PRESENT INPUT','
+ 'PARAMETERS. ','/9X,'UNDER THE LIST OF PARAMETERS YOU WILL SEE A','
+ ',', 'QUESTION ASKING','/9X,'IF YOU WISH TO CONTINUE. IF YOU WANT','
+ 'TO EXIT THE PROGRAM AT ','/9X,' THAT POINT, TYPE N. OTHERWISE','
+ ' TYPE Y. ','/9X,' TO CONTINUE ON TO THE LIST OF INPUT','
+ 'PARAMETERS TYPE A <CR>.' )
READ(*,*)
END

SUBROUTINE READDATA(TEMPIO,DENS,QO,BL,AK,BOIL,ABSORB,JINC,
+ TEMPB,ITIME,ETIME,PCWATR,BLOOD,CP,BK,PL2,PLN2,PL1,PLN1,DE2,
+ DE1,APL1,APLN1,APL2,APLN2,ADE1,ADE2,WATER)
REAL ITIME
DIMENSION CP(l2,2),BK(l2,2),WATER(l2,2)
OPEN(UNIT=1,FILE='RENl2.DAT',FORMATTED',STATUS='OLD')
READ(1,1O)TEMPIO,DENS,QO,BL,AK,BOIL,ABSORB
10 FORMAT(7FlO.S)
READ(1,20)JINC,TEMPB,ITIME,ETIME,PCWATR,BLOOD
20 FORMAT(1110,5F10.5)
READ(1,30)(CP(J,2),J=l,JINC)
30 FORMAT(6F10.5)
READ(1,30)(BK(J,2),J=l,JINC)
READ(1,30)(PL2,PLN2,PL1,PLN1,DE2,DE1)
READ(1,30)(APL1,APLN1,APL2,APLN2,ADE1,ADE2)
READ(1,30,END=40)(WATER(I,2),I=l,JINC)
CLOSE(1)
CALL CLEARSCREEN( $GCLEARSCREEN )
END
SUBROUTINE SHOWVALUE(TEMPIO, DENS, FLUX, BL, AK, JINC, TEMPB,
+ ABSORB, BOIL, PL1, PLN1, DE1, PL2, PLN2, DE2, ETIME, ITIME,
+ NXTRA, BLOOD, APL1, APLN1, ADE1, APL2, APLN2, ADE2, K, XTRA)
REAL ITIME
DIMENSION FLUX(600), XTRA(8)
CALL CLEARSCREEN (SGCLEARSCREEN)
IF(K.NE.1) THEN
WRITE (*,10)
10 FORMAT(///,3OX,'SKIN DIFFUSION DATA'/,3OX,'INPUT PARAMETER LIST')
WRITE(*,20) TEMPIO, DENS, FLUX(1), BL, AK, JINC, TEMPB, ABSORB, BOIL
20 FORMAT(/,4X,'TEMPIO = ',F10.5,6X,'DENS = ',F10.5,7X,'Q1 = ',
+ F10.5,6X,'BL = ',F10.5,10X,'AK = ',F10.5,9X,'JINC = ',I2,/,4X,
+ 'TEMPB = ',F10.5,7X,'ABSORB = ',F10.5,5X,'BOIL = ',F10.5,/)WRITE (*,30) PL1, PLN1, DE1, PL2, PLN2, DE2, ETIME, ITIME, NXTRA, BLOOD
30 FORMAT(4X,'PL1 = ',F10.5,9X,'PLN1 = ',F10.5,7X,'DE1 = ',F10.5,/,4X,
+ 'PL2 = ',F10.5,9X,'PLN2 = ',F10.5,7X,'DE2 = ',F10.1,/,4X,
+ 'ETIME = ',F10.5,7X,'ITIME = ',F10.5,6X,'NXTRA = ',I2,/,4X,
+ 'BLOOD = ',F10.5,/)WRITE(*,40) APL1, APLN1, ADE1, APL2, APLN2, ADE2
40 FORMAT(4X,'APL1 = ',F10.5,8X,'APLN1 = ',F10.5,6X,'ADE1 = ',F10.1,/
+ 'APL2 = ',F10.5,8X,'APLN2 = ',F10.5,6X,'ADE2 = ',F10.1,/
IF (NXTRA.GT.0) WRITE(*,50) (XTRA(I),I=1,NXTRA)
50 FORMAT(5X,'THE EXTRA NODES ARE: ',8F6.1)
ELSE
WRITE(3,10)
WRITE(7,10)
WRITE(3,20) TEMPIO, DENS, FLUX(1), BL, AK, JINC, TEMPB, ABSORB, BOIL
WRITE(7,20) TEMPIO, DENS, FLUX(1), BL, AK, JINC, TEMPB, ABSORB, BOIL
WRITE (3,30) PL1, PLN1, DE1, PL2, PLN2, DE2, ETIME, ITIME, NXTRA, BLOOD
WRITE (7,30) PL1, PLN1, DE1, PL2, PLN2, DE2, ETIME, ITIME, NXTRA, BLOOD
WRITE (3,40) APL1, APLN1, ADE1, APL2, APLN2, ADE2
WRITE (7,40) APL1, APLN1, ADE1, APL2, APLN2, ADE2
IF (NXTRA.GT.0) WRITE(3,50) (XTRA(I),I=1,NXTRA)
IF (NXTRA.GT.0) WRITE(7,50) (XTRA(I),I=1,NXTRA)
K=0ENDIFEND

SUBROUTINE PROCEED(RESP, PROCED, AGAIN)
CHARACTER RESP*1
INTEGER PROCED, AGAIN
WRITE(*,10)
10 FORMAT(///,15X,'DO YOU WISH TO CONTINUE? TYPE Y OR N '),$)
READ(*,20) RESP
20 FORMAT(A1)
IF (RESP.EQ.'Y'.OR.RESP.EQ.'y') THEN
PROCED=0ELSE
PROCED=1CALL ANOTHER(AGAIN)
ENDIFEND
SUBROUTINE ANOTHER(AGAIN)
    CHARACTER RESP*1
    INTEGER AGAIN
    WRITE(*,10)
10 FORMAT(//,15X,'DO YOU WANT TO DO ANOTHER RUN? TYPE Y OR N '$)
    READ(*,20)RESP
20 FORMAT(A1)
    IF(RESP.EQ.'Y'.OR.RESP.EQ.'y') THEN
        AGAIN=0
    ELSE
        AGAIN=1
    END IF
END

SUBROUTINE DESCRIP(DSCRPT,SUMFILE,TFILE)
    CHARACTER SUMFILE*8,TFILE*8
    DIMENSION DSCRPT(20)
    WRITE(*,10)
10 FORMAT(///,SX,'ENTER THE MODEL NAME OR DESCRIPTION (UP TO 80',
+ '/9X,'CHARACTERS). THIS INFORMATION WILL BE USED',/9X,
+ 'AS A TITLE ON THE SUMMARY PAGE. '$)
    READ(*,20)DSCRPT
20 FORMAT(20A4)
    WRITE(3,3C)DSCRPT
30 FORMAT(//,1OX,'MODEL NAME OR DESCRIPTION: ',20A4)
    CALL CLEARSCREEN( SGCLEARSCREEN )
40 FORMAT(///,9X,'NOW ENTER THE SUMMARY FILENAME (UP TO 8',
+ '/9X,'CHARACTERS). THIS FILE WILL CONTAIN A',/9X,'SUMMARY'
+ 'OF THE SIMULATION. '$)
    READ(*,50)SUMFILE
50 FORMAT(A8)
    OPEN(UNIT=7,FILE=SUMFILE,FORM='FORMATTED',STATUS='UNKNOWN')
    WRITE(7,3G)DSCRPT
80 CALL CLEARSCREEN( SGCLEARSCREEN )
80 FORMAT(*,60)
    WRITE(*,60)
60 FORMAT(///,9X,'NOW ENTER THE TEMPERATURE FILENAME (UP TO 8',
+ '/9X,'CHARACTERS).',/9X,'THIS FILE WILL CONTAIN A LIST OF THE',
+ ' TEMPERATURES',/9X,'AT THE VARIOUS NODES DURING THE SIMULATION'
+ '.) '$)
    READ(*,70)TFILE
70 FORMAT(A8)
    OPEN(UNIT=4,FILE=TFILE,FORM='FORMATTED',STATUS='UNKNOWN')
END

SUBROUTINE SUB12(TIME,T,XTIME,JINC,BLUD,CP,BK,NXTRA,XTMP,M,EM)
    DIMENSION T(12),XTIME(12),CP(12,2),BK(12,2),XTMP(8)
    WRITE(3,10)TIME
10 FORMAT(/,45X,5HTIME=,F10.6:,T4,'T= ',6X,'XTIME='/(",G12.4)
    WRITE(*,20)BLUD
20 FORMAT(1X,'BLUD =',F6.5)
WRITE(3,30)(XTIME(I),I=1,JINC)

30 FORMAT(2X,'XTIME=',F10.5)
WRITE(3,40)T(1),CP(1,1),BK(1,1)

40 FORMAT(2X,'T=',G16.5,2X,'CP=',G16.5,2X,'BK=',G16.5)
IF (NXTRA.NE.0) THEN
  DO J=1,NXTRA
    WRITE(3,40)XTMP(J)
  END DO
END IF
WRITE(3,40)(T(J),CP(J,1),BK(J,1),J=2,JINC)
M=M+1
EM = M
END

SUBROUTINE DEPTH(X,Y,N,TD,IERR)
C********************************************************Inverse interpolation on two or three points to determine threshold depth (predicted burn depth) using either Y or LOG(Y)
DIMENSION X(1),Y(1),Z(3)
IERR = 0
IF (N.LT.2) IERR=-1
IF(IERR.NE.-1) THEN
  DO 100 I=1,N
    Z(I) = Y(I)
    Z0 = 1.
    IF (Z(1).NE.0..AND.Z(2).NE.0.) THEN
      IF (N.EQ.3..AND.Z(3).EQ.0.) N=2
      Z0 = 0.
      DO 120 I=1,N
        Z(I) = ALOG(Z(I))
      END DO
    END IF
    HO = Z(2)-Z(1)
    IF (HO.EQ.0.) IERR=-1
    IF (N.NE.2..AND.IERR.NE.-1) THEN
      H1 = Z(3)-Z(2)
      IF (H1.EQ.0.) IERR=-1
      IF(IERR.NE.-1) H2 = Z(3)-Z(1)
      IF (H2.EQ.0.) IERR=-1
      IF(IERR.NE.-1) DZ3 = Z0-Z(3)
    END IF
    IF(IERR.NE.-1) THEN
      DZ2 = Z0-Z(2)
      DZ1 = Z0-Z(1)
      IF (N.NE.2) THEN
        TD = DZ1*DZ2*X(3)/(H1*H2)-DZ1*X(2)*DZ3/(HO*H1)+X(1)*DZ2*DZ3 +/(HO*H2)
      ELSEIF(N.EQ.2) THEN
        TD = (DZ1*X(2)-X(1)*DZ2)/H0
      END IF
    END IF
  END DO
END IF
END
SUBROUTINE SUB102G(W,JINC,D,TMPMAX,TD,TIME,TP)
  DIMENSION W(12),D(12)
  WRITE(3,10)(W(I),I=1,JINC)
10  FORMAT(1X,'W=',E20.5))
  WRITE(7,20)(W(I),EXP(D(I)),I=1,JINC)
20  FORMAT(1X,'W=',E20.5,5X,'AT DEPTH (IN MICRONS)=',G20.6))
  WRITE(3,30)TMPMAX
  WRITE(7,3O)TMPMAX
  WRITE(*,30)TMPMAX
30  FORMAT(1X,'MAXIMUM TEMPERATURE = ',F8.3)
  WRITE(*,40)TD
40  FORMAT(1X,'THRESHOLD DEPTH = ',G20.4)
  WRITE(3,5O)TIME
50  FORMAT(1X,'FINAL TIME = ',F7.2)
  IF(TP.NE.999.) THEN
  WRITE(3,60)TP
60  FORMAT(1X,'TIME TO PAIN IS ',F7.2,' SECONDS. ')
  END IF
  END
SUBROUTINE HARVARD(PROFILE,TFILE,SUMFILE,PTS)
  CHARACTER PROFILE*8,SUMFILE*8,TFILE*8,HG*12
  INTEGER PTS
  WRITE(*,10)
10  FORMAT(///,9X,'TYPE A <CR> TO CONTINUE. ')
  READ(*,*)
  CALL CLEARSCREEN( $GCLEARSCREEN )
20  FORMAT(///,9X,'DO YOU WANT TO PLOT THE TEMPERATURES VS. TIME?',/,9X,'TIME IN HARVARD GRAPHICS? TYPE Y/N '$)
  READ(*,30) HGPLOT
30  FORMAT(A1)
  IF(HGPLOT .EQ. 'Y' .OR. HGPLOT .EQ. 'y') THEN
  CALL PLOTHG(PTS,TFILE,HG)
40  FORMAT(9X,'THE TEMPERATURES FOR THE HARVARD GRAPHICS PLOT',+ ' ARE IN FILE: ',/,9X,A12,'. USE "PRINT" OR "TYPE" AFTER YOU',+ ' EXIT THE PROGRAM',//,9X,'TO SEE IT.')
OPEN(UNIT=4,FILE=TPFILE,FORMATTED)
DO I=1,PTS
   READ(4,40) TIME(I),T1(I),T2(I),T3(I),T4(I),T5(I),
   + T6(I),T7(I),T8(I),T9(I),T10(I),T11(I),T12(I)
40 FORMAT(13(F9.5,2X))
END DO
WRITE(*,10)TPFILE
CLOSE(4)
10 FORMAT(/,SX,'THE TEMPERATURE DATA IS STORED IN FILE: ',A8)
WRITE(*,20)
20 FORMAT(/,9X,'ENTER THE FILE TO STORE HARVARD GRAPHICS ',/,9X,'TEMPERATURES USING UP TO 12 CHARACTERS',/,9X,'INCLUDING'
+ ' THE ENDING .DAT ',$)
READ(*,30) HG
30 FORMAT(A12)
IF(PTS.LE.60) THEN
   OPEN(UNIT=5,FILE=HG,FORM='FORMATTED',STATUS='UNKNOWN')
   DO J=1,PTS
      WRITE(5,60) TIME(J),T1(J),T2(J),T3(J),T4(J),T5(J),
      + T6(J),T7(J),T8(J),T9(J),T10(J),T11(J),T12(J)
60 FORMAT(13(F9.5,2X))
   END DO
   CLOSE(5)
ELSE
   INTERVAL=INT(PTS/60)
   OPEN(UNIT=5,FILE=HG,FORM='FORMATTED',STATUS='UNKNOWN')
   DO J=1,PTS,INTERVAL
      WRITE(5,80) TIME(J),T1(J),T2(J),T3(J),T4(J),T5(J),
      + T6(J),T7(J),T8(J),T9(J),T10(J),T11(J),T12(J)
80 FORMAT(13(F9.5,2X))
   END DO
   CLOSE(5)
END IF
END
APPENDIX C

REN12.DAT

This next file contains the initial values of the variables and constants required by BURNSIM. The file is REN12.DAT.

32.5, 1.0, 3.54, 0.22, 0.01, 100.15, 0.613
12, 4.5, 80.0, 3.02, 0.137, 0.001
0.5139, 0.8513, 0.8678, 0.8681, 0.8561, 0.8086
0.7802, 0.7537, 0.7326, 0.7209, 0.709
0.00059604, 0.0012236, 0.0012541, 0.0012547, 0.0011931, 0.0011439
0.0010912, 0.0010419, 0.0010028, 0.0009810, 0.0008
2.24, 239.47, 1.46, 147.37, 80000.0, 50000.
0.78, 285.52, 60, 117.43, 93534.9, 39109.8
0.137, 0.72596, 0.75574, 0.75638, 0.73439, 0.69632, 0.64869
0.598, 0.55081, 0.51364, 0.49298, 0.3

See the user's manual for definitions of these abbreviations (e.g. TEMPIO).

<table>
<thead>
<tr>
<th>ROW 1</th>
<th>ROW 2</th>
<th>ROW 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>TEMPIO = 32.5</td>
<td>JINC = 12</td>
<td>Cp(1) = 0.5139</td>
</tr>
<tr>
<td>DENS = 1.0</td>
<td>TEMPB = 4.5</td>
<td>Cp(2) = 0.8513</td>
</tr>
<tr>
<td>Q0 = 3.54</td>
<td>ITIME = 80.0</td>
<td>Cp(3) = 0.8678</td>
</tr>
<tr>
<td>BL = 0.22</td>
<td>ETIME = 3.02</td>
<td>Cp(4) = 0.8561</td>
</tr>
<tr>
<td>AK = 0.01</td>
<td>PCWATER = 0.137</td>
<td>Cp(5) = 0.8561</td>
</tr>
<tr>
<td>BOIL = 100.15</td>
<td>BLOOD = 0.001</td>
<td>Cp(6) = 0.8349</td>
</tr>
<tr>
<td>ASORB = 0.613</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ROW 4</th>
<th>ROW 5</th>
<th>ROW 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cp(8) = 0.7802</td>
<td>BK(1) = 0.00059604</td>
<td>BK(8) = 0.0010912</td>
</tr>
<tr>
<td>Cp(9) = 0.7537</td>
<td>BK(2) = 0.0012236</td>
<td>BK(9) = 0.0010419</td>
</tr>
<tr>
<td>Cp(10) = 0.7326</td>
<td>BK(3) = 0.0012541</td>
<td>BK(10) = 0.0010028</td>
</tr>
<tr>
<td>Cp(11) = 0.7209</td>
<td>BK(4) = 0.0012547</td>
<td>BK(11) = 0.0009810</td>
</tr>
<tr>
<td>Cp(12) = 0.7209</td>
<td>BK(5) = 0.0012322</td>
<td>BK(12) = 0.0008</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ROW 7</th>
<th>ROW 8</th>
<th>ROW 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>PL2 = 2.24</td>
<td>APL1 = 0.78</td>
<td>WATER(1) = 0.137</td>
</tr>
<tr>
<td>PLN2 = 239.47</td>
<td>APLN1 = 285.52</td>
<td>WATER(2) = 0.72596</td>
</tr>
<tr>
<td>PL1 = 1.46</td>
<td>APL2 = 0.60</td>
<td>WATER(3) = 0.75574</td>
</tr>
<tr>
<td>PLN1 = 147.37</td>
<td>APLN2 = 117.43</td>
<td>WATER(4) = 0.75638</td>
</tr>
<tr>
<td>DE2 = 80000.</td>
<td>ADE1 = 93534.9</td>
<td>WATER(5) = 0.73439</td>
</tr>
<tr>
<td>DE1 = 50000.</td>
<td>ADE2 = 39109.8</td>
<td>WATER(6) = 0.69632</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ROW 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>WATER(8) = 0.598</td>
</tr>
<tr>
<td>WATER(9) = 0.55081</td>
</tr>
<tr>
<td>WATER(10) = 0.51364</td>
</tr>
<tr>
<td>WATER(11) = 0.49298</td>
</tr>
<tr>
<td>WATER(12) = 0.3</td>
</tr>
</tbody>
</table>