



A11103 461768

NISTIR 4464

**NIST
PUBLICATIONS**

USING THE HARVARD/NIST MARK VI FIRE SIMULATION

**John A. Rockett
Guest Researcher**

**U.S. DEPARTMENT OF COMMERCE
National Institute of Standards
and Technology
National Engineering Laboratory
Center for Fire Research
Gaithersburg, MD 20899**

**U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary
NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
John W. Lyons, Director**

QC
100
.U56
#4464
1990
C.2



USING THE HARVARD/NIST MARK VI FIRE SIMULATION

John A. Rockett
Guest Researcher

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards
and Technology
National Engineering Laboratory
Center for Fire Research
Gaithersburg, MD 20899

November 1990



U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary
NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
John W. Lyons, Director

TABLE OF CONTENTS

TABLE OF CONTENTS	iii
LIST OF FIGURES	iv
LIST OF TABLES	v
ABSTRACT	1
1: INTRODUCTION	2
2: OVERVIEW OF H06	2
3: THE H06.3 DISKS	4
4: GETTING STARTED	5
Unpacking packed files (example H063OL, others similar)	5
4: USING H06.3	6
Main Menu Options	6
Option 13: Quit	6
Option 12: Create new data file	6
Option 1: Re-set default	7
Option 0: No more changes	8
Option 2: Caption and Comments	9
Option 3: Select physical subroutines	12
Option 4: Rooms	13
Option 5: Object Geometry	14
Option 6: Physical/chemical parameters of objects	15
Option 7: Vents	18
Option 8: Wall Properties	19
Option 9: Non-indexed variables	19
Option 10: Recap	19
Option 11: Output options	20
6: BATCH OPERATION	20
7: GENERAL COMMENTS	21
8: SEPARATE PROGRAM MASBANK	25
9: SEPARATE PROGRAM PREPLOT	26
10: USE WITH NON-PC COMPUTERS AND NOTES ON RE-COMPILING H06.3	28
Linking	29
11: EXAMPLE: INSTALLATION ON A NON-MS DOS COMPUTER	30
12: OTHER CHANGES IN H06.3	32
APPENDIX I: H063 SUBROUTINE MAP	35

LIST OF FIGURES

Figure 1. Main Menu	6
Figure 2. Use of menu option 13, "Quit".	7
Figure 3. Use of main menu Option 1.	7
Figure 4. Use of main menu Option 1.	8
Figure 5. Console screen after entering the name of an output file which already exists.	9
Figure 6. Console screen after entering names of both output files but before the final <RET>.	9
Figure 7. Caption edit feature of main menu Option 2.	10
Figure 8. Comment edit, main menu Option 2 (part 1)	11
Figure 9. Comment edit session (part 2).	11
Figure 10. Use of main menu Option 3.	12
Figure 11. Use of main menu Option 4.	13
Figure 12. Use of main menu Option 5.	14
Figure 13. Main menu Option 6, Use of Data Base for material properties.	15
Figure 14. Option 6, Time - Gas Flow Rate data for a Gas Burner fire.	16
Figure 15. Option 7, Number of vents and vent parameters	17
Figure 16. Forced Vent data input	18
Figure 17. File for re-directed input in batch mode.	21
Figure 18. Effect of Burner Diameter on Hot-Cold Layer Interface height and Hot Layer Temperature.	22
Figure 19: Plume flow versus height for various plume algorithms and vent flow versus height	23
Figure 20. Starting PREPLOT	27
Figure 21. Use of PREPLOT when a room variable has been specified.	27
Figure 22. Top of PREPLOT output file used to prepare Figure 22.	28
Figure 23. Plot prepared with MS CHART from PREPLOT output.	29
Figure 24. Concurrent 3252 File H063.CSS used to load and run H063OL.	31

LIST OF TABLES

Table 1: Gas Burner Diameter Study	21
Table 2: Comparison of Plume Algorithms	23
Table 3: Calculation Times for Two Emissivity Calculation Methods	24

USING THE HARVARD/NIST MARK VI FIRE SIMULATION

John A. Rockett

ABSTRACT

Installation and use of the Harvard/NIST VI multi-room fire simulation (computer code) on the IBM PC (and compatibles) is described. Two separate auxiliary programs, MASBANK and PREPLOT are also described and their use illustrated. Use of the input menus is described and illustrated. Comments are made on the effect of the choice of alternative sub-models on predicted results.

Some suggestions for use on other computers are made.

The H06.3 disks for use with this report include:

Disk #1: All files (in packed form) needed to run H06.3 and the two auxiliary programs: MASBANK and PREPLOT. A group of (packed) files including a short data base file and a sample output of H06.3.

Disk #2: Section 1 including 17 out of 25 source code files for H06.3.

Disk #3: Section 2 including the final 8 source code files of H06.3 and source code for MASBANK, PREPLOT and the auxiliary source code file DTIME.

Key words: Fire simulation, computer programs, manuals

USING THE HARVARD/NIST MARK VI FIRE SIMULATION

John A. Rockett

1: INTRODUCTION

The Harvard/NIST VI Multi-Room Fire Simulation has undergone three major revisions since it was delivered by Prof. H.W. Emmons to the Center for Fire Research of the NIST (formerly NBS). These versions of the computer code will be referred to as H06 (as delivered), H06.1, H06.2 and H06.3 (revisions). As delivered, the code was not fully operational and had several features which linked it uniquely to the DEC VAX 11/780 on which it was developed.

The first revision (H06.1), by M. Morita and J.A. Rockett, made the code operational on at least six computers: the UNIVAC-1108, IBM-370/158, Dec VAX-11/780, CONCURRENT-3252, CDC-6600 and IBM-PC compatibles. Because it was in conformance with ANSI standard FORTRAN VII (it passed the FORTRAN verifier on the IBM-370), it could be expected to run on most computers. After the program was running, several minor algorithm errors were discovered and corrected.

The second revision (H06.2), by J.A. Rockett re-dimensioned the simulation from its original 5 room/10 vent limit to 10 rooms/20 vents. Forced ventilation was added based on the algorithm, by H. Mitler, used in NIST/Harvard FIRST.[8] The development prior to H06.3 is detailed in [1-7].

The third, present, revision (H06.3), by J.A. Rockett, re-worked the input. This so enlarged the code that, without "over-lays", it would no longer fit on a PC using MS DOS. This report is the only currently existing documentation for H06.3.

The motivation behind the present changes was two fold. Several users, notably Prof Emmons, had urged simplification of the input. In addition, building batch input files was awkward. If a previously developed input file was edited to change the number of objects, rooms, or vents, it was difficult to get the correct number of carriage returns included towards the end of the file. In H06.3 the batch file problem has been solved. As for simplifying the input, "simple" remains in the mind of the user. The main "simplification" is that data entry is largely free of formatting restrictions. In addition, a menu allows selective changes to be made without spooling through the entire input process. Otherwise the input is very like that of FIRST and the older H06 versions.

The Harvard VI fire simulation has never been suitable for use by computer novices. The present revisions are not intended to make it so, only to make it somewhat less irksome.

Maintenance of H06 is not supported by NIST's CFR or any other organization. It is done as the author of this guide has the time and inclination. The result has not been reviewed except to the extent that people like you, who have the program, use it, and report back their experience. It is known that all big programs have bugs; H06 is no exception. You use it at your own discretion and risk.

2: OVERVIEW OF H06

H06 is a multi-room, single floor, "zone" fire simulation. The physical space to be modeled (collection of rooms) is divided into a number of "zones" (sub-spaces). The zones are chosen so that relatively few physical processes dominate the activities within a zone. Generally physical variables within a zone have values

representative of the average for that variable over the volume of the zone. Zone averaged variables are functions of time.

In H06 each room is assumed to be a rectangular box. Each room is divided into four zones. The gas within the room is divided into two of the zones: a "hot upper gas layer" and a "cool lower gas layer". The gas layer zones have constant area equal to the floor area of the room, but change in thickness as the fire progresses. The third and fourth zones are the walls, ceiling and floor considered as two separate units but with identical physical properties. The upper wall-ceiling zone has the same vertical thickness as the upper gas zone. The lower wall-floor zone is the remainder of the room boundary. Mass and energy flows from the lower to the upper gas layer within a single room via fire plumes. There may be up to three separate fires in any room. Flow between rooms occurs via free vents (doors or windows). Forced vents supply or exhaust air to/from a room. A total of 10 rooms and 20 vents may be specified. However, each room must have at least one free vent either directly to the out-doors or connecting to the out-doors via a chain of free vents through other rooms. Once the above condition has been met for each room, additional free or forced vents may be arbitrarily specified. Other than the 20 vent total, there is no limit to the number of vents assigned to a single room.

H06 assumes that the floor of each room is at the same height. It allows the ceiling heights to vary. However, because no provision is (at present) made for mixing between the hot gas flow out the top of a vent and cool air in the adjacent room, this flexibility in specifying ceiling height must be used with caution.

H06 allows specification of three fire types: a gas burner - fixed fire area, fuel flow rate user specified as a function of time; a pool fire - fixed fire area, fuel flow rate calculated by H06 based on a heat balance at the fire surface; a growing fire - fire area calculated using a user specified fire spread rate modified by the fire intensity, fuel flow rate calculated based on a heat balance at the fire surface. Up to three separate fires can be specified for each room. They need not be of the same type. By combining fires with differing parameters (and type) the burning of complex objects can be approximated. For many burnable objects the skill of the user may be severely taxed in doing this, however. Although fires may occur in multiple rooms, the program has currently very limited provision for spacing the ignition times. In general fires are assumed to ignite at time zero or are ignited by exposure to other fires. This appears to work in a realistic way for objects within a single room. The author has never tested if a fire in one room is able to stimulate ignition of a burnable object in an adjacent room. Although the various fire types yield a fuel flow rate, the burning rate of any fire may be limited either by the rate of fuel generation or by the availability of oxygen. In H06 only the oxygen from the lower gas layer entrained by the fire plume is considered in determining the oxygen available. This may result in earlier oxygen starvation than might occur in an actual case as the real fire is able to use oxygen from the gas in the upper part of the room (in the simulation, the upper gas layer).

Although ceiling, walls and floor of a room are assumed to have the same construction (physical properties) the properties may differ from room to room. However, the connectedness of walls is not considered in calculating the transient heating of the walls. The reverse side of each wall is assumed to be the out-doors even though, in reality, it might be another room. Note also that walls are homogeneous solids, hollow core walls are not treated. Because most wall materials heat slowly compared to the fire development, these limitations are not especially important except in certain specialized situations such as a building with thermally thin (for example, metal) walls.

In H06, the lower gas layer and the lower walls and floor do not change from ambient conditions. Early in a fire's development non-inclusion of floor heating is not too serious as the floor heats slowly. More serious is that no provision is made for a lower gas layer heat balance. Without this, mixing of hot layer gas with the lower layer at the vents cannot be included. It is known that this is often important because of its effect on smoke logging of the lower gas layer. Aside from life safety considerations, vertical radiation heat transfer is significantly altered by even relatively small amounts of smoke in the cool, lower layer.

3: THE H06.3 DISKS

If H06.3 will be used un-modified on a PC (compatible) only Disk #1 is needed. Disks #2 and #3 are needed if you wish to re-compile H06.3 or work with the source code. In order to get all the needed files onto a single 5-1/4" floppy disk, Disk #1 had to be packed. The unpacking process is very easy and described, step by step, below. H06.3 disks may be down-loaded from the "CFR Bullitin Board" or requested from the author.

The H06.3 disks include:

Disk #1:

- 1) H063.EXE, a packed version of the MS DOS executable file H063OL.EXE. This has been packed using PKZIP and ZIP2EXE. These two programs, from PKware, Inc, 7032 N. Ardara Ave., Glendale, WI 53209, have the advantage that they will pack H063OL.EXE whereas MS DOS' EXEPACK will not. H063OL is too big for EXEPACK. Instructions for unpacking H063OL.EXE and other packed files are given below.
- 2) The similarly packed .EXE file for the program, MASBANK, used for maintenance and up-date of the material properties data base files (discussed below). The packed file is MASSBANK.EXE.
- 3) The packed .EXE file for a special purpose pre-plot program, PREPLOT (discussed below). The packed file is PREPLOTZ.EXE.
- 4) A group of files packed together as file DISK1.EXE including:
 - a) A machine readable (ASCII) abstract of this report: Name README.
 - b) Data base files created by MASBANK with a small amount of data. The material values included may be of use, but there are no generally accepted "correct" values. Nobody guarantees these numbers.
 - c) A sample batch input file, REDIR.INF, suitable for use with the "re-directed input" feature of MS DOS. If a machine is targeted which uses a different operating system, this file will need to be altered, but can serve as a guide. REDIR.INF specifies the use of the file REDIR.INP. This is also included on Disk #1.
 - d) Files, DEMO.OUT and DEMO.PLT, which are the result of executing H06.3 with the default (i.e. built-in) input values. This is of interest to assure yourself that the program is running on your machine somewhat the way it runs on others. Past experience shows that different computers give slightly different results. Because H06 is double precision, differences should be minor. If **significant** differences occur suspect something. Note that, using the same FORTRAN source files but changing from the Ryan McFarland to Lahey FORTRAN compiler, on the same machine, resulted in differences as large as the third of five significant figures.

Disk #2:

- 1) A linker input file, H063OL.LNK, for use with "Plink86 plus". If another linker is used, this file will not be directly useful, but can serve as a guide for constructing a similar file for use with your linker. Note however, that the linker used must have program segment overlay capabilities if the resulting .EXE file is to be used on a PC whose operating system is limited to 640 KB. The link file lists the source files of H06.3.
- 2) Section 1 of the H06.3 source code. The source code is divided into 25 files comprising the MAIN program, two BLOCK DATA, and 118 SUBROUTINE/FUNCTION algorithms. Names: xxx.FOR. Disk #2 contains 17 of the 25 sources files. The remainder are on Disk #3.

Disk #3:

- 1) The remaining 8 H06.3 source files.
- 2) The source code for the programs MASBANK and PREPLOT and the auxiliary file DTIME.

The files on Disks 2 and 3 are not packed.

4: GETTING STARTED

The next paragraphs describe installation of H063, un-modified, on a PC compatible. For installation on other machines or for users who intend to modify the code see Section 10: "Use with non-PC Computers and Notes on Re-compiling H06.3", and Section 11: "Example: Installation on a Non-MS DOS Computer".

You need not have a math co-processor to run H06.3 on a PC (compatible). But, if you do not have one, you should be no older than middle age if you expect to use the output. In short, a math co-processor is strongly recommended. The compiled files included here assume a co-processor. You will have to re-compile the sources and re-link the object code if you do not have a math co-processor.

The following installation description assumes you have a hard disk. If you don't have a hard disk you are probably in trouble trying to use H063OL although it can be done if you have a 720 KB or 1.2 MB, 5-1/4" disk drive or a 3-1/2", 1.4 MB drive.

Unpacking packed files (example H063OL, others similar)

Note: In the following "C>" indicates the operating system prompt, ":" the H06 data prompt. The text following the prompt is supplied by the user. Depressing the "Enter" key is indicated by <RET>. Figures illustrating use of the program show the relevant part, not all of the console screen. Text entered by the user has been highlighted.

Create (or select for use) a directory on your hard disk: For example, if \FTN\H063 is to be created and used type:

```
C>MKDIR C:\FTN\H063
```

Copy all the files from Disk #1 (assumed to be in drive A:) into directory C:\FTN\H063. Type:

```
C>COPY/V A:\DISK1\*. * C:\FTN\H063
```

Now unpack H063OL by typing:

```
C>CD\FTN\H063
C>H063
```

The result should be a comment on your screen indicating use of the program PKSFX and the unpacking of H063OL. When this is complete the directory \FTN\H063 should contain both H063.EXE and H063OL.EXE. You can delete H063.EXE as it is no longer needed. H063OL.EXE will be 697.092 KB in size. (Don't confuse the size of the file containing the executable code with the memory required to load the program, 565 KB!)

Unpacking the other files is done in the same way: the files were copied to your hard disk as a part of the process described above. Now type in the name, in turn of each packed file (without extension). The file will automatically unpack itself, placing the resulting unpacked files on your hard disk. Once unpacked, the originalEXE file can be deleted from your hard disk.

H063OL requires 565 KB of memory to run. The "640 KB" allowed by the operating system being actually 655.360 KB, this means that your MS DOS operating system plus any resident utilities and buffer files must not use more than 90 KB. For example, MS DOS 3.2, by itself presents no problem. It uses 49 KB, leaving free 606.256 KB.

Use of H06.3 is described next. Use of the other programs copied from Disk #1: MASBANK and PREPLOT, are described in Sections 8 and 9 respectively.

4: USING H06.3

The following description assumes you are using an MS DOS machine. Use will be essentially the same on any computer, but some details of the interface with the OS may differ. (Batch operation will be discussed later.)

Once the H063OL.EXE file has been generated, normal operation is initiated, assuming that you are already in the directory containing the H063 files, by typing (followed by <RET>)

```
C>H063OL
```

This loads the program and begins operation. The first program action is to display the Main Menu Options (Figure 1). 14 options are offered. Most of the 14 options refer, individually, to features of the input sequence of the older H06 versions. New features are 0, 1, 12 and 13. They will be discussed in the order 13, 12, 1 and 0.

```
Menu for Input Data Changes
-----
0) No more changes, proceed with computation
1) Re-set default values using existing data file
2) Enter/Edit Run Caption and Comments
3) Select physical subroutines
4) Choose number of rooms and room geometry
5) Geometric parameters of objects
6) Physical/chemical parameters of objects
7) Number of vents and vent parameters
8) Wall properties
9) Non-indexed variables
10) Recap current input set
11) Change output options
12) Create new default re-set data file
    based on current configuration
13) Quit

Please enter number of section to be changed
:
```

Figure 1. Main Menu

Option 13: Quit

Option 13: "Quit", stops the program execution (after first asking if you truly want to stop). (Figure 2) If this is answered "Y" the program exits to the system prompt. New files created by the operation up to this point remain catalogued on the working directory. If you answer with "N" the program re-presents the main menu.

Option 12: Create new data file

Option 12: "Create new default re-set data file based on current configuration" saves the current input data. I.e., the input data set resulting from the built-in default values overlaid by any subsequent changes made up to the selection of Option 12.

Option 12 first asks for the name of a new file to be used to save the current input data (Figure 3.) In Figure 3, THREE.INP has been entered.

If the file named already exists on the current directory, the program prints: "File already exists, use another name" and asks if you wish to see the list of names of input files that have previously been created by the option 12 process (Figure 3, lower half). (Note that this will not give all file names on the current directory.) When a satisfactory name has been provided, this name is added to Option 12's list of input files. All the current geometric information and object property data needed to execute the program will be written to this file and the main menu re-presented. You can then make some additional changes, say in fire size, and use Option 12 to create another input file. In this way a set of related inputs may be generated for later execution in batch mode (for which see below).

```

Menu for Input Data Changes
-----
.
13) Quit

Please enter number of section to be changed
:13

QUIT! Are you sure? (Ans Y/N)
:Y

```

Figure 2. Use of menu option 13, "Quit".

```

Menu for Input Data Changes
-----
.
12) Create new default re-set data file
13) Quit

Please enter number of section to be changed
:12

Enter name of NEW default re-set file (12 Characters max)
:THREE.INP

File already exists, use another name

Would you like to see the list of input files logged to H06 (Y/N)
:Y

The following input files have been logged for H06

  1) S-W3B.INP      2) REDIR.INP      3) THREE.INP
  4) S-W7.INP      5) CHATS75K.INP   6) TEM.INP
Enter name of NEW default re-set file (12 Characters max)
:

```

Figure 3. Use of main menu Option 1.

Option 1: Re-set default

Similar to Harvard V, H06.n contains a self consistent set of input data - the same single room with two urethane foam mattress-like burnable objects that is the "default" case of NIST/Harvard FIRST.[8] If you make no changes, this is the case which will be executed. However if, during some earlier session with H06.3, you have prepared input describing some other situation and have saved this using Option 12, then Option 1: "Re-set default values using existing data file", allows this input data set to be recalled and used as the new "default" data. This is useful in moving through a set of related calculations.

Specifying Option 1 produces the screen shown in Figure 4. The list of file names logged by Option 12 is presented and the user asked "Should

any of these be deleted?" This is a simple precaution to help keep the current directory from growing to fill all available disk space. If the answer Y is returned a screen similar to the middle portion of Figure 4 will be presented. The number of the file to be deleted is solicited. Upon giving a number the name of the file is presented with a delete confirmation request. This sequence is repeated until the delete question is answered with an N.

N produces the screen shown at the bottom of Figure 4. If the named file exists in the current directory - it need not be on the name list presented! - the file will be used to re-set the current case data. If the file can't be found, "File name error, ..." is displayed and the file name request repeated. If the file name is found and the reset is successful the main menu is then re-presented.

```

Menu for Input Data Changes
-----
0) No more changes, proceed with computation
1) Re-set default values using existing data file
.
Please enter number of section to be changed
:1
The following input files have been logged for H06

1) S-W3B.INP      2) REDIR.INP      3) THREE.INP
4) S-W7.INP      5) CHATS75K.INP   6) TEM.INP
7) TEST.INP
Should any of these be DELETED? (Y/N)
:Y
Enter number of file to be deleted
:6
Delete 6) TEM.INP Please confirm (Y/N)
:Y
The following input files have been logged for H06

1) S-W3B.INP      2) REDIR.INP      3) THREE.INP
4) S-W7.INP      5) CHATS75K.INP   6) TEST.INP
Should any of these be DELETED? (Y/N)
:N
Enter name of EXISTING file to be used to reset defaults
:THRE.INP
File name error, please enter name of an existing file
Enter name of EXISTING file to be used to reset defaults
:THREE.INP
Read of default reset file complete

Menu for Input Data Changes

```

Figure 4. Use of main menu Option 1.

Given a satisfactory resolution to the output file question, the program asks if you wish to use a second, plot-output file. This file will contain specially formatted output for use with the program PREPLOT¹. If used, it too must be a new file. After entering these file names (and assuming that both were, indeed, new files) the screen should appear similar to Figure 6.

After opening the output file(s) the current input is recapped as the output file header. The input section code currently in memory is over-written by the calculation code and the calculation proceeds using the input values current when the "0" option was chosen.

If you have any question about just what the current input is, Option 10, "Recap current input set", will display the **current** input description and return you to the main menu.

Option 0: No more changes

Option 0: "No more changes, proceed with computation" does almost what it says. However, before proceeding it asks if you want to avail yourself of Option 12. If you answer "N" it switches program control from the input section to the main program. If you answer "Y" it proceeds to Option 12. On completion of Option 12 the main menu is re-presented; i.e., you must eventually answer the Option 0 query with an N to get on with the calculation. Note that there is no requirement that Option 12 be used, the query is presented in case you wanted to save your input in machine readable form but have forgotten to do so.

Before starting the calculation the program asks for the name of the output file, i.e. the file onto which the calculation results will be written (top Figure 5). Output can be sent to the computer console (enter CON) or directly to the printer (enter PRN). More often it will be sent to a **new** disk file. Typing the name of the output file, for example DEMO.OUT, followed by a carriage return <RET>, causes the program to attempt to open a new file with this name. If it discovers a file with this name already catalogued in the current directory, it will indicate this fact and re-request an output file name (bottom Figure 5).

¹ PREPLOT reads the plot file output and prepares sub-files which can be used directly as input to your plot package. As written, PREPLOT's output is easily read by LOTUS 1-2-3. It is also usable with MS CHART. Relatively obvious changes in PREPLOT would, if needed, allow it to produce input files suitable for other plot packages.

Menu options 2-11 allow selective use of separate **pieces** of the input sequence of earlier versions of H06.

```
Enter name of output file, 12 character max.
Enter blank to default to "FIRE.OUT" (a new file)
Enter "PRN" for printer, "CON" for console.
:DEMO.OUT
File already exists, use another name

Enter name of output file, 12 character max.
Enter blank to default to "FIRE.OUT" (a new file)
Enter "PRN" for printer, "CON" for console.
:
.
```

Figure 5. Console screen after entering the name of an output file which already exists.

```
Enter name of output file, 12 character max.
Enter blank to default to "FIRE.OUT" (a new file)
Enter "PRN" for printer, "CON" for console.
:USEDEMO.OUT

Do you want a plot output fire? Answer (Y/N): Y

Enter name for plot output file, 12 characters max.
Enter blank to default to "FIRE.PLT" (a new file)
:USEDEMO.PLT
```

Figure 6. Console screen after entering names of both output files but before the final <RET>.

Option 2: Caption and Comments

Earlier versions of H06 asked for a one line title for the case about to be run and allowed the addition of up to 10 lines of comments about the calculation. Option 2 accomplishes the same purpose but with some additional flexibility. It is a simple line editor which displays the current run caption (for example a caption read from an earlier run file which had been saved with Option 12 and loaded with Option 1) and allows you to edit it. (I am not clever enough to write a **machine independent** "full-screen" editor, hence the present line editor choice.) The editing capability is limited.

The one-line title is edited separately. Three editing functions are available for this. If there is no title, use APpend; to change an existing title use either CHange or, if you wish to replace the entire existing title, use APpend; to move on to the next section (Comments) use ENd. (More will be said about these edit commands in later paragraphs.)

Figure 7 shows the screen after selecting Option 2. In this case there was no existing run title. AP has been used to enter a title, but a typographic error was made. CHange was used to correct this. (Note that, after each edit, the (modified) caption is re-displayed.) The result being now satisfactory, ENd finishes the Caption part of Option 2. The program moves immediately to the Comment Edit section of Option 2 (bottom, Figure 7).

To edit the comments there are five edit functions. An edit function is activated by typing the first two letters of the function. (You may type more but only the first two are sensed.) **Capital letters** are required!

- 1) To add lines use APpend. Type in a new line followed by a <RET>. Continue typing lines until (1) you have said all that is needed. A <RET> (with no text other than blanks) will terminate the append function and display the new set of comments. Or (2) when 10 lines total have been entered the program will automatically terminate the append function. The new set of comments will be displayed. As noted above, AP used with the Caption line, results in a new caption line.
- 2) To delete a line use DElete. The program will ask for the **number** of the line to be deleted. As soon as the number and <RET> have been entered the line will be deleted and lines below it moved up. No second, confirmatory prompt is supplied! The new comment set will be displayed.

3) To insert an extra line within the existing comments use INsert. The program will ask for the number of the line immediately after which the new line is to be added. Enter the line followed by a <RET>. The next program operation will be to push down the comments below the indicated line to make room for the new line. If this will result in loss of the tenth line, a warning will be displayed with the option to continue or retreat.

4) To INsert a comment line before the current first line, enter line number 0 after selecting the INsert function.

4) To change a string of characters within a line without having to re-type the entire line use CHange. The program will ask for the line number within which the change is to be made. (There is **no universal** change feature.) It will then ask you to enter the string to be searched for and replaced and the replacement string. The required syntax is

```
\search string\,replacement string\
```

For example, suppose the comment set contains 6 lines of which the 4th is

```
4: Growing fire, Rinit = 0.200 m.
```

For the case about to be run the initial fire radius has been increased to 0.5 m which we wish to note in the comments. In this case enter

```
CH
4
\0.200\,0.5\
```

The result should be

```
4: Growing fire, Rinit = 0.5 m.
```

Actually the "back slash" can be any character. The program senses the first character entered and uses this for the string delimiter. The only firm requirement is that the de-limiter appear only at the beginning and end of each string and

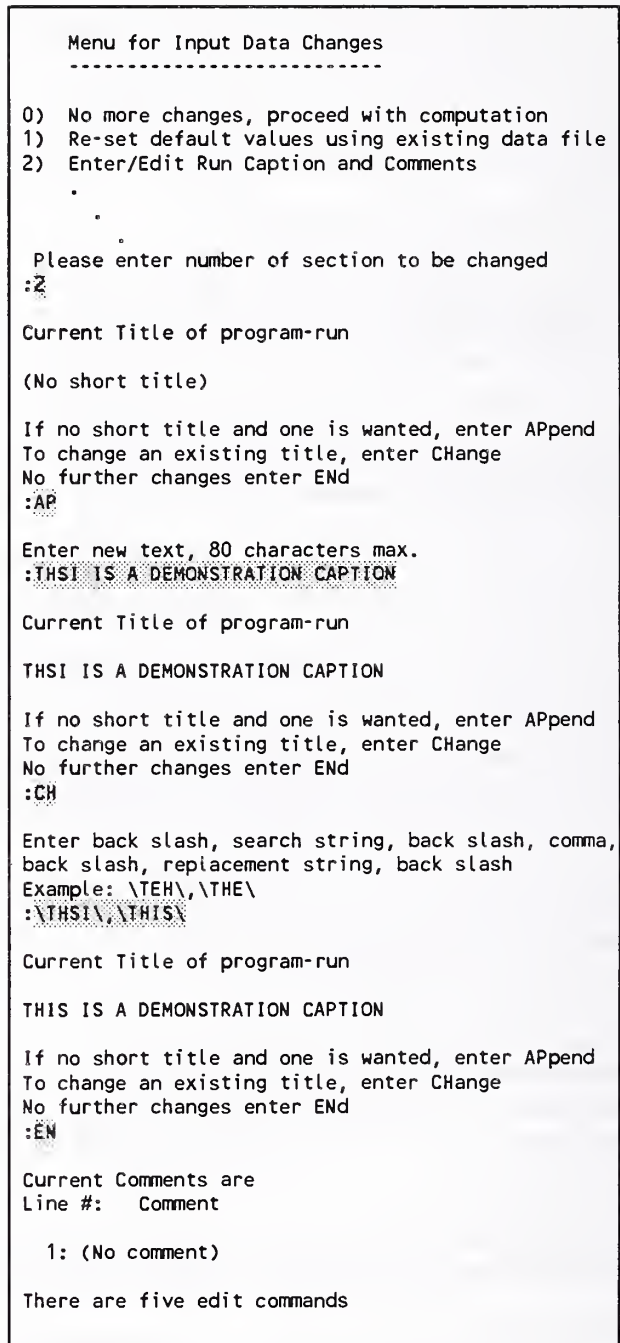


Figure 7. Caption edit feature of main menu Option 2.

that the central two be separated by a comma.

5) To exit from the editor use ENd. This will result in display of the main menu.

```

There are five edit commands
  APPend, DElete, INsert, CHange, ENd
Enter your choice
:AP

Enter new text, 80 characters max.
:LINE #1
:LINE #2
:LINE #3
:
:

Current Comments are
Line #:  Comment

  1: LINE #1

  2: LINE #2

  3: LINE #3

There are five edit commands
  APPend, DElete, INsert, CHange, ENd
Enter your choice
:IN

Enter number of line following which the insertion is to
:2

Enter new text, 80 characters max.
:LINE INSERTED AFTER LINE #2. IT WILL BE NEW LINE #3

Current Comments are
Line #:  Comment

  1: LINE #1

  2: LINE #2

  3: LINE INSERTED AFTER LINE #2. IT WILL BE NEW LINE #3

  4: LINE #3

There are five edit commands
  APPend, DElete, INsert, CHange, ENd
Enter your choice
:DE

Enter number of line to be delted
:2

Current Comments are
Line #:  Comment

  1: LINE #1

  2: LINE INSERTED AFTER LINE #2. IT WILL BE NEW LINE #3

  3: LINE #3

There are five edit commands
  APPend, DElete, INsert, CHange, ENd
Enter your choice
:EN
Menu for Input Data changes

```

Figure 8. Comment edit, main menu Option 2 (part 1)

```

5: L 5
6: L 6
7: L 7
8: L 8

There are five edit commands
  APPend, DElete, INsert, CHange, ENd
Enter your choice
:AP

Enter new text, 80 characters max.
:LINE #9
:LINE #10
:
:

Current Comments are
Line #:  Comment

  1: L 1

  2: L 2

  3: L 3

  4: L 4

  5: L 5

  6: L 6

  7: L 7

  8: L 8

  9: LINE #9

 10: LINE #10

There are five edit commands
  APPend, DElete, INsert, CHange, ENd
Enter your choice
:IN

Enter number of line following which the insert
:$

Enter new text, 80 characters max.
:TEXT TO BE INSERTED AFTER LINE 5

Text will be lost from line 10! Proceed? (Y/N)
:

```

Figure 9. Comment edit session (part 2).

Note that the short title can be edited as a comment by using the CHange command and entering 0 as the line number. The edited result is not displayed, but will appear if you exit (END) and re-enter from the main menu.

Figures 8 and 9 illustrate a comment edit session. Initially there was no comment (see bottom of Figure 7). APPend was used to enter three lines of comments: Line #1, Line #2 and Line #3. (see top of Figure 8) INsert was then used to add an additional comment between lines 2 and 3. To make room for the inserted line, the original line 3 was pushed down to become line 4. (mid Figure 8) Next line 2 was DEleted. (bottom of Figure 8).

Figure 9 shows an edit session where, initially, there were 8 comment lines with text: L 1 - L 8. (Part of the initial comment listing appears at the top of Figure 9.) Two additional lines were appended. Next an attempt was made to insert a line between lines 5 and 6. This led to the warning message: "Text will be lost from line 10! Proceed? (Y/N)". If this query is answered "Y", lines 6-9 will be pushed down. Line 10 will be lost and the insert line will become the new line 6. If the query is answered "N" the INsert will be voided and the edit prompt "There are five ..." presented.

When editing of the comments is completed, EN will exit to the main menu.

If Option 10 is used, it will display the current caption and comments at the top of its first screen.

```
Menu for Input Data Changes
-----
.
.
3) Select physical subroutines
.
.
Please enter number of section to be changed
:3
WHICH VERSION OF TMPO DO YOU WANT      (1 OR 2)
Current choice is  2
1: Short-cut method
2: Integration of transient heat-conduction equation
:2
WHICH VERSION OF ABSRB DO YOU WANT      (1 OR 2)
Current choice is  2
1: Broad band approximation plus soot (Modak)
2: Absorption coefficient proportional to soot mass
:1
WHICH VERSION OF PLUME DO YOU WANT      (1, 2 OR 3)
Current choice is  1
1: MORTON-TAYLOR-TURNER PLUME
   (VIRTUAL POINT SOURCE BELOW FIRE BASE)
2: SAME PLUME AS #1, BUT POINT SOURCE IS AT FIRE BASE
3: McCaffrey (empirical data fit)
4: Cetegen-Kubota-Zukoski (semi-empirical data fit)
:3
Menu for Input Data Changes
-----
```

Option 3: Select physical subroutines

Option 3: "Select physical subroutines" allows selection of alternative subroutines to be used for three calculations: the heating of objects, the gas extinction coefficient, and the entrainment into and mass flow of the fire plume.

Figure 10 illustrates the use of main menu Option 3. In this example, the default (method 2) is used for object temperatures (TMPO); method 1 (Modak) is chosen for extinction coefficient. In selecting the default method (TMPO) a <RET> with no number would have done equally well.

There appears to be no significant speed advantage for the "short cut" method for calculating object temperatures. The present finite difference method, method 2, is generally good but, be warned, it does not work for thermally thin materials - i.e. one represented by a single finite difference layer.

For the gas extinction coefficient of methane flames (or other very low sooting fuels), the use of calculation option 1 (Modak) is recommended.

Figure 10. Use of main menu Option 3.

The plume algorithm options are discussed in Section 7: General comments. The choice of plume algorithm can have a significant effect on simulations results. In the example of Figure 10, plume option 3, the McCaffrey plume has been selected.

Option 4: Rooms

Main menu Option 4: "Choose number of rooms and room geometry" operates slightly differently than earlier versions of H06. Formerly, all the room data had to be reviewed. In the present version, after confirming, or changing the number of rooms, the user is asked which room is to be reviewed.

To confirm the present number of rooms press <RET> or enter the number shown; to change the number of rooms, enter the new number followed by <RET>. (see Figure 11) Newly added objects have the default geometry and properties until amended (see menu options 5 & 6 below).

Upon entering the number of the room whose data is to be reviewed, the room's current data is presented: item number, a description of the item, its units and, last, the current value.

To amend an item, enter the item number a space (or a comma) followed by the new value. Repeat this for as many items as you wish, in any order. If you make a mistake, enter the item again, the last entry for any given item is the one which counts. To finish enter <RET> (a blank line). The new set of values will be displayed and you may either further amend them or, by entering a blank line, move on to the next prompt which asks if you wish to review data for another room. If this is answered Y the room number is requested and one proceeds as above. If the reply is N the main menu is presented.

To use item #100 enter 100<RET>. The program will ask for the number of the room whose data is to be used for the present room.

```

Menu for Input Data Changes
-----
.
.
4) Choose number of rooms and room geometry
.
Please enter number of section to be changed
:4
Enter Number of Rooms. Current value is 3
:3
Enter number of room whose data is to be reviewed
:2
(2 paragraphs of comments/instructions appear)
ROOM NUMBER 2:
0) CHANGE NOTHING (CONTINUE)
1) LENGTH ALONG X [M]: 3.0000E+00
2) LENGTH ALONG Y [M]: 4.0000E+00
3) HEIGHT [M]: 2.5000E+00
4) # OF OBJECTS: 0.
5) AMBIENT TEMPERATURE [K]: 3.0000E+02
100) COPY ALL ABOVE DATA FROM ANOTHER ROOM
(NOTE THAT TYPING "100" ALONE SUFFICES; THERE ARE NO
RELEVANT DATA VALUES NEEDED)

To change items, enter "Item (space) Value<RET>";
END WITH AN EXTRA <RET>
:2.4.5
:4.1
:
ROOM NUMBER 2:
0) CHANGE NOTHING (CONTINUE)
1) LENGTH ALONG X [M]: 3.0000E+00
2) LENGTH ALONG Y [M]: 4.5000E+00
3) HEIGHT [M]: 2.5000E+00
4) # OF OBJECTS: 1.
5) AMBIENT TEMPERATURE [K]: 3.0000E+02
100) COPY ALL ABOVE DATA FROM ANOTHER ROOM
(NOTE THAT TYPING "100" ALONE SUFFICES; THERE ARE NO
RELEVANT DATA VALUES NEEDED)

To change items, enter "Item (space) Value<RET>";
END WITH AN EXTRA <RET>
:
Review data for another room (Y/N)
:N

Menu for Input Data Changes
-----

```

Figure 11. Use of main menu Option 4.

Figure 11 illustrates the above process for room 2 of a three room data set (previously loaded using Option 1.) The top of Figure 11 shows the selection of Option 4 and confirmation that 3 rooms are intended (the number of rooms could be changed by entering a different number). Room 2 is chosen for

editing. After choosing the room number, text (not shown in the figure) will appear, enter <RET> to move on. The current data is presented (mid Figure 11). Entries to change the room length from 4 to 4.5 m and number of objects from 0 to 1 are made, followed by an extra <RET>. In this example no other room geometry changes are to be made so, after the revised data has been presented (bottom of Figure 11) and found correct a <RET> leads to the final query.

Review data for another room (Y/N)

This is answered N; the program returns to the main menu. If data for another room also needed to be changed, the answer Y would have led to a return to the query (see mid Figure 11):

Enter number of room whose data is to be reviewed

Upon entering the room number, the current data for that room would be displayed and the process described above repeated for the second room.

Item #5 of the Room Geometry data set is the ambient temperature. The program allows entry of different temperatures for each room and the out-of-doors. (Rather than a single, adjustable temperature for all.) This flexibility of input is not yet backed up by any program structure to allow for temperature differences. If one temperature is changed, all the room temperatures and the outside temperature (see main menu Option 9) should be changed to the same value.

Option 5: Object Geometry

Main menu Option 5: "Geometric parameters of objects" has also been changed somewhat from that of earlier versions of H06. The principal change is that, similar to Option 4, the coding permits data for a selected object from a selected room to be reviewed. After finishing this review, the user can either move on to another object in the same, or another room, or exit to the main menu.

```

Menu for Input Data Changes
-----
.
.
5) Geometric parameters of objects
.
.
Please enter number of section to be changed
:5

WHEN ENTERING THE OBJECT GEOMETRY BELOW, NOTE THAT THE CARTESIAN ...
.
.
HIT <RET> TO CONTINUE
:
Enter number of room of object whose data is to be reviewed
:1

Room # 1 contains 2 objects. Which ones data do you wish to review?
:1

0) CHANGE NOTHING (CONTINUE)
1) X-COORDINATE [M]: 1.5000E+00
.
.
100) COPY ALL ABOVE DATA FROM ANOTHER OBJECT
.
UNLESS THERE IS A COMPELLING REASON ....
.
To change items, enter "Item (space) Value<RET>";
END WITH AN EXTRA <RET>
:
Review data for another object in this room? (Y/N)
:N
Review data for objects in another room? (Y/N)
:

```

Figure 12. Use of main menu Option 5.

Figure 12 illustrates the use of Option 5. After selecting main menu Option 5, explanatory text is presented. (This is indicated in Figure 12, but not fully presented.) Next, in this example, Room #1 is requested. The program responds, showing the number of objects prescribed (via main menu Option 4, Room menu item #4) for this room. In this case 2 objects. The object number is then solicited. Object 1 is selected and the object's geometric data appears on the screen. (This is only indicated on the figure without completely showing the data.) Changes would be made in the same manner as with Option 4, Room geometry, discussed above - item number (space) new value. (For choice of Gas Burner Diameter, see Section 7: General Comments.) In this example, after examining the data, it was decided that no changes were needed. (A simple <RET> was entered after the data display.) The next query is "Review data for another object in this room? (Y/N)". Note that, if there were only one object in this room this query would be skipped. Here it is answered "N" and results in the query: "Review data for objects in another room". Entering another, valid room number would result in the display: "Room # ... contains ... objects. Which ones data do you wish to review?". The data amendment process would proceed as described above. If the "another room" query is answered "N" the main menu is presented.

Option 6: Physical/chemical parameters of objects

As with Option 5, Option 6 allows data for a selected object from a selected room to be reviewed with the ability to either move on to another object, another room, or exit to the main menu.

Option 6 allows use of a Materials Data Base. It employs the same software as NIST's FIRST. (see [8] pp 20-24, 66-73, 120-123) Thus the Data Base files, MASNAM and MASDAT, created and maintained by the separate program, MASBANK (see Section 8, below), can be shared by the two programs.

Figure 13 shows the use of main menu Option 6. Because the first two queries, "Enter number of room of object whose data is to be reviewed" and "Room # ... contains ... objects. Which ones data do you wish to review?" are identical with those of Option 5 (see Figure 12), these are not shown.

```

Do you want to search the Data Base
for a material with properties suitable for
Object 1 in Room 1? Ans (Y/N)
:Y

THE FOLLOWING MATERIALS ARE INCLUDED IN THE DATA BASE
 1 PMMA                2 POLYETHYLENE
WHICH ONE OF THE ABOVE MATERIALS DO YOU WANT TO USE?
PICK A NUMBER >=1 AND <= 2
1

MATERIAL SELECTED IS 1 PMMA
THE DATA BASE INCLUDES THE FOLLOWING NOTICE ABOUT THIS MATERIAL:

 polymethyl methacrylate (acrylic plastic)

DATA IN THIS BASE IS DERIVED FROM SPECIFIC MATERIALS/OBJECTS TESTED IN
VARIOUS WAYS WHICH MAY, OR MAY NOT, BE APPROPRIATE TO YOUR APPLICATION.
DISPLAY THE DATA FOR REVIEW AND POSSIBLE MODIFICATION? (Y/N):N

ROOM NUMBER 1, OBJECT NUMBER 1 (OBJ KO = 1) - PHYSICAL PROPERTIES:
IS THE OBJECT BURNING? (Y/N): Y

INPUT TYPE OF FIRE (1=GROWING, 2=POOL, 3=BURNER): 1

```

Figure 13 picks-up the sequence after the room number and object number have been chosen. The next lines displayed: "Do you wish to search the Data Base ..." introduces the materials data base. If, as was done in the example shown in Figure 13, this query is answered "Y", the next questions are asked from the Data Base Subroutine, DBANK. This is identical to that of FIRST and its use, as described in [8], applies. The particular Data Base referenced for the example of Figure 13 contains data for only two materials: PMMA and Polyethylene. The PMMA, material #1, is selected. After displaying a

Figure 13. Main menu Option 6, Use of Data Base for material properties.

caveat about the data, the user is asked if he wishes to see the data for this material. Usually this can be answered with "N" as the result will be displayed by, and can be changed within H06.

Before displaying data for the object, two more questions are asked: "Is the object burning?" and, if the answer to this is "Y", the "type of fire?" (see bottom, Figure 13 and top of Figure 14). If the burning question is answered "N", the next question is "Is it flammable?" Note that, if the object is **not** burning and is **not** flammable, an abbreviated list of material properties will be shown (see below).

After receiving answers to these two questions, the data is displayed. For a Growing Fire 18 properties are listed. For Pool and Gas Burner fires, 17. The omitted item is the basic rate of flame spread (the parameter AFIRE, cf. [8,9]) which is not applicable for constant burning-area Pool and Burner fires. For a non-burning, non-flammable object only 4 properties are presented, i.e. those which control the rate of heating of the inert solid. Some data items listed for Gas Burners are not needed, but no separate list for Burner Fires is displayed.

```

ROOM NUMBER 1, OBJECT NUMBER 1 (OBJ KO = 1) - PHYSICAL PROPERTIES:
IS THE OBJECT BURNING? (Y/N): Y

INPUT TYPE OF FIRE (1=GROWING, 2=POOL, 3=BURNER): 3

1) THERMAL CONDUCTIVITY [W/M-K]: 5/4000E-02

.
100) COPY ALL ABOVE DATA FROM ANOTHER OBJECT
To change items, enter "Item (space) Value<RET>";
END WITH AN EXTRA <RET>
:

Current number of segments for Gas Flow Rate curve is 3
NOTE: INITIAL FIRE SIZE PRE-SET TO 0.1 KW/M**3
FIRST ENTRY IS FOR THE SECOND POINT
Gas Burner Flow Curve Segments
Seg # Time [S] Flow [Kg/S] Qdot-nominal [KW]
1 0. 1.1658E-05 3.3457E+03
2 10. 1.0000E-03 2.8700E+04
3 30. 1.0000E-02 2.8700E+05
4 100. 1.0000E-03 2.8700E+04
Do you want to change number of segments in the Gas Flow Rage Curve?
To confirm present number enter <RET>; to change enter new number (Max 30)
:2

Gas Burner Flow Curve Segments
Seg # Time [S] Flow [Kg/S] Qdot-nominal [KW]
1 0. 1.1658E-05 3.3457E+03
2 10. 1.0000E-03 2.8700E+04
3 30. 1.0000E-02 2.8700E+05
To amend a segment enter: Segment Number (space) Time (space) Flow<RET>
If no further changes enter <RET>
:3 40 0.01

Gas Burner Flow Curve Segments
Seg # Time [S] Flow [Kg/S] Qdot-nominal [KW]
1 0. 1.1658E-04 3.3457E+03
2 10. 1.0000E-03 2.8700E+04
3 40. 1.0000E-02 2.8700E+05
To amend a segment enter: Segment Number (space) Time (space) Flow<RET>
If no further changes enter <RET>
:

Review data for objects in another room? (Y/N)
:N

Menu for Input Data changes
-----

```

Figure 14. Option 6, Time - Gas Flow Rate data for a Gas Burner fire.

To change any data item, proceed as with Options 4 or 5: enter the item number (space) new value. Enter an extra <RET> to end the amendment process. As this is now familiar, it is not shown in Figure 13.

If the fire type was either Growing or Pool, or a non-burning object, this extra <RET> will finish consideration of this object. Questions similar to Option 5's will allow selection of another object for review or return to the main menu.

If the fire type was a Gas Burner (Type number 3) then, after completing all modifications to the material data, the additional data needed to define the Gas Burner's piecewise linear, time-flow curve is entered.

For this, the first information displayed is the current number of segments for the time-gas flow curve. (see Figure 14) Note that the **number of segments** is one less than the **number of points** defining the time-gas flow curve since a segment connects successive pairs of points. The current time-gas flow table is also presented. (The initial gas flow rate - at time = 0. - is set by the program.) Included on the table is the "nominal" power output. This is the heat which would be released if the fuel burned with 100% efficiency. (I.e., no account is taken of the heat release rate fraction, Chi.

The first query is to confirm or amend the number of flow curve segments. If the number is correct enter <RET> (or the number displayed). If a different number of segments is desired, enter the new number. The gas flow table is re-presented with the new number of entries. If the number of segments has been increased, the added items will be zero's. In the example of Figure 14, the number of segments was decreased by 1.

Next the table entries can be amended. Enter the table row number (space) time [sec.] (space) and gas flow rate [Kg/sec]. The times should increase with each successive entry, the gas flow rate can increase or decrease. After each amendment the table is re-presented. The last point of the shortened table in Figure 14 was amended. Entry of a <RET> without any numbers finishes the review of data for this object. The next questions are, like Options 5's, if another object's data is to be reviewed.

If the simulation continues beyond the time of the last gas flow point, a constant gas flow rate equal to that of the last point's value will be used.

```

Menu for Input Data Changes
-----
.
.
7) Number of vents and vent parameters
.
.
Please enter number of section to be changed
:7
Enter desired number of vents (Current value = 4, Max = 20)
:
Enter number of vent whose data is to be reviewed
:3
(Instruction text, not shown)
(Top of Vent data, not shown)
6) Side 2 is in room:      3
   on wall:              4
   In-flow coef.:        0.708
7)   Type of vent: 1 -> Free, 2 -> Forced
This is a free vent      1
100) COPY ALL ABOVE DATA FROM ANOTHER VENT
To change item, enter "Item (space) Value<RET>";
END WITH AN EXTRA <ret>
:7 2
:
(Amended Vent Data re-displayed)
(Top of Vent data, not shown)
7)   Type of vent: 1 -> Free, 2 -> Forced
This is a forced flow vent 2
Mixing option            1
100) COPY ALL ABOVE DATA FROM ANOTHER VENT
To change item, enter "Item (space) Value<RET>";
END WITH AN EXTRA <ret>
:
Current number of Segments for Vent Flow Rate Curve is 1
Note: Initial Vent Flow Pre-set to 0 M**3/sec
First entry is for the Second Point

```

Figure 15. Option 7, Number of vents and vent parameters

Option 7: Vents

Option 7: "Number of vents and vent parameters", like Option 4 for rooms, allows data from a single vent to be examined with the possibility of examining another vent's data or returning to the main menu. (see Figure 15) The current number of vents is presented and can be verified by a <RET> or a number entered. If this number differs from the current value, the number of vents will be changed accordingly, provided this number is no larger than 20 (the maximum number of vents for which the program is dimensioned) and allows at least one vent for each room (a requirement of the physical model). Provided an acceptable number is entered, the program then asks which vent's data is to be reviewed. The vent data is then presented and may be modified in the usual way: Item Number (space) New Value. Wall numbering for placement of vents (Item #5) is 0 front; 1, right side as viewed from the front; 2, rear; 3 left side. The opposite side of the vent is placed on the adjoining room's wall by the program and is displayed as a part of Item #6. This placement is often done badly (and should be put under the user's control). The vent flow coefficient is automatically calculated by the program based on the size of the vent relative to area of the wall in which it appears.

```

Current number of Segments for Vent Flow Rate Curve is 1
Note: Initial Vent Flow Pre-set to 0 M**3/sec
First Entry is for the Second Point

      NUMBER OF SEGMENTS FOR FORCED VENT CURVE = 1
      FORCED VENT FLOW CURVE SEGMENTS ARE
Seg #  TIME  FLOW [M**3/Sec]
  1     0.0  0.0000E+00
  2     0.0  0.0000E+00
Do you want to change number of segments in the Gas Flow Rate Curve?
To confirm present number enter <RET>; to change enter new number (Max 30)
:2

      FORCED VENT FLOW CURVE SEGMENTS ARE
Seg #  TIME  FLOW [M**3/Sec]
  1     0.0  0.0000E+00
  2     0.0  0.0000E+00
  3     0.0  0.0000E+00
To amend a segment enter: Segment Number (space) Time (space) Flow<RET>
(Positive flow is OUT)
If no further changes enter <RET>
:2 10. 6.

      (Amended Vent Flow data displayed)
      ("To amend a segment ..." message re-displayed)

:3 40. -6.

      FORCED VENT FLOW CURVE SEGMENTS ARE
Seg #  TIME  FLOW [M**3/Sec]
  1     0.0  0.0000E+00
  2    10.0  0.6000E+01
  3    40.0 -0.6000E+01
To amend a segment enter: Segment Number (space) Time (space) Flow<RET>
(Positive flow is OUT)
If no further changes enter <RET>
:

The current Temperature of incoming forced flow is 300.0 [K].
Enter <RET> to confirm; enter new value to change
:293.

If incoming cool air from a forced vent enters the
hot layer, should it mix, or plunge through to the
cool layer? 1 -> mix, 2 -> plunge, enter your choice
:1

Review data for another Vent? (Y/N)
:N

      Menu for Input Data changes
      -----

```

Figure 16. Forced Vent data input

There must be a "free" vent in every room. In addition, the free vents must be so placed that there is a direct, or continuous room-to-room path from every room to the outside. There may also be one or more "forced" vents in a room (up to the maximum number of 20 vents).

Vent data item #7 specifies the type of vent. If a forced vent is specified, the vent's volumetric flow is specified by a set of time-flow pairs defining a piece-wise linear curve. Note that a positive number corresponds to a flow out of the room, a negative number, flow supplied to the room. The temperature (enthalpy) of gas withdrawn is area averaged by the amount of upper or lower gas layer covering the vent opening. In-flowing gas enters at the temperature specified after all the time-flow pairs have been entered.

Forced vent flow data is presented and edited in much the same way as the Gas Burner flow data. The current number of flow curve segments and the current segment table is presented. The number of segments may then be confirmed or amended following which the table entries may be edited. Editing of the flow segments is terminated by a <RET> without accompanying numbers. (see Figure 16)

The temperature of in-flowing gas is asked (even if no inflow has been specified for any vent). This **need not be** the same temperature as the ambient.

What happens to gas supplied to the room will depend on details of the vent diffuser design and the location of the vent relative to the hot-cold gas layer interface. Little provision is made for this. The forced vent model was taken from a preliminary one developed by H. Mitler for FIRST.[8] The user is asked (mixing option) if the gas entering is to mix completely with the gas layer it encounters at the vent face, or (plunge option) should it plunge through the gas, seeking a layer more nearly of its own temperature. The model believes that the entering gas can do this without entraining room gas! In practice, the mixing option seems preferable especially where the supply vent is placed so low on the wall that it sees primarily gas of its own temperature. A number, 1 or 2, must be entered in answer to the mixing query.

After completing data entry for the mixing option, the user can move on to review data for another vent or return to the main menu.

Option 8: Wall Properties

Option: "Wall Properties", in a way similar to Option 4, allows amendment of data for a selected room with the option of either selecting another room's data for review or return to the main menu. Changes in data are introduced in the usual way.

Option 9: Non-indexed variables

Option 9: "Non-indexed variables" deals with parameters not associated with a specific room or object. As noted above, if the ambient (initial) temperature of the rooms has been changed, the ambient temperature selected here should be the same.

Option 10: Recap

Option 10: "Recap" presents the current input data, a screen at a time, in the format in which it will appear in the header of the calculation output file. A <RET> moves on to the next screen. Option 10 may be called at any time the main menu is displayed. It has the advantage that the data is presented in a different format than in the individual, data amendment screens. With the altered format, errors may be caught that have been missed on the data entry screens.

Option 11: Output options

Option 11: "Change output options" allows adjustment of

- 1) The frequency (in simulation seconds) of display of intermediate calculated results. The default is every 20 simulation seconds. No "pause" requiring a key-stroke to continue is included. With the T5100 computer this data moves across the screen so fast that having it presented is of marginal value. ("Scroll Lock" doesn't help on PC's.) Although this display is useful as a "progress report", it is probably just as well to set this number equal to the simulation length if your computer executes the simulation in close to real time.
- 2) The frequency (in simulation seconds) of writing calculated data to the output files. The default is 10 seconds.
- 3) The total simulation length (in simulation seconds).
- 4) The numerical tolerance. Changing this may have some effect on (a) the total time needed to complete a simulation, (b) the ability to get past a computation failure, (c) the accuracy of the computed results. It is hard to give firm rules for getting past a computational failure. Tightening the tolerance sometimes helps; other times loosening it does. Concerning accuracy of the results, remember that the physics included (and omitted) does not warrant more than one, perhaps an occasional two significant figures. However, it is desirable for the accuracy of the results to be set by the physics, not the numerics. If a loose tolerance is selected, it is advisable to try another, somewhat tighter tolerance to check that the numbers don't change significantly.
- 5) The minimum time step. Increasing this will mainly cause more frequent numerical failures without increasing computational speed. At each new time step the algorithm selects an optimal computational time step. Choosing a larger minimum time step has the effect of restricting its options.
- 6) The next output option concerns a one line message, normally displayed on the console after each time step. This is useful in following the calculation's progress, especially when progress is slow. These "progress reports" are not normally written to the disk file (it makes them pretty big!). To include the progress reports on both the console and disk file (useful in troublesome batch runs) enter 4. To display them only on the console, enter 5 or <RET> (the default). To suppress progress reports completely, enter 8. As noted below, there is about an 8% increase in computational time associated with sending this one line progress report to the console.

6: BATCH OPERATION

Batch operation requires that a batch command file, or similar file written in the Job Control Language of your machine, provide complete instructions for running H06. With MS DOS this is done via a special purpose file with the extension .BAT. Batch files can grow to considerable complexity. The example given here is for the simplest possible case. The batch file is one line. It has the name, REDIR.BAT. Its entire text is

```
H063OL<REDIR.INF
```

Typing, at the system prompt, C>REDIR.BAT<RET> directs the operating system to load (1) H063OL and, (2) to read program input data from the file REDIR.INF, rather than having the data typed from the console. REDIR.INF is an image of the information you would have to type-in from the console to run the program. Figure 17 shows the file REDIR.INF. It designates main menu Option 1 (Re-set input); directs that no files

be deleted, and that the file "REDIR.INP" be used to re-set the input data. Next it calls for main menu Option 0 and declines the use of Option 12. Finally it gives, as the new calculation output file, the name "REDIR.OUT" and opts to use a special plot output file with the name "REDIR.PLT". The last line is a "Control Z" indicating the end of file.

If several cases were to be run sequentially in batch mode, commands to do this would be added to the .BAT file. Each load and run of H063OL would require its own re-directed input file (analogous to REDIR.INF) and each of these would have to direct Option 1 to a corresponding, **previously saved**, input file and name a **new** output file (two if the plot file is opted for).

```

1
N
REDIR.INP
0
N
REDIR.OUT
Y
REDIR.PLT
^Z

```

Figure 17. File for re-directed input in batch mode.

Alternatively, if the input changes were simple enough, other menu options could be designated and the complete list of needed entries tabulated ending with Option 0 followed by an N.

Considerable care is needed in building a .BAT file and its associated files. As much as possible, complicated batch files should be avoided and, if used, well tested before they are turned loose. Since the computer will run unattended in batch mode, any error may go unobserved for a long time. Aside from the time lost, other troubles may result.

7: GENERAL COMMENTS

Often a programmed gas burner is used to represent the burning of a complex object or set of objects. Published data [12,13] provides guidance in choosing gas burner heat release rate - time curve to simulate objects, but often the user must select a burner diameter. The choice of diameter will influence the simulation result and is, therefore, important. A diameter close to that of the actual object is suggested. To illustrate the effect of object diameter, simulations, using the default plume algorithm, were run for a 2.4 x 3.6 x 2.4 M high room. Gas flow to the burner was set, in each case, to produce 250 KW at 10 seconds and 1 MW from 20 seconds until the end of the simulation. Room ventilation was via a door. The door size, 1M wide x 2M high, was chosen to produce near flash-over conditions but to avoid oxygen limited burning. Four burner diameters from 0.25 to 1.414 M were used. For each burner diameter, the semi-apex angle of the flame cone (Main Menu option 6, Physical/Chemical Parameters of Object, item #17) was chosen to yield the same flame height. The principal results (at 100 simulation seconds) are shown in Figure 18, additional results are in Table 1. In Table 1: Qstar is a flame parameter introduced by Cetegen et al.[14]; the variable TEPZR is the heat lost from the plume by radiation; FQPOR is the radiant flux at the surface of the burning object due to its fire plume.

Table 1: Gas Burner Diameter Study

Diameter [M]	Qstar	Flame Cone Angle [Deg]	Vent flow [KG/S]	TEPZR [KW]	FQPOR [KW]
0.250	28.83	8.21	0.954	10.9	28.23
0.500	5.10	16.10	1.115	39.4	44.22
1.000	0.90	30.00	1.265	140.2	62.96
1.414	0.38	39.23	1.316	265.5	72.38

To understand the differences between these simulations, some knowledge of the plume entrainment algorithm used is necessary. (Indeed, to use any computer fire simulation effectively, knowledge of its internal working is essential.) The default algorithm used in Harvard/NIST VI, TMSPLM, is similar to that proposed

by Thomas and Hinkley [15]. The fire is assumed to originate at a point ("virtual source") below the actual fire surface. The displacement of the virtual point source below the fire surface is such that the plume diameter, assumed increasing at a constant rate with height above the point source, will equal the fire diameter at the fire surface. Thus, as burner diameter increases the virtual source must be lowered. The plume entrainment at any height above the fire is calculated as the total entrainment from the virtual source to that height minus the entrainment from the virtual source to the fire surface. Because the plume entrains in a non-linear way, increasing the fire diameter results in an increased entrainment at a given height above the fire. Placing the fire in a room with a restricted ventilation opening results in a lower hot/cool gas layer interface than would be predicted for the same intensity fire stemming from a smaller diameter surface.[16] This is shown in Table 1 and Figure 18. The higher temperature of the hot layer for the smaller diameter fires is a result of the hot layer being thinner. (Higher interface location)

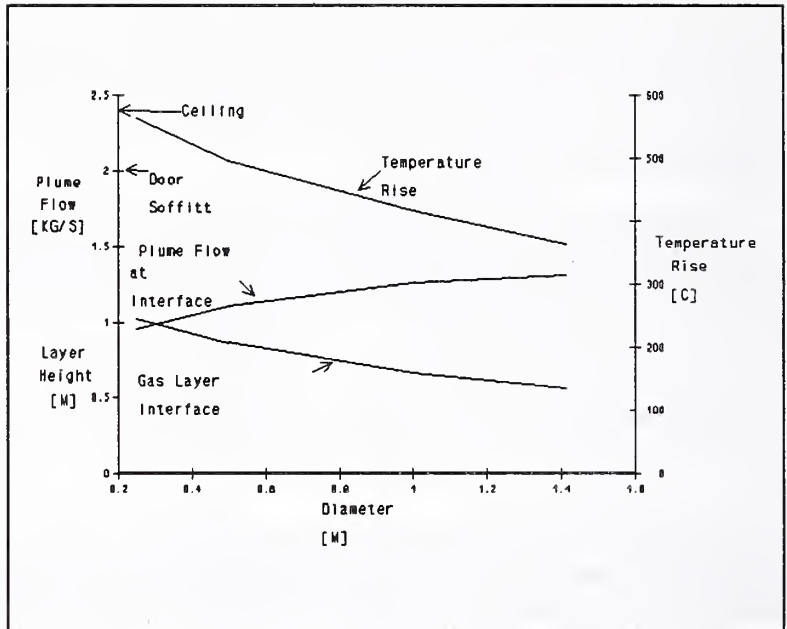


Figure 18. Effect of Burner Diameter on Hot-Cold Layer Interface height and Hot Layer Temperature.

Although, in these simulations, the flame heights were made equal by varying the flame cone apex angle, because of the varying flame diameter the flame volumes were not equal. In Harvard/NIST, flame emissivity is based on flame volume, hence the change in TEPZR and FQPOR. A similar change is found in the radiant flux from the plume to the walls.

Other plume entrainment models, if used in H06.3, yield different results. Indeed, the present default plume model may not be the best, current choice. Unpublished studies by the author of the effect of fire size on simulation predictions found significant differences as the plume algorithm used was varied among six available models. A conclusion from this was that **there is not sufficient experimental data presently available to choose a "best" plume entrainment model from those currently available.** Two of the six gave significantly different values from the others for extremes of fire size. No available data could say they were wrong, only noticeably different from the rest. The plume models presently available in H06.3 are the four roughly equivalent contenders.

Beside the default TMSPLM described above, four additional, optional plume algorithms have been added to H06.3: PTPLM, MPLUM, HESPLM and ZPLUM. PTPLM is a specialization of the default TMSPLM. It always places the point source at the fuel surface; it is otherwise similar to TMSPLM. PTPLM yields noticeably smaller entrainment which results in a thinner and hotter upper gas layer. MPLUM is a semi-empirical correlation of experimental data due to McCaffrey.[17] It entrains roughly the same amount as TMSPLM. Neither PTPLM nor MPLUM have an explicit dependence on fire diameter. HESPLM is a semi-empirical correlation of experimental data by G. Heskestad [24]. ZPLUM is a semi-empirical correlation of experimental data due to Cetegen, Kubota and Zukoski.[14] ZPLUM gives somewhat less entrainment than PTPLM. All four are reasonable representations of the existing plume data. The extent to which they differ is symptomatic of our ignorance.

For the gas burner in a single room, discussed above, the effect of varying the plume algorithm is illustrated in figure 19 and Table 2. Table 2 also shows that, although the McCaffrey and Point Source plumes have no explicit dependence on fire diameter, the overall simulation result using these plume algorithms varies with burner diameter due to other diameter dependent parts of the simulation. In figure 19 the mass entrainment versus height of the five plume models available in H063 is plotted. On the same figure is the vent flow versus hot-cold interface height for a 2 m high vent.

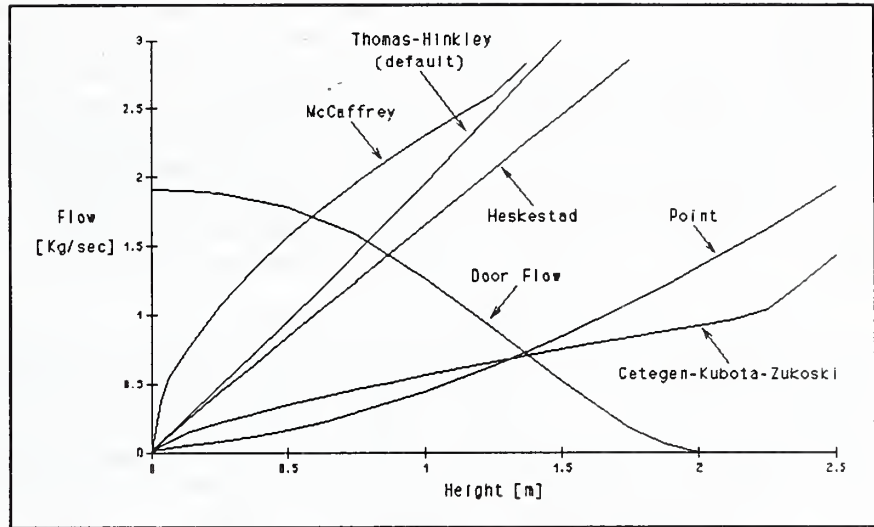


Figure 19: Plume flow versus height for various plume algorithms and vent flow versus height

The intersection of each plume entrainment curve with the vent flow curve gives a steady state "operating point" for that plume paired with the vent [16].

Table 2: Comparison of Plume Algorithms

Algorithm	Burner Diam. [M]	Upper layer temp. rise [C]	Layer thickness [M]	Plume flow at interface [Kg/sec]
McCaffrey	0.5	497	1.54	1.11
Virtual Source		471	1.53	1.11
Point Source		693	1.12	0.64
Cetegen-Kubota-Zukoski		862.	0.91	0.37
McCaffrey	1.0	383.	2.00	1.38
Virtual Source		402.	1.73	1.26
Point Source		636.	1.12	0.64
Cetegen-Kubota-Zukoski		630.	1.13	0.65

A simple explanation of these results is defeated by the effects of radiation. However, if the larger burner diameter data of Table 2 is examined, its lower hot layer temperatures somewhat reduce the radiation effects. Note that, although there is a 25% variation about the average upper layer temperature rise, there is only a 13% variation in the product of upper layer temperature rise and a plume flow at the layer interface. Without radiation losses, this product would be expected to be constant. The difference in predicted behavior is strongly coupled to the differing entrainment of the various plume algorithms. Those entraining less (C-K-Z and Point Source) produce higher layer temperatures and thinner layers.

It is possible that the Harvard simulation, using the default (Virtual Source) plume algorithm, underestimates upper layer gas temperature and over-estimates hot gas layer depth.[7] Under estimation of upper gas layer temperature could either be due to the choice of plume model or it could equally well be due to the

omission of mixing at the vents, floor heating, and the associated heating of the lower gas layer.[10] Unpublished simulations with FIRST, which, at that time, included vent mixing and related effects but did not include floor heating, showed better agreement between Steckler's vent flow data [18] using the McCaffrey and (default) Virtual Source algorithms than when using the Point Source or Cetegen-Kubota-Zukoski algorithms. This was true even after corrections were made for thermally induced walls flows.[19] Further, Hinkley, using his own model for ceiling venting which includes the Thomas-Hinkley Virtual Source plume algorithm [15], finds satisfactory agreement with experimental data.[20] On the other hand, Tanaka found that data from fires in a large enclosure [21] were well modeled by his BRI2 simulation [22] using the Cetegen-Kubota-Zukoski plume. In an earlier version (BRI1 [23]) he had used the McCaffrey plume.

An aside to the above simulations concerns the choice of flame radiation calculation method (Main Menu option 3, item #2). As a part of the above plume algorithm comparisons, simulations were run with both the "Standard" method and the "Modak" method. The calculation times (to 200 simulation seconds) were interesting, see Table 3. The relative calculation times clearly show that the much longer times using the Modak method were not solely related to the more complex emissivity calculation. Most of the difference was related to emissivity calculation induced changes in the number of iterations, the frequency of re-calculation of the Jacobian Matrix, and the size of the successful time steps.

Table 3: Calculation Times for Two Emissivity Calculation Methods

Method	Burner Diameter	Time
Modak	1.414	4:08:19.66
	1.000	0:19:43.06
Standard	1.414	0:04:22.40
	1.000	0:04:08.78

On the Toshiba T5100 (80386/7 running at 16 MH) H063OL compiled with Ryan McFarland FORTRAN version 2.43 executes the standard case in 11:32.19. If the "progress report" at each time step (see Option 11) is not used, it takes 10:53.74. If the Lahey F77L FORTRAN version 3.00 compiler is used, the run times are about 40% longer (16:2.51 versus 11:32.19). The standard case is the same physical set-up as the standard case in FIRST which executes in 4:01.15. (Both take about 9 times longer on the Zenith 120 - 8086/7 running at 5 MH.) These times suggest that the math pack of FIRST is superior to that of H06. However, this is not a fair comparison. It is not clear that the math of FIRST could treat the more complex problem posed by the multi-room geometries of H06.

Although the above shows a clear speed advantage to the Ryan McFarland compiler, the Lahey compiler has been used for the distribution disks. The reason is compiled size. Using the identical source code, linker and link file, the Ryan McFarland executable code required 587 KB of memory while the Lahey version needed only 565 KB. The difference in linked size appears to be related to the way data types are classified by the compiler. This affects the way the data blocks are treated in the overlay process. In order to be able to add enrichments to the code, it was decided to use the Lahey compiler, accepting the execution speed penalty. Comparison of numerical results from the T5100 (32 bit words) with runs on the CDC-855 (60 bit words) shows better agreement with Lahey F77L than when using Ryan McFarland.

Input data sets have occasionally been found which, although continuing to advance, use very small time steps. One nine room case took over 8 hours on the T5100. The calculation proceeded with good speed through most of the simulation. At two times (when nothing obvious was happening to cause trouble) it slowed and almost stopped. After struggling for a long time to advance about one simulation second, it

worked through its difficulty and picked up speed to a reasonable pace. There is no obvious fix for these in-obvious hang-ups.

It may be a year or more before a successor - in flexibility and capability - to the Harvard Fire Simulations is available. Until then, the unique capabilities of these pioneering computer codes should be kept available. This manual is a partial effort to maintain H06.

Please send comments on this manual and/or the use of H06 to:

Dr. John A. Rockett
4701 Alton Pl., N.W.
Washington D.C., 20016-2041
USA

To the extent that time, ability and resources permit, an effort will be made to maintain and improve H06.

8: SEPARATE PROGRAM MASBANK

A separate program, MASBANK, has been written for maintenance of the Harvard/NIST FIRST materials property data base. The same data files can be used with H06.3. MASBANK's use is fully described in [8] pp 66-73. MASBANK has been written to be largely self coaching. It is loaded by typing

```
C>MASBANK
```

A header message is printed and a <CR> (In this guide denoted <RET>) solicited. After this, the program searches for two files: MASNAM and MASDAT. Files with these names must exist on the current directory, even though empty, if the program is to run. If they are found, MASBANK prints a list (up to three columns wide) of the material names, if any, found. MASBANK is currently dimensioned for up to 50 materials. The program solicits another <CR> and then prints a menu of six options:

- A for Add a material to the data base
- C for Change material data already in the data base
- D for Delete a material
- E for Exit to the system
- S for Sort the data base alphabetically
- V for View the data for a material in the data base.

In Adding a new material to the data base its name is first asked. This is immediately typed back and a confirmation of correctness asked. If this query is answered "N" the name entry is re-asked. Next a single, 75 character comment line may be entered. This is intended to supply, for example, a reference to information on the source of the data. The comment is shown and a confirmation of correctness asked. Again, "Y" accepts the comment as made, "N" loops back to allow re-entry of the comment. 20 material properties values are next requested, one at a time. At each step the value just entered is presented along with the material's HARVARD code FORTRAN name, its common property name, and its units. A similar line of information for the next property is presented immediately below. For a new material being added to the data base the property values will all be 0.'s of course. When all 20 values have been entered, the data is recapitulated and the user asked "ANY CORRECTIONS? AN (Y/N)". If corrections are to be made ("Y" answer), they are entered as

```
Item number (comma) new value<RET> `
```

To end the corrections enter

0 (comma) 0<RET>

Following the 0,0 entry the table of data is again displayed. The correction question is re-asked and can, hopefully, be answered "N".

The new list of material names in the data base is presented followed by the six item menu. If errors have been made, option C will permit selective re-entry of data.

MASBANK is easy enough to use by just following the directions. The problem is finding numbers to enter for materials of interest. [8] Appendix A and Table 1 give numbers for four materials. [9] suggests a way of educating a guess for AFIRE (MASBANK item #16). [10] discusses TBURN (MASBANK item #19). The various publications of A. Tewarson of Factory Mutual are very useful. [11] is one of these, there are others more recent, not all as directly useful for H06. Gross [12] has data for a number of complete objects. His report is directed primarily at NIST (NBS) FAST, rather than FIRST or H06, but can be useful.

9: SEPARATE PROGRAM PREPLOT

As mentioned in the discussion of Option 0, PREPLOT is a special purpose interface program for use between H063OL and a user's plot program. It was originally written with LOTUS 1-2-3 as the target plot package. LOTUS macros must be written to produce the plots. PREPLOT's output is also usable with MS CHART. MS CHART has a pronounced predisposition to label charts its own way; it doesn't readily read some of the captions as formatted by PREPLOT. An alternate set of PREPLOT output formats for use with CHART, have not, however, been written. Presumably other plot packages could use PREPLOT's output, but, again, with some possible awkwardness. No testing of alternate plot packages, other than MS CHART, has been attempted.

Each execution of PREPLOT generates one plot file. A single variable is tabulated against Time. The same variable may appear in each of the several rooms. In this case, the value for each, or selected rooms may be included.

To run PREPLOT, assuming that its .EXE file has been transferred with the other files from the disks supplied, enter

```
C>PREPLOT
```

After the program has loaded, it asks (see Figure 20)

```
Enter name of input data file  
:
```

The name entered at the ":" prompt should be that of the second of the two output files names requested at the start of the relevant H063OL run: in the example of Figure 6, USEDEMO.PLT. PREPLOT attempts to OPEN, and read from an existing file with the name supplied. If it can't, it repeats the "Enter name ..." question. If it is successful it prints the reminder

```
Plots are against "Time".
```

This is immediately followed by a list of 14 items: Number, H06 FORTRAN variable name, a variable description and units. (see Figure 20) At the end of the list is the message:

Enter the number of the variable you want to plot against "Time".
If none, enter "0"
:

If, for example, you want to plot upper layer gas temperature versus time for each of the rooms, the response would be "8" (as in Figure 20). If, on the other hand, it was intended to plot one of the Object, Vent or Wall variables against Time, an answer "0" would produce a list of 14 Object variables (numbered 15-28). A similar question would be asked. A number between 15 and 28 would select the corresponding Object variable. An answer "0" would result in a list of 3 Vent variables, numbered 29-31. Another "0" answer would produce the final list of 4 Wall variables, numbered 32-35.

```
C>PREPLOT
Enter name of input data file
:USEDEMO.PLT

Plots are against "Time".

The following ROOM variables are output by Harvard VI:

1) TELZR: Rate of change of energy of hot layer due to radiation [Kw]
2) TELZD: Rate of change of energy of hot layer due to convection [Kw]
.
8) ZKLZZ: Temperature of hot layer [K]
.
14) ZPRZZ: Pressure at floor level in room [m ambient air]

Enter the number of the variable you want to plot against "Time".
If none, enter "0".
:8
```

Figure 20. Starting PREPLOT

Having selected the variable to be plotted, and depending on the type of variable selected, additional questions are asked. Note that PREPLOT learns from the input data file how many rooms, objects per room, vents and walls there are for the H063 run being processed. In the case of a Room variable, the next question is "Enter room number". A room whose data is to be plotted is entered. This is followed by the query: "Do you want to include another room? (Y/N)". (see Figure 20) This pair of questions is continued until either the second is answered "N" or the total number of rooms entered exceeds the available rooms. PREPLOT then moves on to the next series of questions.

For two types of variables, options are available. If a temperature has been selected, it may be left in the H06 form - Kelvin, or converted to degrees Celsius, or to temperature rise above the starting (ambient) value. If the variable selected was the layer depth, it may be plotted either as depth or converted to height above the floor. (see Figure 21)

The next query is for the name of the output file to be used for this set of plot data. This is the file which will provide input to your plot package.

The next set of queries provide label information for the plot. This information may not be directly useful, depending on your plot package, but an entry

```
Enter room number
:1
Do you want to include another room? (Y/N)
:Y
Enter room number
:2
.
Number of rooms already specified = number
available.
The variable chosen is an absolute temperature.
To leave as Kelvin enter "0"
To convert this to Celsius enter "1"
To convert this to temperature rise enter "2"
:1
Enter name of output file
:ZKLZZ.PLT
```

Figure 21. Use of PREPLOT when a room variable has been specified.

which identifies the H063 run is suggested. For MS CHART most of this, as currently formatted by PREPLOT, is not useable. The first query of this set expects two lines, 40 characters each, of chart caption. They are written, enclosed in double quotes (") by PREPLOT, in A42 format, as the first two lines of the output file. The ordinate label is automatically entered as "Time [sec.]". The next query is for the abscissa label. The ordinate and abscissa labels are written as the third line of the output file, again enclosed in double quotes by PREPLOT, in A42, A32 format. If, as in the example of Figures 20 and 21, a room variable has been selected, the fourth line of the output file will be automatically written: "Time [S]", "Room (1)", "Room (2)", etc.

The final query is "Do you want this to be a semi-log plot?" The answer, for H06, usually has to be "N". Zero's or sign changes can't be treated. If a semi-log plot is requested, the data is scanned for possible trouble and, if it is not suitable, PREPLOT prints the message:

```
A Semi-log plot is inappropriate.
The variable list contains both positive and negative
numbers or null values.
A normal plot will be produced.
```

If the data is all negative without nulls, the absolute value is used to compute the logs. A message to this effect is presented.

The data is then written in 1P11E9.2 format. Note that, as currently written, PREPLOT will only write time plus 10 variables on a single line of the output file. This is enough for the maximum 10 rooms for which H063 is dimensioned, but not for the 20 vents. (A possible problem if you wanted more than 10 curves on a single plot!) If, for example, H06 was run for 600 simulation seconds with disk output every 10 seconds, there would be 61 lines of data. These would be PREPLOT output file lines 5 through 65 (line 5 being the "zero time" values). Figure 22 shows the top 10 lines of the PREPLOT output file used to prepare Figure 23, below.

```
"THREE.INP                                "
"UPPER GAS LAYER TEMPERATURE              "
"Time [sec.]"                             "ZKLZZ [C]                                "
"Time [S]"  "Room (1)"  "Room (2)"  "Room (3)"

0.00E-01 2.68E+01 2.68E+01 2.68E+01
1.00E+01 2.88E+01 2.69E+01 2.69E+01
2.00E+01 2.91E+01 2.74E+01 2.71E+01
3.00E+01 2.96E+01 2.78E+01 2.75E+01
4.00E+01 3.02E+01 2.83E+01 2.79E+01
5.00E+01 3.09E+01 2.87E+01 2.82E+01
```

Figure 23 is a plot, prepared from PREPLOT output, of upper gas layer temperature for a three room case. Temperature has been converted to Celsius. The plot was prepared with MS CHART. Ordinate values were taken from column 1 of the output file, beginning at line 5 (see Figure 22). The three sets of temperature data were from columns 2 through 4. Captions were re-entered, not read from the PREPLOT output.

Figure 22. Top of PREPLOT output file used to prepare Figure 22.

10: USE WITH NON-PC COMPUTERS AND NOTES ON RE-COMPILING H06.3

All versions of the Harvard fire simulation were written in ANSI FORTRAN VII. Nevertheless, several features of H06.3 make it non-transportable between machines. These are

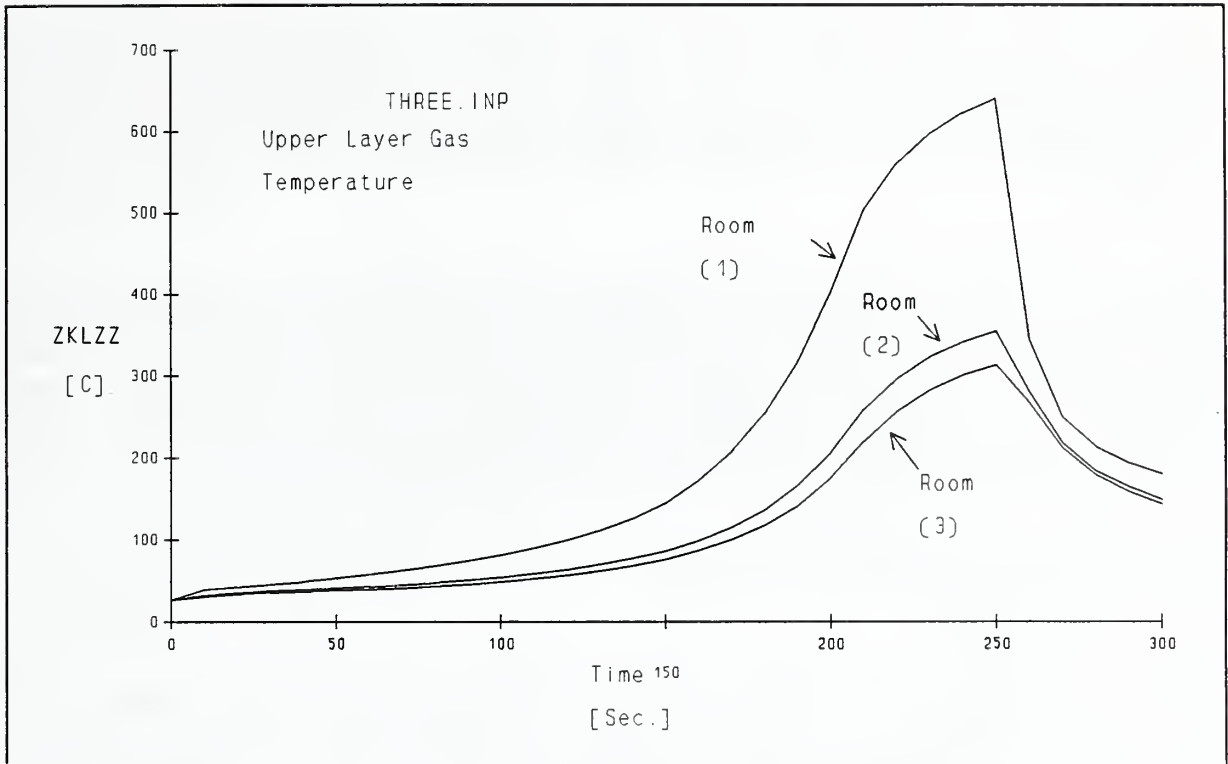


Figure 23. Plot prepared with MS CHART from PREPLOT output.

- 1) OPEN statements in the MAIN PROGRAM, and in SUBROUTINES BINP and DBANK. Although the OPEN statements are standard FORTRAN, those in MAIN refer to the user's console and printer. There appears to be no standard terminology for these "logical units". Thus the console is named "C:" for the CONCURRENT-3252 and "CON" for MS DOS machines. Other forms are necessary for use with the CDC-855/Cyber 205. Similarly, with OPEN's for output to the printer, the reference may have to be changed. Further, the CDC does not allow file names with extensions; this requires changes to the OPEN's in MAIN which include a file allocation. In short, you may have to adjust these OPEN's to your own computer.
- 2) RTIME, a subroutine which reports simulation execution time, accesses the system clock. Subroutines to access the system clock are not a part of the standard FORTRAN VII library. In H06.1-2, RTIME was disabled. H06.3 uses RTIME in a version which works with MS DOS (compiled with Lahey F77L 3.00). The original DEC VAX-11/780 version of RTIME, together with an MS DOS version for use with the Ryan McFarland FORTRAN version 2.43 compiler and a "dummy" version for use with other computers or compilers, has been placed in file DTIME.FOR. The appropriate version for your computer and/or compiler should be substituted for the RTIME in file DBLES5A.FOR or the version there disabled.

Linking

As originally developed, H06 was two separate programs. BINP solicited the user's input and constructed a binary input file. This binary file was read by the second program, DBLES, and executed. The binary

transfer was slow and the binary file, as written, was machine dependent and could not be made machine independent (at least by us). One of the changes between H06 and H06.1 was the elimination of the binary file. This was done in two ways. In one, the "Rockett version" of H06.1, the need for the binary transfer file was eliminated by combining the input and execution programs. In the other, "Morita version", the binary file was re-written as a formatted file. The Morita version of H06.1 also included an extensive diagnostic package. The resultant program is too big to use on an MS DOS PC (640 KB limit). Because of its smaller size, relative simplicity and ease of use, further development has been based on the single program (Rockett) version of H06.1.

Until the re-dimensioning of H06.2, combining the input and execution parts of H06 presented no problem as the total package would fit on a standard PC. H06.3 (H063OL) still fits on a standard PC although it is large, about 565 KB in the overlay version. Obviously there is little room left for the addition of more fire physics/chemistry if the program is to remain below the "640 KB limit" including OS overhead. In spite of the considerable growth to the input section, the execution section remains the largest piece of the program. The operating system limit on enrichment will come into play whether or not the program is split into two. (In fact, splitting the program into two results in a larger execution program than the present overlay version. This is because subroutines, now needed only in the input section of H06.3, must be duplicated in the execution program to select and read the input data file.) The only alternative to this program growth limitation will be OS-2 (not available as this is written) or UNIX (which involves a larger operating system overhead and is not widely used by potential H06 users). As most potential users probably still use MS DOS; H06.3 has been tailored to it.

MS DOS 3.2 Link does not support overlays. The Lahey F77L subset of "Plink86 plus" should be able to handle the overlay used in H06.3, the full "Plink86 plus" can. FORTRAN compilers or Linkers for use with the Mackintosh family of computers have not been investigated. Super-mini and main frame users should have no difficulty finding suitable software on their system and probably need not bother with overlays anyway.

The essential features of the overlay are the use of a root section (always present in memory) and two overlay sections: the input section which is loaded first, used first, and never returned to, and the execution section which is loaded into memory on top of the input section after finishing the input operation. The root section - MAIN program and BLOCK DATA (file H063.FOR) - transfer information (around the overlay) between the two sections. For technical reasons involving the (strange) memory addressing system adopted by MS-DOS and the way PLINK86 stores variables local to an overlay section, the overlay is not very efficient - the overlay version is considerably larger than one might expect. As a result, returning to the original form for H06: two separate programs, may eventually be forced.

A subroutine map of H06.3 is provided in Appendix I. This indicates the files to be included in the root and two overlay sub-sections.

11: EXAMPLE: INSTALLATION ON A NON-MS DOS COMPUTER, THE CONCURRENT-3252

The CONCURRENT FORTRAN's treatment of very small numbers can lead to arithmetic faults. Many FORTRANs automatically convert any number so small that it would cause an underflow fault to 0.0. This is not automatic with the CONCURRENT. Instead, they include a non-standard FORTRAN library subroutine, AFLTOF, which, if called, makes this conversion. A "CALL AFLTOF(0)" had to be inserted as the first executable statement of SUBROUTINE ALDIFF. While unique to the CONCURRENT, this is an example of the sort of problem that might arise in installation on a non-MS DOS machine.

Other needed coding changes have been alluded to above in Section 9. CONCURRENT compatible OPEN statements in the MAIN program (file H063.FOR) either had to be activated or their equivalent included in the Job Control Language (JCL) file used to load and start the program. Not activating these

made construction of Batch Operation files simpler. An equivalent function is included in the JCL file. Subroutine RTIME in file DBLES5A.FOR had to be disabled. It was replaced by the "dummy" version from file DTIME. Nearly the same information can be obtained by adding a "D A", i.e. "Display Accounts" to the JCL file. On the 3252, this causes the CPU Time, Roll Time, and Wait Time for the H063 task to be listed at the end of the calculation. Obviously, on a multi-tasking machine, elapsed time, as reported by RTIME, is not a very significant measure of execution efficiency since it depends strongly on the work load current at the time H063 is run.

In order that the Main Menu, and other input screens, appear correctly, it was also necessary to precede the load instruction with a "PREVENT PROMPT". This inhibits the system prompt. Without this, the system inserted an extra line feed between each line written to the Console screen by the program. This rolled needed information off the top of the screen before it could be read. In addition, the extra characters which constituted the prompt caused some text lines to be too long. These lines were folded, making them difficult to read. Occasionally this also induced rolling the top line off the screen. Even with the PREVENT PROMPT command, the operating system supplied its own prompt when data was needed so that the ":" of the program was augmented, on the next line, with a ">". I.e., when data was needed the screen looked like this:

```
:
>Data entry
```

Although this was annoying, it didn't effect operation. The ":"s in CONCURRENT version of the program code were not deleted!

The CONCURRENT has all kinds of memory so overlays were not needed to link H063.

```
PREVENT PROMPT
LOAD H063OL
$IFNN @1; AS 5,@1; $ENDC
$IF NU @1; AS 5,C:;
$ENDC
AS 6,C:
ST
D A
$EXIT
```

Figure 24 shows the JCL file used on the CONCURRENT. Its name is H063.CSS. (The CONCURRENT Job Control Language is similar, but not the same as IBM's. The names of CONCURRENT JCL files must have the extension .CSS - for Command Substitution System). Of course this file would not be usable on another manufacturer's machine, but it illustrates an approach to operation on a non-MS DOS machine. Interactive program execution is initiated by typing the name of the .CSS file without the .CSS extension. I.e.,

Figure 24. Concurrent 3252 File H063.CSS used to load and run H063OL.

H063

For Batch operation, using the batch input file REDIR.INF, the entry is

H063 REDIR.INF

As mentioned above, the first line of H063.CSS inhibits the operating system prompt. (see Figure 24) The second line loads the linked program into memory. The next line assigns FORTRAN unit 5 (logical unit 5) to the file whose name is entered on the same line as the .CSS file name. In this case REDIR.INF will be assigned to Logical Unit 5. If there had been no file named (as would be the case for interactive mode), the next (fourth) line directs that logical unit 5 be assigned to the console, "C:". Logical unit 6 is always assigned to the console. ST starts the execution and D A lists accounting information when the simulation finishes (as mentioned above). \$EXIT tells the system that no more commands will be found in this file. The file REDIR.INF is identical to the one used with MS DOS machines.

12: OTHER CHANGES IN H06.3

One of the most frequent criticisms of H06 was that input needed to change the room, object or vent data required adherence to a specific format. A set of four SUBROUTINES has been borrowed from Walter Jones "FAST-17": READBF, SSTRING, CONVRT, and DATYPE. They are used to read input data. (They were slightly modified to suit H06 and are in file READDATA.FOR.) By using these, the only requirement is that a delimiter, either a space or a comma, separate the items. The same effect could have been obtained with List Directed Input except that, occasionally, a character string, rather than a number, is needed. With List Directed Input this has to be placed in quotes; with READBF it does not.

The flame extinction coefficient, ZUFZZ, has been made an input item. It may be changed using input option 6, Physical/Chemical properties of objects. In addition, the default value has been changed from 1.55 [1/m] to 1.28. This is the same as the current value used in FIRST [8]. 1.55 is appropriate for extremely heavy sooting flames. 1.28 is better for PU foam or PMMA. Smaller values should be used for Natural Gas (about 0.5) or Propane (about 0.85). User variable ZUFZZ is available in H06.3 versions dated after 3 August, 1990.

There was a "glitch" in H06.2 when a Growing Fire was specified. If the initial fire radius was increased (from the default value of 0.037 m) to more than 0.04051 m, no starting values could be found. It turned out that this problem was present in all the earlier versions of H06 which were still available; it is not known how long the problem has existed, but it seems to be of long standing. To permit larger variation of the initial Growing Fire radius, a new subroutine, RSTART, was introduced. If an initial radius larger than 0.037 m is specified, NWSTAT uses 0.037 m, thus allowing starting values to be obtained. But ERFDE (which calculates the rate of change of fire radius), rather than calling SPREAD, initially calls RSTART. This returns a rate of spread which will cause the fire to grow from 0.037 m to the specified initial radius during the first simulation second. After 1 second, SPREAD (the usual subroutine) is used to calculate the rate of fire growth. This has worked for tests to more than ten times the default, 0.037 m, initial fire radius. The fix can't be guaranteed for all cases, but has caused no trouble so far.

In DBLES, CALL SETSYS has been added before the first CALL NEWSTP. This is needed to properly initialize the array size variables. Also, in DBLES, the array JAC is initialized to 0. These initializations were not done in the original Harvard VI nor subsequent versions.

In EVALFP, where subroutine CNVW1 (which returns the convective heat flux at the wall surface) is called for the outside of a wall, an index error occurred. This too is an error at least since H06.1 and has been corrected.

In SETSYS, two compound IF clauses [IF(A .AND. B) THEN] caused trouble with some optimizing compilers which parse the entire line before acting on it. Trouble arises because, if A is false, B is undefined. These were broken into a pair of separate, nested IF clauses so that the second variable will not be examined if it is undefined. This error is not uncommon in FORTRAN codes.

Subroutine CORCON, which is mentioned by Gahm [1] and appears as one of the H06 files in Volume 2 of his report, was never referenced. It has been a part of the file structure of all H06's, still unused; it has now been dropped from the program.

13: REFERENCES

- [1] Gahm, J B, "Computer Fire Code VI", NBS-GCR-83-4451 (2 volumes), US Dept. of Comm, Nat Bur of Stds, Gaithersburg, MD 20899
- [2] Ramsdell, J, "Variable Elimination in the Home Fire Code", ASME/AIChE Nat Heat Trans Conf, paper 81-HT-3, Milwaukee, Wisc, Aug 2-5, 1981
- [3] Mitler, H, "The Physical Basis for the Harvard Computer Fire Code", Home Fire Project Tech Rpt No 34, Div of App Sc, Harvard Univ, Cambridge MA, Oct 1978
- [4] Mitler, H and Emmons, H W, "Documentation for CFC V, The Fifth Harvard Computer Fire Code", Home Fire Project Tech Rpt No 45, Div of App Sc, Harvard Univ, Cambridge MA Oct 1981
- [5] Rockett J A and Morita, M, "The NBS/Harvard VI Multi-room Fire Simulation", Fire Sc and Tech, Vol 5, No 2, 1985, (159-164) Tokyo, Japan
- [6] Steele, S. and Rockett, J A, "Application of the Harvard Multi-Room Fire Simulation where Forced Ventilation is Important", Second International Symposium on Fire Safety Science, Tokyo, Japan, June 13-17, 1988, Hemisphere Publishing Co., N.Y.
- [7] Rockett, J A, Morita, M and Cooper, L Y, "Comparisons of NBS/Harvard VI Simulations and Full-Scale Multi-room Fire Test Data", NBSIR 87-3567, Nat Bur of Stds, Gaithersburg MD 20899, 1987
- [8] Mitler, H.E. and Rockett, J.A., "User's Guide to FIRST, A Comprehensive Single-Room Fire Model", NBSIR 87-3595, Nat Bur of Stds, Gaithersburg MD 20899, 1987
- [9] Rockett, J A, "Fire Growth in Combat Ships", NBSIR 86-3451, Nat Bur of Stds, Gaithersburg MD 20899, 1986
- [10] Rockett, J A, "Modeling of NBS Mattress Tests with the Harvard Mark V Fire Simulation", Fire and Materials, Vol 6, No 2, 1982, pp 80-95
- [11] Tewarson, A, "Materials Compendium of Fire Property Data", Products Research Committee, Soc. of the Plastics Industries, New York, NY, February 1980
- [12] Gross, D, "Data Sources for Parameters Used in Predictive Modeling of Fire Growth and Smoke Spread", NBSIR 85-3223, Nat Bur of Stds, Gaithersburg MD 20899, 1985
- [13] Babrauskas, V., "Upholstered Furniture Heat Release Rates: Measurements and Estimation", Journal of Fire Sciences, Vol 1, 9-32, January/February 1983
- [14] Cetegen, B.M., Zukoski, E.E. and Kubota, T., "Entrainment and Flame Geometry of Fire Plumes", California Institute of Technology, Daniel and Florence Guggenheim Jet Propulsion Center, August 1982
- [15] Thomas, P.H. and Hinkley, P.L., "Design of roof-venting systems for single-storey buildings", Fire Research Technical Paper No. 10, Department of Scientific and Industrial Research and Fire Officers' Committee, Joint Fire Research Organization, London: Her Majesty's Stationary Office, 1964
- [16] Rockett, J.A., "Fire Induced Gas Flow in an Enclosure", Combustion Science and Technology, Vol 12, Gordon and Breach, Science Publishers Ltd, 1976

- [17] McCaffrey, B.J., "Purely Buoyant Diffusion Flames: Some Experimental Results", NBSIR 79-1910, Nat Bur of Stds, Gaithersburg MD 20899, 1979
- [19] Steckler, K.D., "Fire Induced Flows Through Room Openings - Flow Coefficients", NBSIR 83-2901, Nat Bur of Stds, Gaithersburg MD 20899, 1984
- [20] Hinkley, P.L., "Comparison of an Established Method of Calculation of Smoke Filling of Large Scale Spaces with Recent Experiments", Fire Science and Technology Vol 8, No 1 (1-8), Center for Fire Science and Technology, Science University of Tokyo, Tokyo, Japan, 1988
- [21] Tanaka, T., Yamana, T., "Smoke Control in Large Scale Spaces: Part 1: Analytic Theories for Simple Smoke Control Problems; Part 2: Smoke Control Experiments in a Large Scale Space", Fire Science and Technology, Vol 5, No 1, Sept. 30, 1985
- [22] Nakamura, K. and Tanaka, T., "Predicting Capacity of a Multiroom Fire Model", Fire Safety Science Proceedings of the Second International Symposium, pp 907-916, Hemisphere Publishing, Washington, D.C., USA, 1989
- [23] Tanaka, T., "A Model of Multiroom Fire Spread", NBSIR 83-2718, Nat Bur of Stds, Gaithersburg MD 20899, 1984
- [24] Heskestad, G., "Fire Plume Air Entrainment According to Two Competing Assumptions", Factory Mutual Research Corp. RC86-TP-2, 1151 Boston-Providence Turnpike, Norwood MA 02062, August 1986

APPENDIX I: H063 SUBROUTINE MAP

[Root Section]

H06 (File H063.FOR)

RECAP (File DISPLAY.FOR, part of overlay Input Section)

BLOCK DATA (File H063.FOR)

[Input Section]

BINP (File BINP4.FOR)

MENU (File BINP4.FOR)

READBF (File READDAT.FOR)

READEM (File READWRIT.FOR)

NAMES (File DISPLAY.FOR)

XTITLE (File BINP4.FOR)

READBF

SELSUB (File BINP4.FOR)

READBF

ROOMS (File INPUT4.FOR)

READBF

DISP (File DISPLAY.FOR)

IASK02 (File DISPLAY.FOR)

DSPASK (File DISPLAY.FOR)

IASK02

BRFLOW (File DISPLAY.FOR)

READBF

READBF

READBF

COPINP (File BINP4.FOR)

OBJG (File INPUT4.FOR)

DISP

COPINP

OPJP (File INPUT4.FOR)

DBANK (File DBANK.FOR)

DISP

COPINP

BRFLOW

VENTS (File INPUT4.FOR)

DISP

COPINP

ORIFIC (File BINP4.FOR)

VNTINP (File DISPLAY.FOR)

WALLP (File INPUT4.FOR)

DISP

COPINP

NONIND (File INPUT4.FOR)

DISP

RECAP (File DISPLAY.FOR)

OPTION (File INPUT4.FOR)

READBF

WRITEM (File READWRIT.FOR)

INIT0 (File BINP4.FOR)

[End of Input Section]

[Calculation Section]

DBLES (File DBLES.FOR)
 SETSYS (File SETSYS.FOR)
 RTIME (File DBLES5A.FOR)
 PLTINIT (entry point: PLTOUT, File PLTOUT.FOR)
 NWSTP (File DBLES3G.FOR)
 UNPACK (File DBLES3G.FOR)
 TMPW1I (entry point: TMPW1, File DBLES5A.FOR)
 TMPO2I (entry point: TMPO2, File PHYSUB2.FOR)
 SAVVAR (File WRIT.FOR)
 NWSTAT (File NWSTAT.FOR)
 UNPACK
 SETSYS (File SETSYS.FOR)
 PUTPTR (File DBLES2D.FOR)
 PACKX (File DBLES3G.FOR)
 FUNCT (File DBLES1F.FOR)
 UNPACK
 EVALFP (File EVALFP.FOR)
 RDNL
 CNVL1
 TMSL
 POWL
 LAYDPT
 TMPL
 OXYCNC
 DXDCNC
 MNXCNC
 SUTCNC
 WTRCNC
 ABSRB2
 ABSRB3
 RNLH
 RNWO02
 RNPOO1
 RNFF02
 TMPO2
 PYRO
 TMGAS
 CHEM
 PLMHT
 TMSPLM
 PTPLM
 MPLUM
 ZPLUM
 RDNP
 HEIGHT
 RADLW
 RADFW
 CNVW1
 TMPW1

TMPW2
 DXDPRD
 MNXPRD
 SUTPRD
 WTRPRD
 BURN
 FLUX
 VNTCVR
 CVWID
 PWLPLR
 VTFLOW
 FORV
 FLOW
 SPECFL
 ENTPL
 OXYBRN
 TENPLM
 ERFDE (DBLES1F.FOR)
 NETMAS (File DBLES4A.FOR)
 TCO2M (File DBLES5A.FOR)
 TMNXM (File DBLES5A.FOR)
 TOXYM (File DBLES5A.FOR)
 TH2OM (File DBLES5A.FOR)
 TSUTM (File DBLES5A.FOR)
 SPREAD
 RSTART
 PULRAD
 SCALE (File DBLES2D.FOR)
 EVJAC (File EVJAC.FOR)
 FUNCT (File DBLES1F.FOR)
 ENPACK
 EVALFP
 ERFDE
 DECOMP (File EVJAC.FOR)
 SOLVE (File EVJAC.FOR)
 CORECT (File EVJAC.FOR)
 FUNCT
 SOLVE
 ADINIT (entry point:ALDIFF, file ALDIFF.FOR)
 WRIT (file WRIT.FOR)
 UNPACK
 PLTOUT (File PLTOUT.FOR)
 TENPLM
 VNTFLOW (File DBLES4A.FOR)
 FORV (File DBLES4A.FOR)
 FLOW (File DBLES6A.FOR)
 SCALE (File DBLES2FD.FOR)
 ALDIFF (File ALDIFF.FOR)
 EVJAC
 FUNCT
 UNPACK
 EVALFP

ERFDE
DECOMP
SOLVE
CORECT
FUNCT
UNPDSC (File UNPDSC.FOR)
VARNAM (File DBLES3G.FOR)
LOOKUP (File DBLES3G.FOR)
BLOCK DATA SCALAR (File WRIT.FOR)
[End of Calculation Section]

NISTIR 4464

November 1990

BIBLIOGRAPHIC DATA SHEET

TITLE AND SUBTITLE

Using the Harvard/NIST Mark VI Fire Simulation

AUTHOR(S)

John A. Rockett

PERFORMING ORGANIZATION (IF JOINT OR OTHER THAN NIST, SEE INSTRUCTIONS)

U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
GAITHERSBURG, MD 20899

7. CONTRACT/GRANT NUMBER

8. TYPE OF REPORT AND PERIOD COVERED

SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (STREET, CITY, STATE, ZIP)

9. SUPPLEMENTARY NOTES

10. ABSTRACT (A 200-WORD OR LESS FACTUAL SUMMARY OF MOST SIGNIFICANT INFORMATION. IF DOCUMENT INCLUDES A SIGNIFICANT BIBLIOGRAPHY OR LITERATURE SURVEY, MENTION IT HERE.)

Installation and use of the Harvard/NIST VI multi-room fire simulation (computer code on the IBM PC (and compatibles) is described. Two separate auxiliary programs, MASBANK and PREPLOT are also described and their use illustrated. Use of the input menus is described and illustrated. Comments are made on the effect of the choice of alternative sub-models on predicted results. Some suggestions for use on other computers are made. The H06.3 disks for use with this report include: Disk #1: All files (in packed form) needed to run H06.3 and the two auxiliary programs: MASBANK and PREPLOT: A group of (packed) files including a short data base file and a sample output of H06.3. Disk #2: Section 1 including 17 out of 25 source code files for H06.3. Disk #3: Section 2 including the final 8 source code files of H06.3 and source code for MASBANK, PREPLOT and the auxiliary source code file DTIME.

11. KEY WORDS (6 TO 12 ENTRIES; ALPHABETICAL ORDER; CAPITALIZE ONLY PROPER NAMES; AND SEPARATE KEY WORDS BY SEMICOLONS)

fire simulation; computer programs; manuals

13. AVAILABILITY

UNLIMITED
FOR OFFICIAL DISTRIBUTION. DO NOT RELEASE TO NATIONAL TECHNICAL INFORMATION SERVICE (NTIS).

 ORDER FROM SUPERINTENDENT OF DOCUMENTS, U.S. GOVERNMENT PRINTING OFFICE,
WASHINGTON, DC 20402.

 ORDER FROM NATIONAL TECHNICAL INFORMATION SERVICE (NTIS), SPRINGFIELD, VA 22161.

14. NUMBER OF PRINTED PAGES

44

15. PRICE

A03

