

**APPENDIX B — FIRE AND SMOKE SIMULATOR (FSSIM)
VERSION 1 — USER'S GUIDE**



NRL/MR/6180--04-8806

Fire and Smoke Simulator (FSSIM) Version 1 — User's Guide

JASON E. FLOYD
SEAN P. HUNT

Hughes Associates, Inc.
Baltimore, MD

PATRICIA A. TATEM

ITT Industries — Advanced Engineering Sciences
Alexandria, VA

FREDERICK W. WILLIAMS

Navy Technology Center for Safety and Survivability
Chemistry Division

July 16, 2004



Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE (DD-MM-YYYY) 16 July 2004			2. REPORT TYPE Memorandum Report			3. DATES COVERED (From - To)		
4. TITLE AND SUBTITLE Fire and Smoke Simulator (FSSIM) Version 1 — User's Guide						5a. CONTRACT NUMBER		
						5b. GRANT NUMBER		
						5c. PROGRAM ELEMENT NUMBER		
6. AUTHOR(S) Jason E. Floyd,* Sean P. Hunt,* Patricia A. Tatem,† and Frederick W. Williams						5d. PROJECT NUMBER		
						5e. TASK NUMBER		
						5f. WORK UNIT NUMBER 61-M704-B-3-5		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Research Laboratory, Code 6180 4555 Overlook Avenue, SW Washington, DC 20375-5320						8. PERFORMING ORGANIZATION REPORT NUMBER NRL/MR/6180--04-8806		
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Office of Naval Research 800 North Quincy Street Arlington, VA 22217-5660						10. SPONSOR / MONITOR'S ACRONYM(S)		
						11. SPONSOR / MONITOR'S REPORT NUMBER(S)		
12. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.								
13. SUPPLEMENTARY NOTES *Hughes Associates, Inc., 3610 Commerce Drive, Baltimore, MD †ITT Industries — Advanced Engineering Sciences, Alexandria, VA								
14. ABSTRACT Several computational tools exist for examining fire effects that can be applied to a ship and its crew. Hand calculations can be used to examine simple scenarios in single compartments. Zone models are also suitable for more complex, time-dependent scenarios involving multiple compartments and levels, but stability can be a problem for multi-level scenarios, scenarios with Heating, Ventilation, and Air Conditioning (HVAC) systems, and for post-flashover conditions. Computational fluid dynamics (CFD) models can yield detailed information about temperatures, heat fluxes, and species concentrations; however, the time penalty of this approach currently makes using CFD unfeasible for long periods of real time or large computational domains. Another class of models has traditionally played supporting roles in fire modeling. This class is constituted by a variety of network models, which are used for ventilation systems in buildings or fluid flow in piping networks. These models lack specific physics required for fire modeling. Therefore, to meet the computational speed and algorithm requirements, it was decided to develop Fire and Smoke Simulator (FSSIM) as a network fire model. This document (User's Guide) discusses creation of FSSIM input files, including guidance in transforming a prototype structure to a network representation, and explains the available outputs.								
15. SUBJECT TERMS Fire and smoke spread; Network model; Simulation; Fire model								
16. SECURITY CLASSIFICATION OF:				17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON Frederick W. Williams		
a. REPORT Unclassified	b. ABSTRACT Unclassified	c. THIS PAGE Unclassified		UL	108	19b. TELEPHONE NUMBER (include area code) 202-767-2476		

CONTENTS

1.0	INTRODUCTION	1
1.1	Motivation for Developing FSSIM.....	1
1.2	Network Model Concept.....	2
2.0	BACKGROUND	3
2.1	Documentation Overview	3
2.2	Features of FSSIM	3
2.3	Applications	4
3.0	INSTALLING AND RUNNING FSSIM.....	4
3.1	Computer Hardware and Software Requirements	4
3.2	Running FSSIM	4
3.2.1	Installing FSSIM.....	4
3.2.2	Creating the Input File	5
3.2.3	Starting the FSSIM Calculation	5
3.2.4	Common Runtime Errors.....	6
4.0	FSSIM INPUT FILE STRUCTURE	8
4.1	Overview.....	9
4.1.1	NAMELIST Input Format	9
4.1.2	Variable and Parameter Input Format.....	10
4.1.3	Model Elements	11
4.2	General Simulation and Data Control.....	14
4.2.1	Execution Control (&EXEC)	15
4.2.2	NAMELIST Group Input Counters (&CNTR).....	16
4.2.3	Output Data Options (&OUTP).....	18
4.2.4	Input Data Files (&FNAM).....	20
4.2.5	Tabular Input Data (&CURV)	20
4.2.6	Input Conditional Control (&CTRL)	22
4.3	Compartments.....	24
4.3.1	Compartment Definition (&COMP)	24
4.3.2	Compartment Opening Definition (&JUNC).....	26
4.3.3	Compartment Surface Definition (&SURF)	30
4.3.4	Compartment Surface Composition (&CMPN).....	33
4.3.5	Surface Composition Material Definition (&MTRL).....	35
4.4	Mechanical Ventilation.....	38
4.4.1	Mechanical Ventilation Nodes (&RNOD).....	38
4.4.2	Ducts (&RDCT).....	41
4.4.3	Fans (&RFAN)	45
4.5	Source Fires and Fuel Load	48
4.5.1	Initial Source Fire (&FIRE).....	48
4.5.2	Fuel Load Parameters (&USES).....	52
4.5.3	Fuel Load (&FUEL).....	53

4.6	Detection and Fire Suppression	57
4.6.1	Detection Systems (&DTCT).....	57
4.6.2	Suppression Systems (&SUPR).....	59
5.0	FSSIM OUTPUT	63
5.1	General Output Format	63
5.2	MSU Viewer Output Format	75
6.0	SAMPLE CASES	77
6.1	Sample 1: Fire in Steel Compartment Connected to One Adjacent Compartment.....	77
6.2	Sample 2: Fire in Insulated Steel Compartment with Irregular Plan and Fixed Suppression.....	79
6.3	Sample 3: Fire in ‘U’-Shaped Compartment with Gypsum and Steel Boundaries.....	82
6.4	Sample 4: Multiple Compartment Arrangement Connected by a Corridor.....	85
7.0	REFERENCES	90
8.0	NOMENCLATURE	93

ACRONYMS

ANSI	American National Standards Institute
ASCII	American Standard Code for Information Exchange
ASHRAE	American Society of Heating, Refrigerating, and Air-Conditioning Engineers
CD	Compact Disk
CFD	Computational Fluid Dynamics
CPU	Central Processing Unit
CSV	Comma Separated Value
DOS	Disk Operating System
FSSIM	Fire and Smoke Simulator
GOTHIC	Generation of Thermal Hydraulic Information for Containments
GUI	Graphical User Interface
HAI	Hughes Associates, Inc.
HVAC	Heating, Ventilation, and Air-Conditioning
ISO	International Standards Organization
MSU	Mississippi State University
PC	Personal Computer
RAM	Random Access Memory
RELAP	Reactor Excursion and Leak Analysis Program.
TRAC	Transient Reactor Access Code

FIRE AND SMOKE SIMULATOR (FSSIM) VERSION 1 – USER’S GUIDE

1.0 INTRODUCTION

1.1 Motivation for Developing FSSIM

In both peacetime and during war, fire represents a significant threat to any ship. A fire, whether started by a mechanical failure, intentional or unintentional human actions, or damage from a weapon hit, threatens the ship in a number of ways. The crew’s health and ability to operate the ship is affected by direct exposure to the fire or by the spread of smoke and toxic gasses through the ship by either natural or mechanical ventilation. Electrical systems can be degraded by thermal exposure, exposure to acid gasses in the combustion products, or by electrical failure resulting from soot deposition, which might include hampered cooling or dielectric breakdown from the electrical conductivity of the soot. Mechanical systems can suffer thermal damage. Lastly, fire growth and spread could potentially ignite explosive materials, rocket motors, aviation fuel, or other highly flammable substances, which could possibly result in temperatures or overpressures high enough to affect the ship structurally.

There currently exist a number of computational tools for examining the effects of a fire that can be applied to a ship and its crew. One could use hand calculations for examining simple scenarios in single compartments. Simple rules can be used to extend this approach to multiple compartments. Zone models are suitable for examining somewhat more complex, time-dependent scenarios involving multiple compartments and levels, but stability can be a problem for multi-level scenarios, scenarios with Heating, Ventilation, and Air Conditioning (HVAC) systems, and for post-flashover conditions. Computational fluid dynamics (CFD) models can yield detailed information about temperatures, heat fluxes, and species concentrations; however, the time penalty of this approach currently makes using CFD unfeasible for long periods of real time or large computational domains. There is another class of models which have traditionally played supporting roles in fire modeling. This class is constituted by a variety of network models, which are used for ventilation systems in buildings or fluid flow in piping networks. These models, however, lack specific physics required for fire modeling.

The need for fire modeling occurs through the “cradle-to-grave” lifetime of a ship, which encompasses the phases of concept design, detailed design and hazards evaluation, doctrine and tactics development, ship operation and maintenance, and crew training. In the concept design phase, fire modeling is needed to evaluate ship designs and design philosophies in order to quickly arrive at an overall concept to meet required performance goals. As this concept is refined into a detailed design, fire modeling is continued to evaluate ship vulnerability and recoverability and to begin the process of defining ship operation. Both of these phases could potentially require evaluating the fire behavior of a ship and its system for hundreds of scenarios in design cycles spanning a few months time. Doctrine and tactics planning also require evaluating many separate fire scenarios in combination with a number of candidate doctrines. During the operational phase of a ship’s lifetime fire modeling can support crew training through interactive training simulations. Fire modeling can also aid in the recovery from a fire event by

providing a faster than real time predictive capability to gauge the potential effectiveness of a course of action.

For each of the aforementioned uses of fire modeling, speed and the ability to model large complexly interconnected spaces with ventilation, detection, and suppression with a reasonable degree of accuracy are performance requirements for the model. Hand calculations, while fast, have limitations in applicability and large uncertainties in their results. CFD computations have the potential to be accurate to any desired degree, but are slow. Zone models balance well computational time and accuracy; however, in general these models do not perform well with ventilation systems and they lack the ability to model control systems. To meet the computational speed and algorithm requirements, it was decided to develop Fire and Smoke Simulator (FSSIM) as a network fire model.

1.2 Network Model Concept

A network model represents a one-dimensional abstraction of the physical world. The prototype being modeled is represented by a set of nodes and node connections, where the nodes contain one set of physical variables and the node connections represent transfers of those variables between nodes. Contrast this with a zone model which uses two sets variables or a CFD model which uses a multitude. A network representation allows for maximum physical extent of a simulation with a minimum set of equations, one per geometric volume of interest. Since the number of locations being solved for is kept to a minimum, a network model also has the potential for the faster execution speeds.

There are currently a number of network models in existence that are used for safety related simulations. Some of these include:

- TRAC – Transient Reactor Analysis Code. Used for simulating normal and accident conditions of nuclear power plant cooling systems [Spore, et al., 2000].
- RELAP – Reactor Excursion and Leak Analysis Program. Used for simulating normal and accident conditions of nuclear power plant cooling systems [USNRC, 2001].
- GOTHIC – Generation of Thermal Hydraulic Information for Containments. Used for simulating normal and accident conditions of nuclear power plant containment buildings and reactor buildings [George, et al., 2000a-b].
- MELCOR – Used for simulating normal and accident conditions of nuclear power plant containment buildings and reactor buildings including aerosol dispersion, core melt/core melt interactions, and hydrogen combustion [Gauntt, et al., 2000].
- CONTAM – Used for simulating the dispersal of airborne contaminants through a building and its associated HVAC systems [Dols, Walton, and Denton, 2000].
- COCOSYS – COntainment COdeSYStems Used for simulating accident conditions in light water nuclear power plant containment buildings and reactor buildings [Allelein *et al.*, 1999; Klein-Heßling *et al.*, 2000].

2.0 BACKGROUND

FSSIM is a network fire model written to simulate the spread of fire and smoke in a naval vessel. However, there is nothing in the model to preclude its use for other types of compartmented structures such as a building. FSSIM is written in standard Fortran 95 [Adams *et al.*, 1997; Lahey Computer Systems, 1998] and as such is capable of being compiled on any platform for which a Fortran 95 compiler is available. In FSSIM each compartment in a structure is represented as a single node with surfaces (e.g. bulkheads, decks, and overheads) and vent openings (e.g. doors and hatches) represented as node connections. There is no practical limit (2×10^9 items of any type) to the size of an FSSIM simulation other than available computational resources and time.

2.1 Documentation Overview

FSSIM documentation consists of three volumes. The FSSIM Theory Manual [Floyd *et al.*, 2004a], the FSSIM Users' Manual (this document), and the FSSIM Validation Manual [Floyd *et al.*, 2004b]. The Theory Manual describes the equations solved, and the solution algorithm for the heat and mass transfer along with the equations and algorithms for FSSIM sub-models. The Users' manual discusses creation of FSSIM input files including guidance in transforming a prototype structure to a network representation and explains the available outputs. Lastly, the Validation Manual documents FSSIM performance as compared to experimental data and other fire models.

There is a companion viewer for FSSIM that is under development by Mississippi State University. This viewer will display animated results of FSSIM calculations in a three-dimensional representation of the ship [Haupt *et al.*, 2004].

2.2 Features of FSSIM

FSSIM has many features, including:

- One-dimensional flow model including friction losses and temperature-dependent specific heat.
- One-dimensional multiple layer, temperature-dependent heat transfer.
- N-surface, gray-gas radiation heat transfer including radiation streaming through openings.
- Bidirectional flow through horizontal (hatches) and vertical (doors) flow connections.
- Combustion product species tracking.
- Oxygen- and fuel-limited combustion.
- Multiple user-defined fires along with fire spread via compartment-to-compartment heat transfer.

- HVAC systems including ducts, dampers, and fans with forward and reverse flow losses and multiple fan models.
- Fire detection via heat, smoke, and fire detection.
- Fire spread by compartment-specific criteria.
- Fire suppression via sprinklers, water mist, gaseous agents, aerosol agents, and foam.
- Fire spread prevention via boundary cooling.
- Simple control systems to link operation of equipment to sensors or times.
- Fast, near-real-time execution speed.
- Can be run in a standalone configuration or as a callable subprogram.

2.3 Applications

FSSIM has a wide range of applications, including:

- Real-time simulation of shipboard conditions during a fire.
- Damage control.
- Event tree sequences.

3.0 INSTALLING AND RUNNING FSSIM

3.1 Computer Hardware and Software Requirements

FSSIM requires a moderately fast central processing unit (CPU) with a sufficient amount of random access memory (RAM) to accommodate the complexity of the particular simulation. FSSIM will alert the user if there is not a sufficient amount of memory for an attempted simulation. For a Windows based personal computer (PC), a processor that is as fast as a 400 Mhz Pentium II should be minimally sufficient to execute basic FSSIM configurations. More complex simulations should be executed on a faster system; otherwise real-time simulations may not be attained. For a UNIX-based station, the processor should be at least as fast as the PC platform specifications.

3.2 Running FSSIM

3.2.1 Installing FSSIM

FSSIM may be installed from a compact disk (CD) or it may be downloaded from a designated site. If FSSIM is being installed from a CD, the following applies:

1. Create a sub-directory where FSSIM will be executed.
2. Copy all files from the CD into the sub-directory created. Files should include:

- `FSSIM1.exe` – the executable program.
- `materials.txt` – material properties file containing information for common materials.
- `usetypes.txt` – file containing fuel-loading definitions of common compartment classifications.
- `fuels.txt` – fuel properties file containing information for common fuels.
- `sample1.in`, `sample2.in`, `sample3.in`, and `sample4.in` – sample input files.

3. Run the sample input files to verify proper installation (see Section 3.2.3).

If FSSIM is being downloaded from an `ftp` site or equivalent, the following applies:

1. Create a sub-directory where FSSIM will be executed.
2. Download the compressed file `FSSIM1.ZIP`.
3. Extract the files from `FSSIM1.ZIP` into the sub-directory created in step 1.
4. Run sample input files to verify proper installation (see Section 3.2.3).

3.2.2 Creating the Input File

FSSIM requires an input file containing details regarding the configuration simulated. The input file is an ASCII file that may be edited using any suitable text editor. Section 4 contains details on preparing the input file and Section 6 provides several input sample data files.

3.2.3 Starting the FSSIM Calculation

FSSIM is a command line executable program. The program must be run from a disk operating system (DOS) or UNIX shell.

- Microsoft Windows based PC

Open a Command Prompt window, change directories to where `FSSIM.EXE` and the necessary input files are located, then run the model as follows:

```
fssim inputfile.in
```

where `inputfile.in` is any valid input file. The input file will be read and the simulation begun. The current time and other information will be printed to the screen so that the user may monitor the progress of a simulation. If the input file is not valid, the program will be aborted and an error will be generated. Note that the executable and the input files are not case sensitive in a DOS shell.

The information printed to the screen during a simulation may instead be saved in a file by running FSSIM as follows:

```
fssim inputfile.in > output.out (overwrites output.out)
```

or

```
fssim inputfile.in >> output.out (appends to output.out)
```

where `output.out` is any output file name.

- **UNIX platform**

Open a UNIX shell, change directories to where `FSSIM.EXE` and the necessary input files are located, then run the model as follows:

```
fssim < inputfile.in
```

where `inputfile.in` is any valid input file. The input file will be read and the simulation begun. The current time and other information will be printed to the screen so that the user may monitor the progress of a simulation. If the input file is not valid, the program will be aborted and an error will be generated. Note that the executable and the input files are case sensitive in a UNIX shell.

The information printed to the screen during a simulation may instead be saved in a file by running FSSIM as follows:

```
fssim < inputfile.in > output.out & (overwrites output.out)
```

where `output.out` is any output file name.

3.2.4 Common Runtime Errors

FSSIM may fail to run for a variety of reasons. Common problems are summarized in the table below.

Table 3-1. Common FSSIM errors and Possible User Actions

FSSIM Error	Possible Cause	User Actions
<p>FSSIM does not find one or more compartments, junctions, surfaces, compositions, materials, ducts, duct nodes, fans, suppressors, and/or detectors.</p>	<p>The &CTRL line does not match number of desired inputs.</p>	<ul style="list-style-type: none"> • Make sure that the parameters on the &CTRL line match the number of lines that should be read for each item. • Make sure that all NAMELIST groups are correctly spelled. • Make sure that that NAMELIST groups with parameters set in auxiliary input files are properly set up for file-read.
<p>Program aborts unexpectedly or inexplicably.</p>	<p>Input file format/setup error. Examples include a wrong variable, a NAMELIST syntax error, and an undefined parameter that is required.</p>	<p>Verify that the input file is in the correct format and that all required parameters are included on each NAMELIST line.</p>
	<p>Invalid id parameter value. An id parameter value is a unique integer that identifies various FSSIM entities, such as a compartment, an opening, or a node. These values are used to locate openings, nodes, ducts, fans, etc. relative to each other. FSSIM does not check for valid id parameter values. If an invalid id is used, FSSIM may run, partially run, or abort immediately with or without indicating that there is an invalid id parameter value.</p>	<p>Verify that all id parameters are valid, both as defined and as supplied for parameter arguments.</p>
	<p>Improper physical geometry. This condition may arise when an incorrect id parameter value is supplied as argument, such that an opening or a duct is connected to the wrong compartment or node.</p>	<p>Verify the input configuration.</p>
<p>FSSIM Fails to converge.</p>	<p>See “Program aborts unexpectedly or inexplicably”.</p>	<p>See “Program aborts unexpectedly or inexplicably”.</p>
	<p>Convergence parameters must be adjusted for configuration modeled.</p>	<p>Increase the number of inner or outer timestep loops, decrease minimum timestep size, simplify mechanical ventilation system, and/or add natural ventilation to spaces that are well sealed.</p>
<p>FSSIM will not find an input file.</p>	<p>Incorrect input file or &FNAM specified file is not valid or not located in current directory.</p>	<p>Verify that the input file is in current directory. Verify that any files set using the &FNAM NAMELIST group and used as input vehicles for the particular simulation are located in the current directory.</p>

FSSIM Error	Possible Cause	User Actions
The results do not make physical sense.	Incorrect values used. An example is a heat release rate provided in W where kW is required for input.	Verify that the values provided as parameter arguments have the proper units. Verify that the physical values and quantities supplied are the values intended.
Insufficient memory.	Configuration is too large for computer.	Reduce the size of the problem or use a computer with a greater amount of RAM.
FSSIM fails to run and none of the user actions above correct the problem.	Computer code error.	Contact the primary author: Jason Floyd 3660 Commerce Drive, Suite 817 Baltimore, MD 21227-1652 jfloyd@haifire.com

4.0 FSSIM INPUT FILE STRUCTURE

FSSIM is run using a valid ASCII input file that contains a description of the input geometry, various initial and subsequent conditions, and other user-defined information. FSSIM has some error trapping; however there is a significant reliance on the user to adequately prepare the input file for the configuration desired, both in terms of physical representation and input file integrity. This section summarizes the structure of the FSSIM input files. Specifically, the following are addressed:

- Key elements of FSSIM input file as they relate to physical configurations.
- Input parameters/variables that are accessible to the user via NAMELIST groups and are used to build a simulation.
- The type of input variable or parameter (integer, real number, text, or logical).
- Appropriate variable/parameter input ranges, where applicable.
- The minimum or required input variables/parameters.
- The default conditions.

NAMELIST groups allow the user to access a variety of model parameters or variables, required or optional. These parameters and variables are identified for each NAMELIST group available in the FSSIM model in the subsequent sections. The input format is noted as a real number, an integer, text, or a logical statement. The input variable or parameter may be a single element or an array of elements. Note that real numbers may be used as single precision or double precision variables within the model. Because the input file format is transparent to this, a distinction between the two variable types is not necessary in this guide.

Input data ranges or restrictions that are intuitively obvious are not listed. This includes the restrictions that time, timesteps, convergence criteria, area, volume, length, width, pressure,

density, heat capacity, thermal conductivity, and number of items are positive values greater than zero (unless specifically allowed to be zero); minimum values are less than or equal to maximum values; identifications for unique elements are unique; and related restrictions.

In the following subsections each NAMELIST group will be described. Not all NAMELIST groups will be required for a given simulation. In addition, for those NAMELIST groups needed, not all variables in the group will need to be specified. Each group will contain some variables that the user must set for a successful simulation. NAMELIST may also contain subsets of variables of which only one may be set (*i.e.*, the variables/parameters are mutually exclusive). Many parameters/variables that are necessary for performing a simulation have default values, and thus, do not have to be set by the user unless a value other than the default is desired. Required parameters, default values, mutually exclusive parameters, and other such conditions are identified for each NAMELIST group in the subsequent sections.

4.1 Overview

4.1.1 NAMELIST Input Format

The user builds a simulation model in FSSIM using NAMELIST parameters groups. A NAMELIST parameter group is a means of setting variables and parameters within the model that are associated with a particular group, use, or function. The format is as follows:

```
&NLID variable1=x, variable2=.FALSE., variable3='string', ... /
```

where `&NLID` is the NAMELIST parameter group identifier and all variables that are set are located between the group identifier and a `'` symbol and are separated from each other by a comma. A NAMELIST line is defined as all of the information located between the `'&` and `'/'` symbols and a specific NAMELIST line is defined as all of the information between the group identifier and the `'/'` symbol.

FSSIM uses five character NAMELIST group identifiers. The group identifier must be located in the first five columns of the input file and variables that are set in a NAMELIST group must be separated from each other by a comma. Other than that, there are no formatting restrictions on the input file and NAMELIST format structure. There are also no restrictions on the order in which the variables appear within the NAMELIST group or the order in which the NAMELIST groups appear within an input file. Thus,

```
&NLID variable2=.FALSE., variable3='string', variable1=x, ... /
```

is equivalent to the `&NLID` NAMELIST parameter group example provided above and it may appear anywhere within the input file.

Text located on the same input file line located to the right of a `'!` symbol is treated as a user comment. Thus, NAMELIST lines may be converted to user comments by placing the `'!`

symbol in column 1. In addition, text that appears outside the `&NLID ... /` construct is treated as a user comment. The following are examples of user input file comments:

```
!&NLID variable2=.FALSE., variable3='string', variable1=x, ... /The &NLID line  
and this text are user comments
```

```
&NLID variable2=.FALSE., variable3='string', variable1=x, ... /Only this text  
is a user comment
```

The input is not case sensitive; however, it is conventional to use upper case for the NAMELIST parameter identifier and whatever case or combination of cases are used by the variable or parameter within the NAMELIST group. The only exception to this are text string variables, which are case sensitive. This is of particular importance when naming items that must be retrieved from files. The names must match exactly for a successful retrieval.

There may be one or more instances of a particular NAMELIST parameter group contained in an input file. In such case, a unique identifier is normally required to be supplied by the user. Multiple instances of a NAMELIST parameter group are necessary when there is more than one compartment; multiple surfaces or materials within a compartment; more than one duct, fan, or duct node; multiple fires; multiple openings; *etc.* This is described in greater detail in the sections that follow. If there are more instances of a particular NAMELIST parameter group defined than are necessary or required, only those up to the required number are actually read as input. Thus, if ten instances of a particular NAMELIST group are required and twelve are provided, the first ten are read as input and the remaining two are ignored.

4.1.2 Variable and Parameter Input Format

There are format requirements associated with each type of variable input. They are as follows:

- Real numbers are entered using a decimal:

```
real_variable_example=1.75
```

- Integers are entered without using a decimal:

```
integer_variable_example=5
```

- Logical variables are entered using true or false statements:

```
logical_variable_example1=.FALSE.
```

```
logical_variable_example2=.TRUE.
```

- Text strings are entered using single quotes and should not contain commas to avoid potential output formatting errors:


```
text_string_example1='example string - this string is case sensitive'  
text_string_example2='Example string - this string is case sensitive'
```

Real number variables that are entered as integers will be converted to single or double precision variables in the model. Conversely, integer variables that are entered as real numbers are truncated and converted to an integer in the model. Logical states may be expressed as either `.TRUE.` or `.FALSE.` or equivalently as `.true.` or `.false.`, though the former is the conventional input style.

An integer normally has a value between -2^{15} and $+2^{15}$ (-32,767 and +32,767). Exceptions to this are short integers, which must have a value between -2^7 and $+2^7$ (-127 and +127), and long integers, which must have a value between -2^{31} and $+2^{31}$ (-2,147,483,647 and +2,147,483,647). Short and long integers are used for some types of data and are identified where applicable. Integer values entered that exceed the bounds for the particular integer type will produce erratic results. Text strings that exceed the maximum length are truncated to the maximum length. Note that text strings are case sensitive, such that the two string examples cited above are not equal because the 'e' is capitalized in the second example.

Some NAMELIST group parameters require more than one input value. These are referred to in this document as parameter arrays and they are input on a NAMELIST line using comma separators between the array elements. The following are examples of parameter arrays:

- Integer parameter array with two elements:

```
example_integer_array=1,-1
```

- Real number parameter array with six elements:

```
example_real_number_array=1.53, 2.66, 2.66, 1.0, 1.44, 1.354
```

There are no logical or text string variable arrays used by the FSSIM NAMELIST groups. The input order for parameter arrays may or may not be significant. Descriptions of specific NAMELIST group parameter arrays will indicate whether or not the array order is of significance.

4.1.3 Model Elements

FSSIM is used to simulate a fire and determine the resulting compartment conditions within one or more compartments. The configuration and scenario simulated are entered using up to eighteen different NAMELIST parameter groups. Each parameter group sets related parameters, such as the dimensions of a compartment or the location and size of openings in a compartment. Figure 4-1 depicts a sample compartment and identifies the parameter groups used to set the parameters needed to simulate a fire scenario. Table 4-1 summarizes the parameter

groups and the types of parameters they set. Sections 4-3 through 4-5 summarize the syntax of each NAMELIST parameter group shown in Table 4-1.

Table 4-1. FSSIM NAMELIST Parameter Groups

NAMELIST Parameter Group		Requirements/ Use	Type of parameters set by the NAMELIST Parameter Group
General Simulation and Data Control	&EXEC	Required. One and only one &EXEC line must be provided.	<ul style="list-style-type: none"> • Time and time-related parameters. • Convergence control. • Ambient conditions. • Output format.
	&CNTR	Required. One and only one &CNTR line must be provided.	<ul style="list-style-type: none"> • The number of specific types of NAMELIST parameter groups to be input.
	&OUTP	Optional. If used, only one &OUTP line is provided	<ul style="list-style-type: none"> • Specifies outputs to be written if the variable <code>outtype</code> on &EXEC is <code>.FALSE.</code>
	&FNAM	Optional. If used, only one &FNAM line is provided.	<ul style="list-style-type: none"> • Names of files used to provide input data for specific NAMELIST parameter groups.
	&CURV	Optional. One or more &CURV lines must be provided as referenced by other NAMELIST parameter groups.	<ul style="list-style-type: none"> • Mechanism to enter tabular data, such as material properties of heat release rate.
	&CTRL	Optional. One or more &CTRL lines must be provided if referenced by another NAMELIST group.	<ul style="list-style-type: none"> • Mechanism to turn objects on or off, such as a fan or a suppression system. Also be used to open or close a door or a fire damper.
Compartment Information	&COMP	Required. One &COMP line for each compartment or compartment section is necessary.	<ul style="list-style-type: none"> • Compartment dimensions and volume. • Initial compartment conditions. • Type of combustible fuel load in a compartment.
	&JUNC	Optional. One &JUNC line is provided for each opening, if any.	<ul style="list-style-type: none"> • Opening dimensions, location, and orientation. • Bidirectional or one-way flow. • Type of opening control. • Opening flow loss coefficient and inertial length.
	&SURF	Required. Minimum of six for each unique &COMP line.	<ul style="list-style-type: none"> • Type, location, and orientation of boundary surfaces. • Area and composition of boundary surfaces.
	&CMPN	Required, unless all surfaces are fake. One &CMPN line is required for each unique composition identified by the &SURF lines.	<ul style="list-style-type: none"> • The thickness and type of material composing a solid. • The number, thickness, and type of material composing each individual layer in composite solid.
	&MTRL	Required, unless all surfaces are fake. One &MTRL line is required for each unique material identified by the &CMPN parameter groups.	<ul style="list-style-type: none"> • The thermal conductivity, heat capacity, density, and emissivity of a material.

NAMELIST Parameter Group		Requirements/ Use	Type of parameters set by the NAMELIST Parameter Group
Mechanical Ventilation	&RDCT	Optional. One &RDCT line is provided for each duct segment modeled, if any.	<ul style="list-style-type: none"> • The length and cross section of duct segments. • Flow pressure loss characteristics through ducts. • Duct damper control. • Type of fan in duct. • Constant mass/or volume flow rates through ducts.
	&RNOD	Optional. One &RNOD line is provided for each duct node location modeled, if any.	<ul style="list-style-type: none"> • Location and elevation of node. • Ducts connected to node. • Flow pressure loss characteristics at node.
	&RFAN	Optional. One &RFAN line is provided for each fan modeled, if any.	<ul style="list-style-type: none"> • Type of fan. • Fan performance characteristics. • Flow pressure loss characteristics through fan.
Source Fire and Fuel Load	&FIRE	Optional. One &FIRE line is provided per compartment if a fire is directly specified within one or more compartments.	<ul style="list-style-type: none"> • Fire heat release rate profile. • Fire location. • Radiant fraction of fire heat release rate. • Heat of combustion and heat of vaporization of burning material. • Yield of various gaseous and solid combustion products of burning material.
	&USES	Optional. One &USES line is provided per compartment not containing a fire (&FIRE) if an initially un-ignited fuel load is simulated.	<ul style="list-style-type: none"> • Location of fuel load. • Type of fuel load.
	&FUEL	Optional. &FUEL lines must be provided as referenced by the &USES line(s).	<ul style="list-style-type: none"> • Growth rate of fire when fuel material ignited. • Radiation fraction of heat release rate when fuel material ignited. • Heat of combustion and heat of vaporization of fuel material. • Fuel material ignition criteria. • Yield of various gaseous and solid combustion products of fuel material.
Detection/Fire Suppression	&DTCT	Optional. One &DTCT line for each type of detection system in each compartment where detection is simulated, if any.	<ul style="list-style-type: none"> • Type of detector. • Location of detector. • Detector actuation criteria.
	&SUPR	Optional. One &SUPR line for each type of detection system in each compartment where suppression is simulated, if any.	<ul style="list-style-type: none"> • Type of suppression system. • Location of suppression system. • Suppression system performance characteristics.

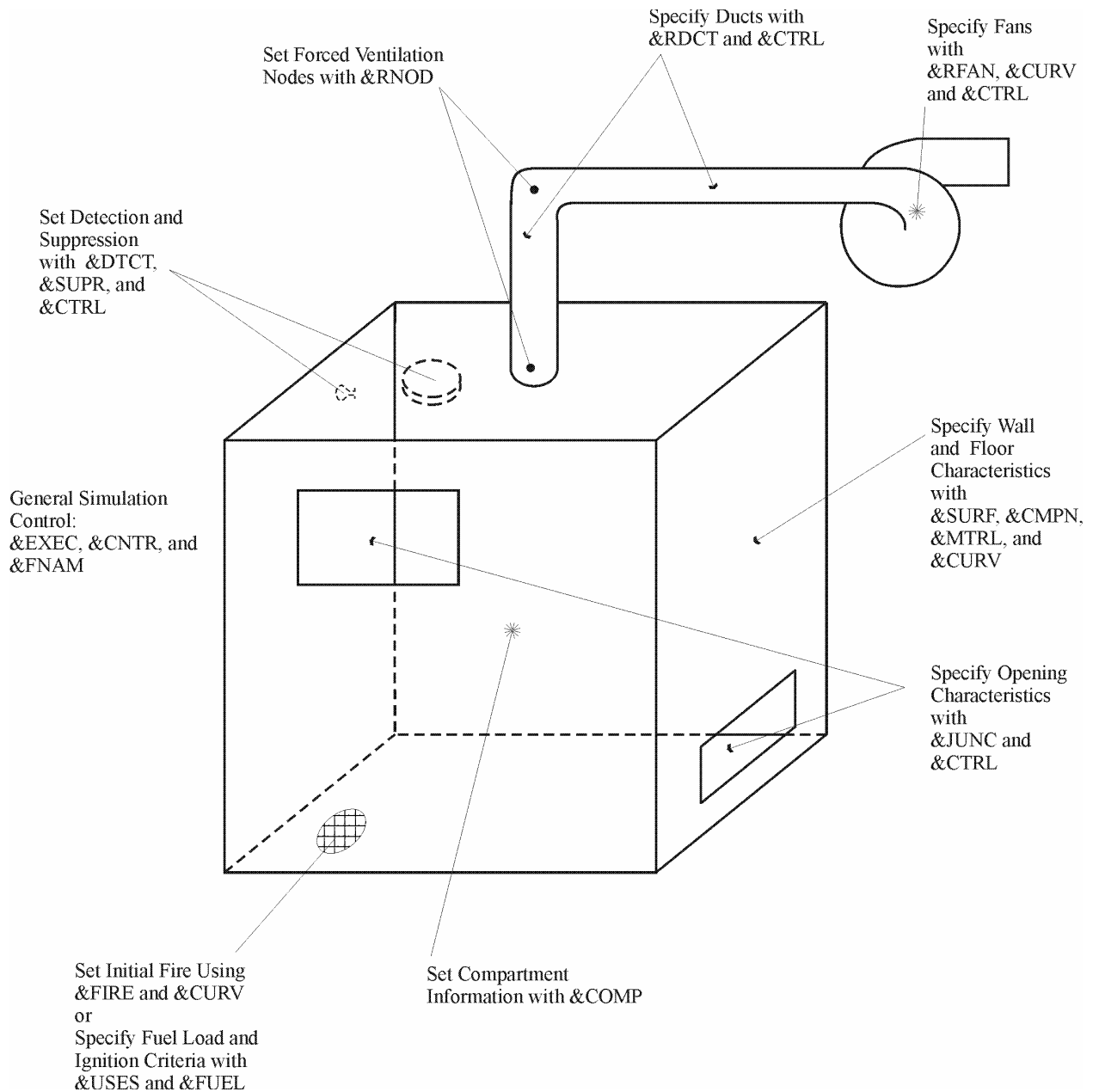


Figure 4-1. Geometry showing how FSSIM NAMELIST Groups Define the Configuration

4.2 General Simulation and Data Control

The general simulation and data control parameter groups are used to set various aspects of a simulation, provide a means of tabular data entry, provide control functions for fans, suppression, detection, *etc.*, and identify the locations of various input files.

4.2.1 Execution Control (&EXEC)

The &EXEC group allows the user to control various aspects of an FSSIM simulation including initial ambient conditions, the solution time, convergence criteria, and the type of output. The input file should contain only one &EXEC parameter group. There is one parameter that must be set on the &EXEC line:

REQUIRED PARAMETER

fileid	(Text string) Text string used to generate output files. There is no default and the string must be 40 characters or less.
--------	--

The remaining &EXEC parameters are optional:

OPTIONAL PARAMETERS

dt	(Real number) The first timestep (s). Default value is 0.01 seconds.
t_max	(Real number) The simulation end time (s). Default value is 100 seconds.
dt_max	(Real number) The maximum timestep size (s). Default value is 1.0 second.
dt_min	(Real number) The minimum timestep size (s). Default value is 0.0001 seconds.
dt_dump	(Real number) The minimum time increment that output is generated (s). The default value is 0.0 seconds, which corresponds to every timestep.
humidity	(Real number) The relative humidity in %. The default is 70.
ninner_max	(Short integer) The maximum number of inner loop (velocity solution) iterations in the network solver. Default is 10.
nout_max	(Short integer) The maximum number of outer loop (pressure solution) iterations in the network solver. Default is 5.
temp_criteria	(Real number) The temperature convergence criteria. Default is 0.1.
pres_criteria	(Real number) The pressure convergence criteria. Default is 0.01.
courant_criteria	(Real number) The Courant convergence criteria. Default is 0.5.
tamb	(Real number) The ambient temperature (K). Default is 298.15K.
pamb	(Real number) The absolute ambient pressure (Pa). Default value is 101,325 Pa.
o2amb	(Real number) The ambient oxygen mass fraction. Default is 0.23.
o2l1	(Real number) The lower limit oxygen mass fraction that will support combustion (kg O ₂ /kg gas). The default value is 0.1 kg O ₂ /kg air.

outtype	(Logical) <code>.FALSE.</code> for ASCII text output and <code>.TRUE.</code> for Mississippi State University (MSU) graphical user interface (GUI) output format. The default state is <code>.TRUE.</code>
---------	--

The default convergence criteria and timestep parameters should be adequate for many simulations. If there is concern regarding solution convergence, the temperature, pressure, and Courant convergence criteria or the initial and maximum timestep could be reduced and the results compared to verify convergence. Likewise, the time required to obtain a solution may be decreased by increasing the tolerances or the initial timestep size. There may be a loss of accuracy if either is decreased. If the model fails to converge for a particular simulation, increasing the number of inner (velocity and fire size) and outer (pressure and courant stability) iterations allowed or decreasing the convergence criteria and initial timestep may improve the model stability and/or the prospects for a converged solution. The user is referred to the Theory Manual for a discussion of the iteration loops.

An example of a `&EXEC` NAMELIST parameter group line is as follows:

```
&EXEC t_max=600.0, dt_dump=1.0, t_amb=302.5, outtype=.FALSE., fileid='sample',
pamb=9.8E4 /
```

In this case, a solution time of 600 seconds is declared, the ambient temperature and pressure are set to 302.5 K and 98,000 Pa, respectively, the output files will begin with the text string `'sample'`, and the output will be in ASCII text format. Output will be generated at intervals no less than 1 second. The default convergence criteria and timestep parameters are used.

4.2.2 NAMELIST Group Input Counters (`&CNTR`)

The `&CNTR` group is used to specify the number of specific NAMELIST groups to be read where the number may vary (*i.e.*, compartment and compartment related information, openings, ventilation and suppression groups, fire and fire related information). There is one parameter that must be set on the `&CNTR` line:

REQUIRED PARAMETERS

ncomp	(Integer) Number of compartment (<code>&COMP</code>) NAMELIST groups to be read. At least one compartment must be specified. The default is zero and if used will generate an error.
-------	--

The remaining `&CNTR` group parameters are optional. However, they may be required to be set if any of the NAMELIST groups linked to the `&CNTR` group variable are used. The following is a complete list of these parameters:

OPTIONAL PARAMETERS

nfire	(Integer) Number of initial source fires (&FIRE) NAMELIST groups to be read. At least one initial fire must be specified, though it may have a zero heat release rate or power. The default is zero.
njunc	(Integer) Number of junctions/openings (&JUNC) NAMELIST groups to be read. The default is zero.
nnotbidir	(Integer) Number of junctions/openings (&JUNC) NAMELIST groups that do not have bidirectional flow. The value must be equal to or less than njunc. The default is zero.
nfuel	(Integer) Number of fuel (&FUEL) NAMELIST groups to be read. The default value is zero.
ntype	(Integer) Number of compartment use type (&USES) NAMELIST groups to be read. The default value is zero.
nsurf	(Long integer) Number of duct node (&SURF) NAMELIST groups to be read. The default value is zero.
ncmpn	(Integer) Number of composition (&CMPN) NAMELIST groups to be read. The default is zero.
nmat	(Integer) Number of material (&MTRL) NAMELIST groups to be read. The default is zero.
nduct	(Integer) Number of duct (&RDCT) NAMELIST groups to be read. The default is zero.
nnode	(Integer) Number of duct (&RNOD) NAMELIST groups to be read. The default is zero.
nfan	(Integer) Number of fan (&RFAN) NAMELIST groups to be read. The default is zero.
ncontrol	(Integer) Number of control (&CTRL) NAMELIST groups to be read. The default is zero.
ndetect	(Integer) Number of detector (&DTCT) NAMELIST groups to be read. The default is zero.
nsuppress	(Integer) Number of suppression (&SUPR) NAMELIST groups to be read. The default is zero.

The number of openings, or junctions, that are not bidirectional (*i.e.*, the solver will allow for flow in each direction across the same opening) must be specified separately from the total number of openings. The model uses the concept of zones to subdivide a network into independent volumes. Spaces that are connected via junctions or ductwork share zones. The zone concept is used to increase the efficiency of the solver and the model currently calculates the number of zones in a network. Zoning is established automatically during the initialization process.

Note that the number of compartments, openings, detectors, *etc.* within a network model are limited by the size of an integer (2^{15}), a sufficiently large number for nearly any practical system. An exception to this is the number of surfaces, which is limited by the size of a long integer (2^{31}).

An example of a &CNTR NAMELIST parameter group line is as follows:

```
&CNTR ncomp=5, nfire=1, njunc=15, nnotbidir=3, nsurf=35, ncmpn=2, nmat=2,
nduct=4, nnode=4, nfan=1, ncontrol=1, ndetect=1 /
```

The above &CNTR line instructs FSSIM to read five &COMP, one &FIRE, fifteen &JUNC, thirty-five &SURE, two &CMPN, two &MTRL, four &RDCT, four &RNOD, one &RFAN, one &CTRL, and one &DTCT NAMELIST groups. In addition, twelve of the openings specified by the &JUNC NAMELIST group have bidirectional flow.

4.2.3 Output Data Options (&OUTP)

The &OUTP group is used to specify the output files written when `outtype=.TRUE.` on &EXEC. All parameters associated with this NAMELIST group are optional. The output file names and file formats are discussed in Section 5.1. The following is a complete list of parameters set by the &OUTP group:

OPTIONAL PARAMETERS

comptemp	(Logical) Write compartment temperature output file. The default is <code>.TRUE.</code>
comppres	(Logical) Write compartment pressure output file. The default is <code>.TRUE.</code>
compq	(Logical) Write compartment radiation absorption/emission to gas and net heat flux to wall output file. The default is <code>.FALSE.</code>
compgas	(Logical) Write compartment gas output file. The default is <code>.TRUE.</code> . The constituent gasses are indicated at the end of this table.
fire	(Logical) Write fire theoretical heat release rate output file and compartment actual heat release rate output file. The default is <code>.TRUE.</code>

nodetemp	(Logical) Write duct node temperature output file. The default is .TRUE .
nodepres	(Logical) Write duct node pressure output file. The default is .TRUE .
nodegas	(Logical) Write duct node gas mass fraction output file. The default is .TRUE . The constituent gasses are indicated at the end of this table.
juncvel	(Logical) Write junction velocity output file. The default is .FALSE .
juncarea	(Logical) Write junction flow area output file. The default is .FALSE .
juncflow	(Logical) Write junction mass flow rate output file. The default is .TRUE .
ductvel	(Logical) Write duct velocity output file. The default is .FALSE .
ductarea	(Logical) Write duct flow area output file. The default is .FALSE .
ductflow	(Logical) Write duct mass flow rate output file. The default is .TRUE .
surftemp	(Logical) Write surface temperature output file. The default is .TRUE .
surfflux	(Logical) Write surface heat flux output file. The default is .FALSE .
detect	(Logical) Write detector alarm status output file. The default is .TRUE .
suppress	(Logical) Write suppression system activation status output file. The default is .TRUE .
gaso2	(Logical) Write oxygen mass fraction to compartment/duct node gas mass fraction output file. The default is .TRUE .
gasco2	(Logical) Write carbon dioxide mass fraction to compartment/duct node gas mass fraction output file. The default is .FALSE .
gash2o	(Logical) Write water vapor fraction to compartment/duct node gas mass fraction output file. The default is .FALSE .
gasco	(Logical) Write carbon monoxide mass fraction to compartment/duct node gas mass fraction output file. The default is .FALSE .
gasc	(Logical) Write soot mass fraction to compartment/duct node gas mass fraction output file. The default is .TRUE .
gasfuel	(Logical) Write unburned fuel mass fraction to compartment/duct node gas mass fraction output file. The default is .FALSE .

It is important to note that there is a limit to the number of concurrently open files in Fortran. This limit as implemented in FSSIM is 250 files. As will be discussed in Section 5.1, the output files are limited to 255 columns of data in order to remain useable by popular COTS spreadsheet software. Thus, given a sufficiently large problem one or more files may be needed for each of the above file types. A fatal error will occur if the user requests more files to be written than can be supported.

4.2.4 Input Data Files (&FNAM)

The &FNAM group is used to specify names of input files to be read for specific NAMELIST parameter groups data. All parameters associated with this NAMELIST group are optional, as is using additional files as data input vehicles. The latter is controlled by other NAMELIST groups. The following is a complete list of parameters set by the &FNAM group:

OPTIONAL PARAMETERS

materialfile	(Text string) Name of file containing &MTRL NAMELIST input data. The default is 'materials.txt' and the string must be 255 characters or less.
fuelfile	(Text string) Name of file containing &FUEL NAMELIST input data. The default is 'fuels.txt' and the string must be 255 characters or less.
usefile	(Text string) Name of file containing &USES NAMELIST input data. The default is 'usetypes.txt' and the string must be 255 characters or less.
compositionfile	(Text string) Name of file containing &CMPN NAMELIST input data. The default is 'compositions.txt' and the string must be 255 characters or less.
fanfile	(Text string) Name of file containing &RFAN NAMELIST input data. The default is 'fans.txt' and the string must be 255 characters or less.
detectorfile	(Text string) Name of file containing &DTCT NAMELIST input data. The default is 'detectors.txt' and the string must be 255 characters or less.
suppressorfile	(Text string) Name of file containing &SUPR NAMELIST input data. The default is 'suppressors.txt' and the string must be 255 characters or less.

An example of a &FNAM NAMELIST parameter group line is as follows:

```
&FNAM materialfile='newmats.txt', fuelfile='AllFuels.dat',  
fanfile='fans.txt', suppressorfile='nozzles.dat' /
```

The example &FNAM line changes the default materialfile, fuelfile, and suppressorfile names to 'newmats.txt', 'AllFuels.dat', and 'nozzles.dat', respectively. The fanfile is also set to the default 'fans.txt' in the NAMELIST group, though it is unnecessary to do so.

4.2.5 Tabular Input Data (&CURV)

The &CURV NAMELIST group is used by other NAMELIST groups as a means of collecting tabular data. Examples of tabular data that may be entered using the &CURV

NAMELIST group parameters include thermal conductivity or thermal heat capacity that vary with temperature, user-defined fire with a transient heat release rate, and fan performance curves. The `&CURV` parameters are read when directed to by a referring NAMELIST parameter group.

The parameters associated with this NAMELIST group are required; however, the group itself is optional. The `&CURV` parameters are as follows:

REQUIRED PARAMETER WHEN A `&CURV` NAMELIST GROUP IS USED

<code>id</code>	(Text string) Name of <code>&CURV</code> data. Used to associate the <code>&CURV</code> data with a referring NAMELIST group. The string must be unique for each set of <code>&CURV id</code> parameters and is used for all lines used to enter the tabular data for a single function or rule. There is no default and the maximum number of characters in the <code>id</code> string is 40.
-----------------	--

The following parameter is optional; however, in nearly all instances the user will supply the appropriate input values:

OPTIONAL PARAMETER

<code>xy (2)</code>	(Real number array) Array of two real numbers separated by a comma. The two numbers are the independent and dependent variables of the underlying function or rule used to generate data, respectively. Note that the independent variable (<code>x</code>) must be in ascending order. The default value is 0.0 for both elements of the <code>xy</code> array.
---------------------	--

A table is created by using multiple `&CURV` lines each with the same `id`. The number of tabular entries is limited either by the particular NAMELIST parameter the data are being assigned to or the size of an integer (2^{15}), whichever is smaller. Note that the number of `&CURV` lines is not declared in the `&CNTR` NAMELIST parameter. Also, multiple NAMELIST parameter groups may access the same `&CURV` data provided that they reference the same `&CURV id`. Typically, the `&CURV` data used to generate a particular table are located beneath the NAMELIST parameter group that references the `&CURV` data. Refer to subsequent sections for specific details of NAMELIST groups that may use the `&CURV` input format.

The `&CURV` parameter group may be located in the primary input file or in any of the `&FNAM` auxiliary input files associated with NAMELIST parameters that use tabulated data; however, it must be located in the same file as the referring NAMELIST group.

An example of a `&CURV` NAMELIST parameter group line is as follows:

```
&CURV id='A36 steel thermal conductivity', xy=0.0, 40.0 /
&CURV id='A36 steel thermal conductivity', xy=100.0, 36.0 /
```

```
&CURV id='A36 steel thermal conductivity', xy=500.0, 30.0 /
&CURV id='A36 steel thermal conductivity', xy=1000.0, 23.0 /
```

which describes a table that may be referenced by another NAMELIST group using the text string 'A36 steel thermal conductivity' and containing four pairs of independent-dependent variables.

4.2.6 Input Conditional Control (&CTRL)

The &CTRL NAMELIST group is used to control the status of various types of devices, namely doors, hatches, fans, sprinklers, nozzles, and other active suppression components. The control functions are accessed by the specific NAMELIST group that specifies the device being controlled using the &CTRL id. There must be a &CTRL parameter group associated with devices set by the &RFAN and &SUPR NAMELIST groups. Doors and hatches, set using the &JUNC NAMELIST parameter group, may or may not use a &CTRL NAMELIST line. Subsequent sections describe these NAMELIST parameter groups in detail. Currently, the &CTRL parameter group only turns devices on and off or open and shut. It is expected that additional functions will be available in this group that may be used to reduce or increase characteristic functions of various devices, such as flow rates and concentrations.

One parameter associated with the &CTRL group is required when the &CURV is used:

REQUIRED PARAMETER WHEN A &CTRL NAMELIST GROUP IS USED

id	(Long integer) Identification number for an individual &CTRL group line. This number is referenced by other NAMELIST parameter groups to access the control functions set by the &CTRL line. The id must be unique among other &CTRL id parameters used in a particular FSSIM simulation and greater than zero.
----	---

There is no requirement for the &CTRL id parameter to be sequential or in ascending or descending order. Any valid integer greater than zero and unique among other &CTRL id parameters is acceptable.

The remaining parameters are optional; however, depending on the particular device being controlled, they may require setting. The following is a complete list of the optional parameters that may be set on the &CTRL parameter group line:

OPTIONAL PARAMETERS

ctype	<p>(Integer) Type of control. The default is 1. 1 – On or off. 2 – Indicates an AND function. Used with <code>inputvar = 10</code> or <code>11</code>. Other types not yet implemented.</p>
trip	<p>(Logical) <code>.TRUE.</code> indicates devices changes state once; <code>.FALSE.</code> indicates device may change states more than once. The default state is <code>.TRUE.</code></p>
initial	<p>(Logical) <code>.TRUE.</code> indicates device is initially on; <code>.FALSE.</code> indicates device is initially off. Used when <code>ctype</code> is set to 1. The default state is <code>.FALSE.</code></p>
inputvar	<p>(Integer) Input variable used to change the state of the device: 1 – time (<code>setpoint</code> units are seconds). 2 – compartment temperature (<code>setpoint</code> units are K). 3 – duct temperature (<code>setpoint</code> units are K). 4 – compartment heat release rate (<code>setpoint</code> units are W). 5 – compartment oxygen mass fraction (<code>setpoint</code> units are kg/kg). 7 – Time delay from trip of control function (s) 8 – compartment surface temperature (<code>setpoint</code> units are K) 10 – a detector (<code>setpoint</code> is not used) 11 – dual control function</p> <p>If the <code>inputvar</code> is 10, the condition that causes a state change is detector activation. The default is 1 (time).</p>
setpoint	<p>(Real number) The condition (e.g. a time, temperature, or heat release rate) that causes the device to change state. The type of condition is set by <code>inputvar</code>. Default is 1.0.</p>
location(2)	<p>(Long integer) The location (device, duct, or compartment identification number [<code>id</code>]) of the <code>inputvar</code> that is used to change the state of the device. Not used if <code>inputvar</code> is set to 1. Otherwise:</p> <ul style="list-style-type: none"> • <code>&COMP id</code> parameter value if <code>inputvar</code> is set to 2, 4, or 5 (location is a compartment). Only <code>location(1)</code> is set. • <code>&RDCT id</code> parameter value if <code>inputvar</code> is set to 3 (location is a inside a duct). Only <code>location(1)</code> is set. • <code>&DTCT id</code> parameter if <code>inputvar</code> is set to 10 (location is a specific detector). Only <code>location(1)</code> is set. • <code>&CTRL id</code> parameter if <code>inputvar</code> is set to 7 (location is a control function). Only <code>location(1)</code> is set. • 2 <code>&CTRL id</code> parameters if <code>inputvar</code> is set to 11 (locations are two control functions to be ANDed). Both <code>location(1)</code> and <code>location(2)</code> are set. <p>The default is 1.</p>

outputvar	(Integer). Output parameter used when <code>ctype</code> is not set to 1. Not yet implemented. Default is 1.
maxbin	(Integer). Function value when device is on. For use when <code>ctype</code> is not set to 1. Not yet implemented. Default is 1.
minbin	(Integer). Function value when device is off. For use when <code>ctype</code> is not set to 1. Not yet implemented. Default is 0.

As noted, `&CTRL` only turns devices on or off in this version of FSSIM; however, it is expected that it may be used to perform other types of function modifications. As such, parameters that are not yet implement but are expected to be used for this purpose are listed above for completeness.

Typically, a sprinkler or suppression device will be actuated using conditions or a detector state within the space where the suppression device is located. Fans are typically turned on or off based on a detection state in a compartment or duct or a duct temperature condition. Note that a device may initially be off then turned on and *vice versa*. Also note that when the `inputvar` is set to 1 (time), the `trip` parameter is inherently `.TRUE.`, though it may actually be set to `.FALSE.`

Multiple NAMELIST parameter groups may access the same `&CTRL` parameter group data provided they reference the same `&CTRL id`. Typically, each `&CTRL` parameter group is located beneath the NAMELIST parameter group that references it in the input file, though this is not a requirement.

An example of a `&CTRL` NAMELIST parameter group line is as follows:

```
&CTRL id=3, ctype=1, inputvar=1, initial=.FALSE., setpoint=120.0 /
```

The above `&CTRL` line has an `id` reference of 3, the control is on/off with the device initially being off, and the device is turned on at 120 seconds. Other NAMELIST groups thus access this control using the integer 3. The `&CTRL` example is typical of an exhaust or supply fan.

4.3 Compartments

Compartments are defined using parameter groups that set the dimensions and volume of a space, the exposed materials and surface area, and the dimensions of openings between two spaces and between a space and the surroundings that lie beyond the modeled domain (ambient).

4.3.1 Compartment Definition (`&COMP`)

The `&COMP` group allows the user to set the physical dimensions of a compartment. One `&COMP` group is required for each compartment in the simulated configuration. There must be at

least one compartment for FSSIM to successfully run. There is one parameter in each &COMP group line that must be set:

REQUIRED PARAMETER

id	(Long integer) Identification number for an individual &COMP parameter group line. The id must be unique among other &COMP parameter id values used in a particular FSSIM simulation and greater than zero. There is no default value.
----	--

There is no requirement for the &COMP id parameter to be sequential or in ascending or descending order. Any valid integer greater than zero and unique among other &COMP id parameters is acceptable.

The remaining parameters are optional, though in most instances compartment dimensions other than the default values will be set. The following is a complete list of the optional parameters that may be set on the &COMP parameter group line:

OPTIONAL PARAMETERS

compname	(Text string) Name of compartment. The default is 'compartment id' where id is the compartment identification number. The compartment name is used primarily for labeling output data. The maximum number of characters for the compartment name is 40.
dx	(Real number) Nominal compartment dimension along the keel or in the direction parallel to the x-axis (m). The default value is 1.0 m.
dy	(Real number) Nominal compartment dimension along the beam or in the direction oriented parallel to the y-axis (m). The default value is 1.0 m.
dz	(Real number) Nominal compartment height or dimension parallel to the z-axis (m). The default value is 1.0 m.
z	(Real number) Nominal elevation of floor above reference datum (m). The default is 0.0 m.
vol	(Real number) Free air volume in compartment (m ³). The default is 1.0 m ³ .
pres	(Real number) Initial pressure in compartment (Pa). The default is pamb set in the &EXEC NAMELIST parameter group.
rho	(Real number) Initial density in compartment (K). The default is calculated from tamb set in the &EXEC NAMELIST parameter group. Note do not specify both temp and rho. If both are set, the rho will override the temp.

temp	(Real number) Initial temperature in compartment (K). The default is t_{amb} set in the &EXEC NAMELIST parameter group. Note do not specify both temp and rho. If both are set, the rho will override the temp.
usetype	(Long integer) &USES NAMELIST group identification number (id) of a &USES line that describes the combustible fuel load and the type of fuel (&FUEL id) in the compartment. A value of -1 indicates that there is no combustible fuel load or that there is an initial fire in the compartment. The default value is -1.
zone	(Long integer) The zone number where the compartment is located. The default is calculated by FSSIM and the value entered here is overridden regardless. This parameter should thus not be used.

The free air volume may be less than or greater than the compartment volume calculated from the nominal dimensions dx , dy , and dz . The reference datum for the parameter z is the elevation that the ambient pressure (p_{amb}) and temperature (t_{amb}) set in the &EXEC group corresponds to.

Multiple compartments may use the same &USES line (the usetype parameters need not be unique among &COMP lines).

An example of a &COMP NAMELIST parameter group line is as follows:

```
&COMP id=1, z=1.05, dx=7.62, dy=3.1, dz=3.76, vol=88.82, temp=311.2,
compname='sample space 1' /
```

The above &COMP line establishes a compartment that is nominally 7.62 m by 3.1 m by 3.76 m high and has a free air volume of 88.82 m³. The floor of this compartment is located 1.05 m above the location where the ambient pressure and ambient temperature are defined on the &EXEC NAMELIST line. The initial temperature of this space is 311.2 K, possibly different from the ambient temperature, and the compartment name is 'sample space 1'.

4.3.2 Compartment Opening Definition (&JUNC)

The &JUNC group is used to set vent opening flow paths between compartments and other compartments and between compartments and the surrounding ambient space. Openings set using the &JUNC NAMELIST group are optional. They may be horizontal as in a deck/overhead boundary or vertical as in a bulkhead/wall boundary. Horizontal openings allow vertical flow and vertical vents allow horizontal flow. Flows through openings set using the &JUNC NAMELIST group are not directly forced. Flows that are forced via a mechanical system or systems are set using the &RDCT, &RNOD, and &RFAN NAMELIST groups.

If an opening is declared using a &JUNC parameter group, then three parameters are required to be set:

REQUIRED PARAMETERS WHEN A &JUNC NAMELIST GROUP IS USED

id	(Long integer) Identification number for an individual &JUNC parameter group line. The id must be unique among other &JUNC parameter id values used in a particular FSSIM simulation and greater than zero. There is no default value.
location(2)	(Long integer array) An array of two long integers separated by a comma. The array references the compartment identification numbers (id in a &COMP NAMELIST group) of the compartments connected by the opening. A value of -1 for one of the values indicates the surrounding ambient space. At least one compartment id parameter used on an &COMP entry must be specified, and a compartment must not be connected to itself (<i>i.e.</i> , parameter array elements must not be the same). There are no default values.
height	(Real number) Opening dimension corresponding to the height for vertical openings (m). The default value is set 0.0 m to avoid causing memory allocation errors; however, for a vertical opening this parameter must be set to a positive non-zero value to correctly simulate flow through the opening. For a horizontal opening this value should remain 0.0 m.

There is no requirement for the &JUNC id parameter to be sequential or in ascending or descending order. Any valid integer greater than zero and unique among other &JUNC id parameters is acceptable. There is also no required order for the location parameter array elements. If an opening is connected to ambient (*i.e.*, the location parameter array element is set to -1), it may be the first or second element in the array.

The remaining &JUNC parameters are optional:

OPTIONAL PARAMETERS

juncname	(Text string) Name of junction. The default is 'junction id' where id is the junction identification number. The junction name is used primarily for labeling output data. The maximum number of characters for the junction name is 40.
elevation	(Real number) Vertical elevation of the opening using the same coordinate system as for the compartment elevations. This elevation corresponds to the bottom of the vent for a vertical opening and the height of the plane at which a vent is located for a horizontal opening. The default value is set 0.0 m to avoid causing memory allocation errors; however, this parameter must be set to correctly simulate flow through the opening.
area	(Real number) The flow area of the vent (m ²). The default value is 1.0 m ² .

door	(Long integer) Control function &CTRL id used to open or close a vent. A value of -1 indicates that the vent is always open. The default value is -1.
orientation	<p>(Short integer) The orientation of the opening with respect to the first element of the location array:</p> <ul style="list-style-type: none"> 1 – overhead or in the plane with maximum compartment z-dimension. 2 – deck/floor or in the plane with minimum compartment z-dimension. 3 – fore or in the plane with maximum compartment x-dimension. 4 – aft or in the plane with minimum compartment x-dimension. 5 – port/larboard or in the plane with maximum compartment y-dimension. 6 – starboard or in the plane with minimum compartment y-dimension. <p>The default value is 3 (fore opening).</p>
bidirectional	(Logical) .TRUE. indicates that the opening has bidirectional flow and .FALSE. indicates that the flow direction in an opening is one-way. The default state is .TRUE.
vflow	(Logical) .TRUE. indicates that opening allows vertical flow, and .FALSE. indicates that the opening does not allow vertical flow. The default state is .TRUE. if the orientation is 1 or 2.
kloss	<p>(Real number) The flow loss coefficient for discharge through an opening. This term may be calculated using the following equation [Sabersky <i>et al.</i>, 1971; Brater <i>et al.</i>, 1996]:</p> $K = \frac{1}{c_d^2}$ <p>where K is the flow loss coefficient and c_d is the discharge coefficient. The discharge coefficient is a function of the opening geometry and the Reynolds Number. The discharge coefficient is tabulated for various types of openings [<i>e.g.</i>, Idelchik, 1986; Sabersky <i>et al.</i>, 1971; Brater <i>et al.</i>, 1996]. The discharge coefficient for typical openings in compartments typically between 0.6-0.7 [Klote, 1995]. The default value for kloss is 2.04, which corresponds to a discharge coefficient of about 0.7.</p> <p>In the case of an opening that spans a compartment wall (such as might occur when breaking a hallway into multiple compartments), kloss should be set to zero.</p>

length	(Real number) The inertial length associated with an opening (m). The inertial length corresponds to an acceleration distance of a fluid mass located in two volumes (presumably with different pressure profiles) connected by an opening. The recommended (and default) inertial length is the horizontal or vertical distance between the center of two connected volumes, or if one volume is ambient, twice the horizontal or vertical distance between the center of the compartment to the opening.
zone	(Long integer) The zone number where the compartment is located. The default is calculated by FSSIM and the value entered here is overridden regardless. This parameter should thus not be used.
width	(Real number) The width of the opening (m). The default is \sqrt{A} (m), where A is the flow area of the vent, <code>area</code> . This parameter is currently not used by FSSIM.

If an opening has vertical flow, then the orientation of the opening must either be in the deck/floor or the overhead and `vflow` must be set to `.TRUE.` Otherwise, vertical flow will not be permitted through the opening.

Most compartment vents should be modeled as bidirectional. Some cases may arise where the flow direction is expected to be in one direction only, such as spaces that are pressurized. When this condition arises, setting the `bidirectional` parameter to `.FALSE.` may reduce the number of calculations performed by FSSIM and possibly decrease the solution time. This parameter should not be set to `.FALSE.` if it is not known whether the flow is bidirectional or if it is possible that bidirectional flow will be result.

The inertial length is used in the FSSIM calculation to influence the acceleration of a fluid as it moves toward and through an opening. The default inertial length is based on other work with such flow calculations [Guantt *et al.*, 2000] and should be adequate for most configurations. Situations that may require modification of this parameter could include spaces connected by an elongated opening such that the acceleration distance is greater than what is determined by default or cases where the acceleration effects on the solution are not desired, in which case the inertial length should be set to a small positive value.

FSSIM will simulate a wide variety of vent conditions, including improperly specified vent height and elevations (*i.e.*, height plus elevation exceed vertical dimension of compartment). As such, considerable care is necessary to ensure the vent specified accurately represents the configuration desired.

An example of a `&JUNC NAMELIST` parameter group line is as follows:

```
&JUNC id=7, location=10,-1, height=2.13, area=1.94, elevation=0.0,
orientation=3, vflow=.FALSE., juncname='Ordinary door' /
```

The above `&JUNC` line sets an opening between a compartment with a `&COMP id` parameter equal to 10 and the surrounding ambient (-1). The opening identification `id` is 7 and is referred

to in output as ‘Ordinary door’. The opening is located in the forward compartment boundary and is thus vertical. It is 2.13 m tall, the bottom is located at elevation of 0 m, and it has an area of 1.94 m². The flow is bidirectional (default condition) and not vertical (by default and necessity [orientation set to 3, a vertical boundary], but also specified with `vflow` parameter on example `&JUNC` line). The vent is always open (the default `door` parameter state of `.FALSE.` is used). The default flow loss coefficient of 2.04 is used and the inertial length is calculated by FSSIM.

4.3.3 Compartment Surface Definition (`&SURF`)

The `&SURF` group is used to define the boundary surfaces in a space for calculating thermal flows. A minimum of six surfaces must be set for each compartment set using the `&COMP` parameter group, one for each possible orientation (overhead/ceiling, deck/floor, fore, aft, port/larboard, and starboard). FSSIM will run if less than the minimum number of surfaces is specified; however, the results will not be correct. The user should ensure proper surface definition.

Three parameters are required to be set on each `&SURF` line:

REQUIRED PARAMETERS

<code>id</code>	(Long integer) Identification number for an individual <code>&SURF</code> parameter group line. The <code>id</code> must be unique among other <code>&SURF</code> parameter <code>id</code> values used in a particular FSSIM simulation and greater than zero. There is no default value.
<code>stype</code>	(Short integer) The type of surface: <ul style="list-style-type: none"> 1 – The surface is between two compartments or a compartment and the surrounding ambient. 2 – The surface is between a compartment and a duct space. There is no default value. Note that compartment-duct surfaces are not yet implemented.

location	<p>(Long integer array) An array of two long integers separated by a comma. The array references the compartment identification numbers (<code>id</code> in a <code>&COMP</code> NAMELIST group) if <code>stype</code> is set to 1 or a compartment identification number, and a duct identification number (<code>id</code> in a <code>&RDCT</code> NAMELIST group) if <code>stype</code> is set to 2. The duct <code>id</code> value must be the second element in the parameter array when <code>stype</code> is set to 2. A compartment <code>id</code> value of <code>-1</code> indicates the surrounding ambient space. If <code>stype</code> is set to 1 at least one compartment <code>id</code> value must be positive and the parameter array elements must not be equal (<i>i.e.</i>, a surface within a compartment). There are no default values. A surface shared between two compartments should only be specified once. The order of the compartment <code>id</code> parameters is not significant; however the <code>orientation</code> parameter must be properly set. Note that compartment-duct surfaces are not yet implemented.</p>
orientation	<p>(Short integer) The orientation of the surface with respect to the compartment identified by the first element of the <code>location</code> array:</p> <ul style="list-style-type: none"> 1 – overhead/ceiling or in the plane with maximum compartment <i>z</i>-dimension. 2 – deck/floor or in the plane with minimum compartment <i>z</i>-dimension. 3 – fore or in the plane with maximum compartment <i>x</i>-dimension. 4 – aft or in the plane with minimum compartment <i>x</i>-dimension. 5 – port or in the plane with maximum compartment <i>y</i>-dimension. 6 – starboard or in the plane with minimum compartment <i>y</i>-dimension. <p>There is no default value for this parameter.</p>

There is no requirement for the `&SURF id` parameter to be sequential or in ascending or descending order. Any valid integer greater than zero and unique among other `&SURF id` parameters is acceptable. There is also no required order for the `location` parameter array elements when `stype` is set to 1. If the surface is adjacent to ambient (*i.e.*, one of the `location` parameter array elements is set to `-1`), it may be the first or second element in the array. If `stype` is set to 2, then the duct identification (`&RDCT id`) must be the second element in the parameter array. The first element may be either a compartment identification number (`&COMP id`) or `-1` (the ambient surroundings). Surfaces must separate different compartment spaces if `stype` is set to 1, such that a surface is not located in a single compartment or space.

A compartment must have at least six surfaces defined, one for each orientation, in order for the radiation heat transfer calculation to function properly. The surfaces may be shared with another compartment or they may be between the compartment and ambient. Note that if the surface is shared, and the compartment `id` is the second element in the `location` array, then the surface orientation for the compartment is the opposite of what is specified by the `orientation` parameter.

The remaining `&SURF` parameters are optional:

OPTIONAL PARAMETERS

surfname	(Text string) Name of surface. The default is 'surface c location (1) c location (2) o orientation' where location(1) is the first compartment identification number specified in the location parameter array, location(2) is the second compartment identification number or the first duct identification number, and orientation is the orientation parameter value. The surface name is used primarily for labeling output data. The maximum number of characters for the surface name is 40.
area	(Real number) The area of the surface (m ²). The default value is 1.0 m ² .
composition	(Long integer) &CMPN NAMELIST group identification number (id) of a &CMPN line that describes the surface composition (materials). The default value is 1.
fake	(Logical) .FALSE. indicates that the surface is a solid boundary and .TRUE. indicates an open boundary that allows thermal radiation heat flow but does not perform the one-dimensional heat transfer calculation. An example of the latter type of surface would be the open end of corridor or a large vent opening in a compartment. The default state is .FALSE.
reverse_composition	(Logical) .TRUE. indicates that the composition definition for the surface is in the opposite direction of the compartment definition, <i>i.e.</i> the composition layer as defined using the first element of both the &CMPN dx and materials parameter arrays are associated with the compartment identified by second element of the location array. The default state is .FALSE.

Fake surfaces are used to calculate the radiant heat flow through openings. This may be a significant energy quantity in spaces where the temperature is high and the opening area is proportionately large. Note that there is no practical limit to the number of surfaces that may be set in a compartment. The total surface area should nominally be about equal to the boundary surface area.

An example of a &SURF NAMELIST parameter group line is as follows:

```
&SURF id=3, stype=1, location=3,2, area=1.94, orientation=3, composition=1,
surfname='Wall section' /
```

The above &SURF has an identification (id) value of 3 and is also identified by the text string 'Wall section'. The surface is between two compartments with id parameters set to 3

and 2. The surface is located in the plane with the maximum x -dimension (fore) of the compartment with an `id` parameter set to 3 and has a surface area of 1.94 m². The heat transfer calculations will be performed for this surface (the default state for the `fake` parameter [`.FALSE.`] is used) and the thickness and specific materials that make up the surface boundary are described by a `&CMPN` line with an `id` parameter of 1. If the `fake` parameter is set to `.TRUE.`, then the `composition` parameter is not used.

4.3.4 Compartment Surface Composition (`&CMPN`)

The `&CMPN` NAMELIST group is used to set the type, number, and thickness of materials that make up a surface boundary. The `&CMPN` NAMELIST group is optional; however, if there is a surface that is not fake, then at least one `&CMPN` line must be provided with an identification number that matches the `&SURF composition` parameter value.

A `&CMPN` NAMELIST line has one parameter that must be set:

REQUIRED PARAMETER WHEN A `&CMPN` NAMELIST GROUP IS USED

<code>id</code>	(Long integer) Identification number for an individual <code>&CMPN</code> parameter group line. The <code>id</code> must be unique among other <code>&CMPN</code> parameter <code>id</code> values used in a particular FSSIM simulation and greater than zero. There is no default value.
-----------------	--

There is no requirement for the `&CMPN id` parameter to be sequential or in ascending or descending order. Any valid integer greater than zero and unique among other `&CMPN id` parameters is acceptable.

The remaining `&CMPN` parameters are optional:

OPTIONAL PARAMETERS

<code>cmpnname</code>	(Text string) Name of composition. The composition name is used primarily to identify the appropriate composition input data to read from a file specified by <code>&FNAM compositionfile</code> parameter. The composition name is also used for labeling output data. The maximum number of characters for the composition name is 40. The default is <code>'composition id'</code> , where <code>id</code> is the composition identification number.
-----------------------	---

fileread	(Logical) <code>.TRUE.</code> indicates that the composition data should be read from the file set by the <code>&FNAM compositionfile</code> parameter and <code>.FALSE.</code> indicates that the composition data are contained within the <code>&COMP id</code> line located in the main input file. The default state is <code>.TRUE.</code> If the <code>cmpnname</code> parameter is not set (<i>i.e.</i> , the default is used), then the composition data will <i>not</i> be read from the file set by the <code>&FNAM compositionfile</code> parameter.
layers	(Short integer) The number of material layers that make up the composition (and referencing surface boundary). The default value is 1 and the maximum number of layers permitted is 50.
materials	(Long integer array) An array of long integers separated by commas consisting of the material identification numbers (<code>&MTRL id</code> parameter values) that make up each layer of the composition. The number of elements must be equal to the <code>layers</code> parameter value and thus not exceed 50. The first material identified in the array will have a surface located in the first compartment set using the first element of the <code>&SURF location</code> parameter. The last material will have an exposed surface located in the second compartment or in the duct set using the second element of the <code>&SURF location</code> parameter array. The intervening materials will be sequentially adjacent; that is, <code>materials</code> array element 2 will be physically located between <code>materials</code> array elements 1 and 3. The default value for this parameter is 1.
dx	(Real number array) An array of real numbers separated by commas consisting of the thickness of each layer (m). The number of elements must be equal to the <code>layers</code> parameter value and thus not exceed 50. The first value corresponds to the first element in the <code>materials</code> array, the second value corresponds the second element, and so forth. The default value is 0.1 m.

Input associated with the `&CMPN NAMELIST` group may be located in a separate input file with a name specified using the `&FNAM compositionfile` parameter. If a different input file is to be used, then the following must be set on the `&CMPN` line in the primary input file:

- `fileread` must be set to `.TRUE.` (the default condition).
- `cmpnname` must be set to a string that can be found in the file set by the `&FNAM compositionfile` parameter.
- `id` must be positive and unique among the `&CMPN id` parameters.

The input file specified by `&FNAM compositionfile` parameter should contain a `&CMPN` line with the `cmpnname` equal to that specified in the primary input file. The line should contain the additional layers, materials, and thickness parameters for the composition material. The `id` parameter must not be reset in the input file specified by `&FNAM compositionfile` parameter.

Two examples of the `&CMPN` NAMELIST parameter are provided, one in which all information is entered in the primary input file and one in which a secondary input file is used. The first example is as follows and is contained entirely within the primary input file:

```
&CMPN id=8, fileread=.FALSE., cmpnname='Ins-steel-ins-paint', layers=4,
materials=1,2,1,3, dx=0.05, 0.01, 0.05, 0.001 /
```

The above `&CMPN` line is not read from a file. It has an identification (`id`) value of 8 and is also identified by the text string `'ins-steel-ins-paint'`. The composition has four layers. The first and third layer are 0.05 m thick and consist of the material with a `&MTRL id` of 1. The second layer is 0.01 m thick and uses the material with a `&MTRL id` of 2, and the fourth layer is 0.001 m thick and uses the material with a `&MTRL id` of 3.

The second example uses the input file set by the `&FNAM compositionfile` parameter to enter the composition layer data. The following line appears in the primary input file:

```
&CMPN id=8, fileread=.TRUE., cmpnname='Ins-steel-ins-paint' /
```

The following line would then appear in the file set by the `&FNAM compositionfile` parameter:

```
&CMPN cmpnname='Ins-steel-ins-paint', layers=4, materials=1,2,1,3, dx=0.05,
0.01, 0.05, 0.001 /
```

The second `&CMPN` NAMELIST example is identical to the first example in terms of the layer information entered.

4.3.5 Surface Composition Material Definition (`&MTRL`)

The `&MTRL` NAMELIST group is used to enter the thermal properties of the surface boundary materials. The `&MTRL` NAMELIST group is optional; however, if there are one or more surfaces that are not fake, then there must be at least one `&CMPN` line and therefore at least one `&MTRL` line. The `&MTRL` line or lines must have an identification number(s) that corresponds to the `&CMPN materials` parameter value(s).

A `&MTRL` NAMELIST line has one parameter that must be set:

REQUIRED PARAMETER WHEN A &MTRL NAMELIST GROUP IS USED

id	(Long integer) Identification number for an individual &MTRL parameter group line. The id must be unique among other &MTRL parameter id values used in a particular FSSIM simulation and greater than zero. There is no default value.
----	--

There is no requirement for the &MTRL id parameter to be sequential or in ascending or descending order. Any valid integer greater than zero and unique among other &MTRL id parameters is acceptable.

The remaining &MTRL parameters are optional:

OPTIONAL PARAMETERS

matname	(Text string) Name of material. The material name is used primarily to identify the appropriate material input data to read from a file specified by &FNAM materialfile parameter. The material name is also used for labeling various output data. The maximum number of characters for the material name is 40. The default is 'material id', where id is the material identification number.
fileread	(Logical) .TRUE. indicates that the material data should be read from the file set by the &FNAM materialfile parameter and .FALSE. indicates that the material data are contained within the &MTRL id line located in the main input file. The default state is .TRUE. If the matname parameter is not set (<i>i.e.</i> , the default is used), then the material data will <i>not</i> be read from the file set by the &FNAM materialfile parameter.
k	(Real number) The thermal conductivity of the material (W/m-K). The default value is 40.0 W/m-K.
cp	(Real number) The heat capacity of the material (J/kg-K). The default value is 486.0 J/kg-K.
rho	(Real number) The density of the material (kg/m ³). The default value is 7,753 kg/m ³ .
e	(Real number) The surface emissivity of the material. The default value is 0.9.

ktable	<p>(Logical) <code>.TRUE.</code> indicates that the thermal conductivity data are input as a tabular function of temperature using one or more (up to 500) <code>&CURV</code> lines. <code>.FALSE.</code> indicates that the thermal conductivity is constant and is equal to the value set by the <code>k</code> parameter.</p> <p>The thermal conductivity is also assumed constant for temperatures that are above or below the temperature range set by the <code>&CURV</code> lines. If the temperature is less than the minimum temperature value, then the thermal conductivity is assumed equal to the thermal conductivity specified for the minimum temperature. The reverse is true if the temperature is greater than the maximum temperature value.</p> <p>The default state is <code>.FALSE.</code></p>
cptable	<p>(Logical) <code>.TRUE.</code> indicates that the heat capacity data are input as tabular function of temperature using one or more (up to 500) <code>&CURV</code> lines. <code>.FALSE.</code> indicates that the heat capacity is constant and is equal to the value set by the <code>cp</code> parameter.</p> <p>The heat capacity is also assumed constant for temperatures that are above or below the temperature range set by the <code>&CURV</code> lines. If the temperature is less than the minimum temperature value, then the heat capacity is assumed equal to the heat capacity specified for the minimum temperature. The reverse is true if the temperature is greater than the maximum temperature value.</p> <p>The default state is <code>.FALSE.</code></p>
kname	<p>(Text string) The name of the <code>&CURV</code> line(s) that contain the thermal conductivity data. There is no default and the maximum number of characters is 40. This parameter is not used if <code>ktable</code> is set to <code>.FALSE.</code></p>
cpname	<p>(Text string) The name of the <code>&CURV</code> line(s) that contain the heat capacity data. There is no default and the maximum number of characters is 40. This parameter is not used if <code>cptable</code> is set to <code>.FALSE.</code></p>

The `&MTRL` parameters may be read from a separate input file as specified by the `&FNAM materialfile` parameter. The method is identical to that described for the `&CMPN NAMELIST` parameter group in Section 4.3.4 and is not repeated here.

The thermal conductivity and heat capacity of a material entered using the `&MTRL NAMELIST` group may be constant or a function of temperature. If constant, then `ktable` or `cptable` must be set to `.FALSE.` If either is a function of temperature, then `ktable` and/or `cptable` must be set to `.TRUE.` and one or more `&CURV` lines with the `id` parameter set to `kname` or `cpname` must be provided. The `&CURV` lines must also be located in the same file in which the `cptable` or `ktable` parameters are set. Refer to Section 4.2.4 for a description of the `&CURV NAMELIST` group.

The following is an example of a `&MTRL` NAMELIST group and the associated `&CURV` lines that use the input file set by the `&FNAM` `materialfile` parameter to enter the composition layer data:

```
&MTRL id=8, fileread=.TRUE., matname='Insulation' / This is located in the
primary input file
```

The following lines would then appear in the file set by the `&FNAM` `materialfile` parameter:

```
&MTRL matname='Insulation', e=0.91, rho=128.0, cp=1157.0, cptable=.FALSE.,
ktable=.TRUE., kname='Ins_k' /
&CURV id='Ins_k', xy= 0.0, 0.03 /
&CURV id='Ins_k', xy= 1000.0, 0.25 /
```

The example above indicates that the material property data identified with an `id` parameter of 8 and a `matname` of 'Insulation' should be read from the input file set by the `&FNAM` `materialfile` parameter. The emissivity is set to 0.91, the density is 128 kg/m³, and the heat capacity is a constant 1,157 J/kg-K. The thermal conductivity varies with temperature and the temperature-conductivity data are entered on two `&CURV` lines with the `id` parameter set to 'Ins_k'. The `&CURV` lines are located in the file set by the `&FNAM` `materialfile` parameter.

4.4 Mechanical Ventilation

Forced ventilation conditions are simulated using duct segments, duct nodes, and fans. Ducts connect two nodes and are assumed uniform in terms of cross section and friction loss between nodes. Fans are added to duct segments and are essentially represented as a discontinuous pressure change that is a function of the local pressure. Mechanical ventilation may be used to simulate mechanical supply, exhaust, or other types of inter-compartment connections involving the transfer of air through ductwork.

4.4.1 Mechanical Ventilation Nodes (`&RNOD`)

The `&RNOD` NAMELIST group is used to enter the node locations of duct segments. The nodes are used to connect ducts to each other, to compartments, and to the ambient surroundings. The `&RNOD` NAMELIST group is optional; however, at least two `&RNOD` lines are required if there is a `&RDCT` NAMELIST group line.

Nodes should be defined at the following locations within a ductwork system:

- At junctions of three or more ducts.
- Between multiple fans
- At compartment or ambient terminal locations.

In addition, nodes may also be used where ducts change shape, cross-sectional area, or where the flow changes direction; however, it is not necessary to do so provided the flow loss coefficients for the duct and all associated fittings is known.

If nodes are set using the `&RNOD` parameter group, then three parameters are required to be set on each `&RNOD` line:

REQUIRED PARAMETERS WHEN A `&RNOD` NAMELIST GROUP IS USED

<code>id</code>	(Long integer) Identification number for an individual <code>&RNOD</code> parameter group line. The <code>id</code> must be unique among other <code>&RNOD</code> parameter <code>id</code> values used in a particular FSSIM simulation and greater than zero. There is no default value.
<code>nducts</code>	(Integer) The number of ducts connected at the node. If the node is a terminus (<code>terminal</code> set to <code>.TRUE.</code>), then <code>nducts</code> must be set to 2. The maximum number of ducts that may be connected to a single node is 30. There is no default value.
<code>ducts</code>	(Long integer array) An array of long integers, separated by commas, containing the identification numbers (<code>&RDCT id</code> parameter values) of the ducts connected at the node. If the node is a terminus (<code>terminal</code> set to <code>.TRUE.</code>), then the first element of the array contains the compartment identification number (<code>&COMP id</code> parameter value) or a <code>-1</code> (ambient surroundings). There are no default values for the array elements.

There is no requirement for the `&RNOD id` parameter to be sequential or in ascending or descending order. Any valid integer greater than zero and unique among other `&RNOD id` parameters is acceptable. The ducts connected at the node must be different; the same duct should not be multiply connected at a node.

The remaining `&RNOD` parameters are optional:

OPTIONAL PARAMETERS

<code>nodename</code>	(Text string) Name of duct node. The duct node name is used primarily for labeling various output data. The maximum number of characters for the duct node name is 40. The default is <code>'ductnode id'</code> , where <code>id</code> is the ductnode identification number, if <code>terminal</code> is <code>.FALSE.</code> , and <code>'ductterminal-c ducts(1)'</code> , where <code>ducts(1)</code> is the first element of the <code>ducts</code> array, if <code>terminal</code> is <code>.TRUE.</code>
<code>elevation</code>	(Real number) The elevation of the node above the reference datum (m). The default value is 0.0 m.

terminal	(Logical) <code>.TRUE.</code> indicates that the node is a duct terminus (compartment or ambient surroundings) and <code>.FALSE.</code> indicates that the node is not a terminus. The default state is <code>.FALSE.</code>																										
loss	<p>(Real number array) An array of real numbers, separated by commas, that correspond to the pressure loss coefficients for branch flow along all possible flow paths at the node. The number of elements in the array is equal to the number of branch flow permutations possible at the node junction. The number of permutations is calculated using:</p> $n_{elem} = N \cdot (N - 1)$ <p>where n_{elem} is the number of elements in the loss array and N is the number of ducts connected to the node (<code>nducts</code>). The maximum number of elements in the array is 870 because the <code>nducts</code> parameter is limited to 30. The sequence of the array is based on the order of the <code>&RDCT id</code> parameters that constitute the <code>ducts</code> parameter array. The order is as follows:</p> <table border="1" data-bbox="529 873 1393 1465"> <thead> <tr> <th>Duct that flow issues from</th> <th>Duct that flow enters into</th> <th>loss array element</th> </tr> </thead> <tbody> <tr> <td rowspan="3">ducts array element 1</td> <td>ducts array element 2</td> <td>1</td> </tr> <tr> <td>ducts array element 3</td> <td>2</td> </tr> <tr> <td>ducts array element <code>nducts</code></td> <td></td> </tr> <tr> <td rowspan="4">ducts array element 2</td> <td>ducts array element 1</td> <td></td> </tr> <tr> <td>ducts array element 3</td> <td></td> </tr> <tr> <td>ducts array element 4</td> <td></td> </tr> <tr> <td>ducts array element <code>nducts</code></td> <td></td> </tr> <tr> <td rowspan="3">ducts array element <code>nducts</code></td> <td>ducts array element 1</td> <td></td> </tr> <tr> <td>ducts array element 2</td> <td></td> </tr> <tr> <td>ducts array element <code>nducts - 1</code></td> <td><code>nducts@</code> <code>(nducts-1)</code></td> </tr> </tbody> </table> <p>The default value is 0.0 for all array elements.</p>	Duct that flow issues from	Duct that flow enters into	loss array element	ducts array element 1	ducts array element 2	1	ducts array element 3	2	ducts array element <code>nducts</code>		ducts array element 2	ducts array element 1		ducts array element 3		ducts array element 4		ducts array element <code>nducts</code>		ducts array element <code>nducts</code>	ducts array element 1		ducts array element 2		ducts array element <code>nducts - 1</code>	<code>nducts@</code> <code>(nducts-1)</code>
Duct that flow issues from	Duct that flow enters into	loss array element																									
ducts array element 1	ducts array element 2	1																									
	ducts array element 3	2																									
	ducts array element <code>nducts</code>																										
ducts array element 2	ducts array element 1																										
	ducts array element 3																										
	ducts array element 4																										
	ducts array element <code>nducts</code>																										
ducts array element <code>nducts</code>	ducts array element 1																										
	ducts array element 2																										
	ducts array element <code>nducts - 1</code>	<code>nducts@</code> <code>(nducts-1)</code>																									
nosolve	(Logical) <code>.TRUE.</code> indicates that the mass or volume flow rate through the junction is fixed (via <code>&RDCT mdot</code> or <code>vdot</code> parameters) and <code>.FALSE.</code> indicates that the mass flow rate should be determined. The default state is <code>.FALSE.</code>																										
zone	(Long integer) The zone number where the compartment is located. The default is calculated by FSSIM and the value entered here is overridden regardless. This parameter should thus not be used.																										

The reference datum is the elevation that corresponds to the `&EXEC pamb` and `tamb` parameter values. The pressure loss coefficients are determined from published data on the particular type of junction ('Y' tee, couple, reducer, cross, or 'Y' cross) [Idelchik, 1986]. Typically, the pressure loss coefficient is a function of the Reynolds number. However, as with other types of duct fittings, it is only weakly dependent on the Reynolds number and will thus not vary significantly over a large range of Reynolds number values.

There may be circumstances where the mass or volume flow rate through a duct is specified, as described in Section 4.4.1. In such case, setting the `nosolve` parameter to `.TRUE.` may reduce the number of calculations performed and possibly decrease the computation time for a simulation. Note that in order to conserve mass, where two or more duct segments ducts join at a node, the net mass flow must balance. As previously noted, if one flow path is left unspecified, FSSIM will determine the appropriate mass flow in the duct segment. In this case, the `nosolve` parameter should be set to `.FALSE.`

An example of a `&RNOD` NAMELIST parameter group line is as follows:

```
&RNOD id=3, nducts=2, terminal=.TRUE., ducts=-1,3, elevation=1.73, loss=0.93,  
0.93, nodename='Duct terminus at ambient' /
```

The above `&RNOD` line is identified by the `id` parameter set to 3 and the `nodename` parameter set to 'Duct terminus at ambient'. The node is a terminus for the duct with a `&RDCT id` parameter of 3 at the ambient surroundings. The elevation of this node is 1.73 m above the reference datum. The pressure loss coefficient of 0.93 applies to flow in both directions.

4.4.2 Ducts (&RDCT)

The `&RDCT` NAMELIST group is used to enter the location and physical characteristics of duct segments. The `&RDCT` NAMELIST group is optional; however, it is an essential NAMELIST group if forced ventilation is being modeled. A duct segment is located between two nodes, defined using the `&RNOD` NAMELIST group.

If one or more ducts are set using the `&RDCT` parameter group, then two parameters are required to be set on each `&RDCT` line:

REQUIRED PARAMETERS WHEN A &RDCT NAMELIST GROUP IS USED

id	(Long integer) Identification number for an individual &RDCT parameter group line. The id must be unique among other &RDCT parameter id values used in a particular FSSIM simulation and greater than zero. There is no default value.
node	(Long integer array) Array of two &RNOD id parameters defining the endpoints of the duct segment. There are no default values for the array elements.

There is no requirement for the &RDCT id parameter to be sequential or in ascending or descending order. Any valid integer greater than zero and unique among other &RDCT id parameters is acceptable. The endpoints of a duct segment defined by the node array must use different &RNOD id parameter values (*viz.*, a duct must not begin and end at the same location).

The remaining &RDCT parameters are optional:

OPTIONAL PARAMETERS

ductname	(Text string) Name of duct. The duct name is used for primarily labeling various output data. The maximum number of characters for the duct name is 40. The default is 'duct id', where id is the duct identification number.
area	(Real number) The nominal flow area of the duct (m ²). The default value is 1.0 m ² .
perimeter	(Real number) The wetted perimeter of the duct segment (m). There is no default value.
hd	(Real number) The hydraulic radius of the duct segment (m). If perimeter parameter is undefined, then the default hydraulic radius is \sqrt{A} , where A is the equal to the area parameter value. Otherwise, the default hydraulic perimeter is: $\frac{4A}{P}$ where A is equal to the area parameter value and P is equal to the perimeter value.

kloss	(Real number array) An array of numbers, separated by commas, containing the turbulent pressure loss coefficients for the duct segment and any fittings that may be located within the duct segment. The first element in the array is the coefficient corresponding to positive flow and the second element corresponds to negative (reverse) flow. Positive flow is defined as flow from the first element in the <code>node</code> array to the second element and <i>vice versa</i> for reverse flow. The pressure loss coefficient includes the pressure losses associated with flow through bends and fittings. It may also include the friction losses to the duct boundaries. If boundary friction losses are included in this parameter, then the <code>roughness</code> parameter should be set to 0.0. The default value is 0.0.
length	(Real number) The length of the duct segment (m). The default value is 1.0 m.
roughness	(Real number) The surface roughness of the duct segment (m). The default value is 0.00091 m. If the friction flow losses for the duct are included in the <code>kloss</code> parameter, roughness should be set to 0.0 m.
damper	(Long integer) Control function <code>&CTRL id</code> used to open or close a damper in a duct segment. A value of <code>-1</code> indicates that there is no damper in the duct. The default value is <code>-1</code> .
fan	(Long integer) Used to specify that a fan is located between the duct segment endpoints as defined by the first element in the <code>node</code> array to the second element. The value is equal to a <code>&RFAN id</code> value for a <code>&RFAN</code> line containing the appropriate fan information. A value of <code>-1</code> indicates that there is no fan associated with this duct segment. The default value is <code>-1</code> .
mdot	(Real number) Mass flow rate through duct (kg/s). Positive flow is from the first element in the <code>node</code> array to the second element. The mass flow is a forced condition and should only be used when there is an absence of information regarding the ductwork. The mass flow rate must be greater than <code>-999</code> kg/s. A value less than or equal to <code>-999</code> kg/s disables this feature. The default value is <code>-999</code> kg/s.
vdot	(Real number) Volume flow rate through duct (m ³ /s). Positive flow is from the first element in the <code>node</code> array to the second element. The volume flow is a forced condition and should only be used when there is an absence of information regarding the ductwork. The volume flow rate must be greater than <code>-999</code> m ³ /s. A value less than or equal to <code>-999</code> m ³ /s disables this feature. The default value is <code>-999</code> m ³ /s.
zone	(Long integer) The zone number where the compartment is located. The default is calculated by FSSIM and the value entered here is overridden regardless. This parameter should thus not be used.
round	(Logical) <code>.TRUE.</code> if the cross-section of the duct is round and <code>.FALSE.</code> if the cross section is not round. The default state is <code>.TRUE.</code> This parameter is currently not used by the mode.

The `perimeter` variable is only used to calculate the hydraulic diameter, `hd`. It is thus not necessary to specify this parameter if the value of `hd` is known.

The pressure loss coefficient is determined by summing the contribution from the various fittings that are between duct segment end points, and may include friction pressure losses to the duct boundary. Typically, the pressure loss coefficient is a function of the Reynolds number and is tabulated for the type of fitting present [e.g., Crane Company, 1991; Idelchik, 1986; Brater, 1996]. In many cases, the pressure loss coefficient is only weakly dependent on the Reynolds number and will thus not vary significantly over a large range of Reynolds numbers. FSSIM will automatically calculate the friction pressure loss as a function of the Reynolds number if a duct roughness is provided. The roughness is material specific and may be estimated from various sources [Crane Company, 1991; Idelchik, 1986; Brater, 1996].

Duct segments may contain a single fan with characteristics defined at the referenced `&RFAN` NAMELIST line. Fans are located midway between duct segment endpoints in the model and are essentially a discontinuous pressure change within the ductwork.

If the duct contains a damper, it may be opened or closed based on an input condition (time, temperature, etc.) using a `&CTRL` function.

There may be circumstances where there is inadequate information regarding the ductwork (length, friction loss, and fitting losses). In this case, it is possible to force a mass or volume flow rate through a duct segment using the `mdot` or `vdot` parameters. When either is set, the pressure loss calculations are not performed and this flow rate is maintained regardless of the duct segment endpoint conditions. Note that in order to conserve mass, where two or more duct segments join at a node, the net mass flow must balance. If either feature is used, then one flow path should remain unspecified such that the flow is calculated within FSSIM to avoid creating a situation where mass is not balanced. Both the mass flow and the volume flow must not be set within a single duct; otherwise, FSSIM will generate an error and terminate the simulation.

An example of a `&RDCT` NAMELIST parameter group line and the associated `&CTRL` line is as follows:

```
&RDCT id=3, node=1,2, area=0.607, length=3.37, kloss=3.89, hd=0.82,  
roughness=0.00055, damper=2, ductname='Duct Segment A' /  
&CTRL id=2, ctype=1, inputvar=3, location=3, initial=.FALSE., setpoint=76.0 /
```

The above `&RDCT` line is identified by the `id` parameter set to 3 and the `ductname` parameter set to 'Duct Segment A'. The duct segment endpoints are defined by nodes with the `&RNOD id` parameters set to 1 and 2. The length of the duct is 3.37 m, the nominal flow area is 0.607 m², and the hydraulic diameter is 0.82 m. The pressure loss coefficient of 3.89, which could include a valve and an elbow, but does not include friction losses at the duct boundary. The latter are calculated by FSSIM using a duct surface roughness of 0.00055 m. There is a damper with actuation criteria input using a `&CTRL` line with an `id` parameter set to 2. The referenced `&CTRL` line forces the damper to close when the temperature within the duct segment with an `id` parameter set to 3 (the referencing duct) reaches 76 °C.

4.4.3 Fans (&RFAN)

The &RFAN NAMELIST group is used to specify the fan characteristics (performance curve, control functions, and pressure loss coefficients). The &RFAN NAMELIST group is optional; however, it is an essential NAMELIST group if forced ventilation is being modeled.

Fans are simulated within a duct segment defined by a &RDCT line. If a &RFAN line is included in the model, then two parameters are required to be set on each &RFAN line:

REQUIRED PARAMETERS WHEN A &RFAN NAMELIST GROUP IS USED

id	(Long integer) Identification number for an individual &RFAN parameter group line. The id must be unique among other &RFAN parameter id values used in a particular FSSIM simulation and greater than zero. There is no default value.
control	(Long integer) Control function &CTRL id used to turn a fan on or off. There is no default value.

There is no requirement for the &RFAN id parameter to be sequential or in ascending or descending order. Any valid integer greater than zero and unique among other &RFAN id parameters is acceptable.

The remaining &RFAN parameters are optional:

OPTIONAL PARAMETERS

fanname	(Text string) Name of the fan. The fan name is used primarily for labeling various output data. The maximum number of characters for the fan name is 40. The default is 'fan id', where id is the fan identification number.
fileread	(Logical) .TRUE. indicates that the fan data should be read from the file set by the &FNAM fanfile parameter and .FALSE. indicates that the fan data are contained within the &RFAN id line located in the main input file. The default state is .TRUE. If the fanname parameter is not set (<i>i.e.</i> , the default is used), then the fan data will <i>not</i> be read from the file set by the &FNAM fanfile parameter.

f _{type}	<p>(Short integer) The type of fan in terms of pressure versus flow characteristics:</p> <ol style="list-style-type: none"> 1 – The fan produces a constant pressure increase equal to <code>delp</code> (Pa) regardless of the flow. 2 – The fan produces a pressure increase that is a quadratic function of the flow. The flow at zero pressure (<code>full</code>) and pressure at zero flow (<code>shutoff</code>) are provided. 3 – The fan flow versus pressure data are provided in tabular format using <code>&CURV</code> lines with an <code>id</code> parameter set to the text string <code>table</code>. The maximum number of pressure versus flow data points is 32,000. <p>The default value is 1.</p>
delp	(Real number) The pressure increase (Pa) at the fan when <code>f_{type}</code> is set to 1. The default value is 0.0 Pa.
full	(Real number) The volumetric flow (m ³ /s) at zero pressure when <code>f_{type}</code> is set to 2. The default value is 1.0 m ³ /s.
shutoff	(Real number) The pressure (Pa) at zero flow when <code>f_{type}</code> is set to 2. The default value is 1,000 Pa.
table	(Text string) The name of the <code>&CURV</code> line(s) that contain the flow vs. fan pressure tabular data. There is no default and the maximum number of characters is 40. This parameter is not used if <code>f_{type}</code> is set to 1 or 2.
reverse	(Logical) <code>.TRUE.</code> indicates that the fan pressure versus flow is positive in the opposite direction of positive flow and <code>.FALSE</code> indicates that the fan pressure versus flow is positive in the direction of positive flow. Positive flow is defined as flow from the first element in the <code>&RDCT node</code> array to the second element in the <code>&RDCT node</code> array of the <code>&RDCT</code> line that references this fan. The default state is <code>.FALSE.</code>
kloss	(Real number) The pressure loss coefficient associated with the fan when the fan is not operating. The default is 1.0.

The `&RFAN` variables may be read from a separate input file as specified by the `&FNAM fanfile` parameter. The method is identical to that described for the `&CMPN NAMELIST` parameter group in Section 4.3.4 and is not repeated here.

If the velocity versus pressure information is entered using `&CURV` lines (*viz.*, `ftype` is set to 3), then the profile must vary with pressure and it should resemble a typical fan curve for a centrifugal or axial fan (See Figure 4-1). The pressure is assumed constant for flow rates that are above or below the flow range set by the `&CURV` lines. If the flow is less than the minimum flow value, then the pressure is assumed equal to the pressure specified for the minimum flow. If the flow exceeds the maximum flow, then the pressure is assumed equal to zero.

The pressure loss coefficient for the fan is applicable only when the fan is not operating but there is flow through the fan. In this case, the fan is treated as a fitting. The pressure loss

coefficient is a function of the particular fan type and the Reynolds number. The coefficient may be available directly from the manufacturer. Estimates could be made from information in some references [e.g., Crane Company, 1991; Idelchik, 1986; Brater, 1996].

An example of a &RFAN NAMELIST parameter group line and the associated &CURV and &CTRL lines is as follows:

```
&RFAN id=6, fileread=.FALSE., control=1, ftype=3, table='Supply Fan Data',  
reverse=.FALSE, kloss=1.13, fanname='Supply fan' /  
&CTRL id=1, ctype=1, inputvar=1, initial=.FALSE., setpoint=120.0 /  
&CURV id='Supply Fan Data', xy=0.0, 96.5 /  
&CURV id='Supply Fan Data', xy=1.41, 49.7 /  
&CURV id='Supply Fan Data', xy=1.99, 62.2 /  
&CURV id='Supply Fan Data', xy=2.3, 5.2 /
```

The above &RFAN line is read from the primary input file and is identified by the *ftype* parameter set to 3 and the text string 'Supply fan'. The fan is initially off but is turned on after 120 seconds from the start of the simulation. The pressure loss coefficient is 1.13 when the fan is not operating and the pressure versus flow data are provided on &CURV lines with the *id* parameter set to 'Supply Fan Data'.

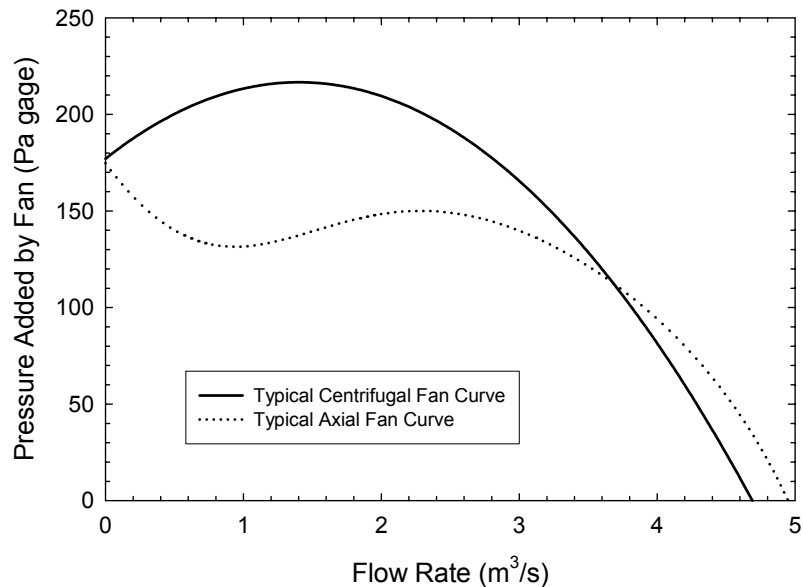


Figure 4-1. Typical Centrifugal and Axial Fan Curves [Hayashi *et al.*, 1985; ASHRAE, 1991]

4.5 Source Fires and Fuel Load

Compartment fires are either initially specified in terms of a heat release rate (power) versus time profile or arise in a given fuel load when specific conditions (temperature) are met. An initial fire is set using the `&FIRE` parameter group. The fuel load in a compartment fuel load is defined using the `&USES` NAMELIST group and the ignition criteria for this fuel load is set using the `&FUEL` parameter group.

A compartment does not require a fire or a fuel load for FSSIM to successfully run. There are some applications, such as verifying the initial ventilation conditions, where it is useful to run FSSIM without including a source fire.

4.5.1 Initial Source Fire (`&FIRE`)

The `&FIRE` NAMELIST group is used to specify the initial heat release rate (power) and species yield fractions of a compartment fire. A particular compartment should not have more than one initial fire specified. A simulation may be performed without setting a `&FIRE` parameter group. Such a model may be useful when verifying the initial or ambient ventilation conditions in compartments.

There is one parameter that must be set if the `&FIRE` line is used:

REQUIRED PARAMETER WHEN A `&FIRE` NAMELIST GROUP IS USED

<code>id</code>	(Long integer) Identification number for an individual <code>&FIRE</code> parameter group line. The <code>id</code> must be unique among other <code>&FIRE</code> parameter <code>id</code> values used in a particular FSSIM simulation and greater than zero. There is no default value.
-----------------	--

There is no requirement for the `&FIRE id` parameter to be sequential or in ascending or descending order. Any valid integer greater than zero and unique among other `&FIRE id` parameters is acceptable.

The remaining `&FIRE` parameters are optional:

OPTIONAL PARAMETERS

f _{type}	<p>(Short integer) The type of heat release rate versus time profile that defines the initial source fire:</p> <ul style="list-style-type: none"> 1 – The heat release rate is constant. 2 – The heat release rate is proportional to time squared (t^2) up to the time set by <code>t2_time</code> or the heat release rate set by <code>t2_hrr</code>, after which the fire is constant. 3 – The heat release rate versus time is provided in tabular format using <code>&CURV</code> lines with an <code>id</code> parameter set to the text string <code>table</code>. The maximum number of heat release rate versus time data points is 32,000. <p>The default value is 1.</p>
f _{subtype}	<p>(Short integer) Fire sub-type parameter. If <code>f_{type}</code> is set to 1, then <code>f_{subtype}</code> is the heat release rate selector:</p> <ul style="list-style-type: none"> 1 – 100 kW. 2 – 250 kW. 3 – 1,000 kW. 4 – constant heat release rate set by the <code>hrr</code> variable. <p>If <code>f_{type}</code> is set to 2, then <code>f_{subtype}</code> is the growth rate constant selector:</p> <ul style="list-style-type: none"> 1 – 0.02778 kW/s² ('slow' growth rate). 2 – 0.01111 kW/s² ('medium' growth rate). 3 – 0.0444 kW/s² ('fast' growth rate). 4 – 0.1778 kW/s² ('ultra fast' growth rate). 5 – growth rate set by the <code>growth</code> parameter. <p>The default value is 1 regardless of the value of <code>f_{type}</code>. This parameter is not used if <code>f_{type}</code> is not set to 1 or 2.</p>
location	<p>(Long integer) The compartment identification (<code>&COMP id</code> parameter) where the fire is located. The default value is 1.</p>
hrr	<p>(Real number) The heat release rate of the source fire (kW). This parameter is only used if <code>f_{type}</code> is set to 1 and <code>f_{subtype}</code> is set to 4. The default value is 100 kW.</p>
growth	<p>(Real number) The growth rate constant for a t^2 source fire (kW/s²). This parameter is only used if <code>f_{type}</code> is set to 2 and <code>f_{subtype}</code> is set to 5. There is no default value unless <code>f_{subtype}</code> is not properly set (<i>i.e.</i>, has a value greater than 5 or less than 1), in which case the default growth rate is 0.01173 kW/s².</p>
t2_time	<p>(Real number) The maximum growth period for a t^2 source fire (s). This parameter is only used if <code>f_{type}</code> is set to 2. The default value is 1×10^5 s.</p>

t2_hrr	(Real number) The maximum heat release rate for a t^2 source fire (kW). This parameter is only used if <code>f_{type}</code> is set to 2. The default value is 1×10^9 kW.
t_start	(Real number) The time from the simulation start that the initial source fire begins (s). The default value is 0.0 s.
t_end	(Real number) The time from the simulation start that the initial source fire ends (s). Note: the fire may end earlier due to suppression or oxygen consumption. The default value is 1×10^5 s.
table	(Text string) The name of the <code>&CURV</code> line(s) that contain the fire heat release rate (power) versus time (measured from <code>t_start</code>) tabular data. Time (s) is the first element of the <code>&CURV xy</code> array and heat release rate (kW) is the second element. There is no default and the maximum number of characters is 40. This parameter is not used if <code>f_{type}</code> is set to 1 or 2.
tpyro	(Real number) The pyrolysis temperature (K). The default value is 370.0 K.
deltah_f	(Real number) The heat of combustion of the fuel material (J/kg). The default value is 44.6×10^6 J/kg.
deltah_p	(Real number) The heat of pyrolysis of the fuel material (J/kg). The default value is 3.65×10^5 J/kg.
chi_r	(Real number) The fraction of source fire heat release rate energy that is released as thermal radiation. The default value is 0.35.

yco2	<p>(Real number) The carbon dioxide yield (kg carbon dioxide produced per kg fuel consumed). The default value is calculated using a chemical yield balance and depends on whether or not the oxygen yield is defined. If the oxygen yield is not defined, then the default carbon dioxide yield is calculated whether or not a value is provided. The value is determined as follows:</p> $y_{CO_2} = \frac{11}{3} - \frac{11}{7}y_{CO} - \frac{11}{3}y_{soot} - \frac{11}{27}y_{H_2O}$ <p>where y_{CO_2} is the yield fraction of carbon dioxide (y_{CO_2} [kg/kg]), y_{CO} is the yield fraction of carbon monoxide (y_{CO} [kg/kg]), y_{soot} is the yield fraction of soot (y_{soot} [kg/kg]), and y_{H_2O} is the yield fraction of water (y_{H_2O} [kg/kg]). If the oxygen yield is defined, then the default carbon dioxide yield is only calculated if a value is not provided using the following:</p> $y_{CO_2} = \frac{11}{8}y_{O_2} - \frac{11}{9}y_{H_2O} - \frac{11}{14}y_{CO}$ <p>where y_{O_2} is the yield fraction of oxygen (y_{O_2} [kg/kg]). To avoid potential rounding error in the model, it is recommended that this parameter be calculated by FSSIM.</p>
yh2o	<p>(Real number) The water yield (kg water produced per kg fuel consumed). The default value is 1.596 kg/kg.</p>
yco	<p>(Real number) The carbon monoxide yield (kg carbon monoxide produced per kg fuel consumed). The default value is 0.012 kg/kg.</p>
yo2	<p>(Real number) The oxygen yield (kg oxygen consumed per kg fuel burned). The default value (undefined condition) is calculated using a chemical yield balance assuming the fuel does not contain significant amounts of oxygen:</p> $y_{O_2} = \frac{8}{11}y_{CO_2} + \frac{8}{9}y_{H_2O} + \frac{4}{7}y_{CO}$ <p>where y_{O_2} is the yield fraction of oxygen (y_{O_2} [kg/kg]), y_{CO_2} is the yield fraction of carbon dioxide (y_{CO_2} [kg/kg]), y_{H_2O} is the yield fraction of water (y_{H_2O} [kg/kg]), and y_{CO} is the yield fraction of carbon monoxide (y_{CO} [kg/kg]).</p>
ysoot	<p>(Real number) The soot yield (kg soot produced per kg fuel consumed). The default value is 0.042 kg/kg.</p>

Nearly any type of source fire heat release rate profile may be specified in FSSIM using the available `ftype` and `fsubtype` options. The particular fire will depend on the fuel package burning properties, orientation, and surface area. References and approaches for estimating the

heat release rate are found in various sources including Lee [1985], Babrauskas [2002], Sardqvist [1993], Nowlen [1986], NFPA 72 [2002], and Madrzykowski [1996].

The heat release rate is assumed constant for times that are above or below the time range set by the `&CURV` lines. If the time is less than the minimum time value, then the heat release rate is assumed equal to the heat release rate specified for the minimum time. The reverse is true if the time is greater than the maximum time value.

The fuel characteristics (heat of combustion, pyrolysis temperature, the heat of pyrolysis, and the species yields) are constant over the course of the simulation and are material specific. Considerable care is necessary when entering the yield data to prevent violations of species conservation. Of particular note is the situation when FSSIM is used to calculate the oxygen yield and/or the carbon dioxide yield. In both cases, it is possible that unrealistic (negative) yields are calculated if the yield of water, carbon monoxide, and soot are over-specified. Sources for estimating the value of the yield parameters include Tewarson [2002], Babrauskas [2003], and Sardqvist [1993].

An example of a `&FIRE` NAMELIST parameter group line is as follows:

```
&FIRE id=12, location=2, ftype=2, fsubtype=3, t2_hrr=1500.0, t_start=0.0,
deltah_f=44100000.0, yh20=1.0, yco=0.03, ysoot=0.07 /
```

The above `&FIRE` line is identified with the `id` parameter set to 12, is located in the compartment with the `&COMP id` parameter of 2, and begins when the simulation starts. The fire has a growth rate that is proportional to the square of time and has a ‘fast’ heat release rate growth constant. The heat release rate becomes constant when it reaches 1,500 kW. The fuel has a carbon monoxide yield of 0.03 kg/kg, a soot yield of 0.07 kg/kg, and a water yield of 1.0 kg/kg. The carbon dioxide and oxygen yields are determined by FSSIM. Based on the equations cited above, the yields are about 2.96 kg CO₂/kg fuel and 3.08 kg O₂/kg fuel, respectively.

4.5.2 Fuel Load Parameters (`&USES`)

The `&USES` parameter group is used to set the fuel load and to associate a compartment or compartments with a `&FUEL` parameter group via the `&USES id` parameter. This parameter group is optional; however, if it is used there are two parameters that must be set:

REQUIRED PARAMETERS WHEN A `&USES` NAMELIST GROUP IS USED

<code>id</code>	(Long integer) Identification number for an individual <code>&USES</code> parameter group line. The <code>id</code> must be unique among other <code>&USES</code> parameter <code>id</code> values used in a particular FSSIM simulation and greater than zero. There is no default value.
<code>fuel</code>	(Long integer) <code>&FUEL id</code> parameter containing the fuel properties and ignition criteria. There is no default value.

There is no requirement for the `&USES id` parameter to be sequential or in ascending or descending order. Any valid integer greater than zero and unique among other `&USES id` parameters is acceptable. A `&FUEL` line may be used by multiple `&USES` lines (there is no requirement for the `fuel` parameter to be unique among `&USES` lines).

The remaining `&USES` parameters are optional. Many are the same type of parameter as were specified for the `&USES` parameter group. The following is a complete list of the optional parameters that may be set on the `&USES` line:

OPTIONAL PARAMETERS

username	(Text string) Name of the fuel load. The fuel load name is used primarily for labeling various output data. The maximum number of characters for the fuel load name is 40. The default is 'usetype id', where <code>id</code> is the fuel load identification number.
fileread	(Logical) <code>.TRUE.</code> indicates that the fuel load parameters should be read from the file set by the <code>&FNAM usefile</code> parameter and <code>.FALSE.</code> indicates that the fuel load data are contained within the <code>&USES id</code> line located in the main input file. The default state is <code>.TRUE.</code> If the <code>username</code> parameter is not set (<i>i.e.</i> , the default is used), then the uses data will <i>not</i> be read from the file set by the <code>&FNAM usefile</code> parameter.
loading	(Real number) The combustible fuel load (kg/m^2). The default value is 1.0 kg/m^2 .

The `&USES` variables may be read from a separate input file as specified by the `&FNAM usefile` parameter. The method is identical to that described for the `&CMPN NAMELIST` parameter group in Section 4.3.4 and is not repeated here.

An example of a `&USES` NAMELIST parameter group line is as follows:

```
&USES ID=2, fuel=6, fileread=.FALSE., loading=3.6, username='Class A Fuel
Load' /
```

The above `&USES` line is read from the primary input file and is identified with the `id` parameter set to 2 and the text string 'Class A Fuel Load'. The fuel combustion properties and ignition criteria are defined by the `&FUEL` line with the `id` parameter of 6. The fuel load is set to 3.6 kg/m^2 .

4.5.3 Fuel Load (`&FUEL`)

The `&FUEL` NAMELIST group is used to specify the composition and ignition criteria for a fuel load within a space. The information provided in this NAMELIST group is generic insofar as it is not placed in any specific location. The fuel is assigned to one or more compartments using the `&USES id` parameter.

This parameter group is optional; however, if it is used there is one parameter that must be set:

REQUIRED PARAMETER WHEN A &FUEL NAMELIST GROUP IS USED

id	(Long integer) Identification number for an individual &FUEL parameter group line. The id must be unique among other &FUEL parameter id values used in a particular FSSIM simulation and greater than zero. There is no default value.
----	--

There is no requirement for the &FUEL id parameter to be sequential or in ascending or descending order. Any valid integer greater than zero and unique among other &FUEL id parameters is acceptable.

The remaining &FUEL parameters are optional. Many are the same type of parameter as were specified for the &FIRE parameter group. The following is a complete list of the optional parameters that may be set on the &FUEL line:

OPTIONAL PARAMETERS

fuelname	(Text string) Name of fuel. The fuel name is used primarily for labeling various output data. The maximum number of characters for the fuel name is 40. The default is 'fuel id', where id is the fan identification number.
fileread	(Logical) .TRUE. indicates that the fuel data should be read from the file set by the &FNAM fuelfile parameter and .FALSE. indicates that the fuel data are contained within the &FUEL id line located in the main input file. The default state is .TRUE. If the fuelname parameter is not set (<i>i.e.</i> , the default is used), then the fuel data will <i>not</i> be read from the file set by the &FNAM fuelfile parameter.
growth	(Real number) The heat release rate growth rate (kW/s ²). The default value is 0.01111 kW/s ² ('medium' t ² growth rate).
pyro_max	(Real number) Maximum fuel pyrolysis rate (kg/m ²). The default value is 0.01 kg/m ² .
tpyro	(Real number) The pyrolysis temperature (K). The default value is 370.0 K.
deltah_f	(Real number) The heat of combustion of the fuel material (J/kg). The default value is 44.6 x 10 ⁶ J/kg.
deltah_p	(Real number) The heat of pyrolyzation of the fuel material (J/kg). The default value is 3.65 x 10 ⁵ J/kg.

chi_r	(Real number) The fraction of source fire heat release rate energy that is released as thermal radiation. The default value is 0.35.
yco2	<p>(Real number) The carbon dioxide yield (kg carbon dioxide produced per kg fuel consumed). The default value is calculated using a chemical mass balance and depends on whether or not the oxygen yield is defined. If the oxygen yield is not defined, then the default carbon dioxide yield is calculated whether or not a value is provided. The value is determined as follows:</p> $y_{CO_2} = \frac{11}{3} - \frac{11}{7} y_{CO} - \frac{11}{3} y_{soot} - \frac{11}{27} y_{H_2O}$ <p>where y_{CO_2} is the yield fraction of carbon dioxide (y_{CO_2} [kg/kg]), y_{CO} is the yield fraction of carbon monoxide (y_{CO} [kg/kg]), y_{soot} is the yield fraction of soot (y_{soot} [kg/kg]), and y_{H_2O} is the yield fraction of water (y_{H_2O} [kg/kg]). If the oxygen yield is defined, then the default carbon dioxide yield is only calculated if a value is not provided using the following:</p> $y_{CO_2} = \frac{11}{8} y_{O_2} - \frac{11}{9} y_{H_2O} - \frac{11}{14} y_{CO}$ <p>where y_{O_2} is the yield fraction of oxygen (y_{O_2} [kg/kg]). To avoid potential rounding error in the model, it is recommended that this parameter be calculated by FSSIM.</p>
yh2o	(Real number) The water yield (kg water produced per kg fuel consumed). The default value is 1.596 kg/kg.
yco	(Real number) The carbon monoxide yield (kg carbon monoxide produced per kg fuel consumed). The default value is 0.012 kg/kg.
yo2	<p>(Real number) The oxygen yield (kg oxygen consumed per kg fuel burned). The default value (undefined condition) is calculated using a chemical yield balance assuming the fuel does not contain significant amounts of oxygen:</p> $y_{O_2} = \frac{8}{11} y_{CO_2} + \frac{8}{9} y_{H_2O} + \frac{4}{7} y_{CO}$ <p>where y_{O_2} is the yield fraction of oxygen (y_{O_2} [kg/kg]), y_{CO_2} is the yield fraction of carbon dioxide (y_{CO_2} [kg/kg]), y_{H_2O} is the yield fraction of water (y_{H_2O} [kg/kg]), and y_{CO} is the yield fraction of carbon monoxide (y_{CO} [kg/kg]).</p>
ysoot	(Real number) The soot yield (kg soot produced per kg fuel consumed). The default value is 0.042 kg/kg.
ignite_ST	(Real number) Non-overhead surface temperature ignition criteria for fuel (K). The default value is 473.0 K.

ignite_OV	(Real number) Overhead surface temperature ignition criteria for fuel (K). The default value is 650.0 K.
ignite_GT	(Real number) Compartment gas temperature ignition criteria for fuel (K). The default value is 473.0 K.
ignite_VT	(Real number) Inflowing gas temperature ignition criteria for fuel (K). The default value is 600.0 K.

The &FUEL variables may be read from a separate input file as specified by the &FNAM fuelfile parameter. The method is identical to that described for the &CMPN NAMELIST parameter group in Section 4.3.4 and is not repeated here.

As with the &FIRE parameter group, the fuel characteristics (heat of combustion, pyrolysis temperature, the heat of pyrolyzation, and the species yields) are constant over the course of the simulation and are material specific. Considerable care is necessary when entering the yield data to prevent violations of species conservation. Of particular note is the situation when FSSIM is used to calculate the oxygen yield and/or the carbon dioxide yield. In both cases, it is possible for that unrealistic (negative) yields are calculated if the yield of water, carbon monoxide, and soot are over-specified. Sources for estimating the value of the yield parameters include Tewarson [2002], Babrauskas [2003], and Sardqvist [1993].

The fuel may ignite via any of four criteria: a non-overhead surface temperature, an overhead surface temperature, the compartment gas temperature, or the temperature of combustion/gas products entering a compartment. If one or more of these criteria are not implemented, then the default values should be adjusted accordingly. Note that the heat release rate of the fuel is proportional to t^2 . The maximum heat release rate is a function of the available oxygen in the space because there is no specific fuel package burning (*i.e.*, a flashover-like condition is assumed).

An example of a &FUEL NAMELIST parameter group line is as follows:

```
&FUEL ID=11, growth=0.00293, deltax_f=14100000.0, yh20=1.0, yco=0.03,
ysoot=0.07, fileread=.FALSE., ignition_ST=436.8, ignition_OV=5000.0,
ignition_GT=773.0, ignition_VT=1273.0, fuelname='Class A Combustibles' /
```

The above &FUEL line is read from the primary input file and is identified with the id parameter set to 11 and the text string 'Class A Combustibles'. The fire has a growth rate constant equal to 0.00293 kW/s², which is nearly equal to a 'slow' t² fire. The heat of combustion for the materials is 14 x 10⁶ J/kg, typical of wood-based products. The fuel has a carbon monoxide yield of 0.03 kg/kg, a soot yield of 0.07 kg/kg, and a water yield of 1.0 kg/kg. The carbon dioxide and oxygen yields are determined by FSSIM. Based on the equations cited above, the yields are about 2.96 kg CO₂/kg fuel and 3.08 kg O₂/kg fuel, respectively. The ignition criteria is either a boundary surface temperature of 436.8 K (the ASTM E119-98 temperature criteria for a fire barrier wall [ASTM E119-98, 1999]), a compartment gas temperature of 773.0 K (flashover), or an inflowing gas temperature of 1273.0 K (flames). The ignition criteria for an overhead surface is set to 5000.0 K to disable the feature.

4.6 Detection and Fire Suppression

Detectors and fire suppression actions may be included in a FSSIM simulation. Detection may be used as a means of controlling fans, dampers, and fire doors. The detection time for a particular type of scenario may also be of interest for reasons beyond the scope of this model, such as evacuation and response time estimates. Suppression is a means of directly interfering with the combustion process within a space. Several types of extinguishing systems are available in FSSIM, including water-based systems or systems using gaseous agents.

4.6.1 Detection Systems (&DTCT)

The &DTCT parameter group is used to specify a detector (location and type). This parameter group is optional; however, if it is used there are two (out of three) parameters that must be set:

REQUIRED PARAMETERS WHEN A &DTCT NAMELIST GROUP IS USED

id	(Long integer) Identification number for an individual &DTCT parameter group line. The id must be unique among other &DTCT parameter id values used in a particular FSSIM simulation and greater than zero. There is no default value.
comp	(Long integer) A compartment identification number (&COMP id parameter value) where the detector is located. -1 indicates that the detector is not in a compartment. This parameter must be -1 if the hvac parameter is greater than 0. The default value is -1.
hvac	(Long integer) A duct identification number (&RDCT id parameter value) where the detector is located. -1 indicates that the detector is not in a duct. This parameter must be -1 if the comp parameter is greater than 0. The default value is -1.

There is no requirement for the &DTCT id parameter to be sequential or in ascending or descending order. Any valid integer greater than zero and unique among other &DTCT id parameters is acceptable. The comp and hvac parameters are mutually exclusive and a fatal error will occur if both are specified. The detector may be located in either a compartment or a duct, but not both. Note that there is no restriction on the number of detectors within a compartment or a duct. Multiple &DTCT parameter groups may reference the same compartment or duct; however, if the detectors are identical they will actuate (or fail to actuate) at the same time.

The remaining &DTCT parameters are optional:

OPTIONAL PARAMETERS

detectorname	(Text string) Name of the detector. The detector name is used primarily for labeling various output data. The maximum number of characters for the detector name is 40. The default is 'detector id', where id is the detector identification number.
fileread	(Logical) <code>.TRUE.</code> indicates that the detector parameters should be read from the file set by the <code>&FNAM detectorfile</code> parameter and <code>.FALSE.</code> indicates that the detector parameters are contained within the <code>&DTCT id</code> line located in the main input file. The default state is <code>.TRUE.</code> If the <code>detectorname</code> parameter is not set (<i>i.e.</i> , the default is used), then the detector data will <i>not</i> be read from the file set by the <code>&FNAM detector</code> parameter.
dtype	(Short integer) The type of detection system: 1 – ionization smoke detector 2 – photoelectric smoke detector 4 – infrared heat detector 5 – ultraviolet electromagnetic radiation detector 6 – rate of rise/thermal element heat detector 8 – fixed temperature detector 9 – carbon monoxide detector 10 – carbon dioxide detector 11 – oxygen sensor The default value is 1.
operational	(Logical) <code>.TRUE.</code> indicates that the detector is functioning normally and <code>.FALSE.</code> indicates that the detector is not functioning. The default state is <code>.TRUE.</code>
rti	(Real number) The Response Time Index (RTI) for the detector device ($m^{0.5}\cdot s^{0.5}$). This parameter is only used by detectors that are thermally actuated (<code>dtype</code> set to 6). The default value is $10 m^{0.5}\cdot s^{0.5}$.
temperature	(Real number) Actuation temperature (K) for detectors that are thermally actuated (<code>dtype</code> set to 6 or 8). The default value is 0.0 K.
det_spacing	(Real number) Average spacing between detectors (m). If this parameter is not set or a negative number is provided, then the detector spacing will be equal to: $L = (x \cdot y)^{1/2}$ where L is the detector spacing (m), x is the dimension of the compartment containing the detector specified by the <code>&COMP dx</code> parameter, and y is the compartment dimension set by the <code>&COMP dy</code> parameter. This parameter is not used if the detector is located in a duct. The default value is $(\&COMP dx \cdot \&COMP dy)^{1/2}$.

min_fire	(Real number) The minimum detectable fire size (kW) for infrared or ultraviolet detectors (dtype set to 4 or 5). The default value is 1.0 kW.
visibitily	(Real number) The obscuration density (m^{-1}) at which actuation occurs for ionization or photoelectric detectors (dtype set to 1 or 2). The default value is 0.0581 m^{-1} .
massfrac	(Real number) The mass fraction of carbon monoxide, carbon dioxide, or oxygen at which detection occurs for a carbon monoxide, carbon dioxide, or oxygen detector (dtype set to 9, 10, or 11, respectively). The default value is 0.01 kg CO, CO ₂ , or O ₂ per kg air.

The &DTCT variables may be read from a separate input file as specified by the &FNAM detectorfile parameter. The method is identical to that described for the &CMPN NAMELIST parameter group in Section 4.3.4 and is not repeated here.

Detector actuation information is specific to the type of detector installed. Information regarding detector set points may be available from the manufacturer of the particular device. Other source of information include NFPA 72 [2002], Geiman and Gottuk [2001], Schifiliti, Meachum, and Custer [2002] and Beyler and DiNenno [1991].

An example application for the &DTCT NAMELIST parameter group is as follows:

```
&DTCT ID=4, comp=6, fileread=.FALSE., dtype=2, visibility=0.18,
detectorname='Photoelectric smoke detector' /
&DTCT ID=5, comp=6, fileread=.FALSE., dtype=8, temperature=343.15,
detectorname='Fixed temperature detector - 70C' /
```

The above &DTCT lines are read from the primary input file. One is identified with the id parameter set to 4 and the text string 'Photoelectric smoke detector' and the other is identified with the id parameter set to 5 and the text string 'Fixed temperature detector - 70C'. Both detectors are located in the compartment with the &COMP id parameter set to 6 and are operational. The first detector is a photoelectric smoke detector that will actuate when the smoke obscuration density is greater than or equal to 0.18 m^{-1} . The second detector is a fixed temperature heat detector that will actuate when the ambient room temperature is 70°C (343.15 K) or greater.

4.6.2 Suppression Systems (&SUPR)

The &SUPR parameter group is used to specify a suppression system (location and type). This parameter group is optional; however, if it is used there are four (out of six) parameters that must be set:

REQUIRED PARAMETERS WHEN A &SUPR NAMELIST GROUP IS USED

id	(Long integer) Identification number for an individual &SUPR parameter group line. The id must be unique among other &SUPR parameter id values used in a particular FSSIM simulation and greater than zero. There is no default value.
comp	(Long integer) A compartment identification number (&COMP id parameter value) where the detector is located. -1 indicates that the suppression is not in a compartment. This parameter must be -1 if the hvac parameter is greater than 0. Because HVAC suppression is not currently implemented, the comp parameter must be positive. The default value is -1.
hvac	(Long integer) A duct identification number (&RDCT id parameter value) where the suppression is located. -1 indicates that the suppression is not in a duct. This parameter must be -1 if the comp parameter is greater than 0. Because HVAC suppression is not currently implemented, this parameter must be set to -1. The default value is -1.
diam	(Real number) The average drop diameter (m) for a mist or sprinkler suppression system (stype set to 1 or 2). This is a required parameter because a mist suppression system is the default and there is no default value for the droplet diameter. If this parameter is not set, an error would thus be generated.
flowrate	(Real number array) An array of real numbers, separated by a comma, containing the mass flow rate (kg/s) of the suppressing agents. The mass flow rate is total agent flow rate into a particular control volume (compartment or duct) from all discharge devices associated with the particular suppression system. The number of array elements is 1 if stype is set to 1, 2, 4, 5, 6, or 7. If stype is set to 3, the number of elements is equal to the ngasses parameter and the elements correspond to the individual gas components as specified in the gas parameter array. The order of the array elements is thus significant. There is/are no default value(s).
control	(Long integer) Control function &CTRL id parameter value used to establish actuation criteria for suppression system. There is no default value.

There is no requirement for the &SUPR id parameter to be sequential or in ascending or descending order. Any valid integer greater than zero and unique among other &SUPR id parameters is acceptable. The comp and hvac parameters are mutually exclusive. The suppression system may be located in either a compartment or a duct, but not both. However, because the HVAC suppression is not currently implemented, the suppression system must be located in a compartment. Note that there is no restriction on the number of suppression systems within a compartment (or a duct when implemented). Multiple &SUPR parameter groups may reference the same compartment (or duct).

The remaining &SUPR parameters are optional:

OPTIONAL PARAMETERS

suppressorname	(Text string) Name of the suppression system. The suppression system name is used primarily for labeling various output data. The maximum number of characters for the suppression system name is 40. The default is 'suppressor id', where id is the suppression system identification number.
fileread	(Logical) .TRUE. indicates that the suppression system parameters should be read from the file set by the &FNAM suppressorfile parameter and .FALSE. indicates that the suppression system parameters are contained within the &SUPR id line located in the main input file. The default state is .TRUE. If the suppressorname parameter is not set (<i>i.e.</i> , the default is used), then the suppressor data will <i>not</i> be read from the file set by the &FNAM suppressorfile parameter.
stype	(Short integer) The type of suppression system: 1 – mist suppression system 2 – sprinkler system 3 – gaseous agent suppression system 4 – hydrofluorocarbon (Halon) suppression system 5 – handline (not implemented) 6 – foam 7 – dry chemical 8 – boundary cooling for a single surface 9 – boundary cooling for a compartment (excludes the overhead) The default value is 1.
efficiency	(Real number from 0 to 1) Denotes the evaporation efficiency for a water mist system. This parameter will reduce the mass evaporation by multiplying the calculated evaporation by the efficiency. The default value is 1.
operational	(Logical) .TRUE. indicates that the suppression system is functioning normally and .FALSE. indicates that the suppression system is not functioning. The default state is .TRUE.
tflow	(Real number) The discharge temperature of the suppression system agent (K). Parameter is used only if stype is set to 1, 2, 3, 4, 5, 6, 8, or 9. The default value is the &EXEC tamb parameter value.
massfrac	(Real number) The required mass fraction of suppression agent for extinguishment (kg agent/kg air). This parameter is used only if stype is set to 4 or 7. The default value is 1×10^{-6} kg agent/kg air.

agent_cp	(Real number) The specific heat of the suppression system agent (J/kg-K). This parameter is required if <code>stype</code> is set to 4; otherwise, it is not used. There is no default value.
discharge_t	(Real number) The discharge time for the suppression system (s). The default value is 1×10^9 s.
ngasses	(Short integer) The number of gasses composing the gaseous suppression system agent. This parameter is required if <code>stype</code> is set to 3; otherwise, it is not used. The maximum number of gasses is 6. There is no default value.
gas	(Short integer array) An array of short integers, from the following list, separated by commas, specifying the gas components of the suppression system. The number of elements is equal to the <code>ngasses</code> parameter and thus may not exceed 6. This parameter is required if <code>stype</code> is set to 3; otherwise, it is not used. The gas components are identified as follows: 1 – nitrogen (N ₂) 2 – oxygen (O ₂) 3 – fuel 4 – carbon dioxide (CO ₂) 5 – water (H ₂ O) 6 – carbon monoxide (CO) The order of the array elements is not significant relative to which gases are specified; however, the gas flow rates specified using the <code>flowrate</code> parameter must correspond to the gases. There are no default values for any of the array elements.
surf	(Long integer) A surface identification number (<code>&SURF id</code> parameter value) where the suppression is targeted. -1 indicates that the suppression system is not targeted at a surface and is thus not a boundary cooling device. This parameter is required if <code>stype</code> is set to 8; otherwise, it is not used. The default value is -1.
expansion	(Long integer) The expansion ratio of the foam used to suppress a fire. This parameter is required if <code>stype</code> is set to 6; otherwise, it is not used. The default value is 10.0.
thickness	(Long integer) The thickness of the foam layer used to suppress a fire (m). This parameter is required if <code>stype</code> is set to 6; otherwise, it is not used. The default value is 0.1 m.

The `&SUPR` variables may be read from a separate input file as specified by the `&FNAM suppressorfile` parameter. The method is identical to that described for the `&CMPN NAMELIST` parameter group in Section 4.3.4 and is not repeated here.

Suppression system parameters are specific to the devices used. Information regarding suppression systems may be obtained from the manufacturers. Information is also available in

various reference material including Flemming [2002], Scheffey and Hickey [2002], Grant [2002], DiNenno [2002], Mawhinney and Back [2002]. NFPA 11 [2002], NFPA 12 [2000], NFPA 13 [2002], NFPA 15 [2001], NFPA 16 [2003], and NFPA 17 [2002].

An example application for the `&SUPR` NAMELIST parameter group and the associated `&CTRL` lines is as follows:

```
&SUPR ID=4, comp=6, fileread=.FALSE., stype=2, flowrate=8.8E-4, diam=0.0007,
control=3, suppressorname='Ordinary sprinkler system' /
&CTRL id=3, ctype=1, trip=.TRUE., initial=.FALSE., inputvar=2,
initial=.FALSE., setpoint=338.15 /
&SUPR ID=5, comp=6, fileread=.FALSE., stype=3, flowrate=1.6E-7, 4.3E-6,
3.9E-6, control=4, ngasses=3, gas=2, 1, 4, discharge_t=35.0,
suppressorname='Gaseous agent' /
&CTRL id=4, ctype=1, trip=.TRUE., initial=.FALSE., inputvar=10,
initial=.FALSE., location=2 /
```

The above `&SUPR` lines are read from the primary input file. One is identified with the `id` parameter set to 4 and the text string 'Ordinary Sprinkler System' and the other is identified with the `id` parameter set to 5 and the text string 'Gaseous agent'. Both suppression systems are located in the compartment with the `&COMP id` parameter set to 6 and are operational but initially not actuated. The first suppression system is an ordinary sprinkler system that actuates when the compartment temperature reaches 65°C (338.15 K), has an average drop diameter of 0.7 mm, and a flow rate of 8.8×10^{-4} kg/s (about 15 gpm). The second suppression system is a gaseous agent consisting of nitrogen, oxygen, and carbon dioxide. The mass flow rates for each constituent are 1.6×10^{-7} kg O₂/s, 4.6×10^{-6} kg N₂/s, and 3.9×10^{-6} kg CO₂/s. The gaseous suppression system actuates by detection and discharges in 35 seconds. The detector characteristics are specified by a `&DTCT` parameter group with an `id` set to 2.

5.0 FSSIM OUTPUT

The results of a FSSIM execution are interpreted using the output data. Currently, there are two output methods available: one that generates several Comma Separated Value (CSV) ASCII files that may be used by a graphing or spreadsheet program of the users choice (general output format) and one that generates a single Comma Separated Value (CSV) ASCII file that is used by the MSU viewer (MSU viewer output format) [Haupt *et al.*, 2004]. As noted in Section 4.2.1, the particular output format is set using the `&EXEC outtype` parameter.

5.1 General Output Format

When the `&EXEC outtype` parameter is set to `.FALSE.`, FSSIM will generate multiple CSV output files, each containing data for related variables. The number of files that are generated will depend on the configuration that is modeled. The output files are generated by appending the output file extension to the `fileid` parameter set by the `&EXEC` group (Section 4.2.1) along with an integer indicating the sequence of the file if more than one is required for the output variable. For example if there were 256 compartments, two output files would be required and they would be named `[fileid]_ctemps1.csv` and `[fileid]_ctemps2.csv`. The specific output files are summarized in Table 5-1 and are described below.

Table 5-1. Summary of FSSIM Output Files – General Output Format

Output File	Output Data Variables
[fileid]_ctemps#.csv	Compartment temperatures
[fileid]_ntemps#.csv	Duct node temperatures
[fileid]_cpres#.csv	Compartment pressures
[fileid]_npres#.csv	Duct node pressures
[fileid]_chrr#.csv	Compartment heat release rate
[fileid]_compq#.csv	Compartment energy balance variables
[fileid]_jvels#.csv	Opening flow velocities
[fileid]_dvels#.csv	Duct flow velocities
[fileid]_jflows#.csv	Opening mass flow rates
[fileid]_dflows#.csv	Duct mass flow rates
[fileid]_jarea#.csv	Opening flow areas
[fileid]_darea#.csv	Duct flow areas
[fileid]_cgas#.csv	Mass fraction of gas components in compartments
[fileid]_ngas#.csv	Mass fraction of gas components in the ductwork
[fileid]_fire#.csv	Fire theoretical maximum heat release rate (i.e. non-O ₂ limited)
[fileid]_stemp#.csv	Surface temperatures
[fileid]_sflux#.csv	Surface heat fluxes
[fileid]_detect#.csv	Actuation status of detectors
[fileid]_suppress#.csv	Actuation status of suppression system

Each output file will have a header consisting a comma separated line of character strings. The format for all headers is:

Time (s), item name(1), item name(2)..., item name (nitems)

where item name is the character string associated with the output quantity (e.g. for a compartment quantity it will be the compartment name, for a surface quantity the surface name)

1. Compartment temperature output data – [fileid]_ctemps#.csv

This file contains the compartment temperature data. There is a description header followed by rows of output data in the following format:

time, temp(c1), temp(c2), ..., temp(ncomp)
--

- time The time (s) that the temperature data correspond to.
- temp(cn) The gas temperature (K) in the compartment set by the cnth &COMP line.
- ncomp The number of &COMP lines as specified on the &CNTR line.

If there are more than 254 compartments (254 temperatures plus the time gives 255 columns), then additional files will be required for each additional block of 254 compartments.

2. Duct node temperature output data – [fileid]_ntemps#.csv

This file contains the compartment temperature data. There is a description header followed by rows of output data in the following format:

time, temp(n1), temp(n2), ..., temp(nnode)
--

- time The time (s) that the temperature data correspond to.
- temp(nj) The gas temperature (K) at the duct node set by the nnth &RNOD line.
- nnode The number of &RNOD lines as specified on the &CNTR line.

If there are no duct nodes, then this file will not be created.

3. Compartment pressure output data – [fileid]_cpres#.csv

This file contains the compartment pressure data. There is a description header followed by rows of output data in the following format:

time, pres(c1), pres(c2), ..., pres(ncomp)

time The time (s) that the pressure data correspond to.
pres(cn) The pressure (Pa) in the compartment set by the cnth &COMP line.
ncomp The number of &COMP lines as specified on the &CNTR line.

4. Duct node pressure output data – [fileid]_npres#.csv

This file contains the duct node pressure data. There is a description header followed by rows of output data in the following format:

time, pres(n1), pres(n2), ..., pres(nnode)

time The time (s) that the pressure data correspond to.
pres(nn) The pressure (Pa) at the duct node set by the nnth &RNOD line.
nnode The number of &RNOD lines as specified on the &CNTR line.

If there are no nodes, then this file will be created.

5. Compartment heat release data – [fileid]_chrr#.csv

This file contains the compartment heat release rate data. There is a description header followed by rows of output data in the following format:

time, hrr(c1), hrr(c2), ..., hrr(ncomps)

time The time (s) that the heat release data correspond to.
hrr(cn) The heat release rate (kW) at the compartment set by the cnth &COMP line.

ncomps The number of &COMP lines as specified on the &CNTR line.

If there are no nodes, then this file will be created.

6. Compartment energy balance output data – [fileid]_compq#.csv

This file contains data relating to the energy balance in a compartment. A positive value indicates that there is a net increase in the energy content of the component. There is a description header followed by rows of output data in the following format:

time, qg(c1), qw(c1), qg(c2), qw(c2), ..., qg(ncomp), qw(ncomp)

time	The time (s) that the energy balance data corresponds to.
qg(cn)	The net radiation absorbed by the gas (W) in the compartment set by the cn th &COMP line.
qw(cn)	The net convection heat flux (W) to the boundaries of the compartment set by the cn th &COMP line.
ncomp	The number of &COMP lines as specified on the &CNTR line.

Since two variables are being written per compartment, additional files are required if there are more than 127 compartments.

7. Opening flow velocity output data – [fileid]_jvels#.csv

This file contains flow velocity data for openings specified using &JUNC NAMELIST group. Positive flow for &JUNC openings is defined as flow from the first compartment specified by the first element of &JUNC location array to the second compartment (or ambient) specified by the second element.

Bidirectional &JUNC type openings are split into two openings in FSSIM and the output data are reported accordingly. The lower portion of a vertical opening with horizontal flow retains the input sequence number. The upper portion of a vertical opening is given the next available sequence number that is greater than the total number of openings as set by the &EXEC njunc parameters. Thus, if three vertical openings were defined and ordered as follows:

```
&junc id=6, bidirectional=.TRUE., juncname='door A' ... /  
&junc id=3, bidirectional=.FALSE., ... /  
&junc id=12, bidirectional=.TRUE., ... /
```

and the `&EXEC njunc` parameter were set to 3 and the `&EXEC nnotbidir` were set to 1, then FSSIM would create five openings with sequential identification numbers as follows:

- Opening 1 – lower portion of opening with `&JUNC id` set to 6
- Opening 2 – lower portion of opening with `&JUNC id` set to 3
- Opening 3 – lower portion of opening with `&JUNC id` set to 12
- Opening 4 – upper portion of opening with `&JUNC id` set to 6
- Opening 5 – upper portion of opening with `&JUNC id` set to 12

The bidirectional `juncname` is obtained by appending '-2' to the name. Thus the `juncname` for opening 4 would be 'door A-2'.

The flow velocity output is reported sequentially in exactly this manner. A similar arrangement applies to horizontal openings, except the notions of lower and upper are not applicable.

The output file contains a description header followed by rows of output data in the following format:

`time, vflow(j1), vflow(j2), ..., vflow(2*njunc-nnotbidir)`

<code>time</code>	The time (s) that the opening velocity flow data corresponds to.
<code>vflow(jn)</code>	The flow velocity (m/s) through the jn^{th} sequential opening.
<code>njunc</code>	The number of <code>&JUNC</code> lines as specified on the <code>&CNTR</code> line.
<code>nnotbidir</code>	The number <code>&JUNC</code> openings that are not bidirectional as specified on the <code>&CNTR</code> line.

If there are no openings, then this file will not be created.

8. Duct flow velocity output data – [fileid]_dvels#.csv

This file contains flow velocity in ducts specified using `&RDCT NAMELIST` group. Positive flow in ducts is defined as flow from the duct node specified by the first element in the `&RDCT node` array to the second duct node specified by the second element of the array.

The output file contains a description header followed by rows of output data in the following format:

`time, vflow(d1), vflow(d2), ..., vflow(nduct)`

time	The time (s) that the opening velocity flow data correspond to.
vflow(dn)	The flow velocity (m/s) through the duct set by the dn th &RDCT line.
nduct	The number of &RDCT lines as specified on the &CNTR line.

If there are no ducts, then this file will not be created.

9. Opening mass flow output data – [fileid]_jflows#.csv

This file contains mass flow data for openings specified using &JUNC NAMELIST group. Positive opening mass flows are defined in the same manner as positive opening flow velocities. The mass flow through openings is reported in the same sequential manner as opening flow velocities.

The output file contains a description header (that indicates when a bidirectional opening is split) followed by rows of output data in the following format:

time, mflow(j1), mflow(j2), ..., mflow(2*njunc-nnotbidir)

time	The time (s) that the opening mass flow data correspond to.
mflow(jn)	The mass flow (kg/s) through the jn th sequential opening.
njunc	The number of &JUNC lines as specified on the &CNTR line.
nnotbidir	The number &JUNC openings that are not bidirectional as specified on the &CNTR line.

If there are no openings, then this file will not be created.

10. Duct mass flow output data – [fileid]_dflows#.csv

This file contains mass flow data for ducts &RDCT NAMELIST group and for ducts. Positive duct mass flows are defined in the same manner as positive duct flow velocities.

The output file contains a description header followed by rows of output data in the following format:

time, mflow(d1), mflow(d2), ..., mflow(nduct)

time The time (s) that the duct mass flow data correspond to.
mflow(dn) The mass flow (kg/s) through the duct set by the dnth &RDCT line.
nduct The number of &RDCT lines as specified on the &CNTR line.

If there are no ducts, then this file will not be created.

11. Opening flow area output data – [fileid]_jarea#.csv

This file contains opening flow area data. The flow area of an opening may vary with time through the use of control functions set using a &CTRL line or for bidirectional flows where the area of each sub-opening is determined as part of the mass balance across the opening. The area for openings set using the &JUNC group is reported in the same manner as the flow velocities.

The output file contains a description header (that indicates when a bidirectional opening is split) followed by rows of output data in the following format:

time, Area(j1), Area(j2), ..., Area(2*njunc-nnotbidir)

time The time (s) that the opening area data correspond to.
Area(jn) The flow area (m²) of the jnth sequential opening.
njunc The number of &JUNC lines as specified on the &CNTR line.
nnotbidir The number &JUNC openings that are not bidirectional as specified on the &CNTR line.

If there are no openings, then this file will not be created.

12. Duct flow area output data – [fileid]_darea#.csv

This file contains duct flow area data. The duct flow area may vary with time through the use of control functions set using a &CTRL line.

The output file contains a description header followed by rows of output data in the following format:

time, Area(d1), Area(d2), ..., Area(nduct)

time The time (s) that the opening area data correspond to.
 Area(d1) The flow area (m²) of the duct set by the first &RDCT line.
 nduct The number of &RDCT lines as specified on the &CNTR line.

If there are no ducts, then this file will not be created.

13. Gas mass fraction output data (compartments) – [fileid]_cgas#.csv

This file contains the mass fraction of various component gases in each compartment. There is a description header followed by rows of output data in the following format:

time, O2(c1), O2(c2), ..., O2(ncomp), CO(c1), CO(c2), ..., CO(ncomp), CO2(c1), CO2(c2), ..., CO2(ncomp), H2O(c1), H2O(c2), ..., H2O(ncomp), C(c1), C(c2), ..., C(ncomp), UHC(c1), UHC(c2), ..., UHC(ncomp)

time The time (s) that the gas concentration data correspond to.
 O2(cn) The oxygen mass fraction (kg O₂/kg gas) in the compartment set by the cnth &COMP line. Only written if gaso2=.TRUE. on the &OUTP line.
 CO(cn) The carbon monoxide mass fraction (kg CO/kg gas) in the compartment set by the cnth &COMP line. Only written if gasco=.TRUE. on the &OUTP line.
 CO2(cn) The carbon dioxide mass fraction (kg CO₂/kg gas) in the compartment set by the cnth &COMP line. Only written if gasco=.TRUE. on the &OUTP line.
 H2O(cn) The water vapor mass fraction (kg H₂O/kg gas) in the compartment set by the cnth &COMP line. Only written if gash2o=.TRUE. on the &OUTP line.
 C(cn) The soot mass fraction (kg soot/kg gas) in the compartment set by the cnth &COMP line. Only written if gasc=.TRUE. on the &OUTP line.

UHC(cn)	The unburned hydrocarbon mass fraction (kg UHC/kg gas) in the compartment set by the cn^{th} &COMP line. Only written if <code>gasfuel=.TRUE.</code> on the &OUTP line.
ncomp	The number of &COMP lines as specified on the &CNTR line.

This file will not be created if all `gas???` variables are set to `.FALSE.` on the &OUTP line. Additional files will be required if there are more than 254 / (number of output gases) compartments.

14. Gas mass fraction output data (ducts) – [fileid]_ngas.csv

This file contains the mass fraction of various component gases at each duct node. There is a description header followed by rows of output data in the following format:

time, O2(n1), O2(n2), ..., O2(nnode), CO(n1), CO(n2), ..., CO(nnode), CO2(n1), CO2(n2), ..., CO2(nnode), H2O(n1), H2O(n2), ..., H2O(nnode), C(n1), C(n2), ..., C(nnode), UHC(n1), UHC(n2), ..., UHC(nnode)

time	The time (s) that the gas concentration data correspond to.
O2(nn)	The oxygen mass fraction (kg O ₂ /kg gas) in the duct node set by the nn^{th} &RNOD line. Only written if <code>gaso2=.TRUE.</code> on the &OUTP line.
CO(nn)	The carbon monoxide mass fraction (kg CO/kg gas) in the duct node set by the nn^{th} &RNOD line. Only written if <code>gasco=.TRUE.</code> on the &OUTP line.
CO2(nn)	The carbon dioxide mass fraction (kg CO ₂ /kg gas) in the duct node set by the nn^{th} &RNOD line. Only written if <code>gasco=.TRUE.</code> on the &OUTP line.
H2O(nn)	The water vapor mass fraction (kg H ₂ O/kg gas) in the duct node set by the nn^{th} &RNOD line. Only written if <code>gash2o=.TRUE.</code> on the &OUTP line.
C(nn)	The soot mass fraction (kg soot/kg gas) in the duct node set by the nn^{th} &RNOD line. Only written if <code>gasc=.TRUE.</code> on the &OUTP line.
UHC(nn)	The unburned hydrocarbon mass fraction (kg UHC/kg gas) in the duct node set by the nn^{th} &RNOD line. Only written if <code>gasfuel=.TRUE.</code> on the &OUTP line.
ncomp	The number of &COMP lines as specified on the &CNTR line.

If there are no ducts, then this file will not be created. Also this file will not be created if all `gas???` variables are set to `.FALSE.` on the `&OUTP` line. Additional files will be required if there are more than 254 / (number of output gases) duct nodes.

15. Energy release output data – `[fileid]_fire#.csv`

This file gives the theoretical maximum energy release data in each compartment. The output file contains a description header followed by rows of output data in the following format:

time, FireHRR(c1), FireHRR(c2), ..., FireHRR(ncomp)

- `time` The time (s) that the energy release data correspond to.
- `FireHRR(cn)` The heat release rate of the fire (kW) as specified by the `&FIRE` line or the `&USES` and `&FUEL` lines for the compartment set by the `cnth` `&COMP` line.
- `ncomp` The number of `&COMP` lines as specified on the `&CNTR` line.

16. Surface temperature output data – `[fileid]_stemp#.csv`

This file contains the surface temperature data for surfaces set using a `&SURF` line. The surface front is defined as the face of the surface that is located in the compartment specified with the first element in the `&SURF location` parameter array. The surface back is the face of the surface that is located in the compartment specified by the second element in the `&SURF location` parameter array.

The output file contains a description header followed by rows of output data in the following format:

time, stempF(s1), stempB(s1), stempF(s2), stempB(s2), ..., stempF(nsurf), stempB(nsurf)

- `time` The time (s) that the surface temperature data correspond to.
- `stempF(sn)` The front temperature (K) of the surface set by the `snth` `&SURF` line.
- `stempB (sn)` The back temperature (K) of the surface set by the `snth` `&SURF` line.
- `nsurf` The number of `&SURF` lines as specified on the `&CNTR` line.

If there are no surfaces, then this file will not be created. Additional output files will be required if there are more than 127 surfaces.

17. Surface heat flux output data – [fileid]_sflux#.csv

This file contains the surface heat flux data for surfaces set using a &SURF line. The surface front and back are as described for the surface temperature data. Positive heat flux is from the compartment to the surface. The output file contains a description header followed by rows of output data in the following format:

time, scF(s1), srF(s1), scB(s1), srB(s1), scF(s2), srF(s2), scB(s2), srB(s2), ..., scF(nsurf), srF(nsurf), scB(nsurf), srB(nsurf)

time	The time (s) that the surface temperature data correspond to.
scF(sn)	The front net convective heat flux (W) of the surface set by the sn th &SURF line.
srF (sn)	The front net radiant heat flux (W) of the surface set by the sn th &SURF line.
scB(sn)	The back net convective heat flux (W) of the surface set by the sn th &SURF line.
srB (sn)	The back net radiant heat flux (W) of the surface set by the sn th &SURF line.
nsurf	The number of &SURF lines as specified on the &CNTR line.

If there are no surfaces, then this file will not be created. Additional output files will be required if there are more than 63 surfaces.

18. Detector actuation status data – [fileid]_detect#.csv

This file contains the actuation status of all detectors in the configuration modeled. A detector is either actuated (1 [. TRUE.]) or not actuated (0 [. FALSE.]). The output file contains a description header followed by rows of output data in the following format:

time, status(d1), status(d2), ..., status(ndetect)

<code>time</code>	The time (s) that the detector actuation status information correspond to.
<code>status(d1)</code>	The detector actuation status of the detector set by the first <code>&DTCT</code> line.
<code>status(d2)</code>	The detector actuation status of the detector set by the second <code>&DTCT</code> line.
<code>status(ndetect)</code>	The detector actuation status of the detector set by the <code>ndetect</code> th <code>&DTCT</code> line.
<code>ndetect</code>	The number of <code>&DTCT</code> lines as specified on the <code>&CNTR</code> line.

If there are no detectors, then this file will not be created.

19. Suppression system actuation status data – `[fileid]_suppress#.csv`

This file contains the actuation status of all suppression systems in the configuration modeled. A suppression system is either actuated (1 [`.TRUE.`]) or not actuated (0 [`.FALSE.`]). This is solely and indication of the control function that operates the suppression system and not whether or not the suppression system has exceeded its `discharge_t`. The output file contains a description header followed by rows of output data in the following format:

`time, status(s1), status(s2), ..., status(nsuppress)`

<code>time</code>	The time (s) that the detector actuation status information correspond to.
<code>status(sn)</code>	The suppression system actuation status of the suppression system set by the <code>sn</code> th <code>&SUPR</code> line.
<code>nsuppress</code>	The number of <code>&SUPR</code> lines as specified on the <code>&CNTR</code> line.

If there are no suppression systems, then this file will not be created.

5.2 MSU Viewer Output Format

When the `&EXEC outtype` parameter is set to `.TRUE.`, FSSIM will generate a single CSV output file containing data in a format that may be read by the MSU viewer [Haupt, *et al.*, 2004]. The output file is named `[fileid]_out.csv`, where `fileid` is set by the `&EXEC` parameter group line (Section 4.2.1).

The MSU output file contains all output information for all timesteps within a single column of data. The output file consists of series of headers followed by data. Headers typically

occupy two sequential rows: the first row contains text that identifies the variable and the second row lists the units that the variable is reported in. Data between timestep headers correspond to the first time below the first timestep header. Data below other headers corresponds to the variable identified by the header the data are below. This pattern is repeated until the last timestep. The data, in the row order in which it is contained, are as follows:

- Time (s) – single row below timestep header.
- Compartment temperature (K) – next n_{comp} rows below appropriate header.
- Front surface temperature (K) – next n_{surf} rows below appropriate header.
- Back surface temperature (K) – next n_{surf} rows below appropriate header.
- Front surface convective heat flux (kW/m^2) – next n_{surf} rows below appropriate header.
- Back surface convective heat flux (kW/m^2) – next n_{surf} rows below appropriate header.
- Front surface radiant heat flux (kW/m^2) – next n_{surf} rows below appropriate header.
- Back surface radiant heat flux (kW/m^2) – next n_{surf} rows below appropriate header.
- Compartment pressure (Pa) – next n_{comp} rows below appropriate header.
- Duct node pressure (Pa) – next n_{node} rows below appropriate header.
- Oxygen mass fraction ($\text{kg O}_2/\text{kg gas}$) – next n_{comp} rows below appropriate header.
- Carbon monoxide mass fraction ($\text{kg CO}/\text{kg gas}$) – next n_{comp} rows below appropriate header.
- Soot mass fraction ($\text{kg soot}/\text{kg gas}$) – next n_{comp} rows below appropriate header.
- Compartment heat release rate (kW) – next n_{comp} rows below appropriate header.
- Compartment fire size (kW) – next n_{comp} rows below appropriate header.
- Suppression system actuation status – next $n_{suppress}$ rows below appropriate header.

where:

n_{comp} is the number of &COMP lines as specified on the &CNTR line.

n_{surf} is the number of &SURF lines as specified on the &CNTR line.

n_{node} is the number of &RNOD lines as specified on the &CNTR line.

$n_{suppress}$ is the number of &SUPR lines as specified on the &CNTR line.

6.0 SAMPLE CASES

Four samples are provided to illustrate various modeling features of FSSIM. The samples are as follows:

- Simple configuration with a fire in a space that is connected to another space via a door. Each space is vented to the exterior via a window. Boundaries are steel.
- A fire in a single space with an irregular floor plan. The boundaries are insulated steel and there is a fixed water-based suppression system.
- A fire in a single 'U'-shaped space that must be broken into three compartments in the model. The boundaries are a mix of steel and gypsum.
- Five spaces connected by a single corridor. Three of the spaces are mechanically vented.

6.1 Sample 1: Fire in Steel Compartment Connected to One Adjacent Compartment

This sample consists of two spaces with steel boundaries that are connected via a single door that remains open. Each space is vented to ambient via a window. A fire is located in one of the spaces. Refer to Figure 6-1 for a depiction of Sample 1. The following details apply:

- Each space is 5 m x 5 m by 3 m tall. The boundaries are 15.8 mm thick carbon steel.
- The door between the spaces is 2.13 m by 0.91 m wide. The door is located in the fore of the fire compartment and the aft of the adjacent compartment.
- The window opening in each space measures 0.91 m by 0.61 m wide and is located 1.22 m above the deck. The window in the fire compartment is located on the starboard side and the window in the adjacent compartment is located on the forward side.
- The source fire is located in Room 1 and is a 'fast' t^2 fire with a peak heat release rate of 1,000 kW. The heat of combustion of the burning material is 28,100 kJ/kg. The default species yields are applicable and the radiant fraction is 0.3.

This is a simple example that illustrates how to construct an input file and set various parameters for a particular simulation.

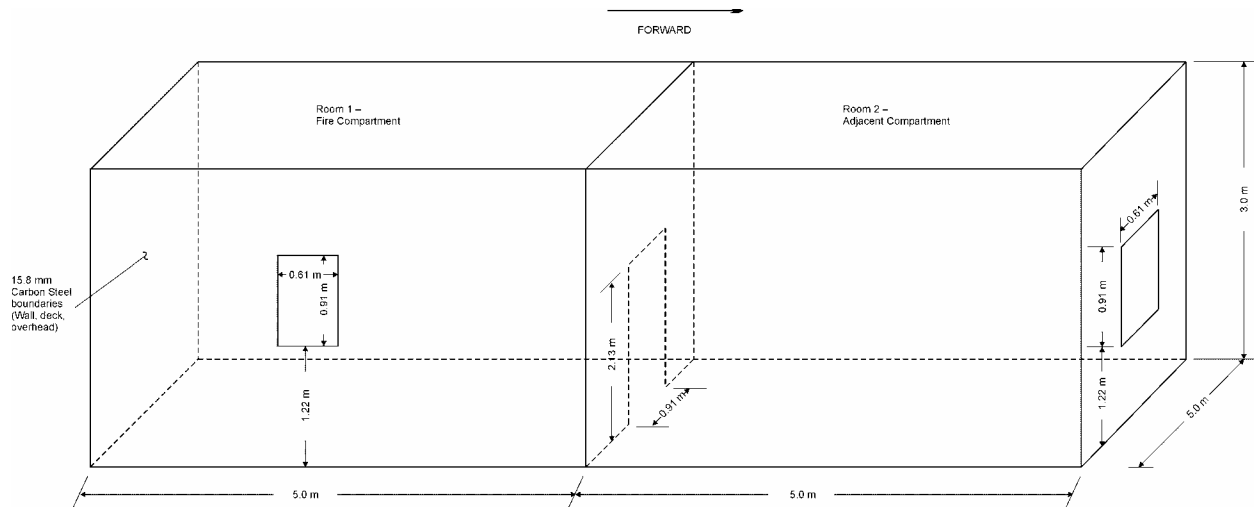


Figure 6-1 Sample 1 (Not to Scale): Two adjacent steel compartments

The input file for Sample 1 is contained in Appendix A-1. The simulation is 1,200 seconds long. The material properties for the carbon steel are contained in the file `materials.txt`. The portions of this file used by Sample 1 are also contained in Appendix A-1. Note that fake surfaces are applied to the openings to allow for thermal radiation heat transfer. The calculated temperature and pressure in the two compartments are shown in Figure 6-2. Figure 6-3 shows the concentration of various gas and combustion products in the fire compartment as well as the heat release rate of the fire. Other outputs available, but not shown, include the boundary temperature, the flux to the boundaries, and vent flow information.

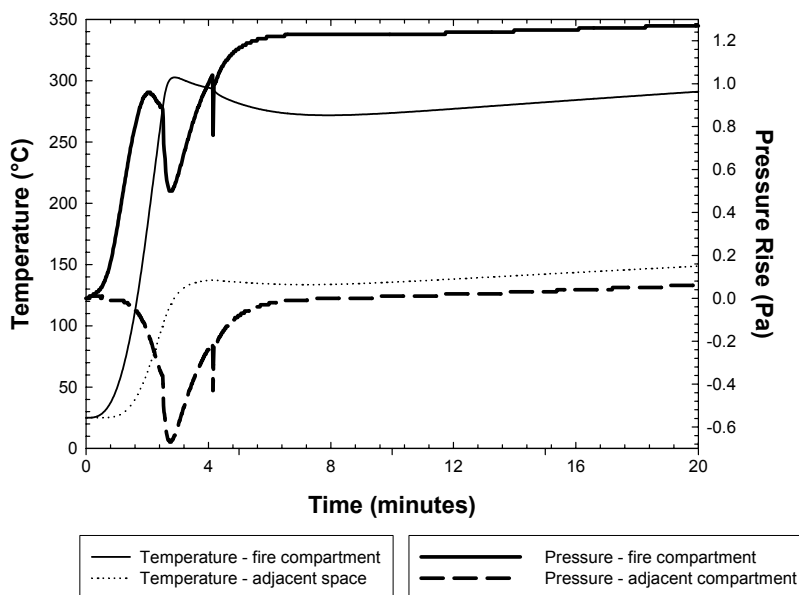


Figure 6-2. Calculated Compartment Temperature and Pressure as a Function of Time

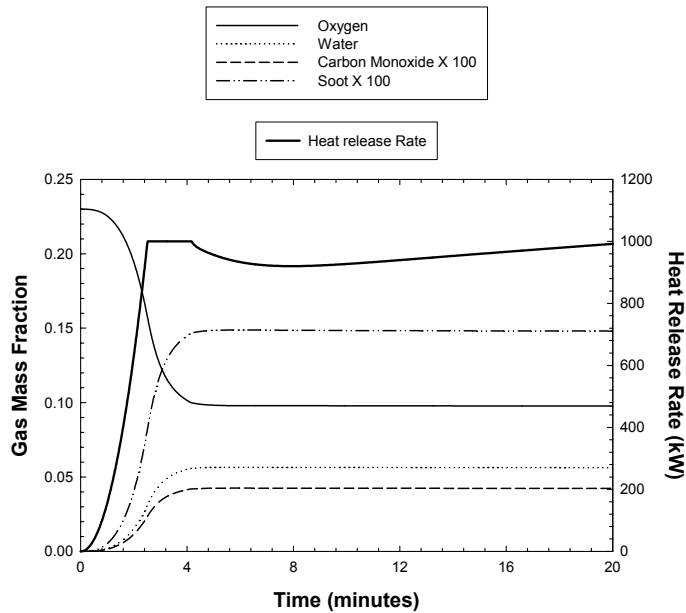


Figure 6-3. Calculated Mass Fraction of Various Gases in Fire Compartment and Fire Heat Release Rate

6.2 Sample 2: Fire in Insulated Steel Compartment with Irregular Plan and Fixed Suppression

This sample consists of a single space with an irregular floor plan. The boundaries are steel with insulation backing. There is one door and one vent near the overhead that connects the space to the ambient surroundings. Refer to Figure 6-4 for a depiction of Sample 2. The following details apply:

- The space has four walls with lengths in clockwise order beginning with the starboard wall of 4.0 m, 3.0 m, 7.0 m, and 5.0 m. The overhead height is 3.0 m.
- There is a 2.13 m tall by 0.91 m wide door in the aft wall that connects the space to ambient conditions.
- There is a 0.3 m by 1.2 m wide vent in the port wall located 0.5 m below the overhead that connects the space to ambient conditions.
- The walls and overhead are 15.8 mm thick carbon steel with 25 mm ceramic fiberboard insulation for backing. The floor is 25 mm thick carbon steel with 38 mm Fiberfrax blanket insulation for backing.
- The source fire is a ‘medium’ t^2 fire with a peak heat release rate of 800 kW. The heat of combustion of the burning material is 44,000 kJ/kg. The default species yields are applicable and the radiant fraction is 0.3.

- The suppression system is based on a single sprinkler actuating in the space. The sprinkler has an actuation temperature of 74°C and a minimum water flow rate of 1.36 kg/s. The average drop diameter is 500 microns.

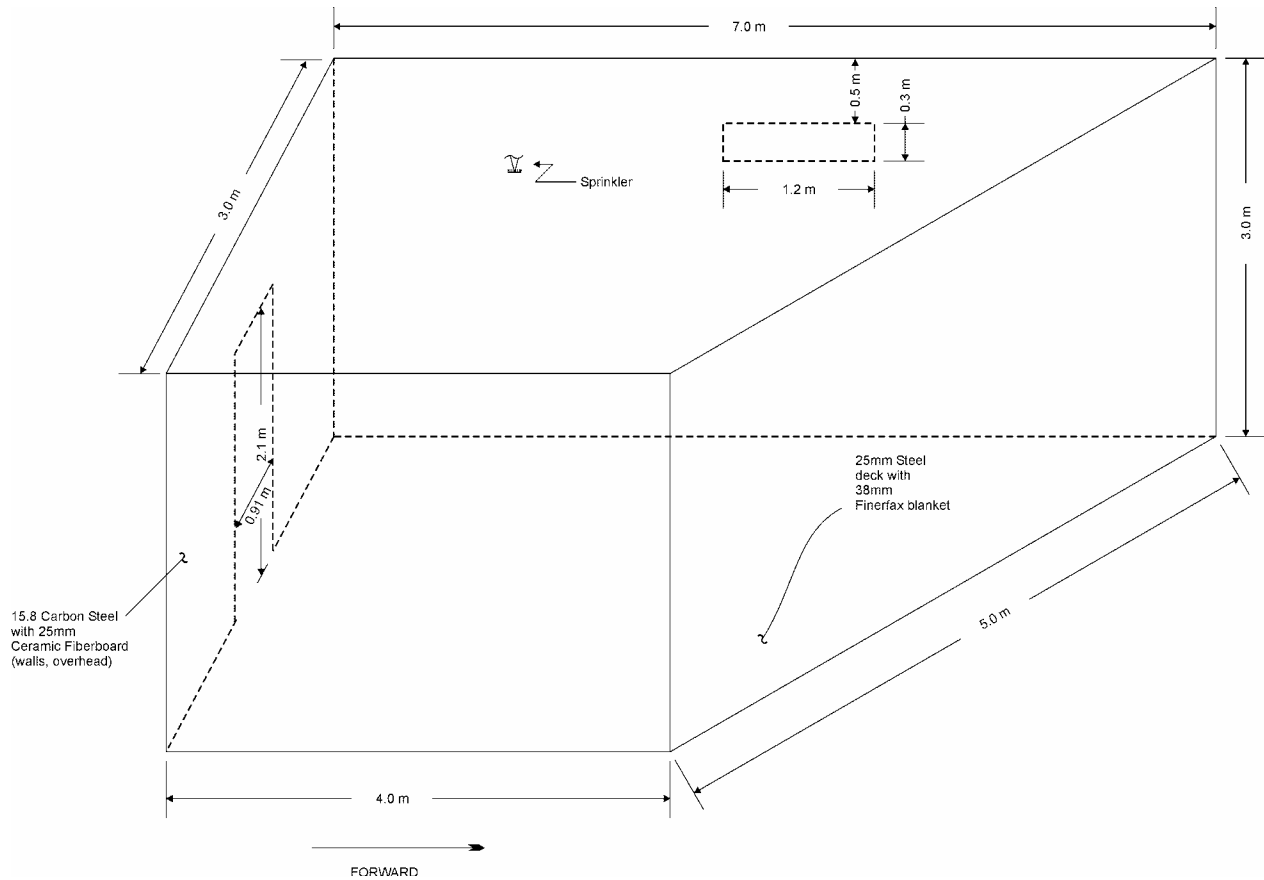


Figure 6-4 Sample 2 (Not to Scale): Simple space with irregular floor plan.

Because FSSIM simulates conditions in volumes with rectangular floor plans and flat decks and overheads, some modifications are necessary to simulate a fire in the space shown in Figure 6-4. The FSSIM compartment definition should have a floor plan with an aspect ratio similar to the actual floor plan. The total volume of the compartment defined in FSSIM should not be less than the actual compartment volume. In the case of Sample 2, the dimension in the port-starboard directions is 3.0 m, which is the actual plane separation of the port and starboard walls. Thus, d_x is set to 3.0 on the &COMP line. Similarly, the dimension in the fore-aft separation range from a minimum of 4.0 m to a maximum of 7.0 m. An average value is used in the simulation, which is done by setting d_y to 5.0 on the &COMP line. In this case, the actual volume of the compartment equals the volume determined by multiplying d_x , d_y , and d_z . It would also be acceptable to set d_y to a value greater than 5.0 but less than or equal to 7.0 without changing the volume (v_{ol}) parameter. This would increase the radiant heat transfer between perpendicular

surfaces and decrease it between parallel surfaces because the radiation configuration factors would change.

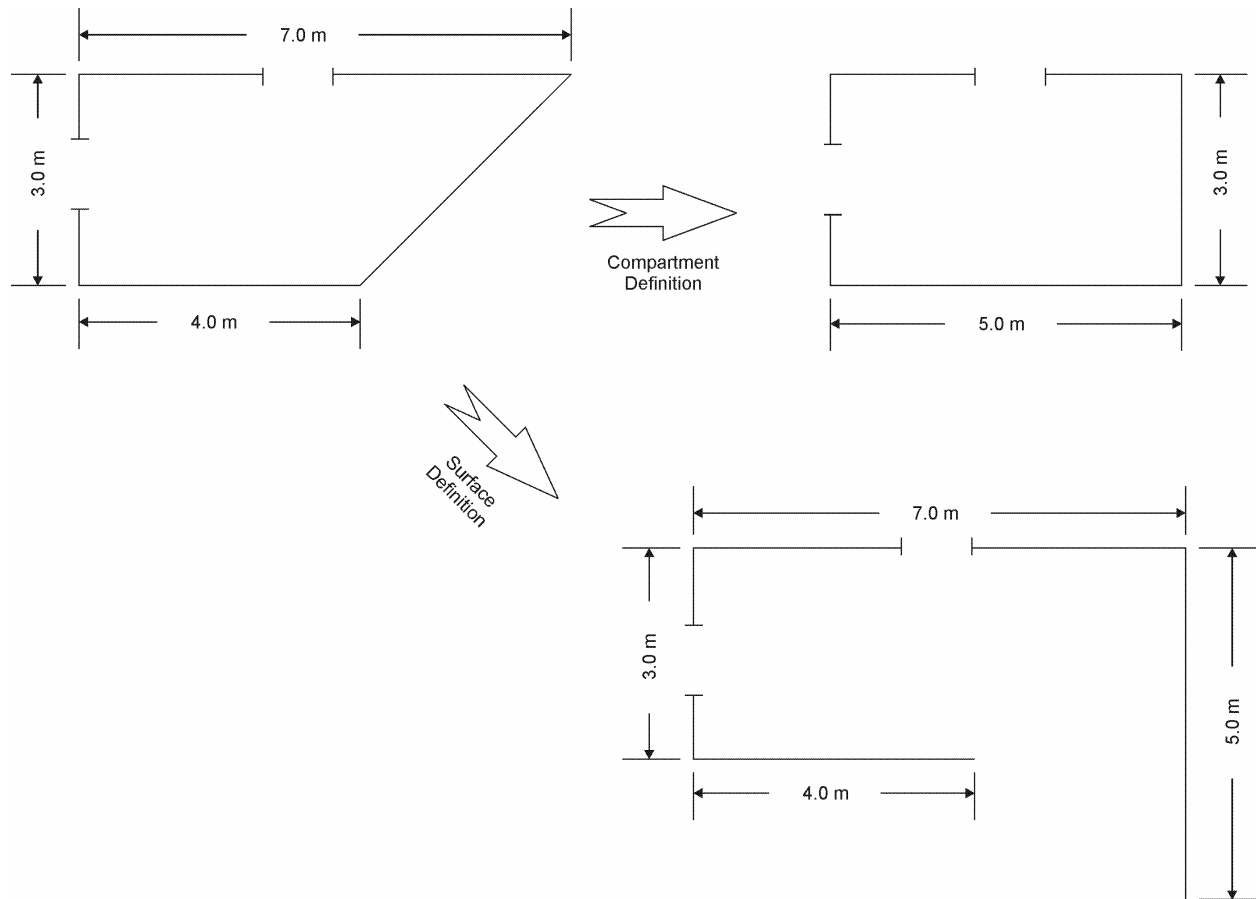


Figure 6-5 Plan View of Sample 2 (Not to Scale)

The input file for Sample 2 is contained in Appendix A-2. The simulation is 1,200 seconds long. The material properties for the carbon steel, ceramic fiberboard, and Fiberfrax blanket insulation are contained in the file 'materials.txt'. The portions of this file used by Sample 2 are also contained in Appendix A-2. Note that fake surfaces are applied to the openings to allow for thermal radiation heat transfer. The calculated temperature and pressure in the two compartments are shown in Figure 6-6. Figure 6-7 shows the concentration of various gas and combustion products in the fire compartment as well as the heat release rate of the fire. Other outputs available, but not shown, include the boundary temperature, the flux to the boundaries, and vent flow information. Figure 6-7 indicates that the sprinkler actuates and begins suppressing the source fire around 4 minutes after ignition.

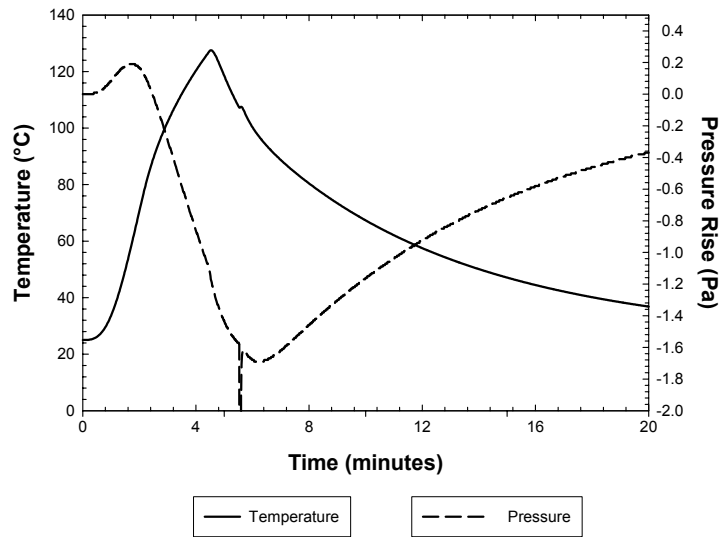


Figure 6-6. Calculated Compartment Temperature and Pressure as a Function of Time

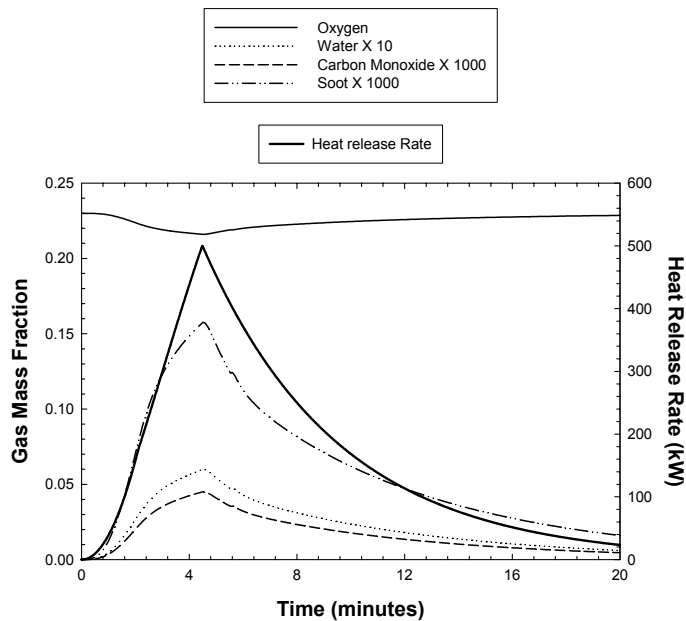


Figure 6-7. Calculated Mass Fraction of Various Gases in Fire Compartment and Fire Heat Release Rate

6.3 Sample 3: Fire in 'U'-Shaped Compartment with Gypsum and Steel Boundaries

This sample consists of a single space with a 'U'-shape floor plan where the 'U' opens to the aft direction. The boundaries are a mix of steel and gypsum wallboard. There is one door in

the port section of the space and two vents, one on the starboard boundary and one on the port boundary. Sample 3 is shown in Figure 6-8. The following details apply:

- The fire is located in the starboard section of the compartment. The default species yields are applicable and the radiant fraction is 0.3.
- The starboard and port walls are 11 m long; the forward wall is 12 m long; the starboard and port sections are 3 m wide and the forward section is 4 m wide.
- The overhead, aft and port walls of the starboard section and the aft and starboard walls of the port section are 15.8 mm thick gypsum. The remaining surfaces are 12.7 mm thick carbon steel.
- The overhead height is 3 m in all three sections.
- The source fire has a linear growth rate from 0 kW to 1,200 kW for 120 seconds after ignition. The heat of combustion of the burning material is 44,000 kJ/kg.

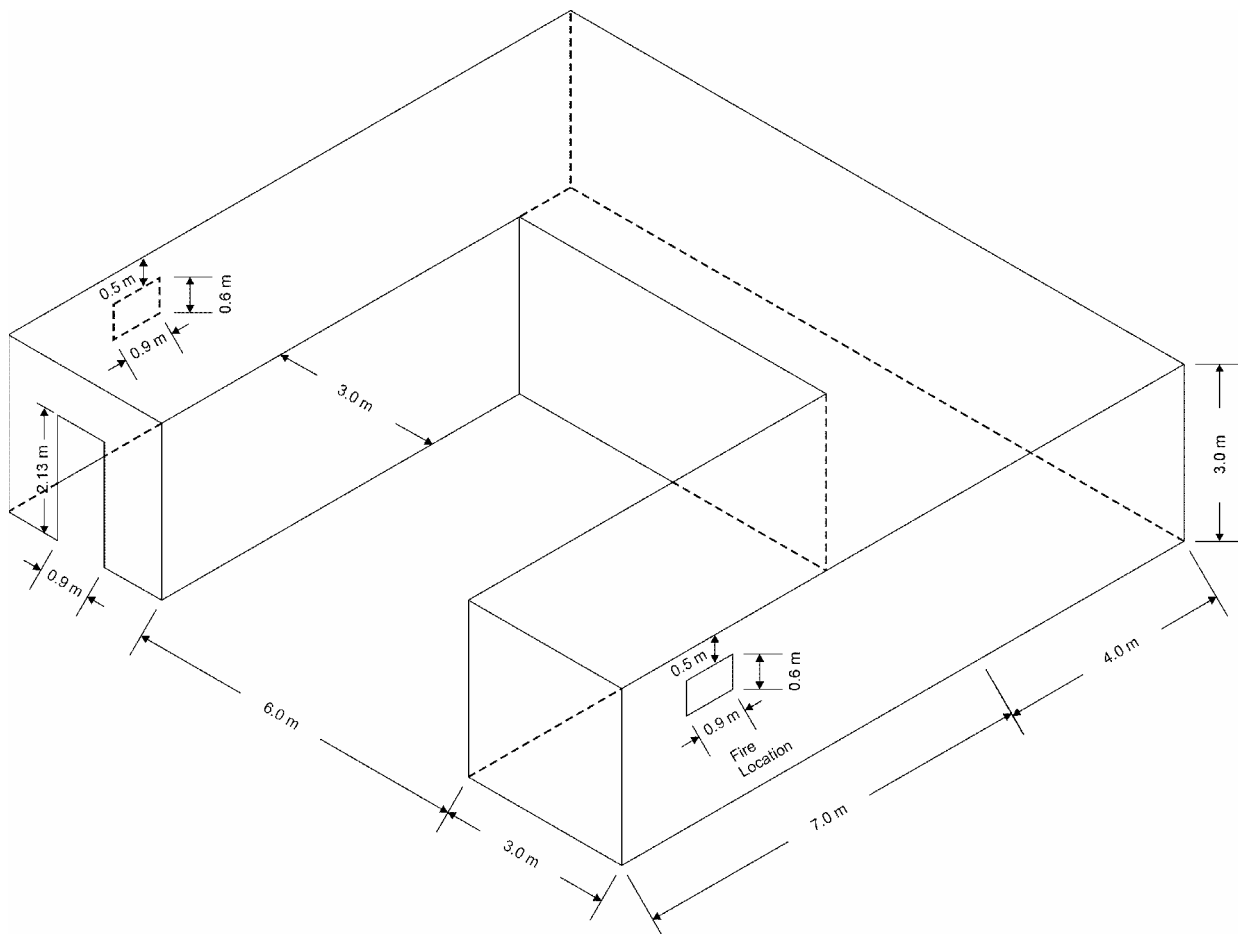


Figure 6-8 Sample 3 (Not to Scale): Single space with a “U”-shaped floor plan

Because FSSIM simulates conditions in volumes with rectangular floor plans and flat decks and overheads, some modifications are necessary to simulate a fire in the space shown in

Figure 6-8. The space was divided into three sub-compartments: the port section, the forward section, and the aft section. The floor plan is shown in Figure 6-9. The three sub-compartments are connected using an opening with a cross sectional area equal to the width of the forward compartment times the height of the overhead. A fake boundary was specified at the connections to allow for radiant heat transfer. The loss coefficient was reduced to 1.0, which corresponds to a discharge coefficient of 1.0.

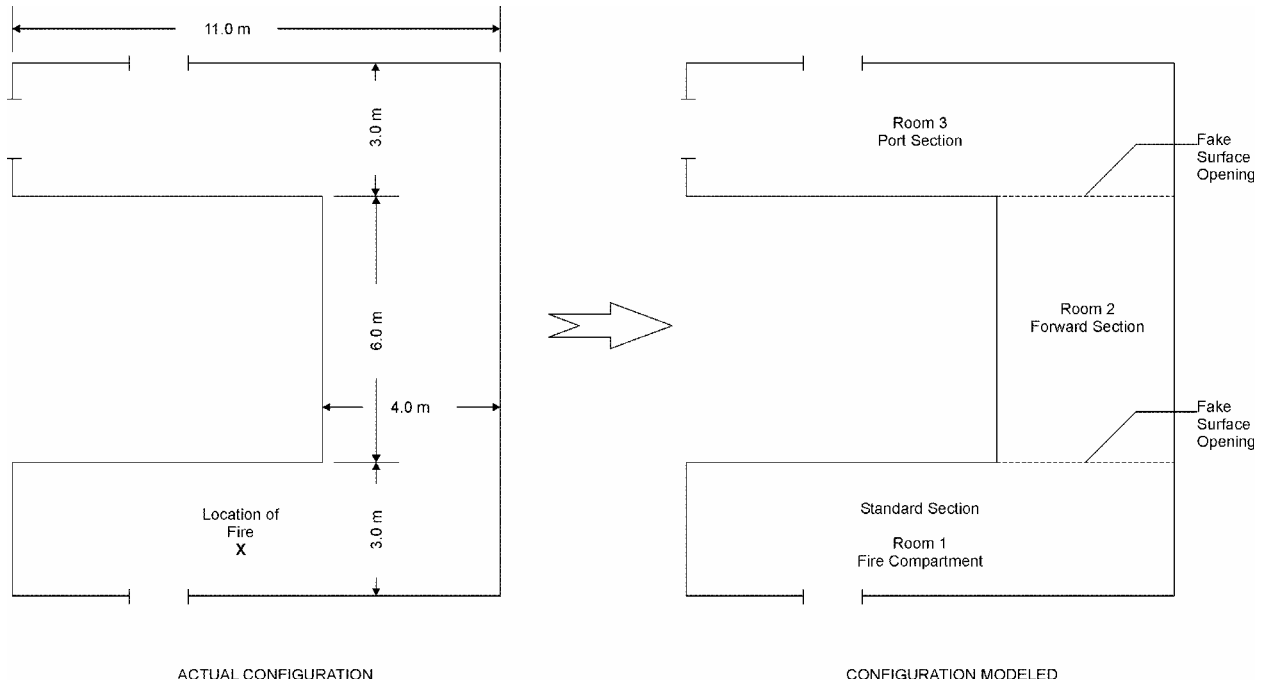


Figure 6-9 Plan View of Sample 3 Configuration Modeled using FSSIM (Not to Scale)

The input file for Sample 3 is contained in Appendix A-3. The simulation is 1,200 seconds long. The material properties for the carbon steel and gypsum are contained in the file 'materials.txt'. The portions of this file used by Sample 3 are also contained in Appendix A-3. Note that fake surfaces are applied to the openings to allow for thermal radiation heat transfer. The calculated temperature and pressure in the two compartments are shown in Figure 6-10. Figure 6-11 shows the concentration of various gas and combustion products in the fire compartment as well as the heat release rate of the fire. Other outputs available, but not shown, include the boundary temperature, the flux to the boundaries, and vent flow information.

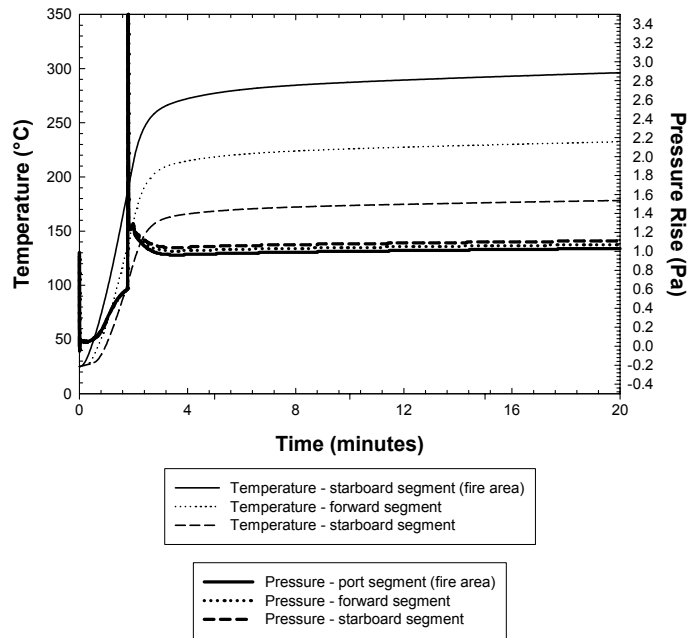


Figure 6-10. Calculated Compartment Temperature and Pressure as a Function of Time

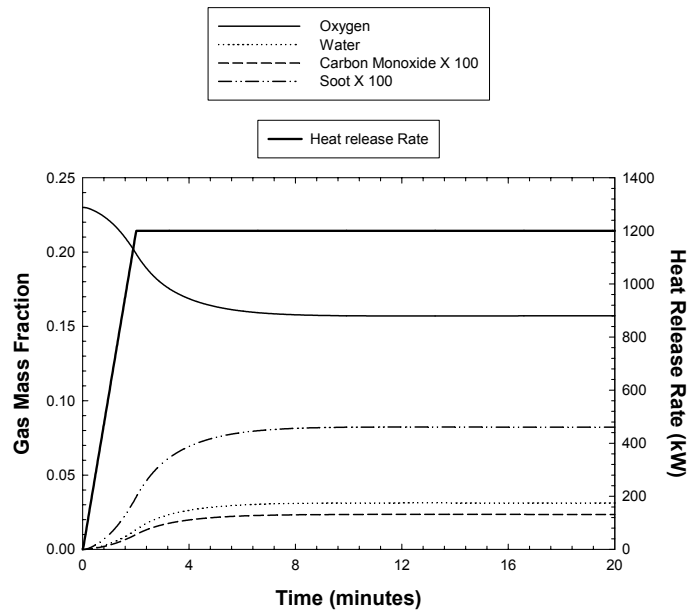


Figure 6-11. Calculated Mass Fraction of Various Gases in Fire Segment and Fire Heat Release Rate

6.4 Sample 4: Multiple Compartment Arrangement Connected by a Corridor

This sample consists of five spaces connected by a corridor. The boundaries are 15.8 mm thick gypsum. Each space is connected to the corridor via a single door. The corridor is

connected to ambient at one end via a standard double door that remains open for the duration of the scenario. Three of the spaces are connected to a mechanical ventilation system. The two spaces that do not have mechanical ventilation have a small vent to ambient located near the ceiling. A plan view of Sample 4 is shown in Figure 6-12. The following details apply:

- The fire is in Room 3 as shown in Figure 6-12. The default species yields are applicable and the radiant fraction is 0.3.
- The plan dimensions are as shown in Figure 6-12. The overhead elevation is 3 m.
- Single doors are 0.91 m wide by 2.13 m tall; double doors are 1.82 m wide by 2.13 m tall.
- The source fire has a medium growth rate with a maximum heat release rate of 2,500 kW. The heat of combustion of the burning material is 14,000 kJ/kg.
- All ductwork associated with the mechanical ventilation system has a rectangular cross section measuring 0.61 m by 0.61 m. The duct centerline elevation is 2.5 m. The duct lengths are as shown in Figure 6-12.
- The duct roughness is 0.0005 m.
- The fan performance curve is quadratic. The flow at zero pressure is 1.86 m³/s and the pressure at zero flow is 180 Pa.
- The pressure loss coefficients associated with the duct ‘T’ connections are as follows:
 - Flow involving ‘T’ stem (flow changes direction): 1.63
 - ‘T’ crosses (no change in flow direction): 0.75
- The pressure loss coefficients associated with an elbow is 1.1 in either flow direction through the elbows.
- The pressure loss coefficient associated with inlets and outlets is 0.0.

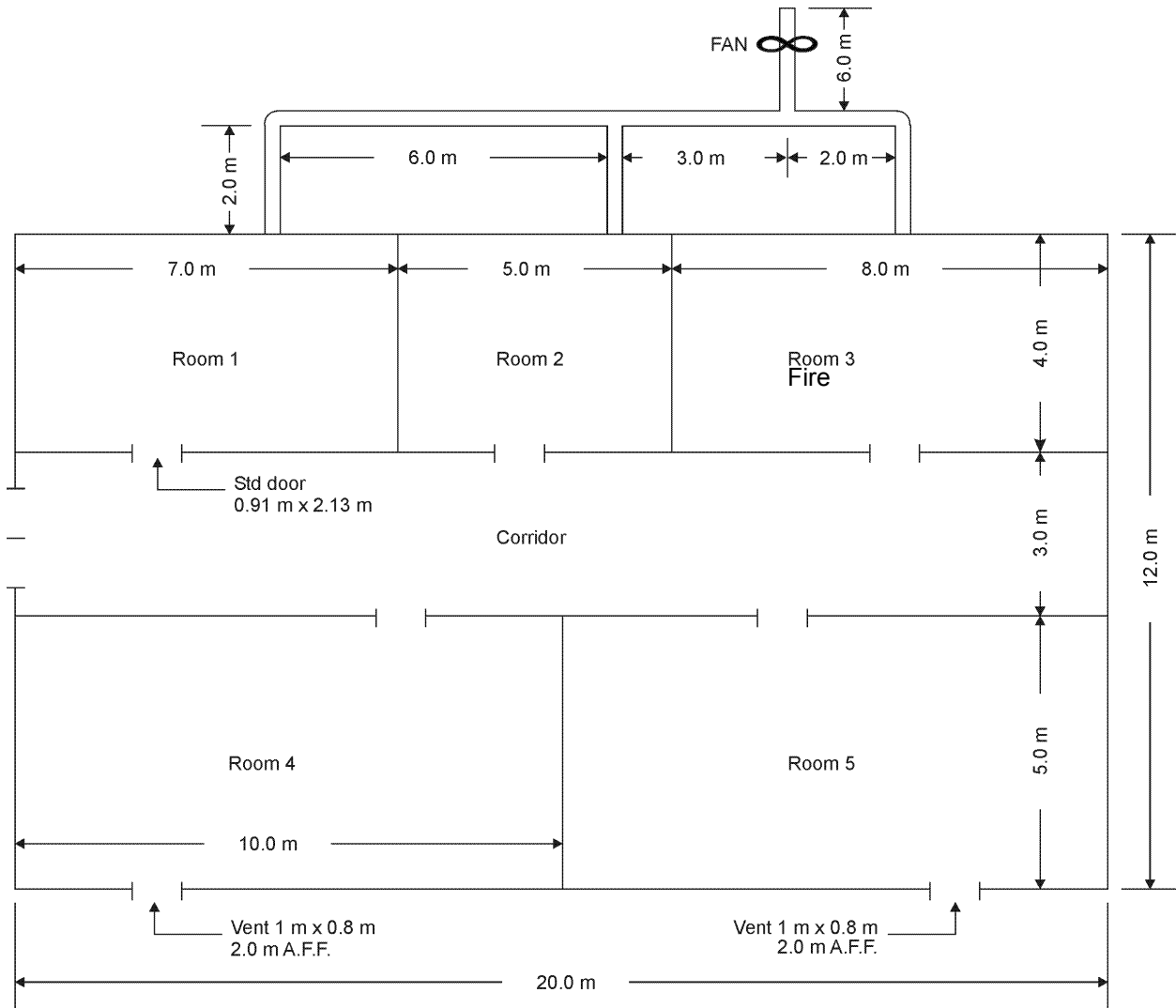


Figure 6-12 Sample 4 (Not to Scale)

As was the case for Samples 2 and 3, some modifications to the actual floor plan are made to simulate a fire in the space shown in Figure 6-12. Each individual space was simulated using the dimensions shown in Figure 6-12. The corridor was subdivided into three approximately equally volume segments for the same reasons described for Sample 3. While this is not necessary, it may improve mass and energy transfers to the remote compartments. The floor plan of the configuration simulated using FSSIM is shown in Figure 6-13. The three sub-compartments are connected using an opening with a cross sectional area equal to the width of the forward compartment times the height of the overhead. A fake boundary was specified at the connections to allow for radiant heat transfer. The loss coefficient was reduced to 1.0, which corresponds to a discharge coefficient of 1.0.

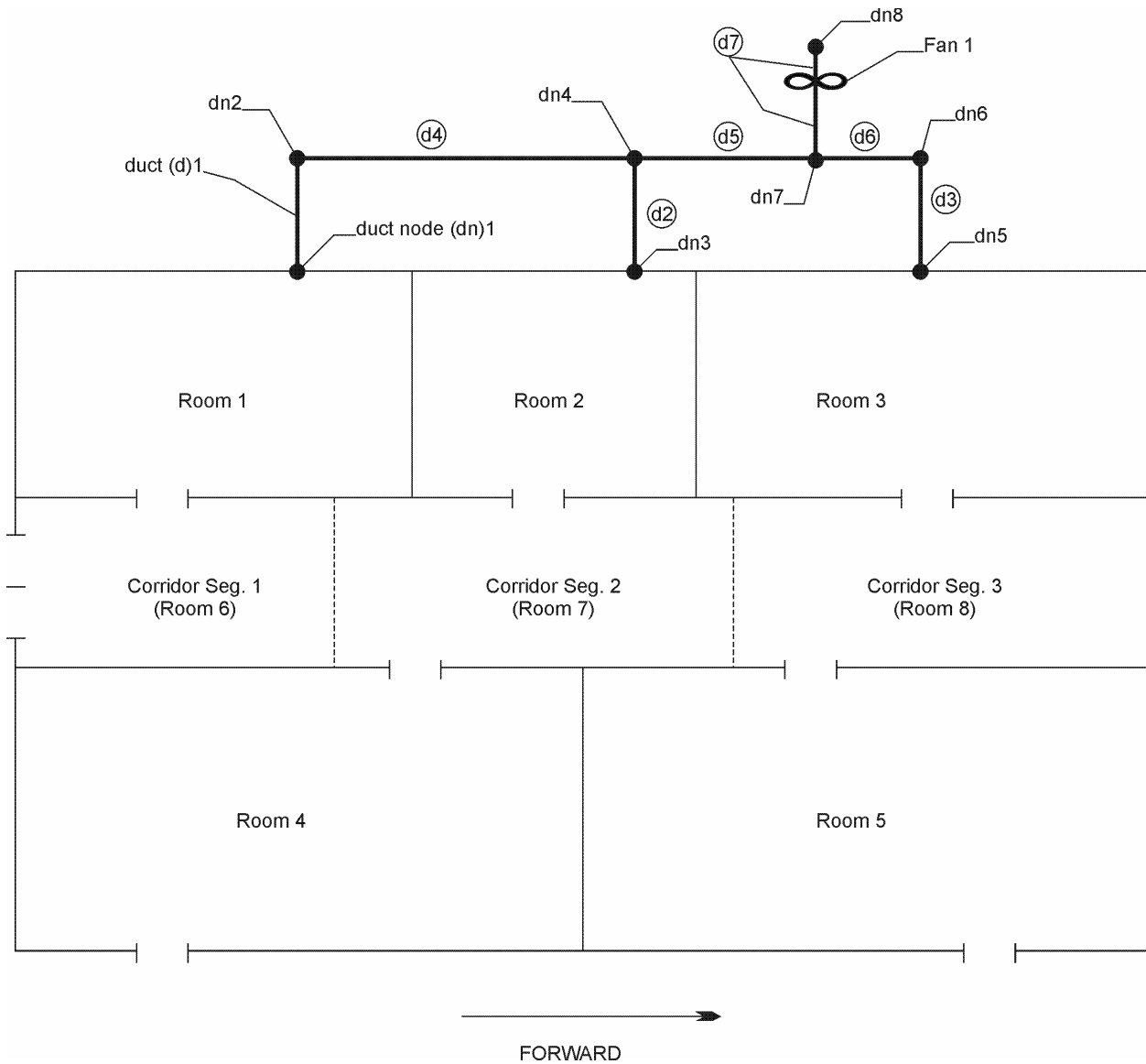


Figure 6-13 Plan View of Sample 4 Configuration Modeled using FSSIM (Not to Scale)

The input file for Sample 4 is contained in Appendix A-4. The simulation is 1,200 seconds long. The material properties for gypsum are contained in the file 'materials.txt'. The portions of this file used by Sample 4 are also contained in Appendix A-4. Note that fake surfaces are applied to the openings to allow for thermal radiation heat transfer. The calculated temperature in the five compartments and the three corridor segments is shown in Figure 6-14. Likewise, the calculated compartment pressures are shown in Figure 6-15. Figure 6-16 shows the concentration of various gas and combustion products in the fire compartment as well as the heat release rate of the fire. Other outputs available, but not shown, include the boundary temperature, the flux to the boundaries, and vent flow information

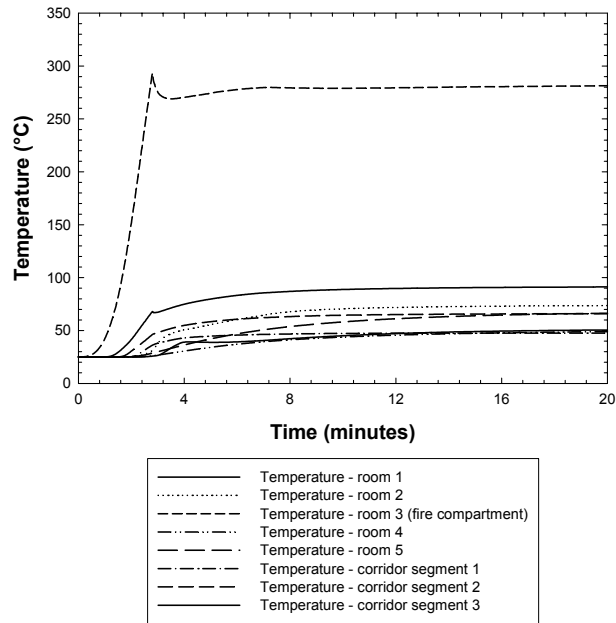


Figure 6-14. Calculated Compartment Temperatures as a Function of Time

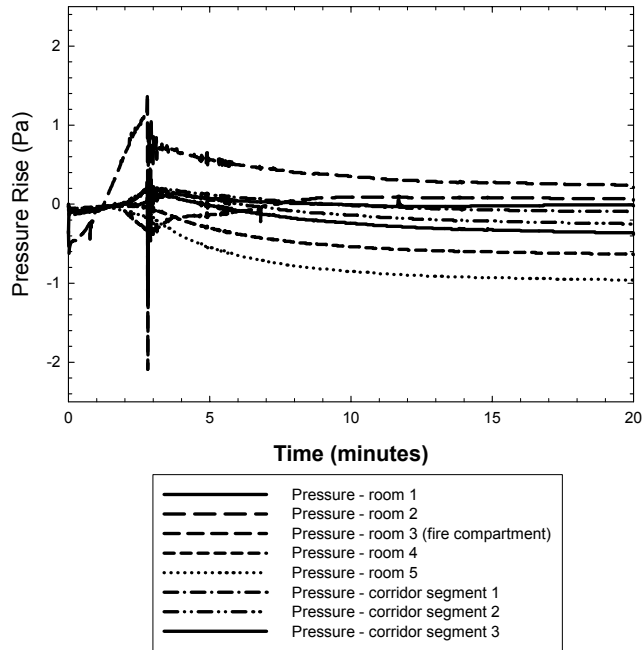


Figure 6-15. Calculated Compartment Pressures as a Function of Time

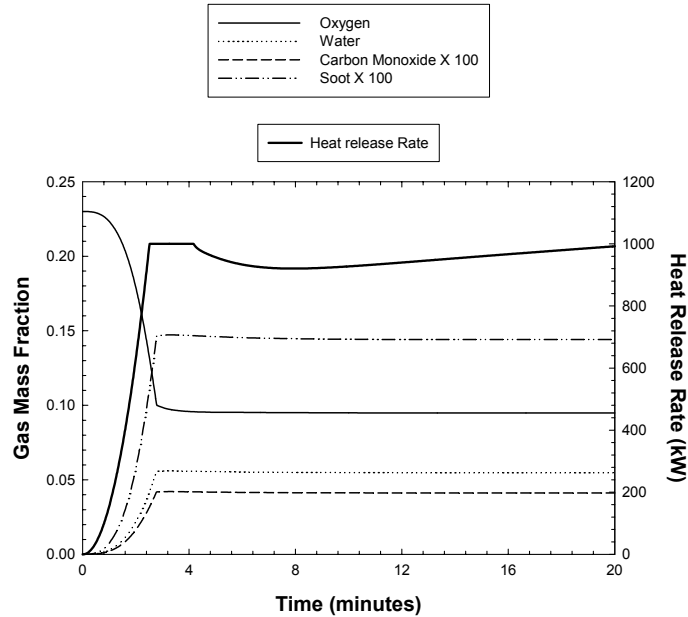


Figure 6-16. Calculated Mass Fraction of Various Gases in Fire Segment and Fire Heat Release Rate

7.0 REFERENCES

Adams, J. C., Brainerd, W. S., Martin, J. T., Smith, B. T., and Wagener, J. L., *Fortran 95 Handbook Complete ISO/ANSI Reference*, The MIT Press, Cambridge, MA, 1997.

Allelein, H. J. *et al.*, "Entwicklung und Verifikation eines Containment-Codesystems (COCOSYS) und eines deutsch-französischen Integralcodes (ASTEC)," GRS-A-2736, GRS-A-2737, Gesellschaft für Anlagen – und Reaktorsicherheit (GRS) mbH, 1999.

ASHRAE, *Heating, Ventilating, and Air-Conditioning Applications 1991 ASHRAE Handbook*, SI Edition, American Society of Heating, Refrigerating, and Air Conditioning Engineers, Inc., Atlanta, GA, 1991.

Babrauskas, V., "Burning Rates," Section 3, Chapter 1, *The SFPE Handbook of Fire Protection Engineering*, 3rd Edition, National Fire Protection Association, Quincy, MA, 2002.

Babrauskas, V., "Tables and Charts," Appendix A, *NFPA Fire Protection Handbook*, Nineteenth Edition, National Fire Protection Association, Quincy, MA, 2003.

Babrauskas, V. and Grayson, S. J., *Heat Release in Fires*, Elsevier Applied Science, New York, New York, 1992.

Beyler, C. and DiNenno, P., "Letters to the Editor," *Fire Technology*, Vol. 27, No. 2, 1991.

Brater, E. F., King, H. W., Lindell, J. E., and Wei, C. Y., *Handbook of Hydraulics*, Seventh Edition, McGraw-Hill, New York, New York, 1996.

Crane Company, "How Fluids Flow through Valves, Fittings, and Pipe," Technical Paper No. 410, Twenty-fifth printing, Engineering Department, Crane Company, Joliet, IL, 1991.

DiNunno, P. J., et al., "Halon Replacement Clean Agent Total Flooding Systems," Section 4, Chapter 7, *The SFPE Handbook of Fire Protection Engineering*, 3rd Edition, National Fire Protection Association, Quincy, MA, 2002.

Dols, W., G. Walton, and K. Denton, "CONTAMW 1.0 User Manual: Mutizone Airflow and Contaminant Transport Analysis Software," NISTIR 6476, National Institute of Standards and Technology, Gaithersburg, MD, 2000.

Gauntt, R. O., Cole, R. K., Erickson, C. M., Gasser, R. D., Rodriguez, S. B., and Young, M. F., "MELCOR Computer Code Manuals," NUREG/CR-6119, Volume 2, Revision 2, U. S. Nuclear Regulatory Commission, Washington, D. C., December, 2000.

Flemming, R. P., "Automatic Sprinkler System Calculations," Section 4, Chapter 3, *The SFPE Handbook of Fire Protection Engineering*, 3rd Edition, National Fire Protection Association, Quincy, MA, 2002.

Floyd, J., S. Hunt, F. Williams, and P. Tatem. "Fire + Smoke Simulator Version 1 (FSSIM) – Theory Manual." NRL. Washington, DC. 2004a.

Floyd, J., F. Williams, and P. Tatem. "Fire + Smoke Simulator Version 1 (FSSIM) – Validation Report." NRL. Washington, DC. 2004b.

Geiman, J. A. and Gottuk, D. T., "Alarm Thresholds for Smoke Detector Modeling," Hughes Associates, Inc., Baltimore, MD, 2001.

George, T., L. Wiles, S. Claybrook, C. Wheeler, J. McElroy, and Singh, A., *GOTHIC Containment Analysis Package Technical Manual Version 7.0*, EPRI RP4444-1, Electric Power Research Institute, Inc., Palo Alto, CA, 2000a.

George, T., L. Wiles, S. Claybrook, C. Wheeler, J. McElroy, and Singh, A., "GOTHIC Containment Analysis Package User Manual Version 7.0," EPRI RP4444-1, Electric Power Research Institute, Inc., Palo Alto, CA, 2000b.

Grant, C. C., "Halon Design Calculations," Section 4, Chapter 6, *The SFPE Handbook of Fire Protection Engineering*, 3rd Edition, National Fire Protection Association, Quincy, MA, 2002.

Hayashi, T., Howell, R., Shibata, M., and Tsuji, K., *Industrial Ventilation and Air Conditioning*, CRC Press, Second Printing, Boca Raton, FL, 1987.

Haupt, T. Manual for FNM-GUI. 2004.

Idelchik, I. E., *Handbook of Hydraulic Resistance*, Hemisphere Publishing Corporation, Second Edition, New York, New York, 1986.

Klein-Heßling *et al.*, "COCOSYS V1.2 Manuals," GRS-P-3, In preparation, Gesellschaft für Anlagen – und Reaktorsicherheit (GRS) mbH, 2000.

Klote, J. H. "Smoke Control," Section 4-12, *Society of Fire Protection Engineers Handbook of Fire Protection Engineering*, Second Edition, National Fire Protection Association, Quincy, MA, 1995.

Lahey Computer Systems, "Lahey/Fujitsu Fortran 95 Language Reference," Lahey Computer Systems, Incline Village, NV, 1998.

Lee, B. T., "Heat Release Rate Characteristics of Some Combustibles Fuel Sources in Nuclear Power Plants," NBSIR 85-3195, U.S. Department of Commerce, National Bureau of Standards (NBS), Washington, DC, July, 1985.

Madrzykowski, Daniel, "Office Work Station Heat Release Rate Study: Full Scale vs. Bench Scale," *Interflam '96*, Proceedings of the 7th International Interflam Conference, Interscience Communications Ltd., Cambridge, England, pp. 47-55, 1996.

Mawhinney, J. R. and Back, G. G., "Water Mist Fire Suppression Systems," Section 4, Chapter 14, *The SFPE Handbook of Fire Protection Engineering*, 3rd Edition, National Fire Protection Association, Quincy, MA, 2002.

NFPA 11, "Low-, Medium-, and High-Expansion Foam," National Fire Protection Association, Quincy, MA, 2002.

NFPA 12, "Carbon Dioxide Extinguishing Systems," National Fire Protection Association, Quincy, MA, 2000.

NFPA 13, "Installation of Sprinkler Systems," National Fire Protection Association, Quincy, MA, 2002.

NFPA 15, "Water Spray Fixed Systems," National Fire Protection Association, Quincy, MA, 2001.

NFPA 16, "Deluge Foam-Water Sprinkler Systems and Foam-Water Spray Systems," National Fire Protection Association, Quincy, MA, 2003.

NFPA 17, "Dry Chemical Extinguishing Systems," National Fire Protection Association, Quincy, MA, 2002.

NFPA 72, "National Fire Alarm Code," National Fire Protection Association, Quincy, MA, 2002.

Nowlen, S. P., "Heat and Mass Release for Some Transient Fuel Sources Fires: A Test Report," NUREG/CR-4680, U.S. Nuclear Regulatory Commission, Washington, DC, October, 1986.

Sabersky, R. H., Acosta, A. J., and Hauptmann, E. G., *Fluid Flow*, Second Edition, Macmillan Publishing Company, New York, New York, 1971.

Sardqvist, S, "Initial Fires RHR, Smoke Production, and CO Generation from Single Items and Room Fire Tests," ISSN 1102-8246, ISRN LUTVDG/TVBE--3070--SE, Lund University, Institute of Technology, Department of Fire Safety, Lund, Sweden, 1993.

Scheffey, J. L. and Hickey, H. E., "Foam System Calculations," Section 4, Chapter 5, *The SFPE Handbook of Fire Protection Engineering*, 3rd Edition, National Fire Protection Association, Quincy, MA, 2002.

Schifiliti, R. P., Meacham, B. J., and Custer, R. L. P., "Design of Detection Systems," Section 4-1, *The SFPE Handbook of Fire Protection Engineering*, 2nd Edition, National Fire Protection Association, Quincy, MA, 2002.

Spore, J., Elson, J., Jolly-Woodruff, S., Knight, T., Lin, J., Nelson, R., *et al.*, "TRAC-M/FORTRAN 90 (Version 3.0) Theory Manual," LA-UR-00-910, Los Alamos National Laboratory, Los Alamos, NM, 2000.

USNRC, *RELAP5/MOD3.3 Code Manual*, United States Nuclear Regulatory Commission, Rockville, MD, 2001.

8.0 NOMENCLATURE

Roman

<i>A</i>	Area (m ²)
<i>c</i>	velocity coefficient
<i>L</i>	Detector separation (m)
<i>n</i>	number of elements in an array
<i>N</i>	Number of ducts
<i>K</i>	Flow loss coefficient
<i>P</i>	Perimeter (m)
<i>x</i>	Compartment x-dimension (m)
<i>y</i>	Yield (kg/kg); Compartment y-dimension (m)

Greek

Superscript

Subscript

<i>CO</i>	Carbon monoxide
-----------	-----------------

CO_2	Carbon dioxide
H_2O	Water
O_2	Oxygen
v	Denotes velocity

Appendix A – Sample input files

A-1 Sample 1

The input file for Sample A-1 is as follows:

```
Sample 1: Two room configuration with t2 fire, 1MW peak

----- General Simulation and Data Control -----
&EXEC t_max=1200.0, fileid='sample1', outtype=.FALSE., pamb=101325.0 /
&CNTR ncomp=2, njunc=3, nnotbidir=0, nfire=1, nsurf=14,
      ncmpn=1, nmat=1, nfan=0, nnode=0, nduct=0, ncontrol=0 /
&FNAM materialfile='materials.txt' /

----- Source Fire and Fuel Load -----
&FIRE id=1, ftype=2, fsubtype=3, location=1, t2_hrr=1000.0, chi_r=0.3,
      deltah_f=28100000.0 /

----- Compartment Information -----
&COMP id=1, compname='Room 1', dx=5.0, dy=5.0, dz=3.0, vol=75.0 /
&COMP id=2, compname='Room 2', dx=5.0, dy=5.0, dz=3.0, vol=75.0 /

&JUNC id=1, location=1,2, height=2.13, elevation=0.0, area=1.948,
      bidirectional=.TRUE., vflow=.FALSE., kloss=2.04, orientation=3 /
&JUNC id=2, location=1,-1, height=0.91, elevation=1.22, area=0.55,
      bidirectional=.TRUE., vflow=.FALSE., kloss=2.16, orientation=6 /
&JUNC id=3, location=2,-1, height=0.91, elevation=1.22, area=0.55,
      bidirectional=.TRUE., vflow=.FALSE., kloss=2.16, orientation=3 /

-- Room 1 --
&SURF id=1, stype=1, location=1,-1, orientation=1, area=25.0, composition=1 /
&SURF id=2, stype=1, location=1,-1, orientation=2, area=25.0, composition=1 /
&SURF id=3, stype=1, location=1,2, orientation=3, area=13.05, composition=1 /
&SURF id=4, stype=1, location=1,-1, orientation=4, area=15.0, composition=1 /
&SURF id=5, stype=1, location=1,-1, orientation=5, area=15.0, composition=1 /
&SURF id=6, stype=1, location=1,-1, orientation=6, area=14.45, composition=1 /

-- Room 2 --
&SURF id=7, stype=1, location=2,-1, orientation=1, area=25.0, composition=1 /
&SURF id=8, stype=1, location=2,-1, orientation=2, area=25.0, composition=1 /
&SURF id=9, stype=1, location=2,-1, orientation=3, area=14.45, composition=1 /
&SURF id=10, stype=1, location=2,-1, orientation=5, area=15.0, composition=1 /
&SURF id=11, stype=1, location=2,-1, orientation=6, area=15.0, composition=1 /

-- Fake surfaces for Radiation heat flow through openings --

&SURF id=50, stype=1, location=1,-1, orientation=6, area=0.55, fake=.TRUE. /
&SURF id=51, stype=1, location=1,2, orientation=3, area=1.948, fake=.TRUE. /
&SURF id=52, stype=1, location=1,-1, orientation=3, area=0.55, fake=.TRUE. /

&CMPN id=1, cmpnname='Carbon Steel Boundary', fileread=.FALSE.,
      layers=1, materials=1, dx=0.0158 /

&MTRL id=1, matname='carbon steel' /

----- Mechanical Ventilation -----
-- None --
```

```
----- Suppression and Detection -----  
-- None --
```

The following carbon steel material property lines are contained in the materials.txt file:

```
&MTRL matname='carbon steel',e=0.8,rho=7860., ktable=.TRUE.,cptable=.TRUE.,  
kname='carbon steel k',cpname='carbon steel cp'/  
  
&CURV id='carbon steel k',xy= 200.15,81.5/  
&CURV id='carbon steel k',xy= 298.15,71.1/  
&CURV id='carbon steel k',xy=1073.15,29.6/  
&CURV id='carbon steel k',xy=1573.15,31.4/  
  
&CURV id='carbon steel cp',xy= 273.15, 438.9/  
&CURV id='carbon steel cp',xy= 348.15, 501.6/  
&CURV id='carbon steel cp',xy= 473.15, 564.3/  
&CURV id='carbon steel cp',xy= 673.15, 627.0/  
&CURV id='carbon steel cp',xy= 873.15, 710.6/  
&CURV id='carbon steel cp',xy= 973.15, 836.0/  
&CURV id='carbon steel cp',xy=1036.15, 836.0/  
&CURV id='carbon steel cp',xy=1041.15,6888.6/  
&CURV id='carbon steel cp',xy=1046.15, 702.2/  
&CURV id='carbon steel cp',xy=1223.15, 668.8/
```

A-2 Sample 2

The input file for Sample A-2 is as follows:

Sample 2: One room configuration with t^2 fire, 800 kW peak

```
----- General Simulation and Data Control -----  
&EXEC t_max=1200.0, fileid='sample2', outtype=.FALSE., pamb=101325.0 /  
  
&CNTR ncomp=1, njunc=2, nnotbidir=0, nfire=1, nsurf=8,  
ncmpn=2, nmat=3, nfan=0, nnode=0, nduct=0, ncontrol=1,  
nsuppress=1 /  
  
&FNAM materialfile='materials.txt' /  
  
----- Source Fire and Fuel Load -----  
&FIRE id=1, ftype=2, fsubtype=2, location=1, t2_hrr=800.0, chi_r=0.3,  
deltah_f=44000000.0 /  
  
----- Compartment Information -----  
&COMP id=1, compname='Room 1', dx=5.5, dy=3.0, dz=3.0, vol=49.5 /  
  
&JUNC id=1, location=1,-1, height=2.13, elevation=0.0, area=1.948,  
bidirectional=.TRUE., vflow=.FALSE., kloss=2.04, orientation=4 /  
&JUNC id=2, location=1,-1, height=0.3, elevation=2.2, area=0.36,  
bidirectional=.TRUE., vflow=.FALSE., kloss=2.16, orientation=5 /  
  
--Room 1 --  
&SURF id=1, stype=1, location=1,-1, orientation=1, area=16.5, composition=1 /  
&SURF id=2, stype=1, location=1,-1, orientation=2, area=16.5, composition=2 /  
&SURF id=3, stype=1, location=1,-1, orientation=3, area=15.0, composition=1 /
```

```

&SURF id=4, stype=1, location=1,-1, orientation=4, area=7.052, composition=1 /
&SURF id=5, stype=1, location=1,-1, orientation=5, area=20.64, composition=1 /
&SURF id=6, stype=1, location=1,-1, orientation=6, area=12.0, composition=1 /

--Fake surfaces for Radiation heat flow through openings--
&SURF id=50, stype=1, location=1,-1, orientation=5, area=0.36, fake=.TRUE. /
&SURF id=51, stype=1, location=1,-1, orientation=4, area=1.948, fake=.TRUE. /

&CMPN id=1, cmpnname='Steel-Ins Boundary A', fileread=.FALSE.,
      layers=2, materials=1,2, dx=0.0158,0.025 /
&CMPN id=2, cmpnname='Steel-Ins Boundary B', fileread=.FALSE.,
      layers=2, materials=1,3, dx=0.025,0.038 /

&MTRL id=1, matname='carbon steel' /
&MTRL id=2, matname='ceramic fiberboard' /
&MTRL id=3, matname='fiberfrax blanket' /

----- Mechanical Ventilation -----
-- None --

----- Suppression and Detection -----
&SUPR id=1, stype=2, operational=.TRUE., comp=1, diam=0.0005,
      flowrate=1.36, control=1 /

-- sprinkler control function --
&CTRL id=1, ctype=1, trip=.FALSE., initial=.FALSE., inputvar=2, location=1,
      setpoint=347.0 /

```

The following carbon steel, fiberfrax blanket insulation, and ceramic fiberboard material property lines are contained in the materials.txt file:

```

&MTRL matname='carbon steel',e=0.8,rho=7860., ktable=.TRUE.,cptable=.TRUE.,
kname='carbon steel k',cpname='carbon steel cp'/

&CURV id='carbon steel k',xy= 200.15,81.5/
&CURV id='carbon steel k',xy= 298.15,71.1/
&CURV id='carbon steel k',xy=1073.15,29.6/
&CURV id='carbon steel k',xy=1573.15,31.4/

&CURV id='carbon steel cp',xy= 273.15, 438.9/
&CURV id='carbon steel cp',xy= 348.15, 501.6/
&CURV id='carbon steel cp',xy= 473.15, 564.3/
&CURV id='carbon steel cp',xy= 673.15, 627.0/
&CURV id='carbon steel cp',xy= 873.15, 710.6/
&CURV id='carbon steel cp',xy= 973.15, 836.0/
&CURV id='carbon steel cp',xy=1036.15, 836.0/
&CURV id='carbon steel cp',xy=1041.15,6888.6/
&CURV id='carbon steel cp',xy=1046.15, 702.2/
&CURV id='carbon steel cp',xy=1223.15, 668.8/

!Properties from manufacturer
&MTRL matname='fiberfrax blanket',e=0.9,rho=128.,ktable=.TRUE.,kname='fiberfrax
blanket k',cp=1130./
&CURV id='fiberfrax blanket k',xy= 473.15,0.04/
&CURV id='fiberfrax blanket k',xy= 673.15,0.07/
&CURV id='fiberfrax blanket k',xy= 873.15,0.12/
&CURV id='fiberfrax blanket k',xy=1073.15,0.18/

&MTRL matname='ceramic fiberboard',e=0.9,rho=256.,ktable=.TRUE.,kname='ceramic
fiberboard k',cp=1130./
&CURV id='ceramic fiberboard k',xy= 311.15,0.043/

```

```

&CURV id='ceramic fiberboard k',xy= 533.15,0.066/
&CURV id='ceramic fiberboard k',xy= 803.15,0.104/
&CURV id='ceramic fiberboard k',xy=1366.15,0.205/

```

A-3 Sample 3

The input file for Sample A-3 is as follows:

Sample 3: U-shaped room configuration with t^2 fire

```

----- General Simulation and Data Control -----
&EXEC t_max=1200.0, fileid='sample3', outtype=.FALSE., pamb=101325.0 /

&CNTR ncomp=3, njunc=5, nnotbidir=0, nfire=1, nsurf=21,
      ncmpn=2, nmat=2, nfan=0, nnode=0, nduct=0, ncontrol=1,
      nsuppress=0 /

&FNAM materialfile='materials.txt' /

----- Source Fire and Fuel Load -----
&FIRE id=1, ftype=3, location=1, chi_r=0.3, deltax_f=44000000.0,
      table='SOURCE FIRE', t_start=0.0 /

&CURV id='SOURCE FIRE', xy=0.0,0.0 /
&CURV id='SOURCE FIRE', xy=120,1200.0 /
&CURV id='SOURCE FIRE', xy=1200,1200.0 /

----- Compartment Information -----
&COMP id=1, compname='Starboard Room', dx=11.0, dy=3.0, dz=3.0, vol=99.0 /
&COMP id=2, compname='Center Room', dx=4.0, dy=6.0, dz=3.0, vol=72.0 /
&COMP id=3, compname='Port Room', dx=11.0, dy=3.0, dz=3.0, vol=99.0 /

-- connections between sub-spaces --
&JUNC id=1, location=1,2, height=3.0, elevation=0.0, area=12.0,
      bidirectional=.TRUE., vflow=.FALSE., kloss=1.00, orientation=5 /
&JUNC id=2, location=2,3, height=3.0, elevation=0.0, area=12.0,
      bidirectional=.TRUE., vflow=.FALSE., kloss=1.00, orientation=5 /

-- openings to ambient --
&JUNC id=3, location=1,-1, height=0.6, elevation=1.9, area=0.54,
      bidirectional=.TRUE., vflow=.FALSE., kloss=2.16, orientation=6 /
&JUNC id=4, location=3,-1, height=0.6, elevation=1.9, area=0.54,
      bidirectional=.TRUE., vflow=.FALSE., kloss=2.16, orientation=5,
      door=1 /
&CTRL id=1, ctype=1, trip=.FALSE., initial=.TRUE., inputvar=2,
      location=3, setpoint=373.15 /

-- door --
&JUNC id=5, location=3,-1, height=2.13, elevation=0.0, area=1.948,
      bidirectional=.TRUE., vflow=.FALSE., kloss=2.04, orientation=4 /

-- Starboard Room --
&SURF id=1, stype=1, location=1,-1, orientation=1, area=33.0, composition=1 /
&SURF id=2, stype=1, location=1,-1, orientation=2, area=33.0, composition=2 /
&SURF id=3, stype=1, location=1,-1, orientation=3, area=9.0, composition=2 /
&SURF id=4, stype=1, location=1,-1, orientation=4, area=9.0, composition=1 /

```



```

&SURF id=5, stype=1, location=1,-1, orientation=5, area=21.0, composition=1 /
&SURF id=6, stype=1, location=1,-1, orientation=6, area=32.46, composition=2 /

-- Center Room --
&SURF id=7, stype=1, location=1,-1, orientation=1, area=18.0, composition=1 /
&SURF id=8, stype=1, location=1,-1, orientation=2, area=18.0, composition=2 /
&SURF id=9, stype=1, location=1,-1, orientation=3, area=18.0, composition=2 /
&SURF id=10, stype=1, location=1,-1, orientation=4, area=18.0, composition=1 /

-- Port Room --
&SURF id=11, stype=1, location=1,-1, orientation=1, area=33.0, composition=1 /
&SURF id=12, stype=1, location=1,-1, orientation=2, area=33.0, composition=2 /
&SURF id=13, stype=1, location=1,-1, orientation=3, area=9.0, composition=2 /
&SURF id=14, stype=1, location=1,-1, orientation=4, area=7.054, composition=1 /
&SURF id=15, stype=1, location=1,-1, orientation=5, area=32.46, composition=2 /
&SURF id=16, stype=1, location=1,-1, orientation=6, area=21.0, composition=1 /

-- Fake surfaces for Radiation heat flow --
&SURF id=50, stype=1, location=1,2, orientation=5, area=12.0, fake=.TRUE. /
&SURF id=51, stype=1, location=2,3, orientation=5, area=12.0, fake=.TRUE. /
&SURF id=52, stype=1, location=1,-1, orientation=6, area=0.54, fake=.TRUE. /
&SURF id=53, stype=1, location=3,-1, orientation=5, area=0.54, fake=.TRUE. /
&SURF id=54, stype=1, location=3,-1, orientation=4, area=1.948, fake=.TRUE. /

&CMPN id=1, cmpnname='Gypsum Boundary', fileread=.FALSE.,
      layers=1, materials=2, dx=0.0158 /
&CMPN id=2, cmpnname='Steel Boundary', fileread=.FALSE.,
      layers=1, materials=1, dx=0.0127 /

&MTRL id=1, matname='carbon steel' /
&MTRL id=2, matname='gypsum' /

----- Mechanical Ventilation -----
-- None --

----- Suppression and Detection -----
-- None --

```

The following carbon steel and gypsum material property lines are contained in the materials.txt file:

```

&MTRL matname='carbon steel',e=0.8,rho=7860., ktable=.TRUE.,cptable=.TRUE.,
kname='carbon steel k',cpname='carbon steel cp'/

&CURV id='carbon steel k',xy= 200.15,81.5/
&CURV id='carbon steel k',xy= 298.15,71.1/
&CURV id='carbon steel k',xy=1073.15,29.6/
&CURV id='carbon steel k',xy=1573.15,31.4/

&CURV id='carbon steel cp',xy= 273.15, 438.9/
&CURV id='carbon steel cp',xy= 348.15, 501.6/
&CURV id='carbon steel cp',xy= 473.15, 564.3/
&CURV id='carbon steel cp',xy= 673.15, 627.0/
&CURV id='carbon steel cp',xy= 873.15, 710.6/
&CURV id='carbon steel cp',xy= 973.15, 836.0/
&CURV id='carbon steel cp',xy=1036.15, 836.0/
&CURV id='carbon steel cp',xy=1041.15,6888.6/
&CURV id='carbon steel cp',xy=1046.15, 702.2/
&CURV id='carbon steel cp',xy=1223.15, 668.8/

```

&MTRL matname='gypsum',e=0.8,rho=211.,k=0.75,cp=1087./

A-4 Sample 4

The input file for Sample A-4 is as follows:

```
Sample 4: five compartments connected with a corridor and a t2 fire

----- General Simulation and Data Control -----
&EXEC t_max=1200.0, fileid='sample4', outtype=.FALSE., pamb=101325.0 /

&CNTR ncomp=8, njunc=10, nnotbidir=0, nfire=1, nsurf=49,
      ncompn=1, nmat=1, nfan=1, nnode=8, nduct=7, ncontrol=1,
      nsuppress=0 /

&FNAM materialfile='materials.txt' /

----- Source Fire and Fuel Load -----
&FIRE id=1, ftype=2, fsubtype=2, location=3, chi_r=0.3, t2_hrr=2500.0,
      deltah_f=14000000.0, t_start=0.0 /

----- Compartment Information -----
&COMP id=1, compname='Room 1', dx=7.0, dy=4.0, dz=3.0, vol=84.0 /
&COMP id=2, compname='Room 2', dx=5.0, dy=4.0, dz=3.0, vol=60.0 /
&COMP id=3, compname='Room 3', dx=8.0, dy=4.0, dz=3.0, vol=96.0 /
&COMP id=4, compname='Room 4', dx=10.0, dy=5.0, dz=3.0, vol=150.0 /
&COMP id=5, compname='Room 5', dx=10.0, dy=5.0, dz=3.0, vol=150.0 /

-- Corridor segments --

&COMP id=10, compname='Corridor Segment 1', dx=6.5, dy=3.0, dz=3.0, vol=58.5 /
&COMP id=11, compname='Corridor Segment 2', dx=7.0, dy=3.0, dz=3.0, vol=63.0 /
&COMP id=12, compname='Corridor Segment 3', dx=6.5, dy=3.0, dz=3.0, vol=58.5 /

-- Doors --
&JUNC id=1, location=1,10, height=2.13, elevation=0.0, area=1.94,
      bidirectional=.TRUE., vflow=.FALSE., kloss=2.04, orientation=6 /
&JUNC id=2, location=2,11, height=2.13, elevation=0.0, area=1.94,
      bidirectional=.TRUE., vflow=.FALSE., kloss=2.04, orientation=6 /
&JUNC id=3, location=3,12, height=2.13, elevation=0.0, area=1.94,
      bidirectional=.TRUE., vflow=.FALSE., kloss=2.04, orientation=6 /
&JUNC id=4, location=5,12, height=2.13, elevation=0.0, area=1.94,
      bidirectional=.TRUE., vflow=.FALSE., kloss=2.04, orientation=5 /
&JUNC id=5, location=4,11, height=2.13, elevation=0.0, area=1.94,
      bidirectional=.TRUE., vflow=.FALSE., kloss=2.04, orientation=5 /
&JUNC id=6, location=10,-1, height=2.13, elevation=0.0, area=3.88,
      bidirectional=.TRUE., vflow=.FALSE., kloss=2.04, orientation=5 /

-- openings to ambient --
&JUNC id=7, location=4,-1, height=0.8, elevation=2.0, area=1.6,
      bidirectional=.TRUE., vflow=.FALSE., kloss=2.04, orientation=6 /
&JUNC id=8, location=5,-1, height=0.8, elevation=2.0, area=1.6,
      bidirectional=.TRUE., vflow=.FALSE., kloss=2.04, orientation=6 /

-- openings between corridor segments --
&JUNC id=9, location=10,11, height=3.0, elevation=0.0, area=9.0,
      bidirectional=.TRUE., vflow=.FALSE., kloss=1.0, orientation=3 /
&JUNC id=10, location=11,12, height=3.0, elevation=0.0, area=9.0,
```

```

bidirectional=.TRUE., vflow=.FALSE., kloss=1.0, orientation=3 /

-- Room 1 --
&SURF id=1, stype=1, location=1,-1, orientation=1, area=28.0, composition=1 /
&SURF id=2, stype=1, location=1,-1, orientation=2, area=28.0, composition=1 /
&SURF id=3, stype=1, location=1,2, orientation=3, area=12.0, composition=1 /
&SURF id=4, stype=1, location=1,-1, orientation=4, area=12.0, composition=1 /
&SURF id=5, stype=1, location=1,-1, orientation=5, area=21.0, composition=1 /
&SURF id=6, stype=1, location=1,10, orientation=6, area=17.56, composition=1 /
&SURF id=7, stype=1, location=1,11, orientation=6, area=1.5, composition=1 /

-- Room 2 --
&SURF id=8, stype=1, location=2,-1, orientation=1, area=20.0, composition=1 /
&SURF id=9, stype=1, location=2,-1, orientation=2, area=20.0, composition=1 /
&SURF id=10, stype=1, location=2,3, orientation=3, area=12.0, composition=1 /
&SURF id=11, stype=1, location=2,-1, orientation=5, area=15.0, composition=1 /
&SURF id=12, stype=1, location=2,11, orientation=6, area=13.06, composition=1 /

-- Room 3 --
&SURF id=13, stype=1, location=3,-1, orientation=1, area=32.0, composition=1 /
&SURF id=14, stype=1, location=3,-1, orientation=2, area=32.0, composition=1 /
&SURF id=15, stype=1, location=3,-1, orientation=3, area=12.0, composition=1 /
&SURF id=16, stype=1, location=3,-1, orientation=5, area=24.0, composition=1 /
&SURF id=17, stype=1, location=3,11, orientation=6, area=4.5, composition=1 /
&SURF id=18, stype=1, location=3,12, orientation=6, area=17.56, composition=1 /

-- Room 4 --
&SURF id=19, stype=1, location=4,-1, orientation=1, area=50.0, composition=1 /
&SURF id=20, stype=1, location=4,-1, orientation=2, area=50.0, composition=1 /
&SURF id=21, stype=1, location=4,5, orientation=3, area=15.0, composition=1 /
&SURF id=22, stype=1, location=4,-1, orientation=4, area=15.0, composition=1 /
&SURF id=23, stype=1, location=4,10, orientation=5, area=19.5, composition=1 /
&SURF id=24, stype=1, location=4,11, orientation=5, area=8.56, composition=1 /
&SURF id=25, stype=1, location=4,-1, orientation=6, area=29.2, composition=1 /

-- Room 5 --
&SURF id=26, stype=1, location=5,-1, orientation=1, area=50.0, composition=1 /
&SURF id=27, stype=1, location=5,-1, orientation=2, area=50.0, composition=1 /
&SURF id=28, stype=1, location=5,-1, orientation=3, area=15.0, composition=1 /
&SURF id=29, stype=1, location=5,11, orientation=5, area=10.5, composition=1 /
&SURF id=30, stype=1, location=5,12, orientation=5, area=17.56, composition=1 /
&SURF id=31, stype=1, location=5,-1, orientation=6, area=29.2, composition=1 /

-- Compartment segment 1 --
&SURF id=32, stype=1, location=10,-1, orientation=1, area=19.5, composition=1 /
&SURF id=33, stype=1, location=10,-1, orientation=2, area=19.5, composition=1 /
&SURF id=34, stype=1, location=10,-1, orientation=4, area=3.54, composition=1 /

-- Compartment segment 1 --
&SURF id=35, stype=1, location=11,-1, orientation=1, area=19.5, composition=1 /
&SURF id=36, stype=1, location=11,-1, orientation=2, area=19.5, composition=1 /

-- Compartment segment 1 --
&SURF id=37, stype=1, location=12,-1, orientation=1, area=19.5, composition=1 /
&SURF id=38, stype=1, location=12,-1, orientation=2, area=19.5, composition=1 /
&SURF id=39, stype=1, location=12,-1, orientation=3, area=9.0, composition=1 /

-- Fake surfaces for Radiation heat flow --

&SURF id=50, stype=1, location=1,10, orientation=6, area=1.94, fake=.TRUE. /
&SURF id=51, stype=1, location=2,11, orientation=6, area=1.94, fake=.TRUE. /

```

```

&SURF id=52, stype=1, location=3,12, orientation=6, area=1.94, fake=.TRUE. /
&SURF id=53, stype=1, location=4,11, orientation=5, area=1.94, fake=.TRUE. /
&SURF id=54, stype=1, location=5,12, orientation=5, area=1.94, fake=.TRUE. /
&SURF id=55, stype=1, location=10,-1, orientation=4, area=3.88, fake=.TRUE. /
&SURF id=56, stype=1, location=4,-1, orientation=6, area=0.8, fake=.TRUE. /
&SURF id=57, stype=1, location=5,-1, orientation=6, area=0.8, fake=.TRUE. /
&SURF id=58, stype=1, location=10,11, orientation=3, area=9.0, fake=.TRUE. /
&SURF id=59, stype=1, location=11,12, orientation=3, area=9.0, fake=.TRUE. /

```

```

&CMPN id=1, cmpnname='Gypsum Boundary', fileread=.FALSE.,
      layers=1, materials=2, dx=0.0158 /
&MTRL id=2, matname='gypsum' /

```

----- Mechanical Ventilation -----

-- nodes --

```

&RNOD id=1, nducts=2, ducts=1,1, terminal=.TRUE., elevation=2.5,
      loss=0.0,0.0 /
&RNOD id=2, nducts=2, ducts=1,4, terminal=.FALSE., elevation=2.5,
      loss=1.1,1.1 /
&RNOD id=3, nducts=2, ducts=2,2, terminal=.TRUE., elevation=2.5,
      loss=0.0,0.0 /
&RNOD id=4, nducts=3, ducts=2,4,5, terminal=.FALSE., elevation=2.5,
      loss=1.63,1.63,1.63,0.75,1.63,0.75 /
&RNOD id=5, nducts=2, ducts=3,3, terminal=.TRUE., elevation=2.5,
      loss=0.0,0.0 /
&RNOD id=6, nducts=2, ducts=3,6, terminal=.FALSE., elevation=2.5,
      loss=1.1,1.1 /
&RNOD id=7, nducts=3, ducts=5,6,7, terminal=.FALSE., elevation=2.5,
      loss=0.75,1.63,0.75,1.63,1.63 /
&RNOD id=8, nducts=2, ducts=-1,7, terminal=.TRUE., elevation=2.5,
      loss=0.0,0.0 /

```

-- ducts --

```

&RDCT id=1, node=1,2, area=0.3721, perimeter=2.44, length=2.0,
      roughness=0.0005 /
&RDCT id=2, node=3,4, area=0.3721, perimeter=2.44, length=2.0,
      roughness=0.0005 /
&RDCT id=3, node=5,6, area=0.3721, perimeter=2.44, length=2.0,
      roughness=0.0005 /
&RDCT id=4, node=2,4, area=0.3721, perimeter=2.44, length=6.0,
      roughness=0.0005 /
&RDCT id=5, node=4,7, area=0.3721, perimeter=2.44, length=3.0,
      roughness=0.0005 /
&RDCT id=6, node=6,7, area=0.3721, perimeter=2.44, length=2.0,
      roughness=0.0005 /
&RDCT id=7, node=7,8, area=0.3721, perimeter=2.44, length=6.0,
      roughness=0.0005, fan=1 /

```

-- fan --

```

&RFAN id=1, fileread=.FALSE., ftype=2, full=1.83, shutoff=180,
      reverse=.FALSE., control=1 /

```

```

&CTRL id=1, ctype=1, trip=.FALSE., initial=.TRUE., inputvar=1,
      setpoint=10000.00 /

```

----- Suppression and Detection -----

-- None --

The following gypsum material property line is contained in the `materials.txt` file:

```
&MTRL matname='gypsum',e=0.8,rho=211.,k=0.75,cp=1087./
```