HEAT CONDUCTION TRANSFER FUNCTIONS FOR MULTI-LAYER STRUCTURES

by

TERRY DEL HUBBS

B.S., Kansas State University, 1975

A MASTER'S THESIS

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Mechanical Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1977

Approved by:

J. Garth Thompson

Major Professor

LD
2668
T4
1977
H82
C.2
Document

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

$\theta(x,t)$	slab temperature (°F)
$\theta(x,0) = \theta_{Go}$	initial slab temperature (°F)
f	heat flux (BTU/hr)
v	temperature deviation from initial temperature (°F)
x	local position with respect to x coordinate (ft)
t	time (hr)
α_r	thermal diffusivity of $r^{\underline{th}}$ layer (ft^2/hr)
k_r	thermal conductivity of $r^{\underline{th}}$ layer (BTU/hr-ft-°F)
ρ_r	density of $r^{\underline{th}}$ layer (lb_m/ft^3)
c_r	specific heat of $r^{\underline{th}}$ layer (BTU/lb_m-°F)
Δx_r	thickness of $r^{\underline{th}}$ layer (ft)
R_1	inner thermal surface resistance (hr-°F/BTU)
R_n	outer thermal surface resistance (hr-°F/BTU)
s	the Laplace variable
T	Laplace transform of temperature, θ
F	Laplace transform of heat flux, f
V	Laplace transform of v
β_m	roots of the function B=0
Δ	sampling period of the z-transforms (hr)
R(z)	z-transfer function of Laplace transfer function, -A/B
W(z)	z-transfer function of Laplace transfer function, 1/B

CHAPTER I

INTRODUCTION

Problem

In recent years, the words "energy shortage" have become commonplace. Some argue that the energy crunch is an imaginary crisis being used by energy producers to increase profits, while others insist that the energy shortage is a real and grave situation of severe consequence being used by energy producers to increase profits. All agree that energy costs have increased astronomically.

In the area of building climate control, economic factors have made some traditional engineering design techniques impracticable, improved the feasibility of other concepts, and coincidentally inspired new and innovative design tools. Much work has been done in the simulation of building air conditioning systems. The term air conditioning is not limited to cooling, but implies the entire process of conditioning air. Some primary concerns have been predicting heating and cooling loads accurately so that 1) heating, cooling, and air handling equipment may be sized properly, 2) various building configurations can be easily compared, and 3) changes made in building materials may be evaluated. In most cases, the simulation consists of a large digital computer program made up of several complicated subroutines. Often, a subroutine is written to handle a unique part of the physical problem, for example, one subroutine may find heat loss due to conduction heat transfer through walls, another subroutine may handle solar radiation and shading, while yet another simulates systems and equipment. These programs are most often used with a one hour stepsize and depend upon

U.S. Weather Bureau tapes for hourly environmental data, as well as user supplied data such as building dimensions, materials, etc. With a one hour stepsize, the availability of data is good, heating and cooling loads can be obtained to sufficiently satisfy many problems, and the computing time required is reasonable. An hourly stepsize also takes full advantage of the conduction transfer functions strong points, namely speed and accuracy. The output is often an hour-by-hour profile of building heat gain or loss. In most cases, to obtain hour-by-hour heat loss the inside temperature must be assumed to be constant. The hourly stepsize itself is sometimes limiting. For example, an hourly step may bypass some effects of a fast dynamic control system. Also, large inaccuracy could result due to input that deviates appreciably during an hour, and is then approximated with an hourly increment. These and similar restrictions limit the flexibility of these simulations. Such programs are written for and conform to studies of physical building characteristics reasonably well but are not equally well suited to studies of the buildings automatic climate controls. Control installations and changes are of paramount economic importance as such actions give quick payback [1].

In order to investigate the effect of control system dynamics on energy consumption, a simulation with a stepsize short enough to manifest the dynamics of the control system is required. The problem approached in this research is the investigation of the effect on the conduction transfer functions and associated simulation algorithm of shortening the stepsize.

Purpose

The purpose of this thesis is to demonstrate and satisfactorily document an algorithm for transient heat conduction through multi-layer walls that may be used as an element in a computer simulation to study effects of variations in the automatic temperature control systems of buildings. The algorithm should 1) be general enough to handle walls of any number of layers, 2) have a variable time increment which is short enough to be used with fast dynamic elements, 3) have no restrictive input, such as constant or periodic boundary conditions, 4) be reasonably accurate, 5) require as little computing time as possible, and 6) have few restricting assumptions.

This thesis will document a study of the thermal response factor technique applied to this problem with these criteria.

The development of the thermal response factor concept cannot be attributed to a single individual. Mitalas and Stephenson [2] suggest Nessi and Nisolle [3] as early investigators and cite Mackey and Wright [4] as having worked on the same problem with restricted boundary conditions. T. Kusuda [5] has also contributed to the method and attributes significant contributions to Mitalas and Arseneault [6].

Analytical solutions to one-dimensional unsteady-state systems with specified boundary conditions occupy substantial space in heat-flow and applied mathematics texts. Since the boundary conditions in practical applications may change in a manner unique to the particular system and application, even the most extensive compilation of formal solutions is unlikely to match a given practical case completely. However, with the use of a digital computer, differential difference methods may prove to be powerful tools [7]. Finite difference approximations are a well-known approach to solving transient heat transfer problems. Although the computational procedures involved are less complicated than with the response factor concept, small grid sizes are required for finite difference time and space coordinates if computational stability is to be retained for a multi-layer heat flow problem [5]. The response factor method does not have these limitations.

The basic steps in the development of the response factor algorithm are

1) Obtain the differential equation to model heat conduction in a slab.

2) Derive expressions for temperature and heat flux by applying Laplace transforms.

3) Generalize the analysis to "n" layers.

4) Assume an input form and apply z-transforms to the Laplace transfer functions.

5) Manipulate the z-transfer functions into the form of infinite power series in z^{-1} and obtain the coefficients of the z terms.

6) Find the solution in time by expanding the z-polynomials.

The derivation of the algorithm follows the basic outline as listed above and is found in Chapter II. Chapter III explains the computer programs used to develop and apply the response factors, while Chapter IV contains examples where the response factors are used to obtain solutions to practical problems.

CHAPTER II

DERIVATIONS

Single Layer Structure

Consider a semi-infinite slab of a homogeneous material with a uniform initial temperature, $\theta(x,0)$. Conduction heat transfer through the slab is described by the one-dimensional heat-conduction equation,

$$\frac{\partial^2\theta}{\partial x^2} = \frac{1}{\alpha}\frac{\partial\theta}{\partial t} \quad \text{where} \quad \alpha = \frac{k}{\rho c} \; . \quad [8] \qquad (1)$$

$\theta(x,t)$ - temperature
x - local position with respect to x coordinate
t - time
α - thermal diffusivity
k - thermal conductivity
ρ - density
c - specific heat

Below is a schematic cross-section of the layer, its initial condition, and the heat flux, f, at the left surface of the slab.

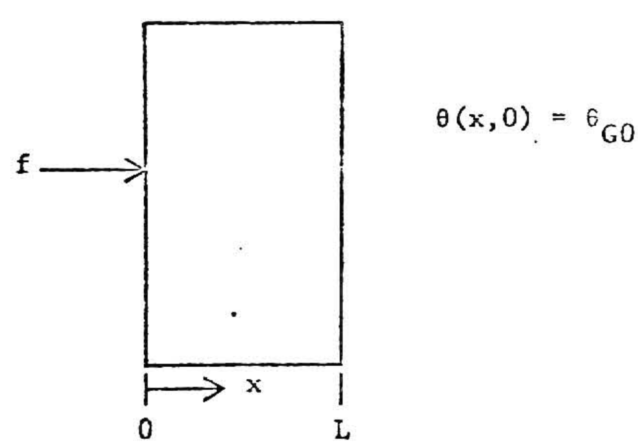

Figure 1. Single-Layer Slab.

Below, the Laplace transform of the one-dimensional heat conduction equation is obtained [9].

$$\mathcal{L}[\frac{\partial^2\theta}{\partial x^2}] = \mathcal{L}[\frac{1}{\alpha}\frac{\partial\theta}{\partial t}]$$

$$\frac{d^2T}{dx^2} = \frac{s}{\alpha}\,[T - \frac{\theta_{Go}}{s}] \qquad (2)$$

T - Laplace transform of temperature, θ

s - the Laplace variable.

Equation (2) may be simplified by expressing the temperature as a deviation from the initial, uniform temperature.

$$v(x,t) = \theta(x,t) - \theta(x,0)\ .$$

Applying the Laplace transform to this expression yields

$$V = T - \frac{\theta_{Go}}{s}\ . \qquad (3)$$

V - the Laplace transform of v.

Differentiating (3) twice with respect to x results in

$$\frac{d^2V}{dx^2} = \frac{d^2T}{dx^2}\ . \qquad (4)$$

Substituting (3) and (4) into (2) yields

$$\frac{d^2V}{dx^2} = \frac{s}{\alpha}\,V\ . \qquad (5)$$

The significance of these operations is that equation (5) is not a partial differential equation like equation (1), but rather an ordinary differential equation where s is an algebraic variable.

The general solution to equation (5) is [10]

$$V_{(x,s)} = \frac{(V_0)'}{\sqrt{s/\alpha}}\,\sinh(\sqrt{s/\alpha}\;x) + V_0\,\cosh(\sqrt{s/\alpha}\;x) \qquad \text{where} \qquad (6)$$

$$V_0 = \mathcal{L}[v(0,t)] = \mathcal{L}[\theta(0,t) - \theta_{Go}]\ .$$

The heat flux, f, at any x (see Figure 1) is

$$f = -k \frac{d\theta}{dx} . \quad [8]$$

Taking the Laplace transform yields

$$F = -k \frac{dT}{dx} \qquad \text{where} \tag{7}$$

F - Laplace transform of heat flux, f(x,t).

Differentiating equation (3) with respect to x yields

$$\frac{dT}{dx} = \frac{dV}{dx} ,$$

thus equation (7) may be written

$$F = -k \frac{dV}{dx} . \tag{8}$$

Equation (8) relates the heat flux at any point x in the slab to the gradient of the temperature at that point.

At x=0, it follows that

$$\frac{dV_0}{dx} = (V_0)' = -\frac{F_0}{k} ,$$

which may be substituted into the general solution, equation (6),

$$V(x) = -F_0 \frac{1}{k} \sqrt{\alpha/s} \sinh(\sqrt{s/\alpha}\ x) + V_0 \cosh(\sqrt{s/\alpha}\ x) . \tag{9}$$

The substitution eliminated the derivative in the solution, and both coefficients, F_0 and V_0, are for x=0.

To obtain an expression for F, equation (9) is differentiated with respect to x,

$$\frac{dV}{dx} = -F_0 \frac{1}{k} \cosh(\sqrt{s/\alpha}\ x) + V_0 \sqrt{s/\alpha} \sinh(\sqrt{s/\alpha}\ x) .$$

Substituting into equation (8) yields

$$F = F_0 \cosh(\sqrt{s/\alpha}\, x) - V_0\, k\sqrt{s/\alpha}\, \sinh(\sqrt{s/\alpha}\, x)\ . \qquad (10)$$

Equations (9) and (10) are now placed in matrix form.

$$\begin{bmatrix} V \\ \\ F \end{bmatrix} = \begin{bmatrix} \cosh(\sqrt{s/\alpha}\, x) & -\frac{1}{k}\sqrt{\alpha/x}\, \sinh(\sqrt{s/\alpha}\, x) \\ \\ -k\sqrt{s/\alpha}\, \sinh(\sqrt{s/\alpha}\, x) & \cosh(\sqrt{s/\alpha}\, x) \end{bmatrix} \begin{bmatrix} V_0 \\ \\ F_0 \end{bmatrix}$$

The expressions in this matrix relate the temperature and heat flux at any x in the layer to the temperature and heat flux at x=0. Below is a matrix for the temperature and flux at x=L.

$$\begin{bmatrix} V_L \\ \\ F_L \end{bmatrix} = \begin{bmatrix} \cosh(\sqrt{s/\alpha}\, L) & -\frac{1}{k}\sqrt{\alpha/s}\, \sinh(\sqrt{s/\alpha}\, L) \\ \\ -k\sqrt{s/\alpha}\, \sinh(\sqrt{s/\alpha}\, L) & \cosh(\sqrt{s/\alpha}\, L) \end{bmatrix} \begin{bmatrix} V_0 \\ \\ F_0 \end{bmatrix} \qquad (11)$$

Note that this matrix relates the temperature and flux at one face to the temperature and flux at the other face. Also note that L is the thickness of the layer.

Multi-Layer Structures

Following the procedure of Carslaw and Jaeger (11), a multi-layer wall composed of n slabs as in Figure 2 is considered.

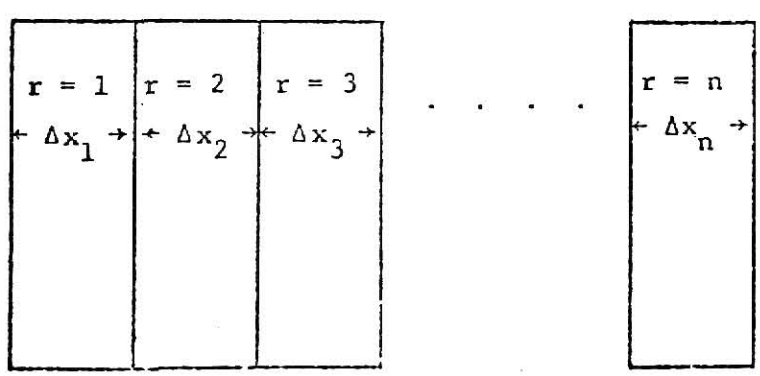

Figure 2. Multi-Layer Slab.

By generalizing matrix (11), the following matrix may be written for use with any particular layer in Figure 2. Recall that the substitution $v(x,t) = \theta(x,t) - \theta_{Go}$ was made in the derivation of the expressions in matrix (11) to oblige the initial condition $\theta(x,0) = \theta_{Go}$. Therefore each layer must be at the initial temperature θ_{Go}.

$$\begin{bmatrix} V_{r+1} \\ F_{r+1} \end{bmatrix} = \begin{bmatrix} \cosh(\sqrt{s/\alpha_r}\,\Delta x_r) & -\frac{1}{k_r}\sqrt{\alpha_r/s}\,\sinh(\sqrt{s/\alpha_r}\,\Delta x_r) \\ -k_r\sqrt{s/\alpha_r}\,\sinh(\sqrt{s/\alpha_r}\,\Delta x_r) & \cosh(\sqrt{s/\alpha_r}\,\Delta x_r) \end{bmatrix} \begin{bmatrix} V_r \\ F_r \end{bmatrix}$$

For brevity, the following substitutions are made.

$$A_r = \cosh(\sqrt{s/\alpha_r}\,\Delta x_r)$$

$$B_r = -\frac{1}{k_r}\sqrt{\alpha_r/s}\,\sinh(\sqrt{s/\alpha_r}\,\Delta x_r)$$

$$C_r = -k_r\sqrt{s/\alpha_r}\,\sinh(\sqrt{s/\alpha_r}\,\Delta x_r)$$

$$D_r = \cosh(\sqrt{s/\alpha_r}\,\Delta x_r)$$

For layer one, r=1, and

$$\begin{bmatrix} V_2 \\ F_2 \end{bmatrix} = \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} V_1 \\ F_1 \end{bmatrix} .$$

Similarly for layer two, r=2, and

$$\begin{bmatrix} V_3 \\ F_3 \end{bmatrix} = \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} \begin{bmatrix} V_2 \\ F_2 \end{bmatrix} .$$

If there is perfect thermal contact between the slabs, simple substitution yields

$$\begin{bmatrix} V_3 \\ F_3 \end{bmatrix} = \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} V_1 \\ F_1 \end{bmatrix} .$$

In like manner, the expressions relating inside and outside face temperature and heat flux for an n layer structure in matrix form are

$$\begin{bmatrix} V_{n+1} \\ F_{n+1} \end{bmatrix} = \begin{bmatrix} A_n & B_n \\ C_n & D_n \end{bmatrix} \begin{bmatrix} A_{n-1} & B_{n-1} \\ C_{n-1} & D_{n-1} \end{bmatrix} \cdots \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} V_1 \\ F_1 \end{bmatrix} . \qquad (12)$$

Surface Resistances

A linear thermal resistance, R, may be included at the surface of a layer as shown in Figure 3. Modeling the surface film in this way neglects any thermal capacitance of the film but accounts for the resistance to the flow of heat through it. Physically, the capacitance of the film is minute and may be neglected. The matrix notation relating the temperature and flux at the resistance to the temperature and heat flux at the slab surface is

$$\begin{bmatrix} V_1 \\ F_1 \end{bmatrix} = \begin{bmatrix} 1 & -R \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V_I \\ F_I \end{bmatrix} . \qquad [11]$$

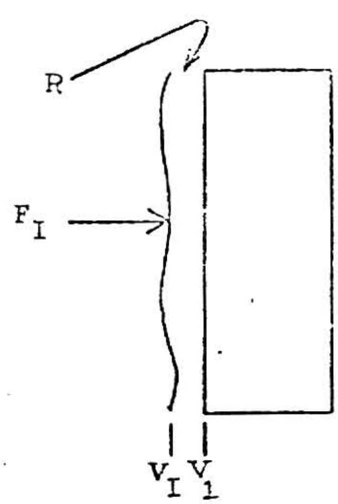

Figure 3. Surface Resistance.

Thermal resistances may be considered at both outer surfaces of the multi-layer slab as shown in Figure 4.

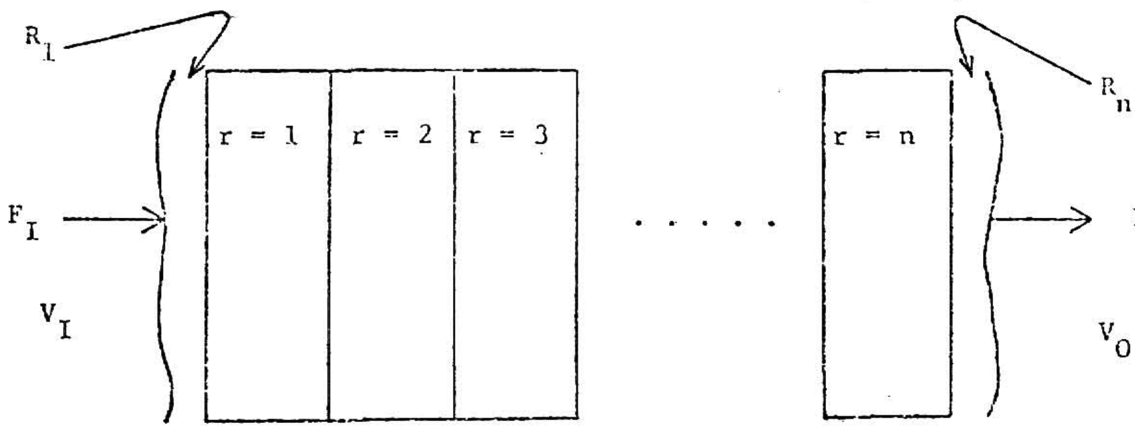

Figure 4. Multi-Layer Slab and Resistances.

The expressions relating inside and outside temperature and heat flux are

$$\begin{bmatrix} V_O \\ F_O \end{bmatrix} = \begin{bmatrix} 1 & -R_n \\ 0 & 1 \end{bmatrix} \begin{bmatrix} A_n & B_n \\ C_n & D_n \end{bmatrix} \begin{bmatrix} A_{n-1} & B_{n-1} \\ C_{n-1} & D_{n-1} \end{bmatrix} \cdots \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} 1 & -R_1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V_I \\ F_I \end{bmatrix} . \tag{13}$$

The product of two 2×2 matrices is a 2×2 matrix. Therefore, the results of multiplying the 2×2 matrices together will be a 2×2. The resultant 2×2 matrix will be symbolized as

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} . \tag{14}$$

Expression (13) is now simplified to

$$\begin{bmatrix} V_O \\ F_o \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} V_I \\ F_I \end{bmatrix} . \tag{15}$$

Expression (15) may be manipulated so the heat fluxes are both to the left of the equals sign. By matrix multiplication of expression (15), the following equations are obtained:

$$V_0 = AV_I + BF_I \tag{16}$$

$$F_0 = CV_I + DF_I \tag{17}$$

From equation (16)

$$F_I = -\frac{A}{B} V_I + \frac{1}{B} V_0 \tag{18}$$

By substituting this equation into equation (17), an expression for the Laplace transform of the heat flux at R_1 as a function of the inside and outside temperatures is obtainable.

$$F_0 = \frac{(BC - AD)}{B} V_I + \frac{D}{B} V_0 \tag{19}$$

This equation may be simplified by recognizing that the term (BC - AD) is the determinant of matrix (14). The determinant may be evaluated as follows.

$$\Gamma = \begin{vmatrix} A & B \\ C & D \end{vmatrix} = \Gamma_{Rn} \cdot \Gamma_n \cdots \Gamma_2 \cdot \Gamma_1 \cdot \Gamma_{R_1} = \begin{vmatrix} 1 & -R_n \\ 0 & 1 \end{vmatrix} \begin{vmatrix} A_n & B_n \\ C_n & D_n \end{vmatrix}$$

$$\cdots \begin{vmatrix} A_2 & B_2 \\ C_2 & D_2 \end{vmatrix} \begin{vmatrix} A_1 & B_1 \\ C_1 & D_1 \end{vmatrix} \begin{vmatrix} 1 & -R_1 \\ 0 & 1 \end{vmatrix}$$

It is easily found that $\Gamma_{R_n} = \Gamma_{R_1} = 1$.

Now consider the determinant of the matrix for layer r.

$$\Gamma_r = \begin{vmatrix} \cosh\sqrt{s/\alpha_r}\ \Delta x_r) & -\frac{1}{k_r}\sqrt{\alpha_r/s}\ \sinh(\sqrt{s/\alpha_r}\ \Delta s_r) \\ -k_r\sqrt{s/\alpha_r}\ \sinh(\sqrt{s/\alpha_r}\ \Delta x_r) & \cosh(\sqrt{s/\alpha_r}\ \Delta x_r \end{vmatrix}$$

$$= \cosh^2(\sqrt{s/\alpha_r}\ \Delta x_r) - \sinh^2(\sqrt{s/\alpha_r}\ \Delta x_r) = 1$$

It may be seen that the determinant of matrix (14) is equal to 1.

Equation (19) now simplifies to

$$F_O = \frac{1}{B} V_I + \frac{D}{B} V_O \; . \tag{20}$$

Equations (18) and (20) are now placed in matrix form.

$$\begin{bmatrix} F_I \\ F_O \end{bmatrix} = \begin{bmatrix} -\frac{A}{B} & \frac{1}{B} \\ \frac{1}{B} & \frac{D}{B} \end{bmatrix} \begin{bmatrix} V_I \\ V_O \end{bmatrix} \tag{21}$$

The above matrices relate the inside and outside temperatures to the inside and outside heat fluxes. The terms in the 2×2 matrix are the transfer functions in the Laplace domain.

z- Transform Method

For this procedure to be of practical value, the inverse Laplace transforms must be found. Inverse Laplace transforms may be found in a variety of ways. For example direct integration may be used, tables of Laplace transforms are sometimes helpful, the partial fraction expansion method may be applied, or z-transforms can be employed. However, the complexity of the Laplace transforms, in this case, nearly eliminates direct integration or the use of tables. The z-transform technique as outlined in reference [2] will be used in this work. With this method, an input form is selected and multiplied by the transfer function to obtain the Laplace transform of the output. The z-transform of the output, the input times the transfer function, is then found. The z-transfer function is then obtained by dividing the z-transform of the output by the z-transform of the input used to obtain the Laplace transform of the output. What follows is a detailed development of the z-transform method.

Recall that the terms in the 2×2 matrix of expression (21), A, B, and D, are all functions of the Laplace variable, s. It may be helpful to return to expression (13) and page 8 to again see how the terms are developed. One finds that the terms are composed of sums of products of sinh and cosh terms. So, the upper left Laplace transfer function, $-\frac{A}{B}$, consists of a string of cosh and sinh terms divided by some other array of cosh and sinh terms. The size of the string of terms is determined by "n", the number of layers in the structure. The hyperbolic functions have an infinite set of roots, but are approximated with a finite number of roots.

The upper equation from expression (21) is

$$F_I = -\frac{A}{B} V_I + \frac{1}{B} V_O \; . \tag{22}$$

Here, the Laplace transform of the output, F_I, is the sum of a transfer function, $-\frac{A}{B}$, times an input, V_I, and another transfer function, $\frac{1}{B}$, times another input, V_O. The z-transform of the output heat flux will be composed of the sum of the two responses due to each input. A detailed derivation of the z-transfer function for the $-\frac{A}{B}$ transfer function is included in this thesis. The other z-transfer functions are easily developed following the same pattern.

To obtain a z-transfer function, an input form must be chosen. The selection of the input form is equivalent to designating a function for interpolating between the discrete values given by the z-transform coefficients [2]. A step function is synonymous to representing the input by a staircase approximation, while a ramp input is equivalent to linear interpolation between the discrete values. Inputs of higher order than the ramp input yield unstable transfer functions because the z-transfer functions have poles at the zeroes of the z-transform of the input. A step input form is satisfactory for small time increments and slowly changing inputs, while a ramp input is preferable for larger time increments or rapidly deviating inputs. Ramp inputs are used in the following developments.

The Laplace transform of the output, O, may be found with the following equation.

$$O = -\frac{A}{B} I$$

O - Laplace transform of the output

$-\frac{A}{B}$ - Laplace transfer function

I - Laplace transform of the input

In this case, the input is a ramp function, and

$$I = \frac{1}{s^2} \quad . \qquad [12]$$

Therefore,

$$O = -\frac{A}{s^2 B} \; .$$

This output has a double pole at s=0, and other poles at the roots of B. See Appendix A for a discussion of the roots of the expression B.

The partial fraction expansion of O is

$$O = \frac{p_2}{s^2} + \frac{p_1}{s} + \frac{q_1}{s + \beta_1} + \frac{q_2}{s + \beta_2} + \ldots + \frac{q_n}{s + \beta_n} + \ldots \qquad (23)$$

where the β's are the roots of the expression B, and p_2, p_1, and the q's are defined below.

$$p_2 = [-\frac{A}{B}]\Big|_{s=0} \qquad p_1 = -[\frac{B\frac{dA}{ds} - A\frac{dB}{ds}}{B^2}]\Big|_{s=0} \qquad q_m = [-\frac{A}{s^2 B}(s+\beta_m)]\Big|_{s=-\beta_m}$$

The q's are the residues of O. In the present form, the evaluation of a q term would result in an indeterminant solution. However, L'Hospital's Rule may be used to find an expression which yields determinant results. This expression is shown below.

$$q_m = [-\frac{A}{s^2 \frac{dB}{ds}}]\Big|_{s=-\beta_m}$$

Realize that p_2, p_1, and the q's are all constants that have been evaluated at specific values of s, namely 0 and $-\beta_m$. Thus their presence will not affect the form of the z-transform of the function in which they are located. If the z-transform of the partial fraction expansion of O, Equation (21), is then found, the result is O(z).

$$O(z) = \frac{p_2 \Delta}{z(1 - z^{-1})^2} + \frac{p_1}{(1-z^{-1})} + \sum_{m=1}^{\infty} \left[\frac{q_m}{(1 - e^{-\beta_m \Delta} z^{-1})}\right]$$

Δ - sampling period of the z-transforms.

When a common denominator is found,

$$O(z) = \left(\frac{1}{z(1-z^{-1})^2 \prod_{m=1}^{\infty} (1 - e^{-\beta_m \Delta} z^{-1})}\right) [p_2\Delta \prod_{m=1}^{\infty} (1 - e^{-\beta_m \Delta} z^{-1})$$

$$+ p_1 z(1-z^{-1}) \prod_{m=1}^{\infty} (1 - e^{-\beta_m \Delta} z^{-1})$$

$$+ z(1-z^{-1})^2 \prod_{m=1}^{\infty} (1 - e^{-\beta_m \Delta} z^{-1}) \cdot [\sum_{m=1}^{\infty} (\frac{q_m}{1 - e^{-\beta_m \Delta} z^{-1}})] .$$

From this expression for the output, it is possible to find a z-transfer function, R(z), by dividing by the z-transform of the input. Recall that the input was a ramp function. The z-transform of a ramp function is

$$I(z) = \frac{\Delta}{z(1 - z^{-1})^2} .$$

Now, R(z), the z-transfer function of the Laplace transfer function $-\frac{A}{B}$ is found to be

$$R(z) = \frac{O(z)}{I(z)} = \left(\frac{1}{\prod_{m=1}^{\infty} (1 - e^{-\beta_m \Delta} z^{-1})} \right) [p_2 \prod_{m=1}^{\infty} (1 - e^{-\beta_m \Delta} z^{-1})$$

$$+ \frac{p_1}{\Delta} z(1-z^{-1}) \prod_{m=1}^{\infty} (1 - e^{-\beta_m \Delta} z^{-1})$$

$$+ \frac{z(1-z^{-1})^2}{\Delta} \prod_{m=1}^{\infty} \cdot (1 - e^{-\beta_m \Delta} z^{-1}) [\sum_{m=1}^{\infty} (\frac{q_m}{1 - e^{-\beta_m \Delta} z^{-1}})] \quad . \tag{24}$$

This z-transfer function, or pulse transfer function may be used to relate the pulsed output of the wall to its pulsed input.

In summary, an input form has been selected and the Laplace transform of the output was found by multiplying that input by a Laplace transfer function that reflects the thermal characteristics of a multi-layer wall. The z-transform of the output was then found. This output was divided by the z-transform of the input to yield a z-transfer function for the wall.

The z-transfer function was not obtained by taking the z-transform of the Laplace transfer function. If the z-transfer function were obtained in that way, and then multiplied by the z-transform of the input, the sampled output obtained would be the response due to a "sampled" input rather than a continuous input. That is

$$IG(z) \neq I(z)\ G(z)$$

where the G term is the transfer function. See reference (12), page 644.

The transfer function, $R(z)$, may be put in the form of a ratio of two infinite power series in z^{-1}. In practice however, a finite set of roots of the function B is employed resulting in series of finite numbers of terms.

$$R(z) = \frac{a_1 + a_2 z^{-1} + a_3 z^{-1} + \ldots + a_{n+1}\ z^{-n}}{b_1 + b_2 z^{-1} + b_3 z^{-2} + \ldots + b_{n+1}\ z^{-n}}$$

a - numerator coefficients

b - denominator coefficients

n - number of roots of function B

The coefficients may be evaluated by expanding the numerator and denominator of equation (24). Calculation of these coefficients is discussed in Chapter III.

R(z), the z-transfer function for the Laplace transfer function - $\frac{A}{B}$ has now been derived. By following the same steps, another z-transfer function, which is called W(z), may be developed for the Laplace transfer function $\frac{1}{B}$. (See expression (21)). When the partial fraction expansion of the $\frac{1}{B}$ transfer function is found, the following constants are necessary.

$$u_2 = [\frac{1}{B}]\Big|_{s=0} \quad , \quad u_1 - [\frac{\frac{dB}{ds}}{B^2}]\Big|_{s=0} \quad , \quad v_m = [\frac{1}{s^2 \frac{dB}{ds}}]\Big|_{s=-\beta_m}$$

As before, the Laplace transfer function may be maneuvered into a z-transfer function in the following form.

$$W(z) = \frac{c_1 + c_2 z^{-1} + c_3 z^{-2} + \ldots + c_{n+1}\ z^{-n}}{d_1 + d_2 a^{-1} + d_3 z^{-2} + \ldots + d_{n+1}\ z^{-n}}$$

c - numerator coefficients

d - denominator coefficients

n - number of roots of function B

When the two z-transfer functions have been derived, it is found that they have the same denominator. Then

$$b_1 + b_2 z^{-1} + b_3 z^{-2} + \ldots + b_{n+1} z^{-n} = d_1 + d_2 z^{-1} + d_3 z^{-2} + \ldots + d_{n+1} z^{-n} .$$

Inversion of z-Transforms

For the moment, consider an input of a train of weighted impulses as in Figure 5.

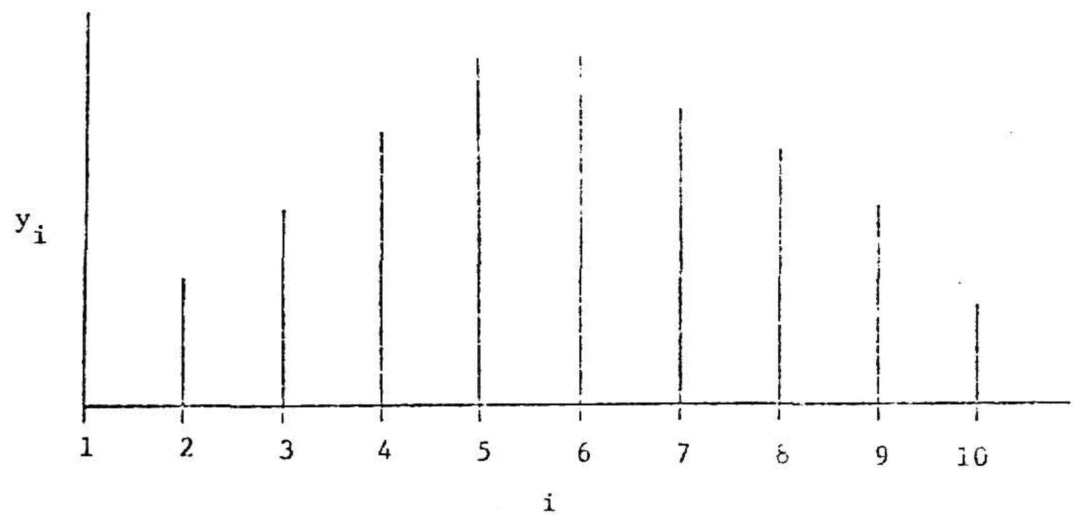

Figure 5. Weighted Impulses.

The impulses are equally spaced and each impulse is weighted by a constant, y_i. I(z), the z-transform of this input function is

$$I(z) = \sum_{i=1}^{\infty} y_i z^{-i+1} = y_1 + y_2 z^{-1} + y_3 z^{-2} + \ldots .$$

Similarly, the z-transform of an output function of the same form may be expressed as

$$O(z) = O_1 + O_2 z^{-1} + O_3 z^{-2} + \ldots .$$

It is known that

$$O(z) = G(z) I(z) . \quad [12]$$

O(z) - z-transform of the output

G(z) - z-transfer function

I(z) - z-transform of the input

And for the transfer function R(z) it follows that

$$O_1 + O_2 z^{-1} + O_3 z^{-2} + \ldots = \left(\frac{a_1 + a_2 z^{-1} + a_3 z^{-2} + \ldots + a_{n+1} z^{-n}}{b_1 + b_2 z^{-1} + b_3 z^{-2} + \ldots + b_{n+1} z^{-n}}\right)$$

$$\cdot \, (y_1 + y_2 z^{-1} + y_3 z^{-2} + \ldots) \quad .$$

By clearing the denominator and multiplying, equation (25) may be derived.

$$O_1 b_1 + (O_2 b_1 + O_1 b_2) z^{-1} + (O_3 b_1 + O_2 b_2 + O_1 b_3) z^{-2} + (O_4 b_1 + O_3 b_2 +$$
$$+ O_2 b_3 + O_1 b_4) z^{-3} + \ldots = y_1 a_1 + (y_2 a_1 + y_1 a_2) z^{-1} + (y_3 a_1 +$$
$$+ y_2 a_2 + y_1 a_3) z^{-2} + (y_4 a_1 + y_3 a_2 + y_2 a_3 + y_1 a_4) z^{-3} + \ldots \quad . \tag{25}$$

Both sides of equation (25) are polynomials, so the coefficients of the various powers of z^{-1} on each side of the equation must be equivalent. Furthermore, each side of the equation is convergent and the coefficient of the z^{-k} term in each series is the value of the function at time $k\Delta$. Δ is the sampling period of the z-transforms. A set of equalities may be formulated from equation (25) and are listed below in Table 1.

Table 1. Polynomial Coefficients Expressed as Equalities.

$k\Delta$	Equalities
0	$O_1 b_1 = y_1 a_1$
1Δ	$O_2 b_1 + O_1 b_2 = y_2 a_1 + y_1 a_2$
2Δ	$O_3 b_1 + O_2 b_2 + O_1 b_3 = y_3 a_1 + y_2 a_2 + y_1 a_3$
3Δ	$O_4 b_1 + O_3 b_2 + O_2 b_3 + O_1 b_4 = y_4 a_1 + y_3 a_2 + y_2 a_3 + y_1 a_4$

If the output at kΔ, O_{k+1}, is unknown but the input and transfer function are known, the output may be found by solving the equations. Values of the output are used in subsequent equations.

This technique may be extended to include the transfer function W(z) and invert the z-transformation of equation (22) to obtain values of the heat flux at specific points in time. As shown above, coefficients of like powers of an expanded polynomial can be equated yielding expressions that may be used to find the heat flux at t = kΔ, f_{k+1}, when the input temperatures and transfer functions are known. A few of these expressions are contained in Table 2.

Table 2. Equalities for Calculation of Heat Flux.

kΔ	Equalities
0	$f_1b_1 = U_1a_1 + V_1c_1$
1Δ	$f_2b_1 + f_1b_2 = U_2a_1 + U_1a_2 + V_2c_1 + V_1c_2$
2Δ	$f_3b_1 + f_2b_2 + f_1b_3 = U_3a_1 + U_2a_2 + U_1a_3 + V_3c_1 + V_2c_2 + V_1c_3$
3Δ	$f_4b_1 + f_3b_2 + f_2b_3 + f_1b_4 = U_4a_1 + U_3a_2 + U_2a_3 + U_1a_4 + V_4c_1 + V_3c_2 + V_2c_3 + V_1c_4$
⋮	⋮

f - heat flux at film surface, f_I (see Figure 4).

U - previously defined by V_I

V - previously defined by V_O

U_{k+1} and V_{k+1} are discrete values of the inside and outside temperatures at t = kΔ, and likewise the output, f_{k+1}, is discrete values of the heat flux at t = kΔ. The z-transfer functions, however, were developed with a ramp input. A discussion of this may be located in Appendix B.

CHAPTER III

CALCULATION OF THE z-TRANSFER FUNCTIONS

To obtain the coefficients of the z-transfer functions, a variety of numerical constants must be found. Except for a few, the constants are evaluated from large, complicated functions. Evaluating the constants by hand would be impractical. A digital computer is a necessary tool for the application of the concept. In most cases, the computer programming may be done in a straight-forward, brute force manner. However, a few scraps of ingenuity greatly improve some of the programs' flexibility and comprehensiveness. The computer programs in this chapter are all done in double precision for increased accuracy. It was found that extended accuracy is necessary for reliable results as the stepsize in time is decreased. All of the programs are composed of an array of general subroutines. The subroutines were written with extreme versatility and will no doubt prove useful to others extending this work.

Roots of Function B

First, it is necessary to acquire the roots of the function B. It is known that there is an infinite set of negative real roots of this function. As the roots become larger, they contribute less and less to the value of the coefficients. If the values of the denominator coefficients are to be accurate within $\pm 10^{-8}$, then

$$e^{-\beta_{max}\Delta} < 10^{-8} \quad . \qquad [2]$$

From this it is noted that the value of the stepsize, Δ, is also a factor in determining how many roots of B are necessary.

The program that calculates the roots, β_m, of the function B is referred to in the discussion as Root-finder. Input to the program is N, the number of layers, R1, the inner surface resistance, RN, the outer surface resistance, A, the thermal diffusivity of each layer, CON, the specific heat of each layer, and DELX, each layer's thickness. Root-finder is initialized at some starting value of the variable s. The value of B is then found at this s, then s is decremented and the value of B is again calculated. If the signs of the values of B differ, it is known that a root of the function B lies between the initial and the decremented value of s. When the general position of a root has been located, program control is sent to a subroutine named BETA. This subroutine converges on the root quickly but must have the value of the function's derivative to do so. For this reason, another subroutine, DERIV, is utilized. DERIV finds the derivative of each element of matrix (14) for a specified value of s. Two support subroutines are called by other subroutines. One is called MATMUL and does matrix multiplication. The other, HMADD, performs matrix addition.

The size of the constant used to decrement s is not strictly defined. A large constant reduces the computing time required to find the roots, but caution should be exercised so that too large a constant is not used, as two roots could be stepped over thus resulting in no change in sign of the function.

When the subroutines FIGURE and DERIV are executed, the value of the function and derivative for each element of the 2x2 matrix (14) at a particular s is calculated. Therefore, these subroutines may be used to calculate other constants also.

A Fortran listing of the ROOTFINDER program is located in Appendix C. The roots are printed and punched onto cards when ROOTFINDER is executed.

Evaluation of z-Coefficients

Once the roots have been obtained, other constants essential to the evaluation of the z-transfer function coefficients may be found. The constants, p_2, p_1, u_2, and u_1 do not depend on the roots of the function B as they are evaluated at s = 0. The expressions for p_2, p_1, u_2, and u_1 are again listed.

$$p_2 = [-\frac{A}{B}]\Big|_{s=0} \quad , \quad p_1 = -[\frac{B\frac{dA}{ds} - A\frac{dB}{ds}}{B^2}]\Big|_{s=0} \quad ,$$

$$u_2 = [\frac{1}{B}]\Big|_{s=0} \quad , \quad u_1 = -[\frac{\frac{dB}{ds}}{B^2}]\Big|_{s=0} \tag{26}$$

If zero is substituted for the variable s in the functions A, B, $\frac{dA}{ds}$, or $\frac{dB}{ds}$, indeterminant forms are obtained. However, simpler expressions may be derived that produce numerical results.

It has been shown that

$$\begin{bmatrix} A_r & B_r \\ C_r & D_r \end{bmatrix} = \begin{bmatrix} \cosh(\sqrt{s/\alpha_r}\,\Delta x_r) & -\frac{1}{k_r}\sqrt{\alpha_r/s}\,\sinh(\sqrt{s/\alpha_r}\,\Delta x_r) \\ -k_r\sqrt{s/\alpha_r}\,\sinh(\sqrt{s/\alpha_r}\,\Delta x_r) & \cosh(\sqrt{s/\alpha_r}\,\Delta x_r) \end{bmatrix} .$$

When s equals zero, the B_r term is indeterminant, but by applying L'Hospital's Rule or a series expansion the following equality may be derived.

$$\begin{bmatrix} A_r & B_r \\ C_r & D_r \end{bmatrix}\Bigg|_{s=0} = \begin{bmatrix} 1 & -\frac{\Delta x_r}{k_r} \\ 0 & 1 \end{bmatrix} . \tag{27}$$

By expanding, it may be seen that

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}\Bigg|_{s=0} = \begin{bmatrix} 1 & -R_1 - \dfrac{\Delta x_1}{k_1} - \dfrac{\Delta x_2}{k_2} - \cdots - \dfrac{\Delta x_n}{k_n} - R_n \\ 0 & 1 \end{bmatrix} \tag{28}$$

By differentiating, it can be shown that

$$\begin{bmatrix} \dfrac{dA_r}{ds} & \dfrac{dB_r}{ds} \\ \dfrac{dC_r}{ds} & \dfrac{dD_r}{ds} \end{bmatrix} = \begin{bmatrix} \dfrac{\Delta x_r}{2\sqrt{\alpha_r s}} \sinh(\sqrt{s/\alpha_r}\,\Delta x_r) & -\dfrac{\Delta x_r}{2k_r s} \cosh(\sqrt{s/\alpha_r}\,\Delta x_r) + \dfrac{1}{2k_r s}\sqrt{\alpha_r/s}\,\sinh(\sqrt{s/\alpha_r}\,\Delta x_r) \\ -\dfrac{k_r \Delta x_r}{2\alpha_r} \cosh(\sqrt{s/\alpha_r}\,\Delta x_r) - \dfrac{k_r}{2\sqrt{\alpha_r/s}}\,\Delta x_r & \dfrac{\Delta x_r}{2\sqrt{\alpha_r/s}} \sinh(\sqrt{s/\alpha_r}\,\Delta x_r) \end{bmatrix}.$$

And with techniques used in evaluation of expression (17), the much simplified result below is found.

$$\begin{bmatrix} \dfrac{dA_r}{ds} & \dfrac{dB_r}{ds} \\ \dfrac{dC_r}{ds} & \dfrac{dD_r}{ds} \end{bmatrix}\Bigg|_{s=0} = \begin{bmatrix} \dfrac{\Delta x_r^2}{2\alpha_r} & -\dfrac{\Delta x_r^3}{6\alpha_r k_r} \\ -\dfrac{k_r \Delta x_r}{\alpha_r} & \dfrac{\Delta x_r^2}{2\alpha_r} \end{bmatrix} \tag{29}$$

The derivatives of the elements of expression (14) in general are

$$\begin{aligned} \begin{bmatrix} \dfrac{dA}{ds} & \dfrac{dB}{ds} \\ \dfrac{dC}{ds} & \dfrac{dB}{ds} \end{bmatrix} &= \begin{bmatrix} 1 & -R_n \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \dfrac{dA_n}{ds} & \dfrac{dB_n}{ds} \\ \dfrac{dC_n}{ds} & \dfrac{dD_n}{ds} \end{bmatrix} \begin{bmatrix} A_{n-1} & B_{n-1} \\ C_{n-1} & D_{n-1} \end{bmatrix} \cdots \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} 1 & -R_I \\ 0 & 1 \end{bmatrix} \\ &+ \begin{bmatrix} 1 & -R_n \\ 0 & 1 \end{bmatrix} \begin{bmatrix} A_n & B_n \\ C_n & D_n \end{bmatrix} \begin{bmatrix} \dfrac{dA_{n-1}}{ds} & \dfrac{dB_{n-1}}{ds} \\ \dfrac{dC_{n-1}}{ds} & \dfrac{dD_{n-1}}{ds} \end{bmatrix} \cdots \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} 1 & -R_I \\ 0 & 1 \end{bmatrix} \\ &+ \cdots + \begin{bmatrix} 1 & -R_n \\ 0 & 1 \end{bmatrix} \begin{bmatrix} A_n & B_n \\ C_n & D_n \end{bmatrix} \begin{bmatrix} A_{n-1} & B_{n-1} \\ C_{n-1} & D_{n-1} \end{bmatrix} \cdots \\ &\cdots \begin{bmatrix} \dfrac{dA_1}{ds} & \dfrac{dB_1}{ds} \\ \dfrac{dC_1}{ds} & \dfrac{dD_1}{ds} \end{bmatrix} \begin{bmatrix} 1 & -R_I \\ 0 & 1 \end{bmatrix} . \end{aligned} \tag{30}$$

Expressions (28), (29), and (30) are used separately and combined to evaluate the constants in expression (26). A subroutine, CONSTS, utilizes these expressions and finds numerical results for the constants p_2, p_1, u_2, and u_1. These constants are then used by the main program to find the z-coefficients for the transfer functions R(z) and W(z). In this thesis, the main program being referenced will be called Z-Coefficients.

Input to the Z-Coefficients program is N, the number of roots, NLAY, the number of structure layers, DELTA, the size of the time increment, RI, the inner surface resistance, RN, the outer surface resistance, A, the thermal diffusivity of each layer, CON, each layers specific heat, DELX, the thickness of each layer, and BETA, the roots of the function B. Once this data has been read, Z-Coefficients calls the subroutine CONSTS and thus obtains values for p_2, p_1, u_2, and u_1 to be used later. The denominator polynomial is then calculated by looping through a few statements that initialize and then call a polynomial multiplication subroutine, PMPY. When this loop has been executed, the denominator coefficients are printed. Recall that R(z) and W(z) have the same denominators.

Two other sets of constants must be calculated before the numerator coefficients can be evaluated. These are q_m and v_m, and are evaluated at values of the roots as shown.

$$q_m = \left[- \frac{A}{s^2 \frac{dB}{ds}} \right] \Bigg|_{s=-\beta_m} \quad , \quad v_m = \left[\frac{1}{s^2 \frac{dB}{ds}} \right] \Bigg|_{s=-\beta_m}$$

These constants are found with relative ease as subroutine FIGURE calculates A, B, C or D at any negative real values, subroutine DERIV calculates the derivatives at any negative real value, and the roots are already known.

Once the necessary constants have been found, the numerator coefficients may be calculated. The numerator coefficients may be found by expanding the numerators of the z-transfer functions, see expression (24), but a quicker and clearer approach is suggested in reference [2].

From equation (23),

$$\frac{A}{s^2 B} = \frac{p_2}{s^2} + \frac{p_1}{s} + \sum_{m=1}^{\infty} \left(\frac{q_m}{s+\beta_m}\right) .$$

The inverse Laplace transform of this equation will be called O, some output, and is found to be

$$O(t) = L^{-1}\left[\frac{A}{s^2 B}\right] = p_2 t + p_1 + \sum_{m=1}^{\infty} q_m e^{-\beta_m t} .$$

This equality can be evaluated for $t = \Delta, 2\Delta, \ldots, k\Delta$ to find the values of the output at $O_2, O_3, \ldots, O_{k+1}$. These values of the output at multiples of Δ may be recognized as the sampled values of the output function. In essence, the above equation is used to find values of the z-transform of the output, $O(z)$.

It has been shown previously that in general

$$O(z) = \frac{N(z)}{D(z)} \cdot I(z) .$$

Here we seek the numerator, $N(z)$, and

$$N(z) = \frac{D(z)}{I(z)} \cdot O(z) .$$

From expression (24), it is known that

$$D(z) = \prod_{m=1}^{\infty} (1 - e^{-\beta_m \Delta} z^{-1}), \quad \text{and}$$

the z-transform of a ramp input is

$$I(z) = \frac{\Delta}{z(1-z^{-1})^2} .$$

Combining,

$$N(z) = \frac{z(1-z^{-1})^2}{\Delta} \prod_{m=1}^{\infty} (1 - e^{-\beta_m \Delta} z^{-1}) \cdot O(z) .$$

This may be reduced to

$$N(z) = \frac{z(1-z^{-1})^2}{\Delta} (b_1 + b_2 z^{-1} + b_3 z^{-2} + \ldots) \cdot O(z)$$

because the denominator may be expressed in the equivalent infinite series form in z^{-1}. The output may be expressed in the same form, so

$$N(z) = \frac{z(1-z^{-1})^2}{\Delta} (b_1 + b_2 z^{-1} + b_3 z^{-2} + \ldots)(O_1 + O_2 z^{-1} + O_3 z^{-2} + \ldots) .$$

When this equation is expanded it is found to contain a z^1 term. But, because the output at $t = 0$, O_1 is to be zero, the coefficient of this term is zero. Below, $N(z)$ is partially expanded.

$$N(z) = a_1 + a_2 z^{-1} + \ldots = (O_2 b_1 + O_1 b_2 - 2O_1 b_1) + (O_3 b_1 + O_2 b_2 + O_1 b_3 - 2O_1 b_2 + O_1 b_1) z^{-1} + \ldots$$

The computer program Z-Coefficients uses this concept to evaluate the numerator coefficients for the transfer functions R(z) and W(z). Because both transfer functions have the same denominator, many of the calculations need be performed only once. To take full advantage of the likenesses, a subroutine called THENUM was constructed. THENUM finds the coefficients for the output polynomial according to constants passed to it and then finds the numerator coefficients. A listing of Z-COEFFICIENTS is in Appendix D. The z-coefficients are punched onto cards as well as printed when the program is executed.

It would be a simple task to combine the Rootfinder and Z-Coefficient programs into a single program. However, they were not merged because roots of a particular wall may be used with any size step in time. In other words, to obtain z-coefficients for a variety of stepsizes for one wall, the roots only need to be calculated once. If the programs were combined, the roots would have to be found each time a run was made to obtain z-coefficients for a different stepsize.

Calculation of Heat Flux

When the z-coefficients for a particular structure and stepsize have been attained, they are applied in a short iterative subprogram called FCALC that uses the equations from Table 2 to figure the heat flux at multiples of the time stepsize, Δ. Input to the subroutine is the z-coefficients, A, B, and C, the number of coefficients in the individual numerators and denominator, ITHETA, the initial wall temperature, TGO, the sampled inside temperature, TIN, the sampled outside temperature, TOUT, and the surface area of the structure, SAREA.

It has been theoretically proven that there is an infinite set of negative real roots of the function B, and therefore there is also an infinite set of numerator and denominator coefficients. Fortunately, the values of the coefficients quickly converge to zero and so those coefficients beyond some finite number of coefficients may be truncated. The resulting error will of course depend on the magnitude of the truncated numbers but should be negligible.

The FCALC subroutine is constructed in a manner such that it truncates those coefficients beyond the initial ITHETA coefficients of a numerator or denominator polynomial. The subroutine FCALC is located in Appendix E.

CHAPTER IV

APPLICATIONS

The programs described in Chapter III are quite general in nature and may be exercised on a variety of problems. This chapter deals with a few of those applications.

Two-Layer Wall

As an example, consider the case of a layered wall as described in Table 3. This particular wall was used by Kusuda [5], and also Mitalas

Table 3. Double-Layer Wall

Layer r	Material	Thermal Diffusivity (ft^2/hr)	Thermal Conductivity ($\frac{BTU}{hr\text{-}ft\text{-}°F}$)	Thickness (ft)	Resistance ($\frac{ft^2\text{-}°F\text{-}hr}{BTU}$)
-	Inside Surface Resistance	-	-	-	.83333
1	Common Brick	.09014	.42	.3333	-
2	Face Brick	.028	.77	.333	-
-	Outside Surface Resistance	-	-	-	.33333

and Stephenson [2]. It was used here so that this work could be verified.

When the data from Table 3 is entered into the Rootfinder program, the roots are printed, as shown in Table 4, and are also punched onto computer cards to be used by the z-coefficients program.

The z-coefficients for various time stepsizes may be computed. Tables 5 and 6 contain the z-coefficients for stepsizes of 0.1 hr and 0.2 hr, respectively for this particular wall. The z-coefficients are punched onto cards as well as printed when the z-coefficients program is executed.

Table 4. Roots of Double-Layer Wall.

ROOT S	VALUE OF FUNCTION B	DERIVATIVE OF FUNCTION B
-0.1743564895101859D 00	-0.0000000000000001	8.8940418637566510
-0.8432694983033066D 00	0.0000000000000000	-3.7797504691904740
-0.2565017965951630D 01	-0.0000000000000002	2.7572451845512920
-0.4852721524718708D 01	-0.0000000000000000	-2.5754774155284310
-0.8846706295323562D 01	-0.0000000000000004	2.4213817385476550
-0.1283071051485510D 02	-0.0000000000000040	-2.3854502222884680
-0.1912259905218523D 02	-0.0000000000000048	2.3768087035232910
-0.2497022678074264D 02	-0.0000000000000005	-2.3312875410609280
-0.3328366468070444D 02	-0.0000000000000161	2.3970685043904460
-0.4138646432284575D 02	0.0000000000000182	-2.3223570635453210
-0.5128558483671618D 02	-0.0000000000000504	2.4108819580441560
-0.6208162611280464D 02	0.0000000000000139	-2.3141199866501440
-0.7319013223035003D 02	-0.0000000000000354	2.3854942408267920
-0.8692650202388925D 02	0.0000000000000054	-2.2977240581664350
-0.9915817991364163D 02	-0.0000000000000008	2.3341394989424200
-0.1157024765087853D 03	0.0000000000000109	-2.2983109016393630
-0.1294076826596802D 03	-0.0000000000000186	2.2961333670992520
-0.1482046885839709D 03	0.0000000000000057	-2.3327806523320460
-0.1641155307097936D 03	0.0000000000000681	2.2915965398418360
-0.1843558997140948D 03	0.0000000000000017	-2.3777534023878680

ILLEGIBLE DOCUMENT

THE FOLLOWING DOCUMENT(S) IS OF POOR LEGIBILITY IN THE ORIGINAL

THIS IS THE BEST COPY AVAILABLE

P-474

Table 5. z-Coefficients for DELTA = 0.1 hr.

STEPSIZE= 0.1 HR

NUMERATOR P(Z)	DENOMINATOR R(Z) AND W(Z)	NUMERATOR W(Z)
A(1)=-0.2591423565353883D 01	B(1)= 0.1000000000000000D 01	C(1)= 0.3108624468950438D-13
A(2)= 0.1135242918564857D 02	B(2)=-0.4271866558267723D 01	C(2)=-0.2039505626872213D-12
A(3)=-0.2072242723703434D 02	B(3)= 0.7591639323235606D 01	C(3)=-0.4150546957673254D-09
A(4)= 0.2046444366696574D 02	B(4)=-0.7284085046228002D 01	C(4)=-0.5581815512944143D-07
A(5)=-0.1189551792132419D 02	B(5)= 0.4103621913336194D 01	C(5)=-0.9493934405823400D-06
A(6)= 0.4167576317476500D 01	B(6)=-0.1387013575092094D 01	C(6)=-0.4149725312678341D-05
A(7)=-0.8688806605493597D 00	B(7)= 0.2774005151065439D 00	C(7)=-0.6189386438059239D-05
A(8)= 0.1039016602344559D 00	B(8)=-0.3150555060206599D-01	C(8)=-0.3581295623969245D-05
A(9)=-0.6733341569847039D-02	B(9)= 0.1930695803602597D-02	C(9)=-0.8474771621565948D-06
A(10)= 0.2198018123892255D-03	B(10)=-0.5635764231409930D-04	C(10)=-0.8266966236811080D-07
A(11)=-0.3367477866434306D-05	B(11)= 0.7356640335522530D-06	C(11)=-0.3243243446441188D-08
A(12)= 0.2257551219229261D-07	B(12)=-0.3833690455532131D-08	C(12)=-0.4894975261167957D-10
A(13)=-0.5327690080576011D-10	B(13)= 0.7059280351052376D-11	C(13)=-0.3946426881594245D-12
A(14)= 0.6353227534358221D-13	B(14)=-0.4171432515717200D-14	C(14)= 0.1400354441777731D-12
A(15)= 0.1577905393455523D-12	B(15)= 0.6856302733873412D-18	C(15)=-0.1308376099294248D-12
A(16)=-0.5447056944847376D-13	B(16)=-0.3039658021039731D-22	C(16)= 0.8088157397513427D-13
A(17)= 0.1567249516794574D-12	B(17)= 0.2935664919135673D-27	C(17)=-0.1099769014721717D-13
A(18)=-0.4866139086663356D-13	B(18)=-0.6354915322419894D-33	C(18)=-0.2004370237161521D-13
A(19)= 0.2062534564553716D-12	B(19)= 0.2347333081736970D-39	C(19)=-0.1232917979433173D-13
A(20)=-0.9509946352276781D-13	B(20)=-0.5550557439525150D-73	C(20)= 0.6111799602756399D-13
A(21)= 0.2166446819378502D-12	B(21)= 0.0000000000000000D 00	C(21)=-0.7807316942785040D-13
A(22)= 0.3759374648016096D-13	B(22)= 0.0000000000000000D 00	C(22)= 0.7770035509149453D-13
A(23)= 0.1553494320521048D-12	B(23)= 0.0000000000000000D 00	C(23)=-0.7836803257603766D-13
A(24)=-0.5737116055730067D-14	B(24)= 0.0000000000000000D 00	C(24)= 0.5550632496312106D-13
A(25)= 0.2530535978811056D-12	B(25)= 0.0000000000000000D 00	C(25)=-0.9421719053181015D-14
A(26)=-0.1363274830170636D-12	B(26)= 0.0000000000000000D 00	C(26)=-0.3675669576613195D-13
A(27)= 0.3172723986754729D-12	B(27)= 0.0000000000000000D 00	C(27)= 0.5931171537381189D-13
A(28)=-0.1010628204942511D-12	B(28)= 0.0000000000000000D 00	C(28)=-0.6651475824682597D-13
A(29)= 0.2570430968085012D-12	B(29)= 0.0000000000000000D 00	C(29)= 0.6489519784095455D-13
A(30)=-0.2051272603044409D-13	B(30)= 0.0000000000000000D 00	C(30)=-0.5081639484307308D-13
A(31)= 0.1756277033001991D-12	B(31)= 0.0000000000000000D 00	C(31)= 0.1272486644951851D-13
A(32)= 0.1312473065436501D-12	B(32)= 0.0000000000000000D 00	C(32)= 0.4003459127570496D-13
A(33)= 0.1001595310068952D-12	B(33)= 0.0000000000000000D 00	C(33)=-0.6549426029343707D-13
A(34)= 0.1924336029988220D-12	B(34)= 0.0000000000000000D 00	C(34)= 0.1152095528111193D-12
A(35)= 0.5758595548100166D-13	B(35)= 0.0000000000000000D 00	C(35)=-0.1083598718651658D-12
A(36)= 0.1244317681827835D-12	B(36)= 0.0000000000000000D 00	C(36)= 0.8193176921198659D-13
A(37)= 0.8625474187126713D-13	B(37)= 0.0000000000000000D 00	C(37)=-0.3275347952051970D-13
A(38)= 0.1374863048627084D-12	B(38)= 0.0000000000000000D 00	C(38)= 0.3204073607845432D-14
A(39)= 0.1130086923507296D-12	B(39)= 0.0000000000000000D 00	C(39)= 0.1603223522468496D-13
A(40)= 0.1213511743814636D-12	B(40)= 0.0000000000000000D 00	C(40)= 0.1118127847231920D-14
A(41)= 0.1216024302294568D-12	B(41)= 0.0000000000000000D 00	C(41)=-0.5066950849229730D-14
A(42)= 0.2057601177897052D-12	B(42)= 0.0000000000000000D 00	C(42)= 0.3065775857968649D-13
A(43)= 0.1683856425792445D-12	B(43)= 0.0000000000000000D 00	C(43)=-0.5383245115916717D-13

Table 6. z-Coefficients for DELTA = 0.2 hr.

STEPSIZE= 0.2 HR

NUMERATOR R(Z)	DENOMINATOR R(Z) AND W(Z)	NUMERATOR W(Z)
A(1)=-0.2450318060473988D 01	B(1)= 0.1000000000000000D 01	C(1)=-0.4440892098500626D-14
A(2)= 0.7865987688652078D 01	B(2)=-0.3065565245174912D 01	C(2)=-0.2924699231436791D-07
A(3)=-0.9739422552866042D 01	B(3)= 0.3606952807650774D 01	C(3)=-0.5536230854313999D-05
A(4)= 0.5856843858313248D 01	B(4)=-0.2046936192526890D 01	C(4)=-0.7017987693699457D-04
A(5)=-0.1777315781653922D 01	B(5)= 0.5799113025576784D 00	C(5)=-0.1788315042989248D-03
A(6)= 0.2600736619315713D 00	B(6)=-0.7767681174270157D-01	C(6)=-0.1271018409726776D-03
A(7)=-0.1664998132164863D-01	B(7)= 0.4383663459460430D-02	C(7)=-0.2797771814192925D-04
A(8)= 0.4019948283106530D-03	B(8)=-0.8522096338270362D-04	C(8)=-0.1396691630646155D-05
A(9)=-0.3365639574504596D-05	B(9)= 0.4841340503788205D-06	C(9)=-0.3632238268217000D-07
A(10)= 0.8667345131826319D-09	B(10)=-0.5567404401173144D-07	C(10)=-0.1754807673965813D-09
A(11)=-0.6259785420175267D-11	B(11)= 0.1401404559025283D-12	C(11)=-0.1940173963953727D-12
A(12)= 0.4296486891925910D-13	B(12)=-0.4736153864444310D-17	C(12)= 0.2793461754947493D-13
A(13)=-0.1527088534594231D-13	B(13)= 0.1930644508953164D-22	C(13)=-0.4527591889895102D-13
A(14)= 0.1534207512646719D-14	B(14)=-0.7657341503664792D-29	C(14)= 0.4552857534071127D-13
A(15)=-0.3500165475615112D-13	B(15)= 0.2261678851913260D-36	C(15)=-0.3533915250654972D-13
A(16)= 0.1636582041076212D-13	B(16)=-0.2099364035898194D-73	C(16)= 0.1509555397198161D-13
A(17)=-0.3541456433690100D-13	B(17)= 0.0000000000000000D 00	C(17)= 0.1532908603477975D-14
A(18)=-0.6702473066155914D-14	B(18)= 0.0000000000000000D 00	C(18)=-0.9461275159342279D-14
A(19)=-0.1151467793807833D-13	B(19)= 0.0000000000000000D 00	C(19)= 0.8591459700182019D-14
A(20)=-0.3239645047523795D-13	B(20)= 0.0000000000000000D 00	C(20)=-0.3059030213731624D-14
A(21)=-0.1908446066733335D-13	B(21)= 0.0000000000000000D 00	C(21)=-0.2373925452766650D-14
A(22)=-0.2109467015436922D-13	B(22)= 0.0000000000000000D 00	C(22)= 0.7747391173904187D-15
A(23)=-0.3995976452710050D-13	B(23)= 0.0000000000000000D 00	C(23)= 0.3557306210078458D-14
A(24)=-0.2289570571697460D-13	B(24)= 0.0000000000000000D 00	C(24)=-0.6929858902963416D-14
A(25)=-0.2759630271472070D-13	B(25)= 0.0000000000000000D 00	C(25)= 0.7917030341681344D-14
A(26)=-0.3067072474034698D-13	B(26)= 0.0000000000000000D 00	C(26)=-0.1140153745363019D-13
A(27)=-0.1477253462603453D-15	B(27)= 0.0000000000000000D 00	C(27)= 0.1213013967568192D-13
A(28)=-0.2566266464336019D-13	B(28)= 0.0000000000000000D 00	C(28)=-0.1567451452762477D-13
A(29)=-0.1324102864073722D-14	B(29)= 0.0000000000000000D 00	C(29)= 0.2204223587434549D-13
A(30)=-0.5604152761691871D-13	B(30)= 0.0000000000000000D 00	C(30)=-0.2757015610977618D-13
A(31)= 0.2267793658503176D-13	B(31)= 0.0000000000000000D 00	C(31)= 0.1954447870542794D-13
A(32)=-0.4274036587005435D-13	B(32)= 0.0000000000000000D 00	C(32)=-0.1530076489624841D-13
A(33)=-0.2946629366261267D-13	B(33)= 0.0000000000000000D 00	C(33)= 0.1442457175563824D-13

Once the z-coefficients have been obtained, the heat flux at the inside surface, f_I may be found with the FCALC subroutine where the inside and outside temperatures are input. For this example, Figures 6 and 7 demonstrate the inside and outside temperatures. The wall was considered to have been at

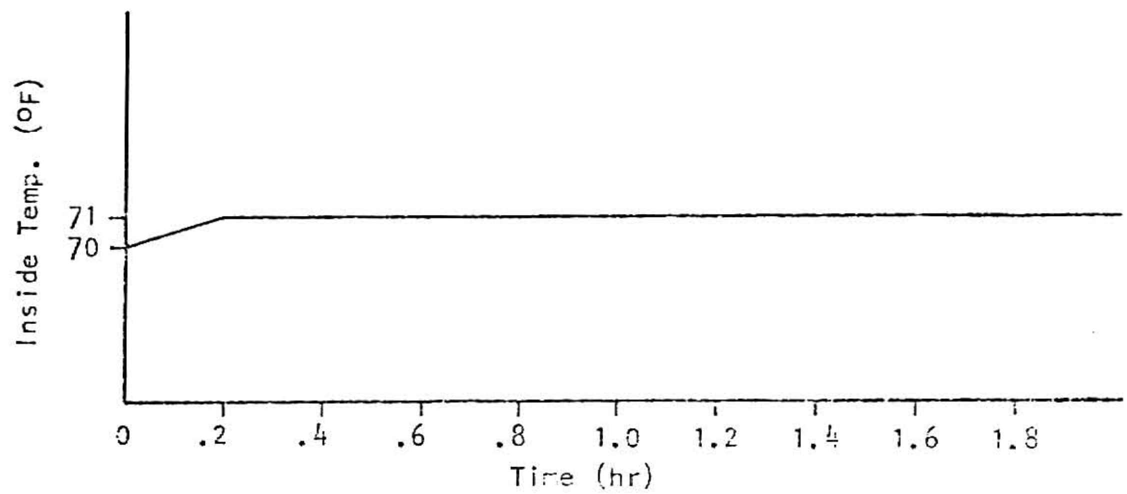

Figure 6. Inside Temperature.

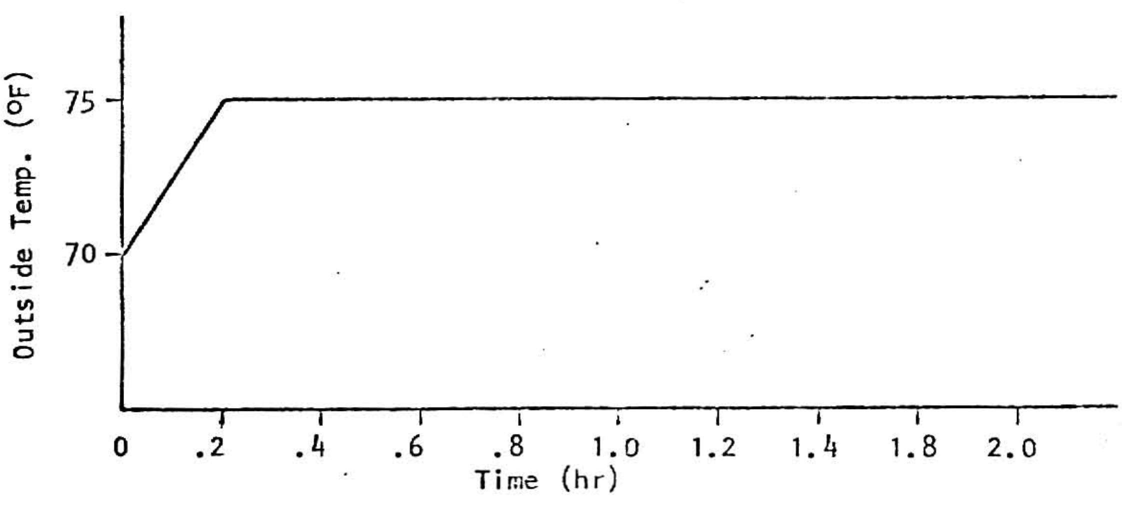

Figure 7. Outside Temperature.

an initial temperature of 70°F. Tables 7 and 8 contain the output from the FCALC subroutine for the 0.1 hr and 0.2 hr stepsizes. It may be seen that the two sets of data agree very well at the same points in time. Figure 8 is a plot of the data. As expected, the inside heat flux is immediately positive and eventually falls to negative values because of the higher outside temperature.

3-Layer Wall

For comparison, consider the three-layer wall as described by Table 9. The rootfinder program was used to obtain the roots, then the z-coefficients were obtained, and finally an output was obtained from the FCALC subroutine for the inputs used on the other wall as shown in Figures 6 and 7. The wall

Table 9. Triple-Layer Wall.

Layer	Material	Thermal Diffusivity (ft^2/hr)	Thermal Conductivity ($\frac{BTU}{hr\text{-}ft\text{-}°F}$)	Thickness (ft)	Resistance ($\frac{ft^2\text{-}°F\text{-}hr}{BTU}$)
-	Inside Surface Resistance	-	-	-	.83333
1	Gypsum Board	.012375	.25	.0416667	-
2	Insulation	.3015873	.19	.5	-
3	Gypsum Board	.012375	.25	.0416667	-
-	Outside Surface Resistance	-	-	-	.33333

was again assumed to have been at an initial temperature of 70°F. Table 10 contains the output of the FCALC subroutine. The responses of both walls are plotted in Figure 9. From Figure 9, it may be seen that the 3-layer wall responds faster than the 2-layer wall, and that it has a higher overall thermal resistance because of the lower heat flow at steady-state.

Table 7. Heat Flux for DELTA = 0.1 hr.

F(1)= 0.0000000000000000D 00	F(57)= 0.3942761539552927D-02	F(113)=-0.1036517361326644D 01	F(169)=-0.1432276641576101D 01
F(2)= 0.1295711924972158D 01	F(58)=-0.2434405923768498D-01	F(114)=-0.1047486641362629D 01	F(170)=-0.1436411108144102D 01
F(3)= 0.2450318060455893D 01	F(59)=-0.5219628226293319D-01	F(115)=-0.1058266354577382D 01	F(171)=-0.1440474116318400D 01
F(4)= 0.2231062195293558D 01	F(60)=-0.7961703130600096D-01	F(116)=-0.1068360413776795D 01	F(172)=-0.1444466901268349D 01
F(5)= 0.2055994511141836D 01	F(61)=-0.1066097298016836D 00	F(117)=-0.1079271768161636D 01	F(173)=-0.1448390576451018D 01
F(6)= 0.1994372644855570D 01	F(62)=-0.1331780067341737D 00	F(118)=-0.1089550354266368D 01	F(174)=-0.1452246634372321D 01
F(7)= 0.1910804378737433D 01	F(63)=-0.1553259635754612D 00	F(119)=-0.1099558910873235D 01	F(175)=-0.1456035946952009D 01
F(8)= 0.1840771731802160D 01	F(64)=-0.1850575671039698D 00	F(120)=-0.1109440595710479D 01	F(176)=-0.1459759765875100D 01
F(9)= 0.1779903299860435D 01	F(65)=-0.2133777175767544D 00	F(121)=-0.1119151870102733D 01	F(177)=-0.1463419222954355D 01
F(10)= 0.1725730115011655D 01	F(66)=-0.2352892671367108D 00	F(122)=-0.1128695557666032D 01	F(178)=-0.1467015430402359D 01
F(11)= 0.1675492954778551D 01	F(67)=-0.2597984500392414D 00	F(123)=-0.1138074533385704D 01	F(179)=-0.1470549481307962D 01
F(12)= 0.1613331512381027D 01	F(68)=-0.2839094503510365D 00	F(124)=-0.1147291624322667D 01	F(180)=-0.1474022449826633D 01
F(13)= 0.1587762993347683D 01	F(69)=-0.3076270931117171D 00	F(125)=-0.1156349610260344D 01	F(181)=-0.1477435591575773D 01
F(14)= 0.1546533626053308D 01	F(70)=-0.3309562862201775D 00	F(126)=-0.1165251224431155D 01	F(182)=-0.1480785345400730D 01
F(15)= 0.1506471031033636D 01	F(71)=-0.3539020055842473D 00	F(127)=-0.1173999154181698D 01	F(183)=-0.1484085326385623D 01
F(16)= 0.1467233475264420D 01	F(72)=-0.3764692416069441D 00	F(128)=-0.1182596069165972D 01	F(184)=-0.1487324340766645D 01
F(17)= 0.1428465350106658D 01	F(73)=-0.3986621068540508D 00	F(129)=-0.1191044484494253D 01	F(185)=-0.1490507371633977D 01
F(18)= 0.1399950855567711D 01	F(74)=-0.4204888251014726D 00	F(130)=-0.1199347036472114D 01	F(186)=-0.1493635386538376D 01
F(19)= 0.1351533250603424D 01	F(75)=-0.4419513207983254D 00	F(131)=-0.1207506208221026D 01	F(187)=-0.1496709533629373D 01
F(20)= 0.1313113307288435D 01	F(76)=-0.4630355810292035D 00	F(132)=-0.1215524467823421D 01	F(188)=-0.1499730155305295D 01
F(21)= 0.1277463183676396D 01	F(77)=-0.4838074333160979D 00	F(133)=-0.1223404241586547D 01	F(189)=-0.1502698761829924D 01
F(22)= 0.1236351382760954D 01	F(78)=-0.5042113255266188D 00	F(134)=-0.1231147914591643D 01	F(190)=-0.1505616358255565D 01
F(23)= 0.1197359992082541D 01	F(79)=-0.5242776117123050D 00	F(135)=-0.1238757911300614D 01	F(191)=-0.1508482913549933D 01
F(24)= 0.1158645554584317D 01	F(80)=-0.5439763596849593D 00	F(136)=-0.1246236296679796D 01	F(192)=-0.1511300252714360D 01
F(25)= 0.1115931233780518D 01	F(81)=-0.5633377747953565D 00	F(137)=-0.1253585575857875D 01	F(193)=-0.1514068726384750D 01
F(26)= 0.1033981212568412D 01	F(82)=-0.5824551794985513D 00	F(138)=-0.1260087975736748D 01	F(194)=-0.1516789650789744D 01
F(27)= 0.1042128498497202D 01	F(83)=-0.6011944863887275D 00	F(139)=-0.1267790544512650D 01	F(195)=-0.1519463362480190D 01
F(28)= 0.1003335604732610D 01	F(84)=-0.6196178393661269D 00	F(140)=-0.1274890375458151D 01	F(196)=-0.1522090527301510D 01
F(29)= 0.9645571042503880D 00	F(85)=-0.6377298050031571D 00	F(141)=-0.1281734801394088D 01	F(197)=-0.1524673044193485D 01
F(30)= 0.9259102402344488D 00	F(86)=-0.6555342198667866D 00	F(142)=-0.1288470801432550D 01	F(198)=-0.1527210550354729D 01
F(31)= 0.8874028357553899D 00	F(87)=-0.6730316163846791D 00	F(143)=-0.1295090416495407D 01	F(199)=-0.1529704157691936D 01
F(32)= 0.8495689103920114D 00	F(88)=-0.6902402365281858D 00	F(144)=-0.1301599660519390D 01	F(200)=-0.1532154584221406D 01
F(33)= 0.8103382138146134D 00	F(89)=-0.7071512762625180D 00	F(145)=-0.1307988501033662D 01	F(201)=-0.1534562054700215D 01
F(34)= 0.7713413616105335D 00	F(90)=-0.7237739977184126D 00	F(146)=-0.1314270875720687D 01	F(202)=-0.1536927940180142D 01
F(35)= 0.7354062745434554D 00	F(91)=-0.7401130623966060D 00	F(147)=-0.1320444763024209D 01	F(203)=-0.1539425504434689D 01
F(36)= 0.6980551930948073D 00	F(92)=-0.7561730772631543D 00	F(148)=-0.1326511944614006D 01	F(204)=-0.1541540485525578D 01
F(37)= 0.6601014346875332D 00	F(93)=-0.7719585935822395D 00	F(149)=-0.1332474146022961D 01	F(205)=-0.1543786432106756D 01
F(38)= 0.6243323192152199D 00	F(94)=-0.7874741059496344D 00	F(150)=-0.1338333417140369D 01	F(206)=-0.1545993554860055D 01
F(39)= 0.5879364457050500D 00	F(95)=-0.8027230514856374D 00	F(151)=-0.1344091436752458D 01	F(207)=-0.1548162528743950D 01
F(40)= 0.5519343295505513D 00	F(96)=-0.8177128092008405D 00	F(152)=-0.1349749953065349D 01	F(208)=-0.1550294013128536D 01
F(41)= 0.5163103035010309D 00	F(97)=-0.8324465946944718D 00	F(153)=-0.1355310684219915D 01	F(209)=-0.1552318555567148D 01
F(42)= 0.4810764704650036D 00	F(98)=-0.8469239837870587D 00	F(154)=-0.1360775318799558D 01	F(210)=-0.1554447094093917D 01
F(43)= 0.4452445435121414D 00	F(99)=-0.8611548652704312D 00	F(155)=-0.1366145516339605D 01	F(211)=-0.1556469553217929D 01
F(44)= 0.4110217356594543D 00	F(100)=-0.8751414837647008D 00	F(156)=-0.1371422901169226D 01	F(212)=-0.1558457348312912D 01
F(45)= 0.3778222741587596D 00	F(101)=-0.8888873256720632D 00	F(157)=-0.1376609095955644D 01	F(213)=-0.1560411383704926D 01
F(46)= 0.3442467903522440D 00	F(102)=-0.9023982140181157D 00	F(158)=-0.1381705656280758D 01	F(214)=-0.1562331153275032D 01
F(47)= 0.3111027146836547D 00	F(103)=-0.9156763115772162D 00	F(159)=-0.1386714136740224D 01	F(215)=-0.1564217740660253D 01
F(48)= 0.2783953163243996D 00	F(104)=-0.9287261300388459D 00	F(160)=-0.1391636053555086D 01	F(216)=-0.1566071719330912D 01
F(49)= 0.2451262651219128D 00	F(105)=-0.9415515103133314D 00	F(161)=-0.1396472917926030D 01	F(217)=-0.1567393652964423D 01
F(50)= 0.2143932245273325D 00	F(106)=-0.9541562427549843D 00	F(162)=-0.1401226183320328D 01	F(218)=-0.1569568409541756D 01
F(51)= 0.1329174824532662D 00	F(107)=-0.9665443577506942D 00	F(163)=-0.1405897299211572D 01	F(219)=-0.1571443570553819D 01
F(52)= 0.1519792573694085D 00	F(108)=-0.9787135277268409D 00	F(164)=-0.1410487584832268D 01	F(220)=-0.1573172674569248D 01
F(53)= 0.1214853687937729D 00	F(109)=-0.9906933679957160D 00	F(165)=-0.1414996734535377D 01	F(221)=-0.1574871871854995D 01
F(54)= 0.9143676242111263D-01	F(110)=-0.1002424370634270D 01	F(166)=-0.1419431870232907D 01	F(222)=-0.1576541695357021D 01
F(55)= 0.6183097034827374D-01	F(111)=-0.1013999737150241D 01	F(167)=-0.1423788247767597D 01	F(223)=-0.1578182664733101D 01
F(56)= 0.3266696762768890D-01	F(112)=-0.1025359147786670D 01	F(168)=-0.1428069461357886D 01	F(224)=-0.1579795206847114D 01

Table 8. Heat Flux for DELTA = 0.2 hr.

F(1)=	0.0000000000000000D 00	F(57)=	-0.1036517359523149D 01	F(113)=	-0.1581379993865511D 01	F(169)=	-0.1658696966144289D 01
F(2)=	0.2450318035882974D 01	F(58)=	-0.1058266352761224D 01	F(114)=	-0.1584467751736520D 01	F(170)=	-0.1659135045056449D 01
F(3)=	0.2095945115991375D 01	F(59)=	-0.1079271766333368D 01	F(115)=	-0.1587449691357654D 01	F(171)=	-0.1659558118735664D 01
F(4)=	0.1910304978737818D 01	F(60)=	-0.1099553809033400D 01	F(116)=	-0.1590329439130446D 01	F(172)=	-0.1659966689689932D 01
F(5)=	0.1779902289665680D 01	F(61)=	-0.1119151868251861D 01	F(117)=	-0.1593110497101920D 01	F(173)=	-0.1660361258794910D 01
F(6)=	0.1676482547793222D 01	F(62)=	-0.1138074531528301D 01	F(118)=	-0.1595796247622996D 01	F(174)=	-0.1660742305895178D 01
F(7)=	0.1597762573356943D 01	F(63)=	-0.1156347608388890D 01	F(119)=	-0.1598389556661043D 01	F(175)=	-0.1661110294402792D 01
F(8)=	0.1505471010357777D 01	F(64)=	-0.1173799152300640D 01	F(120)=	-0.1600894778571535D 01	F(176)=	-0.1661465671839351D 01
F(9)=	0.1423465350151358D 01	F(65)=	-0.1191034442604016D 01	F(121)=	-0.1603313759533711D 01	F(177)=	-0.1661808870367733D 01
F(10)=	0.1351533253690039D 01	F(66)=	-0.1207596206311247D 01	F(122)=	-0.1605649841334768D 01	F(178)=	-0.1662140337387696D 01
F(11)=	0.1274631808858241D 01	F(67)=	-0.1223434235679135D 01	F(123)=	-0.1607905864947197D 01	F(179)=	-0.1662460385961456D 01
F(12)=	0.1197395962278211D 01	F(68)=	-0.1238757829481173D 01	F(124)=	-0.1610086573683541D 01	F(180)=	-0.1662766953473755D 01
F(13)=	0.1119831233963371D 01	F(69)=	-0.1253385573914745D 01	F(125)=	-0.1612185618032752D 01	F(181)=	-0.1663068031145544D 01
F(14)=	0.1042123685203483D 01	F(70)=	-0.1267935443196317D 01	F(126)=	-0.1614220555882467D 01	F(182)=	-0.1663356257336471D 01
F(15)=	0.9635571044506123D 00	F(71)=	-0.1281734799457352D 01	F(127)=	-0.1616182456630240D 01	F(183)=	-0.1663634713549553D 01
F(16)=	0.8874123862170114D 00	F(72)=	-0.1295090416551115D 01	F(128)=	-0.1618077912689248D 01	F(184)=	-0.1663903556727377D 01
F(17)=	0.8109332843476330D 00	F(73)=	-0.1307588495082990D 01	F(129)=	-0.1619908922768587D 01	F(185)=	-0.1664163219615132D 01
F(18)=	0.7354062751355237D 00	F(74)=	-0.1320444701067217D 01	F(130)=	-0.1621675414219500D 01	F(186)=	-0.1664413972572578D 01
F(19)=	0.6610149475320213D 00	F(75)=	-0.1332474140599900D 01	F(131)=	-0.1623332236242993D 01	F(187)=	-0.1664656131557676D 01
F(20)=	0.5879364406415360D 00	F(76)=	-0.1344091434783036D 01	F(132)=	-0.1625030566090246D 01	F(188)=	-0.1664889942467925D 01
F(21)=	0.5163100647316692D 00	F(77)=	-0.1355310682245716D 01	F(133)=	-0.1626627406605907D 01	F(189)=	-0.1665115838508260D 01
F(22)=	0.4462445443362446D 00	F(78)=	-0.1366145514350111D 01	F(134)=	-0.1628159994274710D 01	F(190)=	-0.1665331354727671D 01
F(23)=	0.3778222790341070D 00	F(79)=	-0.1376604940110630D 01	F(135)=	-0.1629644298638278D 01	F(191)=	-0.1665544576395726D 01
F(24)=	0.3111023357616173D 00	F(80)=	-0.1386714134770762D 01	F(136)=	-0.1631073025163502D 01	F(192)=	-0.1665747085652147D 01
F(25)=	0.2461266262534395D 00	F(81)=	-0.1396472915934910D 01	F(137)=	-0.1632462617463501D 01	F(193)=	-0.1665944431871322D 01
F(26)=	0.1829174834750745D 00	F(82)=	-0.1405957247212951D 01	F(138)=	-0.1633795759367445D 01	F(194)=	-0.1666134141556654D 01
F(27)=	0.1214453656538782D 00	F(83)=	-0.1414198732536466D 01	F(139)=	-0.1635091077019129D 01	F(195)=	-0.1666317350620593D 01
F(28)=	0.6183397164957915D-01	F(84)=	-0.1423784328576058D 01	F(140)=	-0.1636338140825557D 01	F(196)=	-0.1666494280668208D 01
F(29)=	0.3942767631156274D-02	F(85)=	-0.1432727663556517D 01	F(141)=	-0.1637542467390253D 01	F(197)=	-0.1666656514727015D 01
F(30)=	-0.5219523108257593D-01	F(86)=	-0.1440474114303723D 01	F(142)=	-0.1638705521319521D 01	F(198)=	-0.1666833158222336D 01
F(31)=	-0.1056337285643128D 00	F(87)=	-0.1448350674432757D 01	F(143)=	-0.1639128711704260D 01	F(199)=	-0.1666985514195635D 01
F(32)=	-0.1559325967723284D 00	F(88)=	-0.1456033594493031D 01	F(144)=	-0.1640713420512916D 01	F(200)=	-0.1667143408999738D 01
F(33)=	-0.2103771744811378D 00	F(89)=	-0.1463415220915372D 01	F(145)=	-0.1641960550365675D 01	F(201)=	-0.1667292327770827D 01
F(34)=	-0.2597734487264069D 00	F(90)=	-0.1470594947927990D 01	F(146)=	-0.1642372582046184D 01	F(202)=	-0.1667435557279184D 01
F(35)=	-0.3076270517615937D 00	F(91)=	-0.1477435385544545D 01	F(147)=	-0.1643945544320077D 01	F(203)=	-0.1667574166045063D 01
F(36)=	-0.3539032003203953D 00	F(92)=	-0.1484385324351444D 01	F(148)=	-0.1644151025603996D 01	F(204)=	-0.1667708324664656D 01
F(37)=	-0.3986631854847645D 00	F(93)=	-0.1490507356001565D 01	F(149)=	-0.1645934173894579D 01	F(205)=	-0.1667837255885696D 01
F(38)=	-0.4419551317361165D 00	F(94)=	-0.1496709334254309D 01	F(150)=	-0.1646669439666861D 01	F(206)=	-0.1667967213695292D 01
F(39)=	-0.4938177511854113D 00	F(95)=	-0.1502765975978791D 01	F(151)=	-0.1647513364221418D 01	F(207)=	-0.1668082556750996D 01
F(40)=	-0.5262726102275345D 00	F(96)=	-0.1508482953500620D 01	F(152)=	-0.1648354507987751D 01	F(208)=	-0.1668199130634537D 01
F(41)=	-0.5633377732797474D 00	F(97)=	-0.1514068347659122D 01	F(153)=	-0.1649147031577076D 01	F(209)=	-0.1668311571564455D 01
F(42)=	-0.6011934748503713D 00	F(98)=	-0.1519463394193443D 01	F(154)=	-0.1649912194000275D 01	F(210)=	-0.1668420159061652D 01
F(43)=	-0.6377290344149103D 00	F(99)=	-0.1524673047141695D 01	F(155)=	-0.1650651526836772D 01	F(211)=	-0.1668525025232711D 01
F(44)=	-0.6730361622634717D 00	F(100)=	-0.1529674155647701D 01	F(156)=	-0.1651365325264179D 01	F(212)=	-0.1668626297605960D 01
F(45)=	-0.7071512746539985D 00	F(101)=	-0.1534962452944227D 01	F(157)=	-0.1652054566967495D 01	F(213)=	-0.1668724609934974D 01
F(46)=	-0.7401130607718841D 00	F(102)=	-0.1539755342238925D 01	F(158)=	-0.1652720386081346D 01	F(214)=	-0.1668814549539561D 01
F(47)=	-0.7715489919394232D 00	F(103)=	-0.1543780436088800D 01	F(159)=	-0.1653363289818657D 01	F(215)=	-0.1668905762610530D 01
F(48)=	-0.8027243496222650D 00	F(104)=	-0.1548163526682136D 01	F(160)=	-0.1653984157843825D 01	F(216)=	-0.1668979849520741D 01
F(49)=	-0.8324445578129350D 00	F(105)=	-0.1552398653923703D 01	F(161)=	-0.1654583750111454D 01	F(217)=	-0.1669082918452771D 01
F(50)=	-0.8611546257140948D 00	F(106)=	-0.1556469951152739D 01	F(162)=	-0.1655162794172063D 01	F(218)=	-0.1669165171660470D 01
F(51)=	-0.8888792396143495D 00	F(107)=	-0.1560411318163899D 01	F(163)=	-0.1655721954218865D 01	F(219)=	-0.1669244409453071D 01
F(52)=	-0.9136763118403064D 00	F(108)=	-0.1564217785718590D 01	F(164)=	-0.1656262733031217D 01	F(220)=	-0.1669321028315695D 01
F(53)=	-0.9415515085660616D 00	F(109)=	-0.1567893650854537D 01	F(165)=	-0.1656785359920642D 01	F(221)=	-0.1669395021426398D 01
F(54)=	-0.9665440355988495D 00	F(110)=	-0.1571443588922517D 01	F(166)=	-0.1657287215146875D 01	F(222)=	-0.1669466478771939D 01
F(55)=	-0.9906335062231936D 00	F(111)=	-0.1574717186978348D 01	F(167)=	-0.1657773610650956D 01	F(223)=	-0.1669535482725231D 01
F(56)=	-0.1013348735355958D 01	F(112)=	-0.1578102662659173D 01	F(168)=	-0.1658243337233130D 01	F(224)=	-0.1669602130791346D 01

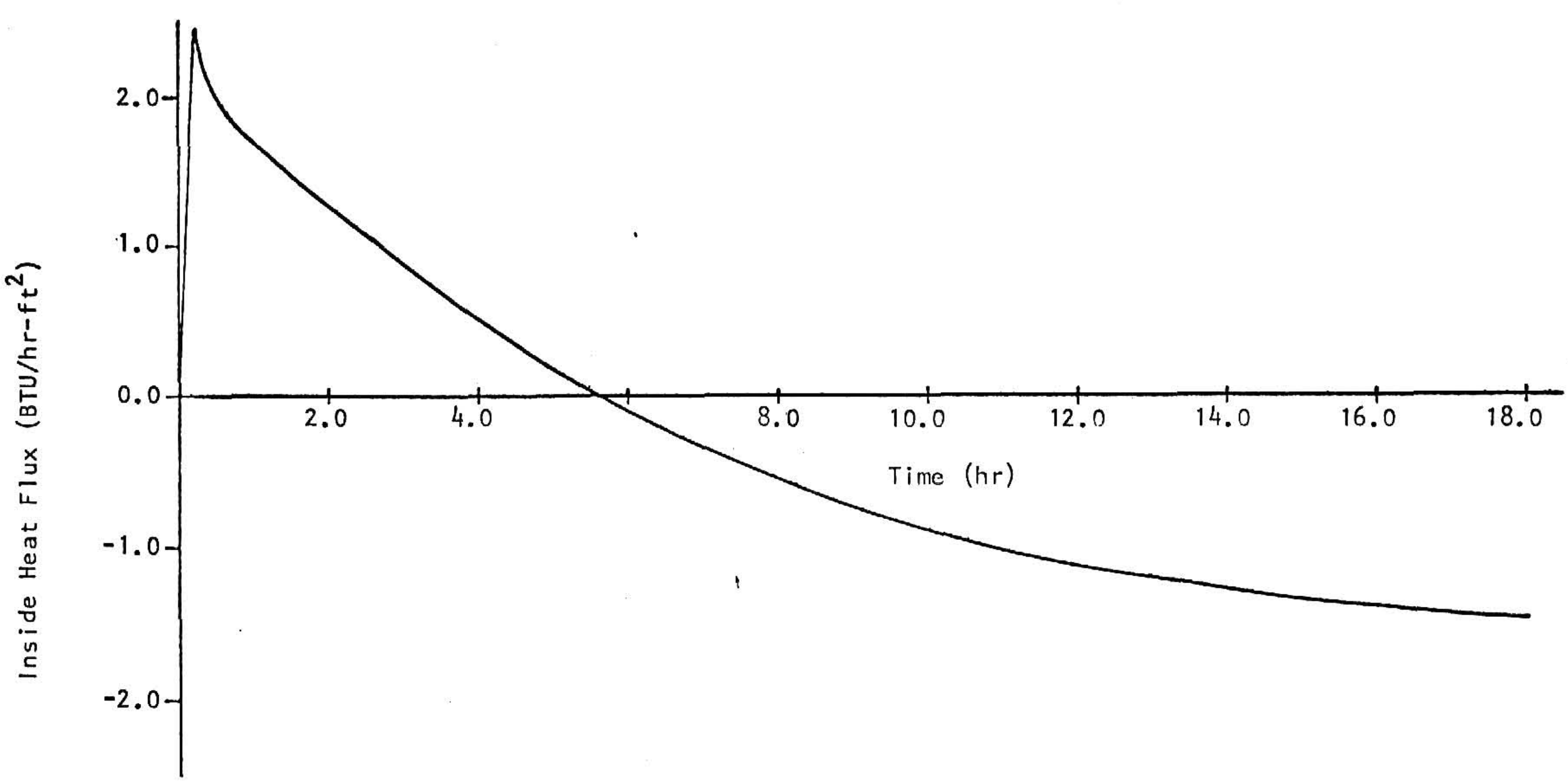

Figure 8. Heat Flux Response of Double-Layer Wall.

Table 10. Heat Flux for 3-Layer Wall.

F(1)= 0.0000000000000000D 00	F(57)=-0.9668052856749166D 00	F(113)=-0.9681538599652008D 00	F(169)=-0.9681543657658547D 00
F(2)= 0.1120793071780950D 01	F(58)=-0.9669825456933146D 00	F(114)=-0.9681539264488221D 00	F(170)=-0.9681543657907934D 00
F(3)= 0.1557979547328655D 01	F(59)=-0.9671365149065565D 00	F(115)=-0.9681539841965581D 00	F(171)=-0.9681543658124459D 00
F(4)= 0.1481466369119585D 01	F(60)=-0.9672702535846122D 00	F(116)=-0.9681540343573907D 00	F(172)=-0.9681543658312630D 00
F(5)= 0.1132639343855699D 01	F(61)=-0.9673864198536026D 00	F(117)=-0.9681540779270917D 00	F(173)=-0.9681543658476045D 00
F(6)= 0.8431186316601843D 00	F(62)=-0.9674873227310350D 00	F(118)=-0.9681541157720373D 00	F(174)=-0.9681543658617737D 00
F(7)= 0.6007561343529406D 00	F(63)=-0.9675749676194117D 00	F(119)=-0.9681541486444205D 00	F(175)=-0.9681543658741230D 00
F(8)= 0.3997575595696971D 00	F(64)=-0.9676510965635483D 00	F(120)=-0.9681541771976014D 00	F(176)=-0.9681543658848170D 00
F(9)= 0.2096931132745951D 00	F(65)=-0.9677172226850326D 00	F(121)=-0.9681542015993341D 00	F(177)=-0.9681543658941339D 00
F(10)= 0.5331130325023911D-01	F(66)=-0.9677746602871018D 00	F(122)=-0.9681542235419616D 00	F(178)=-0.9681543659022186D 00
F(11)=-0.2247120347762867D-01	F(67)=-0.9678245535827723D 00	F(123)=-0.9681542427540229D 00	F(179)=-0.9681543659092368D 00
F(12)=-0.1957784533445472D 00	F(68)=-0.9678678863844866D 00	F(124)=-0.9681542585075536D 00	F(180)=-0.9681543659153328D 00
F(13)=-0.3014187409754510D 00	F(69)=-0.9679055278913302D 00	F(125)=-0.9681542722[illegible]D 00	F(181)=-0.9681543659206278D 00
F(14)=-0.3995113836544550D 00	F(70)=-0.9679382234183112D 00	F(126)=-0.9681542854841590D 00	F(182)=-0.9681543659252274D 00
F(15)=-0.4759914532896221D 00	F(71)=-0.9679666230471152D 00	F(127)=-0.9681542955400842D 00	F(183)=-0.9681543659292224D 00
F(16)=-0.5321378071566411D 00	F(72)=-0.9679912911612643D 00	F(128)=-0.9681543047921973D 00	F(184)=-0.9681543659326925D 00
F(17)=-0.5856029172347157D 00	F(73)=-0.9680127130561674D 00	F(129)=-0.9681543128286472D 00	F(185)=-0.9681543659357067D 00
F(18)=-0.6394763072247557D 00	F(74)=-0.9680313296056798D 00	F(130)=-0.9681543196051632D 00	F(186)=-0.9681543659383248D 00
F(19)=-0.6927532375035861D 00	F(75)=-0.9680474957267387D 00	F(131)=-0.9681543258724880D 00	F(187)=-0.9681543659405992D 00
F(20)=-0.7303552413387106D 00	F(76)=-0.9680615377317276D 00	F(132)=-0.9681543311913390D 00	F(188)=-0.9681543659425747D 00
F(21)=-0.7529371306862307D 00	F(77)=-0.9680737347147568D 00	F(133)=-0.9681543357137794D 00	F(189)=-0.9681543659442907D 00
F(22)=-0.7812504350671512D 00	F(78)=-0.9680843290090352D 00	F(134)=-0.9681543396873602D 00	F(190)=-0.9681543659457810D 00
F(23)=-0.8056335633931190D 00	F(79)=-0.9680935531454816D 00	F(135)=-0.9681543431388175D 00	F(191)=-0.9681543659470752D 00
F(24)=-0.8271717307516603D 00	F(80)=-0.9681015246650914D 00	F(136)=-0.9681543461367305D 00	F(192)=-0.9681543659481754D 00
F(25)=-0.8457154630977341D 00	F(81)=-0.9681084676603552D 00	F(137)=-0.9681543487408467D 00	F(193)=-0.9681543659491758D 00
F(26)=-0.8618129933916445D 00	F(82)=-0.9681144908377013D 00	F(138)=-0.9681543510027490D 00	F(194)=-0.9681543659500242D 00
F(27)=-0.8757923114461770D 00	F(83)=-0.9681197366514572D 00	F(139)=-0.9681543529672646D 00	F(195)=-0.9681543659507636D 00
F(28)=-0.8879329557701076D 00	F(84)=-0.9681242867427507D 00	F(140)=-0.9681543546740107D 00	F(196)=-0.9681543659514060D 00
F(29)=-0.8984770274245491D 00	F(85)=-0.9681282385420170D 00	F(141)=-0.9681543561563371D 00	F(197)=-0.9681543659519565D 00
F(30)=-0.9076346903384300D 00	F(86)=-0.9681316718457958D 00	F(142)=-0.9681543574438962D 00	F(198)=-0.9681543659524352D 00
F(31)=-0.9155193397587397D 00	F(87)=-0.9681346536973771D 00	F(143)=-0.9681543585622789D 00	F(199)=-0.9681543659528587D 00
F(32)=-0.9224906511303362D 00	F(88)=-0.9681372437506336D 00	F(144)=-0.9681543595337139D 00	F(200)=-0.9681543659532230D 00
F(33)=-0.9284306586179353D 00	F(89)=-0.9681394934854750D 00	F(145)=-0.9681543603772560D 00	F(201)=-0.9681543659535394D 00
F(34)=-0.9335703126927049D 00	F(90)=-0.9681414476280270D 00	F(146)=-0.9681543611104352D 00	F(202)=-0.9681543659538146D 00
F(35)=-0.9382345320374833D 00	F(91)=-0.9681431450063715D 00	F(147)=-0.9681543617470692D 00	F(203)=-0.9681543659540533D 00
F(36)=-0.9421661423160280D 00	F(92)=-0.9681446193610133D 00	F(148)=-0.9681543623000373D 00	F(204)=-0.9681543659542606D 00
F(37)=-0.9455810596261822D 00	F(93)=-0.9681458999557154D 00	F(149)=-0.9681543627803571D 00	F(205)=-0.9681543659544408D 00
F(38)=-0.9485477227714863D 00	F(94)=-0.9681470123639294D 00	F(150)=-0.9681543631975662D 00	F(206)=-0.9681543659545973D 00
F(39)=-0.9511236017581198D 00	F(95)=-0.9681479785747016D 00	F(151)=-0.9681543635599571D 00	F(207)=-0.9681543659547331D 00
F(40)=-0.9533614210559442D 00	F(96)=-0.9681488178321072D 00	F(152)=-0.9681543638747322D 00	F(208)=-0.9681543659548512D 00
F(41)=-0.9553351798149778D 00	F(97)=-0.9681495468016921D 00	F(153)=-0.9681543641481480D 00	F(209)=-0.9681543659549536D 00
F(42)=-0.9569935229383778D 00	F(98)=-0.9681501800182065D 00	F(154)=-0.9681543643856337D 00	F(210)=-0.9681543659550426D 00
F(43)=-0.9584600156733627D 00	F(99)=-0.9681507300212530D 00	F(155)=-0.9681543645919248D 00	F(211)=-0.9681543659551159D 00
F(44)=-0.9597331149418111D 00	F(100)=-0.9681512077577254D 00	F(156)=-0.9681543647711084D 00	F(212)=-0.9681543659551870D 00
F(45)=-0.9608407322262456D 00	F(101)=-0.9681516227729551D 00	F(157)=-0.9681543649267448D 00	F(213)=-0.9681543659552456D 00
F(46)=-0.9618012713315712D 00	F(102)=-0.9681519831646325D 00	F(158)=-0.9681543650619334D 00	F(214)=-0.9681543659552963D 00
F(47)=-0.9626360337409978D 00	F(103)=-0.9681522962467462D 00	F(159)=-0.9681543651793554D 00	F(215)=-0.9681543659553405D 00
F(48)=-0.9633511110887650D 00	F(104)=-0.9681525681921242D 00	F(160)=-0.9681543652813560D 00	F(216)=-0.9681543659553789D 00
F(49)=-0.9639909163876667D 00	F(105)=-0.9681528044057374D 00	F(161)=-0.9681543653699512D 00	F(217)=-0.9681543659554120D 00
F(50)=-0.9645379682630443D 00	F(106)=-0.9681530095825255D 00	F(162)=-0.9681543654469058D 00	F(218)=-0.9681543659554408D 00
F(51)=-0.9650131140327557D 00	F(107)=-0.9681531878005122D 00	F(163)=-0.9681543655137451D 00	F(219)=-0.9681543659554659D 00
F(52)=-0.9654253773686315D 00	F(108)=-0.9681533426018952D 00	F(164)=-0.9681543655718198D 00	F(220)=-0.9681543659554877D 00
F(53)=-0.9657344393115662D 00	F(109)=-0.9681534770634538D 00	F(165)=-0.9681543656222418D 00	F(221)=-0.9681543659555067D 00
F(54)=-0.9660357832863320D 00	F(110)=-0.9681535938576995D 00	F(166)=-0.9681543656660472D 00	F(222)=-0.9681543659555234D 00
F(55)=-0.9663662674515552D 00	F(111)=-0.9681536953055941D 00	F(167)=-0.9681543657040970D 00	F(223)=-0.9681543659555376D 00
F(56)=-0.9666012116492816D 00	F(112)=-0.9681537834246991D 00	F(168)=-0.9681543657371470D 00	F(224)=-0.9681543659555502D 00

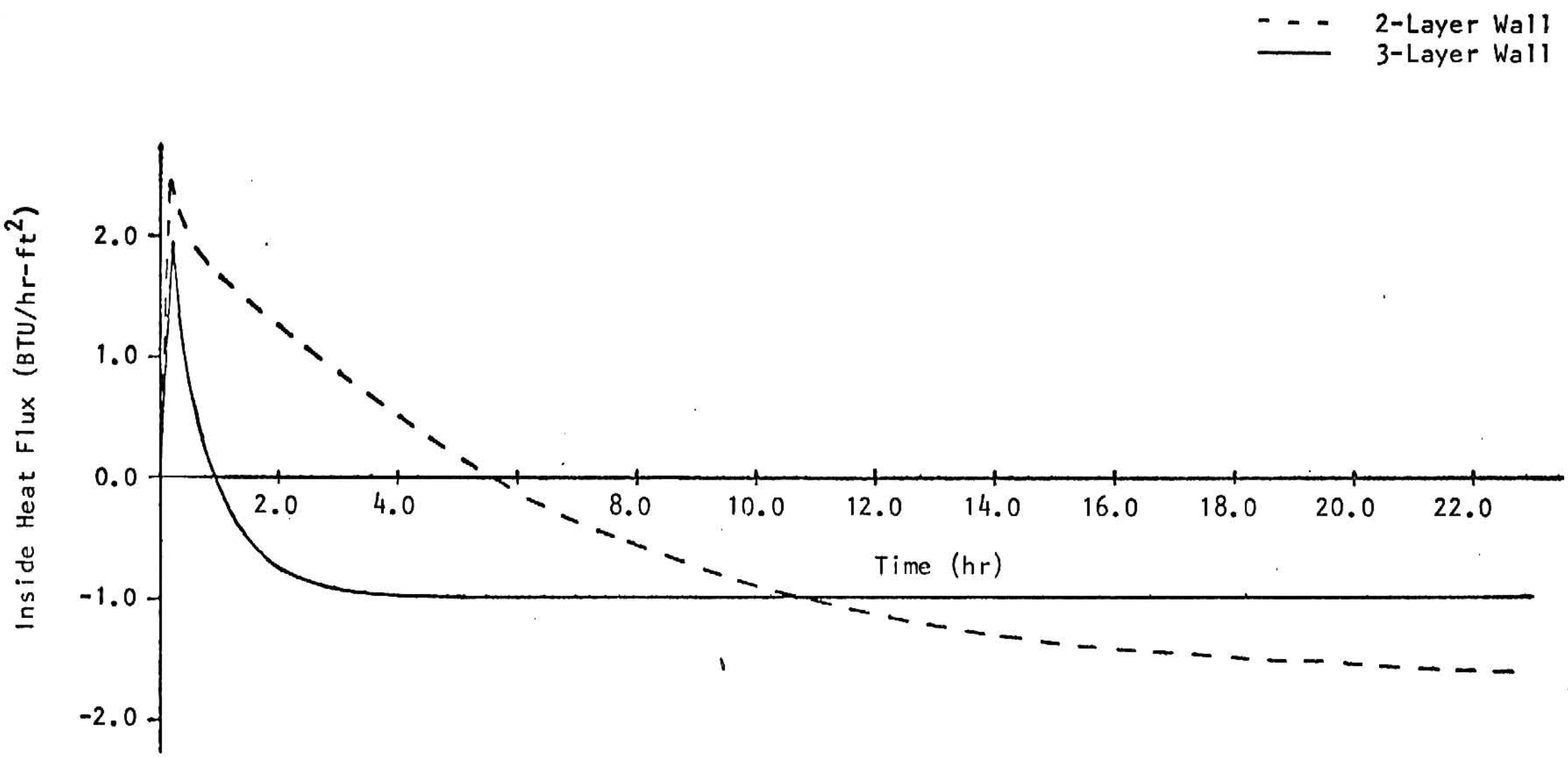

Figure 9. Heat Fluxes of 2-Layer and 3-Layer Walls.

Temperature Control System

As a final application the heat conduction transfer functions will be used in the model of the temperature control system illustrated in Figure 10.

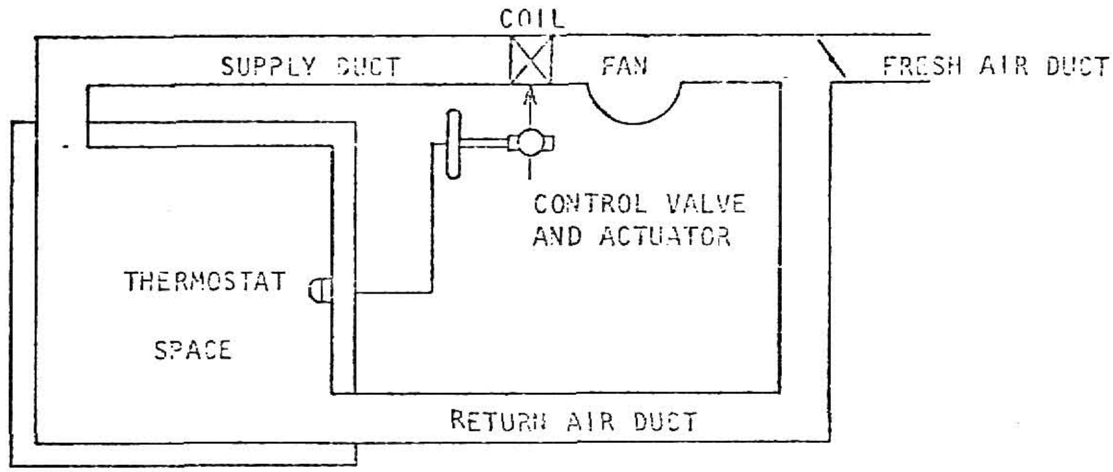

Figure 10. Air Conditioning Control System.

The transfer functions are used to simulate the walls while the rest of the model is developed in Appendix F. The time delays considered in this demonstration are the thermal lag at the coil, the space lag, and the lag of the walls. Other time delays are considered negligible. A simplified block diagram of the system is shown in Figure 11. The heat flow deviation into the space, F_I, and the space temperature deviation, V_R, are the desired responses. To obtain these values, a Runge-Kutta numerical integration technique is employed [13]. The heat flux at the wall is simultaneously calculated with the heat conduction transfer function algorithm. Steady-state values are used to initialize the variables of the control system. The system is excited by changing some parameter, most probably the thermostatic setpoint or the outside temperature. Subroutine FCALC is

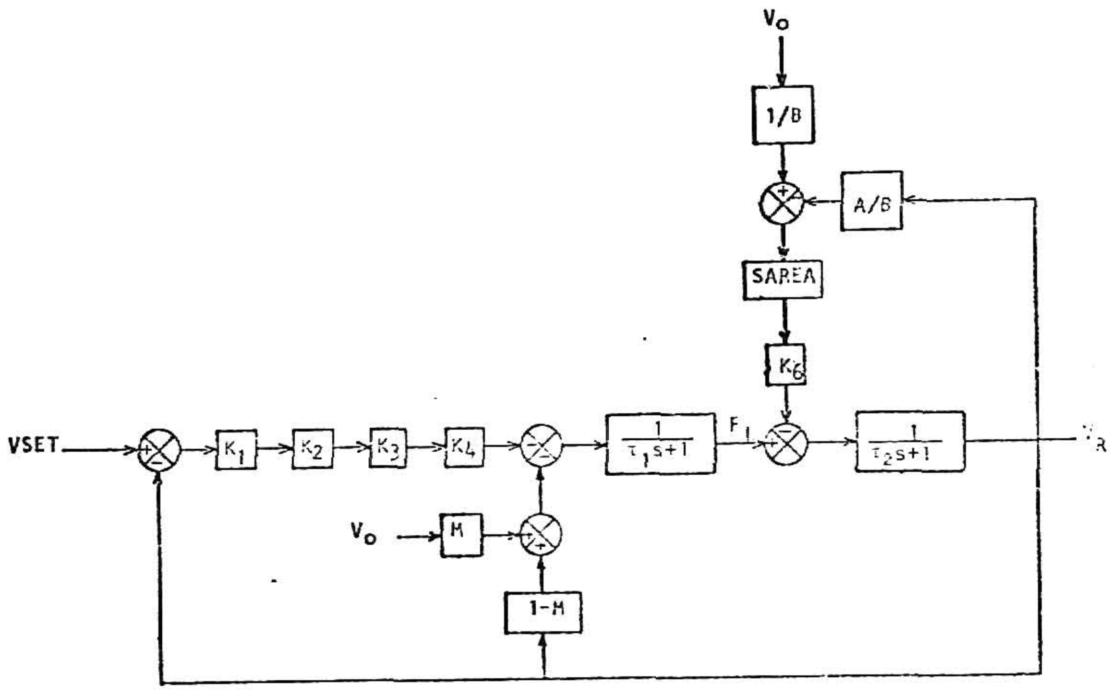

Figure 11. Air Conditioning Control System Block Diagram.

executed to derive an inside heat flux. This value is used by the subroutine FCT to calculate values for the derivatives of the variables being integrated during a finite time stepsize which is specified by the user. A new heat flux is calculated at the beginning of each step to be used for the calculation of the derivatives during that step. The time stepsizes may be split into an integral number of smaller stepsizes. The Runge-Kutta integration is then operated over these smaller stepsizes for increased stability. Stepsize depends on the system constants and the input. For small time constants and rapidly changing input, small stepsizes for the calculation of the heat flux are necessary. These steps may be broken into four to ten smaller steps for the integration technique. Larger increments may be used for less stringent systems and inputs. Appendix G contains a listing of the computer simulation for the above system.

Constants were derived according to Appendix F and references [14] and [15]. The constants used for the following examples are for a 9' x 12' x 20' room with one exterior wall. Values of the constants used are shown below. Constant values may vary widely, but care should be exercised as the Runge-Kutta integration scheme may become unstable for some sets of constants, as previously discussed. The walls are comprised of the materials as described in Table 3.

Table 11. Air Conditioning Control System Constants.

Description	Constant	Value
Thermostat gain	K_1	6.5 psi/°F
Actuator gain	K_2	0.15 in/psi
Valve gain	K_3	2481.75 BTU/hr/in
Coil gain	K_4	2.273×10^{-3} °F-hr/BTU
Gain of room	K_6	2.273×10^{-3} °F-hr/BTU
Make-up air ratio	M	0.2 ft^3/ft^3
Exterior wall surface area	SAREA	250.0 ft^2
Time constant of coil	τ_1	0.022 hr
Time constant of room	τ_2	0.08 hr

It is most convenient to start the simulation with steady-state values and then allow some parameter, R, the set point, or θ_o, the outside temperature, to deviate from its initial value. Steady-state values may be obtained as follows.

The differential equations describing the system in Figure 11 may be written as

$$\frac{df_I}{dt} = [(1-M)V_R + MV_o + CON(VSET - V_R) - f_I]/\tau_1 \quad \text{and} \quad \frac{dV_R}{dt} = (f_I - k_6F_o - V_R)/\tau_2 \ .$$

f_I - heat flow deviation into the air space supplied from the inlet duct

CON - $K_1K_2K_3K_4$

In the steady-state, the derivatives are equal to zero and the differential equations reduce to

$$f_I = (1\text{-}M)V_R + MV_o + CON(VSET - V_R) \quad \text{and} \quad f_I = k_6F_o + V_R \ .$$

It is also known that in the steady-state

$$f_o = u(\theta_R - \theta_o) \ , \quad \text{where}$$

u - overall thermal conductance of the wall.

The overall thermal conductance may be calculated for a multi-layer structure and surface resistances with the equation

$$u = \frac{1}{R_1 + \frac{\Delta x_1}{k_1} + \frac{\Delta x_2}{k_2} + \ldots + \frac{\Delta x_n}{k_n} + R_n} \ . \qquad [8]$$

Once values of the constants have been selected, steady-state values of f_I, θ_R, and f_o may be calculated.

For the first example, the simulation was started from the steady-state condition where the inside temperature, outside temperature, and set point all equal 70°F. The thermostat set point is then increased to 75°F. Figure 12 is a plot of the room temperature deviation from the initial 70°F. It may be observed that the room temperature increases quickly, falls slightly, and then very gradually increases to a new steady-state condition.

As a final example, the system was started from the same steady-state at 70°F. This time the outside temperature was reduced to 50°F according to Figure 13 and held constant. Figure 14 is a plot of f_I, the heat input to the space, and f_o, the heat loss through the walls. Both responses have

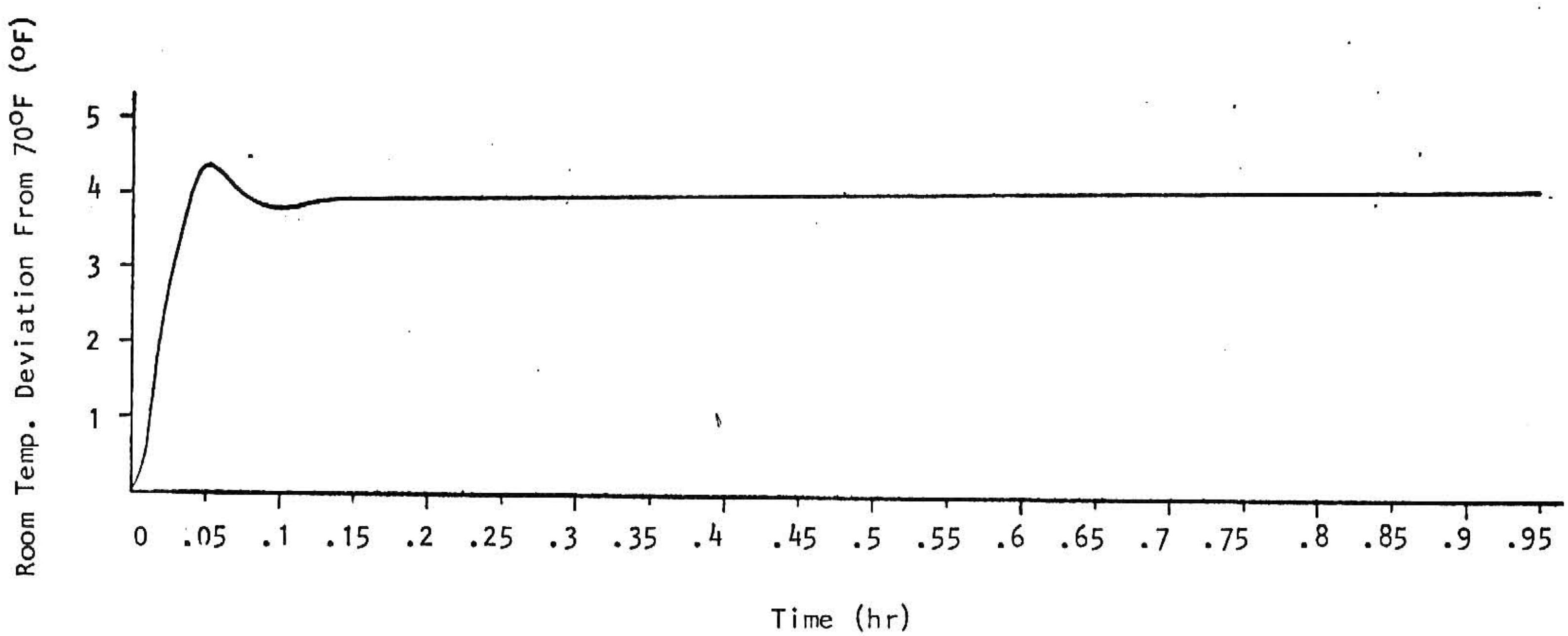

Figure 12. Room Temperature Deviation for 75°F Setpoint.

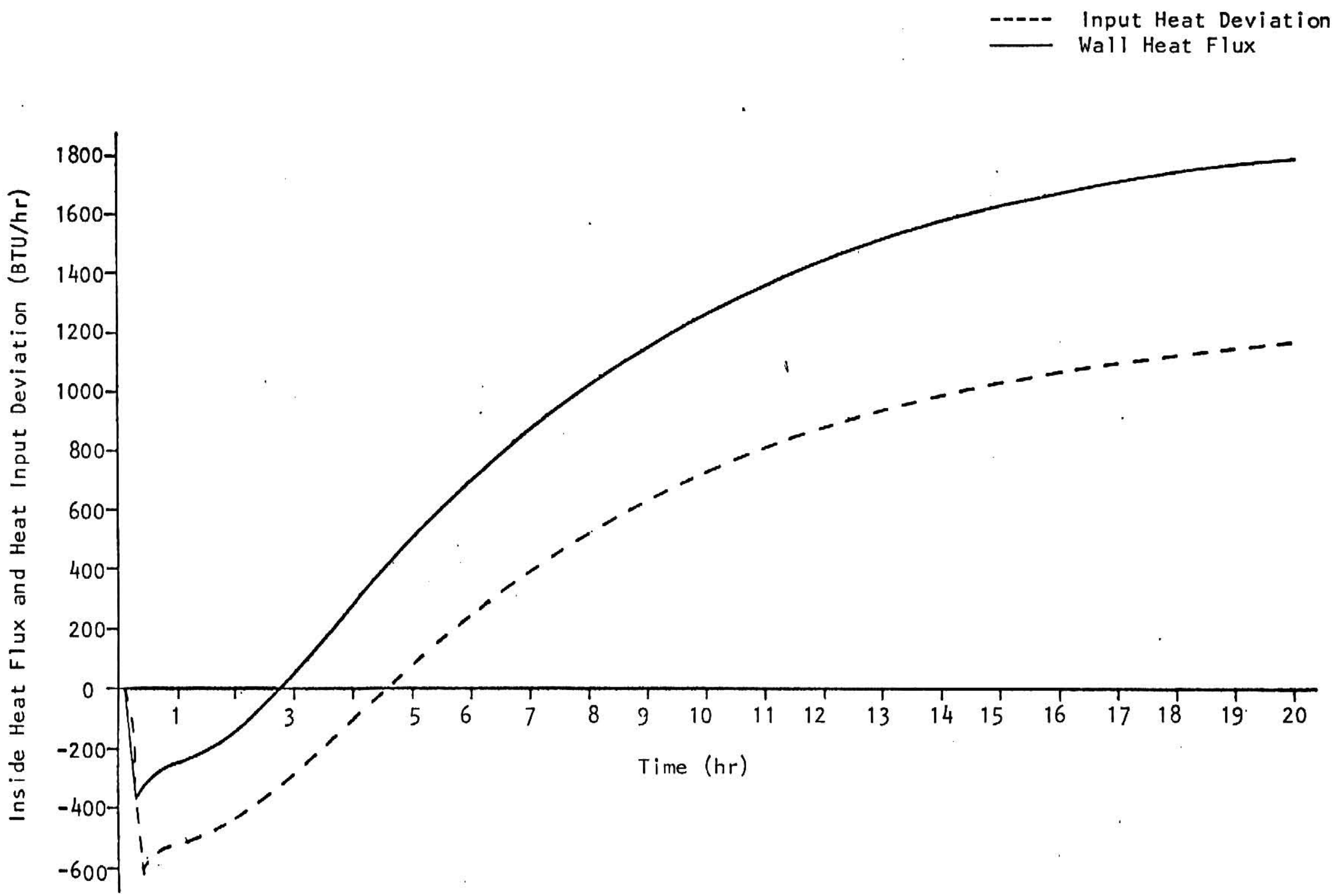

Figure 14. Inside Heat Flux and Heat Input Deviation for 50°F Outside Temperature.

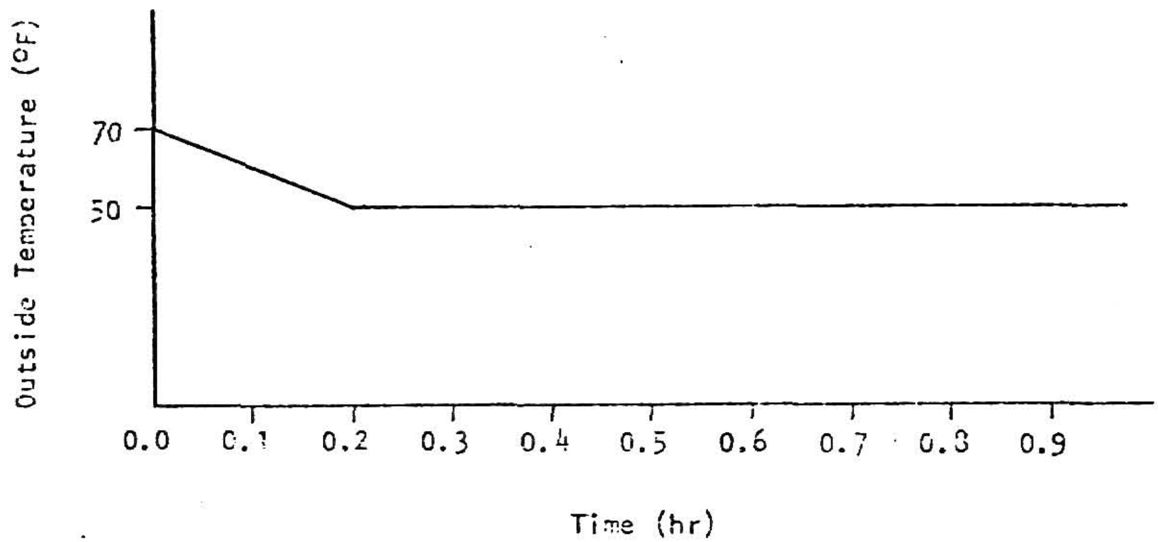

Figure 13. Outside Temperature Profile.

the same general form, and both immediately fall to negative values before reaching positive steady-state values. The heat input to the room is initially negative because the air being supplied to the rooms consists partly of 50°F outside air as well as return air. This air mixture is not as warm as the air being removed resulting in a cooler inside space temperature. The walls are initially at 70°F and begin to transfer stored heat to the cooler air resulting in negative values of f_o. Eventually, the stored heat in the walls is exhausted and heat begins to flow from the room to the walls. Meanwhile, the drop in room temperature is sensed and the heating system is activated resulting in positive values of f_I. The inside temperature deviation from the initial 70°F is plotted in Figure 15. The temperature immediately drops because of the cooler mixed air pouring into the room, and shows a slight increase as the coil starts to heat the cool air. Eventually, the temperature levels to a lower steady-state

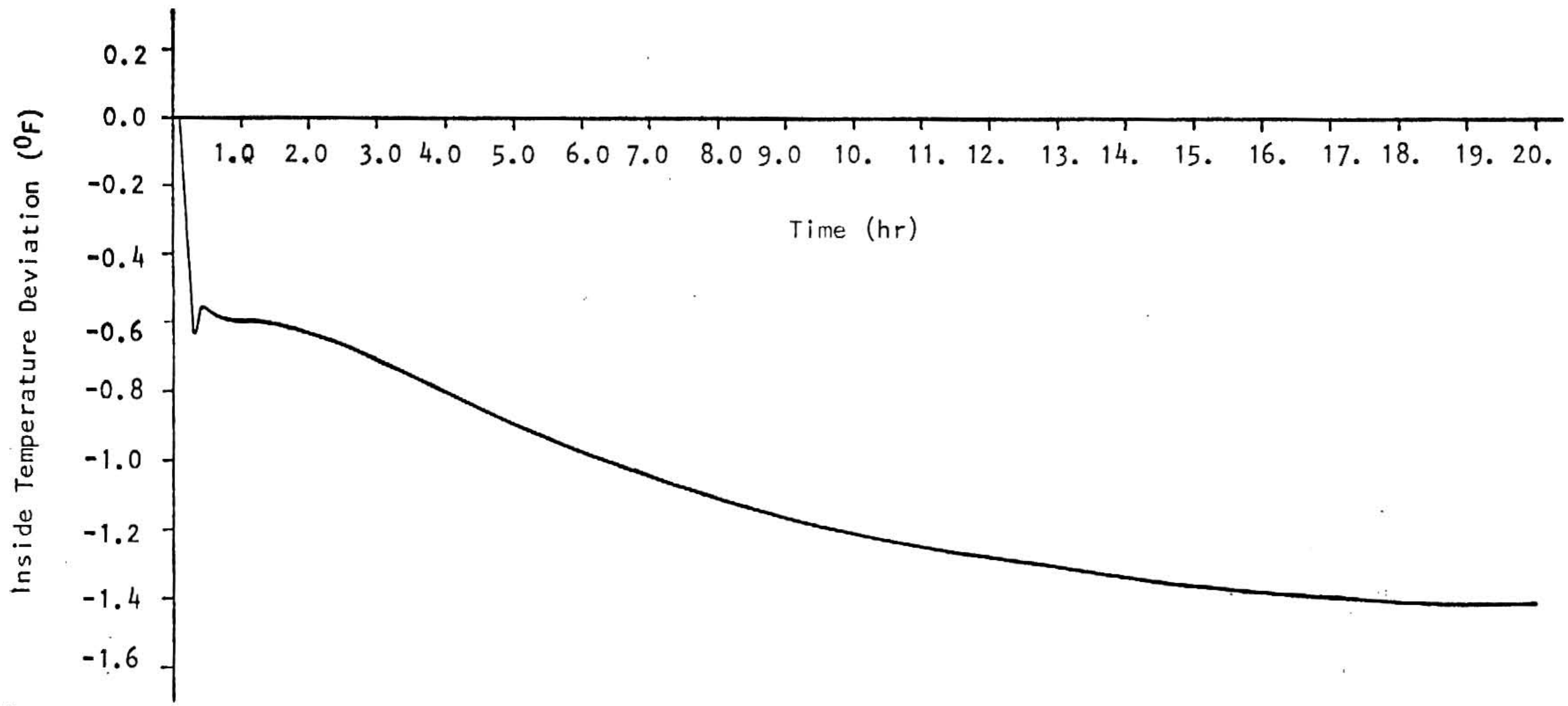

Figure 15. Room Temperature Deviation for 50°F Outside Temperature.

temperature, demonstrating approximately 1.5°F offset. The heat flow into the space, the heat loss through the walls, and the space temperature all approach new steady-state values.

These applications have been included to demonstrate the usefulness of the heat conduction algorithm. Constant input temperatures as employed in the examples are not necessary. Practically all input forms could be accurately approximated by discrete values with small time increments.

CHAPTER V

CONCLUSION

An algorithm for the calculation of heat conduction in multi-layer structures has been derived, calculated, and demonstrated in this thesis. This algorithm may be applied to walls of any number of layers and may include thermal resistances at the inner and outer surfaces of the wall. The algorithm has a variable time increment and has been tested for increments as small as 0.025 hr. Input to the algorithm is discrete values of the inside and outside surface temperatures at multiples of the time increment. Accuracy may be specified by the user and depends primarily upon truncation errors. The algorithm's strongest point is speed. For time stepsizes of reasonable size, 0.1 hr - 1.0 hr., the algorithm requires little computing time and is therefore inexpensive to use. The algorithm is well suited for applications in which a wall or set of walls are to be studied a number of times. This advantage makes it recommendable for use in the modeling of buildings and simulation of control systems. The algorithm is easily applied to the analysis of dynamic control systems to obtain system responses quickly and cheaply, where the time stepsize is quite small. The stepsize may be adjusted according to the dynamics of the system. For instance, a system with large time constants could be analyzed with a larger stepsize than one with short time constants. As smaller stepsizes are required, the number of z-coefficients must be increased to keep truncation errors to a minimum. If too few coefficients are employed or too large a stepsize is used, results may be erratic. As the time stepsize is decreased, more roots are necessary to calculate the z-coefficients to sustain good accuracy. The larger sets of z-coefficients for very small stepsizes require extra computing time, and thus the algorithm's advantages diminish. It has been found that stepsizes smaller than 0.025 hr may produce unreasonable results with too few roots, and the computing time required makes other techniques more enticing. However, for many practical problems stepsizes smaller than 0.1 hr are unnecessary and so the heat conduction transfer function algorithm may be quite advantageous.

LIST OF REFERENCES

1. "Controls Seen as Quickest Energy Retrofit Payback," Air Cond., Heating, and Refrigeration News, February, 1977.

2. Mitalas, G. P., and D. G. Stephenson, "Calculation of Heat Conduction Transfer Functions for Multi-Layer Slabs," ASHRAE Transactions, Vol. 77, Part II, 1971.

3. Nessi, A. and L. Nisolle, "Régimes variables de fonctionnement dans les installations de chauffage central," DUNOD, 1925; and "Résoltuion pratique des problèmes de discontinuité de fonctionnement dans les installations de chauffage central," DUNOD, 1933.

4. Mackey, C. O. and L. T. Wright, "Periodic Heat Flow - Composite Walls or Roofs," ASHRAE Transactions, Vol. 52, 1946.

5. Kusuda, T., "Thermal Response Factors for Multi-Layer Structures of Various Heat Conduction Systems," ASHRAE Transactions, Vo.. 75, Part I, 1969.

6. Mitalas, G. P., and J. G. Arseneault, Fortran IV Program to Calculate Heat Flux Response Factors for Multi-Layer Slabs, National Research Council of Canada, Division of Building Research, 1967.

7. Schenck, Hilbert J., Fortran Methods in Heat Flow, The Ronald Press Company, New York, New York, 1963.

8. Holman, J. P., Heat Transfer, 3rd ed., McGraw-Hill, New York, 1972.

9. Dorf, Richard C. Modern Control Systems, 2nd ed., Addison-Wesley, Reading, Mass., 1974.

10. Churchill, R. V., Fourier Series and Boundary Value Problems, 2nd ed., McGraw-Hill, New York, 1963.

11. Carslaw, H. S., and J. C. Jaeger, Conduction of Heat in Solids, 2nd ed., Oxford University Press, Ely House, London, 1973.

12. Ogata, Katsuhiko, Modern Control Engineering, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1970.

13. Ralston, A. and H. S. Wilf, Mathematical Methods for Digital Computers, Wiley, New York, London, 1960.

14. Shih, H. Y., "Control Systems and Principals," Powers Regulator Company, Northbrook, Ill., 1970.

15. Pearson, J. T., Leonard, R. G., and McCutchan, R. D., "Gain and Time Constant for Finned Serpentine Crossflow Heat Exchangers," ASHRAE Transactions, Vol. 80, Part II, 1974.

APPENDIX A - ROOTS OF B=0.

An indication as to the position of the roots of the function B may be gained by observing the stability of a multi-layer walls response. Obviously, a wall displays extremely stable characteristics. Therefore, it may be deduced that there are no roots in the right-half of the s-plane [12].

The possible root locations may be narrowed further by algebraically investigating the function B. B is composed of a series of sinh and cosh terms and depends on the number of layers in the wall. Initially, the values of the individual terms of the 2x2 matrices composing B were examined by allowing s to be zero, a positive real, complex, and a negative real variable. A function B was then expanded for a multi-layer wall and s was again allowed to be zero, positive real, etc. Cancellation of terms due to sign differences was also probed.

For a two-layer wall the hyperbolic function B is

$$B = -\frac{\sqrt{\alpha_2}}{k_2\sqrt{s}}\cosh\left(\frac{\Delta x_1}{\sqrt{\alpha_1}}\sqrt{s}\right)\sinh\left(\frac{\Delta x_2}{\sqrt{\alpha_2}}\sqrt{s}\right) - \frac{\sqrt{\alpha_1}}{k_1\sqrt{s}}\cosh\left(\frac{\Delta x_2}{\sqrt{\alpha_2}}\sqrt{s}\right)\sinh\left(\frac{\Delta x_1}{\sqrt{\alpha_1}}\sqrt{s}\right) .$$

And, when B=0, to find the roots, the following equation is found.

$$\frac{k_1}{k_2}\tanh\left(\frac{\Delta x_2}{\sqrt{\alpha_2}}\sqrt{s}\right) + \frac{\alpha_1}{\alpha_2}\tanh\left(\frac{\Delta x_1}{\sqrt{\alpha_1}}\sqrt{s}\right) = 0$$

If s were some complex variable, the square root of s is also a complex variable which will be called c+id. Then

$$\frac{k_1}{k_2}\tanh\left[\frac{\Delta x_2}{\sqrt{\alpha_2}}(c+id)\right] + \sqrt{\frac{\alpha_1}{\alpha_2}}\tanh\left[\frac{\Delta x_1}{\sqrt{\alpha_1}}(c+id)\right] = 0 .$$

From this, it may be shown that values of c and d cannot be found so that s could be a complex variable. The other variable cases were examined in like manner. It was concluded that the function B has an infinite set of roots all of which lie on the negative real axis in the s-plane.

APPENDIX B - EQUIVALENT Z-TRANSFORMATION TECHNIQUES

In this Appendix it will be demonstrated that the output of a simple transfer function, G(s), with a triangular pulse input is equivalent to the output of a transfer function developed from G(s) by assuming a ramp input and using a weighted impulse input instead.

A triangular pulse input, v, is pictured below.

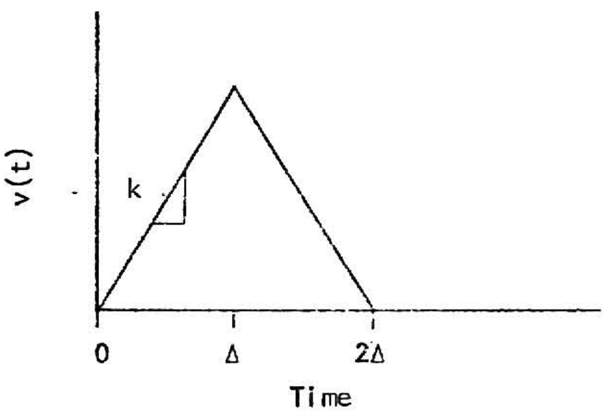

Figure 16. Triangular Pulse.

$$v(t) = \begin{cases} kt & \text{for} \quad 0 \leq t < \Delta \\ k\Delta - k(t - \Delta) & \text{for} \quad \Delta \leq t < 2\Delta \\ 0 & \text{for} \quad t \geq 2\Delta \end{cases}$$

The Laplace transform of the triangular pulse, V, is found to be

$$V = k\left(\frac{e^{-s\Delta} - 1}{s}\right)^2 .$$

For this demonstration, the transfer function is

$$G(s) = \frac{2}{(s+1)(s+2)} .$$

The Laplace transform of the output, O, is found to be

$$O = G(s) \cdot V = \frac{2k(e^{-s\Delta} - 1)^2}{s^2(s+1)(s+2)} .$$

Expanding,

$$0 = \frac{2ke^{-2s\Delta}}{s^2(s+1)(s+2)} - \frac{4ke^{-s\Delta}}{s^2(s+1)(s+2)} + \frac{2k}{s^2(s+1)(s+2)} \ . \tag{31}$$

Each of these three terms have the term

$$\frac{2k}{s^2(s+1)(s+2)}$$

in common. This term may be expressed in partial fractions as

$$\frac{2k}{s^2(s+1)(s+2)} = \frac{k}{s^2} - \frac{3k}{2s} + \frac{sk}{(s+1)} - \frac{k}{2(s+2)} \ .$$

This expansion aids in finding the z-transform of 0, the output. The z-transform of each of the three terms in equation (31) is then found. The equality may then be manipulated to the form

$$0(z) = \frac{k(z-1)^2}{z^2} [\frac{\Delta z}{(z-1)^2} - \frac{3z}{2(z-1)} + \frac{2z}{z - e^{-\Delta}} - \frac{z}{2(z - e^{-2\Delta})}] \ , \tag{32}$$

the z-transform of the output.

Now assume the input to be a ramp. The Laplace transform of the output is then

$$0 = \frac{1}{s^2} G(s) = \frac{2}{s^2(s+1)(s+2)}$$

This equation may now be expressed in partial fraction form as

$$0 = \frac{1}{s^2} - \frac{3}{2s} + \frac{2}{(s+1)} + \frac{1}{2(s+2)} \ .$$

The z-transform of this is

$$0(z) = \frac{\Delta z}{(z-1)^2} - \frac{3z}{2(z-1)} + \frac{2z}{z - e^{-\Delta}} - \frac{z}{2(z - e^{-2\Delta})} \ .$$

R(z), the z-transfer function, may be obtained by dividing this output by the z-transform of the ramp input. The z-transform of a ramp, I(z) is

$$I(z) = \frac{\Delta z}{(z-1)^2} \ .$$

It follows that

$$R(z) = \frac{O(z)}{I(z)} = \frac{(z-1)^2}{\Delta z} \left[\frac{\Delta z}{(z-1)^2} - \frac{3z}{2(z-1)} + \frac{2z}{z - e^{-\Delta}} - \frac{z}{2(z - e^{-2\Delta})}\right] \ .$$

This transfer function is now multiplied by a weighted impulse input. The impulse is weighted by a constant such that the magnitude of the impulse is equal to the height of the triangular pulse input displayed in Figure 14. The impulse also occurs at the same point in time, Δ, that the triangular pulse reaches its peak. Below, the impulse is drawn with a solid line while the triangular pulse is pictured with dashed lines.

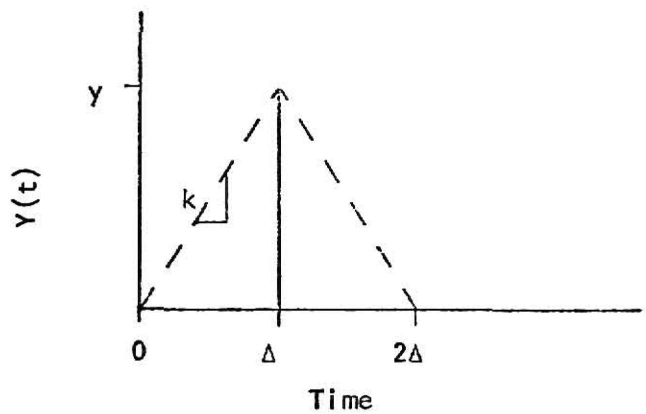

Figure 17. Weighted Impulse.

The z-transform of this weighted impulse input, $Y(z)$, is

$$Y(z) = Z[y\, \delta(t - \Delta)] = yz^{-1} \ .$$

"y" is the magnitude of the constant that "weights" the impulse. In terms of the slope of the dashed input it is found that

$$y = k\Delta \ .$$

The z-transform of the output, $O(z)$, from a transfer function developed with a ramp due to a weighted impulse input is

$$O(z) = R(z) \cdot Y(z) = \frac{k(z-1)^2}{z^2} \left[\frac{\Delta z}{(z-1)^2} - \frac{3z}{2(z-1)} + \frac{2z}{z - e^{-\Delta}} - \frac{z}{2(z - e^{-2\Delta})}\right] .$$

It may be seen that this is the same output as found in equation (32). Thus, it has been demonstrated that equivalent outputs may be obtained by 1) using a triangular pulse input, or 2) finding a transfer function with a ramp input and then using a weighted unit impulse as an actual input.

APPENDIX C - ROOTFINDER PROGRAM.

```
$JOB              TDH,TIME=(0,30),PUNCH,NOLIST
      IMPLICIT REAL*8(A-H,O-Z)
C***********************************************************************
C     THIS PROGRAM IS A ROOTFINDER.  IT LOCATES NEGATIVE REAL ROOTS OF
C     THE FUNCTION B FOR THE VARIABLE S.  IT DOES SO BY OBSERVING SIGN
C     CHANGES IN THE VALUE OF THE FUNCTION B AS THE VARIABLE S IS
C     DECREMENTED.
C***********************************************************************
    2 FORMAT('1',34X,1(65('_')))
    3 FORMAT(' ',33X,'|',9X,'ROOT',10X,'|',6X,'VALUE OF',6X,'|',4X,
     1'DERIVATIVE',6X,'|')
    4 FORMAT(' ',33X,'|',10X,'S',12X,'|',5X,'FUNCTION B',5X,'|',3X,'OF F
     1UNCTION B',4X,'|')
    5 FORMAT('+',34X,1(65('_')))
    6 FORMAT(' ',33X,'|',D23.16,'|',F20.16,'|',F20.16,'|')
   10 FORMAT(I5)
   11 FORMAT(F20.16)
   13 FORMAT('0','A ROOT HAS BEEN LOCATED AT S=',F10.4,5X,'B=',F10.4,
     15X,'IER=',I2)
   15 FORMAT('0','IT BLOWS UP AT X=',F10.6,5X,'B=',E15.4,5X,'DERF=',
     1E15.4,5X,'IER=',I2)
   16 FORMAT(D23.16)
      DIMENSION A(4),CON(4),DELX(4),H(2,2),HT(2,2),HTMX(2,2)
C***********************************************************************
C     N    - NUMBER OF LAYERS
C     R1   - INNER SURFACE RESISTANCE
C     RN   - OUTER SURFACE RESISTANCE
C     A    - THERMAL DIFFUSIVITY
C     CON  - SPECIFIC HEAT
C     DELX- LAYER THICKNESS
C***********************************************************************
      READ(5,10) N
      READ(5,11) R1
      READ(5,11) RN
      DO 20 M=1,N
   20 READ(5,11) A(M)
      DO 21 M=1,N
   21 READ(5,11) CON(M)
      DO 22 M=1,N
   22 READ(5,11) DELX(M)
      WRITE(6,2)
      WRITE(6,3)
      WRITE(6,4)
      WRITE(6,5)
C     THE VARIABLE S IS INITIALIZED TO THE FOLLOWING VALUE.
      S=0.0001D0
      S=DABS(S)
   34 CONTINUE
      CALL FIGURE(S,N,R1,A,CON,DELX,RN,F)
      IF (F)37,31,38
   37 CONTINUE
   30 S=S+0.01D0
      IF (S.GE.5.0001)GO TO 35
```

```
      CALL FIGURE(S,N,R1,A,CCN,DELX,RN,F)
      IF (F)30,31,33
   38 CONTINUE
   32 S=S+0.01D0
      IF (S.GE.5.0001)GO TO 35
      CALL FIGURE(S,N,R1,A,CCN,DELX,RN,F)
      IF (F)33,31,32
   33 CONTINUE
      XST=S
C     EPS IS THE VALUE OF THE MAXIMUM ALLOWABLE ERROR IN A ROOT.
      EPS=1.0D-14
      IEND=100
      CALL BETA(X,F,DERF,R1,A,CON,DELX,RN,XST,EPS,IEND,IER,N)
      IF (X)42,42,43
   42 WRITE(6,15)X,F,DERF,IER
      GO TO 36
   43 CONTINUE
      W=-X
      WRITE(6, 6) W,F,DERF
      PUNCH16,W
      S=S+0.01D0
      IF (S.LT.5.0001)GO TO 34
      GO TO 36
   31 WRITE(6,13) S,F,IER
      PUNCH16,S
      S=S+0.01D0
      IF (S.LT.5.0001)GO TO 34
      GO TO 36
   35 WRITE(6,5)
   36 CONTINUE
      WRITE(6,2)
      STOP
      END
      SUBROUTINE FIGURE(S,N,R1,A,CON,DELX,RN,F)
      IMPLICIT REAL*8(A-H,O-Z)
C*********************************************************************
C     SUBROUTINE FIGURE FINDS NUMERICAL VALUES FOR EACH ELEMENT OF THE
C     2X2 MATRIX, MATRIX (12), FOR A SPECIFIC VALUE OF S.  VALUES ARE
C     FOUND FOR ANY N LAYER STRUCTURE BY CALLING A MATRIX MULTIPLICATION
C     SUBROUTINE.
C*********************************************************************
      DIMENSION A(N),CCN(N),DELX(N),H(2,2),HT(2,2),HTMX(2,2)
      M=0
      H(1,1)= 1.D0
      H(1,2)= R1
      H(2,1)= 0.D0
      H(2,2)= 1.D0
   23 CONTINUE
      M=M+1
      HT(1,1)=DCOS(DSQRT(S/A(M))*DELX(M))
      HT(1,2)=(-1.D0/CON(M))*DSQRT(A(M)/S)*DSIN(DSQRT(S/A(M))*DELX(M))
      HT(2,1)=CCN(M)*DSQRT(S/A(M))*DSIN(DSQRT(S/A(M))*DELX(M))
      HT(2,2)=DCOS(DSQRT(S/A(M))*DELX(M))
```

```
      CALL MATMUL(H,HT,HTMX,2,2,2)
      DO 24 I=1,2
      DO 25 J=1,2
      FAKE=HTMX(I,J)
      H(I,J)=FAKE
   25 CONTINUE
   24 CONTINUE
      IF (M.LT.N)GO TO 23
      HT(1,1)= 1.D0
      HT(1,2)= RN
      HT(2,1)= 0.D0
      HT(2,2)= 1.D0
      CALL MATMUL(H,HT,HTMX,2,2,2)
      F=HTMX(1,2)
      RETURN
      END
      SUBROUTINE BETA(X,F,DERF,R1,A,CON,DELX,RN,XST,EPS,IEND,IER,N)
      IMPLICIT REAL*8(A-H,O-Z)
C*************************************************************************
C     BETA SUBROUTINE IS USED TO QUICKLY AND ACCURATELY CONVERGE ON A
C     ROOT ONCE THE GENERAL LOCATION OF THE ROOT HAS BEEN DETERMINED.
C     IT NEEDS THE VALUE OF THE FUNCTION AS WELL AS ITS DERIVATIVE AT
C     PARTICULAR VALUES OF S.
C*************************************************************************
      DIMENSION A(N),CON(N),DELX(N)
C     PREPARE ITERATION
      IER=0
   34 CONTINUE
      X=XST
      TOL=X
      CALL FIGURE(TOL,N,R1,A,CON,DELX,RN,F)
      CALL DERIV(TOL,N,R1,A,CON,DELX,RN,DERF)
      TOLF=1.0D2*EPS
C     START ITERATION LOOP
      DO 6 I=1,IEND
      IF (F)1,7,1
C     EQN. NOT SATISFIED BY X
    1 IF (DERF)2,3,2
C     ITERATION IS POSSIBLE
    2 DX=F/DERF
      X=X-DX
      IF (X)30,30,31
   30 CONTINUE
      XST=XST-1.0D-2
      GO TO 34
   31 CONTINUE
      TOL=X
      CALL FIGURE(TOL,N,R1,A,CON,DELX,RN,F)
      CALL DERIV(TOL,N,R1,A,CON,DELX,RN,DERF)
C     TEST ON SATISFACTORY ACCURACY
      TOL=EPS
```

```
      G=DABS(X)
      IF (G-1.D0)4,4,3
    3 TOL=TCL*G
    4 IF (DABS(DX)-TCL)5,5,6
    5 IF (DABS(F)-TOLF)7,7,6
    6 CONTINUE
C     END OF ITERATION LOOP
C     NO CONVERGENCE AFTER IEND ITERATION STEPS. ERROR RETURN.
      IER=1
    7 RETURN
C     ERROR RETURN IN CASE OF ZERO DIVISOR
    8 IER=2
      RETURN
      END
      SUBROUTINE DERIV(S,N,R1,A,CON,DELX,RN,DERF)
      IMPLICIT REAL*8(A-H,O-Z)
C*************************************************************************
C     SUBROUTINE DERIV FINDS NUMERICAL VALUES FOR THE DERIVATIVE OF
C     EACH ELEMENT OF THE 2X2 MATRIX, MATRIX (12), FOR A SPECIFIC
C     VALUE OF S.
C*************************************************************************
      DIMENSION A(N),CON(N),DELX(N),H(2,2),HT(2,2),HHT(2,2),DH(2,2),
     1HDH(2,2),HH(2,2),HTMX(2,2),HNEW(2,2),HNEWER(2,2),HHTMX(2,2)
      L=0
   67 L=L+1
      M=1
      H(1,1)=1.D0
      H(1,2)=R1
      H(2,1)=0.D0
      H(2,2)=1.D0
      IF (M.EQ.L)GO TO 68
   54 HT(1,1)=DCOS(DSQRT(S/A(M))*DELX(M))
      HT(1,2)=(-1.D0/CON(M))*DSQRT(A(M)/S)*DSIN(DSQRT(S/A(M))*DELX(M))
      HT(2,1)=CON(M)*DSQRT(S/A(M))*DSIN(DSQRT(S/A(M))*DELX(M))
      HT(2,2)=DCOS(DSQRT(S/A(M))*DELX(M))
      CALL MATMUL(H,HT,HHT,2,2,2)
      DO 55 I=1,2
      DO 56 J=1,2
      FAKE=HHT(I,J)
      H(I,J)=FAKE
   56 CONTINUE
   55 CONTINUE
      M=M+1
      IF (M.NE.L)GO TO 54
   68 DH(1,1)=-(DELX(M)/(2.D0*DSQRT(A(M)*S)))*DSIN(DSQRT(S/A(M))
     1*DELX(M))
      DH(1,2)=-(DELX(M)/(2.D0*CON(M)*S))*DCOS(DSQRT(S/A(M))*DELX(M))+
     1(1.D0/(2.D0*CON(M)))*DSQRT(A(M)/(S**3))*DSIN(DSQRT(S/A(M))
     1*DELX(M))
      DH(2,1)=((CON(M)*DELX(M))/(2.D0*A(M)))*DCOS(DSQRT(S/A(M))*
     1DELX(M))+(CON(M)/(2.D0*DSQRT(A(M)*S)))*DSIN(DSQRT(S/A(M))*
     1DELX(M))
```

```
      DH(2,2)=-(DELX(M)/(2.D0*DSQRT(A(M)*S)))*DSIN(DSQRT(S/A(M))*
     1DELX(M))
      CALL MATMUL(H,DH,HDH,2,2,2)
      IF (M.NE.1)GO TO 69
      GO TO 59
   69 CONTINUE
      IF (M.EQ.N)GO TO 70
   59 M=M+1
      HH(1,1)=DCOS(DSQRT(S/A(M))*DELX(M))
      HH(1,2)=(-1.D0/CCN(M))*DSQRT(A(M)/S)*DSIN(DSQRT(S/A(M))*DELX(M))
      HH(2,1)=CCN(M)*DSQRT(S/A(M))*DSIN(DSQRT(S/A(M))*DELX(M))
      HH(2,2)=DCOS(DSQRT(S/A(M))*DELX(M))
      CALL MATMUL(HDH,HH,HTMX,2,2,2)
      DO 57 I=1,2
      DO 58 J=1,2
      DUM=HTMX(I,J)
      HDH(I,J)=DUM
   58 CONTINUE
   57 CONTINUE
      IF (M.LT.N)GO TO 59
   70 H(1,1)=1.D0
      H(1,2)=RN
      H(2,1)=0.D0
      H(2,2)=1.D0
      CALL MATMUL(HDH,H,HTMX,2,2,2)
      IF (L.GT.1)GO TO 60
      DO 61 I=1,2
      DO 62 J=1,2
      DUMM=HTMX(I,J)
      HNEW(I,J)=DUMM
   62 CONTINUE
   61 CONTINUE
      IF (L.EQ.1)GO TO 67
   60 CONTINUE
      DO 63 I=1,2
      DO 64 J=1,2
      DUMMI=HTMX(I,J)
      HNEWER(I,J)=DUMMI
   64 CONTINUE
   63 CONTINUE
      CALL HMADD(HNEW,HNEWER,RHTMX,2,2)
      DO 65 I=1,2
      DO 66 J=1,2
      DUMMIE=RHTMX(I,J)
      HNEW(I,J)=DUMMIE
   66 CONTINUE
   65 CONTINUE
      IF (L.LT.N)GO TO 67
      DERF=HNEW(1,2)
      RETURN
      END
```

```
      SUBROUTINE MATMUL(A,B,C,M,N,L)
      IMPLICIT REAL*8(A-H,O-Z)
C     THIS SUBROUTINE PERFORMS MATRIX MULTIPLICATION.
      DIMENSION A(M,N), B(N,L), C(M,L)
      DO 300 I=1,M
      DO 200 J=1,L
      SUM=0.0D0
      SUM=0.0
      DO 100 K=1,N
  100 SUM=SUM+A(I,K)*B(K,J)
  200 C(I,J)=SUM
  300 CONTINUE
      RETURN
      END
      SUBROUTINE HMADD(A,B,R,M,N)
      IMPLICIT REAL*8(A-H,O-Z)
C     THIS SUBROUTINE PERFORMS MATRIX ADDITION.
      DIMENSION A(M,N),B(M,N),R(M,N)
      DO 10 I=1,2
      DO 11 J=1,2
      R(I,J)=A(I,J)+B(I,J)
   11 CONTINUE
   10 CONTINUE
      RETURN
      END
$ENTRY
```

APPENDIX D - Z-COEFFICIENTS PROGRAM

```
$JOB            TDH,TIME=(0,09),PUNCH,NCLIST
      IMPLICIT REAL*8(A-H,C-Z)
C***********************************************************************
C     THIS PRCGRAM FINCS THE NUMERATCR AND DENOMINATOR COEFFICIENTS OF
C     PCLYNCMIALS IN Z**-1 FCR THE Z- TRANSFER FUNCTIONS R(Z) AND W(Z).
C***********************************************************************
      DIMENSICN A(10),CCN(10),DELX(10),X(70),Y(70),Z(70),RZ(70),
     1AN(70),DN(70),AUP(70),CUP(70),BETA(70),BOTM(70)
   50 FCRMAT(I5)
   51 FCRMAT(D23.16)
   52 FORMAT(' ','BEE2=',F20.16)
   53 FORMAT(' ','BEE1=',F20.16)
   54 FORMAT(' ','CEE2=',F20.16)
   55 FORMAT(' ','CEE1=',F20.16)
   56 FCRMAT(' ','B(',I2,')=',D24.16)
   57 FORMAT(' ','AN(',I2,')=',F20.16,10X,'DN(',I2,')=',F20.16)
   58 FCRMAT(' ','A(',I2,')=',D24.16,10X,'C(',I2,')=',D24.16)
   59 FCRMAT(D24.16)
   60 FORMAT('1',55X,'STEPSIZE=',F4.1,1X,'HR')
   61 FORMAT('-',25X,'NUMERATCR',24X,'DENOMINATOR',24X,'NUMERATOR')
   62 FORMAT(' ',27X,'R(Z)',26X,'R(Z) AND W(Z)',25X,'W(Z)')
   63 FORMAT('0')
   64 FORMAT(' ',15X,'A(',I2,')=',D23.16,5X,'B(',I2,')=',D23.16,5X,
     1'C(',I2,')=',D23.16)
   65 FCRMAT('1')
C***********************************************************************
C     INPUT TO THIS PRCGRAM IS:
C     N     - NUMBER OF RCOTS
C     NLAY  - NUMBER OF LAYERS
C     DELTA- STEPSIZE IN TIME
C     RI    - INNER SURFACE RESISTANCE
C     RN    - CUTER SURFACE RESISTANCE
C     A     - THERMAL DIFFUSIVITY
C     CON   - SPECIFIC HEAT
C     DELX  - LAYER THICKNESS
C     BETA  - RCCTS OF THE FUNCTION B
C***********************************************************************
      READ(5,50) N
      READ(5,50) NLAY
    . READ(5,51) DELTA
      READ(5,51) R1
      READ(5,51) RN
      DO 10 J=1,NLAY
   10 READ(5,51) A(J)
      DO 11 J=1,NLAY
   11 READ(5,51) CON(J)
      DO 12 J=1,NLAY
   12 READ(5,51) DELX(J)
      DO 13 J=1,N
   13 READ(5,51) BETA(J)
      CALL CCNSTS(CEE2,CEE1,BEE2,BEE1,R1,A,CON,DELX,RN,NLAY)
      WRITE(6,52) BEE2
      WRITE(6,53) BEE1
```

```
      WRITE(6,54) CEE2
      WRITE(6,55) CEE1
      NI=N+1
C***********************************************************************
C     THE (1-EXP(-BETA(M)DELTA)Z**-1 TERMS ARE NOW MULTIPLIED
C     TOGETHER AND THE RESULTING POLYNOMIAL IS THE DENOMINATOR OF BOTH
C     Z-TRANSFER FUNCTIONS.
C***********************************************************************
      IDIMX=2
      X(1)=1.D0
      X(2)=-DEXP(BETA(1)*DELTA)
      M=1
   22 M=M+1
      Y(1)=1.D0
      Y(2)=-DEXP(BETA(M)*DELTA)
      IDIMY=2
      CALL PMPY(Z,IDIMZ,X,IDIMX,Y,IDIMY,NI)
      DO 21 I=1,IDIMZ
      DUM=Z(I)
      X(I)=DUM
   21 CONTINUE
      IDIMX=IDIMZ
      IF (M.LT.N) GO TO 22
      DO 23 I=1,IDIMX
      FAKE=X(I)
      BOTM(I)=FAKE
      PUNCH59,X(I)
   23 WRITE(6,56) I,X(I)
C***********************************************************************
C     THIS FINDS ((1-Z**-1)**2)*(THE DENOMINATOR POLYNOMIAL) TO BE
C     USED IN EVALUATION OF THE NUMERATOR COEFFICIENTS FOR EACH TRANSFER
C     FUNCTION.
C***********************************************************************
      Y(1)=1.D0
      Y(2)=-2.D0
      Y(3)=1.D0
      IDIMY=3
      N2=N+2
      N3=N+3
      CALL PMPY(RZ,IDIMR,X,IDIMX,Y,IDIMY,N3)
C***********************************************************************
C     THE CONSTANTS Q(M) AND V(M) ARE EVALUATED HERE.
C***********************************************************************
      DO 25 I=1,N
      S=DABS(BETA(I))
      CALL FIGURE(S,NLAY,R1,A,CON,DELX,RN,APE)
      CALL DERIV(S,NLAY,R1,A,CON,DELX,RN,BARK)
      BARK=-BARK
      AN(I)=APE/((S**2)*BARK)
      DN(I)=1.D0/((S**2)*BARK)
   25 WRITE(6,57) I,AN(I),I,DN(I)
      CALL THENUM(RZ,BEE2,BEE1,AN,AUP,DELTA,BETA,N3)
```

```
      CALL THENUM(RZ,CEE2,CEE1,DN,CUP,DELTA,BETA,N3)
      DO 26 I=1,N3
      PUNCH59,AUP(I)
      PUNCH59,CUP(I)
   26 WRITE(6,58) I,AUP(I),I,CUP(I)
      WRITE(6,60) DELTA
      WRITE(6,61)
      WRITE(6,62)
      WRITE(6,63)
      BOTM(N2)=0.0D0
      BOTM(N3)=0.0D0
      DO 27 I=1,N3
   27 WRITE(6,64) I,AUP(I),I,BOTM(I),I,CUP(I)
      WRITE(6,65)
      STOP
      END
      SUBROUTINE THENUM(X,BC2,BC1,AD,TCP,DELTA,BETA,N3)
      IMPLICIT REAL*8(A-H,O-Z)
C*************************************************************************
C     SUBROUTINE THENUM CALCULATES THE NUMERATOR COEFFICIENTS FOR THE
C     Z- TRANSFER FUNCTIONS.
C*************************************************************************
      DIMENSION AD(N3),BETA(N3),O(70),TOP(70),X(70)
      N=N3-3
C*************************************************************************
C     VALUES OF THE INVERSE LAPLACE TRANSFER FUNCTION ARE CALCULATED
C     AT MULTIPLES OF DELTA.
C*************************************************************************
      TIME=0.0D0
      IPHI=1
   11 IPHI=IPHI+1
      TIME=TIME+DELTA
      PART=0.0D0
      DO 10 I=1,N
      SIZE=BETA(I)*TIME
      IF(SIZE.LE.-160.0D0)GO TO 12
   10 PART=PART+AD(I)*(DEXP(SIZE))
   12 O(IPHI)=(BC2*TIME)+BC1+PART
      IF(IPHI.LT.N+4) GO TO 11
C     NOW IPHI EQUALS N+4
C*************************************************************************
C     THE NUMERATOR COEFFICIENTS ARE NOW OBTAINED FROM THE DENOMINATOR,
C     INPUT, AND OUTPUT.
C*************************************************************************
      IOTA=0
   14 IOTA=IOTA+1
      TOP(IOTA)=0.0D0
      DO 13 L=1,IOTA
      K=IOTA-L+2
   13 TOP(IOTA)=TCP(IOTA)+O(K)*X(L)
      TOP(IOTA)=TOP(IOTA)/DELTA
      IF(IOTA.LT.IPHI-1) GO TO 14
      RETURN
      END
```

```
      SUBROUTINE CONSTS(CEE2,CEE1,BEE2,BEE1,R1,A,CON,DELX,RN,N)
C*********************************************************************
C     CONSTS FINDS THE NUMERICAL VALUES OF P2, P1, V2, AND V1.  THESE
C     CONSTANTS ARE ALL EVALUATED AT S=0.
C*********************************************************************
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(N),CON(N),DELX(N)
      DIMENSION H(2,2),HT(2,2),HHT(2,2),DH(2,2),HDH(2,2),HH(2,2)
      DIMENSION HTMX(2,2),HNEW(2,2),HNEWER(2,2),RHTMX(2,2)
C*********************************************************************
C     (A/B) AND (1./B) ARE CALCULATED AT S=0.
C     A=1. AT S=0.
C*********************************************************************
      HBO=R1+RN
      DO 90 M=1,N
      HB=-(DELX(M)/CON(M))
   90 HBO=HBO+HB
      BEE2=(1.D0/HBO)
      CEE2=BEE2
C*********************************************************************
C     THIS SECTION CALCULATES THE VALUE OF THE DERIVATIVE FOR EACH
C     ELEMENT OF THE 2X2 HEAT MATRIX AT S=0.
C*********************************************************************
      L=0
   67 L=L+1
      M=1
      H(1,1)= 1.D0
      H(1,2)= R1
      H(2,1)= 0.0D0
      H(2,2)= 1.D0
      IF (M.EQ.L)GO TO 68
   54 HT(1,1)=1.D0
      HT(1,2)=-(DELX(M)/CON(M))
      HT(2,1)=0.0D0
      HT(2,2)=1.D0
      CALL MATMUL(H,HT,HHT,2,2,2)
      DO 55 I=1,2
      DO 56 J=1,2
      FAKE=HHT(I,J)
      H(I,J)=FAKE
   56 CONTINUE
   55 CONTINUE
      M=M+1
      IF (M.NE.L)GO TO 54
   68 DH(1,1)=(DELX(M)**2)/(2.D0*A(M))
      DH(1,2)=-((DELX(M)**3)/(6.D0*CON(M)*A(M)))
      DH(2,1)=-((CON(M)*DELX(M))/A(M))
      DH(2,2)=(DELX(M)**2)/(2.D0*A(M))
      CALL MATMUL(H,DH,HDH,2,2,2)
      IF (M.NE.1)GO TO 69
      GO TO 59
   69 CONTINUE
```

```
      IF (M.EQ.N)GO TO 70
   59 M=M+1
      HH(1,1)=1.D0
      HH(1,2)=-(DELX(M)/CON(M))
      HH(2,1)=0.0D0
      HH(2,2)=1.D0
      CALL MATMUL(HDH,HH,HTMX,2,2,2)
      DO 57 I=1,2
      DO 58 J=1,2
      DUM=HTMX(I,J)
      HDH(I,J)=DUM
   58 CONTINUE
   57 CONTINUE
      IF (M.LT.N)GO TO 59
   70 H(1,1)=1.D0
      H(1,2)=RN
      H(2,1)=0.0D0
      H(2,2)=1.D0
      CALL MATMUL(HDH,H,HTMX,2,2,2)
      IF (L.GT.1)GO TO 60
      DO 61 I=1,2
      DO 62 J=1,2
      DUMM=HTMX(I,J)
      HNEW(I,J)=DUMM
   62 CONTINUE
   61 CONTINUE
      IF (L.EQ.1)GO TO 67
   60 CONTINUE
      DO 63 I=1,2
      DO 64 J=1,2
      DUMMI=HTMX(I,J)
      HNEWER(I,J)=DUMMI
   64 CONTINUE
   63 CONTINUE
      CALL HMADD(HNEW,HNEWER,RHTMX,2,2)
      DO 65 I=1,2
      DO 66 J=1,2
      DUMMIE=RHTMX(I,J)
      HNEW(I,J)=DUMMIE
   66 CONTINUE
   65 CONTINUE
      IF (L.LT.N)GO TO 67
C***********************************************************************
C     THE DERIVATIVES CALCULATED ABOVE ARE USED TO OBTAIN VALUES FOR
C     P1 AND V1.
C***********************************************************************
      DADSO=HNEW(1,1)
      DBDSO=HNEW(1,2)
      BDADSO=HBO*DADSO
      ADBDSO=DBDSO
      BEE1=((BDADSO-ADBDSO)/(HBO**2))
      CEE1=(-DBDSO/(HBO**2))
      RETURN
      END
```

```
      SUBROUTINE FIGURE(S,N,R1,A,CON,DELX,RN,F)
      IMPLICIT REAL*8(A-H,O-Z)
C***********************************************************************
C     SUBROUTINE FIGURE FINDS NUMERICAL VALUES FOR EACH ELEMENT OF THE
C     2X2 MATRIX, MATRIX (12), FOR A SPECIFIC VALUE OF S.  VALUES ARE
C     FOUND FOR ANY N LAYER STRUCTURE BY CALLING A MATRIX MULTIPLICATION
C     SUBROUTINE.
C***********************************************************************
      DIMENSION A(N),CON(N),DELX(N),H(2,2),HT(2,2),HTMX(2,2)
      M=0
      H(1,1)= 1.D0
      H(1,2)= R1
      H(2,1)= 0.D0
      H(2,2)= 1.D0
   23 CONTINUE
      M=M+1
      HT(1,1)=DCOS(DSQRT(S/A(M))*DELX(M))
      HT(1,2)=(-1.D0/CON(M))*DSQRT(A(M)/S)*DSIN(DSQRT(S/A(M))*DELX(M))
      HT(2,1)=CON(M)*DSQRT(S/A(M))*DSIN(DSQRT(S/A(M))*DELX(M))
      HT(2,2)=DCOS(DSQRT(S/A(M))*DELX(M))
      CALL MATMUL(H,HT,HTMX,2,2,2)
      DO 24 I=1,2
      DO 25 J=1,2
      FAKE=HTMX(I,J)
      H(I,J)=FAKE
   25 CONTINUE
   24 CONTINUE
      IF (M.LT.N)GO TO 23
      HT(1,1)= 1.D0
      HT(1,2)= RN
      HT(2,1)= 0.D0
      HT(2,2)= 1.D0
      CALL MATMUL(H,HT,HTMX,2,2,2)
      F=HTMX(1,1)
      RETURN
      END
      SUBROUTINE DERIV(S,N,R1,A,CON,DELX,RN,DERF)
      IMPLICIT REAL*8(A-H,O-Z)
C***********************************************************************
C     SUBROUTINE DERIV FINDS NUMERICAL VALUES FOR THE DERIVATIVE OF
C     EACH ELEMENT OF THE 2X2 MATRIX, MATRIX (12), FOR A SPECIFIC
C     VALUE OF S.
C***********************************************************************
      DIMENSION A(N),CON(N),DELX(N),H(2,2),HT(2,2),HHT(2,2),DH(2,2),
     1HDH(2,2),HH(2,2),HTMX(2,2),HNEW(2,2),HNEWER(2,2),RHTMX(2,2)
      L=0
   67 L=L+1
      M=1
      H(1,1)=1.D0
      H(1,2)=R1
      H(2,1)=0.D0
      H(2,2)=1.D0
      IF (M.EQ.L)GO TO 68
```

```
54 HT(1,1)=DCOS(DSQRT(S/A(M))*DELX(M))
   HT(1,2)=(-1.D0/CON(M))*DSQRT(A(M)/S)*DSIN(DSQRT(S/A(M))*DELX(M))
   HT(2,1)=CON(M)*DSQRT(S/A(M))*DSIN(DSQRT(S/A(M))*DELX(M))
   HT(2,2)=DCOS(DSQRT(S/A(M))*DELX(M))
   CALL MATMUL(H,HT,HHT,2,2,2)
   DO 55 I=1,2
   DO 56 J=1,2
   FAKE=HHT(I,J)
   H(I,J)=FAKE
56 CONTINUE
55 CONTINUE
   M=M+1
   IF (M.NE.L)GO TO 54
68 DH(1,1)=-(DELX(M)/(2.D0*DSQRT(A(M)*S)))*DSIN(DSQRT(S/A(M))
  1*DELX(M))
   DH(1,2)=-(DELX(M)/(2.D0*CON(M)*S))*DCOS(DSQRT(S/A(M))*DELX(M))+
  1(1.D0/(2.D0*CON(M)))*DSQRT(A(M)/(S**3))*DSIN(DSQRT(S/A(M))
  1*DELX(M))
   DH(2,1)=((CON(M)*DELX(M))/(2.D0*A(M)))*DCOS(DSQRT(S/A(M))*
  1DELX(M))+(CON(M)/(2.D0*DSQRT(A(M)*S)))*DSIN(DSQRT(S/A(M))*
  1DELX(M))
   DH(2,2)=-(DELX(M)/(2.D0*DSQRT(A(M)*S)))*DSIN(DSQRT(S/A(M))*
  1DELX(M))
   CALL MATMUL(H,DH,HDH,2,2,2)
   IF (M.NE.1)GO TO 69
   GO TO 59
69 CONTINUE
   IF (M.EQ.N)GO TO 70
59 M=M+1
   HH(1,1)=DCOS(DSQRT(S/A(M))*DELX(M))
   HH(1,2)=(-1.D0/CON(M))*DSQRT(A(M)/S)*DSIN(DSQRT(S/A(M))*DELX(M))
   HH(2,1)=CON(M)*DSQRT(S/A(M))*DSIN(DSQRT(S/A(M))*DELX(M))
   HH(2,2)=DCOS(DSQRT(S/A(M))*DELX(M))
   CALL MATMUL(HDH,HH,HTMX,2,2,2)
   DO 57 I=1,2
   DO 58 J=1,2
   DUM=HTMX(I,J)
   HDH(I,J)=DUM
58 CONTINUE
57 CONTINUE
   IF (M.LT.N)GO TO 59
70 H(1,1)=1.D0
   H(1,2)=RN
   H(2,1)=0.D0
   H(2,2)=1.D0
   CALL MATMUL(HDH,H,HTMX,2,2,2)
   IF (L.GT.1)GO TO 60
   DO 61 I=1,2
   DO 62 J=1,2
   DUMM=HTMX(I,J)
   HNEW(I,J)=DUMM
62 CONTINUE
61 CONTINUE
```

```
      IF (L.EQ.1)GO TO 67
   60 CONTINUE
      DO 63 I=1,2
      DO 64 J=1,2
      DUMMI=HTMX(I,J)
      HNEWER(I,J)=DUMMI
   64 CONTINUE
   63 CONTINUE
      CALL HMADD(HNEW,HNEWER,RHTMX,2,2)
      DO 65 I=1,2
      DO 66 J=1,2
      DUMMIE=RHTMX(I,J)
      HNEW(I,J)=DUMMIE
   66 CONTINUE
   65 CONTINUE
      IF (L.LT.N)GO TO 67
      DERF=HNEW(1,2)
      RETURN
      END
      SUBROUTINE PMPY(Z,IDIMZ,X,IDIMX,Y,IDIMY,NI)
      IMPLICIT REAL*8(A-H,O-Z)
C*******************************************************************************
C     PMPY PERFORMS POLYNOMIAL MULTIPLICATION.  THE IDIM,S ARE
C     DIMENSIONS OF THE POLYNOMIALS.  THE COEFFICIENTS OF THE INPUT
C     POLYNOMIALS ARE THE ARRAYS X AND Y AND Z IS THE ARRAY OF
C     COEFFICIENTS OF THE RESULTANT POLYNOMIAL.
C*******************************************************************************
      DIMENSION Z(NI),X(NI),Y(NI)
   20 IDIMZ=IDIMX+IDIMY-1
      DO 30 I=1,IDIMZ
   30 Z(I)=0.0D0
      DO 40 I=1,IDIMX
      DO 40 J=1,IDIMY
      IF(DABS(X(I)).LE.1.0D-40)GO TO 43
      GO TO 42
   43 X(I)=0.0D0
   42 CONTINUE
      K=I+J-1
   40 Z(K)=X(I)*Y(J)+Z(K)
   50 RETURN
      END
      SUBROUTINE MATMUL(A,B,C,M,N,L)
      IMPLICIT REAL*8(A-H,O-Z)
C     THIS SUBROUTINE PERFORMS MATRIX MULTIPLICATION.
      DIMENSION A(M,N), B(N,L), C(M,L)
      DO 300 I=1,M
      DO 200 J=1,L
      SUM=0.0D0
      DO 100 K=1,N
  100 SUM=SUM+A(I,K)*B(K,J)
  200 C(I,J)=SUM
  300 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE HMADD(A,B,R,M,N)
      IMPLICIT REAL*8(A-H,O-Z)
C     THIS SUBROUTINE PERFORMS MATRIX ADDITION.
      DIMENSION A(M,N),B(M,N),R(M,N)
      DO 10 I=1,2
      DO 11 J=1,2
      R(I,J)=A(I,J)+B(I,J)
   11 CONTINUE
   10 CONTINUE
      RETURN
      END
$ENTRY
```

APPENDIX E - FCALC PROGRAM

```
$JOB
      IMPLICIT REAL*8(A-H,C-Z)
      DIMENSION A(70),B(70),C(70),TIN(801),TOUT(801),F(801)
   51 FORMAT(' ','F(',I4,')=',D23.16)
   22 FORMAT(D24.16)
   24 FORMAT(I5)
      READ(5,24) ITHETA
      DO 8 I=1,ITHETA
    8 READ(5,22) A(I)
      DO 9 I=1,ITHETA
    9 READ(5,22) B(I)
      DO 10 I=1,ITHETA
   10 READ(5,22) C(I)
      SAREA=1.0D0
      IPT=0
      IEND=225
      TGO=70.0D0
      DO 50 I= 4,IEND
      TIN(I)=71.0D0
   50 TOUT(I)=75.0D0
      TIN(1)=70.0D0
      TOUT(1)=70.0D0
      DO 61 I=1,2
      TIN(I+1)=(0.5000D0*I)+70.0D0
   61 TOUT(I+1)=(2.5000D0*I)+70.0D0
   52 IPT=IPT+1
      CALL FCALC(A,B,C,ITHETA,TGO,TIN,TOUT,IPT,F,SAREA)
      IF(IPT.LT.IEND)GO TO 52
      DO 62 I=1,IEND
   62 WRITE(6,51) I,F(I)
      STOP
      END
      SUBROUTINE FCALC(A,B,C,ITHETA,TGO,TIN,TOUT,IPT,F,SAREA)
      IMPLICIT REAL*8(A-H,C-Z)
C***********************************************************************
C     FCALC FINDS THE HEAT FLUX AT THE INNER SURFACE OF A MULTILAYER
C     STRUCTURE, WHEN IT IS GIVEN THE Z-COEFFICIENTS FOR THE STRUCTURE,
C     A, B, AND C, THE NUMBER OF INDIVIDUAL COEFFICIENTS, ITHETA, THE
C     INITIAL WALL TEMP, TGO, INSIDE TEMP, TIN, OUTSIDE TEMP, TOUT,
C     AND THE SURFACE AREA OF THE STRUCTURE, SAREA.  FCALC USES THE
C     EQUATIONS FROM TABLE 2.
C***********************************************************************
      DIMENSION A(ITHETA),B(ITHETA),C(ITHETA),TIN(IPT),TOUT(IPT),
     1F(IPT),FLT( 801),FRT( 801),U( 801),V( 801)
      U(IPT)=TIN(IPT)-TGO
      V(IPT)=TOUT(IPT)-TGO
      IF(IPT.GT.ITHETA)GO TO 11
      IPHI=IPT
      RT=0.0D0
      RTER=0.0D0
      GT=0.0D0
      N=0
   10 N=N+1
```

```
      M=IPHI-N+1
      IF (N.EQ.1)GO TO 12
      GT=GT+(F(M)*B(N))
   12 FLT(IPHI)=GT
      RT=RT+(U(M)*A(N))
      RTER=RTER+(V(M)*C(N))
      IF (N.LT.IPHI)GO TO 10
      FRT(IPHI)=-RT+RTER
      F(IPHI)=(FRT(IPHI)-FLT(IPHI))
      GO TO 30
   11 IFLY=IPT
   50 RT=0.0D0
      GT=0.0D0
      GTER=0.0D0
      M=0
      IFLY=IFLY+1
      N=IFLY
   51 M=M+1
      N=N-1
      GT=(U(N)*A(M))+GT
      GTER=(V(N)*C(M))+GTER
      IF (M.EQ.1)GO TO 51
      RT=RT+(F(N)*B(M))
      IF (M.LT.ITHETA)GO TO 51
      IGOT=IFLY-1
      F(IGOT)=(-GT-RT+GTER)
   30 CONTINUE
      RETURN
      END
$ENTRY
```

APPENDIX F - DERIVATION OF CONTROL SYSTEM EQUATIONS

The development of this air conditioning control system model is based upon the application of heat and mass balance equations to the various components of the system and the implementation of an air temperature sensor feedback to control the input of heat into the controlled environment. In the development of models of this type it is common practice to express the variables of the system in terms of their deviation from an equilibrium condition. In the following development, the equilibrium temperature will be represented by θ_{Go}. As illustrated in Figure 10, the system to be modeled consists of a controlled room air space supplied by heat from a steam coil. The steam flow is controlled by a valve whose action is regulated in response to the temperature difference between the set point temperature and the sensed temperature of the room air. Supply air is made up of return air and fresh air.

First a differential equation describing the space temperature, θ_R, will be developed. The space is pictured below with the necessary nomenclature.

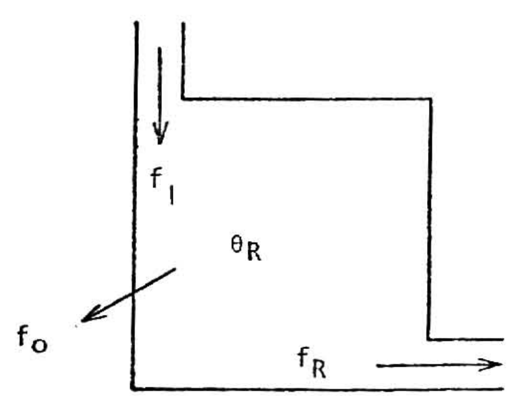

Figure 18. Room Air Space.

f_I - rate of heat flow into the space (BTU/hr)

f_O - rate of heat loss from air space through walls (BTU/hr)

f_R - rate of heat loss from air space through return duct (BTU/hr)

θ_I - inlet air temperature (°F)

θ_O - outside air temperature (°F)

θ_R - room air temperature (°F)

c - specific heat of air = 0.24 BTU/(lb_m-°F)

ρ - density of air (lb_m/ft^3)

V - volume of room (ft^3)

From a basic heat balance, it is known that the net heat flow into the air space during some time is equal to the change in internal energy of the air during that period. It follows that

$$f_I - f_R - f_O = \rho CV \frac{d\theta_R}{dt} .$$

The heat flow into the room less the heat returned is governed by the flowrate of air into and out of the space times the heat content of the air, so

$$f_I - f_R = \rho CA\dot{x}(\theta_I - \theta_R)$$

where A - area of ducts

$\dot{x}$ - air velocity in the ducts.

It has been assumed that the air density is constant, the room is at a uniform temperature, the return air is at room temperature, and the flowrate into the room equals the flowrate out of the room.

If the equations are combined, and the temperatures are expressed as deviations from the initial, equilibrium temperature then

$$\rho CV \frac{dv_R}{dt} = \rho CA\dot{x}(v_I - v_R) - f_o .$$

$v_r = \theta_R - \theta_{Go}$

$v_i = \theta_I - \theta_{Go}$

The Laplace transform is

$$(\tau_2 s+1)V_R = V_I - k_6 F_o$$

where V_R - the Laplace transform of v_R

V_I - the Laplace transform of v_I

F_O - the Laplace transform of f_o

$\tau_2 = V/A\dot{x}$ and

$k_6 = 1/\rho C A\dot{x}$.

Rearranging yields

$$V_R = \left(\frac{k_6}{\tau_2 s+1}\right)\left(\frac{1}{k_6} V_I - F_O\right) .$$

The schematic diagram for this relation is

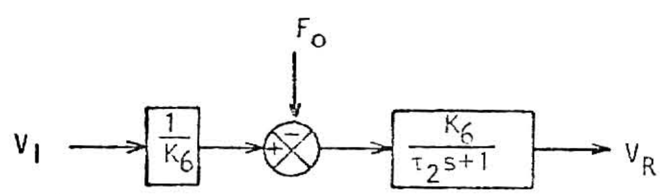

Figure 19. Room Block Diagram.

Applying the heat balance to the heating coil yields

$$f_s = \rho C A\dot{x}(\theta_I - \theta_M) + \rho_E C_E V_E \frac{d\theta_I}{dt}$$

where $\theta_M = M\theta_o + (1-M)\theta_R$ and

f_s - heat flowrate into the coil from the steam (BTU/hr)

θ_M - mixed air temperature entering the coil (°F)

M - fraction of outside make-up air

ρ_E - density of exchanger (lb_m/ft^3)

C_E - specific heat of exchanger material (BTU/lb_m-°F)

V_E - volume of exchanger (ft^3)

Taking the temperatures as deviations from the equilibrium temperature yields

$$(\tau_1 s+1)V_I = MV_o + (1-M)V_R + K_4 F_s$$

where F_s - the Laplace transform of f_s

$\tau_1 = \rho_E C_E V_E / \rho C A \dot{x}$

and $K_4 = 1/\rho C A \dot{x}$.

Rearranging,

$$V_I = \left(\frac{k_4}{\tau_1 s+1}\right)\left(\frac{M}{k_4} V_o + \frac{(1-M)}{k_4} V_R + F_s\right) \quad .$$

In block diagram form, the relation is

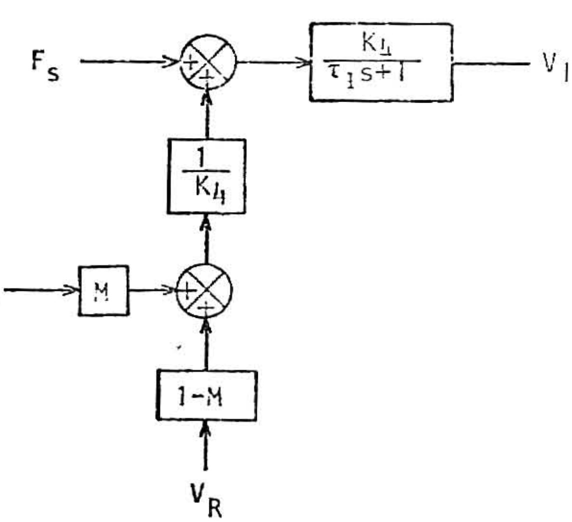

Figure 20. Heating Coil Block Diagram.

The steam flowrate into the coil may be assumed to be proportional to the valve opening, which is proportional to the difference in the setpoint and sensed room temperature, thus

$$F_s = K_1K_2K_3(VSET - V_R)$$

where K_1 - thermostat gain

K_2 - actuator gain

K_3 - valve gain

$VSET = R - \theta_{Go}$

R - set point temperature.

In block diagram form,

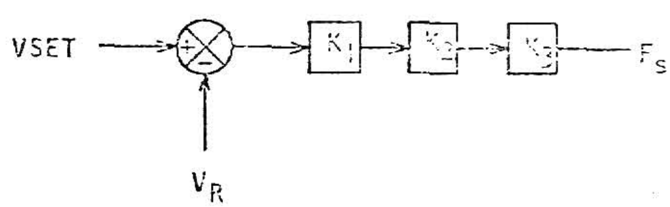

Figure 21. Steam Flow Control Block Diagram.

Heat conduction per unit area through the walls is given by equation (18).

$$F_o = -\frac{A}{B} V_I + \frac{1}{B} V_o \ .$$

When the area of the wall is considered, the total heat loss due to heat conduction through the walls for this development is given by the following relation. Observe that the variable, F_o, is used to signify BTU per hour rather than the previous BTU per hour per square foot.

$$F_o = \text{SAREA}(-\frac{A}{B} V_R + \frac{1}{B} V_o) \ .$$

SAREA - surface area of the wall

In block diagram form,

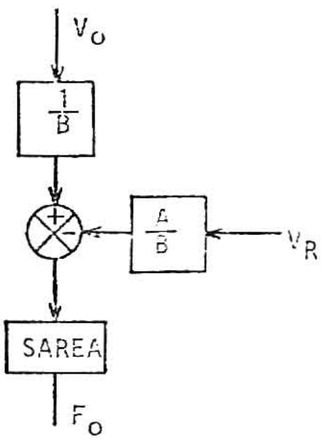

Figure 22. Block Diagram for Multi-Layer Wall.

The block diagrams may be combined and then reduced to obtain the following block diagram for the entire temperature control system including the walls as shown in Figure 10.

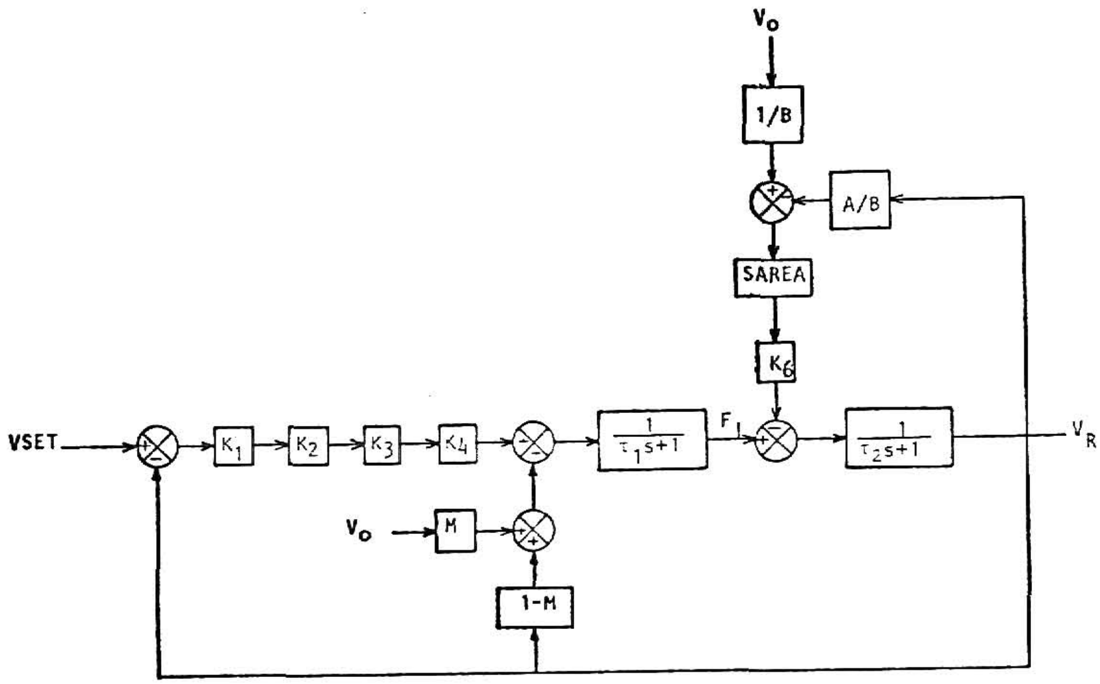

Figure 23. Air Conditioning Control System Block Diagram.

APPENDIX G - SYSTEM SIMULATION PROGRAM

```
$JCB
      IMPLICIT REAL*8(A-H,C-Z)
      DIMENSION YO(3),Y1(3),Y2(3),Y3(3),Y4(3),QO(3),Q1(3),Q2(3),
     1Q3(3),Q4(3),DERY(3),A(70),B(70),C(70),U(220),V(220),TOUT(220),
     1F(220)
   20 FORMAT('0','X=',F21.16,5X,'Y1=',F25.16,5X,'Y2=',F25.16)
   21 FORMAT(' ',25X,'DERY1=',F25.16,5X,'DERY2=',F25.16)
   22 FORMAT(F24.16)
   23 FORMAT(' ','IPT=',I5,5X,'F=',F25.16)
   24 FORMAT(I5)
      READ(5,24) ITHETA
      DO 8 I=1,ITHETA
    8 READ(5,22) A(I)
      DO 9 I=1,ITHETA
    9 READ(5,22) B(I)
      DO 10 I=1,ITHETA
   10 READ(5,22) C(I)
      SAREA=250.0D0
      NDIM=2
      XGO=0.0D0
      XEND=10.5D0
      H=0.025D0
      DELTA=0.10D0
      IEND=((2.0D0*(XEND-XGO))/DELTA)+1.0D0
      TGO=70.0D0
      YO(1)=0.0D0
      YO(2)=0.0D0
      DERY(1)=0.0D0
      DERY(2)=0.0D0
      DO 11 I=1,IEND
      TOUT(I)=50.0D0
   11 V(I)=TOUT(I)-TGO
      V(1)=0.0D0
      V(2)=-10.0D0
      X=XGO
      IPT=0
      DO 1 I=1,NDIM
    1 QO(I)=0.0D0
   13 IPT=IPT+1
      U(IPT)=YO(2)
      CALL FCALC(A,B,C,ITHETA,TGO,U,V,IPT,F)
      FI=YO(1)/2.273D-3
      WRITE(6,20) X,FI,YO(2)
      WRITE(6,21) DERY(1),DERY(2)
      FOUT=F(IPT)*SAREA
      WRITE(6,23) IPT,FOUT
      IF(X.GE.XEND)GO TO 12
      ITRK=0
    7 ITRK=ITRK+1
      IF(ITRK.GT.  4) GO TO 13
      CALL FCT(X,YO,DERY,V,TGO,IPT,F,SAREA)
      DO 2 I=1,NDIM
      R1=H*DERY(I)
```

```
   Q1(I)=Q0(I)+(1.5D0*(R1-2.0D0*Q0(I)))-.5D0*R1
 2 Y1(I)=Y0(I)+(.5D0*(R1-2.D0*Q0(I)))
   X=X+.5D0*H
   CALL FCT(X,Y1,DERY,V,TGO,IPT,F,SAREA)
   DO 3 I=1,NDIM
   R2=H*DERY(I)
   Q2(I)=Q1(I)+(.8786796564403573D0)*(R2-Q1(I))-(.29289321881345250D0
  1)*R2
 3 Y2(I)=Y1(I)+(.29289321881345250D0)*(R2-Q1(I))
   CALL FCT(X,Y2,DERY,V,TGO,IPT,F,SAREA)
   DO 4 I=1,NDIM
   R3=H*DERY(I)
   Q3(I)=Q2(I)+(5.121320343559642D0)*(R3-Q2(I))-(1.707106781186547D0
  1)*R3
 4 Y3(I)=Y2(I)+(1.707106781186547D0)*(R3-Q2(I))
   X=X+.5D0*H
   CALL FCT(X,Y3,DERY,V,TGO,IPT,F,SAREA)
   DO 5 I=1,NDIM
   R4=H*DERY(I)
   Q4(I)=Q3(I)+(.5D0*(R4-2.D0*Q3(I)))-.5D0*R4
 5 Y4(I)=Y3(I)+(.1666666666666667D0)*(R4-2.D0*Q3(I))
   DO 6 I=1,NDIM
   DUM=Q4(I)
   Q0(I)=DUM
   FAKE=Y4(I)
 6 Y0(I)=FAKE
   GO TO 7
12 CONTINUE
   STOP
   END
   SUBROUTINE FCT(X,Y,DERY,V,TGO,IPT,F,SAREA)
   IMPLICIT REAL*8(A-H,O-Z)
   DIMENSION Y(3),DERY(3),F(220),V(220)
   FNOW=F(IPT)
   AMIX=0.20D0
   TAU1=0.022D0
   TAU2=0.08D0
   CON=5.5D0
   CON6=2.273D-3
   FNOW=FNOW*SAREA*CON6
   R=70.0D0
   VSET=R-TGO
   DERY(1)=((AMIX*V(IPT))+((1.0-AMIX)*Y(2))+(CON*(VSET-Y(2)))-Y(1))
  1/TAU1
   DERY(2)=(Y(1)-FNOW-Y(2))/TAU2
   RETURN
   END
```

```
      SUBROUTINE FCALC(A,B,C,ITHETA,TGO,U,V,IPT,F)
      IMPLICIT REAL*8(A-H,O-Z)
C*****************************************************************************
C     FCALC FINDS THE HEAT FLUX AT THE INNER SURFACE OF A MULTILAYER
C     STRUCTURE, WHEN IT IS GIVEN THE Z-COEFFICIENTS FOR THE STRUCTURE,
C     A, B, AND C, THE NUMBER OF INDIVIDUAL COEFFICIENTS, ITHETA, THE
C     INITIAL WALL TEMP, TGO, INSIDE TEMP, TIN, OUTSIDE TEMP, TOUT,
C     AND THE SURFACE AREA OF THE STRUCTURE, SAREA.  FCALC USES THE
C     EQUATIONS FROM TABLE 2.
C*****************************************************************************
      DIMENSION A(ITHETA),B(ITHETA),C(ITHETA),
     1F(IPT),FLT(120),FRT(120),U(120),V(120)
      IF(IPT.GT.ITHETA)GO TO 11
      IPHI=IPT
      RT=0.0D0
      RTER=0.0D0
      GT=0.0D0
      N=0
   10 N=N+1
      M=IPHI-N+1
      IF (N.EQ.1)GO TO 12
      GT=GT+(F(M)*B(N))
   12 FLT(IPHI)=GT
      RT=RT+(U(M)*A(N))
      RTER=RTER+(V(M)*C(N))
      IF (N.LT.IPHI)GO TO 10
      FRT(IPHI)=-RT+RTER
      F(IPHI)=(FRT(IPHI)-FLT(IPHI))
      GO TO 30
   11 IFLY=IPT
   50 RT=0.0D0
      GT=0.0D0
      GTER=0.0D0
      M=0
      IFLY=IFLY+1
      N=IFLY
   51 M=M+1
      N=N-1
      GT=(U(N)*A(M))+GT
      GTER=(V(N)*C(M))+GTER
      IF (M.EQ.1)GO TO 51
      RT=RT+(F(N)*B(M))
      IF (M.LT.ITHETA)GO TO 51
      IGOT=IFLY-1
      F(IGOT)=(-GT-RT+GTER)
   30 CONTINUE
      RETURN
      END
$ENTRY
```

ACKNOWLEDGEMENTS

The author wishes to extend sincere appreciation to the Mechanical Engineering faculty for very generously giving time and wisdom throughout his education.

A special recognition goes to my wife, Susan, for her faithful encouragement and economic support during this study.

The assistance and inspiration from Dr. J. Garth Thompson are deeply appreciated.

Appreciation is also extended to Dr. Everett E. Haft and Dr. Robert L. Gorton for serving as committee members.

The author also acknowledges the Mechanical Engineering Department for the financial aid it provided.

VITA

Terry Del Hubbs

Candidate for the Degree
Master of Science

THESIS: HEAT CONDUCTION TRANSFER FUNCTIONS FOR MULTI-LAYER STRUCTURES

MAJOR FIELD: Mechanical Engineering

BIOGRAPHICAL:

Personal Data: Born in Salina, Kansas, November 15, 1953; the son of the late Delwin Victor Hubbs and Vivian M. Herbel.

Education: Graduated from Dorrance High School in 1971; received a Bachelor of Science degree in Mechanical Engineering from Kansas State University in May 1975; completed requirements for the Master of Science degree in April 1977.

Honors and Socities: 1975 Mac Short Award recipient; Pi Tau Sigma, mechanical engineering honorary; Triangle Fraternity, engineering, architecture, and science fraternity.

HEAT CONDUCTION TRANSFER FUNCTIONS FOR MULTI-LAYER STRUCTURES

by

TERRY DEL HUBBS

B.S., Kansas State University, 1975

AN ABSTRACT OF A MASTER'S THESIS

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Mechanical Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1977

NAME: Terry D. Hubbs DATE OF DEGREE: May 20, 1977

INSTITUTION: Kansas State University LOCATION: Manhattan, Kansas

TITLE OF STUDY: HEAT CONDUCTION TRANSFER FUNCTIONS FOR MULTI-LAYER STRUCTURES

PAGES IN STUDY: 90 CANDIDATE FOR DEGREE OF MASTER OF SCIENCE

MAJOR FIELD: Mechanical Engineering

To investigate the effect of control system dynamics on energy consumption, a simulation with a time stepsize short enough to manifest the dynamics of the control system is required. The purpose of this thesis is to demonstrate and satisfactorily document an algorithm for transient heat conduction through multi-layer walls that may be used as an element in such a simulation.

The algorithm developed and applied is the response factor technique. Laplace transforms are employed to attain conduction transfer functions that relate inner temperature and heat flux to outer temperature and heat flux for multi-layer structures. The Laplace transforms are inverted with z-transforms, placed in the form of a ratio of infinite power series in z^{-1}, and expanded to acquire solutions as a function of time. A detailed derivation of the concept is included.

Computer programs that evaluate the constants necessary to apply the algorithm are listed and described. One program finds roots of hyperbolic functions that describe the structure, another calculates the coefficients of the z-transfer functions, and a third uses the coefficients to compute heat flux at the inner surface of the wall.

The concept is demonstrated on a double and then a triple-layer wall with identical inputs and the outputs are compared. A room temperature

control system is modeled where the walls are simulated with the response factor algorithm. The model is subjected to various inputs.

This response factor algorithm may be used on walls with any number of layers, with or without surface resistances, and has a variable time stepsize. Accuracy depends primarily upon truncation error. The algorithm's strongest point is speed, resulting in reduced computing expense. This advantage makes it recommendable for applications where a wall or set of walls are to be studied a number of times.