2008

Mathematical Analysis of a Geothermal System

Ivan P. Erceg
Cleveland State University

MATHEMATICAL ANALYSIS OF A GEOTHERMAL SYSTEM

IVAN P. ERCEG

Bachelor of Science in Mechanical Engineering

Cleveland State University

May, 2002

Submitted in partial fulfillment of requirements for the degree

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

at the

CLEVELAND STATE UNIVERSITY
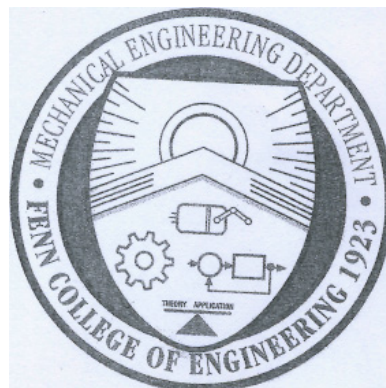
September, 2008

This thesis has been approved

for the Department of Mechanical Engineering

and the College of Graduate Studies by

_____

Thesis Chairperson, Dr. Rama Gorla

_____

Department & Date

_____

Dr. Earnest Poulos

_____

Department & Date

_____

Dr. Majid Rashidi

_____

Department & Date

_____

Dr. Asuquo B. Ebiana

_____

Department & Date

DEDICATION

I dedicate this thesis paper in loving memory of Angela "Angie" Langford, the

Department of Mechanical Engineering's secretary who recently passed away. Angie

really cared about the students and did her best to help. She really encouraged and

cheered me on with my studies from an undergraduate and into my master's program.

Once she told me that I had more potential than the average student. I wish she would

have stuck around longer to see me finish my thesis.

ACKNOWLEDGEMENT

MATHEMATICAL ANALYSIS OF A GEOTHERMAL SYSTEM

IVAN P. ERCEG

ABSTRACT

The issue being examined is to design a more economical and efficient therefore superior geothermal system than currently in use in industry. Current geothermal systems are designed and built 300 feet into the ground. After researching ground temperature gradients for Ohio we found out that below 10 feet of depth, the temperature varies by 1 degree Fahrenheit per 100 feet depth. Our goal is to utilize the heat as close to the surface as possible and greatly reduce the need to dig so deeply into the ground.

The procedure used to go about designing a superior geothermal system is to model an oversized tank going down about 50 feet in depth. Then analyze the model using ground temperature gradient data and the SINDATHERMAL Analyzer program. By investigating different glycol flow rates, glycol supply temperatures, and tank dimensions we are able to investigate various designs and analyze the results for optimization.

Our results were such that by using our central tank design we were able to design a Geothermal system superior in terms of performance, construction costs, and operating costs to what is in use in industry today.

Our conclusion is that there is no need to drill down 300 feet using traditional Geothermal designs when, by using our design, drilling down 50 feet and using a central tank will result in superior heat flow.

TABLE OF CONTENTS

PAGE

LIST OF FIGURES

NOMENCLATURE

TSW   tank switch time, minutes

PI   constant

Cpgl   fluid specific heat, btu/lb-F

VIS   dynamic viscocity, centipoise

VISCgl   dynamic viscocity, lbm/ft-min

KKf   glycol thermal conductivity, btu/hr-ft-F

Kfgl   glycol thermal conductivity, btu/min-ft-F

Rhoegl   glycol fluid density, lbm/cu.ft

GLFR   glycol fluid mass flow rate, gpm

GM   glycol flow rate, lbm/min

Tgi   glycol inlet temperature, deg. F

Rhoetk   tank wall density, lbm/cu-ft

KKt   tank wall thermal conductivity, btu/hr-ft-F

Ktk   tank wall thermal conductivity, btu/min-ft-F

Cptk   tank wall specific heat, btu/lb-F

Kt   tube wall thermal conductivity, btu/hr-ft-F

Ktb   tube wall thermal conductivity, btu/min-ft-F

Cptb   tube specific heat, btu/lb-F

Rhoetb   tube density, lbm/cu-ft

Kgi   ground thermal conductivity, btu/hr-ft-F

Kg   ground thermal conductivity, btu/min-ft-F

Cpg   ground specific heat, btu/lb-F

Rhoeg   ground density, lbm/cu-ft

Tmw   tank minimum wall thickness, inch

Tmb   tank top mid and bottom thickness, inch

Ttmb   tank top mid and bottom thickness, ft

DTOI   tank outside diameter, inch

Dto   outer tank diameters, ft

DTII   inside tank diameter, inch

Dti   inner tank diameter, ft

Ltk   tank height length, ft

Nttk   tank number of nodal increments

Rto   tank outer radius, ft

Rti   tank inner radius, ft

Rtm   mid-wall radius, ft

Rtc   center tank radius, ft

Dltk   vertical tank nodal increment, ft

Twt   tube minimum wall thickness, inch

Dpoi   outer tube diameter, inch

Dpii   inner tube diameter, inch

Dpo   outer tube diameter, ft

Dpi   inner tube diameter, ft

Ltb   pipe length is the same as tank height, ft

Ntb   tube number of node increments

Rpo   tank outer radius, ft

Rpi   tank inner radius, ft

Dltb   tube nodal incremental length, ft

FDGa1   ground depth to top of first tank, ft

Lg1   ground vertical height above top of tank, ft

Lg2   ground vertical height below bottom of tank, ft

Lt1   single tank height, ft

Rgrad   ground outer radius surrounding the tank, ft

FDGt1   ground depth to top of tank, ft

FDGt2   ground depth to bottom of 1st tank, ft

FDGt3   ground depth to bottom of second tank, ft

FDGb1   ground depth below second tank bottom, ft

Dla1   vertical ground nodal increment above tank top,ft

Dlb1   vertical ground nodal increment below tank bottom,ft

Prgl   Prandtl number

Redtb   Reynold s number

Nutb   Nusselt number

Htb   convection heat transfer coef.btu/min-ft2-F

Aptb   node inner tube area, sq-ft

GCtb   wall convection conductance

Fcontb   moving fluid conductance

NFVoltb   node fluid volume

Captb   tube node capacity

Fgtb   radial tube wall conductance

Prtk	Prandtl number
Redtk	Reynold s number
Nutk	Nusselt number
Htk	convection heat transfer coef.btu/min-ft2-F
Aptk	nodal inner tank area, sq-ft
FCOntk	moving fluid conductance
NFVoltk	node fluid volume
Captk	node glycol fluid capacitance
Ftmb	tank top,mid-section and bottom conductance
Ftmw	tank mid-wall radial conduction
Ftcon	tank wall convection conductance
Ctwc	tank wall capacitance
Ctsc	tank top-mid-bottom wall capacitance
Ftfk	moving fluid conductance
Ftrc	tank radial glycol conduction

CHAPTER I

INTRODUCTION

1.1 GEOTHERMAL ENERGY

The earth's crust is heated by the decay or radioactive elements. At the very center, it is

believed temperatures exceed 11,000 degrees Fahrenheit, hotter than the surface of the

sun. A cross-section of the Earth reveals three concentric layers. The Surface of the Earth

consists of a thin, hard crust ranging from 10 to 100 kilometers thick. This is called the

crust and this is what we see walking around on the surface of the Earth and what we call

home. Under that, a donut-shaped mantle 2,900 kilometers thick. It consists of viscous

molten rock that flows very slowly, on a geological time scale.

At the center of the Earth lies a two-part core. The inner part is about the size of our

moon, and has a density of essentially steel. The outer core surrounding it is an ocean of

liquid metal 2,300 kilometers thick. The Earth's rotation makes this ocean flow and swirl,

and the moving metal generates the planet's magnetic field. Most of Earth's heat is stored

in the mantle and there are four sources that keep it hot.

First, there's the heat left over from when gravity first condensed a planet from the cloud of hot gases and particles in pre-Earth space. As the molten ball cooled, some 4 billion years ago, the outside hardened and formed a crust. The mantle is still cooling down. This heat is estimated to contribute to 5 to 10 percent of the total.

To explain gravitational heat, freshly formed Earth was not of a consistent density. In a gravitational sorting process called differentiation, the denser, heavier parts were drawn to the center, and the less dense areas were displaced outwards. The friction created by this process generated considerable heat, which, like the original heat, still has not fully dissipated.

Then there's latent heat. This type arises from the core's expanding as the Earth cools from the inside out. Just as freezing water turns to ice, that liquid metal is turning solid—and adding volume in the process. The inner core is becoming larger by about a centimeter every thousand years. The heat released by this expansion is seeping into the mantle.

For all this the vast majority of the heat in Earth's interior—up to 90 percent—is fueled by the decaying of radioactive isotopes like Potassium 40, Uranium 238, 235, and Thorium 232 contained within the mantle. These isotopes radiate heat as they shed excess energy and move toward stability. Radioactivity is present not only in the mantle, but in the rocks of Earth's crust. For example, a 1-kilogram block of granite on the surface emanates a tiny but measurable amount of heat (about as much as a .000000001 watt light bulb) through radioactive decay. The heat flow from the interior to the surface is only 1/20,000 as great as the energy received from the Sun. [1]

The heat is carried by magma or water beneath the earth's surface. Some of the heat reaches the surface and manifests itself in geysers and hot springs throughout the world. Geothermal energy is energy derived from the heat of the earth's core. It is clean, abundant, and reliable. If properly developed, it can offer a renewable and sustainable energy source. There are three primary applications of geothermal energy: electricity generation, direct use of heat, and ground-source heat pumps. Direct use includes applications such as heating buildings or greenhouses and drying foods, whereas ground source heat pumps are used to heat and cool buildings using surface soils as a heat reservoir. Further, the pressurized steam from superheated water beneath the earth's surface can be used to power turbines and thus generate electricity.

Although geothermal power seems ideal in that it is naturally occurring and does not require structures to trap or collect the energy (as in solar panels or windmills), it does have limitations. The greatest drawback is that naturally occurring geothermal vents are not widely available. Artificial vents have been successfully drilled in the ground to reach the hot rocks below and then injected with water for the production of steam. However, oftentimes the source of heat is far too deep for this method to work well.

Nor can geothermal power realistically generate enough electricity for the entire United States or any other large industrialized nation. A good-sized hot spring can power at most a moderate sized city of around 50,000 people. There isn't enough viable hot springs to power all the cities in any large country. [2]

1.2   ADVANTAGES & DISADVANTAGES

Advantages

- Theoretically inexhaustible energy source

- No pollution

- Often an excellent supplement to other renewable energy sources

- Does not require structures such as solar panels or windmills to collect the energy – can be directly used to heat or produce electricity (thus very cheap)

Disadvantages

- Not available in many locations

- Not much power per vent

1.3 USES OF GEOTHERMAL ENERGY

Some applications of geothermal energy use the earth's temperatures near the surface, while other require drilling miles into the earth. The three main uses of geothermal energy are:

1) Direct Use and District Heating System which uses hot water from springs or reservoirs near the surface

2) Electricity Generation in a power plant requires water or steam at very high temperature (300 to 700 degrees Fahrenheit). Geothermal power plants are generally built where geothermal reservoirs are located within a mile or two of the surface.

3) Geothermal Heat Pumps use stable ground or water temperatures near the earth's surface to control building temperatures above the ground.

1.4 OTHER USES OF GEOTHERMAL ENERGY

Geothermal Steam Pipes

Geothermal Prawn Farming

Greenhouse Orchids

1.4.1   INDUSTRIAL PROCESS

Most of New Zealand's geothermal energy goes to produce electricity, but it can be used for any processes where heat is required. The main non-electrical user is the Tasman Pulp and Paper Mill at Kawerau, which was built in 1957 and deliberately sited to take advantage of the underlying geothermal field. The heat is used for digesting wood pulp, drying timber and paper, and generating electricity.

1.4.2   GEOTHERMAL PRAWN FARMING

The world's only geothermal heated prawn farm was established in 1987 on the banks of the Waikato River, next to the Wairakei power station. The first prawns were imported from Malaysia in 1988, and by 2005 the 5.8-hectare farm was producing about 20 tones per year. The farm heats its own water with heat exchangers, which draw heat from the power station's waste water before it flows back into the Waikato River.

This is a good example of what is known as 'cascade use', where geothermal heat has a function past its primary purpose. Cascading improves the overall efficiency of a resource by using its waste products. In the case of the prawn farm, cascading also reduces the discharge of hot water into the river, where it can harm aquatic life.

1.4.3   HORTICULTURE

Geothermal waters are used for heating greenhouses on a small scale (covering 10 hectares in total), especially for the commercial, out-of-season production of vegetables, flowers and fruit. This includes a large greenhouse (0.8 hectares) for growing orchids for export, and another set up to grow capsicums with heat from Kawerau geothermal field.

1.4.4   CROP AND TIMBER DRYING

Drying Lucerne (alfalfa) using geothermal energy was pioneered in ohaki in the 1970s. Geothermal heat from the ohaki power station has been used to make high-protein pellets to feed stock and to process dried juice into a protein concentrate. A timber-drying operation on site produces fence posts and poles, mainly for the local farming industry. The Tasman Pulp and Paper Mill uses geothermal steam in heat exchangers to heat clean air to 140 °C for timber drying.

GEOTHERMAL POWER (from the Greek words geo, meaning earth, and therme, meaning heat) is energy generated by heat stored beneath the Earth's surface. It is a developing field where serious energy potential has been identified. The ever growing juxtapose of uses for geothermal energy can be classified into direct uses and electricity generation. The use that can be derived depends on the operating temperature at the source. Direct use requires using geothermal heat directly (without a heat pump or power plant) for non-electricity generation purposes such as heating of buildings, industrial processes, domestic heating, greenhouses, aquaculture, public baths and pools. Direct use can take place at high and moderate to low temperature geothermal resources, depending on the use and scale required.

Even very low temperatures resources can be exploited for heat applications using ground-source heat pumps to facilitate space heating and cooling. Research has shown that ground-source heat pumps can be a lucrative investment.

- Where there are cold winters or hot summers

- Where electricity is over 12c/kWh

- Where gas is not available and where new construction is taking place

Almost all high temperature geothermal resources produce a mixture of water and steam at temperatures up to 300 °C, but rarely above that. Commercial geothermal energy operations access geothermal fluids by drilling wells typically 1000 to 3000 meters deep, while domestic wells for heating are usually only around 100 meters deep. When hot water above 90 to 100 °C flows up the well, the release of pressure allows some of the fluid to boil into steam, resulting in two-phase mixture at the surface. The water and steam are physically separated before use.

Potential energy use at various indicative field temperatures is:

30-69°C   Thermo culture, Bathing

70-140°C   Space and Water Heating, Drying

140-220°C   Drying, Process Heat, Binary Electrical plant

220+°C   Steam Turbine and Binary Electricity or Process Steam

1.5   WHERE IS GEOTHERMAL ENERGY FOUND?

Most geothermal reservoirs are deep underground with no visible clues showing above ground. Geothermal energy can sometimes find its way to the surface in the form of:

- Volcanoes and fumaroles (holes where volcanic gases are released)
- Hot springs
- Geysers

The most active geothermal resources are usually found along major plate boundaries where earthquakes and volcanoes are concentrated. Most of the geothermal activity in the world occurs in an area called the Ring of Fire. This area rims the Pacific Ocean. When magma comes close to the surface it heats ground water found trapped in porous rock or water running along fractured rock surfaces and faults. Such hydrothermal resources have

two common ingredients: water (hydro) and heat (thermal). Naturally occurring large areas of hydrothermal resources are called geothermal reservoirs. Geologists use different methods to look for geothermal reservoirs. Drilling a well and testing the temperature deep underground is the only way to be sure a geothermal reservoir really exists. Most of the geothermal reservoirs in the United States are located in the western states, Alaska, and Hawaii [2].

1.6   GEOTHERMAL HEAT PUMP

The system being modeled mathematically is a geothermal heat pump. A geothermal heat pump system is a heating and/or an air conditioning system that uses the Earth's ability to store heat in the ground and water thermal masses. These systems operate based on the stability of underground temperatures: the ground a few feet below surface has a very stable temperature throughout the year, depending upon location's annual climate. A geothermal heat pump uses that available heat in the winter and puts heat back into the ground in the summer. A geothermal system differs from a conventional furnace or boiler by its ability to transfer heat versus the standard method of producing heat. As energy costs continue to rise and pollution concerns continue to be a hot topic, geothermal systems may hold a solution to both of these concerns.

Geothermal heat pumps are also known as "GeoExchange" systems and "ground-source heat pumps." The latter term is useful as it clearly distinguishes the technology from air-source heat pumps. Geothermal heat pumps, which can be used in almost any region, should also be distinguished from geothermal heating. Geothermal heating is used in areas where exceptionally high underground

temperatures, such as those at hot springs and steam vents, are used to heat indoor spaces without the use of a heat pump.

A direct exchange or DX geothermal heat pump system is a geothermal heat pump system in which the refrigerant circulates through tubing placed in the ground. The refrigerant exchanges heat directly with the soil through the walls of the tubing. This simplicity allows the system to reach high efficiencies while using a relatively shorter and smaller set of buried tubing, reducing installation cost. DX systems, like water-source systems, can also be used to heat water in the house for use in radiant heating applications and for domestic hot water, as well as for cooling applications. [3]

The improvement made to the geothermal heat pump system is instead of using tubing running through the ground heat will be exchanged directly with a central tank placed in the ground resulting in even more installation savings and improved heat flow.

CHAPTER II

SCOPE AND APPROACH

The present investigation was undertaken in order to study and improve the heat flow of

various geothermal systems. The first step taken was to research the temperature versus

depth in Ohio.

2.1   GEOTHERMAL TEMPERATURE GRADIENT [4]

The temperature of the earth varies with the depth. The relation between the temperature

and the depth is given by the following equations;

Above 10 feet, the temperature is given by the equation:

$$T_{ground} = .6z + 49° \text{ F} \hspace{4cm} [2\text{-}1]$$

Where z < 10 feet in depth.

Under 10 feet, the temperature is given by the following equation:

$$T_{ground} = .01z + 54.9° \text{ F} \hspace{4cm} [2\text{-}2]$$

Where z > 10 feet in depth.

Figure (2.1) on the next page gives a graphical representation of the ground temperature

per function of depth.

Fig (2.1) Geothermal Temperature Gradients in Ohio

Ground Temperature, deg F (y-axis: 0, 10, 20, 30, 40, 50, 60, 70)

Ground Depth, ft (x-axis: 0, 5, 10, 60, 110, 160, 210, 260, 310, 360, 410, 460, 510, 560)
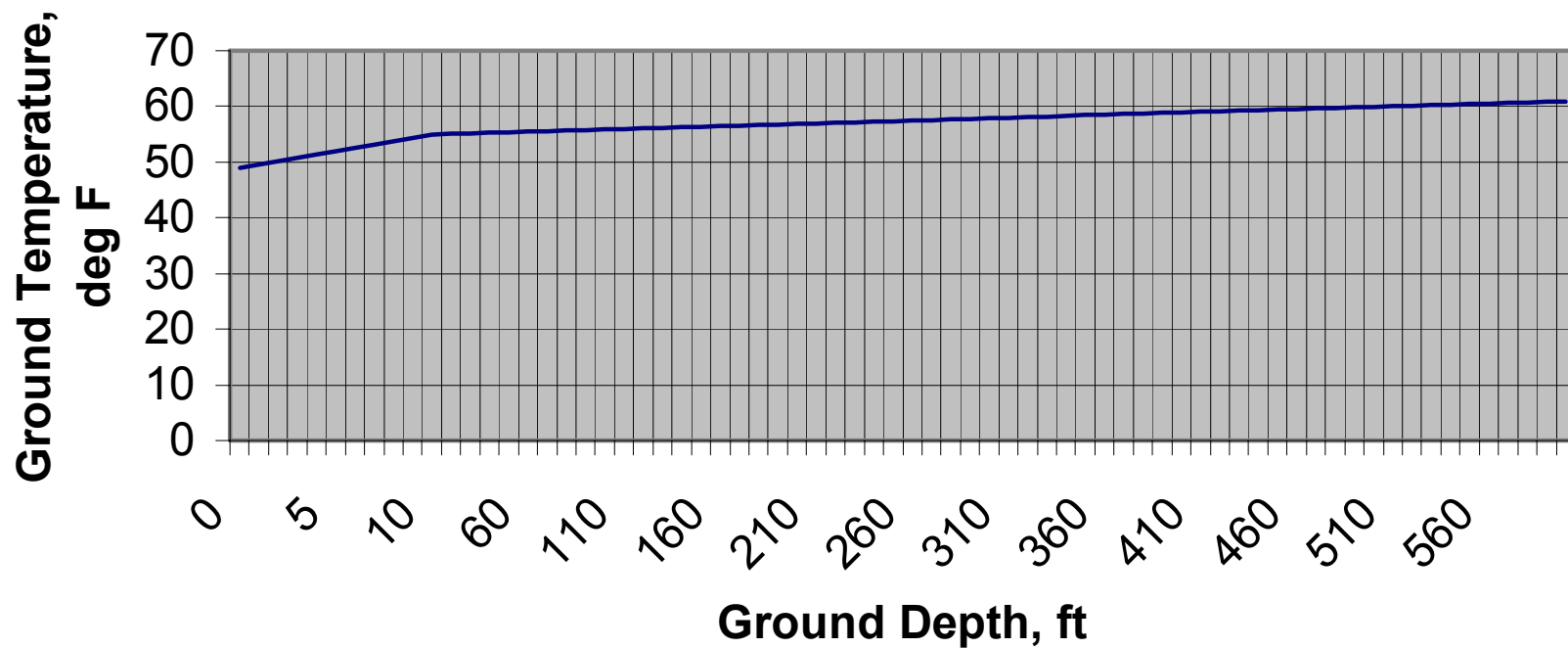
Analyzing Figure 2.1, we can observe that after 10 feet of depth, the ground temperature changes roughly 1 degree F per 100 feet. It is obvious that digging deeper into the Earth gives little or no advantage in terms of temperature change. Using this data our goal is to design a geothermal system optimizing the heat of the ground as close to the surface as possible. In other words we would like to minimize the depth needed in order to have a functional geothermal system.

2.2   DESIGN OF A GEOTHERMAL SYSTEM

After researching the ground temperature gradient the next step was to design a geothermal system and determine how to analyze our system. SolidWorks was used to create a three dimensional model and two dimensional drawings of our geothermal system. The design we choose is a cylindrical tank, but two variations. One variation has the cylindrical tank divided into two sections as shown in figure 2.2. The other design is just a cylindrical tank with one cavity. After we had 2-D drawings and 3-D models of our geothermal system, we could then start to describe the heat flow of the geothermal system. The first step was to recognize that the heat flow was two dimensional instead of three dimensional. Our main assumption for this was that the temperature of the ground only varied with depth (z), and not in the y direction. The other assumption is that heat within the tank varied only with relation to distance from the center or radially (r) and not in the y direction. Next, we decided to describe the heat flow using an RC network. We then determined where and how many nodes to use to model the system. Once the system was modeled with nodes we determined the resistances and capacitances of the nodes. After all this is known we were then able to write a program to use with SINDATHERMAL Analyzer. The two following figures describe our systems by illustrating the tanks, nodes, boundaries, and heat flow of our system.

FIG (2.2) - GEOTHERMAL SYSTEM: TANKS, NODES,
BOUNDARIES, HEAT FLOW
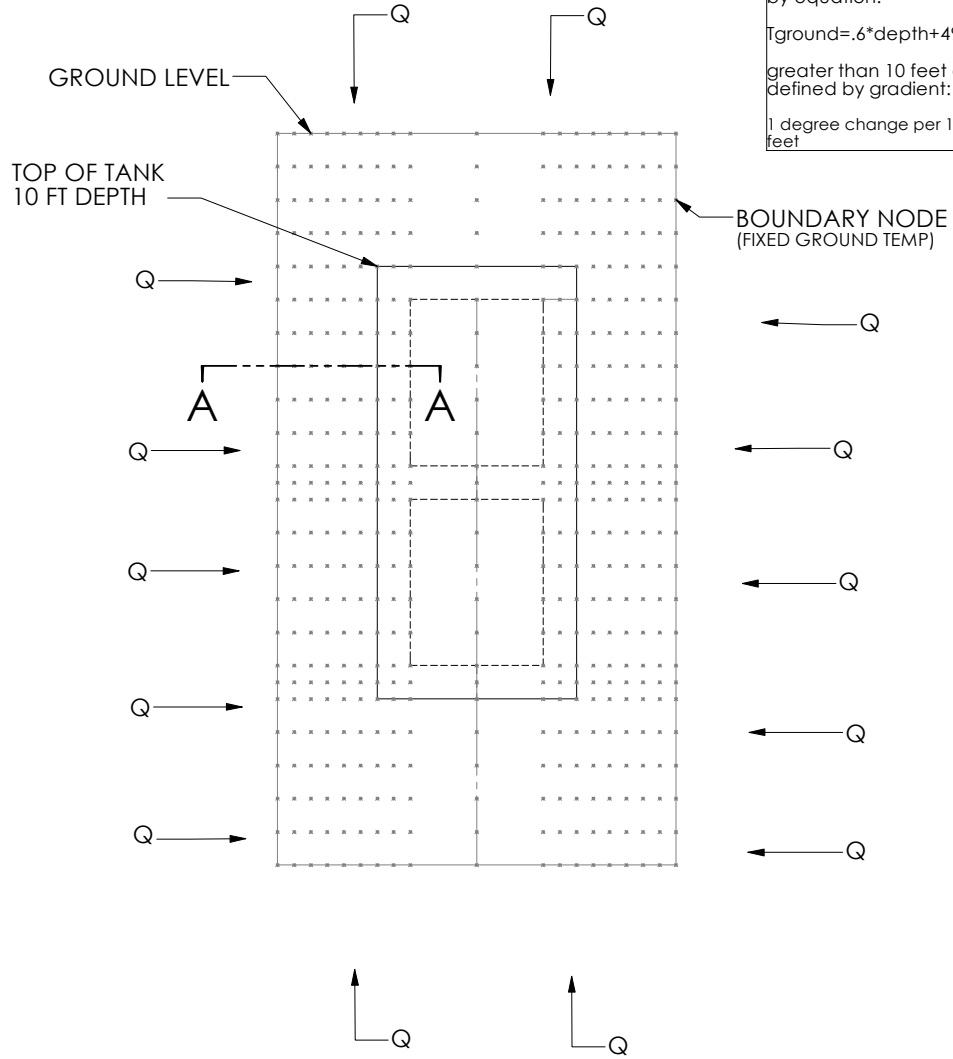
Q   Q

GROUND LEVEL

TOP OF TANK
10 FT DEPTH

BOUNDARY NODE
(FIXED GROUND TEMP)

Q

Q

Q   A   A   Q

Q   Q

Q   Q

Q   Q

Q   Q

			NAME	DATE	
		DIMENSIONS ARE IN FEET TOLERANCES: FRACTIONAL ± ANGULAR: MACH± BEND ± TWO PLACE DECIMAL ± THREE PLACE DECIMAL ±	DRAWN		
			CHECKED		
			ENG APPR.		
			MFG APPR.		
		MATERIAL	Q.A.		
			COMMENTS:		
NEXT ASSY	USED ON	FINISH			
APPLICATION		NOT TO SCALE DRAWING			

PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS
DRAWING IS THE SOLE PROPERTY OF
IVAN P ERCEG. ANY
REPRODUCTION IN PART OR AS A WHOLE
WITHOUT THE WRITTEN PERMISSION OF
IVAN P ERCEG IS PROHIBITED.

SIZE **A** DWG. NO.   FIG (2.2)   REV.

SCALE:1:128   WEIGHT: SHEET 1 OF 2

14

FIG (2.3)
SECTION A-A
TOP VIEW OF GEOTHERMAL SYSTEM

SAMPLE NODE
NUMBERS AND
CONNECTIONS

Q   Q

50

46
47   44
45
42
43   41

TANK WALL
(INNER)
TANK WALL
(OUTER)

GROUND LAYER

Q   Q

BOUNDARY LAYER
(FIXED GROUND TEMP)

		DIMENSIONS ARE IN INCHES		NAME	DATE		
		TOLERANCES:	DRAWN				
		FRACTIONAL ±	CHECKED				
		ANGULAR: MACH± BEND ±	ENG APPR.				
		TWO PLACE DECIMAL ±	MFG APPR.				
		THREE PLACE DECIMAL ±	Q.A.				
		MATERIAL	COMMENTS:				
NEXT ASSY	USED ON	FINISH					
APPLICATION		DO NOT SCALE DRAWING					

SIZE DWG. NO.   REV.
A   FIG (2.3)
SCALE:1:128   WEIGHT: SHEET 2 OF 2

2.3   ANALYZING A GEOTHERMAL SYSTEM

After researching ground temperature gradients and modeling a physical system the next step is to describe the heat flow using differential equations. The next chapter explains the steps involved.

CHAPTER III

MATHEMATICAL FORMULATION AND ANALYSIS

3.1 GOVERNING DIFFERENTIAL EQUATIONS/ BOUNDARY CONDITIONS [5]

The problem we are facing is heat flow in a cylindrical system in the z and r directions.

Fortunately we have differential equations to describe the heat flow. The two-

dimensional heat-conduction equation for cylindrical coordinates is;

$$\frac{\partial^2 T}{\partial r^2} + \frac{1}{r}\frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial z^2} = \frac{1}{\alpha}\frac{\partial T}{\partial \tau} \qquad [3\text{-}1]$$

With the following boundary conditions (please refer to Fig 2.2);

Example Node 50; fixed temperature; T(z) using equations [2-1] or [2-2].

Example Node 40; Inlet of Glycol Tank; T = C

As the glycol is circulating inside the tank we have convection between the glycol and

the tank wall at Example Node 42;

$$q = hA\left(T_{wall} - T_{glycol}\right) \qquad [3\text{-}2]$$

3.2   CONVERSION TO RC NETWORK [6]

By converting the differential equations into a RC network we simplify the calculations.

Each Node has a resistance and/or capacitance.
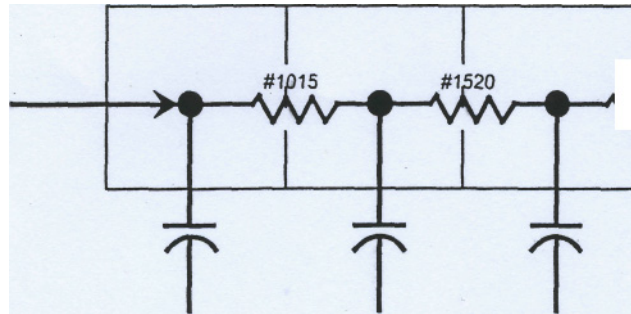
Fig (3.1) Sample RC Network

The following equations describe the heat flow in the radial (r) direction.

Area for heat flow in a cylindrical system radial direction;

$$A_r = 2\pi rL \qquad\qquad [3\text{-}3]$$

Fourier's law conduction radial direction;

$$q_r = -kA_r \frac{dT}{dr} \qquad\qquad [3\text{-}4]$$

Solving Fourier's law with the boundary conditions;

$$T = T_i \qquad \text{at} \qquad r = r_i$$

$$T = T_O \qquad \text{at} \qquad r = r_o$$

Solution to Fourier's law conduction radial direction;

$$q = \frac{2\pi kL(T_i - T_o)}{\ln(r_o / r_i)} \qquad\qquad [3\text{-}5]$$

The thermal resistance in this case is;

$$R_{th} = \frac{\ln(r_o / r_i)}{2\pi kL}$$

[3-6]

In our geothermal system we have a multiple layer cylindrical wall. For a multiple layer cylindrical wall the governing differential equation describing heat flow in the (r) direction is;

$$q_r = \frac{2\pi L(T_1 - T_\infty)}{\ln(r_2 / r_1)/k_A + \ln(r_3 / r_2)/k_B + ... + \ln(r_\infty / r_{\infty-1})/k_Z}$$

[3-7]

The following example illustrates a multiple cylindrical section;
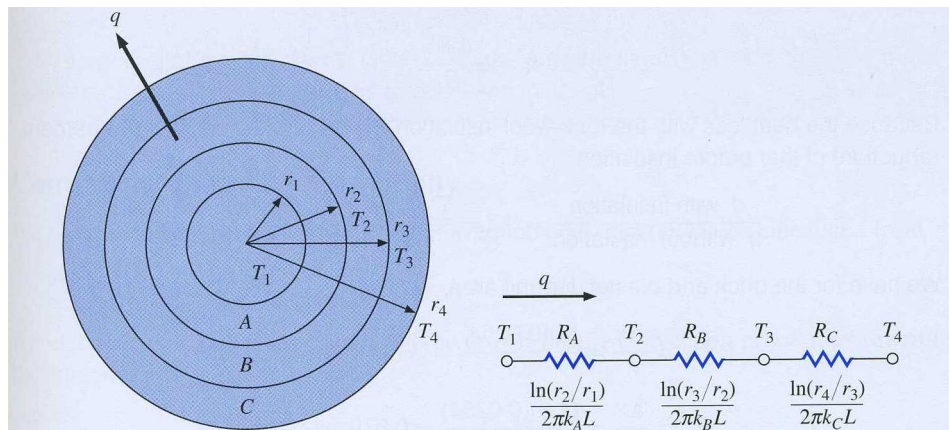
Fig (3.2) Sample multiple cylindrical section

Once we have our equations for our RC network we can begin to construct a program for SINDATHERMAL Analyzer.

19

3.3   SINDATHERMAL ANALYZER

SINDA/FLUINT is a comprehensive finite-difference, lumped parameter (circuit or network analogy) tool for heat transfer design analysis and fluid flow analysis in complex systems. We are using SINDA to describe our RC network and to achieve results. The next step in our geothermal study is to write a program in SINDA.

3.3.1   REGISTRAR

The first step in programming SINDA is to define our variables in the registrar. All of our variables in the nomenclature, whether independent or dependent, are defined in our registrar. The next step in programming SINDA is defining the nodes.

3.3.2   DEFINING NODES [7]

For analysis of the heat at several points on each layer, nodes are defined. The nodes represent the thermal capacitance of a volume of material. Nodes are defined on the exterior (ground) surface, and in the glycol tank. The heat conduction paths are called CONDUCTORS and are assigned arbitrary reference numbers. The thermal capacity (or CAPACITANCE) of each bar node is equal to the product of its density, volume and specific heat. The conductance of a conductor which represents a heat flow path through a material is equal to the product of the material's thermal conductivity and for cross-sectional area of the flow-path, divided by the length of the path.

Nodes represent a point at which energy is conserved. Each node has a single characteristic temperature "T". Nodes may represent the temperature of a finite volume of material. They may be used more abstractly to represent boundary conditions, mass less interfaces or edges. Each node is assigned a unique number and each conductor is assigned a unique number. The node numbers have an increment of 100, the conductance

numbers have an increment of 1000 for every layer going down and each layer is 10 feet deep.

BOUNDARY NODES

These nodes are the ones at the boundary of the system. In this case the very last ones on the ground are the boundary nodes. They are given a predefined capacitance value of zero and are given a negative node number. Boundary nodes have an infinite capacitance and hence usually represent sources or sinks, large masses, or ideally controlled temperature zones. The general format is:

N#, Ti, C

N# - Node number (negative)

Ti – Initial Temperature of the node

C – Capacitance (equal to zero)

ARITHMATIC NODES

These nodes are defined at the surface of intersection of two different materials. Arithmetic nodes have zero capacitance: energy flowing into arithmetic node must balance the energy flowing out at all times. Arithmetic nodes may be used to represent edges, interfaces, negligibly small masses (e.g., radiation shields or foils), and any other temperature to which no mass can or should be assigned. The capacitance is given some arbitrary negative number. The general format is:

N#, Ti, C

N# - Node number

Ti – Initial Temperature of the node

C – Capacitance (negative)

DIFFUSION NODES

Diffusion nodes have a finite capacitance "C", usually equal to the product of mass and specific heat (m*Cp). Diffusion nodes may represent a finite cell within a messed volume, or may represent a higher level component. The general format is:

N#, Ti, C

N# - Node number

Ti – Initial Temperature of the node

C – Capacitance (mass*specific heat)

CONDUCTORS

Conductors describe the means by which heat flows from one node to another. Each conductor has a single characteristic conductance "G" (inverse of resistance). Conductors represent energy paths via solid conduction, contact conduction, convection, advection, radiation, etc. The general format is:

#N, N1, N2, C

#N – Conductance number

N1, N2 – the nodes through which heat flows

C – Conductance

3.3.3   ASSIGNING NODES

Now that the definitions of nodes are known the next step is to assign a node definition to each node. The node can be one of the three following: boundary, diffusion, or arithmetic. Please refer to Appendix D page 113-114 for a map of the nodes to follow along while reading this section.

The boundary nodes in the RC network are the nodes in the x-direction along the ground surface, the nodes in the z-direction 20 feet radial from the center of the tank, as well as the nodes in the x-direction 20 feet in the z-direction from the bottom of the tank, or simply stated, the nodes running along the outer borders of the node map. The boundary nodes are nodes numbered; 181-189, 200-240, 20-160, and 151-159 respectively.

The arithmetic nodes are the nodes that are at the intersection of two materials and that are not boundary nodes. These nodes are found at the intersections of the glycol inside the tank and the tank wall and at the intersections of the tank wall and the earth. According to our node map these nodes are numbered 1, 3, 301, 303, 401, 403, 12, 14, 22, 24, 32, 34, 42, 44, 52, 54, 62, 64, 72, 74, 82, 84, 92, 94, 102, and 104. Basically these nodes are all around the inside and outside walls of the tank.

The rest of the nodes are diffusion nodes. These nodes are not boundary nodes and are not found at intersections of material but are the finite cells of the material. These nodes are found at the axial z center of the tanks and above and below the tanks at the axial z center. There is a column of these nodes at the center of the tank wall in order to accurately model the heat transfer through the wall. Finally these nodes are also found between the tank wall and the outer boundary 20 feet in the x-direction from the axial z center. These nodes divide up the earth into equal segments to model the heat flow accurately. For the 16 inch diameter and 30 foot long tank, these segments are approximately 3.75 feet wide in the radial (x) direction and 3 feet long in the z-direction. The dimensions of the node segments are dependent variables based on the tank dimensions and are programmed as such in SINDA.

Once the nodes are defined, the initial temperature of the nodes are a function of depth and given by equation 2.1 or 2.2. This is how the initial temperature of the nodes is determined. To note the glycol inlet point is assigned as a boundary node and the temperature of the glycol inlet temperature is an independent variable in the registrar. Next the conductors are defined between adjacent nodes wherever there is a heat flow.

3.3.4   EXAMPLE NODE

An example node is arbitrarily chosen as an example to illustrate how SINDA is programmed. The example node is 41. The input line to define node 41 in SINDA is;

$$41, Td5, CT5$$

Where 41 is the node number, Td5 is the initial temperature which is a dependent variable, and CT5 is the capacitance of the node. Since 41 is a positive number and CT5 is some non-negative non-zero capacitance we can see that node 41 is a diffusion node. Since the initial temperature is an independent variable all the related equations from the registrar will be listed in order to illustrate how the value for Td5 is calculated. These equations are all taken directly from the SINDA program in appendix A. The equations for Td5 are;

$$Td5 = 54.9 + FDG5/100.0$$

$$FDG5 = FDG4 + Dltk$$

$$FDG4 = FDG3 + Dltk$$

$$FDG3 = FDG2 + Dltk$$

$$FDG2 = FDG1 + Dltk/2.0$$

$$FDG1 = FDGt1$$

$$FDGt1 = FDGa1$$

24

$$FDGa1 = 10.0$$

$$Dltk = Ltk/Nttk$$

$$Ltk = 20.0$$

$$Nttk = 10.0$$

Calculating the equations and variables the initial temperature of node 41 is 55.7 °F.

Next the capacitance is defined. Again all the related equations will be taken from

SINDA to illustrate how the value for the capacitance is obtained. The equations for CT5

are;

$$CT5 = \frac{(\pi * Dti^2)}{4} * dltk * Cpgl * Rhoegl$$

$$Dti = DTII/12.0$$

$$DTII = DTOI -2.0 * Tmw$$

$$DTOI = 16.0$$

$$Tmw = 1.455$$

$$Dltk = Ltk/Nttk$$

$$Ltk = 20.0$$

$$Nttk = 10.0$$

$$Cpgl = .796$$

$$Rhoegl = 67.61$$

Calculating the equations and variables the initial capacitance of node 41 is 82.85

BTU/min-°F.

Now the conductors will be defined where there is a heat flow. Node 31 is connected

above node 41 in the z-direction. Node 51 is connected below node 41 in the z-direction.

And finally node 42 is connected to node 41 in the x-direction. These are the three

connections to node 41. The format for the SINDA conductance input is;

G#, #G, IG, NA, INA, NB, INB, G

Where G# is the actual number of the first conductor to be generated, #G is the total

number of conductors to be generated, IG is the increment to be added to the current

conductor number to form the conductor number of the next conductor, NA and NB are

the actual node numbers of the pair of nodes connected by the first conductor, INA and

INB are the increments to be added to NA and NB to arrive at the node numbers for the

pair of nodes connected by the next conductor, and finally G is the conductance of all

conductors.

The vertical conductance for node 41 is defined by the following SINDA input;

GEN 36, 10, 1, 11, 10, 12, 10, Ftrc

This input defines the entire vertical conductance for all nodes along the axial z-center of

the tank.

The equation for the conductance Ftrc is;

$$Ftrc = 2.0 * \pi * \frac{Kfgl}{\ln(\frac{Rti}{Rtc})}$$

The related equations and variables for Ftrc are;

$$Kfgl = KKf/60.0$$

$$KKf = 0.217$$

$$Rti = Dti/2.0$$

$$Dti = DTII/2.0$$

$$DTII = DTOI - 2.0*Tmw$$

$$DTOI = 16.0$$

$$Tmw = 1.778$$

Solving for Ftrc, the value for Ftrc is 0.005 BTU/min-°F^4.

Next the conductor in the x direction is defined by the following SINDA input;

$$GEN\ 201,\ 4,\ 1,\ 11,\ 10,\ 21,\ 10\ Ftv*2.0$$

The equation for the conductance Ftv is;

$$Ftv = \frac{\pi * Kfgl * \dfrac{Dto^2}{4.0}}{\dfrac{Dltk}{2.0}}$$

The related equations and variables for Ftv are;

$$Kfgl = KKf/60.0$$

$$KKf = 0.217$$

$$Dto = DTOI/12.0$$

$$DTOI = 16.0$$

$$Dltk = Ltk/Nttk$$

$$Ltk = 30.0$$

$$Nttk = 5.0$$

Solving for 2*Ftv, the value for 2*Ftv is 0.014 BTU/min-°F^4.

This concludes the sample node example. This was done with every single node in the SINDA program.

3.3.5   TEST RUNS

After the nodes and conductors are defined, then the finishing touches to the program are completed we have our RC network and are ready to run the program and achieve results.

By editing the registrar we can adjust our independent variables. The independent variables of the geothermal study are;

Tank Height

Tank Radius

Glycol Inlet Temperature

Glycol Mass Flow Rate

The goal of our test runs is to see how much heat can be transferred with the surrounding earth in terms of Btu/hour. The next chapter describes our various test runs and results.

CHAPTER IV

RESULTS AND DISCUSSION

This chapter discusses the test runs and their results. There is no benefit to using the two

tank design due to significant heat transfer between the two tanks. Below the results are

listed.

Fig 4.1 through Fig 4.5 studies the effect the tank diameter has on heat transfer.

These tests conclude that a larger diameter doesn't hinder heat transfer between the

glycol and the earth. A larger diameter is advantageous due to the fact more glycol or

other fluid can be stored. The largest diameter tested is 16 in, the smallest 5 in.

These tests conclude that a tank with a minimum diameter of 12 in and tank height of 30

feet or a minimum diameter of 16 in and tank height of 20 feet are needed to sustain a

perpetual heat transfer with glycol flying at 4 gpm at a temperature of 30 °F. The two

tanks provide a heat transfer of 40,000 BTU/HR and 50,000 BTU/HR respectively.

Fig 4.6 studies the effect the mass flow rate has on heat transfer.

This study shows heat transfer for a tank 30 feet deep and 16 inches in diameter and various mass flow rates. The heat transfer does not significantly degrade over time for a glycol mass flow rate of less than 6 gpm and 30 °F and provide 60,000 BTU/HR.

Fig 4.7 studies the effect the tank height has on heat transfer.

The tank should be 30 feet deep otherwise heat transfer is significantly degraded over time for glycol mass flow rate of 4 gpm and a temperature of 40 °F. This mass flow rate and temperature provide us with 25,000 BTU/HR.

Fig 4.8 studies the effect the glycol return temperature has on heat transfer.

This study shows that the geothermal system can handle temperature typically found in the summer and winter, between 30 °F and 70°F at a mass flow rate of 4 gpm.

Fig 4.9 studies the largest tank with a 20 foot depth at different mass flow rates.

This study concludes that the tank needs to be 30 feet in length to function at higher mass flow rates. The best result is a mass flow rate of 12 gpm and a temperature of 30°F results in sustained heat transfer of 100,000 BTU/HR.

Fig 4.10 studies our largest tank at different tank heights.

This study shows how heat transfer is effected by tank heights. The maximum tank height tested was 40 feet. This tank provided 120,000 BTU/HR at a glycol mass flow rate of 12 gpm and temperature of 30°F.

Fig 4.11 studies our largest tank diameter of 16 inches with a 20 foot depth at different glycol inlet temperatures. This study confirms that a 20 foot depth is not deep enough and shows the heat transfer degradation.

Our conclusion is that a tank that is 16 inch in diameter and has a length of 30 feet can sufficiently heat an average home in the winter or cool it in the summer.

Fig (4.1) - Geothermal System Perfomance
Tank Dimensions: Diameter = 5 in, Height = variable
Glycol Return Temp = 30 F, Mass flow rate = 4 gpm
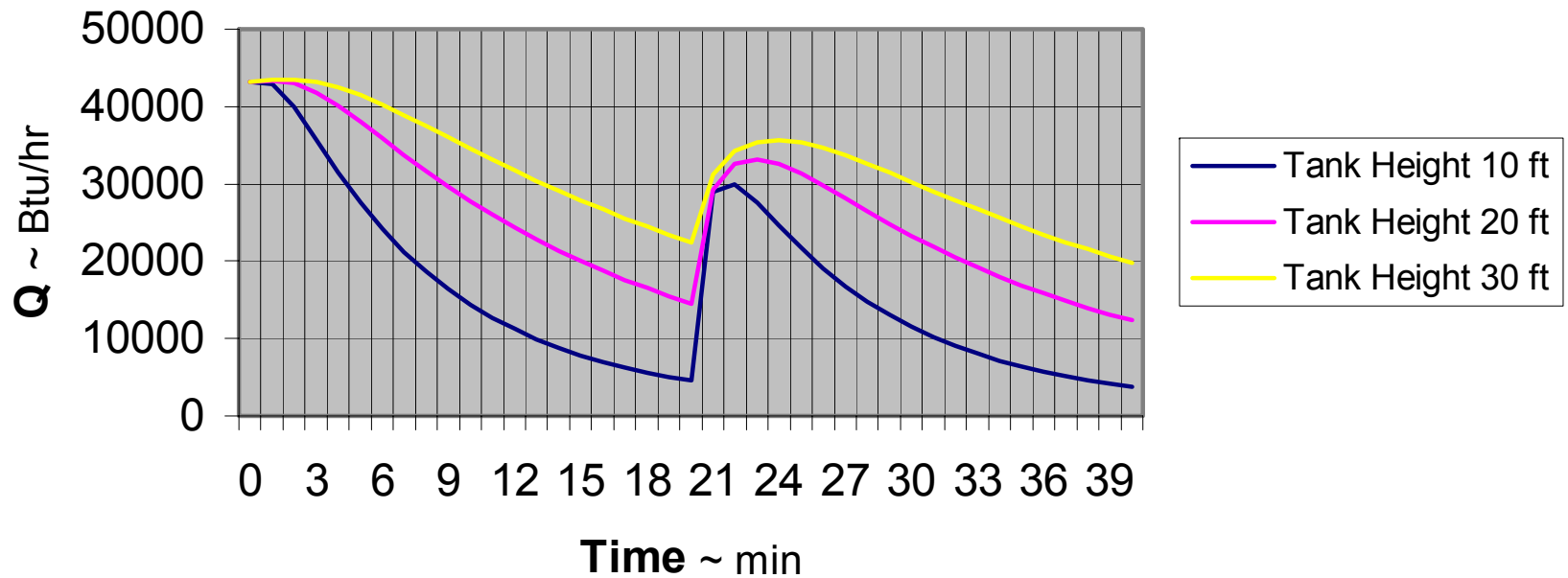
Fig (4.2) - Geothermal System Perfomance

Tank Dimensions: Diameter = 8 in, Height = variable
Glycol Return Temp = 30 F, Mass flow rate = 4 gpm

Legend:
- Tank Height 10 ft
- Tank Height 20 ft
- Tank Height 30 ft

X-axis: **Time** ~ min
Y-axis: **Q** ~ Btu/hr
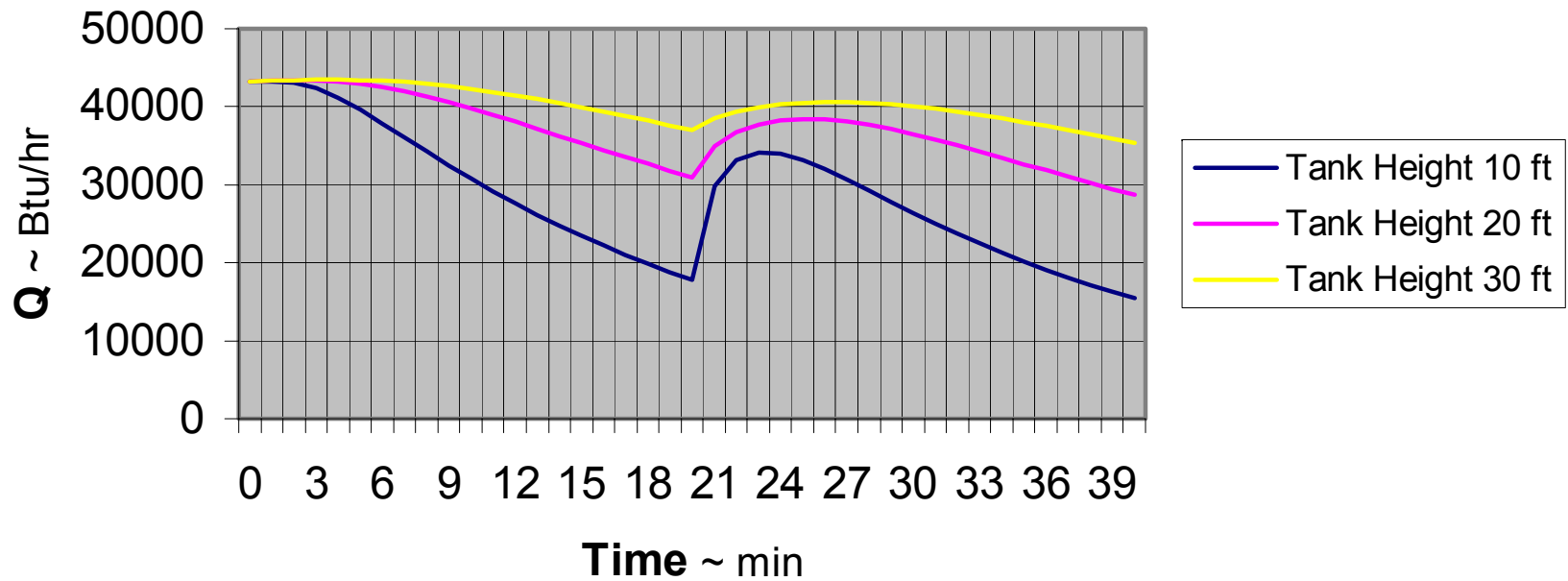
Fig (4.3) - Geothermal System Perfomance

Tank Dimensions: Diameter = 10 in, Height = variable
Glycol Return Temp = 30 F, Mass flow rate = 4 gpm
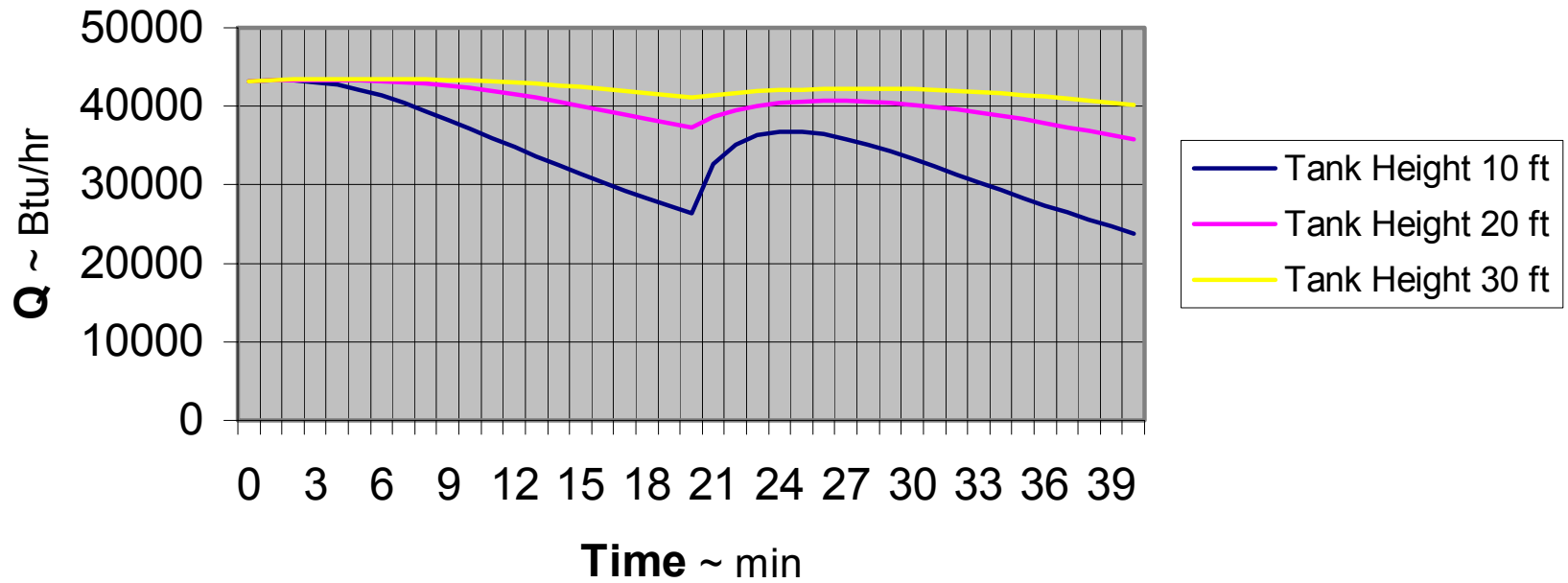
Fig (4.4) - Geothermal System Perfomance

Tank Dimensions: Diameter = 12 in, Height = variable
Glycol Return Temp = 30 F, Mass flow rate = 4 gpm
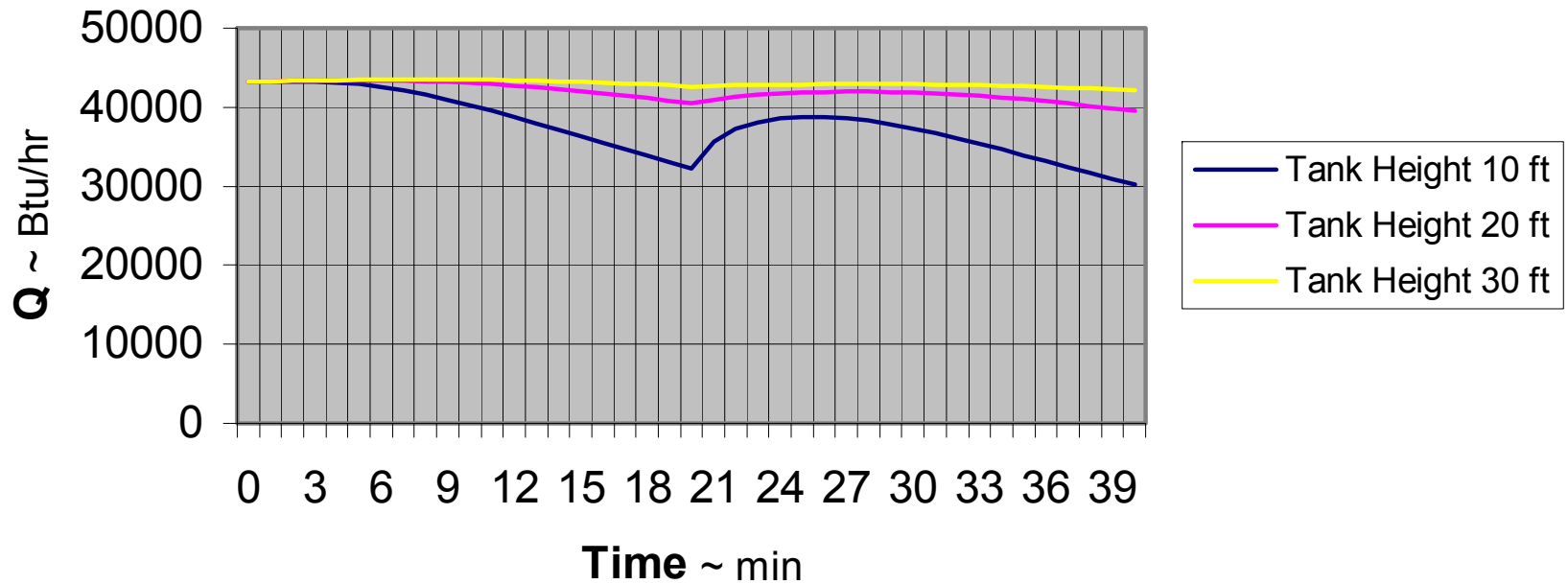
Fig (4.5) - Geothermal System Perfomance

Tank Dimensions: Diameter = 16 in, Height = variable
Glycol Return Temp = 30 F, Mass flow rate = 4 gpm
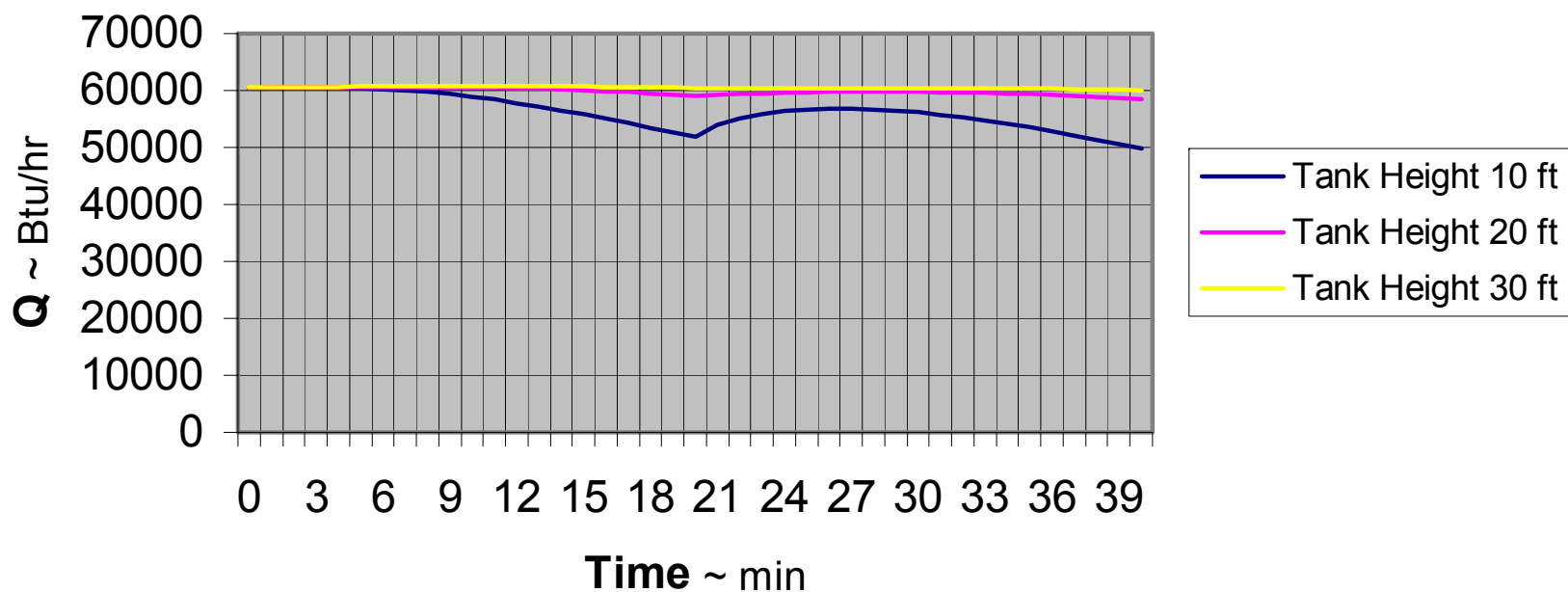
Fig (4.6) - Geothermal System Perfomance
Tank Dimensions: Diameter = 16 in, Height = 30 ft
Glycol Return Temp 30 F, Mass flow rate = variable
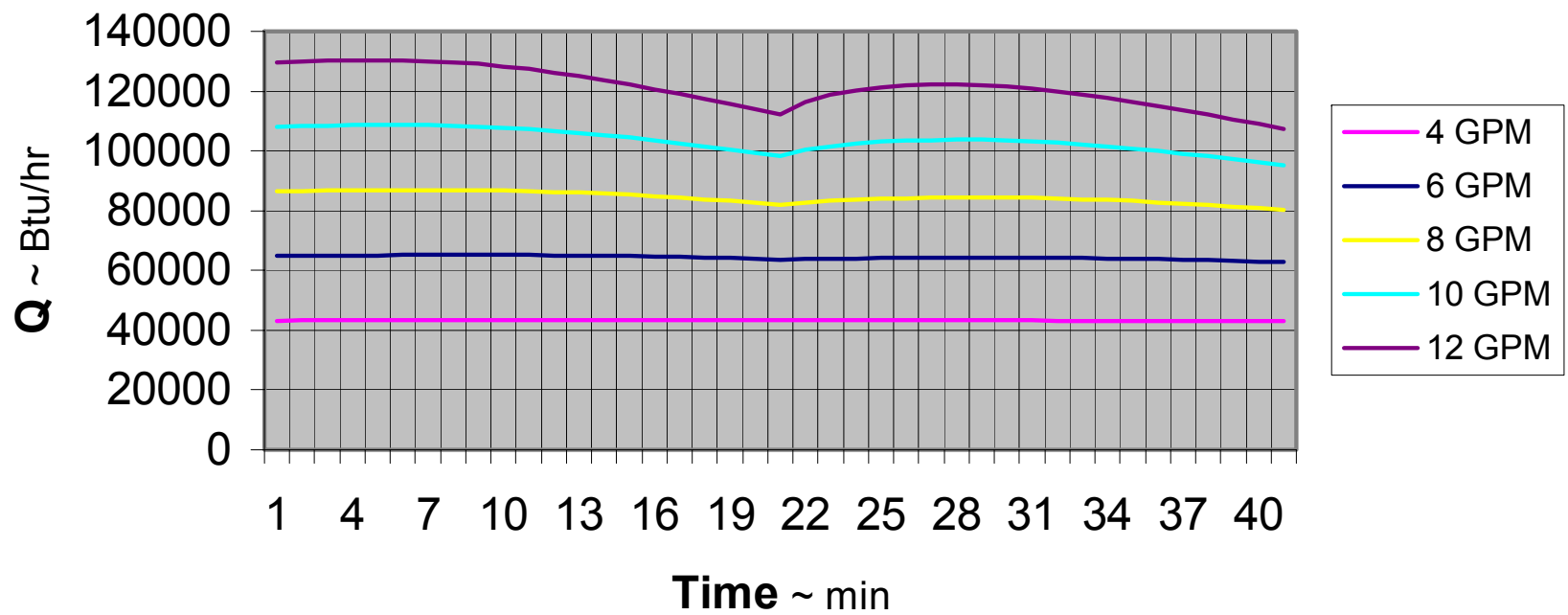
Fig (4.7) - Geothermal System Perfomance
Tank Dimensions: Diameter = 16 in, Height = variable
Glycol Return Temp 40 F, Mass flow rate = 4 gpm
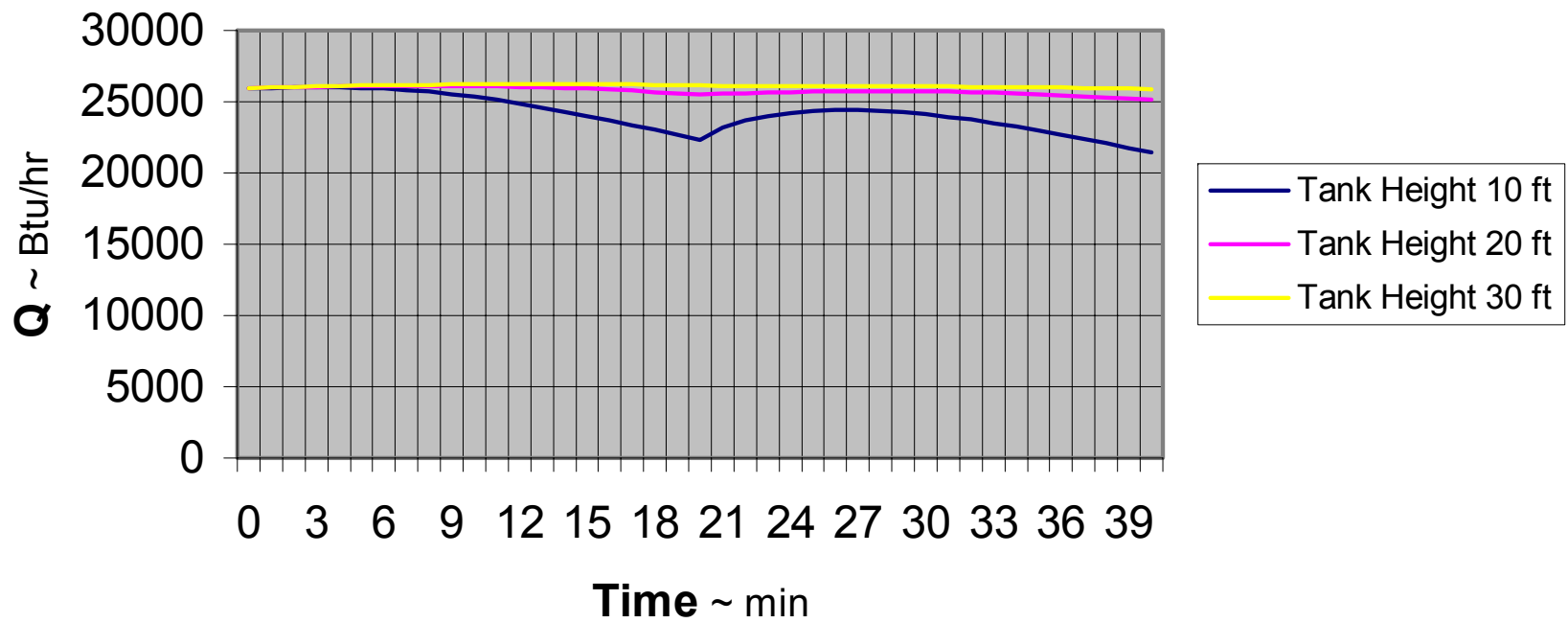
Fig (4.8) - Geothermal System Perfomance
Tank Dimensions: Diameter = 16 in, Height = 30 ft
Glycol Return Temp = variable, Mass flow rate = 4 gpm

Legend:
— Glycol Return Temp 30 F
— Glycol Return Temp 40 F
— Glycol Return Temp 60 F
— Glycol Return Temp 70 F

Y-axis: **Q** ~ Btu/hr
X-axis: **Time** ~ min
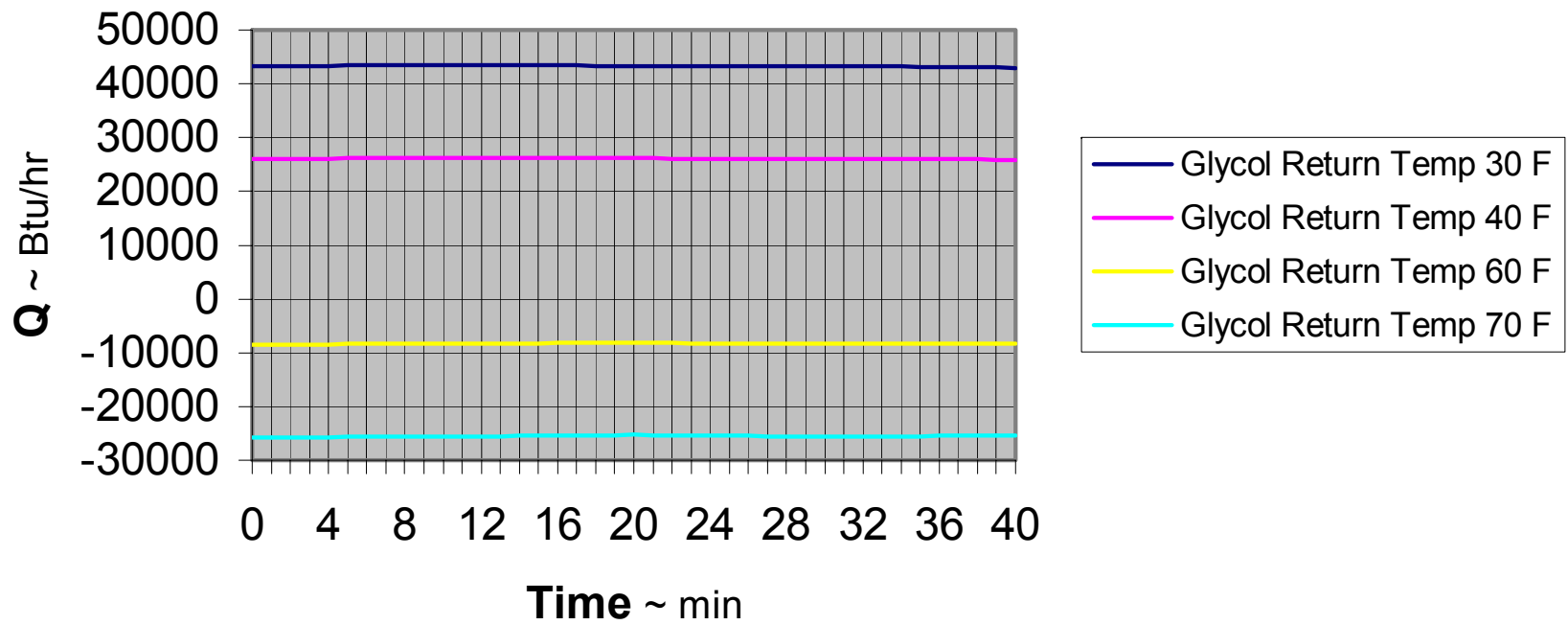
Fig (4.9) - Geothermal System Perfomance

Tank Dimensions: Diameter = 16 in, Height = 20 (ft)
Glycol Return Temp = 30 F, Mass flow rate = variable
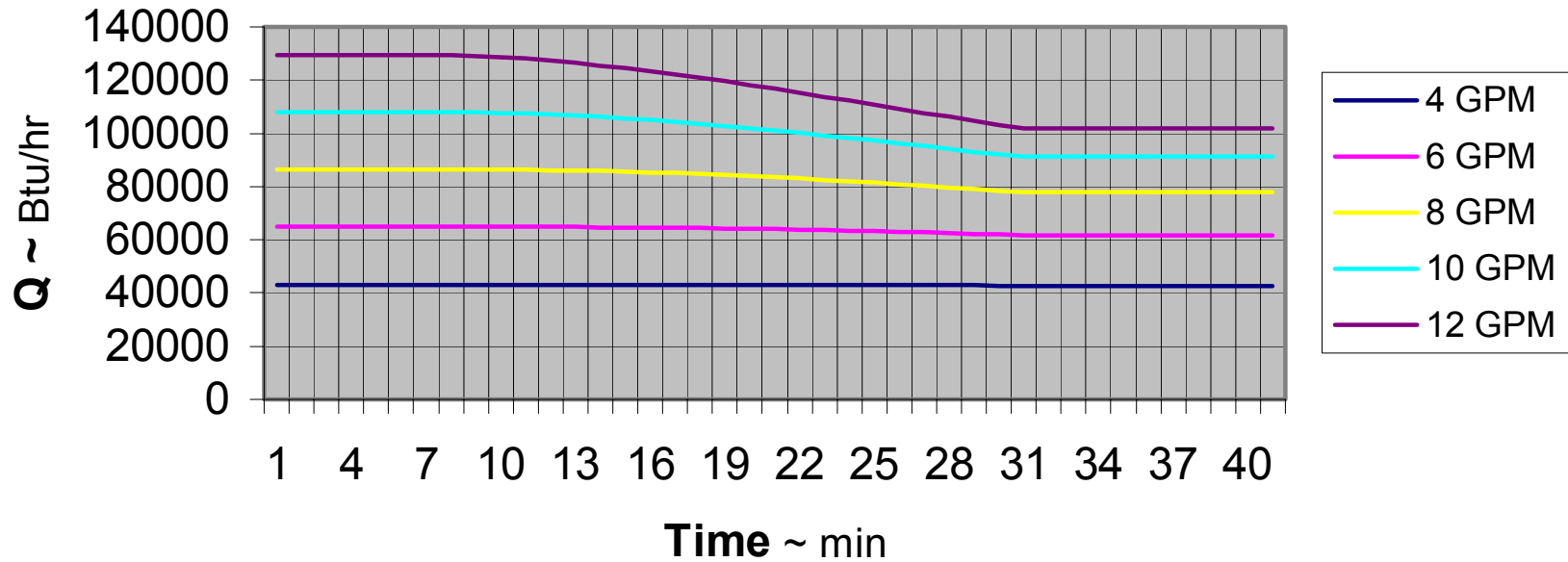
Fig (4.10) - Geothermal System Perfomance

Tank Dimensions: Diameter = 16 in, Height = variable
Glycol Return Temp = 30 F, Mass flow rate = 12 gpm
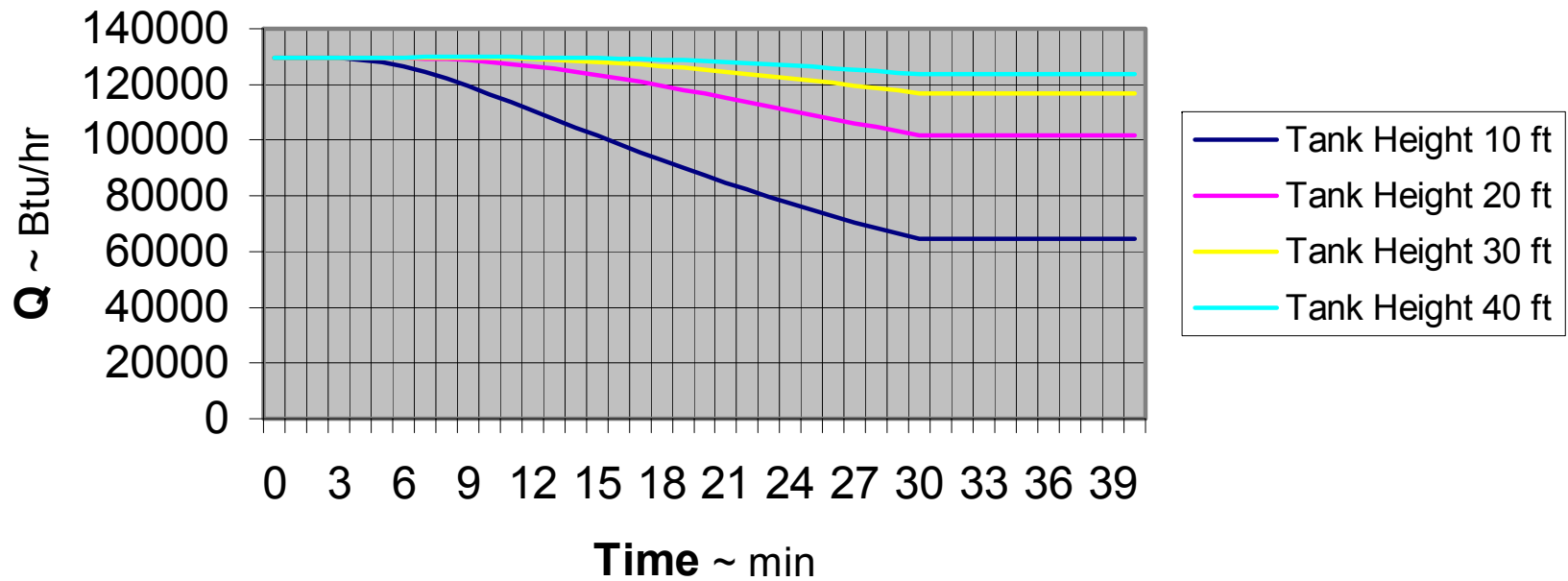
Fig (4.11) - Geothermal System Perfomance

Tank Dimensions: Diameter = 16 in, Height = 20 ft
Glycol Return Temp = variable, Mass flow rate = 4 gpm

Legend:
- Glycol Inlet Temp 30 F
- Glycol Inlet Temp 40 F
- Glycol Inlet Temp 50 F
- Glycol Inlet Temp 60 F
- Glycol Inlet Temp 70 F

Y-axis: Q ~ Btu/hr
X-axis: Time ~ min
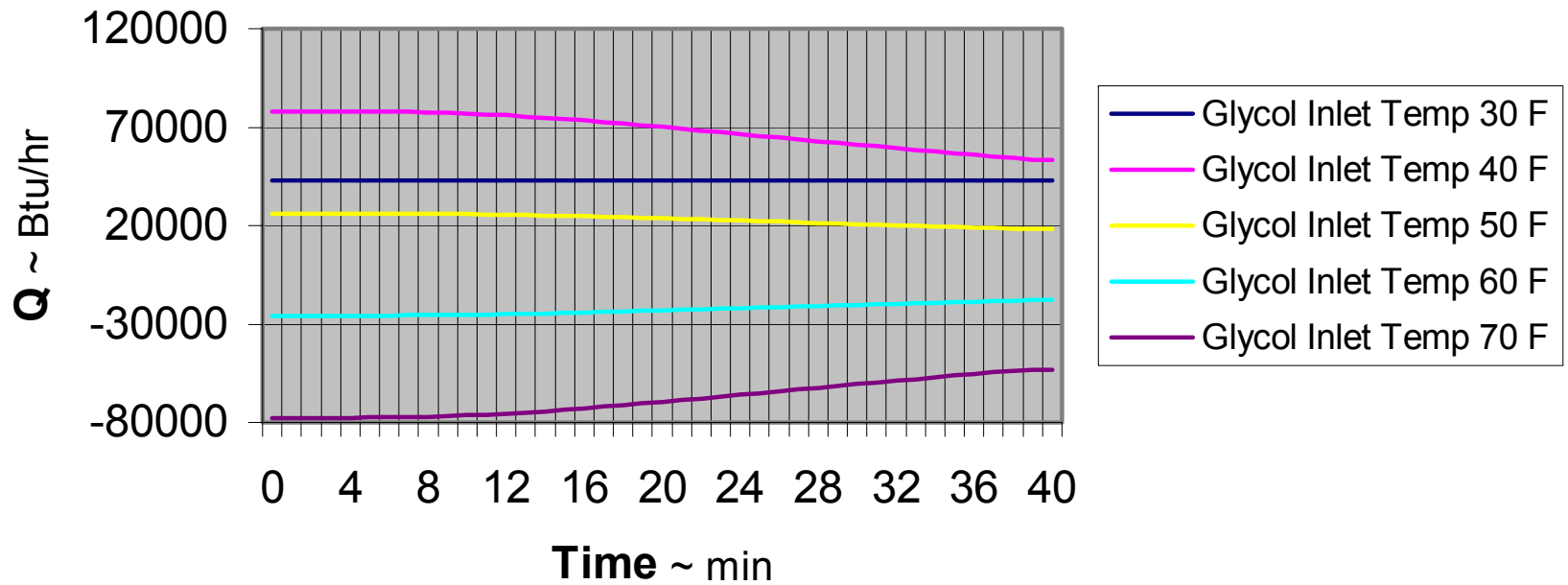
CHAPTER V

CONCLUDING REMARKS

There are several reasons why our geothermal system is superior to the systems in use in

industry. Our system has superior heat flow as proven in our figures. The installation cost

is lower in our system, the main reason is that to install our system there is only a need to

dig 40 feet deep instead of 300 feet. The other important reason is only a vertical hole is

necessary rather than trenches. Operating costs for our system our lower due to the fact

that our fluid is only pumped 10 feet vertically instead of 300 feet saving on vertical

distance and drag forces. The final reason our system is superior is that it is more

compact than current systems. A 16 inch or so vertical hole 40 feet long is all the space

our system occupies allowing it to be installed easier and in tighter spaces than current

geothermal systems. In conclusion, our system is superior to a geothermal system in use

in industry based on the previously mentioned reasons.

BIBLIOGRAPHY

(1)   Anuta, Joe (2006-03-30). "Probing Question: What heats the earth's core?", physorg.com. Retrieved on 2007-09-19.

(2)   Purnender Kancharla, "Geothermal System using a Central Tank", Cleveland State University, 2 - 8, (2007)

(3)   "International Ground Source Heat Pump Association: About Us". www.wikipedia.com. Retrieved on 2008-08-05.

(4)   Rohit Kothlapuram, "Geothermal System of CSU Administration Building", Cleveland State University, 14, (2007)

(5)   J.P. Holman, "Heat Transfer Ninth Edition", 5, The McGraw-Hill Companies Inc., (2002)

(6)   J.P. Holman, "Heat Transfer Ninth Edition", 29, The McGraw-Hill Companies Inc., (2002)

(7)   Purnender Kancharla, "Geothermal System using a Central Tank", Cleveland State University, 10, (2007)

APPENDICES

APPENDIX A: SINDA THERMAL PROGRAM

HEADER OPTIONS DATA
TITLE TWO TANK GEOTHERMAL SYSTEM
    OUTPUT   = GEOTANK2.OUT
    MODEL   = GTANK2
HEADER REGISTERED DATA,SUB1
    TSW = 20.0 $ TANK SWITCH TIME, MINUTES
    PI = 3.14159 $ CONSTANT

C   CIRCULATING GLYCOL FLUID THERMAL PHYSICAL PROPERTIES AND MASS FLOW RATE

    Cpgl   = 0.796 $ FLUID SPECIFIC HEAT, btu/lb-F
    VIS   = 6.05 $ DYNAMIC VISCOCITY,centipoise
    VISCgl = 0.0403182*VIS $ DYNAMIC VISCOCITY, lbm/ft-min
    KKf   = 0.217 $ GLYCOL THERMAL CONDUCTIVITY, btu/hr-ft-F
    Kfgl = KKf/60.0 $ GLYCOL THERMAL CONDUCTIVITY, btu/min-ft-F
    Rhoegl = 67.61 $ GLYCOL FLUID DENSITY,lbm/cu.ft
    GLFR = 10.00 $ GLYCOL FLUID MASS FLOW RATE,gpm
    GM   = GLFR*(231.0/1728.0)*Rhoegl $ GLYCOL FLOW RATE,lbm/min
    Tgi   = 30.0 $ GLYCOL INLET TEMPERATURE,deg. F

C   TANK THERMAL PHYSICAL PROPERTIES

    Rhoetk = 59.92883 $ TANK WALL DENSITY,lbm/cu-ft
    KKt   = 0.1907 $ TANK WALL THERMAL CONDUCTIVITY,btu/hr-ft-F
    Ktk   = KKt/60.0 $ TANK WALL THERMAL CONDUCTIVITY,btu/min-ft-F
    Cptk = .50159 $ TANK WALL SPECIFIC HEAT,BTU/lb-F

C   TUBE THERMAL PHYSICAL PROPERTIES

    Kt   = 0.1907 $ TUBE WALL THERMAL CONDUCTIVITY,btu/hr-ft-F
    Ktb   = Kt/60.0 $ TUBE WALL THERMAL CONDUCTIVITY,btu/min-ft-F

```
        Cptb   = 0.50159                    $ TUBE  SPECIFIC HEAT,BTU/lb-F
        Rhoetb = 59.92883                     $ TUBE DENSITY,lbm/cu-ft


C     GROUND THERMAL PHYSICAL PROPERTIES


        Kgi   = .30                          $ GROUND  THERMAL CONDUCTIVITY,BTU/hr-ft-F
        Kg    =Kgi/60.0                       $ GROUND  THERMAL CONDUCTIVITY,BTU/min-ft-F
        Cpg   = .44                          $ GROUND  SPECIFIC HEAT,BTU/lb-F
        Rhoeg = 128.0                         $ GROUND  DENSITY,  lbm/cu-ft


C     TANK DIMENSIONS


        Tmw    = 1.778                       $ TANK MINIMUM WALL THICKNESS,inch
        DTOI  = 16.00                        $ TANK OUTSIDE DIAMETER,inch
        Ltk   = 30.0                        $ TANK HEIGHT LENGTH,ft
        Ttmb  = TmW/12.0                      $ TANK TOP MID AND BOTTOM THICKNESS,ft
        Dto   = DTOI/12.0                     $ OUTER TANK DIAMETERS,ft
        DTII  = DTOI-2.0*TMW                   $ INSIDE TANK DIAMETER,inch
        Dti   = DTII/12.0                     $ INNER TANK DIAMETER,ft
        Nttk  = 5.0                         $ TANK NUMBER OF NODAL INCREMENTS
        Rto   = Dto/2.0                      $ TANK OUTER RADIUS,ft
        Rti   = Dti/2.0                      $ TANK INNER RADIUS,ft
        Rtm    = (Rti+Rto)/2.0                $ MID-WALL RADIUS,ft
        Rtc   = Rti/100.0                     $ CENTER TANK RADIUS,ft
        Dltk  = Ltk/nttk                      $ VERTICAL TANK NODAL INCREMENT,ft


C     TUBE DIMENSIONS
        Twt   = .119                         $ TUBE MINIMUM WALL THICKNESS,inch
        Dpoi  = 1.315                         $ OUTER TUBE DIAMETER,Inch
        Dpii  = Dpoi-2.0*Twt                   $ INNER TUBE DIAMETER,Inch
        Dpo   = Dpoi/12.0                     $ OUTER TUBE DIAMETER,ft
```

```
Dpi   = Dpii/12.0                $ INNER TUBE DIAMETER,ft
Ltb   = Ltk                      $ PIPE LENGTH IS THE SAME AS TANK HEIGHT, ft
Ntb   = 5.0                      $ TUBE  NUMBER OF NODE INCREMENTS
Rpo   = Dpo/2.0                  $ TANK OUTER RADIUS,ft
Rpi   = Dpi/2.0                  $ TANK INNER RADIUS,ft
Dltb  = Ltb/Ntb                  $ TUBE  NODAL INCREMENTAL LENTGH,ft

C     GROUND DIMENSIONS SURROUNDING TANK

FDGa1 = 10.0                     $ GROUND DEPTH TO TOP OF FIRST TANK,ft
Lg1   = 5.0                      $ GROUND VERTICAL HEIGHT ABOVE TOP OF TANK,ft
Lg2   = 5.0                      $ GROUND VERTICAL HEIGHT BELOW BOTTOM OF TANK,ft
Lt1   = Ltk                      $ SINGLE TANK HEIGHT,ft
Rgrad = 15.0                     $ GROUND OUTER RADIUS SURROUNDING THE TANK,FT

C     GROUND DEPTH DIMENSIONAL CALCULATIONS START HERE

FDGt1  = FDGa1                   $ GROUND DEPTH TO TOP OF TANK,FT
FDGt2  = FDGt1+Lt1               $ GROUND DEPTH TO BOTTOM OF 1st TANK,ft
FDGt3  = FDGt2+Lt1               $ GROUND DEPTH TO BOTTOM OF SECOND TANK,ft
FDGb1  = FDGt3+Lg2               $ GROUND DEPTH BELOW SECOND TANK BOTTOM,ft
Dla1   = Lg1/nttk                $ VERTICAL GROUND NODAL INCREMENT ABOVE TANK TOP,ft
Dlb1   = Lg2/nttk                $ VERTICAL GROUND NODAL INCREMENT BELOW TANK BOTTOM,ft

C     TUBE HEAT TRANSFER CALCULATIONS

Prgl  = Cpgl*VISCgl/Kfgl         $ PRANDTL NUMBER
Redtb = 4.0*GM/(PI*Dpi*VISCgl)   $ REYNOLD S NUMBER
Nutb  = 0.023*(Redtb)**0.8*(Prgl)**0.4   $ NUSSELT NUMBER
Htb   = Nutb*Kfgl/Dpi            $ CONVECTION HEAT TRANSFER COEF.BTU/min-ft2-F
Aptb  = PI*Dpi*Dltb              $ NODE INNER TUBE AREA, sq-ft
```

```
      GCtb  = Htb*Aptb                    $ WALL CONVECTION CONDUCTANCE
      Fcontb = GM*Cpgl                     $ MOVING FLUID CONDUCTANCE
      NFVoltb= PI*Dpi**2*Dltb/4.0          $ NODE FLUID VOLUME
      Captb = Rhoegl*NFVoltb*Cpgl          $ TUBE NODE CAPACITY
      Fgtb  = 2.0*PI*Ktb*Dltb/ln(Dpo/Dpi)    $ RADIAL TUBE WALL CONDUCTANCE


C     TANK HEAT TRANSFER CALCULATIONS

      Prtk   = Cpgl*Viscgl/Kfgl            $ PRANDTL NUMBER
      Redtk  = 4.0*GM/(PI*Dti*VIScgl)      $ REYNOLD S NUMBER
      Nutk  = 0.023*(Redtk)**0.8*(Prgl)**0.4    $ NUSSELT NUMBER
      Htk   = Nutk*Kfgl/Dti               $ CONVECTION HEAT TRANSFER COEF.BTU/min-ft2-F
      Aptk  = PI*Dti*Dltk                  $ NODAL INNER TANK AREA, sq-ft
      FCOntk = GM*Cpgl                     $ MOVING FLUID CONDUCTANCE
      NFVoltk= PI*Dti**2*Dltk/4.0          $ NODE FLUID VOLUME
      Captk = Rhoegl*NFVoltk*Cpgl          $ NODE GLYCOL FLUID CAPACITANCE
      Ftmb  = Ktk*PI*Dti**2/4.0/Ttmb       $ TANK TOP,MID-SECTION AND BOTTOM CONDUCTANCE
      Ftmw  = 2.0*PI*Ktk/ln(Rto/Rti)       $ TANK MID-WALL RADIAL CONDUCTION
      Ftcon = Htk*Aptk                     $ TANK WALL CONVECTION CONDUCTANCE
      Ctwc  = PI*(Rto**2-Rti**2)*Dltk*Rhoetk*Cptk $ TANK WALL CAPACITANCE
      Ctsc  = (PI*Dti**2/4.0)*Dltk*Rhoetk*Cptk   $ TANK TOP-MID-BOTTOM WALL CAPACITANCE
      Ftfk  = GM*Cpgl                      $ MOVING FLUID CONDUCTANCE
      Ftrc  = 2.0*PI*Kfgl/ln(Rti/Rtc)      $ TANK RADIAL GLYCOL CONDUCTION


c     GROUND CONDUCTANCES AND CAPACITANCE CALCULATIONS START HERE



C     GROUND NODAL RADIAL DIMENSIONS SURROUNDING THE TANK

      DVt=PI*(Rgrad**2-Rto**2)*Dltk/nttk
      Rx1=Rto
```

49

```
      Rx2=SQRT(DVt/(PI*DLtk)+Rx1**2)
      Rx3=SQRT(DVt/(PI*DLtk)+Rx2**2)
      Rx4=SQRT(DVt/(PI*DLtk)+Rx3**2)
      Rx5=SQRT(DVt/(PI*DLtk)+Rx4**2)
      Rx6=SQRT(DVt/(PI*DLtk)+Rx5**2)


C     GROUND RADII

      R1   =  Rx1
      R2   = (Rx1 + Rx2)/2.0
      R3   = (Rx2 + Rx3)/2.0
      R4   = (Rx3 + Rx4)/2.0
      R5   = (Rx4 + Rx5)/2.0
      R6   = (Rx5 + Rx6)/2.0
      R7   = Rgrad


C     GROUND INITIAL TEMPERATURE CALCULATIONS AT VARIOUS DEPTHS

C     GROUND NODAL DISTANCES OF GROUND ABOVE THE TANK

      Dag1=FDGt1-Lg1
      Dag2=Dag1+Dla1/2.0
      Dag3=Dag1+Dla1
      Dag4=Dag1+Dla1
      Dag5=Dag1+Dla1
      Dag6=Dag1+Dla1
      Dag7=Dag1+Dla1/2.0


C     GROUND NODAL DISTANCES FROM TOP OF TANK,VERTICALLY DOWN

      FDG1=FDGt1
```

```
      FDG2=FDG1+Dltk/2.0
      FDG3=FDG2+Dltk
      FDG4=FDG3+Dltk
      FDG5=FDG4+Dltk
      FDG6=FDG5+Dltk
      FDG7=FDG6+Dltk
      FDG8=FDG7+Dltk
      FDG9=FDG8+Dltk
      FDG10=FDG9+Dltk
      FDG11=FDG10+Dltk
      FDG12=FDG11+(Dltk)/2.0
C
      FDG13=FDG12+Dlb1/2.0
      FDG14=FDG13+Dlb1
      FDG15=FDG14+Dlb1
      FDG16=FDG15+Dlb1
      FDG17=FDG16+Dlb1
      FDG18=FDG17+Dlb1/2.0


C     INITIAL GROUND TEMPERATURES AT VARIOUS DEPTHS BELOW TANK TOP

C     GROUND ABOVE TANK

      Tda1 = 0.6*Dag1+49.0
      Tda2 = 0.6*Dag2+49.0
      Tda3 = 0.6*Dag3+49.0
      Tda4 = 0.6*Dag4+49.0
      Tda5 = 0.6*Dag5+49.0
      Tda6 = 0.6*Dag6+49.0
      Tda7 = 0.6*Dag7+49.0
```

```
C     TANK TOP AND DOWN GROUND TEMPERATURE

      Td1  = 0.6*FDG1+49.0        $ IF GROUND DEPTH IS LESS OR EQUAL TO 10 ft
      Td2  = 54.9+FDG2/100.0
      Td3  = 54.9+FDG3/100.0
      Td4  = 54.9+FDG4/100.0
      Td5  = 54.9+FDG5/100.0
      Td6  = 54.9+FDG6/100.0
      Td7  = 54.9+FDG7/100.0
      Td8  = 54.9+FDG8/100.0
      Td9  = 54.9+FDG9/100.0
      Td10 = 54.9+FDG10/100.0
      Td11 = 54.9+FDG11/100.0
      Td12 = 54.9+FDG12/100.0


C     TANK BOTTOM AND DOWN

      Tdb1 = 54.9+FDG13/100.0
      Tdb2 = 54.9+FDG14/100.0
      Tdb3 = 54.9+FDG15/100.0
      Tdb4 = 54.9+FDG16/100.0
      Tdb5 = 54.9+FDG17/100.0
      Tdb6 = 54.9+FDG18/100.0


C     CAPACITANCE CALCULATIONS OF GROUND ABOVE TANK TOP

      Ca1  = 1.0
      Ca2  = (PI*Dto**2/4.0)*dla1*Cpg*Rhoeg
      Ca3  = (PI*Dto**2/4.0)*dla1*Cpg*Rhoeg
      Ca4  = (PI*Dto**2/4.0)*dla1*Cpg*Rhoeg
```

Ca5 = (PI*Dto**2/4.0)*dla1*Cpg*Rhoeg
Ca6 = (PI*Dto**2/4.0)*dla1*Cpg*Rhoeg
Ca7 = 1.0
Ca8 = 1.0
Ca9 = Rhoeg*Cpg*PI*(Rx2**2-Rx1**2)*Dla1
Ca10 = Rhoeg*Cpg*PI*(Rx3**2-Rx2**2)*Dla1
Ca11 = Rhoeg*Cpg*PI*(Rx4**2-Rx3**2)*Dla1
Ca12 = Rhoeg*Cpg*PI*(Rx5**2-Rx4**2)*Dla1
Ca13 = Rhoeg*Cpg*PI*(Rx6**2-Rx5**2)*Dla1
Ca14 = 1.0

C   TANK NODAL CAPACITANCE OF GLYCOL

Ct1 = (PI*Dti**2/4.0)*dltk*Cpgl*Rhoegl
Ct2 = (PI*Dti**2/4.0)*dltk*Cpgl*Rhoegl
Ct3 = (PI*Dti**2/4.0)*dltk*Cpgl*Rhoegl
Ct4 = (PI*Dti**2/4.0)*dltk*Cpgl*Rhoegl
Ct5 = (PI*Dti**2/4.0)*dltk*Cpgl*Rhoegl
Ct6 = (PI*Dti**2/4.0)*dltk*Cpgl*Rhoegl
Ct7 = (PI*Dti**2/4.0)*dltk*Cpgl*Rhoegl
Ct8 = (PI*Dti**2/4.0)*dltk*Cpgl*Rhoegl
Ct9 = (PI*Dti**2/4.0)*dltk*Cpgl*Rhoegl
Ct10 = (PI*Dti**2/4.0)*dltk*Cpgl*Rhoegl
Ct11 = (PI*Dti**2/4.0)*dltk*Cpgl*Rhoegl

C   TANK SURROUNDING GROUND NODAL CAPACITANCE

Ct15 = Rhoeg*Cpg*PI*(Rx2**2-Rx1**2)*Dltk
Ct16 = Rhoeg*Cpg*PI*(Rx3**2-Rx2**2)*Dltk
Ct17 = Rhoeg*Cpg*PI*(Rx4**2-Rx3**2)*Dltk
Ct18 = Rhoeg*Cpg*PI*(Rx5**2-Rx4**2)*Dltk

```
Ct19  = Rhoeg*Cpg*PI*(Rx6**2-Rx5**2)*Dltk
Ct20  = 1.0


C     GROUND NODAL CAPACITANCE BELOW TANK


Cb1   = (PI*Dto**2/4.0)*dlb1*Cpg*Rhoeg
Cb2   = (PI*Dto**2/4.0)*dlb1*Cpg*Rhoeg
Cb3   = (PI*Dto**2/4.0)*dlb1*Cpg*Rhoeg
Cb4   = (PI*Dto**2/4.0)*dlb1*Cpg*Rhoeg
Cb5   = (PI*Dto**2/4.0)*dlb1*Cpg*Rhoeg
Cb6   = 1.0
Cb7   = PI*(Rx2**2-Rx1**2)*Dltk*Cpg*Rhoeg
Cb8   = PI*(Rx3**2-Rx2**2)*Dltk*Cpg*Rhoeg
Cb9   = PI*(Rx4**2-Rx3**2)*Dltk*Cpg*Rhoeg
Cb10  = PI*(Rx5**2-Rx4**2)*Dltk*Cpg*Rhoeg
Cb11  = PI*(Rx6**2-Rx5**2)*Dltk*Cpg*Rhoeg
Cb12  =1.0


C     CONDUCTANCE CALCULATIONS START HERE


C     RADIAL GROUND  CONDUCTANCES ABOVE THE TANK


Far1   = 2.0*Kg*PI*Dla1/ln(Rto/Rtc)
Farr1  = 2.0*Kg*PI*dla1/ln(R2/R1)
Farr2  = 2.0*Kg*PI*dla1/ln(R3/R2)
Farr3  = 2.0*Kg*PI*dla1/ln(R4/R3)
Farr4  = 2.0*Kg*PI*dla1/ln(R5/R4)
Farr5  = 2.0*Kg*PI*dla1/ln(R6/R5)
Farr6  = 2.0*Kg*PI*dla1/ln(R7/R6)


C     TANK AND GROUND RADIAL CONDUCTANCES
```

```
Ftr1    = 2.0*Kg*PI*Dltk/ln(R2/R1)        $ TANK SURROUNDING GROUND
Ftr2    = 2.0*Kg*PI*Dltk/ln(R3/R2)
Ftr3    = 2.0*Kg*PI*Dltk/ln(R4/R3)
Ftr4    = 2.0*Kg*PI*Dltk/ln(R5/R4)
Ftr5    = 2.0*Kg*PI*Dltk/ln(R6/R5)
Ftr6    = 2.0*Kg*PI*Dltk/ln(R7/R6)


C    GROUND RADIAL CONDUCTANCES BELOW TANK


Fbr1    = 2.0*Kg*PI*Dlb1/ln(R2/R1)
Fbr2    = 2.0*Kg*PI*Dlb1/ln(R3/R2)
Fbr3    = 2.0*Kg*PI*Dlb1/ln(R4/R3)
Fbr4    = 2.0*Kg*PI*Dlb1/ln(R5/R4)
Fbr5    = 2.0*Kg*PI*Dlb1/ln(R6/R5)
Fbr6    = 2.0*Kg*PI*Dlb1/ln(R7/R6)



C    GROUND VERTICAL CONDUCTANCE ABOVE THE TANK


Fav  = (PI*Kg*Dto**2/4.0)/(dla1/2.0)
Fav1  = PI*Kg*(Rx2**2-Rx1**2)/(dla1/2.0)
Fav2  = PI*Kg*(Rx3**2-Rx2**2)/(dla1/2.0)
Fav3  = PI*Kg*(Rx4**2-Rx3**2)/(dla1/2.0)
Fav4  = PI*Kg*(Rx5**2-Rx4**2)/(dla1/2.0)
Fav5  = PI*Kg*(Rx6**2-Rx5**2)/(dla1/2.0)


C    GLYCOL VERTICAL CONDUCTANCE IN THE TANK


Ftv  = (PI*Kfgl*Dto**2/4.0)/(dltk/2.0)
```

C   VERTICAL CONDUCTANCE OF GROUND SURROUNDING THE TANK

```
Ftv1  =  PI*Kg*(Rx2**2-Rx1**2)/(dltk/2.0)
Ftv2  =  PI*Kg*(Rx3**2-Rx2**2)/(dltk/2.0)
Ftv3  =  PI*Kg*(Rx4**2-Rx3**2)/(dltk/2.0)
Ftv4  =  PI*Kg*(Rx5**2-Rx4**2)/(dltk/2.0)
Ftv5  =  PI*Kg*(Rx6**2-Rx5**2)/(dltk/2.0)
```

C   GROUND VERTICAL CONDUCTANCE BELOW THE TANK

```
Fbv   = (PI*Kg*Dto**2/4.0)/(dlb1/2.0)
```

```
Fbv1  =  PI*Kg*(Rx2**2-Rx1**2)/(dlb1/2.0)
Fbv2  =  PI*Kg*(Rx3**2-Rx2**2)/(dlb1/2.0)
Fbv3  =  PI*Kg*(Rx4**2-Rx3**2)/(dlb1/2.0)
Fbv4  =  PI*Kg*(Rx5**2-Rx4**2)/(dlb1/2.0)
Fbv5  =  PI*Kg*(Rx6**2-Rx5**2)/(dlb1/2.0)
P11=0.0
P12=0.0
P13=0.0
P14=0.0
P15=0.0
HEADER NODE DATA, SUB1
```

C   BOUNDARY NODE POINTS OF GROUND ABOVE TANK TOP

```
-181,Tda1,1.0
GEN -185,5,1,Tda1,1.0
-200,Tda2,1.0
-210,Tda3,1.0
-220,Tda4,1.0
```

56

```
-230,Tda5,1.0
-240,Tda6,1.0


C    ARITHMETIC NODE POINTS ABOVE TANK TOP

192,Tda2,-1.0
202,Tda3,-1.0
212,Tda4,-1.0
222,Tda5,-1.0
232,Tda6,-1.0


C    DIFUSSION NODE POINTS ABOVE TANK GROUND

191,Tda2,Ca2
201,Tda3,Ca3
211,Tda4,Ca4
221,Tda5,Ca5
231,Tda6,Ca6
GEN 195,5,1,Tda2,Ca9
GEN 205,5,1,Tda3,Ca10
GEN 215,5,1,Tda4,Ca11
GEN 225,5,1,Tda5,Ca12
GEN 235,5,1,Tda6,Ca13


C    END OF GROUND NODE POINTS ABOVE TANK TOP

C    BOUNDARY NODE POINTS FOR TANK SUROUNDING GROUND

-20,Td2,1.0
-30,Td3,1.0
-40,Td4,1.0
```

-50,Td5,1.0
-60,Td6,1.0
-70,Td7,1.0
-80,Td8,1.0
-90,Td9,1.0
-100,Td10,1.0
-110,Td11,1.0
-120,Tdb1,1.0
-130,Tdb2,1.0
-140,Tdb3,1.0
-150,Tdb4,1.0
-160,Tdb5,1.0
-161,Tdb6,1.0
GEN -175,5,1,Tdb6,1.0

C   TANK ARITHMETIC NODE POINTS

GEN 5,5,1,Td1,-1.0
GEN 1,2,2,Td1,-1.0
GEN 301,2,2,Td1,-1.0
GEN 401,2,2,Td1,-1.0
GEN 12,2,2,Td2,-1.0
GEN 22,2,2,Td3,-1.0
GEN 32,2,2,Td4,-1.0
GEN 42,2,2,Td5,-1.0
GEN 52,2,2,Td6,-1.0
GEN 62,2,2,Td7,-1.0
GEN 72,2,2,Td8,-1.0
GEN 82,2,2,Td9,-1.0
GEN 92,2,2,Td10,-1.0
GEN 102,2,2,Td11,-1.0

C   TANK WALL DIFUSSION NODE POINTS
      2, Td2, Ctsc
      302,Td6,Ctsc
      402,Td11,Ctsc
      13,Td2,Ctwc
      23,Td3,Ctwc
      33,Td4,Ctwc
      43,Td5,Ctwc
      53,Td6,Ctwc
      63,Td7,Ctwc
      73,Td8,Ctwc
      83,Td9,Ctwc
      93,Td10,Ctwc
      103,Td11,Ctwc

C   DIFUSSION NODES OF GROUND SUROUNDING TANK(BELOW TANK TOP)

      GEN 15,5,1,Td2,Ct15
      GEN 25,5,1,Td3,Ct16
      GEN 35,5,1,Td4,Ct17
      GEN 45,5,1,Td5,Ct18
      GEN 55,5,1,Td6,Ct19
      GEN 65,5,1,Td7,Ct15
      GEN 75,5,1,Td8,Ct16
      GEN 85,5,1,Td9,Ct17
      GEN 95,5,1,Td10,Ct18
      GEN 105,5,1,Td11,Ct19
      GEN 115,5,1,Td12,-1.0

C   GROUND BELOW TANK

```
GEN 125,5,1,Tdb1,Cb1
GEN 135,5,1,Tdb2,Cb2
GEN 145,5,1,Tdb3,Cb3
GEN 155,5,1,Tdb4,Cb4
GEN 165,5,1,Tdb4,Cb4
```

C   GLYCOL INSIDE TANK

```
11,Td2,CT2
21,Td3,CT3
31,Td4,CT4
41,Td5,CT5
51,Td6,CT6
61,Td7,CT7
71,Td8,CT8
81,Td9,CT9
91,Td10,CT10
101,Td11,CT11
111,Tdb1,CT1
121,Tdb2,CT2
131,Tdb3,CT3
141,Tdb4,CT4
151,Tdb5,CT5
```

C   ARITHMETIC NODES BELOW TANK

```
112,Tdb1,-1.0
122,Tdb2,-1.0
132,Tdb3,-1.0
142,Tdb4,-1.0
```

152,Tdb5,-1.0

C   GLYCOL RETURN TEMPERATURE
-300,Tgi,CT1 $ THIS NODE WILL BE CONNECTED TO NODE #51 OR NODE #101

C   GLYCOL SUPPLY TEMPERATURE

600,Td2,CT2 $ THIS NODE WILL BE CONNECTED TO #61 OR NODE #11

HEADER CONDUCTOR DATA, SUB1

C   "RADIAL CONDUCTION CONNECTIONS"

C   RADIAL CONDUCTANCES IN GROUND ABOVE TANK

GEN 1,5,1,191,10,192,10,Farr1
GEN 6,5,1,192,10,195,10,Farr1
GEN 11,5,1,195,10,196,10,Farr2
GEN 16,5,1,196,10,197,10,Farr3
GEN 21,5,1,197,10,198,10,Farr4
GEN 26,5,1,198,10,199,10,Farr5
GEN 31,5,1,199,10,200,10,Farr6

C   RADIAL CONDUCTANCE IN TANK

GEN 36,10,1,11,10,12,10,Ftrc
GEN 46,10,1,12,10,13,10,Ftmw
GEN 56,10,1,13,10,14,10,Ftmw
GEN 66,10,1,14,10,15,10,Ftr1
GEN 76,10,1,15,10,16,10,Ftr2

```
GEN 86,10,1,16,10,17,10,Ftr3
GEN 96,10,1,17,10,18,10,Ftr4
GEN 106,10,1,18,10,19,10,Ftr5
GEN 116,10,1,19,10,20,10,Ftr6


C    BELOW TANK


GEN 126,5,1,111,10,112,10,Ftrc
GEN 131,5,1,112,10,125,10,Fbr1
GEN 136,5,1,125,10,126,10,Fbr2
GEN 141,5,1,126,10,127,10,Fbr3
GEN 146,5,1,127,10,128,10,Fbr4
GEN 151,5,1,128,10,129,10,Fbr5
GEN 156,5,1,129,10,120,10,Fbr6


C    VERTICAL CONNECTION IN GROUND ABOVE TANK


161,181,191,Fav
GEN 162,4,1,191,10,201,10,Fav*2.0
166,231,1,Fav
167,185,195,Fav1
168,186,196,Fav2
169,187,197,Fav3
170,188,198,Fav4
171,189,199,Fav5
GEN 172,4,1,195,10,205,10,Fav1*2.0
GEN 176,4,1,196,10,206,10,Fav2*2.0
GEN 180,4,1,197,10,207,10,Fav3*2.0
GEN 184,4,1,198,10,208,10,Fav4*2.0
GEN 188,4,1,199,10,209,10,Fav5*2.0
192,231,1,Fav
```

193,235,5,Fav1
194,236,6,Fav2
195,237,7,Fav3
196,238,8,Fav4
197,239,9,Fav5

C   VERTICAL CONECTIONS IN TANK AND SURROUNDING GROUND

198, 1,2,Ftmb/2.0
199, 2,3,Ftmb/2.0
200, 3,11,Ftv

GEN 201,4,1,11,10,21,10,Ftv*2.0
206,51,301,Ftv
207,301,302,Ftmb/2.0
208,302,303,Ftmb/2.0
209,303,61,Ftv
GEN,210,4,1,61,10,71,10,Ftv*2.0
214,101,401,Ftv
215,401,402,Ftmb/2.0
216,402,403,Ftmb/2.0

C   GROUND BELOW TANK

217,403,111,Fbv
GEN 218,4,1,111,10,121,10,Fbv*2.0
222,151,161,Fbv

C   VERTICAL CONNECTION IN TANK WALL

GEN 223,9,1,13,10,23,10,Ftmw

232,5,15,Ftv1
233,6,16,Ftv2
234,7,17,Ftv3
235,8,18,Ftv4
236,9,19,Ftv5
GEN 237,9,1,15,10,25,10,Ftv1*2.0
GEN 246,9,1,16,10,26,10,Ftv2*2.0
GEN 255,9,1,17,10,27,10,Ftv3*2.0
GEN 264,9,1,18,10,28,10,Ftv4*2.0
GEN 273,9,1,19,10,29,10,Ftv5*2.0

282,105,115,Ftv1
283,106,116,Ftv2
284,107,117,Ftv3
285,108,118,Ftv4
286,109,119,Ftv5

C   GROUND BELOW TANK

287,115,125,Fav1
288,116,126,Fav2
289,117,127,Fav3
290,118,128,Fav4
291,119,129,Fav5

GEN 316,4,1,125,10,135,10,Fav1*2
GEN 320,4,1,126,10,136,10,Fav2*2
GEN 324,4,1,127,10,137,10,Fav3*2
GEN 328,4,1,128,10,138,10,Fav4*2
GEN 332,4,1,129,10,139,10,Fav5*2

```
      336,165,175,Fav1
      337,166,176,Fav2
      338,167,177,Fav3
      339,168,178,Fav4
      340,169,179,Fav5


C     TANK TUBE CONNECTIONS

      350,300,101,G350   $ LOWER TANK CONNECTIONS
      351,61,600,G351    $ LOWER TANK CONNECTIONS
      352,300,51,G352    $ UPPER TANK CONNECTIONS
      353,11,600,G353    $ UPPER TANK CONNECTIONS
HEADER CONTROL DATA, GLOBAL
      TIMEND = 40.0
      OUTPUT = 1.0
      DTIMEH = 0.1
HEADER VARIABLES1,SUB1
M     IF(TIMEN.LE.TSW) GO TO 1000
M     IF(TIMEN.GT.TSW.AND.TIMEN.LE.2.0*TSW) GO TO 2000
M     IF(TIMEN.GT.2.0*TSW.AND.TIMEN.LE.3.0*TSW) GO TO 1000
M     IF(TIMEN.GT.3.0*TSW.AND.TIMEN.LE.4.0*TSW) GO TO 2000
 1000 CONTINUE
M     G210 = FCOntk        $ LOWER TANK ACTIVATED CONNECTIONS
M     G211 = FCOntk
M     G212 = FCOntk
M     G213 = FCOntk
M     G41  = Ftcon
M     G42  = Ftcon
M     G43  = Ftcon
M     G44  = Ftcon
M     G45  = Ftcon
```

```
M     G350 = FCOntk
M     G351 = FCOntk
C                     $ UPPER TANK DE-ACTIVATE CONNECTIONS
M     G201 = FTV*2.0
M     G202 = FTV*2.0
M     G203 = FTV*2.0
M     G204 = FTV*2.0
M     G36  = Ftrc
M     G37  = Ftrc
M     G38  = Ftrc
M     G39  = Ftrc
M     G40  = Ftrc
M     G352 = 0.0
M     G353 = 0.0


M     GO TO 1001
C
 2000  CONTINUE

M     G201 = FCOntk          $ UPPER TANK  ACTIVATED CONNECTIONS
M     G202 = FCOntk
M     G203 = FCOntk
M     G204 = FCOntk
M     G36  = Ftcon
M     G37  = Ftcon
M     G38  = Ftcon
M     G39  = Ftcon
M     G40  = Ftcon
M     G352 = FCOntk
M     G353 = FCOntk
C                     $ LOWER TANK DE-ACTIVATED CONNECTIONS
```

```
M      G210 = FTV*2.0
M      G211 = FTV*2.0
M      G212 = FTV*2.0
M      G213 = FTV*2.0
M      G41  = Ftrc
M      G42  = Ftrc
M      G43  = Ftrc
M      G44  = Ftrc
M      G45  = Ftrc
M      G350 = 0.0
M      G351 = 0.0
 1001  CONTINUE
HEADER OPERATION DATA
BUILD GTANK2,SUB1
      CALL TRANSIENT
HEADER OUTPUT CALLS,SUB1
       P11=T300
       P12=T600
       P13 = FCONtK*(T600-T300)*60.0
       P14= GLFR
       P15=TIMEN
       WRITE(6,15) P11,P12,P13,P14,P15
  15    FORMAT(1X,'Tin=',1F10.3,1X,'To=',1F10.3,1X,'Qo=',1F15.3,1X,'gpm=',1F10.3,'TM='1F10.1)
       CALL TPRINT ('SUB1')
C      CALL QPRINT ('SUB1')
END OF DATA
```

APPENDIX B: LAMSON VYLON PIPE CATALOG

LAMSON VYLON PIPE
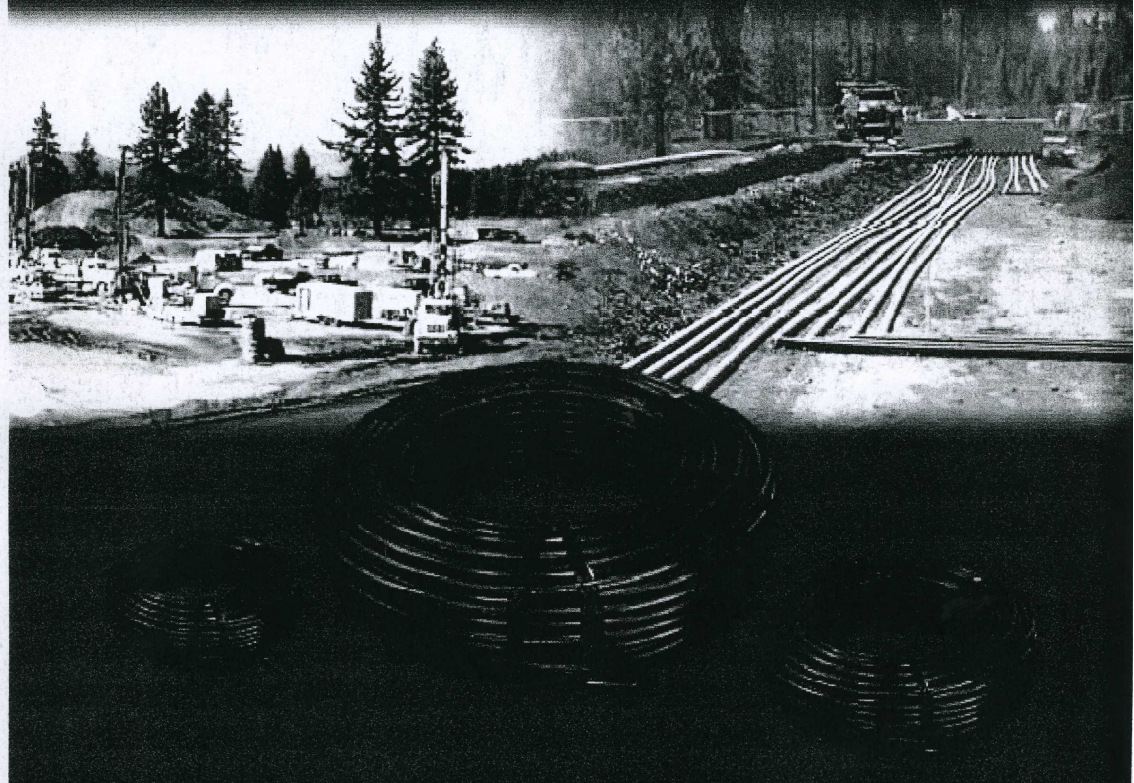
Geothermal
High Density Polyethylene Pipe

LAMSON VYLON® PIPE

Geothermal Pipe – 50 Year Limited Warranty

Lamson & Sessions will warrant its Geothermal pipe, Geothermal "U"-Bend, and Geothermal pipe loop assemblies for a period of 50 years from the date of manufacture. Lamson & Sessions will replace free of charge and with freight prepaid the new pipe quantity equivalent to the quantity of pipe that failed.

WARRANTY WILL APPLY IF:	WARRANTY WILL **NOT** APPLY IF:
• Geothermal pipe and Geothermal loop assemblies are properly installed, utilized, tested and inspected per the International Ground Source Heat Pump Association (IGSHPA) standards	• The design, installation, inspection or testing of the Geothermal pipe and Geothermal loop assemblies was not in accordance with the recommendations of the IGSHPA, applicable code requirements, if any, and industry accepted guidelines
• Geothermal pipe and Geothermal loop assemblies are tested before sealing/grouting the final installation as per IGSHPA standards	• Installed product is affected by a natural disaster or by earth movement, whether caused by persons or by natural cause

• Your supplier is notified immediately	• Product affected by
	  • Improper handling/storage
	  • Inadequate freeze protection
	  • Improper temperatures
	  • Improper pressure levels

• Lamson & Sessions receives written notice explaining the defect within 30 days of discovery of defect	• Product is exposed to petroleum products
	THIS WARRANTY DOES NOT COVER:
	• Components in the closed-loop Geothermal system other than the pipe or loop assemblies • Pipe and fitting joining process, method, tools, or equipment • Uses for other than Geothermal installations

This warranty is in lieu of all other warranties, express, implied or statutory, including but not limited to any warranty for merchantability of fitness for a particular purpose and shall constitute the sole remedy to any original purchaser of Lamson & Sessions' Geothermal pipe.

LAMSON VYLON PIPE

25701 Science Park Drive • Cleveland, Ohio 44122
Phone: 800-382-0862 • Fax: 216-766-6577 • www.vylonpipe.com

GE01 Rev 4/04 SP

Geothermal High Density Polyethylene Pipe

Geothermal Pipe Loops Coiled with "U"-Bend* Attached

Pipe Size	SDR 11 O.D.	Min. Wall	Approx Wt./100 ft	Coil Length (Feet)	Coil Dimensions (Inches) I.D.	O.D.	Width
3/4"	1.050	0.095	12 lbs.	310	30.8	44.0	6.00
				410	32.0	44.0	7.50
				510	30.0	48.5	6.00
1"	1.315	0.120	19 lbs.	310	30.0	44.0	8.00
				410	30.0	48.0	9.25
				510	30.0	48.5	9.50
				610	30.0	48.5	11.00
1 1/4"	1.660	0.151	31 lbs.	710	48.0	76.0	9.00
				810	48.0	71.0	12.00

Made to Order Lengths
SDR 9 Available
*"U"-Bend Rated at SDR 9
*Patent Pending

Pipe Loop Assembly
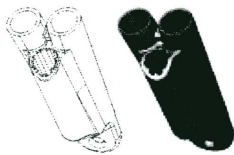
"U"-Bend

Head Loss per 100 ft of SDR 11 PE Pipe

GPM	3/4 inch	1 inch	1-1/4 inch	1-1/2 inch	2 inch
1	0.23	0.08	0.02	0.01	
2	0.83	0.27	0.09	0.04	0.01
3	1.76	0.58	0.18	0.09	0.03
4	2.99	0.99	0.31	0.16	0.05
5	4.52	1.49	0.47	0.24	0.08
6	6.34	2.09	0.66	0.34	0.11
7	8.44	2.78	0.88	0.45	0.15
8	10.80	3.56	1.12	0.58	0.19
9	13.44	4.43	1.39	0.72	0.24
10	16.33	5.39	1.70	0.87	0.29
11	19.48	6.43	2.02	1.04	0.35
12	22.89	7.55	2.38	1.23	0.41
13	26.55	8.76	2.76	1.42	0.48
14	30.45	10.05	3.16	1.63	0.55
15	34.60	11.42	3.59	1.85	0.62
20		19.45	6.12	3.16	1.06
25		29.40	9.25	4.77	1.61
30		41.22	12.97	6.69	2.25
40			22.10	11.40	3.84
50			33.40	17.23	5.81
75				36.51	12.31
100					20.96

*U.S. gallons per minute
"Head loss based on Hazen-Williams Formula, C=150 for water"
ASTM D3035 nominal outside diameter and average wall thickness

Technical Information

Property	ASTM Test Method	Typical Values English Units	SI Units
Density (Natural)	D 4883	——	0.944 g/cc
Density (Black)		—	0.955 g/cc
Melt Index [1]	D 1238	—	12.5 g/10 min
Tensile Strength			
@ Yield (2 in/min)	D 638	3300 psi	22.8 MPa
@ Break (2 in/min)	D 638	4500 psi	31.0 MPa
Elongation			
@ Break (2 in/min)	D 638	>800%	>800%
Flexural Modulus [2]	D 790	120,000 psi	827 MPa
Notched Izod Impact Strength	D 256	6 ft-lbf/in	0.32 kJ/m
Hardness (Shore D)	D 2240	68	68
Vicat Softening Point	D 1525	259° F	126° C
Brittleness Temperature	D 746	<-180° F	<-118° C
Hydrostatic Design Basis			
@ 23° C	D 2837	1600 psi	11.0 MPa
@ 60° C	D 2837	800 psi	5.5 MPa
Minimum Required Strength	ISO 9080	—	8.0 MPa
Environmental Stress Crack Resistance [3]	D 1693	>2000 hrs	>2000 hrs
Environmental Stress Crack Resistance [4]	D 1693	>5000 hrs	>5000 hrs
Pipe Ring ESCR [5]	F 1248	>5000 hrs	>5000 hrs
Notch Tensile (PENT)	F 1473	>100 hrs	>100 hrs
Carbon Black Concentration	D 1603	2.3%	2.3%
Cell Classification	D 3350	345464C	345464C

[1] 190° C/21600 g   [2] 2% Secant-Method 1   [3] Condition B, 10%   [4] Condition C   [5] Two inch, SIDR 19

Geothermal High Density Polyethylene Pipe

Specifications and Data

SCOPE
- Meets overall design requirements for use in closed-loop ground source heat exchangers
- Pipe configured as specified in ASTM D 3035
- Standard specification for Polyethylene (PE) Plastic Pipe (DR-PR) based on controlled outside diameter or ASTM D 2447

PIPE
- Conforms to requirements of ASTM D 3035 or ASTM D 2447
- Loop pipe shall be joined at one end with a factory-installed "U"-Bend with a standard dimension ratio (SDR) of 9
- "U"-Bends will be a single piece, injection molded from PE 3408 HDPE resin with a minimum cell classification of PE345434C
- Working pressure rating of 160 psi water at 73°F for SDR 11 and 200 psi water at 73°F for SDR 9

MATERIAL
- All pipe and heat fused materials are manufactured from virgin high density polyethylene resin exceeding or meeting ASTM D 2513, Sections 4.1 & 4.2
- Hydrostatic Design Basis – 1600 psi at 73°F per ASTM D 2837 and ASTM D 3350
- Listed in PPI-TR4 as PE 3408

MARKING
- Pipe marked with identification, nominal size, trade name, material designation or cell classification, pressure rating at 73°F, ASTM standard, date of manufacture
- Coiled loop pipe shall also have sequential footage marking at two-foot intervals

INSTALLATION
- Installation & backfill shall comply with IGSHPA guidelines, local, state & federal regulations

- When performing heat fusion, IGSHPA's recommended fusion procedures or fusion tool manufacturer's joining procedure for PE 3408 should be utilized

TESTING
- DO NOT TEST WITH AIR OR GAS
- The loop assembly shall be hydrostatically tested with water at a pressure not greater than 150% of the pipe pressure rating

GEOTHERMAL PIPE
- Available in coils 3/4"-4"
- Straight lengths (available up to 16 inch) using PE 3408 resin
- For commercial, institutional, residential or industrial applications

GEOTHERMAL LOOPS
- Custom designed to your job requirements
- Coils are configured with a parallel supply and return pipe heat fused to a patent pending "U"-Bend to make a leak tight joint utilizing PE 3408 resin
- 3/4", 1", & 1-1/4" SDR 11 rated @ 160 psi
- 3/4", 1", & 1-1/4" SDR 9 rated @ 200 psi
- Outside width of "U"-Bend less than 2.50" for 3/4", 3.00" for 1", 3.50" for 1-1/4"

FEATURES AND BENEFITS OF LAMSON VYLON PIPE GEOTHERMAL PRODUCTS
- Fusionable
- Flexible
- Abrasion resistant
- Footage markers on loop piping
- Permanent indent printing
- Made from PE 3408 high density polyethylene resin
- Resistant to most chemical compounds and aggressive soils

High Density Polyethylene IPS Pipe and Pressure Ratings

Iron Pipe Size (IPS)		265 psi SDR 7		200 psi SDR 9		160 psi SDR 11		130 psi SDR 13.5		110 psi SDR 15.5		100 psi SDR 17		90 psi SDR 19		80 psi SDR 21		65 psi SDR 26		50 psi SDR 32.5	
Nom. O.D.	Act. O.D.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.
1"	1.315	0.188	0.291	0.146	0.234	0.119	0.196	0.097	0.163												
1 1/4"	1.66	0.237	0.463	0.184	0.374	0.150	0.313	0.122	0.259	0.107	0.228										
1 1/2"	1.9	0.271	0.607	0.211	0.489	0.172	0.41	0.140	0.34	0.123	0.299	0.112	0.274								
2"	2.375	0.339	0.948	0.264	0.765	0.216	0.640	0.176	0.531	0.153	0.467	0.140	0.429								
3"	3.500	0.500	2.059	0.389	1.661	0.318	1.390	0.259	1.153	0.226	1.015	0.206	0.931	0.184	0.838	0.167	0.763				
4"	4.500	0.643	3.404	0.500	2.746	0.409	2.297	0.333	1.907	0.290	1.678	0.265	1.539	0.237	1.387	0.214	1.261	0.173	1.028	0.138	0.829
5"	5.563	0.795	5.202	0.618	4.196	0.506	3.511	0.412	2.914	0.359	2.564	0.327	2.352	0.293	2.120	0.265	1.927	0.214	1.571	0.171	1.267
6"	6.625	0.946	7.378	0.736	5.951	0.602	4.980	0.491	4.133	0.427	3.636	0.390	3.336	0.349	3.007	0.315	2.733	0.255	2.228	0.204	1.797
8"	8.625	1.232	12.505	0.958	10.086	0.784	8.440	0.639	7.004	0.556	6.164	0.507	5.654	0.454	5.093	0.411	4.631	0.332	3.777	0.265	3.046
10"	10.750	1.536	19.426	1.194	15.668	0.977	13.111	0.796	10.881	0.694	9.575	0.632	8.783	0.566	7.913	0.512	7.195	0.413	5.867	0.331	4.731
12"	12.750	1.821	27.326	1.417	22.041	1.159	18.443	0.944	15.306	0.823	13.469	0.750	12.355	0.671	11.127	0.607	10.121	0.490	8.253	0.392	6.655
14"	14.000	2.000	32.947	1.556	26.575	1.273	22.237	1.037	18.455	0.903	16.239	0.824	14.896	0.737	13.419	0.667	12.203	0.538	9.951	0.431	8.024
16"	16.000	2.286	43.033	1.778	34.710	1.455	29.044	1.187	24.104	1.032	21.210	0.941	19.457	0.842	17.521	0.762	15.938	0.615	12.997	0.492	10.481

High Density Polyethylene DIPS Pipe and Pressure Ratings

Ductile Iron Pipe Size (DIPS)		265 psi SDR 7		200 psi SDR 9		160 psi SDR 11		130 psi SDR 13.5		110 psi SDR 15.5		100 psi SDR 17		90 psi SDR 19		80 psi SDR 21		65 psi SDR 26		50 psi SDR 32.5	
Nom. O.D.	Act. O.D.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.	Min. Wall	Wt. Lb./Ft.
4"	4.800	0.686	3.835	0.533	30.930	0.436	2.588	0.356	2.148	0.310	1.890	0.282	1.734	0.253	1.561	0.229	1.420	0.185	1.158	0.148	0.934
6"	6.900	0.986	7.924	0.767	6.391	0.627	5.348	0.511	4.438	0.445	3.909	0.406	3.583	0.363	3.227	0.329	2.935	0.265	2.393	0.212	1.930
8"	9.050	1.293	13.630	1.006	10.995	0.823	9.200	0.670	7.635	0.584	6.719	0.532	6.163	0.476	5.551	0.431	5.049	0.348	4.117	0.278	3.320
10"	11.100	1.586	20.510	1.233	16.540	1.009	13.840	0.822	11.490	0.716	10.107	0.653	9.272	0.584	8.350	0.529	7.595	0.427	6.193	0.342	4.994
12"	13.200	1.886	29.000	1.467	23.390	1.200	19.570	0.978	16.240	0.852	14.290	0.776	13.110	0.695	11.809	0.629	10.740	0.508	8.758	0.406	7.063
14"	15.300	2.186	38.960	1.700	31.420	1.391	26.300	1.133	21.820	0.987	19.200	0.900	17.620	0.805	15.865	0.729	14.430	0.588	11.767	0.471	9.489

*Iron Pipe Size - Pressures are based on using water at 23°C (73°F) and are determined by using standard formulas for the industry.

NOTE: Service factors should be utilized to compensate for the effect of substances other than water, and for other temperatures.

Density	0.955
Corr. Factor	1.045

APPENDIX C: SINDA/FLUINT THERMAL ANALYZER FOREWARD

FOREWORD

SINDA/FLUINT is a significantly upgraded version of a long sequence of predecessor codes dating back into the 1960s. Major improvements over older SINDA codes include submodels, registers and expressions (an underlying spreadsheet), design optimization and test data correlation, statistical design methods, and a comprehensive fluid flow analysis capability: FLUINT.*

SINDA/FLUINT has been well received since its first introduction in 1985 (under the name SINDA '85) having supplanted other SINDA versions and SINDA-like codes, and becoming a best-seller and a NASA Space Act Award winner in the process. Originally intended to satisfy the ever-changing modeling needs of the spacecraft and launch vehicle thermal community, the program has diffused into other industries such as nuclear, aircraft, electronic packaging, HVAC, and automotive, and into other specialties and subsystems such as propulsion and environmental control.

SINDA is intended primarily for designing and analyzing thermal systems represented in electrical analog, lumped parameter form. Given properly prepared inputs, SINDA can solve both finite difference and finite element equations: it is better classified as an "equation solver" than its traditional classification as a "finite difference solver." Potential SINDA users who are unfamiliar with thermal system modeling techniques should refer to page xxii in this preface for training materials.

The FLUINT capability was originally added to SINDA in 1986. It is now arguably the most advanced one dimensional thermal/hydraulic code available, solving arbitrary fluid flow networks without presuming the application. The networks may contain single-phase vapor or liquid as the working fluid, or combinations of liquid and/or vapor at various locations in the network, and even mixtures of working fluids. These networks are solved in conjunction with one or more accompanying thermal networks, accounting for the energy flows in and out of the fluid system(s).

After all the major expansions in the last ten years, the improvements in this version are comparatively modest, although numerous. Improvements include the addition of the automatic choked flow detection and modeling, conductor heat rates (HRs), expanded model sizes, and more simulation support for orifices, heatpipes, and Peltier devices (TECs). See VERSION 4.5 vs. VERSION 4.4 on page xv for more detailed information.

The improvements that yield the most analytic power, however, are hidden ... or at least external to SINDA/FLUINT.

Until the 1990's, SINDA/FLUINT was primarily used as a batch style code run with manually-generated text input files, although archaic codes such as TRASYS were often used to prepare certain inputs such as radiation factors. Now, thanks to a new generation of graphical interfaces, SINDA/FLUINT is primarily used as a "solution engine." These interfaces include the nongeometric sketch-pad-style Sinaps*Plus*®, and more recently the geometry-based Thermal Desktop® (for SINDA conduction/capacitance calculations) with its companion modules RadCAD® (SINDA radiation calculations) and FloCAD® (FLUINT circuits and convective heat transfer calculations). The latest

---

* *SINDA/FLUINT* is used to refer to the whole program, especially when used to distinguish it from older thermal-only versions. *SINDA* is used to refer to the thermal capabilities, and *FLUINT* to the fluid flow capabilities.

release of FloCAD, in particular, contains major expansions for facilitated modeling of heat pipes and single- or two-phase fluid loops coupled to the structural (nonfluidic) thermal model. It has significantly improved the ease of use for FLUINT modeling.

The entire Thermal Desktop suite is fully parametric. More importantly, SINDA/FLUINT can be run dynamically from within it, calling for new Thermal Desktop/RadCAD/FloCAD solutions *during* SINDA/FLUINT execution. This enables parametric and sensitivity analyses to include complex geometric, property, and environmental variations. It also brings SINDA/FLUINT's Advanced Design modules (Section 5) to be used as drivers for the entire analysis chain. In other words, the traditional separation between Thermal Math Modeling (TMM) software such as SINDA and Geometric Math Modeling (GMM) software such as TRASYS has been completely eliminated for the first time.

A pressure/flow analysis of a system containing an arbitrary piping network can be performed simultaneously with the thermal analysis during transient or steady state solutions. This permits the mutual influences of thermal and fluid problems to be included in the analysis. This fluid network analysis capability is called FLUINT. In the 3.0 release in 1994, the ability to handle fluid mixtures instead of just single substances was added to the code. The Version 4.1 and 4.2 releases added complex mixture phenomena such as dissolution, and full nonequilibrium two-phase control volumes.

In the 4.0 release in 1997, a new spreadsheet-like capability was introduced along with a top-level method for dealing with complex modeling tasks like design optimization and test data correlation. In Version 4.3, this top-level capability was further extended with statistical design methods intended to measure and improve design reliability. These revolutionary capabilities apply to all parts of the code, whether part of thermal or fluid networks. *Therefore, users of older versions of SINDA should take particular heed of those improvements.* In fact, more development has been made in recent years than at any time in the history of SINDA, and this trend is continuing.

Within this manual, SINDA/FLUINT will always be regarded as a thermal and/or fluid network analyzer. The knowledgeable user will, however, recognize the program's ability to solve electrical networks or any other mathematical model that can be represented by a lumped-parameter network.

1.4   Graphical User Interfaces

SINDA/FLUINT was used in a stand-alone fashion for many years, with input files generated in a text editor. While it may still be used in this fashion, almost all new users in the last decade generate models and run SINDA/FLUINT through one of its graphical user interfaces (GUIs): Sinaps*Plus*® (page xviii in the preface), Thermal Desktop® (page xix and page xx in the preface), or a variety of third-party and in-house interfaces. The novice user is strongly encouraged to investigate these more modern methods of using SINDA/FLUINT.

However, in order to support all SINDA/FLUINT users, this manual is written assuming no GUI is used. Nonetheless, references to circumstances special to any such software may be mentioned from time to time.

1.5   Introduction to the R-C Network

To introduce the user to the nature of the R-C network input required by SINDA thermal modeling, as well as to illustrate the basic flexibility of the program, a short sample problem, from engineering statement to completed input file, will be presented in the following paragraphs. Note that fluid networks (introduced in Section 3) are fundamentally different from the thermal networks that will be introduced here. Also note that, despite the use of US Customary units in these examples, *SINDA/FLUINT can employ several unit sets with equal ease.*

The following examples will illustrate the complete process of transforming an engineering problem into a SINDA/FLUINT input file. Only the most elementary features will be employed so that the new user will not be overwhelmed with detail. At the same time, however, a small amount of program logic will be included to expose the new user to the versatility and flexibility that are possible within the SINDA/FLUINT system.

1.5.1   Transiently Heated Bar

Consider a bar of metal as shown in Figure 1-1. It is 0.1 inch thick, 1.0 inch wide, and 3.0 inches long, and is fully insulated except for one end. A 10 watt heater is embedded at the insulated end of the bar, and the uninsulated end radiates to deep space. The bar is initially at a temperature of 70°F. After turning on the heat, it is desired to know the time, to the nearest minute, when the center of the bar reaches a temperature of 200°F, and the temperature distribution in the bar at this time.

Density:	$0.3\ lb_m/in^3$
Specific Heat:	$0.2\ BTU/lb_m\text{-}°F$
Conductivity:	$0.5\ BTU/ft\text{-}min\text{-}°F$
Emissivity:	0.1

Heater

Deep Space
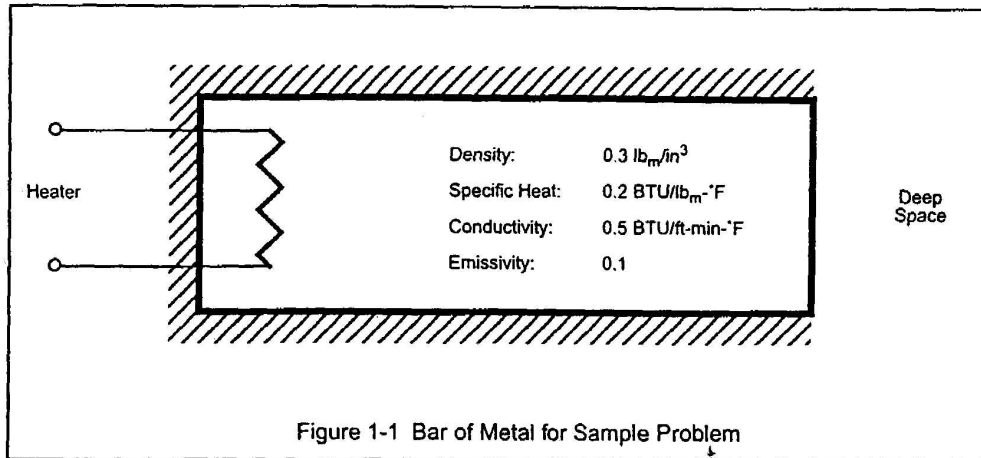
Figure 1-1 Bar of Metal for Sample Problem

Figure 1-2 shows the lumped parameter/R-C network which represents the stated problem. The mass of the bar has been divided into three* portions (shown schematically as capacitors) which are called *nodes*, and each node has been assigned an arbitrary identification number. The heat conduction paths, (shown schematically as resistors) which are called *conductors*, have also been assigned arbitrary identification numbers. A heat *source*, Q, which represents the heater in the insulated end of the bar, is shown entering node 10. Deep space is represented by a constant temperature node at -460°F (shown schematically as a ground).

---

* The purpose of this short example is to introduce concepts, and not to teach good modeling practices. An odd number of portions was chosen such that one would represent the mass at the center of the bar, which is of particular interest to this problem. Three nodes were selected to keep the demonstration simple, whereas more careful engineering judgement might have resulted in a selection of five or seven nodes. Many other nodalizations are possible and some are likely better (e.g., representing the exposed surface temperature with an arithmetic [massless] node, as will be described later).

Introduction and Overview
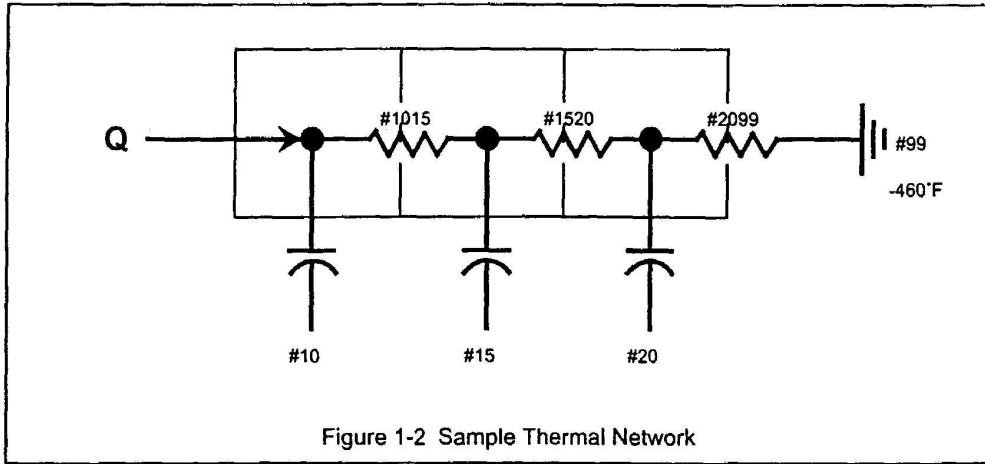
Figure 1-2 Sample Thermal Network

The thermal energy storage capacity (or capacitance) of each bar node is equal to the product of its density, volume, and specific heat (in this case, $0.3*(0.1*1.0*3.0)*0.2/3 = 0.006$ BTU/°F). The three bar nodes may be defined by specifying their identification number, initial temperature, and capacitance as follows:

```
10, 70.0, 0.006          $ record 1
15, 70.0, 0.006          $ record 2
20, 70.0, 0.006          $ record 3
```

The node representing deep space may be defined as follows:

```
-99, -460.0, 0.0         $ record 4
```

The *conductance* of a conductor which represents a heat flow path through a material is equal to the product of the material's thermal conductivity and the cross-sectional area of the flow path, divided by the length of the path. The conductance of bar conductors 1015 and 1520 is, therefore, $(0.5/12.0)*(0.1*1.0)/(3.0/3) = 0.00417$ BTU/min-°F.

The conductance of a conductor which represents heat flow by radiation is equal to the product of the Stefan-Boltzmann constant and the emissivity, area, and view factor from the surface. In this case, conductor 2099 has a conductance of $0.1*(0.1*1.0)*0.1712e-8/(144*60) = 1.98 \times 10^{-15}$ BTU/min-R^4. In SINDA, the three conductors may be defined by specifying their number, the numbers of their adjoining nodes, and their conductances as follows:

```
1015, 10, 15, 0.00417     $ record 5
1520, 15, 20, 0.00417     $ record 6
-2099, 20, 99, 1.98E-15   $ record 7
```

78

Since the bar will undoubtedly reach the temperature of interest in less than 1000 minutes and the output results need be accurate only to the nearest minute, the SINDA end time and output time control variables may be set, appropriately, as follows:

```
TIMEND = 1000.0, OUTPUT = 1.0 $ record 8
```

To impress the heat source on node 10, the following record is prepared, where unit conversions into consistent units are written in pseudo-Fortran:

```
10, 10.0*3.413/60.0              $ record 9
```

Since a transient analysis of the bar is desired, a routine from the SINDA library which performs transient analysis by the Forward-Backward (Crank-Nicholson) finite differencing technique will be called into action, with the following record:

```
CALL TRANSIENT              $ record 10
```

Finally, to suppress output until the center of the bar reaches 200°F, at which time it is desirable to print the temperatures of the nodes and to end the problem, the following logic statements are required:

```
IF (T15 .GE. 200.0) THEN          $ record 11
   CALL TPRINT('SUB1') $ PRINT TEMPS  $ record 12
   TIMEND = TIMEN $ END PROBLEM NOW   $ record 13
ENDIF                             $ record 14
```

Table 1-1 shows how the 14 records prepared specifically for the sample problem have been inserted in a "real" input file which contains the additional records necessary to separate, for example, the node data from the conductor data, etc. This list of records constitutes the complete SINDA/FLUINT input required to solve the stated problem.

1.5.2   Transiently Heated Bar Using Spreadsheet Options

With a little forethought, the user should instead have created a section listing important dimensions and properties to be referenced in other parts of the input. For example:

```
HEADER REGISTER DATA
          DENS      = 0.3
          CP        = 0.2
          CON       = 0.5
          EMIS      = 0.1
          THICK     = 0.1
          WIDE      = 1.0
          LONG      = 3.0
          AREA      = THICK*WIDE
          VOLUME    = AREA*LONG
```

Table 1-1 Complete Input File for Sample Problem

```
HEADER OPTIONS DATA
  TITLE HEATED BAR SAMPLE PROBLEM
        OUTPUT  = BAR.OUT
        MODEL   = TEST
HEADER NODE DATA, SUB1
        10, 70.0, 0.006                  $      record 1
        15, 70.0, 0.006                  $      record 2
        20, 70.0, 0.006                  $      record 3
        -99, -460., 0.0                  $      record 4
HEADER CONDUCTOR DATA, SUB1
        1015, 10, 15, 0.00417            $      record 5
        1520, 15, 20, 0.00417            $      record 6
        -2099, 20, 99, 1.98E-15          $      record 7
HEADER CONTROL DATA, GLOBAL
        TIMEND = 1000.0, OUTPUT = 1.0 $        record 8
HEADER SOURCE DATA, SUB1
        10, 10.0*3.413/60.0              $      record 9
HEADER OPERATIONS
BUILD ALL
        CALL TRANSIENT                   $      record 10
HEADER OUTPUT CALLS, SUB1
        IF(T15 .GE. 200.0) THEN          $      record 11
            CALL TPRINT('SUB1')          $      record 12
            TIMEND = TIMEN               $      record 13
        ENDIF                            $      record 14
END OF DATA
```

Table 1-2 shows the utility of this approach by showing the previous input file restructured to take advantage of these spreadsheet-like *registers*. In addition to providing centralized control and utilities for fast, consistent changes, registers facilitate model maintenance and improve self-documentation. They also enable rapid parametric and sensitivity analyses, as well as complete design optimization options.

Expressions can be used almost anywhere within SINDA/FLUINT, and they can be arbitrarily complex. They can even refer to model inputs or outputs, and can contain conditional operators. For example, the logic to terminate the run can be placed in the definition of TIMEND (the problem end time):

```
TIMEND = (sub1.t15 >= 200.0) ? timen : 1000.0
```

1.5.3 Optimizing the Shape of a Heated Bar

If the cross section of the bar were variable at three locations (cross section), as shown in Figure 1-3, then the optimum shape of the bar could be found as a function of some thermal performance criterion. As shown in that figure, the nodalization has changed somewhat: node 1 is located at the edge of the bar and node 3 is located at the opposite edge to take into account the temperature drop across the entire length. Node 2 is located in the middle. The cross sectional thickness at the ends and center of the bar, denoted X_1, X_2, and X_3, are variables to be calculated by the program. X1, X2, and X3 are therefore added as new registers. The masses of the nodes are then calculated as:

```
1,500.0,(0.5*X1)*CP*MPER
2,500.0,(0.5*(X1+X3)+X2)*CP*MPER
3,500.0,(0.5*X3)*CP*MPER
```

Introduction and Overview

80

3.8 NODE DATA

Four types of nodes may be defined in the NODE DATA block: diffusion, arithmetic, boundary, and heater.

Diffusion nodes have thermal mass (capacitance) and thus store and release energy. In the transient network solution routines, diffusion node temperatures are calculated by a finite difference representation of the partial differential heat transfer equation. Three locations in core memory are reserved for each diffusion node in an active submodel: one location to store the temperature of the node, one to store its capacitance, and one for the heat source* (if any) impressed on the node. Diffusion node data input options are provided to accommodate capacitance values which are not constant (i.e., vary with temperature, etc.)

Arithmetic nodes have a capacitance of zero. The temperatures of arithmetic nodes are calculated using a finite difference representation of Poisson's equation**. Since there is no capacitance value to store, only two core locations are reserved for each arithmetic node: one for the temperature, and one for the impressed heat source (if any).

Boundary nodes have no capacitance and may not receive an impressed heat source. A single core location is reserved to store the temperature of each boundary node. These temperatures are not altered by the network solution routines, but may be modified as desired by the user.

Heater nodes are used for the purpose of establishing heater power requirements. They act as boundary nodes except that they report the amount of heat removed or added to maintain the given temperature. This energy is reported using calls to HNQCAL.

3.8.1 Node Data Input

All NODE DATA records appear after:

HEADER NODE DATA, smn [,fac] [,tcon]

where:

    smn .... submodel name
    fac .... capacitance multiplying factor
    tcon ... temperature conversion flag
    [ ] .... indicates optional argument

The submodel name is any identifying name, up to 8 characters. tcon is a code word used to convert the temperature on the NODE DATA records from one units base to another, if desired. Allowable tcon options are:

---

* As with SINDA, these source locations are used by the solution routines as net heat rates. The impressed heat rates must be reinitialized each time step if they are changed at all.

** That is, arithmetic nodes are brought into a steady state heat balance with the boundary conditions and any neighboring diffusion nodes.

```
FTC ....... Fahrenheit to Centigrade
CTF ....... Centigrade to Fahrenheit
FTR ....... Fahrenheit to Rankine
RTF ....... Rankine to Fahrenheit
CTK ....... Centigrade to Kelvin
KTC ....... Kelvin to Centigrade
FTK ....... Fahrenheit to Kelvin
KTF ....... Kelvin to Fahrenheit
CTR ....... Centigrade to Rankine
RTC ....... Rankine to Centigrade
RTK ....... Rankine to Kelvin
KTR ....... Kelvin to Rankine
```

Please note that it is the user's responsibility to ensure that all submodels are consistent as to temperature and other dimensional parameters. Temperature units are controlled by tcon and control constant ABSZRO. (See Section 4.6: Advanced Control Constant Usage.)

There are several codes which may be used on NODE DATA records. The codes provide options which enable the user to input a large variety of nodes in a simple, convenient manner. For all options, the following points apply:

1.   Diffusion nodes must be given a positive node number and a positive capacitance.

2.   Arithmetic nodes must be given a positive node number and a negative (not zero) capacitance. (The negative capacitance value acts as a signal that the node is of the arithmetic type.)

3.   Boundary nodes must be given a negative node number and a non-negative capacitance (e.g., 0.0). The capacitance value is ignored but is present for the sake of consistency, and the negative sign on the node number serves as a flag to signify that the node is of the boundary typ

4.   Heater nodes must be given a negative node number and a negative capacitance. Heater nodes act the same as boundary nodes as far as their effect on the node/conductor network. The difference is that the user can conveniently printout the net heat required to maintain the fixed temperature, thus obtaining heater sizing information.

5.   An initial temperature must be specified for all nodes, regardless of type.

6.   Actual node numbers may be as large as six digits (e.g., 999999).

Table 3–6 summarizes the NODE DATA input options which are available to the user. Before proceeding to a detailed discussion of each option, two points should be clarified: (1) *impressed heat sources are not input with node data*; they are input in the source data block, in which case they are transferred to the source locations automatically when needed and (2) the tables of nodal capacitances and heat sources are always accessible to the user in the logic blocks, and hence, a "constant capacitance value," from the standpoint of NODE DATA input, need not be held constant during the entire course of a solution.

Standard Option ("3 blank" card code) – The simplest NODE DATA record utilizes the blank code. Records using this code must contain three data values for each node, as follows:

Table 3-6: Summary of NODE Data Input Options

OPTION (CODE)	NODE TYPE				DESCRIPTION
	D	A	B	H	
3 blanks	■	■	■	■	To input a single node where the capacitance is given as a single, constant value
CAL	■				To input a single node where the capacitance will be calculated by the preprocessor from four factors input by the user
GEN	■	■	■	■	To generate a group of nodes, each having the same initial temperature and the same capacitance
SIV SPV	■				To input a single node where the capacitance varies with temperature. For SIV, the capacitance is found by interpolation using an array of temperature vs. capacitance. For SPV, the capacitance is found by computing an Nth order polynomial function of temperature
SIM SPM	■				To generate a group of nodes, each having the same initial temperature and the same temperature-varying capacitance. For SIM, C is found by interpolation using an array of T vs. C. For SPM, C is found by computing a polynomial in T
DIV DPV	■				To input a single node consisting of two materials which have different temperature-varying capacitances. For DIV, C1 and C2 are taken from arrays of T vs. C. For DPV, C1 and C2 are computed from polynomials in T
DIM DPM	■				To generate a group of nodes, each of which consists of the same two materials having temperature-varying capacitances. For DIM, C1 and C2 are taken from arrays of T vs. C. For DPM, C1 and C2 are computed from polynomials in T.
BIV	■				To input a single node where the capacitance is a function of time and temperature. The capacitance is found by interpolation using an array of time and temperature vs. capacitance

```
N#,Ti,C                           $(basic format)
5,70.0,0.75*2.0                   $ example 1
6,80.0,-1.0                       $ example 2
-9,90.0,0.0                       $ example 3
-7,70.0,0.0,8,85.0,-1.0           $ example 4
```

where:

N#   . . . . . Actual node number assigned by the user (integer data value)

Ti   . . . . . Initial temperature of the node (floating point data value)

C   . . . . . . Capacitance of the node (floating point data value)

The order and type of the three data values must always be exactly as shown. The rules determining the type of input node are as follows: if N# and C are positive, then the triplet defines a diffusion node; if N# is positive and C is negative, an arithmetic node is defined; if N# is negative and C is zero, then a boundary node is defined. More than one node may be defined on a single card by placing more than one set of three data values on it.

Example 1 creates a diffusion node with an actual node number of 5, an initial temperature of 70.0°, and a capacitance of 1.5. Example 2 creates arithmetic node number 6 with an initial temperature of 80.0°. Example 3 creates boundary node number 9 (not -9, the negative sign serves only as a flag!) with an initial temperature of 90.0°. Example 4 shows how the data values for two nodes may be placed on the same record. The two nodes created in example 4 are as follows: boundary node number 7 at a temperature of 70.0°, and arithmetic node number 8 at a temperature of 85.0°.

CAL Option – The SINDA '85 feature that allows Fortran arithmetic (limited to +,-,*,/) in data fields make the CAL input option obsolete. The CAL option, therefore, is available but not documented for SINDA '85.

GEN Option – The GEN option is used to generate and input a group of nodes, each having the same initial temperature and the same capacitance. This option requires five data values for each group of similar nodes, as follows:

```
GEN N#,#N,IN,Ti,C          $(basic format)
GEN -20,3,5,70.0,1.2        $ example 5
```

where:

N#	. . . . .	Actual node number of the first node to be generated (integer)
#N	. . . . .	Total number of nodes to be generated (integer)
IN	. . . . .	Increment to be added to the current node number to form the node number of the next node (non–zero integer)
Ti	. . . . .	Initial temperature of all nodes (floating point)
C	. . . . . .	Capacitance of all nodes (floating point)

If C and N# are positive, a group of diffusion nodes will be generated. If C is negative a group of arithmetic nodes is produced. If N# is negative, then the nodes will all be of the boundary type.

It is important to remember that the negative sign on the N# serves only as a flag to identify a boundary node during input. For reference purposes, the negative sign is deleted. Hence, example 5 will generate three boundary nodes having a temperature of 70.0°, with the node numbers being 20, 25, and 30 (not −20, −15, and −10). The minus sign on the initial N# signifies that the entire group will be boundary nodes. A minus sign in the initial N# and the initial C signifies that the entire group will be heater nodes.

The eight–data–value GEN option, like the CAL option, has been made obsolete by the Fortran arithmetic feature of SINDA '85. It is available but not documented for SINDA '85.

Automated Variable Capacitance Options – As stated earlier, the network solution routines call for the execution of the operations in the VARIABLES 1 block prior to performing each iteration on the heat transfer equations. VARIABLES 0 is called after each time step update in a transient solution. Using this feature, the user can set time or temperature variable capacitances by specifying VARIABLES 1 or VARIABLES 0 operations which would update values in the node capacitance table. The new capacitance could be calculated using the appropriate algorithm, or interpolation subroutine and independent variable. Though straightforward, this approach tends to be rather cumbersome. It is, however, more computer time efficient than the automated options described next.

The automated options cause temperature–dependent capacitances to be calculated wholly within the network solution routines, at the end of the VARIABLES 1 operations, and therefore relieve the user of the task of preparing and inputting capacitance computation algorithms. As specified by the user, one of two methods is used to update the current value

of a temperature-dependent capacitance: (1) interpolation, and (2) polynomial evaluation. For interpolation, the user must input, in an ARRAY DATA block, a doublet array of (temperature, capacitance) ordered pairs representing points on the curve of T vs. C. The capacitance is calculated by performing linear interpolation on this array using the current temperature of the node as the independent variable. For polynomial evaluation, the user must input a singlet array of coefficients (P0,P1,P2,...,Pn) in an ARRAY DATA block. The capacitance is then calculated by evaluating the nth order polynomial in T (i.e., $P0+P1*T+P2*T**2+...+Pn*T**n$). These two methods enable the automated options to be applicable to a vast majority of the variable capacitance nodes encountered in modeling real systems.

To enable the same array of points or coefficients to be used for all nodes made of the same material, all automated options cause the computed capacitance to be multiplied by a factor before being inserted in the capacitance table. In this way, for example, the array can be used to compute Cp, and the multiplying factor can be used for V. In another application, the multiplying factor might be used to scale the capacitance to some set of consistent units.

In all cases, the multiplying factor may be input as a floating point data value, or as a user constant reference of the form Kn or smn.Kn, where n is the actual constant number. If the latter form is used, the referenced constant must be a floating point value.

The interpolation options (except BIV*) require a doublet array of temperature vs. capacitance to be input in an ARRAY DATA block. (Capacitance is used loosely in this context. The array could contain, for example, values of temperature vs. specific heat, as long as the associated multiplying factor had units of mass.) The array must contain an even number of floating point data values, as follows:

$$T1,C1, \quad T2,C2, \quad ..., \quad Ti,Ci, \quad ..., \quad Tn,Cn$$

The temperature values, Ti, must be strictly increasing with i. If the node temperature at some time is less than T1, then C1 will be used in lieu of performing an extrapolation; if the temperature is greater than Tn, then Cn will be used.

The polynomial options require that an array of coefficients be input in an ARRAY DATA block. The array must consist solely of n+1 floating point values, where n is the order of the polynomial. The coefficients should be entered in the array in the order of increasing powers of temperature (e.g., the first data value is the constant term, P0, and the (n+1)th data value is the coefficient, Pn, of $T**n$). For both classes of options, the arrays are referenced by using the form An or smn.An where n is the array number.

SIV and SPV Options – These options allow the user to define single nodes having temperature varying capacitances. The SIV option assumes that the referenced array is to be used for linear interpolation, whereas the SPV option assumes that the array contains polynomial coefficients. Both options require four data values for each node, as follows:

---

* The BIV option requires a bivariate array of time and temperature vs capacitance.

```
SIV N#,Ti,AP,F              $ (basic format)
SPV N#,Ti,AC,F              $ (basic format)
SPV 25,70.0,A8,0.0001       $ example 6
SIV 25,70.0,A9,0.0001       $ example 7
SPV 25,70.0,A8,K5           $ example 8
SIV 25,70.0,A9,K5           $ example 9
```

where:

N#	. . . . .	Actual node number (integer)
Ti	. . . . .	Initial temperature of node (floating point)
AP,AC	. .	Reference to an array of T vs. C points, or polynomial coefficients, respectively
F	. . . . . .	Multiplying factor input as a floating point data value, or a user constant reference of the form: Km or smn.Km, where m is the user constant number

The following example is presented to clarify the usage of these two options. Consider a temperature varying capacitance given by the equation:

$$C = (0.0001) \times (0.005 \times T^2 + 1.0)$$

This equation represents the curve shown in Figure 3-3. To use the SPV option to define node 25, which has a capacitance given by the equation above, the user would prepare a NODE DATA record as shown in example 6. The referenced array of polynomial coefficients, user array number 8, would be entered in an ARRAY DATA block, as follows:

```
8=1.0,0.0,0.005
```

To use the interpolation option, SIV, to define this node, the user would prepare a node data record as shown in example 7. The referenced array of T vs. C points, A9, represents a piecewise linear approximation of the curve in Figure 3-3, and would be entered in the appropriate ARRAY DATA block as follows:

```
9 = 0.0,1.0,10.0,1.5,20.0,3.0,40.0,9.0
```

Examples 6 and 7 show that the multiplying factor, 0.0001, was entered directly on the NODE DATA record as a floating point data value. This factor could also have been entered as a user constant in a USER DATA block, in which case, it would be referenced by its actual constant number on the NODE DATA records (see examples 8 and 9). The required USER DATA record would appear as follows:

```
5 = 0.0001
```

SIM and SPM Options – These two options combine the features of the SIV and SPV options with those of the GEN option. That is, they allow the user to generate and input a group of nodes all having the same initial temperature and the same temperature varying capacitance. The SIM option generates nodes whose capacitance will be evaluated by interpolation, and the SPM option generates nodes whose capacitance will be evaluated from a polynomial. Six data values are required to define each group of nodes, as follows:
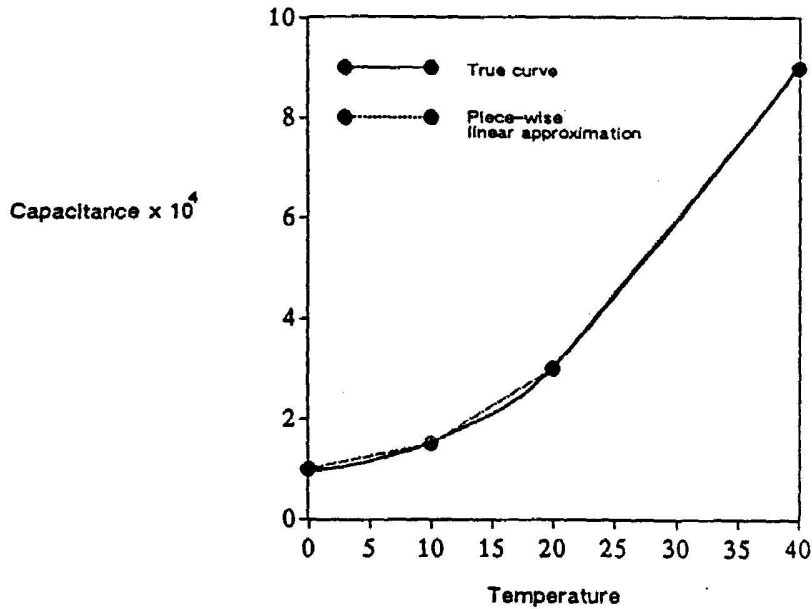
Figure 3-3: Curve of Temperature-Varying Capacitance

```
SIM N#,#N,IN,Ti,AP,F              $(basic format)
SPM N#,#N,IN,Ti,AC,F              $(basic format)
SIM 25,3,5,70.0,UA(9),UC(5)       $ example 10
```

where:

N#   . . . . .   Actual node number of the first node (integer)

#N   . . . . .   Total number of nodes to be generated (integer)

IN   . . . . .   Increment to be added to the current node number to form the number of the next node (non-zero integer)

Ti   . . . . .   Initial temperature of all nodes (floating point)

AP,AC   . .   Reference to an array of points of T vs. C, or polynomial coefficients, respectively. An or smn.An form, where n is the array number)

F   . . . . . . Multiplying factor (floating point data value or user constant reference of the form: Km or smn.Km, where m is the constant number)

Example 10 defines three nodes, numbers 25, 30, and 35, whose capacitance will be computed by interpolating on array 9, with the results multiplied by constant 5.

DIV and DPV Options – These options allow the user to define a node which is made of two materials, each having different temperature varying capacitance properties. Both require an array and a multiplying factor for each of the two materials. For the DIV option, the result of interpolating on the first array, times the first multiplying factor, is added to the result of doing likewise with the second array and the second multiplying factor. The action of the DPV option is similar, except that the arrays are assumed to contain polynomial coefficients. The formats are shown below:

```
DIV N#,Ti,AP1,F1,AP2,F1        $(basic format)
DPV N#,Ti,AC1,F1,AC2,F2        $(basic format)
DIV 40,70.0,A8,K5,A6,2.0       $ example 11
```

where:

N#	Actual node number (integer)
Ti	Initial temperature (floating point)
AP1 , AP2	References to arrays of T vs. C points (An or smn.An form, where n is the array number)
AC1 , AC2	References to arrays of polynomial coefficients (UA(n) form)
F1 , F2 . .	Multiplying factors (floating point data value, or user constant reference of the form Km or smn.Km, where m is the constant number)

Example 11 defines node 40, whose capacitance will be computed as the sum of interpolation on array 8 times constant 5, plus interpolation on array 6 times 2.0.

If one of the materials does not have a temperature varying capacitance, the user may substitute a floating point data value for one of the array references of the NODE DATA card. This substitute value will then be used in place of the result of interpolation (or polynomial evaluation) on the corresponding array. This feature makes the following alternate record formats possible:

```
DIV N#,Ti,SUB,F1,AP2,F2        $(alternate format)
DPV N#,Ti,SUB,F1,AC2,F2        $(alternate format)
DIV N#,Ti,AP1,F1,SUB,F2        $(alternate format)
DPV N#,Ti,AC1,F1,SUB,F         $(alternate format)
DIV 35,70.0,1.0,K5,A9,0.003    $ example 12
```

where **SUB** is a floating point data value.

Example 12 illustrates the case where one of the two materials has a constant capacitance. Since the capacitance of the first material will be calculated as $1.0 * K5$, it would be quite acceptable to alter the value of constant K5, at some point in an operations block, and thereby effect a change in the capacitance of this "constant capacitance" material.

DIM and DPM Options – These two options combine the features of the DIV and DPV options with those of the GEN options. They are used to generate and input a group of nodes, all of which have the same initial temperature and all of which are of the same two materials having temperature varying capacitances. The DIM option causes the capacitance for the two materials to be evaluated by interpolating on the referenced arrays, and the DPM option computes the capacitances from polynomial coefficients supplied in the arrays. As for the other dual material options, multiplying factors are applied to the results of the calculations on the arrays. Eight data values are required to define each group of nodes, as follows:

```
DIM N#,#N,IN,Ti,AP1,F1,AP2,F2    $(basic format)
DPM N#,#N,IN,Ti,AC1,F1,AC2,F2    $(basic format)
DPM 30,3,5,70.0,A8,0.1,A6,K5     $ example 13
```

where:

N#	. . . . .	Actual node number of the first node (integer)
#N	. . . . .	Total number of nodes to be generated (integer)
IN	. . . . .	Increment to be added to the current node number to form the node number of the next node (non–zero integer)
Ti	. . . . .	Initial temperature of all nodes (floating point)
AP1,AP2		References to arrays of points of T vs. C (An or smn.An form, where n is the array number)
AC1,AC2		References to arrays of polynomial coefficients (An or smn.An form)
F1,F2	. .	Multiplying factors (floating point data value, or user constant reference of the form: Km or smn.Km, where m is the constant number)

Example 13 defines 3 nodes, numbers 30, 35, and 40, each made of two materials which have a temperature varying capacitance. Polynomial coefficients for the first material are supplied in array 8, and coefficients for the second material are supplied in array 6.

As for the DIV and DPV options, if one of the materials has a constant capacitance, the array reference for that material may be replaced by a floating point data value. The value so supplied or referenced will be used in place of the result of interpolations or polynomial evaluation on the corresponding array. This feature permits the following formats:

```
DIM N#,#N,IN,Ti,SUB,F1,AP2,F2    $(alt format)
DPM N#,#N,IN,Ti,SUB,F1,AC2,F2    $(alt format)
DIM N#,#N,IN,Ti,AP1,F1,SUB,F2    $(alt format)
DPM N#,#N,IN,Ti,AC1,F1,SUB,F2    $(alt format)
```

where SUB is a floating point data value.

BIV Option – The BIV options enables the user to define a node having a capacitance which is a function of both time and temperature. The format for such a node is similar to that for the SIV option, and requires four data values for each node, as follows:

```
BIV N#,Ti,AB,F                    $ (basic format)
BIV 55,70.0,A2,K4                 $ example 14
```

where:

N# . . . . . Actual node number (integer)

Ti . . . . . Initial temperature of the node (floating point)

AB . . . . . Reference to a bivariate array of temperature and time vs. capaci-
              tance. (An or smn.An form)

F . . . . . . Multiplying factor (Floating point data value, or user constant refer-
               ence of the form: Km or smn.Km, where m = constant number)

Examples 14 defines node 55 whose capacitance, as a function of time and temperature,
is supplied in bivariate array number 2. Section 3.7.2 contains a description of the structure
of a bivariate array. In this application, the X independent variable should be temperature,
the Y independent variable should be mean time, and the Z dependent variable should be
capacitance. When evaluating a node defined with the BIV option, the network solution
routines will perform a bivariate interpolation on the referenced array. Using the current
temperature of the node and the mean time for the current computation interval (control
constant TIMEM) as the values of the independent variables.

3.8.2   Node Data Referencing

Each node is assigned an arbitrary submodel name and reference number by the user in
the NODE DATA blocks. As each node is accepted by the preprocessor, it is renumbered in
sequence. The user assigned reference number is called the actual, or node identification
number, and the preprocessor assigned sequence number is called the relative number.
There is no restriction as to what order the four node types appear in the input file. After
sorting by the preprocessor, they will be in diffusion, arithmetic, heater, and boundary order
within each submodel. This is done for efficiency in solution routine execution and memory
utilization.

Although the Fortran subroutines resulting from the translation of the SINDA operations
block use the relative numbering system to access the node tables, it would be tedious and
very error prone for the user to do so directly. Hence, the preprocessor maintains a table of
actual vs. relative node numbers, and performs the conversion from the former to the latter
whenever a temperature, capacitance or source table reference is encountered in the logic
blocks. The node tables are referenced by the following forms:

```
smn.Tn = temperature
smn.Qn = source
smn.Cn = capacitance
```

where n is the Actual node number

By making the actual numbering system available outside the NODE DATA block, the preprocessor relieves the user of the burden of keeping track of the relative input order of the nodes. For example, the temperature of actual node number 6 in submodel SUBN is referenced by SUBN.T6, its capacitance by SUBN.C6, and its source by SUBN.Q6, regardless of the relative position of these quantities in their respective tables.

The smn.X(n) format is the "long" or complete reference form. This may be shortened to simply Xn when the submodel name is being supplied by an argument in the preceding header record. If the reference is to a T, Q or C outside the submodel specified in the header record, then the long form is required. The OPERATIONS DATA block is used for initialization and its logic relates to the entire model (as opposed to a single submodel) thus, the long form is the only one allowed in OPERATIONS DATA, unless a DEFMOD is used.

The Input File

3.9   SOURCE DATA

SOURCE DATA blocks provide the user with a convenient means for defining heat sources or sinks which are to be impressed on the nodes. Source must always be input in units of *energy per unit time*. A positive source means energy *input* to the node.

All SOURCE DATA records appear following a header record:

HEADER SOURCE DATA, smn

where smn is a submodel name.

The user should note that SOURCE DATA is really defining a node attribute, so there must be strict correspondence between the full node identification (submodel name plus node number) in all records within all the NODE DATA and SOURCE DATA blocks.

3.9.1   Source Data Input

The SOURCE DATA input options and their associated 3-character codes are summarized in Table 3-7.

Table 3-7: Summary of SOURCE DATA Input Options

OPTION (CODE)	DESCRIPTION
3 blanks	To impress a constant heat source on a single node
GEN	To impress the same heat source on several nodes
SIV	To impress a temperature-varying heat source on a node
SIT[†]	To impress a time-varying heat source on a node
DIT[†]	To impress the sum of two time-varying heat sources on a node
TVS	To impress a time-varying heat source on a node
TVD	To impress the sum of two time-varying heat sources on a node
DTV	To impress the sum of a time-varying source and a temperature-varying source on a node
CYC[†]	To impress a cyclic time-dependent source on a node
PER	To impress a cyclic time-dependent source on a node

[†]   Beginning with SINDA '85, the SIT, DIT, and CYC options have been replaced with the TVS, TVD, and PER options respectively. The obsolete options are included only for processing old SINDA files.

Standard Option (3 blanks) – This option impresses a constant source on a single node. The input format is as follows:

```
N#,Q •                              $(basic format)
4,1.8,6,K3  or smn.K3               $  (example 1)
```

where:

    N# .... Actual node number of a node defined in a NODE DATA block (integer)

    Q .... The value of the source (floating point data value, or user constant reference of the form: Kn or smn.Kn, where n is the actual number of a floating point user constant)

Example 1 will cause a heat rate of 1.8 to be impressed on node number 4, and the floating point heat rate stored in user constant number 3 to be impressed on node number 6.

GEN Option – The GEN options allows the user to impress the same heat source on a group of several nodes. The format is as follows:

```
GEN N#,#N,IN,Q                      $(basic format)
GEN 6,3,1,4.3               .        $(example 2)
```

where:

    N# ..... Actual node number of the first node to receive the source (integer)

    #N ..... The total number of nodes to receive this source (integer)

    IN ..... The increment to be added to the current node number to form the next node number (non–zero integer)

    Q ...... The value of the source (floating point data value, or user constant reference of the form: Kn or smn.Kn, where n is the user constant number)

Example 2 will cause a heat rate of 4.3 to be impressed on nodes 6, 7, and 8.

Automated Variable Source Options – The automated options provide for operations which will automatically evaluate the current value of a heat source which varies with time or temperature (or both), and place this value in the appropriate source location.

All options cause the source to be evaluated by performing linear interpolation on arrays of values which represent points on a curve of time or temperature vs. heat rate. If the heat source is temperature dependent, the values are entered in the ARRAY DATA block in the following doublet array format:

```
x1,y1,  x2,y2,  ...,  xn,yn
```

  •   In order to enforce the exact correlation between source data and node data, the submodel reference form: smn.N# is not allowed in any SOURCE DATA option.

where:

yi . . . . . Dependent variable – heat rate

xi . . . . . Independent variable – temperature; xi (i = 1,2,...,n) is strictly increasing with i

xi,yi . . Floating point numbers

and are referenced in the SOURCE DATA block by using the form: Am or smn.Am, where m is the array reference number.

For temperature–dependent sources, the value of the independent variable used for interpolation will be the current temperature of the node receiving the source. If the value of the independent variable is less than x1, then y1 will be used in lieu of performing an extrapolation. Similarly, if the value is greater than xn, then yn will be used. To provide further versatility, the value of the heat rate found from interpolation will be multiplied by a factor before being entered in the source table. This multiplying factor can assume any significance that is convenient for the user. If the multiplying factor has any units, the product of these units and those of the dependent variables in the array must be units of heat rate.

SIV Option – The SIV option allows the user to specify a temperature varying source. The record format is as follows:

```
SIV N#,A,F              $(basic format)
SIV 8,A4,1.0            $(example 3)
```

where:

N# . . . . . Actual node number of the node which will receive this source (integer)

A . . . . . . Reference to doublet array of temperature vs. heat rate, of the form: An or smn.An, where n is the actual array number

F . . . . . . Multiplying factor (floating point data value, or a reference to a floating point user constant of the form: Km or smn.Km, and where m is the actual constant number)

The following discussion, and example 3 above, illustrates the use of the SIV option.

Assume that the heat rate defined by the graph in Figure 3–4 is to be impressed on node number 8. This graph can be represented by the following doublet array, which would be entered in an ARRAY DATA block:

```
4 =0.0,0.0,10.0,0.1,20.0,0.05,30.0,0.05,40.0,0.0
```

NOTE: These doublet arrays are not allowed in TARRAY DATA blocks. The required heat source is then defined in the source data by the record shown in example 3. Since the referenced array (array number 4) does not need to be scaled or modified in any way, the multiplying factor is merely given as the floating point data value 1.0.
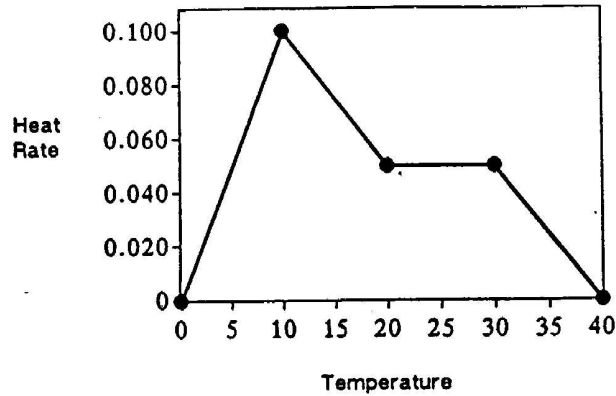
Figure 3-4: Temperature-Varying Heat Rate

TVS Option – The TVS option allows the user to apply a time varying source to a node. The difference between the TVS and SIV options is that the array referenced with the TVS option may be entered in either the ARRAY DATA or TARRAY DATA blocks and must be in the singlet form. There is a restriction that both the time array and q-arrays must be in the same data block. The ARRAY or TARRAY DATA formats are as follows:

```
nt    =  t1,t2,t3 - - - - tn
nq    =  q1,q2,q3 - - - - qn
```

where:

```
nt    . . . . .  Time array number
ti    . . . . .  Time values (independent variable, values strictly increasing with i)
nq    . . . . .  Heat rate array number
qi    . . . . .  Heat rate values (dependent variable)
ti,qi  . .  Floating point numbers
```

The TVS SOURCE DATA input format is as follows:

```
TVS N#, At, Aq, F          $(basic format)
TVS 16,A1,A9,K13           $(example 4)
```

where:

N# . . . . . Actual node number of the node which will receive this source (integer)

At . . . . Reference to singlet time array of the form: An or smn.An, where n is the array number

Aq . . . . Reference to singlet heat source array same form

F . . . . . . Multiplying factor (floating point data value, or a reference to a floating point user constant of the form: Km or smn.Km, where m is the actual constant number)

Example 4 causes the results of interpolating on arrays 1 and 9, (using the mean time for the computation interval (TIMEM) as the independent variable) multiplied by the value in user constant 13 to be inserted in the source location for node number 16, Q16.

TVD Option – The TVD option is used to specify a heat source which is the sum of two separate, time-varying sources. The user must supply two singlet time vs. heat rate arrays, in either an ARRAY or TARRAY DATA block, and a multiplying factor for each of the two sources, as follows:

```
TVD  N#,At,Aq,FA,Bt,Bq,FB        $(basic format)
TVD  3,A1,A5,K2,A1,A6,K4         $(example 5)
```

where:

N# . . . . Actual node number of the node which will receive this source (integer)

At,Aq,Bt,Bq . References to singlet arrays of time vs. heat rate for source A and source B, respectively of the form: An or smn.An, where n is the array number

FA,FB . . . . . . Multiplying factors to be applied to the results of interpolation. (floating point data values, or references to floating point user constants, of the form: Km or smn.Km, where m is the actual constant number)

Example 5 defines a source to be impressed on node 3, which will be evaluated by taking the sum of the interpolation result from array 5 times the value of user constant 2, and the interpolation result from array 6 times the value of user constant 4. Both interpolations use time array 1 for the independent variable data.

DTV Option – This option enables the user to specify a source which is the sum of a time varying source and a temperature varying source. The DTV option requires separate doublet arrays for time vs. heat rate and temperature vs. heat rate, and represents the summation $Q = f(t) + g(T)$. The record format is as follows:

```
DTV  N#,At,Ft,AT,FT               $(basic format)
DTV  26,A14,K1,A16,K3             $(example 6)
```

where:

N# . . . . . Actual node number of the node which will receive this source (integer)

At . . . . . Reference to a doublet array of time vs. heat rate, (form: An or smn.An, where n is the array number)

Ft . . . . . Multiplying factor to be applied to the result of interpolation on array At (floating point data value or a user constant reference of the form: Km or smn.Km, where m = actual constant number)

AT . . . . . Reference to a doublet array of temperature vs. heat rate, form: An or smn.An, where n is the array number

FT . . . . . Multiplying factor to be applied to the result of interpolation on array AT (floating point data value, or a reference to a floating point user constant of the form: Km or smn.Km, where m is the actual constant number)

Example 6 defines a source, to be impressed on node 26, which will be evaluated at each iteration, as the sum of a time varying component (calculated from array 14 and user constant 1) and a temperature varying component (calculated from array 16 and constant 3).

The DTV option may be used to define a source which is the sum of (1) a time varying source and a constant source, or (2) a constant source and a temperature varying source. (Case 1 is equivalent in effect to the alternate form for the TVD option, but is also made available to the DTV option for the sake of consistency.) Case 1 is accomplished by replacing the array reference, At, with a floating point data value. Case 2 is accomplished by replacing the array reference, AT, with a floating point data value. In either case, the value of the constant source is taken as the product of the data value, or referenced constant, and its associated multiplying factor. The alternate formats are as follows:

```
DTV  N#,SUB,Ft,AT,FT              $(case 1)
DTV  N#,At,Ft,SUB,FT              $(case 2)
```

where SUB is a floating point data value.

PER Option – The PER option in the SOURCE DATA block provides for automated cyclic interpolation of time varying heat rates. The record format is as follows:

```
PER  N#,At,Aq,F,D,P
PER  12,A1,A5,1.0,.5,1.0
PER  13,A1,A5,1.0,.75,1.0          $(example 7)
PER  14,A1,A5,1.0,0.,1.0
PER  15,A1,A5,1.0,.25,1.0
```

where:

N# . . . . .   Actual node number (positive integer)

At,Aq . .   Reference to singlet arrays of mean time vs. heat rate points (integer) The time array must start at zero. Reference form is An or smn.An where n is the actual array number

F . . . . . .   Multiplying factor (floating point data value, or reference to a user constant)

D . . . . . .   Phase displacement as a decimal fraction of the total period. (Must be a floating point literal with a value between 0.0 and 1.0. A user constant reference is not allowed.)

P . . . . . .   Period specified in time units (floating point data value or reference to a user constant). P is usually the last time point in the At array, but not restricted to this value.

If P is less than the last At array value, any values between P and the last array value are not used. If P is greater than the last array time value, the heating value used for the time period between the last array time value and P is constant and equal to the heating rate input for the last array time value.

Example 7 defines cyclic sources to be impressed on nodes 12, 13, 14, and 15 of a cylindrical spacecraft rotating clockwise normal to the sun at one revolution per hour. The heating rates to the four small surfaces shown in Figure 3-5 can be obtained as follows.

Figure 3-5: Cyclic Heat Source Example
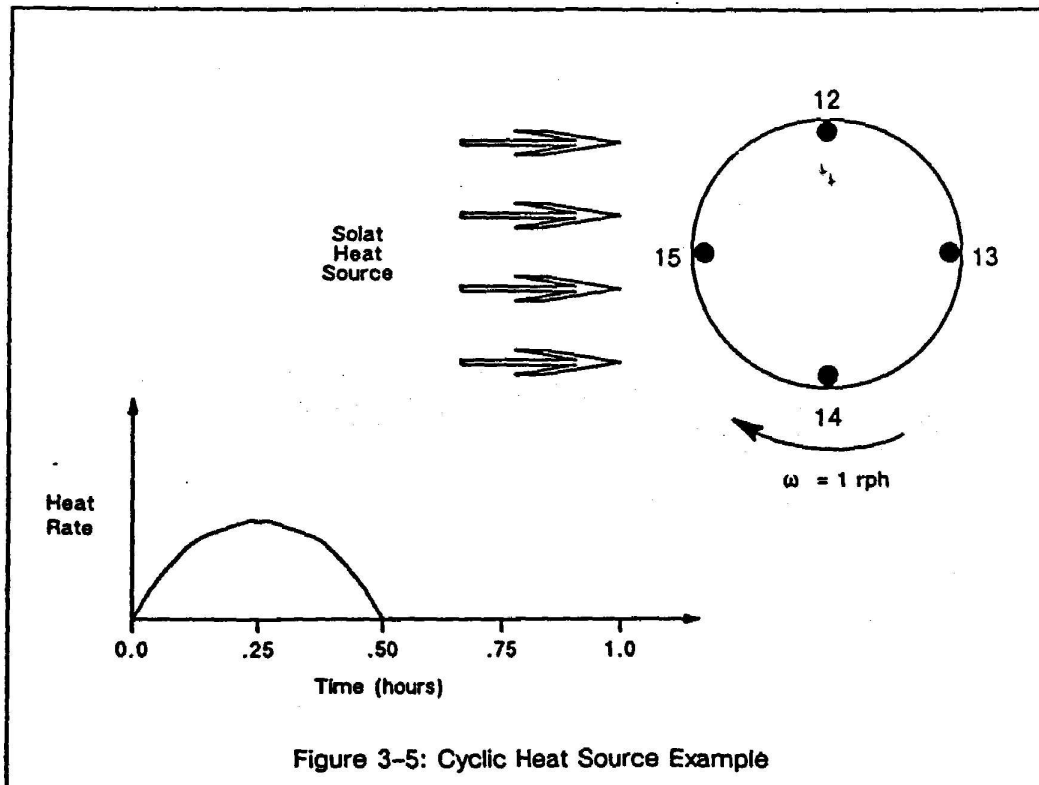
The heating data is entered in an ARRAY DATA or TARRAY DATA block in the usual tabular form. For this case, data is assumed to be entered into arrays 1 and 5. With the cylindrical spacecraft rotating clockwise and with the above orientation of each surface with respect to the sun at time zero, the phase displacements are as follows:

NODE	PHASE DISPLACEMENT
14	0.0
15	0.25
12	0.50
13	0.75

The period is 1.0 hour.

3.9.2   Source Data Updating

The time and temperature dependent SOURCE DATA inputs defined in the SOURCE DATA blocks are actually applied (that is, stored in the source locations) at the end of VARIABLES 0 and VARIABLES 1 for time and temperature dependent operations, respectively. The DTV option (time and temperature dependent) is done in VARIABLES 1 prior to each temperature iteration.

These updates are done at the end of the VARIABLES 0 and VARIABLES 1 subroutines through calls to program subroutines QVTIME and QVTEMP. The preprocessor normally supplies these calls subsequent to any user–supplied logic in the VARIABLES 0 or VARIABLES 1 blocks. This gives the user an opportunity to update any user constants or arrays that may affect the SOURCE DATA calculations before they are done. If intervention is required after the QVTIME/QVTEMP calls, the user may supply a CALL QVTIME statement anywhere in the VARIABLES 0 input and it will not be overridden by a preproc. sor–supplied call. This is also true for QVTEMP in VARIABLES 1.

To conform with previous SINDA practice, *all time and temperature dependent of source terms are set to zero at the beginning of Variables 0*. All constant value sources input in the SOURCE DATA block are also reset to their initial values at the beginning of VARIABLES 0.

All source data referencing in the logic blocks is done using the forms: Qn or smn.Qn where smn is a submodel name and n is the actual node number. As usual, the smn. prefix is required only in the OPERATIONS or SUBROUTINE DATA blocks and for cross–references to a node outside the submodel stated in the header or DEFMOD record currently in effect. With this accessing capability, the user may update source data values at will with statements in any logic block.

3.10   CONDUCTOR DATA

Two basic types of conductors may be defined by the user: linear and radiation. The conductance of a linear conductor is input in units of *energy per unit time per unit degree*[*], and the heat flow rate through such a conductor is calculated in the network solution routines as:

$$Q = G \cdot (T_i - T_j)$$

where:

        Q .... Heat rate (energy/time)
        G .... Conductance
        T .... Temperature

Several types of physical heat transfer mechanisms can be modeled as linear conductors. For heat transfer by conduction, the conductance should be computed as:

$$G = \frac{k \cdot A}{L}$$

where:

        k .... Thermal conductivity (energy/length–time–deg)
        A .... Cross-sectional area of the conduction path (length^2)
        L .... Length of the conduction path (length)

For heat transfer by convection, the conductance should be computed as:

$$G = h \cdot A$$

where:

        h .... Convective film coefficient (energy/length^2–time–deg)
        A .... Surface area (length^2)

For heat transfer by mass flow, the conductance should be computed as:

$$G = \dot{m} \cdot Cp$$

where:

        $\dot{m}$ .... Mass flow rate (mass/time)
        Cp .... Specific heat of the flowing material (energy/mass–deg)

The conductance of a radiation conductor is input in units[*] of *energy per unit time per degree*[4]. The value input as the conductance of a radiation conductor should be computed as:

---

[*] Temperature can be in degrees F, C, R, or K as long as all data (including mass, length, and time) contributing to the G-value constitute a consistent set. Consistency must also exist among all the capacitance and temperature values in the set of active submodels. The fac and tcon arguments on HEADER records and the FAC macro are provided for this. The temperatures must also be consistent with the value ABSZRO. When fluid models are active, the conventions implied by the UID control flag must also be followed.

$$G_{input} = \sigma \cdot \mathcal{F} \cdot A$$

where:

σ .... Stefan–Boltzmann constant (energy/length^2–time–deg^4, e.g., 0.1714E–8 BTU/ft^2–hr–R^4 or 5.667E–8 W/m^2–K^4)

$\mathcal{F}$ .... Gray–body factor

A .... Surface area (length^2)

All conductor input options require that the pair of nodes to which a conductor is attached be specified by their actual node numbers. The user should remember that the negative sign preceding boundary node numbers on node data cards is *not* part of the actual node number. On conductor data cards, negative signs prefixing node numbers are not flagged as errors because they are used to signify one way conductors (see below).

To facilitate the modeling of fluid loops without a fluid submodel, SINDA allows any conductor to be specified as a one–way conductor. One–way conductors permit the modeling of heat transport by fluid (mass) flow, and their conductances are always a function of $\dot{m} \cdot Cp$. Such a conductor is defined by prefixing the node number of one of the adjoining nodes with a minus sign. The node so designated** will not be allowed to lose or gain heat through the conductor, even though the temperature of the node will be used to calculate a heat flow. In other words, the solution subroutines will compute the heat transferred through a one–way conductor as though it were an ordinary conductor. This heat will not, however, be allowed to enter (or leave) the node prefixed by the minus sign; it will be allowed to leave (or enter) the unsigned node.

If both nodes connected by a conductor are in the same submodel, which is also named in the preceding HEADER CONDUCTOR DATA record, just the actual node number (N#) will suffice. Since inter–submodel heat transfer is allowed, the form smn.N# is available when one of the pair is in another submodel. The submodel format for upstream nodes in one way conductors is smn.–N#.

A conductor may also appear with both node numbers identified with submodels outside the conductor's submodel. This allows the user to define one or more heat transfer paths between submodels to be changed or eliminated simply by using the BUILD macroinstruction. The user should keep in mind that the physical constituents of his model are primarily diffusion and arithmetic nodes, secondarily conductors and boundary nodes.

3.10.1   Conductor Data Input

All Conductor data records appear following a header record:

HEADER CONDUCTOR DATA, smn [,fac]

---

*   In addition to the units consistency required for linear conductors, radiation conductors must also be consistent with the SIGMA value specified in CONTROL Data.

**   If fluid flows from node A to node B, then node A (the upstream node) should be flagged with a minus sign to signal a one–way conductor. If the upstream node is outside of the current submodel, the correct designation is smn.–N#.   *NOTE: One–way conductor representations of fluid flow is completely unrelated to FLUINT and fluid subnetworks.*

where:

smn   ....   a submodel name

fac   ....   conductance multiplying factor

[ ]   ....   denotes optional argument

The following general format rules apply for all conductor input options:

1.   Linear conductors must have a positive conductor number.

2.   Radiation conductors must have a negative conductor number.

3.   Multiple conductors, or groups of same-valued conductors, may be defined on a single input record using the Standard (3 blanks) option code. All other option codes appear only once per record, together with their data values.

4.   Conductor identification numbers may consist of up to 6 digits (e.g., 999999).

Table 3-8 summarizes the available conductor data input options. Like all other model parameters, conductance values may be referenced and changed in the logic blocks. Hence, a "constant" value input in conductor data need not remain constant during the entire problem solution.

Standard Option (3 blanks card code) – The simplest conductor data card utilizes the 3 blank option and is formatted as follows:

```
G#,NA,NB,G                      $(basic format)
-4,7,9,3.5E-13                  $(example 1)
18,31,POD2.8,1.8                $(example 2)
```

where:

G#   .....   Actual conductor number (integer)

NA,NB   ..   Actual node numbers of the nodes to which this conductor is connected (integer data value or smn.integer)

G   ......   Conductance (floating point data value)

Example 1 defines conductor number 4 as a radiation conductor connected between nodes 7 and 9 and having a conductance value of 3.5x10-13. Example 2 defines conductor number 18 as a linear conductor connected between nodes 31 and node 8 of submodel POD2, having a conductance value of 1.8.

Since a conductor has a single attribute, conductance, regardless of its type, a single mathematical conductor may be used to represent any number of physical conductors. In this way, any number of conductors having the same conductance may be assigned to the same conductor number, as follows:

Table 3-8: Summary of CONDUCTOR Data Input Options

OPTION (CODE)	DESCRIPTION
3 blanks	To input single conductors
CAL[†]	To input single conductors where G is a strict function of 4 factors
GEN	To generate several conductors with the same conductance
SIV SPV	To input single temperature-varying conductors. SIV uses array interpolation, SPV uses a polynomial function of temperature
SIT SPM	Same as SIV and SPV for groups of conductors
DIV DPV	To input single conductors that model two materials with different temperature-varying conductances. DIV uses two arrays of G vs. T, and DPV uses polynomial functions of T.
DIM DPM	Same as DIV and DPV for groups of conductors
BIV	To input a single conductor that is both time- and temperature-varying. The conductance is found by interpolation using a bivariate array.
PIV PIM	To input conductors where the conductance is a function of temperature and a time-independent variable. PIV is for single conductors, PIM is for multiple conductors
TVS	To input a time-varying conductor
PER	To input a cyclic time-varying conductor

[†] Beginning with SINDA '85, floating point arithmetic (+,-,*,/) in data fields makes the CAL option obsolete. The CAL option is available but not documented.

```
G#,NA1,NB1,NA2,NB2,...  ,NAx,NBx,G $(alt format)
11,2,3,4,7,3.8                     $(example 3)
14,7,6,-9,8,2.1                    $(example 4)
```

where NAi, NBi $(i = 1,2,...,x)$ are node numbers of adjoining node pairs (integer or smn.integer)

Example 3 defines linear conductor number 11 as having a value of 3.8; it is connected between nodes 2 and 3, and, also, between nodes 4 and 7. Even though the conductor number for these two conductors is the same, no physical connection between node (2 or 3) and node (4 or 7) has been explicitly or implicitly established. Example 4 defines a linear conductor, having a value of 2.1, connected between nodes 7 and 6, and connected as a one-way conductor between nodes 9 and 8. In the latter case, heat will flow from and to node 8, but will be prevented from entering or leaving node 9 (through this conduction path).

The inclusion of more than a single (NA, NB) node pair for a given conductor is allowed only with the standard (3 blanks) conductor data input option. Since the total number of data values required to define a conductor (using the alternate format) is variable, the general requirement that each conductor be defined on a single card is relaxed to allow the definition to be split across several cards. In this manner, an unlimited number of like-valued conductors may be modeled by one conductor.

GEN Option – The GEN option is used to generate and input a group of conductors each having the same conductance value. This option requires eight data values for each group of similar conductors, as follows:

```
GEN G#,#G,IG,NA,INA,NB,INB,G      $(basic format)
GEN -5,3,1,10,1,20,1,4.8E-12      $(example 7)
GEN 30,3,0,-40,1,50,-1,2.6        $(example 8)
```

where:

G#	.....	Actual number of the first conductor to be generated (integer)
#G	.....	Total number of conductors to be generated (integer)
IG	.....	Increment to be added to the current conductor number to form the conductor number of the next conductor (integer)
NA,NB	..	Actual node numbers of the pair of nodes connected by the first conductor (integer)
INA,INB	.	Increments to be added to NA and NB, respectively, to arrive at the node numbers for the pair of nodes connected by the next conductor (integer)
G	......	Conductance of all conductors (floating point)

A negative sign preceding the G# causes all conductors to be defined as radiation conductors. The negative sign, however, is never part of the conductor number in the arithmetic sense. Therefore, the conductor numbers generated in example 7 would be 5, 6 and 7 (not -5, -4 and -3), and all three would be radiation conductors. Similarly, a negative sign preceding NA or NB would define the group of conductors as being the one-way type. This negative sign, too, is never part of the node number in the arithmetic sense. Therefore, the NA, NB node pairs generated in example 8 would be (40, 50), (41, 49), and (42, 48). The increment values, IG, INA, and INB, may assume an integer value (positive, negative, or zero)*. Example 7 is equivalent to the following cards prepared under the standard (3-blanks) option:

```
-5,10,20,4.8E-12
-6,11,21,4.8E-12
-7,12,22,4.8E-12
```

Example 8 is equivalent to the following record, which uses the alternate form of the standard (3 blanks) option:

```
30,-40,50,-41,49,-42,48,2.6
```

A "fan" of 5 like-valued conductors may be generated as follows:

---

* When IG is zero, a single conductor will be connected between each generated NA,NB pair. The standard option, which permits several such pairs to be specified for a single conductor, is another way to create a multiply-defined conductor.

GEN 51,5,1,22,0,31,10,6.12*.025/16.2 $Example 9

The zero entry for INA generates node pairs as follows: (22,31), (22,41), (22,51), (22,61), (22,71). Note the arithmetic in the G value field.*

Automated Variable Conductance Options – As stated earlier, the network solution routines call for the execution of the operations in the VARIABLES 1 block prior to performing each iteration on the heat transfer equations. Using this feature, the user could get a temperature-dependent conductance by specifying VARIABLES 1 operations which would insert a new conductance value in the conductance table. This new conductance could be calculated using any convenient algorithm. This is probably the most efficient way, from a computer time standpoint, for this operation but it tends to be rather cumbersome.

The automated options cause the current value of variable conductors to be calculated within the network solution routines at the end of the VARIABLES 1 operations, and therefore relieve the user of the task of preparing and inputting conductance computation algorithms. As specified by the user, one of two standard methods is used to calculate the current value of a variable conductor: (1) interpolation, and (2) polynomial evaluation. For interpolation, the user must input, in the ARRAY DATA block, a bivariate array of (temperature, conductance) ordered pairs representing points on the curve of T vs. G. The conductance is then calculated by performing linear interpolation on this array using temperature as the independent variable. For polynomial evaluation, the user must input an array of coefficients (P0,P1,P2...Pn) in the ARRAY DATA block. The conductance would then be calculated by evaluating the nth order polynomial in T (i.e., P0+P1*T+P2*T**2+...+Pn*T**n). To provide further versatility, the value of the temperature, T, used as the independent variable in both of the above methods, may be taken as the average of the temperatures of the two nodes to which the conductor is connected, or it may be set equal to the temperature of only one of the nodes. This feature is explained along with the options to which it applies.

To enable the same array of points or coefficients to be used for all conductors passing through the same type of material, all options cause the computed conductance to be multiplied by a factor before being inserted in memory. In this way, for example, the array can be used to compute conductivity, k, and the multiplying factor can be used for A/L in the case of linear conductors or, emissivity, ε, and A, respectively, in the case of radiation conductors. In another application, the multiplying factor could be used to scale the conductance to some set of consistent units.

All of the automated options require a multiplying factor which may be input as a floating point data value, or as a user constant of the form: Kn or smn.Kn, where n = actual constant number. If the latter form is used, the referenced constant must be a floating point value. Note that the user may access and change this constant, and thus the conductor value algorithm, at any point in the solution process.

---

* The eleven-data-value GEN option has been made obsolete by the Fortran arithmetic feature of SINDA '85. It is available but not documented.

105

The interpolation options (except BIV[*]) require a doublet array of temperature vs. conductance. (Conductance is used loosely in this context. The array could contain, for example, values of temperature vs. conductivity, as long as the associated multiplying factor had units of area/length.) The array must contain an even number of floating point data values, as follows:

T1,G1,T2,G2,...,Tn,Gn

The temperature values, Tj, must be strictly increasing with j. If the value of the effective temperature (average or single node) at some time is less than T1, then G1 will be used in lieu of performing an extrapolation; if the temperature is greater than Tn, then Gn will be used.

The polynomial options require that a singlet array of coefficients be input in the ARRAY DATA block. The array must consist of n+1 floating point values, where n is the order of the polynomial. The coefficients should be entered in the array in the order of increasing powers of temperature (e.g., the first data value is the constant term, P0, and the (n+1)th data value is the coefficient, Pn, of $T^{**}n$).

For both classes of options, the arrays are referenced by using the form: An or smn.An, where n is the array number.

SIV and SPV Options – These options allow the user to define single conductors having temperature varying conductance. The SIV option assumes the referenced array contains points of T vs. G and is to be used for interpolation, whereas the SPV option assumes that the array contains polynomial coefficients. Both options require five data values for each conductor as follows:

```
SIV G#,NA,NB,AP,F              $(basic format)
SPV G#,NA,NB,AC,F              $(basic format)
SIV -12,24,-25,A(3),-0.0001    $(example 10)
SPV 4,5,6,A2,K11               $(example 11)
SIV 4,5,6,A3,0.0001            $(example 12)
```

where:

G#   . . . . . Actual conductor number (integer)

NA,NB . . Actual node numbers of the pair of nodes to which the conductor is connected (integer or smn.integer)

AP,AC . . A reference to an array of points, or coefficients, respectively, of the form: An or smn.An, where: n is the array number

F   . . . . . . Multiplying factor, input as a positive floating point data value, or a user constant reference of the form: Km or smn.Km, m is the actual number or a user constant (arithmetic expression not allowed). The constant so referenced must have been input as a floating point data value in the USER DATA block.

---

[*]   The BIV option requires a bivariate array of time *and* temperature vs. conductance.

As for all other options, this type of conductor may be specified as a one-way conductor by prefixing NA or NB with a minus sign.

Normally, the average temperature of the two adjoining nodes (i.e., (TNA + TNB)/2) will be used as the independent value for interpolation or polynomial evaluation. If the user wishes to use the temperature of one of the nodes only then this node is identified by placing its number as the NA data value (not the NB value), and the single temperature option is indicated by prefixing the multiplying factor or user constant with a minus sign.

Since the negative serves only as a flag, the multiplying factor will always be interpreted as a positive quantity. Example 10 illustrates the correct input for a radiation conductor connected between nodes 24 and 25 but conducting no heat to or from node 25, wherein the temperature of node 24 will be used as the independent variable for interpolating on the T vs. G points supplied in array number 3, with the result of the interpolation being multiplied by 0.0001 (NOT -0.00011) prior to insertion into memory. The above example, while unrealistic, encompasses all three methods for using minus signs as flags to establish conductor definition and evaluation options.

Examples 11 and 12 are discussed in detail in the following paragraphs.

Consider the thermal network shown in Figure 3-6. The conductance of the conductor is defined by the following equation:

$$G = (0.0001) \times (0.005 \times T^2 + 1.0)$$

where T is the average temperature of adjoining nodes.

Example 11 illustrates the correct method of defining this conductor using the SPV option. The required coefficients of the polynomial would be entered in the ARRAY DATA block with the following record:

```
2=1.0,0.0,0.005
```

The required multiplying factor would be entered in the USER DATA block with the following record:
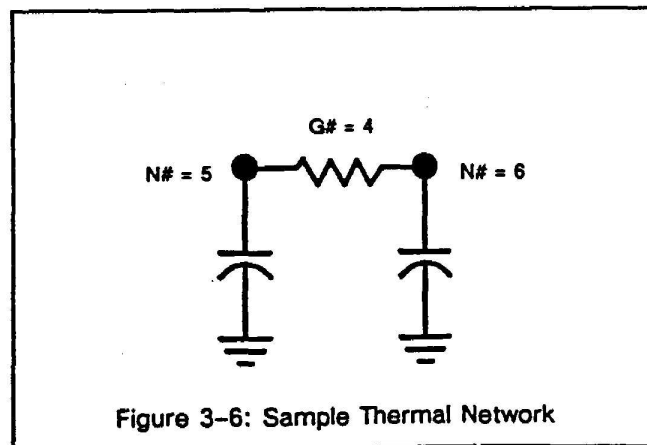
Figure 3-6: Sample Thermal Network

107

```
11=0.0001
```

An alternate method of defining this conductor is illustrated in Example 12. This example uses the SIV option and thus requires an array of T vs. G points for interpolation. The array is developed by approximating the T vs. G curve with a series of straight lines as shown in Figure 3-7. The points which define this approximation are entered in the ARRAY DATA block as follows:

```
3=0.0,1.0,10.0,1.5,20.0,3.0,40.0,9.0
```

The multiplying factor is supplied directly on the conductor data card as the floating point data value, 0.0001, although a user constant reference, such as K11, could have been used instead.

SIM and SPM Options – These two options combine the features of the SIV and SPV options with those of the GEN option. That is, they allow the user to generate and input a group of conductors all having the same temperature varying conductance. The SIM option generates conductors whose conductance will be evaluated by interpolation, and the SPM option generates conductors whose capacitance will be evaluated from a polynomial. Nine data values are required to define each group of conductors as follows:

```
SIM G#,#G,IG,NA,INA,NB,INB,AP,F  $(basic format)
SPM G#,#G,IG,NA,INA,NB,INB,AC,F  $(basic format)
SIM 10,3,1,5,1,6,1,A3,K11         $(example 13)
```
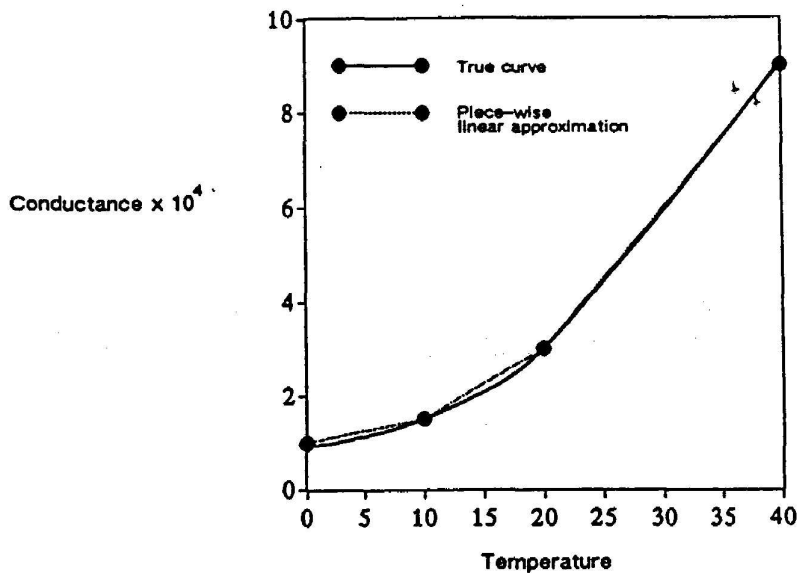

Figure 3-7: Curve of Temperature-Varying Conductance

108

where:

G#   . . . . . Actual conductor number of first conductor to be generated (integer)

#G   . . . . . Total number of conductors to be generated (integer)

IG   . . . . . Increment to be added to the current conductor number to form the conductor number of the next conductor to be generated (non-zero integer)

NA,NB   . . Actual node numbers of the pair of nodes to which the first conductor is connected (integer or smn.integer)

INA,INB   Increments to be added to NA and NB, respectively, to form the node numbers for the pair of nodes connected by the next conductor (integer)

AP,AC   . . A reference to an array of T vs. G points, or polynomial coefficients, respectively, of the form: An or smn.An where n is the array number

F   . . . . . . Multiplying factor input as a floating point data value, or a user constant reference of the form: Km or smn.Km, where m is the constant number of a floating point constant

The conductors defined by Example 13 are equivalent to those defined by the following:

```
SIV 10,5,6,A3,K11
SIV 11,6,7,A3,K11
SIV 12,7,8,A3,K11
```

As for all other options, a negative sign prefixing the G# establishes the entire group as radiation conductors; a negative sign prefixing NA or NB defines a group of one-way conductors: a negative sign preceding F (where F is input as a data value, not a user constant reference) establishes the temperature of node NA as the value of the independent variable to be used for interpolation or polynomial evaluation. Negative signs preceding G#, NA, or NB serve as flags to establish the type of conductors, in accordance with the usual conventions. Negative signs preceding IG, INA, or INB, however, indicate that the data value is, indeed, a negative arithmetic quantity. IG must be non-zero because multiply-defined conductors are permitted only with the standard (3 blanks) and GEN options.

DIV and DPV Options – These options allow the user to define a conductor which passes through two materials, each having a different temperature varying conductance. The DIV option uses the interpolation method to evaluate the conductance, whereas the DPM option uses a polynomial. Both options permit one material to have a constant conductance. Both options require seven data values for each conductor as follows:

```
DIV G#,NA,NB,APA,FA,APB,FB      $(basic format)
DPV G#,NA,NB,ACA,FA,ACB,FB      $(basic format)
DPV 10,20,30,A8,K9,A14,0.03     $(example 14)
```

where:

G# . . . . . Actual conductor number (integer)

NA,NB . . Actual number of the pair of nodes to which the conductor is connected (integer or smn.integer)

APA,APB References to arrays of T vs. G points of the form: An or smn.An, where n is the array number

ACA,ACB References to arrays of polynomial coefficients, of the form: An or smn.An

FA,FB . . Multiplying factors input as floating point data values or as user constant references of the form: Km or smn.Km, where m is the user constant number

Using interpolation (or polynomial evaluation) as the calculation scheme, the first conductance, GA, is computed by using the temperature of node NA as the independent variable for operating on array APA (or ACA) with the result multiplied by FA. In similar fashion, the temperature of node NB, array APB (or ACB) and factor FB are used to compute the second conductance, GB. These two conductances are then combined to form the total conductance, GT, which is entered in memory. The combination algorithm depends on the type of the conductor, as follows:

Linear (series conduction):
$$GT = \left(GA^{-1} + GB^{-1} \right)^{-1}$$

Radiation*:
$$GT = GA \cdot GB$$

If one of the two materials does not have a temperature varying conductance, this material may be identified by replacing its corresponding array reference with a floating point data value. The constant conductance of this material will then be the product of this data value or constant and its associated multiplying factor. This conductance is combined with the variable conductance to form GT, as described above. For completeness, all possible alternate formats are listed, as follows:

```
DIV  G#,NA,NB,SUB,FA,APB,FB      $(GA = SUB*FA)
DPV  G#,NA,NB,SUB,FA,ACB,FB      $(GA = SUB*FA)
DIV  G#,NA,NB,APA,FA,SUB,FB      $(GB = SUB*FB)
DPV  G#,NA,NB,ACA,FA,SUB,FB      $(GB = SUB*FB)
DPV  80,90,91,4.0,3.0,A6,K14     $(example 15)
```

---

* This option should be used with caution to avoid the calculation of an effective radiation conductance of:
$$GT = \sigma^2 \mathfrak{F}_1 \mathfrak{F}_2 A_1 A_2$$

where SUB is a floating point data value

In example 14, the temperature of node 20 will be used with array 8 and constant 9 to compute GA, and the temperature of node 30 will be used with array 14 and the value 0.03 to compute GB. The total conductance, GT, will be entered for conductor number 10. In example 15, the value of GA will be computed as 12.0 (= 4.0*3.0), and the value of GB will be determined from T(91), A6, and K14.

DIM and DPM Options – These two options combine the features of the DIV and DPV options with those of the GEN option. They are used to generate and input a group of conductors all made of the same two materials having different temperature varying conductances. The DIM option causes the conductances for the two materials to be evaluated by interpolating on the referenced arrays, and the DPM option computes the conductances from polynomial coefficients supplied in the arrays. As for the other variables conductance options, multiplying factors are applied to the results of the operations on the arrays. Eleven data values are required to define each group of conductors, as follows

```
C (basic format)
     DIM G#,#G,IG,NA,INA,NB,INB,APA,FA,APB,FB
C (basic format)
     DPM G#,#G,IG,NA,INA,NB,INB,ACA,FA,ACB,FB
C (example 16)
     DIM 10,3,2,20,1,30,-1,A12,K3,A13,K4
```

where:

G#	Actual conductor number of the first conductor to be generated (integer)
#G	Total number of conductors to be generated (integer)
IG	Increment to be added to the current conductor number to form the conductor number of the next conductor (non–zero integer)
NA,NB ..	Actual node numbers of the pair of nodes to which the first conductor is connected (integer or smn.integer)
INA,INB	Increments to be added to NA and NB, respectively, to form the NA, NB pair associated with the next conductor (integer)
APA,APB	References to arrays of T vs. G points, of the form: An or smn.An, where n is the number of the array being referenced.
ACA,ACB	References to arrays of polynomial coefficients, of the form: An or smn.An
FA,FB ..	Multiplying factors input as floating point data values, or as user constant references of the form: Km or smn.Km, where m is the actual constant number of a floating point value

Negative signs preceding G#, NA, or NB serve as flags to establish the type of the conductors, in accordance with the usual conventions. Negative signs preceding IG, INA, or INB, however, indicate that the data value is, indeed, a negative arithmetic quantity. Since the conductance of each conductor is dependent on the specific nodes to which it is connected, the IG value may not be zero (i.e., multiply–defined conductors are not allowed).

111

Since the DIM and DPM options are equivalent, in effect, to a series of conductors defined using the DIV or DPV options, the rules for evaluating the conductance of each conductor generated under the former pair of options are the same as those described in the section covering the latter pair of options.

For either option, one of the two materials may be defined as having a constant conductance by replacing the array reference for the material with a floating point data value. This feature makes the following alternate formats acceptable.

```
DIM  G#,#G,IG,NA,INA,NB,INB,SUB,FA,APB,FB
DIM  G#,#G,IG,NA,INA,NB,INB,APA,FA,SUB,FB
DPM  G#,#G,IG,NA,INA,NB,INB,SUB,FA,ACB,FB
DPM  G#,#G,IG,NA,INA,NB,INB,ACA,FA,SUB,FB
```

where SUB is a floating point data value

The conductors defined in example 16 are equivalent to those which are defined by the following:

```
DIV  10,20,30,A12,K3,A13,K4
DIV  12,21,29,A12,K3,A13,K4
DIV  14,22,28,A12,K3,A13,K4
```

BIV Option – The BIV option enables the user to define a conductor having a conductance which is a function of both time and temperature. The card format for such a conductor is similar to that for the SIV option and requires five data values for each conductor, as follows:

```
BIV  G#,NA,NB,APB,F              $(basic format)
BIV  19,8,11,A43, K28           $(example 17)
```

where:

G# .... Actual conductor number (integer)

NA,NB ..... Actual node numbers of the pair of nodes to which this conductor is connected (integer)

APB .... Reference to a bivariate array (of temperature and time vs. conductance points) of the form: An or smn.An, where n is the array number

F .... Multiplying factor input as a floating point data value or a user constant reference of the form Km or smn.Km, where m is the constant number of a floating point user constant

The temperature value used is the average value of the temperatures of the two nodes (NA, NB) to which the conductor is connected. A negative value of F is not permitted in this option. The time value used is the mean time for the computational interval (control constant TIMEM).

Section 3.7.2 describes the structure of a bivariate array. In this application, the X independent variable should be temperature, the Y independent variable should be time, and the Z dependent variable should be conductance.

Example 17 defines conductor 19, connected between nodes 8 and 11. Array 43 has a bivariate structure, and constant 28 contains a floating point value.

PIV and PIM Options – The PIV option enables the user to define a conductor having a conductance which is a function of both an arbitrary time dependent variable and temperature. The PIM option combines the features of the PIV option with those of the GEN option.

The arbitrary variable is first interpolated from the AT array as a function of time. This interpolated variable along with temperature are used as the independent variables entering into the bivariate array AAT to interpolate the conductance.

A common usage of these functions is interpolating the conductance of an insulation material where the conductance is a function of temperature and pressure. The pressure versus time history is entered into array AT and the conductance as a function of temperature and pressure is entered into array AAT.

To specify the PIV or PIM options the following data are entered into the conductor data block.

```
PIV  G#,NA,NB,AT,FT,AAT,FAT
PIV  G#,NA,NB,SUB,FT,AAT,FAT
PIV  G#,NA,NB,AT,FT,SUB,FAT
PIM  G#,#G,IG,NA,INA,NB,INB,AT,FT,AAT,FAT
PIM  G#,#G,IG,NA,INA,NB,INB,SUB,FT,AAT,FAT
PIM  G#,#G,IG,NA,INA,NB,INB,AT,FT,SUB,FAT
PIM  1,3,1,1,1,2,1,A6,K4,A20     $(example 18)
PIV  4,2,5,A6,0.870,A20,K2       $(example 19)
```

where:

G# .... Actual conductor number (non-zero integer)

NA,NB ..... Actual node numbers of the nodes to which the conductor is connected (integer or smn.integer – see Note 3)

#G .... Total number of conductors to be generated (non-zero integer)

IG .... Increment to the conductor number (integer)

INA,INB .... Increments to the node numbers, NA and NB, respectively (integer)

AT .... Reference to doublet array of an arbitrary variable (A) versus time (t). A = f(t)

AAT .... Reference to bivariate array of conductance (G) versus a temperature variable (T) and an arbitrary variable (A). G = f(T,A). The temperature used in calculation is the average of NA and NB. [Note 4].

FT,FAT .... Multiplication factors (floating point data values, or reference to user constants.)

SUB .... Floating point data value (literal) used as a substitute for an array reference. A user constant *cannot* be used.

Note 1: ...... In both the PIM and PIV options when: SUB,FT,AAT,FAT is used, the arbitrary variable (A) is evaluated as A = SUB*FT. When: AT,FT,SUB,FAT is used, the arbitrary variable (A) is evaluated by interpolation. A = F(t); the conductance (G) is evaluated as G = A*FT*SUB*FAT.

Note 2: ...... Conductor type is specified by the sign of the G# as follows:
LINEAR = positive conductor number
RADIATION = negative conductor number

Note 3: ...... The negative signs on node numbers are considered flags and are ignored in the arithmetic sense. Preceding NA or NB with a minus sign will cause the conductor to be treated as a one-way conductor for fluid flow modeling (not related to FLUINT modeling). The *upstream* node should be identified with the minus sign. The minus sign appears in the smn format as smn.-integer.

Note 4: ...... Consistent with the bivariate array input format described in Section 3.7.2 the arbitrary variable (A) corresponds to the Y values and the temperature (T) corresponds to the X values. The units of the *arbitrary variable entered into the bivariate array* must be equal to the product of the arbitrary variable and the multiplication factor (FT) for the time dependent doublet array.

The following examples demonstrate the use of the PIV and PIM options. With reference to the Figure 3-8, it is desired to input the conductors as a function of pressure and temperature.

The following conditions are assumed:

1) A doublet array of pressure (Newtons/cm^2 versus time (sec) has been entered into the ARRAY DATA block as array no. 6.

2) A bivariate array of conductance (BTU/hr-°F) versus pressure (psi) and temperature (°F) has been entered into the ARRAY DATA block as array no. 20.

114

3)   The constant 1.45 which changes Newtons/cm2 to PSI has been entered into a USER DATA block as K4.

Consistent with the above conditions, conductors 1, 2 and 3 can be specified using the PIM option, as shown in Example 18.

The following conditions are assumed for conductor no. 4:

1)   Orthotropic effects reduce the conductance to 60 percent of the table value. Therefore, 0.6 has been entered into the USER DATA block as K2.

2)   The pressure is determined to be 60 percent of the table value. Therefore, the pressure multiplication factors will be 0.6 * 1.45 = 0.870 (the factor 1.45 changes Newtons/cm^2 to psi).

Consistent with the above assumptions, conductor no. 4 can be specified using the PIV option, as shown in Example 19.

TVS and PER Options – These options allow the user to define time-variable conductors. The time dependency is determined by interpolating on two singlet time and conductance arrays which may appear in either ARRAY DATA or TARRAY DATA blocks. There is a restriction that both arrays must appear in the same block.

If a problem involves time-dependent radiation conductors, (e.g., an articulating spacecraft in orbit) a huge volume of array data may be involved. This is the reason for restricting this option to singlet arrays, which require a minimum number of separate data values. Another memory conservation device is the TARRAY DATA block. This approach stores the arrays on a disk file and only brings the data necessary for current interpolation into memory as required.

The TARRAY DATA block is the most memory-conserving, but imposes a processing time penalty because of the data transfer operations from files to core memory. The user should opt for the ARRAY DATA block unless the memory limit is exceeded. Switching from one to the other is a simple matter of changing the header record(s). Probably the most efficient mode, up to a point, is to forego the TVS and PER options, place the appropriate subroutine calls in VARIABLES 0 and use the ARRAY DATA block. This is practical

Figure 3-8: Example PIV and PIM Network
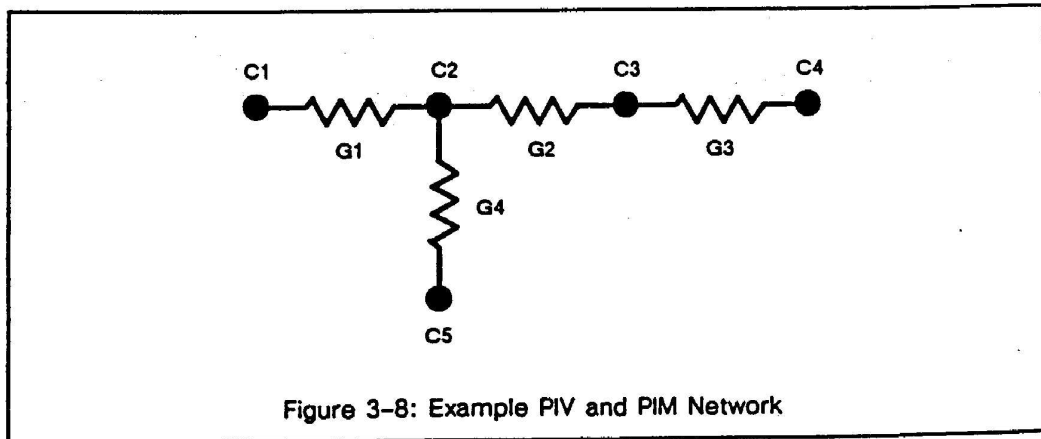
if the subroutine calls can be provided by another computer program, (e.g., TRASYS) but when one or more thousands of subroutine calls appear in VARIABLES 0, Fortran compilation time will definitely be nontrivial. Some compilers will fail on such large routines. The point where this approach compares poorly with TVS & PER must be determined by the user for his particular computer system.

The ARRAY or TARRAY DATA formats required for TVS & PER are:

```
nt = t1,t2,t3 - - - tn
ng = g1,g2,g3 - - - gn
```

where:

nt . . . . . time array number
ti . . . . . time values, strictly increasing with i
ng . . . . . conductance array number
gi . . . . . conductance values (dependent variable) (ti, gi are all floating point data values)

The CONDUCTOR DATA formats are as follows:

```
    TVS G#,NA,NB,At,Ag,F
C (example 20):
    TVS 16,120,DOOR.18,A1,A1412,K12
    PER G#,NA,NB,At,Ag,F,D,P
C (example 21):
    PER 23,131,62,POD.A1,POD.A113,1.0,0.0,1.522
```

where:

G# . . . . . Conductor number, integer

NA , NB . . Actual node numbers to which the conductor is connected, integer or smn.integer

At . . . . . Reference to singlet time array of the form An or smn.An where n is the actual array number

Ag . . . . . Reference to singlet conductance array of the form An or smn.An where n is the actual array number

F . . . . . . Conductance multiplying factor, floating point number or user constant reference, form Km or smn.Km where m is the actual constant number

D . . . . . . Phase displacement as a decimal fraction of the total period, floating point value between 0.0 and 1.0. A user constant reference is not allowed

P . . . . . . Period–specified in time units (floating point data value or reference to a user constant). P is usually the last time point in the At array, but not restricted to this value. If P is less than the last At array value, any values between P and the last array value are not used. If P is greater than the last array time value, the G value used for the time period between the last array time value and P is constant and equal to the G value for the last array time value

Example 20 defines a time–variable conductor no. 16 between node 120 and node 18 of submodel DOOR using time array 1 and conductance array 1412. The conductance value is multiplied by user constant 12.

Example 21 defines a time–variable conductor no. 23 between nodes 131 and 62. The time and conductance arrays are 1 and 113 of submodel POD, respectively. The period is 1.522 hours with no displacement offset.

3.10.2   Conductor Data Updating

The time and temperature dependent conductor data inputs defined in the CONDUCTOR DATA blocks are actually applied (that is, stored in the conductor locations) at the end of VARIABLES 0 and VARIABLES 1 for time and temperature–dependent operations, respectively. The BIV option (time and temperature dependent) is done in VARIABLES 1 prior to each temperature iteration.

These updates are done at the end of the VARIABLES 0 and VARIABLES 1 subroutines using calls to program subroutines GVTIME and GVTEMP. The preprocessor normally inserts these calls subsequent to any user-supplied logic in the VARIABLES 0 or VARIABLES 1 blocks. This gives the user an opportunity to update any user constants or arrays that may affect the conductor data calculations before they are done. If intervention is required after the GVTIME/GVTEMP calls, the user may supply a CALL GVTIME statement anywhere in the VARIABLES 0 input and it will not be overridden by a preprocessor-inserted call. This is also true for GVTEMP in VARIABLES 1.

All conductor value referencing in the logic blocks is done using the forms: Gn o:
smn.Gn where smn is a submodel name and n is the actual conductor number. As usual, the
smn. prefix is required only in the OPERATIONS DATA block and for cross-references to a
node outside the submodel stated in the header card currently in effect. With this referenc-
ing capability, the user may update G-values at will with statements in the logic blocks.

APPENDIX D: HAND SKETCHES AND DATA SUMMARY

Fig(1) TANK AND SURROUNDING GROUND Node GRID NETWORK AND NUMBERING SYSTEM
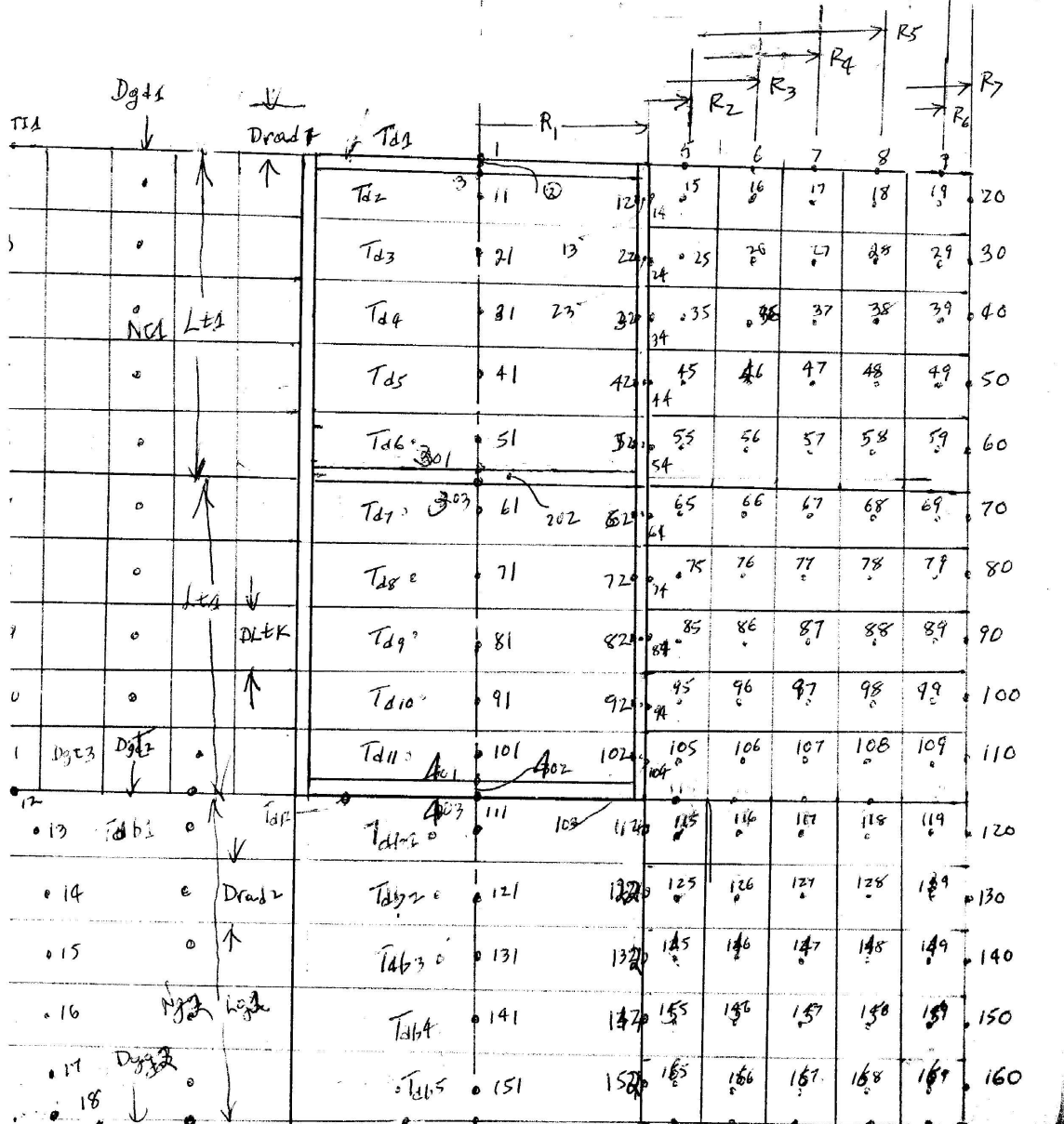
Fig(2) Ground Node Connections above Tank Top

Ground Surface

					C9 185	C10 186	C11 187	C12 188	C13 189	C14
Dag1	Tau1	Ca1	181							
Dag2	Tau2	Ca2	191	192	195	196	197	198	199	200
3	Tau3	Ca3	201	202	205	206	207	208	209	210
4	Tau4	Ca4	211	212	215	216	217	218	219	220
5	Tau5	Ca5	221	222	225	226	227	228	229	230
6	Tau6	Ca6	231	232	235	236	237	238	239	240
Dag7	Tau7	Ca7			5	6	7	8	9	
					R2	R3	R4	R5	R6 R7	

Lg1

2
3

R1
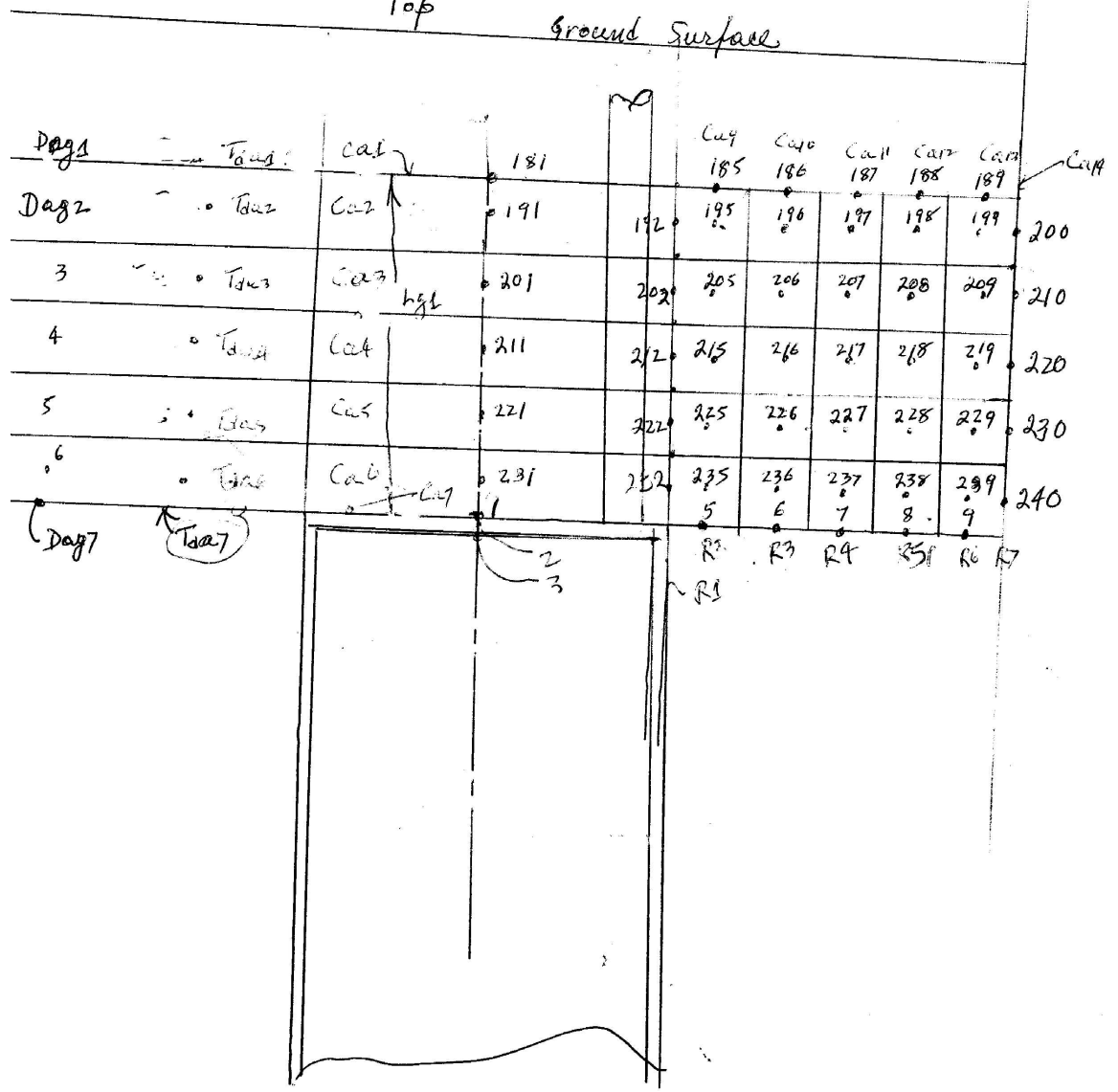
Fig (1) Tank and Surrounding Ground Note
Grid Numbering System.

122
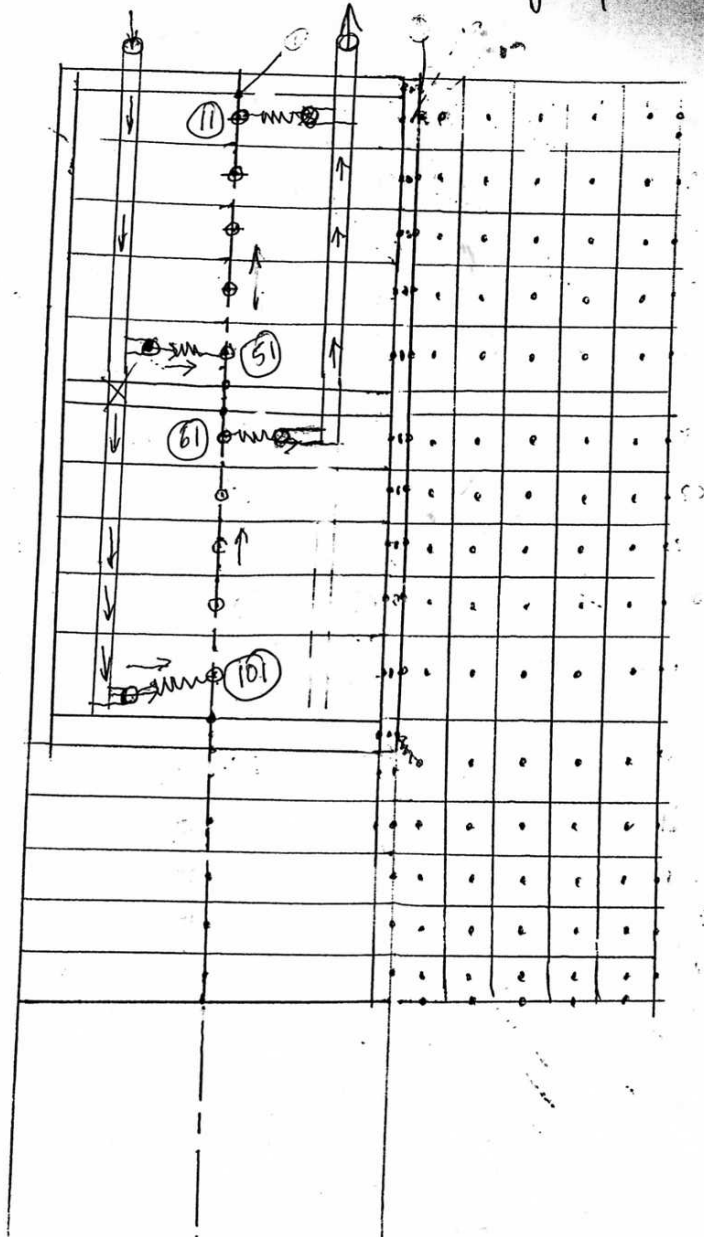
SUMMARY OF DATA

(A) **Glycol Data** at 40°F

$$V_{IS} = 6.05 \text{ Centipoise}$$
$$\rho = 67.61 \, lb/ft^3$$
$$K = .217 \, BTU/hr\text{-}ft\text{-}°F$$
$$C_p = .796 \, BTU/lb°F$$

$$K = .126 \, W/m \, K \times .5777$$
$$\rho = 466 \, Kg/m^3 \times 6.243 \times 10^{-2}$$
$$C_p = 879 \, J/Kg \, K \times 2.388 \times 10^{-4}$$

Dry - Soil
$$\rho = 1500 \, Kg/m^3 \, (93.6450 \, lb/ft^3)$$
$$C_p = 1842 \, J/Kg \, K \quad .4398696 \, BTU/lb$$
$$K = .35 \, W/m \, K \quad .202195 \, BTU/hr \, ft$$

(B) **Earth-Ground Data**

$$C_p = .44 \, BTU/lb°F$$
$$\rho = 128 \, lb/ft^3$$
$$K = .30 \, BTU/hr \, ft^2°F$$

(C) **Tube & Tank Wall Data**

$$T = 30°C$$
$$K = .33 \, W/m \, °C \quad BTU/hr \, ft°F/1.7303 \, W/m°C$$
$$\rho = 960 \, Kg/cm^3 \times lb/ft^3/16.019 \, Kg/m^3$$
$$C_p = 2.1 \, KJ/Kg°C \times 1 BTU/lb°F/4.18669 \, KJ/Kg°C$$

$$\boxed{\begin{aligned} K &= .1907184 \, BTU/hr \, ft°F \\ \rho &= 59.92883 \, lb/ft^3 \\ C_p &= .5015896 \, BTU/lb°F \end{aligned}}$$

Soil
$$T = 300°K$$
$$\rho = 2050 \, Kg/m^3/16.019$$
$$K = .52 \, W/m \, K/1.7303$$
$$C_0 = 1840 \, J/Kg \, K/$$

123

FIGURE A5.4
(*Continued*)

Properties of a 30% (by wt.) solution of inhibited ethylene glycol

Temperature, °F	Viscosity, cP	Density, lb/ft^3	Thermal conductivity, Btu/(h · ft · °F)	Specific heat, Btu/(lb · °F)
20	4.65	65.86	0.249	0.887
40	3.03	65.62	0.257	0.892
60	2.13	65.34	0.264	0.898
80	1.58	65.04	0.270	0.903
100	1.23	64.69	0.276	0.908
120	0.98	64.32	0.281	0.913
140	0.81	63.91	0.286	0.918
160	0.68	63.46	0.290	0.923
180	0.58	62.99	0.293	0.929
200	0.50	62.48	0.296	0.934

Properties of a 50% (by wt.) solution of inhibited ethylene glycol

Temperature, °F	Viscosity, cP	Density, lb/ft^3	Thermal conductivity, Btu/(h · ft · °F)	Specific heat, Btu/(lb · °F)
20	9.87	67.95	0.212	0.786
40	6.05	67.61	0.217	0.796
60	3.99	67.25	0.222	0.806
80	2.80	66.85	0.226	0.816
100	2.06	66.43	0.230	0.826
120	1.58	65.99	0.234	0.836
140	1.25	65.52	0.237	0.846
160	1.02	65.02	0.240	0.856
180	0.85	64.49	0.242	0.865
200	0.72	63.94	0.244	0.875

Properties of a 30% (by wt.) solution of inhibited propylene glycol

Temperature, °F	Viscosity, cP	Density, lb/ft^3	Thermal conductivity, Btu/(h · ft · °F)	Specific heat, Btu/(lb · °F)
20	9.08	64.95	0.242	0.926
40	5.44	64.69	0.249	0.930
60	3.44	64.40	0.255	0.935
80	2.32	64.08	0.261	0.939
100	1.65	63.72	0.266	0.943
120	1.24	63.34	0.270	0.948
140	0.97	62.92	0.274	0.952
160	0.80	62.48	0.278	0.956
180	0.67	62.00	0.281	0.961
200	0.59	61.49	0.283	0.965

Properties of a 50% (by wt.) solution of inhibited propylene glycol

Temperature, °F	Viscosity, cP	Density, lb/ft^3	Thermal conductivity, Btu/(h · ft · °F)	Specific heat, Btu/(lb · °F)
20	26.86	66.35	0.192	0.832
40	13.89	65.95	0.197	0.842
60	7.96	65.53	0.203	0.852
80	4.96	65.09	0.207	0.861
100	3.32	64.63	0.212	0.871
120	2.35	64.15	0.215	0.881
140	1.74	63.65	0.219	0.891
160	1.35	63.12	0.221	0.901
180	1.08	62.57	0.224	0.910
200	0.89	62.00	0.225	0.920

Conversions

Multiply	By	To obtain
cP	0.001	Pa · s
lb/ft^3	16.01	kg/m^3
Btu/(h · ft · °F)	1.73	W/(m · K)
Btu/(lb · °F)	4.19	kJ/(kg · °C)
cP	0.000672	lb$_m$/(ft · s)

823

124

TABLE B–3 THERMOPHYSICAL PROPERTIES FOR NONMETALS

Substance	C_p $\dfrac{\text{Btu}}{\text{lbm °F}}$	ρ lbm/ft^3	t °F	k Btu/hr ft^2 °F	α ft^2/hr
Structural					
Bakelite	0.38	79.3	68	0.134	0.0044
Bricks					
Common	0.20	100	68	0.40	0.02
Face		128	68	0.76	
Carborundum brick			1,110	10.7	
			2,550	6.4	
Chrome brick	0.20	188	392	1.34	0.036
			1,022	1.43	0.038
			1,652	1.15	0.031
Fire clay brick (burnt 2426°F)	0.23	128	932	0.60	0.020
			1,472	0.62	0.021
			2,012	0.63	0.021
Concrete	0.21	119–114	68	0.47–0.81	0.019–0.027
Glass, plate	0.2	169	68	0.44	0.013
Wood, cross grain:					
Oak	0.57	38–30	86	0.096	0.0049
Wood, radial:					
Oak	0.57	38–30	68	0.10–0.12	0.0043–0.0047
Insulating					
Asbestos		29.3	−328	0.043	
			32	0.090	
Asbestos sheet			124	0.096	
Asbestos felt			100	0.045	
(20 laminations			300	0.055	
per in.)			500	0.065	
Cardboard, corrugated				0.037	
Corkboard		10	86	0.025	
Cork, ground		9.4	86	0.025	
Diatomaceous earth (pow-					
dered)		10	200	0.029	
			400	0.038	
			600	0.048	
Fiber insulating board	.	14.8	70	0.028	
Glass wool		1.5	20	0.0217	
			100	0.0313	
			200	0.0435	
Kapok			86	0.020	
Magnesia, 85%		16.9	100	0.039	
			200	0.041	
			300	0.043	
			400	0.046	
Rock wool		4.0	20	0.0150	
			100	0.0224	
			200	0.0317	
Miscellaneous					
Clay	0.21	91.0	68	0.739	0.039
Coal, anthracite	0.30	75–94	68	0.15	0.005–0.006
Cotton	0.31	5	68	0.034	0.075
Earth, coarse	0.44	128	68	0.30	0.0054
Ice	0.46	57	32	1.28	0.048
Rubber, hard		78.4	32	0.087	

Used with permission, from A. J. Chapman, *Heat Transfer*, 3d ed., Macmillan, New York, 1974.

125

Table A-3 I Properties of nonmetals.[†](*Continued*)

Substance	Temperature °C	k W/m·°C	ρ kg/m³	c kJ/kg·°C	$\alpha \times 10^7$ m²/s
Structural and heat-resistant materials					
Plexiglass	30	0.2	1200	1.5	1.1
Polyethylene	30	0.33	960	2.1	1.64
Polypropylene	30	0.16	1150	1.9	0.73
Polystrene	30	0.14	1000	1.3	1.1
Polyvinylchloride	30	0.09	1700	1.1	0.48
Rubber, hard	30	0.15	1200	2.0	0.62
Silicon carbide	30	490	3150	0.68	2290
Stone:					
Granite		1.73–3.98	2640	0.82	8–18
Limestone	100–300	1.26–1.33	2500	0.90	5.6–5.9
Marble		2.07–2.94	2500–2700	0.80	10–13.6
Sandstone	40	1.83	2160–2300	0.71	11.2–11.9
Teflon	30	0.35	2200	1.05	1.5
Titanium dioxide	30	8.4	4150	0.7	29
Wood (across the grain):					
Balsa, 8.8 lb/ft²	30	0.055	140		
Cypress	30	0.097	460		
Fir	23	0.11	420	2.72	0.96
Maple or oak	30	0.166	540	2.4	1.28
Yellow pine	23	0.147	640	2.8	0.82
White pine	30	0.112	430		
Acoustic tile	30	0.06	290	1.3	1.6
Insulating materials					
Asbestos:					
Loosely packed	−45	0.149			
	0	0.154	470–570	0.816	3.3–4
	100	0.161			
Asbestos-cement boards	20	0.74			
Sheets	51	0.166			
Felt, 40 laminations/in	38	0.057			
	150	0.069			
	260	0.083			
20 laminations/in	38	0.078			
	150	0.095			
	260	0.112			
Corrugated, 4 plies/in	38	0.087			
	93	0.100			
	150	0.119			
Asbestos cement	—	2.08			
Balsam wood, 2.2 lb/ft³	32	0.04	35		
Cardboard, corrugated	—	0.064			
Celotex	32	0.048			
Corkboard, 10 lb/ft³	30	0.043	160		
Cork, regranulated	32	0.045	45–120	1.88	2–5.3
Ground	32	0.043	150		
Diamond, Type IIa, insulator	30	2300	3500	0.509	12,900
Diatomaceous earth (Sil-o-cel)	0	0.061	320		
Felt, hair	30	0.036	130–200		
Wool	30	0.052	330		

Handwritten margin notes:

$1 \dfrac{btu}{hr\,ft\,°F} = 1.7307 \dfrac{W}{m\,°C}$

$1 \dfrac{lb}{ft^3} = 16.0185 \times 10^4 \dfrac{kg}{m^3}$

$1\ centipoise = 2.42 \dfrac{lb\cdot m}{hr\cdot ft}$

Geothermal High Density Polyethylene Pipe

Specifications and Data

SCOPE
- Meets overall design requirements for use in closed-loop ground source heat exchangers
- Pipe configured as specified in ASTM D 3035
- Standard specification for Polyethylene (PE) Plastic Pipe (DR-PR) based on controlled outside diameter or ASTM D 2447

PIPE
- Conforms to requirements of ASTM D 3035 or ASTM D 2447
- Loop pipe shall be joined at one end with a factory-installed "U"-Bend with a standard dimension ratio (SDR) of 9
- "U"-Bends will be a single piece, injection molded from PE 3408 HDPE resin with a minimum cell classification of PE345434C
- Working pressure rating of 160 psi water at 73°F for SDR 11 and 200 psi water at 73°F for SDR 9

MATERIAL
- All pipe and heat fused materials are manufactured from virgin high density polyethylene resin exceeding or meeting ASTM D 2513, Sections 4.1 & 4.2
- Hydrostatic Design Basis – 1600 psi at 73°F per ASTM D 2837 and ASTM D 3350
- Listed in PPI-TR4 as PE 3408

MARKING
- Pipe marked with identification, nominal size, trade name, material designation or cell classification, pressure rating at 73°F, ASTM standard, date of manufacture
- Coiled loop pipe shall also have sequential footage marking at two-foot intervals

INSTALLATION
- Installation & backfill shall comply with IGSHPA guidelines, local, state & federal regulations

- When performing heat fusion, IGSHPA's recommended fusion procedures or fusion tool manufacturer's joining procedure for PE 3408 should be utilized

TESTING
- DO NOT TEST WITH AIR OR GAS
- The loop assembly shall be hydrostatically tested with water at a pressure not greater than 150% of the pipe pressure rating

GEOTHERMAL PIPE
- Available in coils 3/4"-4"
- Straight lengths (available up to 16 inch) using PE 3408 resin
- For commercial, institutional, residential or industrial applications

GEOTHERMAL LOOPS
- Custom designed to your job requirements
- Coils are configured with a parallel supply and return pipe heat fused to a patent pending "U"-Bend to make a leak tight joint utilizing PE 3408 resin
- 3/4", 1", & 1-1/4" SDR 11 rated @ 160 psi
- 3/4", 1", & 1-1/4" SDR 9 rated @ 200 psi
- Outside width of "U"-Bend less than 2.50" for 3/4", 3.00" for 1", 3.50" for 1-1/4"

FEATURES AND BENEFITS OF LAMSON VYLON PIPE GEOTHERMAL PRODUCTS
- Fusionable
- Flexible
- Abrasion resistant
- Footage markers on loop piping
- Permanent indent printing
- Made from PE 3408 high density polyethylene resin
- Resistant to most chemical compounds and aggressive soils

High Density Polyethylene IPS Pipe and Pressure Ratings

Iron Pipe Size (IPS) Nom. O.D.	Act. O.D.	265 psi SDR 7 Min. Wall	Wt. Lb./Ft.	200 psi SDR 9 Min. Wall	Wt. Lb./Ft.	160 psi SDR 11 Min. Wall	Wt. Lb./Ft.	130 psi SDR 13.5 Min. Wall	Wt. Lb./Ft.	110 psi SDR 15.5 Min. Wall	Wt. Lb./Ft.	100 psi SDR 17 Min. Wall	Wt. Lb./Ft.	90 psi SDR 19 Min. Wall	Wt. Lb./Ft.	80 psi SDR 21 Min. Wall	Wt. Lb./Ft.	65 psi SDR 26 Min. Wall	Wt. Lb./Ft.	50 psi SDR 32.5 Min. Wall	Wt. Lb./Ft.
1"	1.315	0.188	0.291	0.146	0.234	0.119	0.196	0.097	0.163												
1 1/4"	1.66	0.237	0.463	0.184	0.374	0.150	0.313	0.122	0.259	0.107	0.228										
1 1/2"	1.9	0.271	0.607	0.211	0.489	0.172	0.41	0.140	0.34	0.123	0.299	0.112	0.274								
2"	2.375	0.339	0.948	0.264	0.765	0.216	0.640	0.176	0.531	0.153	0.467	0.140	0.429								
3"	3.500	0.500	2.059	0.389	1.661	0.318	1.390	0.259	1.153	0.226	1.015	0.206	0.931	0.184	0.838	0.167	0.763				
4"	4.500	0.643	3.404	0.500	2.746	0.409	2.297	0.333	1.907	0.290	1.678	0.265	1.539	0.237	1.387	0.214	1.261	0.173	1.028	0.138	0.829
5"	5.563	0.795	5.202	0.618	4.196	0.506	3.511	0.412	2.914	0.359	2.564	0.327	2.352	0.293	2.120	0.265	1.927	0.214	1.571	0.171	1.267
6"	6.625	0.946	7.378	0.736	5.951	0.602	4.980	0.491	4.133	0.427	3.636	0.390	3.336	0.349	3.007	0.315	2.733	0.255	2.228	0.204	1.797
8"	8.625	1.232	12.505	0.958	10.086	0.784	8.440	0.639	7.004	0.556	6.164	0.507	5.654	0.454	5.093	0.411	4.631	0.332	3.777	0.265	3.046
10"	10.750	1.536	19.426	1.194	15.668	0.977	13.111	0.796	10.881	0.694	9.575	0.632	8.783	0.566	7.913	0.512	7.195	0.413	5.867	0.331	4.731
12"	12.750	1.821	27.326	1.417	22.041	1.159	18.443	0.944	15.306	0.823	13.469	0.750	12.355	0.671	11.127	0.607	10.121	0.490	8.253	0.392	6.655
14"	14.000	2.000	32.947	1.556	26.575	1.273	22.237	1.037	18.455	0.903	16.239	0.824	14.896	0.737	13.419	0.667	12.203	0.538	9.951	0.431	8.024
16"	16.000	2.286	43.033	1.778	34.710	1.455	29.044	1.187	24.104	1.032	21.210	0.941	19.457	0.842	17.521	0.762	15.938	0.615	12.997	0.492	10.481

High Density Polyethylene DIPS Pipe and Pressure Ratings

Ductile Iron Pipe Size (DIPS) Nom. O.D.	Act. O.D.	265 psi SDR 7 Min. Wall	Wt. Lb./Ft.	200 psi SDR 9 Min. Wall	Wt. Lb./Ft.	160 psi SDR 11 Min. Wall	Wt. Lb./Ft.	130 psi SDR 13.5 Min. Wall	Wt. Lb./Ft.	110 psi SDR 15.5 Min. Wall	Wt. Lb./Ft.	100 psi SDR 17 Min. Wall	Wt. Lb./Ft.	90 psi SDR 19 Min. Wall	Wt. Lb./Ft.	80 psi SDR 21 Min. Wall	Wt. Lb./Ft.	65 psi SDR 26 Min. Wall	Wt. Lb./Ft.	50 psi SDR 32.5 Min. Wall	Wt. Lb./Ft.
4"	4.800	0.686	3.835	0.533	3.930	0.436	2.588	0.356	2.148	0.310	1.890	0.282	1.734	0.253	1.561	0.229	1.420	0.185	1.158	0.148	0.934
5"	6.900	0.986	7.924	0.767	6.391	0.627	5.348	0.511	4.438	0.445	3.909	0.406	3.583	0.363	3.227	0.329	2.935	0.265	2.393	0.212	1.930
8"	9.050	1.293	13.630	1.006	10.995	0.823	9.200	0.670	7.635	0.584	6.719	0.532	6.163	0.476	5.551	0.431	5.049	0.348	4.117	0.278	3.320
10"	11.100	1.586	20.510	1.233	16.540	1.009	13.840	0.822	11.490	0.716	10.107	0.653	9.272	0.584	8.350	0.529	7.595	0.427	6.193	0.342	4.994
12"	13.200	1.886	29.000	1.467	23.390	1.200	19.570	0.978	16.240	0.852	14.290	0.776	13.110	0.695	11.809	0.629	10.740	0.508	8.758	0.406	7.063
14"	15.300	2.186	38.960	1.700	31.420	1.391	26.300	1.133	21.820	0.987	19.200	0.900	17.620	0.805	15.865	0.729	14.430	0.588	11.767	0.471	9.489

*Iron Pipe Size - Pressures are based on using water at 23°C (73°F) and are determined by using standard formulas for the industry.
NOTE: Service factors should be utilized to compensate for the effect of substances other than water, and for other temperatures.

Density 0.955
Corr. Factor 1.045

Table 1-65. PROPERTIES OF COMMERCIAL PLASTICS *(Continued)*

Properties	Chemical class	Polyethylene	Polyethylene	Polyethylene
	Resin type	Thermoplastic	Thermoplastic	Thermoplastic
	Subclass or modification	Low density	Medium density	High density
ELECTRICAL PROPERTIES				
D.C. resistivity, ohm-cm		$>10^{15}$	$>10^{15}$	$>10^{15}$
Dielectric constant, 60 cps		2.3–2.35	2.3	2.3–2.35
Dielectric constant, 10^6 cps		2.3–2.35	2.3	2.3–2.35
Dissipation factor, 60 cps		<0.0005	<0.0005	<0.0005
Dissipation factor, 10^6 cps		<0.0005	<0.0005	<0.0005
MECHANICAL PROPERTIES				
Modulus of elasticity, 10^3 psi		14–38	35–90	85–160
Tensile strength, psi		1,000–1,400	1,200–3,500	3,100–5,500
Ultimate elongation, %		400–700	50–600	15–100
Yield stress, psi		1,100–1,700	1,500–2,600	2,400–5,000
Yield strain, %		20–40	10–20	5–10
Rockwell hardness				R 30–R 50
Notched Izod impact strength, ft lb/in.		No break	0.5–>16	1.5–20
Specific gravity		0.91–0.925	0.926–0.941	0.941–0.965
THERMAL PROPERTIES				
Burning rate		Very slow	Slow	Slow
Heat distortion, 264 psi, °C				
Specific heat, cal/g		0.55	0.55	0.55
Linear thermal expansion coefficient, 10^{-5}, °C		10–20	14–16	11–13
Maximum continuous service temperature, °C		60–77	71–93	92–200
CHEMICAL RESISTANCE				
Mineral acids, weak		Good	Excellent	Excellent
Mineral acids, strong		Good	Excellent	Excellent
Oxidizing acids, concentrated		Good to poor	Good to poor	Good to poor
Alkalies, weak		Good	Excellent	Excellent
Alkalies, strong		Good	Excellent	Excellent
Alcohols		Excellent to poor	Excellent to poor	Excellent to poor
Ketones		Excellent to poor	Excellent to poor	Excellent to poor
Esters		Excellent to poor	Excellent to poor	Excellent to poor
Hydrocarbons, aliphatic		Fair	Fair	Fair
Hydrocarbons, aromatic		Fair	Good	Fair
Oils: vegetable, animal, mineral		Good	Excellent	Good
MISCELLANEOUS PROPERTIES				
Clarity		Translucent	Translucent	Translucent
Color		Colorless	Colorless	Colorless
Refractive index, n_D		1.50–1.54	1.52–1.54	1.54
Applicable ASTM specifications and test methods		D702, D788, D257, D638, D696, D543, D150, D412, D1248, D542	D257, D150, D412, D256, D696, D543, D638, D785, D1248, D542	D257, D150, D41 D256, D696, D54 D638, D785, D1; D542
FORMS AVAILABLE Cs—castings, F—film, Fb—fibers, I—impregnants, L—laminations, Lq—lacquers, Mf—monofilaments, P—powder, pellet, or granules, R—rods, tubes, or other extruded forms, S—sheets.		F, Mf, P, R, S	F, Mf, P, R, S	F, Fb, Mf, P, R, S
FABRICATION Cl—calendering, Cs—casting, E—extrusion, F—hot forming or drawing, I—impregnation, MB—blow molding, MC—compression molding, MI—injection molding, S—spreading.		Cl, E, F, MB, MC, MI	Cl, E, F, MB, MC, MI	Cl, E, F, MB, MC,