2/3
9-9-75
LA-6001-T

Thesis

UC-25

Reporting Date: June 1975

Issued: July 1975

**The Use of a General-Purpose Heat-Transfer Code
for Casting Simulation**

by

William Carl Erickson

los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87545

An Affirmative Action/Equal Opportunity Employer

UNITED STATES
ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION
CONTRACT W-7405-ENG. 36

11/2/75

This report is derived from a dissertation submitted to the Graduate School of The University of New Mexico, Albuquerque, NM, in partial fulfillment of the requirements for the Degree of Master of Science. The report is the independent work of the author. It has not been edited by the Technical Information staff.

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22151
Price: Printed Copy \$7.60 Microfiche \$2.25

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Energy Research and Development Administration, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.

CONTENTS

I.	Introduction.....	1
II.	Casting Solidification and Simulation.....	2
	Solidification Simulation.....	4
	Heat of Fusion.....	5
	Color Film Output.....	10
III.	CINDA-3G.....	12
IV.	Software Required to Implement Problem.....	17
	PRECIND.....	17
	DIVSUB.....	21
	Film Plotting Routines.....	21
V.	Verification of LQSLTR.....	28
VI.	Experimental Procedure and Results.....	31
	Equipment.....	31
	Uranium Casting K-184.....	35
	Lead Casting K-189.....	39
	Lead Casting K-191.....	43
VII.	Computer Simulation Results.....	47
	Uranium Casting Simulation.....	47
	Lead Casting Simulation.....	51
	Simulation of Steel Cast in Sand.....	66
VIII.	Summary and Conclusions.....	74
IX.	Future Work.....	76
	Experimental.....	76
	Computer Graphics.....	77
	Modification of CINDA-3G.....	79
X.	Use of the Computer.....	82
XI.	References.....	85
	Appendix A. SINDA Subroutine LQSLTR.....	86
	Appendix B. Preprocessor PRECIND.....	88
	Appendix C. Subroutines for Generating Color Plots.....	105
	Appendix D. Model and Data for Verification of LQSLTR.....	124
	Appendix E. Experimental Results.....	126
	Appendix F. Thermal Properties.....	131

Appendix G. CINDA-3G Models and Calculated Results.....	135
Appendix H. Computer-Generated Color Slides.....	147

FIGURES

1. Execution Sequence for CINDA-3G and Associated Software.....	18
2. Flow Diagram for PRECIND.....	19
3. Procedure for Initialization of Film Processing Program.....	22
4. Film Plotting Sequence.....	23
5. Colors Plotted by TMPLOT and Their Respective Codes.....	26
6. Comparison of the Specific Heat and LQSLTR Methods for Handling the Heat of Fusion.....	30
7. Furnace and Instrumentation Used for Experimental Casting Heats.....	32
8. K-Furnace Heating Chamber.....	33
9. K-Furnace Casting Chamber.....	34
10. Initial Mold Temperature Variations for Uranium Casting.....	36
11. Cooling Curves for Uranium Casting K-184.....	38
12. Thermocouple Placement for Casting K-189.....	41
13. Cooling Curves for Lead Casting K-189.....	42
14. Cooling Curves at Axial Center for Lead Casting K-191.....	45
15. Two-dimensional Model for 1.5 in. Diam by 6 in. Long Rod.....	48
16. Calculated Cooling Curves for Uranium Rod K-184.....	50
17. Calculated Temperatures for Lead Casting K-189.....	52
18. Calculated Temperature-Location Plots for Lead Casting.....	54
19. Three-Dimensional Model for Lead Cylinder.....	56
20. Calculated Cooling Curves Using Three-Dimensional Model.....	58
21. Effect of Thermocouple on Calculated Results 0.31 in. from Centerline.....	59
22. Cooling Curves for Lead Casting K-189 Calculated Using One- Dimensional Model.....	62
23. Cooling Curves for Lead Casting K-191 Calculated Using One- Dimensional Model.....	64
24. Two-Dimensional Model for L-Shaped Steel Casting.....	68
25. Effect of Thermal Conductivity of Sand on the Solid Interface Location after 60 Minutes.....	69
26. Temperature Gradient Along Axis of Symmetry for L-Shaped Steel Casting.....	71

27.	Cooling Curve for Steel Casting at the Axial Center/Axis of Symmetry Intersection.....	72
28.	Color Slides.....	150

TABLES

1. Gap Formed Between Lead Casting K-189 and Mold.....	42
2. Color Codes and Their Temperature Ranges for Lead Casting Simulation Plots.....	52
3. Thermal Diffusivities of Materials in Three-Dimensional Model.....	57
4. Thermal Conductance Values.....	63
5. Thermal Conductivity of Sand.....	70
6. Calling Sequence for Calculating Variable Properties.....	81
7. Node Data for One-Dimensional Model.....	124
8. Calculated Temperature of Centerline Nodes.....	125
9. Time-Temperature Data for Run K-184.....	126
10. Chemical Analysis of Lead Used to Calibrate Thermocouples.....	127
11. Thermocouple Calibration Data.....	127
12. Thermocouple Data from Run K-189.....	128
13. Experimental Data from Casting K-191.....	129
14. Thermal Conductivity.....	131
15. Specific Heat and Density as a Function of Temperature.....	133
16. Latent Heat of Fusion.....	134
17. Node Dimensions for Two-Dimensional Rod Model.....	135
18. Calculated Temperature for Uranium Rod K-184.....	137
19. Calculated Data for Lead Casting K-189.....	138
20. Node Dimensions for Three-Dimensional Rod Model.....	139
21. Calculated Data Using Three-Dimensional Model.....	141
22. One-Dimensional Calculations for Lead Cast in a 890S Graphite Mold...	143
23. One-Dimensional Calculations for Lead Cast in a CS-312 Graphite Mold.	144
24. Calculated Temperature Along the Axis of Symmetry of Steel Casting...	146
25. Experimental Data for Steel Casting Along Axis of Symmetry.....	146
26. Calculated and Experimental Data at Centerline and Axis of Symmetry for Steel Castings.....	147
27. Color Codes and Their Maximum Temperature.....	148
28. Key to Slides.....	149

ABSTRACT

The practical use of numerical techniques in simulating casting solidification dictate that a general purpose heat transfer code be used and that results be obtained in an easy-to-analyze format. Color film plotting routines were developed for use with NASA's CINDA-3G heat transfer code; the combination of which meet the above criteria. The subroutine LQSLTR written for SINDA, the successor to CINDA-3G, was verified by comparing calculated results obtained using LQSLTR with those obtained using the specific heat method for handling the heat of fusion. Excellent agreement existed when similar data was used. When the more restrictive requirement of a 1°F melting range was used, comparable results were obtained.

Uranium and lead rod castings were cast in instrumented graphite molds and the solidification sequence simulated using CINDA-3G. Discrepancies attributed to initial assumptions of instantaneous mold filling, uniform melt temperature, and intimate metal/mold contact were encountered. Further calculations using a model incorporating a gap between the mold and casting showed that the intimate contact assumption could not be used; a three-dimensional model also showed that the thermocouple assemblies used with the platinum - platinum-10% rhodium were a significant perturbation to the system.

An L-shaped steel casting was simulated and the results compared to those reported in the literature. The experimental data for this casting were reproduced within the accuracy permitted by the thermal conductivity of the sand, thus demonstrating that agreement can be obtained when the mold material does not act as a chill.

I. INTRODUCTION

As the use of computers expands into the various engineering disciplines, that portion of computer science concerned with computer applications is becoming all the more important. Within the field of metallurgy, the metallurgist/computer-scientist is using the computer for applications varying from process control to theoretical studies. One application of particular interest to foundry metallurgists is the simulation of casting solidification.

Establishing a casting procedure for a new part is often a time consuming and expensive process. Thus, if a tool were available to allow the metallurgist to determine the casting's solidification pattern prior to making an actual casting, significant savings could be realized. For computer simulation to be a useful tool, results must be obtained with a minimum amount of effort. A reasonable way to try this is through the use of a general purpose computer code such as CINDA-3G (Chrysler Improved Numerical Differencing Analyzer for 3rd Generation Computers), developed for NASA by Chrysler Corporation.

The subject of this thesis is the determination of the effectiveness of a general purpose heat transfer code, CINDA-3G, to casting simulation and its use in uranium casting work. Chapter II discusses casting simulation. Handling the heat of fusion is a major problem and this is discussed in detail. The verification of LQSLTR, the SINDA subroutine for handling the heat of fusion, is discussed in Chapter V. The use of color film output in casting simulation is also covered. Chapter III discusses the operation and capabilities of the CINDA-3G computer code; Chapter IV describes the software required to wed the color film output and CINDA-3G; and Chapter VI presents the experimental results which are then compared to the calculated data in Chapter VII.

II. CASTING SOLIDIFICATION AND SIMULATION

The solidification process can be considered from the atomic, microscopic, or macroscopic viewpoints.¹ From the atomic viewpoint, solidification is defined by the motion and energy of individual atoms. This approach is useful to the research metallurgist and solid state physicist in defining fundamental properties of metals and their alloys. The microscopic viewpoint describes the solidification process based on the growth of finite-size particles. Factors such as crystal geometry, microscopic composition gradients in the melt, constitutional supercooling, and the effect of the latent heat of fusion on supercooling are considered. These factors are important in determining such items as casting microstructure and chemical homogeneity and are discussed by Flemings.²

The macroscopic viewpoint is the area which this thesis addresses. This approach deals with fluid flow within the mold and heat transfer from the metal to the mold and surrounding environment. In this context, casting solidification rate is determined by the combined properties of the metal, mold, and casting environment. The basic thermal properties which combine to determine the rate are:

- 1) Thermal conductivity of the liquid metal, the solid metal and the mold material.
- 2) Specific heat of the liquid metal, the solid metal and the mold material.
- 3) Latent heat of fusion of the metal.

While these are the primary properties that control the solidification rate, several other factors can become significant if the proper set of conditions are present. Some of these conditions are listed below.

1) The heat lost from the surface of mold can become significant when the mold cannot be considered an infinite heat sink.

When this condition exists, radiation or convection can become the dominant factor.

2) Heat transfer across the gap formed between the casting and mold after partial solidification of the casting can reduce heat flow.

Such a gap is the result of the volume change encountered in the liquid-to-solid phase transformation and by subsequent differences in the thermal contraction of the metal and mold. Durham and Berry have shown this to be an important factor near chills in chill castings.³ (Chills are high thermal conductivity material placed in the mold to promote early solidification of the metal in that area.)

3) Convection currents within the molten metal will affect the casting's temperature gradients.

In large castings such as multi-ton steel ingots, a significant temperature gradient is formed between the top and bottom of the casting. This results in physical movement of the liquid metal which significantly changes the thermal history of the mold independent of pure heat transfer considerations. Cole discusses these aspects in his paper "Transport Processes and Fluid Flow".⁴

4) Heat loss during mold filling can result in initial temperature variations.

When molten metal strikes the cold mold surface, heat is lost. This results in initial metal temperatures in the mold lower than the pouring temperature and in all probability, a temperature gradient within the melt.

Solidification Simulation

The major heat transfer mode during casting solidification is conduction. Here, the heat conducted in direction x during time t is given by the differential equation

$$\partial Q = -kA(\partial T/\partial x)\partial t$$

where Q = quantity of heat conducted

k = thermal conductivity

x = distance that the heat is conducted

A = area through which heat is conducted

T = temperature

t = time

This equation can be solved using analytical, graphical or numerical techniques. Analytical techniques involve developing an exact mathematical solution to the problem. In all but the simplest problems, the mathematics become so complex that the results cannot be applied to practical problems. Typical assumptions made in analytical solutions are: congruent (fixed temperature) melting, no melt superheating, constant mold/metal interface temperature, and temperature independent thermal properties. The fact that solutions are restricted to problems of no superheating and temperature independent properties make this approach impractical for industrial applications. Furthermore, geometric constraints such as perfect, free standing spheres or cylinders of semi-infinite length are normally applied to make the problem solvable.⁵

Analytical solutions are the only exact answers to the solidification problem; however, they are only exact within the restraints of the simplifying assumptions. Thus the "exact" answer is in fact an approximation when applied to real-world applications. If an approximate answer is to be used, other techniques are available which are easier to use and offer more freedom in defining the problem.

A simple, relatively fast, manual method for graphically determining temperature as a function of location and time was developed by Schmidt and discussed by Ruddie.⁶ While this method is relatively fast, the results are not sufficiently accurate for the initial time steps and the method does not allow for handling the latent heat of fusion or temperature varying thermal properties.

Numerical techniques are often used for solving non-steady-state heat flow problems. The two principal advantages of numerical techniques are versatility and the fact that they lend themselves to solution on digital computers. Versatility is particularly important in solidification problems for it allows the use of temperature varying thermal properties. In addition to increasing the accuracy via more accurate properties, it eliminates the problems associated with superheated melts by simply adjusting the required properties to reflect the phase change encountered.

This work will use finite difference approximation (FDA) methods. Finite difference approximation equations are formed by expanding the appropriate differential equation using Taylor's expansion theorem.⁷ For heat transfer problems, the differential equation of interest is:

$$(\partial T / \partial t) = k(\partial^2 T / \partial x^2)$$

Heat of Fusion

The differential equations and, therefore, the resulting FDA equations assume the function is continuous. This is not true when a phase change occurs. When a phase change is present, special techniques must be applied to account for the latent heat liberated at the transition, i. e., melting, point. While most methods give good results for materials which melt over a temperature range, they produce varying degrees of success for congruent melting materials.

The easiest method used to account for the latent heat of fusion is to combine the latent heat with the specific heat. The specific heat must first

be expressed as a function of temperature. The function is then adjusted over the melting (solidification) range such that the added area under the specific heat-temperature curve is equal to the latent heat of fusion.

The major disadvantage in combining the latent and specific heats in this manner lies in simulating congruent melting materials. Here the melting point must be arbitrarily defined as a temperature range instead of its fixed temperature. The melting range must be selected such that the possibility of the temperature changing from above the liquidus temperature to below the solidus temperature in a single time step has been reduced or eliminated. If such a temperature change should occur, the latent heat of fusion would be ignored for that particular node. Similarly, the melting temperature range must be expanded for alloys which solidify over a small temperature range.

The alternative to expanding the temperature range is to include logic in the computer code that will insure that the latent heat of fusion is considered for each node. To do this would negate the primary advantages of the specific heat method, namely, simplicity and minimal memory requirements.

Pehike, et. al., presented data for two metals using this technique.⁸ The first was silicon brass (82.5Cu-13.5Zn-4Si) which has a solidus temperature of 1510° F and a liquidus temperature of 1683° F; the same temperature range was used for liberating the heat of fusion. The second metal was pure aluminum which melts at 1220° F and was simulated to melt over the temperature range 1197-1242° F ($\Delta=45^\circ$). The experimental vs. calculated time/temperature data all agreed within one percent. The interface location as a function of time also was reported to give excellent agreement, but quantitative data was not presented.

A second method originally suggested by Evers, et. al., is to convert the latent heat of fusion to a equivalent number of degrees by dividing the latent heat of fusion by the specific heat.⁹ A table which contains the number of equivalent degrees for each node is then used to account for the heat of fusion. As the temperature passes through the melting point, the node temperature is corrected according to the following procedure

```

If (equiv deg) > 0 and T < m.p.
    (equiv deg) = (equiv deg) - (m.p. - T)
    T = m.p.
if (equiv deg) < 0, T = T + (equiv deg)
    (equiv deg) = 0.

```

No correction is made if the temperature is greater than the melting point after the calculation for the current time step or if the table value for the node is equal to zero before the current calculation.

The above discussion is for a congruent melting material. For materials which melt over a temperature range, the calculation $T = \text{m.p.}$ is replaced by an algorithm which allows the temperature to decrease in a prescribed manner through the melting range as the heat of fusion is released. The algorithm can be based on a linear drop in temperature over the solidification range or the fraction solid present as determined by the phase diagram for the system being simulated.

This method requires additional computer memory--1 word/node (equivalent degrees) for the congruent melting materials and 2 words/node (equivalent degrees and temperature at $t-\Delta t$) for materials which melt over a temperature range.

A technique similar to this approach is used in SINDA (Systems Improved Numerical Differencing Analyzer).¹⁰ The subroutine LQSLTR written for SINDA (Appendix A) has been incorporated into the CINDA-3G code and is the routine used in this work.

The SINDA approach assumes that the latent heat of fusion is liberated at a fixed rate (BTU/°F). The user defines the heat loss rate for each node taking into account the latent heat of fusion, node volume, and the temperature range over which the material solidifies. A minimum range of one degree must be used.

In use, the new temperature is first calculated ignoring the latent heat

of fusion, but taking into account the specific heat. After these calculations are made, the temperature is adjusted to account for the latent heat. The calculations are based on the node temperature at the previous time step as well as the heat loss rate. By so doing, the simulator can handle solidification, remelting, and resolidification. The code requires two words of storage for each node to implement the algorithm.

The one degree minimum solidification range forces an artificial melting temperature range for congruent melting materials. For theoretical work this may be unacceptable; for engineering applications, it does not pose a problem. The major disadvantage of this technique is the fixed heat loss rate. For congruent melting materials and solid solution alloys, this method is satisfactory. However, for hyper- or hypo-eutectic alloys, calculated temperatures in the liquid + solid range could be in error.

Alloys of the eutectic composition solidify at a fixed temperature while compositions on either side of the eutectic composition will solidify in the same manner as solid solution alloys until the eutectic temperature is reached. When the eutectic temperature is reached, the remainder of the liquid will solidify. Thus, metal alloys belonging to this system will have released only a fraction of their latent heat of fusion when the final solidification temperature is reached. The simulator using the fixed heat loss rate will indicate all the latent heat has been released when the metal arrives at the eutectic temperature. The results for temperatures above the liquidus and below the solidus should not be affected. However, where the temperatures within the liquid + solid range are of importance, logic could be included in the phase transformation routine which would allow the heat loss rate to be changed when the temperature reaches the eutectic temperature.

A more satisfactory means of treating these alloys is combining the fixed heat loss rate and specific heat methods. The portion of the latent heat of fusion which is released above the eutectic temperature could be incorporated into the specific heat. The remainder of the latent heat of fusion would then be handled by the SINDA routine and released over a one degree temperature range bounded on the bottom by the eutectic temperature.

The TRUMP general purpose heat transfer code¹¹ written at Lawrence Radiation Laboratory uses yet another variation of this technique. The general equations solved by TRUMP include internal heat generation, heat resulting from chemical reactions, mass flow effects and surface conduction considerations as well as thermal conductance. Provisions are also available for considering interface conductance such as gap formation between the mold and metal part. When a node is undergoing a phase transformation, $(\partial T/\partial t)$ in the general equation is replaced by $[(-\Delta H/c)(\partial a/\partial t)]$ where c is the node heat capacity and a is the volume fraction of the low temperature phase.

In implementing the above algorithm, the volume fractions are first estimated based on calculated phase transformation and temperature change rates. The new temperatures are computed based on the estimates and then corrected based on the difference between the estimated and actual fraction of the total phase change which has occurred. The TRUMP manual compares its calculated values with analytical results for a hypothetical material cast as an semi-infinite slab. The answers compare favorably.

This approach is limited to congruent melting materials; is more complex than those previously discussed; and requires a minimum of 2 words/node additional storage. The fact that phase transformation rate calculations as well as estimated and final temperature change calculations must be made suggest this technique may be more time consuming than others discussed.

In all of the above techniques, care must be taken in plotting the liquid/solid interface as a function of time. The node temperature is assumed to be the temperature at the center of the node (with the entire node at this temperature) while the heat of fusion is based on the entire volume of the node. Thus, when plotting the interface location for a node which has just completed solidification, the interface is positioned at the leading edge, not the center of the node. It follows that if the precise interface position is desired, a positioning algorithm should be used to calculate the interface position within the node.

Lazaridis approached the problem for congruent melting metals by defining

the velocity the interface was moving through the casting.¹² He showed that the rate the liquid/solid interface advanced was a function of the temperature gradients in the liquid and solid at the interface and the latent heat of fusion. FDA equations were then developed and substituted into the energy conservation equation defining the rate of movement of the interface. The problem was subsequently implemented using explicit FDA techniques. Standard heat transfer equations were used for all nodes except those in which the interface was present and those into which the interface would next move; the developed FDA equations were used for these interface nodes.

This technique gives good results for congruent melting materials, but is not applicable to metals which melt over a temperature range and requires considerable logic within the computer code to implement. While the technique is useful in special-purpose heat transfer codes, the need for special FDA equations at the liquid/solid interface would make it difficult to incorporate into an existing general purpose heat transfer code such as CINDA-3G.

Color Film Output

Ease in analyzing the results is a second requirement for the successful use of computer simulation of casting solidification. Printed output must be pre-edited to keep the user from being overwhelmed by the data. That is, points and times must be selected by the user prior to the computer run being made. The selection is based on previous experience and unexpected anomalous behavior might not be detected.

Microfilm offers the potential of presenting the data in a form more easily analyzed. However, isothermal and isochronal plots also require that the output be pre-edited, i. e., the times and locations must be selected prior to the run. The use of color microfilm or grey-tone black and white enables the entire cross section being simulated to be viewed at fixed times. Plotting equipment for grey-tone microfilm is just becoming generally available. Preliminary work in other simulation fields indicates that achieving the desired results requires very high plotting density and associated large amounts of computer time.

The use of color microfilm has been shown to be a potentially useful tool in casting simulation.¹³ Using this technique, definitive colors are assigned to temperature ranges and the color corresponding to the temperature range in which the point's temperature lies is plotted. This allows the temperature relationship of all points to be presented in a single plot. These plots can be made at preselected times, or time intervals, or after every time step. Using the latter approach and plotting the information on 16-mm color film allows the results to be viewed as a color movie showing the solidification sequence for the entire cross section.

III. CINDA-3G

One of the criteria stated as necessary for computer simulation to be a useful tool in casting simulation was ease of preparing the problem for the computer. One means of accomplishing this is through the use of a general purpose heat transfer code. One such family of codes is the CINDA codes developed by NASA.^{10,14,15} The original version (CINDA) was coded for the IBM-7094 computer in the mid-1960's. The code was later rewritten, with additional features, for the UNIVAC-1108 computer and was called CINDA-3G. It is this version, adapted for the CDC 6600 and 7600 supercomputers, which has been used for this work. A later version, SINDA, has been written and released, but is not available for CDC computers such as those used at LASL.

The CINDA-3G computer code consists of three basic parts: 1) preprocessor, 2) program library, and 3) error package. The CROS operating system provides more useful error diagnostic information than does the error package and was used for this work.

The CINDA-3G preprocessor reads the data prepared by the user and transforms the data to a form acceptable to the library routines. The initial data input identifies the problem using a user-defined title and defines the type of problem to be solved and the storage sequence to be used. Two types of problem can be solved: general (non-network) and thermal (network).

Once the type of problem has been defined, the preprocessor processes the input data, renumbering the nodes and conductors to minimize storage requirements. The assigned node and conductor numbers are then stored according to the type of problem being run and are output to a file (file 12) which is read by the program during execution. The preprocessor also stores all array values and constants in consecutive locations; this information is also written to file 12 for subsequent use. In addition to generating the data required by the differencing routines, the preprocessor also prints the input data and a listing of the input node and conductor numbers with their

associated assigned values. This is necessary since output from the CINDA-3G print routines refers to the assigned relative node numbers, not the user designated numbers.

The preprocessor constructs a main program plus four subroutines which are compiled by the local compilers. The FORTRAN program constructed by the preprocessor is written to file 13 for subsequent reading and compilation.

The routines formed by the preprocessor are *LINKO*, *EXECTN*, *VARBL1*, *VARBL2*, and *OUTCAL*. *LINKO* is the only major routine to which the user cannot input code. *LINKO* calls the appropriate library routines to read the data generated by the preprocessor and initialize all arrays and constants. It then calls subroutine *EXECTN*.

Subroutine *EXECTN* consists of user initiated calls to CINDA-3G library routines as well as FORTRAN statements and calls to user written subroutines. These computations are those required to complete initialization of the system prior to calling the differencing routine selected to solve the problem. If any final output of a nature other than generated by *OUTCAL* is desired after the problem has been completed, the instructions would be included after the call to the differencing routine.

The differencing routine is called by *EXECTN*. While the user cannot input statements to the differencing routine itself, he can input instructions to three subroutines called by it. Prior to the calculations for every time step, *VARBL1* is called. When temperature varying thermal properties are used, the preprocessor inserts calls to library routines into *VARBL1*. These library routines then calculate the current value of the property(s) and stores them for use by the differencing routine. The user may also input instructions into *VARBL1* for items such as storing the temperatures from the previous time step for use by the phase transformation subroutine.

VARBL2 is similar to *VARBL1* except it is called after the calculations for the present time step. Again the user can input logic and instructions such as calling the subroutine to handle the latent heat of fusion.

At the end of each time step, the differencing routine checks to see if output is required at that time. If so, subroutine OUTCAL is called and the required calls and instructions are executed.

User instructions to these four subroutines can take two forms. The first is a command to the preprocessor to generate a call to library or user subroutines. This is the most desirable form since arguments to the instruction as input by the user can refer to user-defined numbering schemes. The preprocessor will then convert the CINDA instruction into a legitimate FORTRAN statement and also supply the correct array subscripts for the arguments. The second method is to input FORTRAN statements. These statements are copied directly into the subroutine without any argument conversion. Thus, these statements can be a major source of error if a node, conductor, or array value is added or deleted. Such a change would result in changes in the relative number. If the associated FORTRAN statements are not corrected, the wrong array values will be referenced.

Three steady-state and five transient thermal analysis differencing routines are available in the CINDA-3G program library. The steady-state routines ignore the effect of specific heat and are not applicable to the solidification problem. Three of the transient thermal analysis routines use explicit techniques and two use implicit methods.

Explicit techniques calculate the temperature of a node at time t in terms of the temperatures of surrounding nodes at time $t-\Delta t$. Thus, the temperature can be calculated directly using algebraic equations. The major disadvantage of explicit methods is that the equations become unstable if the time interval is too long for the spacing used. The maximum time step is related to the node spacing as follows:

$$\Delta t = (m_p C_p \Delta x \Delta y) / k$$

where m = constant

k = thermal conductivity

ρ = density

C_p = specific heat

Δx = space between nodes (x direction)

Δy = space between node centers (y direction)

Implicit methods calculate the temperature at time t in terms of the surrounding node temperatures also at time t . This requires that a set of simultaneous equations be solved for each node and results in more time per iteration than does the explicit method. However, implicit techniques have the advantage that they are unconditionally stable, thus allowing larger time steps and a reduction in the computer time required. Care must be exercised in defining the time step and at least two computer runs must be made to assure that the results are accurate. The first run is made using a time step Δt and the second is made using a time step $\Delta t/2$. If the results do not vary significantly, it can be assumed that the results are correct for the model input. However, if unacceptable variations occur, a third run using $\Delta t/4$ should be made. This process is continued until acceptable results are achieved.

Two implicit techniques are included in CINDA-3G. One is a backward differencing technique and the other is a forward-backward technique. The manual indicates that approximately 25% fewer iterations are required when the forward-backward method is used.

The relatively small steps required for the forward differencing explicit technique can be partially overcome by using two accelerated differencing routines provided by CINDA-3G. While both of the accelerated techniques reduce computational time, they also result in less accurate answers. Thus they are useful in obtaining preliminary results and doing parametric analyses where trends are of interest, but they should not be used for final results where maximum accuracy is desired.

The CINDA codes solve the network for a thermal analog model. In such a model, the thermal properties are represented by their electrical counterpart. Each node has a capacitance (thermal energy) defined as the product of the nodes volume, density, and specific heat. The exceptions to this are

arithmetic nodes and boundary (mathematical boundary) nodes. Arithmetic nodes do not have the ability to store heat, but the temperature is calculated for each time step. The boundary nodes are constant temperature nodes whose temperatures can only be changed by the user.

Each node is connected to one or more nodes by conductors. These conductors can transmit thermal energy by conduction, convection, or radiation. The conductance for conduction nodes is equal to the product of the thermal conductivity and energy cross-sectional flow area divided by the path length. When radiation conductors are present, the Fahrenheit temperature scale must be used. This is dictated by the algorithm used to calculate the nonlinear, radiation transfer.

In addition, each node may have a heat source associated with it. The value of the heat source, whether constant or varying with respect to time, is set in VARBL1 prior to the calculations for each time step.

Each node is defined as a volume of uniform temperature material which can be further characterized by its lumped parameter ($C/\Sigma G$). Care must be exercised in defining the network so that the variation between the minimum and maximum lumped parameter values is minimized. If the ratio of the minimum to maximum values is less than a factor of 10, good results can be expected provided care is taken in defining parameters associated with the differencing routines. If the minimum and maximum lumped parameters do not vary by more than 2 orders of magnitude, the results can be considered to have merit.

IV. SOFTWARE REQUIRED TO IMPLEMENT PROBLEM

Solving the solidification problem using color film output requires the use of special programs and routines in addition to the CINDA-3G program library. Figure 1 lists the execution sequence for this problem. The preprocessor (PRECIND) reads the input file and creates fileset 55. The prime functions of the preprocessor are to 1) generate the information required to associate the relative node numbers as assigned by CINDA-3G with microfilm plotting coordinates and 2) transform SIM and DIM commands to CGS and CGD commands, respectively. The code also strips the comments from the input cards.

The CINDA-3G preprocessor then reads fileset 55, generating the data required by the differencing routines and assembling the program for compilation as described in Chapter III. Because of the large number of subroutine calls generated by CGS and CGD commands, VARBL1 could not be compiled by the available compilers. Thus fileset 13 containing the program to be compiled is read by program DIVSUB and VARBL1 is divided into two subroutines. The entire program is output to fileset 77.

The program existing on fileset 77 is then compiled as are user input subroutines which reside on fileset 55. The object code from the compilation plus the user library routines are loaded and the program is executed.

PRECIND

Figure 2 shows a general flow diagram for PRECIND. This routine has two primary tasks. The first is generating the plotting locations and associated relative node numbers for the film plotting routines; the second is to convert SINDA mnemonic codes SIM and DIM to CINDA-3G codes CGS and CGD. A third, secondary task is to strip the comments from the input cards. This avoids an idiosyncrasy involving location of comments on data input cards which exists within the LASL version of CINDA-3G. The problem manifests itself as an

Figure 1.

Execution Sequence for CINDA-3G and Associated Software

Figure 2.
Flow Diagram for PRECIND

illegal character present in a variable format constructed by the CINDA-3G preprocessor. In all probability it is the result of differences between the UNIVAC 1108/IBM 360 and CDC 7600 word sizes and system routines. A complete listing of PRECIND is contained in Appendix B.

Three choices were considered as a means of relating the nodes and their film plotting coordinates. Where the node spacing is constant and the input order is fixed, a program can be written to automatically generate the film plotting coordinates. However, to place such a restriction on CINDA-3G network formulation would negate much of the versatility present in CINDA-3G.

The second possibility consisted of the film initialization subroutine reading node numbers and coordinate data punched on cards. This approach has two distinct disadvantages. First, preparing the data is time consuming and second, the relative node number is input order dependent. The latter fact makes it necessary to input as many as 4000 cards in an exact order. These disadvantages were significant enough to eliminate the method from further consideration.

The third method involves placing the film coordinates on the same card as the CINDA-3G node data. This can be done using the comment feature of the node input format but it requires that either a preprocessor be written or the CINDA-3G preprocessor be modified; the former approach was chosen.

The fact that a film plot is to be made for a run is indicated by a CINDA-3G remark card with dollar signs in columns 12 and 13. When this card is encountered in the title block, control is shifted from PRECIND to subroutine PRCFILM. The coordinate location of the axis of symmetry is stored. If no axis of symmetry exists, the left boundary location is input as a negative number. PRCFILM then processes all data up to but not including the conductor data. Relative node numbers are assigned in the same manner as the CINDA-3G preprocessor and these numbers plus the node's film coordinates are saved for reading by the film initialization routine. This procedure allows the film locations to be included on the node cards for all runs, but only processed when the REM \$\$ card is included in the title block.

The second major task of PRECIND is performed by PSINDA and allows multiple nodes and conductors with temperature varying properties to be input by one card. CINDA-3G has this capability for fixed thermal properties via the GEN command, but does not have it for variable properties. The SIM and DIM commands provide multiple input capability in SINDA. Therefore, the SINDA format for SIM and DIM has been used. By so doing, the input deck is compatible in this respect with SINDA should this code become available at LASL.

DIVSUB

Because of the large number of subroutine calls generated by the variable property commands in subroutine VARBL1, the subroutine must be divided to allow compilation by the LASL RUN compiler. DIVSUB reads fileset 13 (created by CINDA-3G preprocessor) and copies all records to fileset 77. The first 1700 records of VARBL1 are copied as part of VARBL1. A call to VARBL1A is then inserted and the remainder of the original VARBL1 copied to VARBL1A. The appropriate return, end, common blocks, and dimension statements are also added.

DIVSUB also replaces the program card in LINKO to eliminate creation of files used for restarts, etc., which are features of CINDA-3G not used in this problem. This is done to conserve LCM space. Each fileset is automatically allotted 13,332 words in large core storage as a buffer. By eliminating the filesets from the program card, these buffers are also eliminated and the space made available to the plotting routines.

Film Plotting Routines

A series of initialization and film plotting routines were written (see Appendix C). General flow diagrams are presented in Figures 3 and 4, respectively. The film initialization routines read the data generated by PRECIND and store the relative node number (i. e., subscript to the temperature array) and plotting commands. It then fills in the plotting

Figure 3.

Procedure for Initialization of Film Processing Program

Figure 4.
Film Plotting Sequence

matrix so that a uniform density plot is obtained. If this is not done, visual indications of different shades of a color may in fact be the result of variations in plotting density (points/unit area) and not actual changes in color.

The TEMP matrix referred to below is used for two purposes. When used for film plotting routine initialization, the matrix is initially set to -1. As node data are read, the corresponding locations in TEMP are set to +1. After all the node data have been read, the matrix is used by subsequent routines to assure that a uniform density plot is generated. During determining the temperature and color to be plotted for each point, this matrix is used to store the temperature of the nodes. By so doing, temperatures do not have to be recalculated for a matrix location when that location's temperature is used to calculate the temperature of a second point.

Four classes of plotting temperatures were established. The first is primary-centerline. These temperatures correspond directly to those nodes that lie on the axis of symmetry and whose temperature is calculated by CINDA-3G. For these temperatures, the node subscript, 4020 command, and the TEMP matrix y coordinate are stored in small core memory (SCM) arrays. At this time, the film initialization and plotting routines assume the plotting axis of symmetry is about the y axis.

The second class of temperatures is primary-matrix. Like the primary-centerline, these temperatures refer to plotting locations with corresponding nodes whose temperatures are calculated by CINDA-3G. These temperatures differ from the centerline locations in that two points per node are plotted when axial plotting symmetry exists. Thus for each node, the node subscript, two 4020 commands, and one set of TEMP matrix coordinates are stored in large core memory (LCM). The second set of 4020 commands is stored for all plots, even those which do not possess a plotting axis of symmetry. When a negative axis value is read, a flag is set and COLSTOR will only plot the first point.

The last two classes are calculated-centerline and calculated-matrix

temperatures. These temperatures differ from their primary counterparts in that they do not have a corresponding node whose temperature is calculated by CINDA-3G. These plotting points are required to obtain a uniform plotting density. For this work, axial distances between locations is 5 Raster points. Therefore, temperatures for intermediate points are calculated at 5 Raster point intervals assuming a linear temperature variation between points. The 5 Raster point spacing requires that the plotting coordinates input with the node data are multiples of 5. For centerline nodes, the TEMP coordinates for the points of known temperature, multiplication factor, 4020 commands, and TEMP y coordinate for each point are stored in SCM. For the matrix points, the same information plus TEMP x coordinate and a second 4020 command are stored in LCM.

The film plotting routine (TMPPLOT) is called by the CINDA-3G subroutine OUTCAL. TMPPLOT processes the points in the same order as the points had been initialized. The node temperature for the primary points and the calculated temperature for the calculated locations are stored in TEMP and the correct temperature range for the point is determined. Subroutine COLSTOR is then called passing as arguments the location of the first 4020 command to be stored, the color code for the point, and the number of words to be stored for each strike.

COLSTOR decodes the color code into the proper filter combinations for 1 or 2 strikes. Using double strikes, the code is capable of plotting up to 22 colors. Colors capable of being plotted along with their color codes are shown in Figure 5. The codes used are those devised by R. Elliott of LASL and used in his color routine COLORO.

Once the color code has been decoded, the 4020 command(s) are stored in the proper SCM buffer arrays for that color. (Colors plotted by one strike are the primary colors red, green, and blue and the complementary colors magenta, yellow, and cyan. Because of the green hue to white plotted on the FR-80, white is plotted as a double strike -- red and cyan.) When the buffer for a particular color becomes full, the contents are transferred to LCM.

Figure 5.

Colors Plotted by TMPLOT and Their Respective Codes

After the color of all locations has been determined, the SCM buffers are flushed and the points plotted on film. Colors are plotted in the following order: cyan, blue, magenta, red, yellow, and green. The time caption for the frame is plotted in cyan before the first point is plotted and the mold/casting outline is drawn using the system white after the last point has been plotted.

COMPASS subroutine PLOTIT was written to eliminate one subroutine call per strike. Using only FORTRAN subroutines, it was necessary to call the system routine PLT for each strike. PLT formed the command from the two film coordinate values and the plotting character passed to it as arguments and then called system routine BS4020. By forming the commands in FILMINT, the command for each point is formed only once per run. BS4020 can not be called directly from a FORTRAN subroutine as BS4020 expects the command (argument) to be in register X1 and FORTRAN passes the arguments in core. Therefore, PLOTIT was written in COMPASS to eliminate unnecessary subroutine calls, associated "bookkeeping" performed with these calls, and duplication of command formation.

The 1.5 in. diam rod simulated in this work and discussed later, requires 8800 points per frame. Assuming half these points require two strikes, approximately 13,200 plots per frame are made. A FORTRAN subroutine call requires about 50 clock cycles and it requires about 23 cycles to form a command. This gives approximately 73 cycles which are eliminated for each plot under the COMPASS PLOTIT version. The cycle time for the CDC 7600 computer is 27.5×10^{-9} seconds. In plotting a 2000 frame movie (2 minutes at 16 frames/second), about 54 seconds computing time is saved by the COMPASS version.

V. VERIFICATION OF LQSLTR

One of the prime objectives of this work was the verification of the SINDA liquid-solid transformation subroutine LQSLTR. One means of accomplishing this is comparing the results of simulation runs using this method and another established procedure such as the commonly used method of incorporating the heat of fusion into the specific heat. This method was chosen as a known standard against which to compare LQSLTR.

Lead cast into a graphite mold was used as the verifying system. The model data and computed results are tabulated in Appendix D and the thermal properties are listed in Appendix F. The model used consisted of two rows of 10 nodes each traversing the one-dimensional section from the center to edge of a 1.5 in. diam rod. While a one row model could have been used, two rows decreases the maximum/minimum CSG-value ratio. (The CSG-value for the node having the maximum CSG-value in this model was reduced by ~17% using this procedure.) The graphite wall was 1.5 in. thick and was divided into 10 nodes.

The interface was treated in the following manner. Three arithmetic nodes (nodes having no heat storage capability) were created on the surfaces of the nodes adjacent to the interface (2 lead and 1 graphite nodes). Conductors connecting the lead arithmetic nodes to the graphite arithmetic node were assigned a value equal to the surface area of the lead node. The thermal conductance through these conductors was then calculated using the temperature of the lead surface and temperature varying conductance values. The thermal conductance values below the melting point are typical of those reported by Durham and Berry³ while those above the melting point were chosen to approach intimate contact conditions.

All variables were held constant in the three runs except those affecting the treatment of the heat of fusion, i. e., minimum and maximum melting range temperature and the specific heat of the lead. Run 1 was made using LQSLTR and the minimum 1° F temperature range; Run 2 used LQSLTR and a 50° F melting range; and Run 3 incorporated the heat of fusion into the specific heat using

a 50° F temperature range. The results are summarized in Figure 6.

The results for the two runs made using a 50° F melting range are essentially identical. The heat of fusion is started to be liberated at ~ 22.5 seconds and is completed at ~ 25.2 seconds. The maximum temperature difference at any set time is 7° F (1%) while the average temperature variation is 1.5° F (0.2%). This excellent agreement verifies the correctness and applicability of the SINDA routine LQSLTR.

The data from the run made using a 1° F melting range displaces the curve slightly to the right (longer time). Liberation of the heat of fusion started at 24.8 seconds and was completed at 26.0 seconds. Here the heat of fusion was liberated in 1.2 seconds as compared to 2.7 seconds for the 50° F melting range. This shorter time is realistic.

The problem of truncation errors when the heat of fusion is incorporated into the specific heat is sometimes regarded as a source of error in this method. The excellent agreement between the two 50° F melting range curves shows this is not a factor when reasonable time steps are used as LQSLTR accounts for the exact amount of latent heat released by the node during solidification.

Figure 6.
Comparison of the Specific Heat and LQSLTR Methods for Handling the Heat of Fusion

VI. EXPERIMENTAL PROCEDURE AND RESULTS

Three instrumented castings were made to generate experimental data for comparing with the calculated results of CINDA-3G. Comparison of these runs and calculated results will be discussed in Chapter VII. The following will describe the equipment used and the experimental procedure and results.

Equipment

The castings were cast in a vacuum induction furnace (Figure 7). The furnace was powered by two 15 KW, 10 KHz motor-generator sets. One set was used to melt the charge and the second to preheat the mold for the uranium casting. The furnace pressure can be reduced to $25\text{--}50 \times 10^{-3}$ torr using the roughing pump. Typical pressures when the charge is at temperature range from 75 to 150×10^{-3} torr depending on the charge material, mold and crucible, and charge temperature.

Figure 8 shows the upper chamber of the furnace after completion of a uranium run. The charge is placed in a graphite crucible which also serves as the susceptor. For lead runs, the crucible is uncoated. However, a zirconium silicate slurry coating is used for uranium melts to prevent excessive carbon pickup during the run. The induction coil couples to the susceptor/crucible which in turn heats the charge by radiation and conduction. A graphite stopper rod prevents the molten metal from entering the mold during melting and is mechanically raised to facilitate pouring.

Figure 9 shows the lower furnace chamber after the uranium run. The induction coil used to heat the graphite mold is not used for casting lead.

The instrumentation used for recording time-temperature data from the mold is shown in Figure 7. Seven continuous plotting Brown strip chart recorders of varying chart speed were used. In addition, two Hewlett-Packard X-Y recorders and two 12-channel Brown multipoint recorders were used. The multipoint recorders recorded one temperature every 5 seconds. Thus the

Figure 7.

Furnace and Instrumentation Used for Experimental Casting Heats

Figure 8.

K-Furnace Heating Chamber

Figure 9.

K-Furnace Casting Chamber

temperature for each channel is recorded once every minute. The one minute interval was reduced by shorting channels. This reduced the number of thermocouples which could be read, but also decreased the time interval between readings for a given thermocouple.

For Run K-191, two additional Brown strip chart recorders were used. These units recorded the temperature when the thermocouple reading was changing at less than some fixed rate. This is advantageous when slow heating/cooling rates are encountered in that the temperature will be recorded at rates approaching 1 reading per second. However, several seconds can pass without a temperature being recorded when the temperature change rate is high.

A marking unit was used to give a reference time mark on up to eight continuous plotting recorders. This was accomplished by actuating a relay which shorted all attached thermocouples simultaneously. Relative times could then be determined from this reference point. Unfortunately, such a device is not applicable to multipoint recorders.

Uranium Casting K-184

The intent of this work was to compare CINDA-3G calculated solidification data with experimental data generated for a cast uranium rod 1.5 in. diam x 6.0 in. long with a 2.0 in. diam by 3.12 in. long riser. Run K-184 was made as a preliminary run to check out the equipment and detect potential experimental problems and was combined with the last of a series of runs designed to determine initial axial and radial temperature variations in the graphite mold. As such, only three platinum - platinum-10% rhodium thermocouples were inserted for measuring the metal temperature; these thermocouples were not calibrated.

The initial mold temperature data from run K-184 and previous runs in which the mold was heated, but no metal was poured, is shown in Figure 10. This data was generated using the same mold/induction coil setup and was recorded for a constant reference thermocouple reading of 1382⁰ F.

Figure 10.

Initial Mold Temperature Variations for Uranium Casting

The charge was heated to 2310° F. This is 130° F lower than the temperature at which uranium is normally poured at LASL. Because this was a preliminary run and because of other work using the furnace at that time, the induction heating coil was not changed. As a result, a larger-than-needed coil was used which resulted in poor coupling and 2310° F was the maximum temperature obtainable.

The molten uranium was poured into the zirconium-silicate coated graphite mold. The three Pt-Pt/Rh thermocouples consisted of .016 in. diam thermocouple wire which had been drawn from 0.020 in. diam wire and then annealed. The wires were inserted in nominal 0.125 in. diam alumina two-hole thermocouple spaghetti. The tip was approximately 0.062 in. from the end of the alumina with the tip positioned at the centerline of the rod. The three thermocouple assemblies were inserted radially through the mold wall at axial locations 3.37, 6.00, and 8.91 in., respectively, from the top of the mold.

The results are shown in Figure 11 and tabulated in Appendix F. These curves showed good agreement with respect to each other. The bottom location solidified first with the center and top locations completing solidification at approximately the same time. The low arrest temperature (2048° F) for the bottom location must be attributed to the use of uncalibrated thermocouples. An arrest of this length cannot be caused by undercooling and in all probability should have been 2070° F. The mold temperature as measured 0.12 in. from the inside surface and 3.62 in. from the top was 1418° F at pour. This thermocouple was read once every minute. Approximately 15 seconds after pour, the temperature had increased to 1580° F. After 1 min. 15 sec. the temperature was 1553° F. The results will be further discussed when compared to the calculated results in Chapter VII.

Attempts to obtain data from more heavily instrumented molds with calibrated thermocouples were unsuccessful. This was the result of the platinum thermocouples reacting with the molten uranium when the standard 2435° F pouring temperature was used. Lower pouring temperatures were not used because metallography performed on the thermocouples used in run K-184 revealed that the reaction had started to occur when the melt was poured at

Figure 11.

Cooling Curves for Uranium Casting K-184

2310° F. This reaction explains the anomalous change in slope at 2030° F in the record of the top thermocouple. Similar thermocouple problems were reported by the French Atomic Energy Commission.¹⁶ They were monitoring casting procedures using slow pouring rates and could only use the times at which the arrests occurred, not the arrest temperatures themselves.

Further lowering of the melt temperature would have required raising the mold temperature to obtain a sound casting. This would have necessitated redetermining the initial mold temperature distribution and, at that time, it was felt the same information could be obtained more easily and with less disruption to the work load in the Foundry by pouring lead into a room temperature graphite mold.

Prior to discontinuing the uranium phase of the work, the use of nickel and tantalum thermocouple sheaths was investigated. The thermocouples were not grounded to the sheaths and when used in this condition, the maximum temperature recorded for any thermocouple was below the melting point of uranium. Therefore, no usable data was obtained.

Lead Casting K-189

Run K-189 was the first of two instrumented lead castings. Prior to making this run, it was found that lead and platinum formed a low temperature eutectic at 95% Pb and 545° F. Consequently, bare Pt-Pt/Rh thermocouples could not be used. While the temperature is such that these Pt-base thermocouples did not have to be used, the instrumentation was set up for Pt-Pt/Rh thermocouples and they were used. Metallic and polyimide coatings were evaluated and it was found that 0.0005 to 0.001 in. copper plated on the thermocouples after welding prevent reaction with the lead and did not significantly reduce the thermocouple response time. As a further precaution, an air setting Al_2O_3 base cement was applied to the thermocouple assembly in such a way that only the bead itself was exposed to the molten lead.

The thermocouple/recorder units were calibrated as an entity using high purity lead. The thermal arrest was recorded and compared to the theoretical

melting point. Calibration results are shown in Appendix E.

Eight platinum - platinum-10% rhodium thermocouples were used to measure the metal temperatures. These thermocouples were monitored using continuous plotting recorders. Eight chromel-alumel thermocouples were used in the mold. One thermocouple was read using a continuous plotting recorder and the remaining thermocouples were monitored using a multipoint recorder. Temperatures were recorded once every 30 or 60 seconds depending on location. In addition, a seventeenth thermocouple was used to measure melt temperatures. Thermocouple placements are shown in Figure 12.

The molten metal cools rapidly for all locations as would be expected when pouring into a chill mold (See Figure 13). [Typical permanent (chill) molds are made of ferrous materials. The thermal diffusivity (thermal conductivity/specific heat) of steel is approximately $0.2 \text{ ft}^2/\text{hr}$. This compares to a thermal diffusivity of $1.4 \text{ ft}^2/\text{hr}$ for 890S graphite in the across-grain direction. The thermal diffusivity of a slow cooling sand mold is approximately $0.013 \text{ ft}^2/\text{hr}$.] Solidification is complete within 5 seconds 0.75 in. from the bottom of the rod and 34 seconds 0.87 in. above the top of the rod (into the riser). The tabulated results for these thermocouples and two thermocouples located at 0.5r are given in Appendix E. The remaining three Pt-Pt/Rh thermocouples did not survive the run.

The maximum temperature recorded in the mold was 216°F 1 min. 45 seconds after pour. This temperature was recorded 0.12 in. from the mold/riser interface, 2.25 in. from the top. There did not appear to be a significant radial temperature variation in the mold. The dominant factor in mold temperature variations was the riser. The three mold thermocouples located near the riser recorded comparable readings while the remaining five thermocouples recorded somewhat lower temperatures ($25\text{--}30^{\circ} \text{F}$) 1 minute after pour. The time-temperature curve for the thermocouple located 0.12 in. from the rod/mold interface (5.25 in. from the top of the mold) is shown in Figure 13. This curve is representative of the lower temperature curves.

Lead undergoes a relatively large shrinkage during solidification and cooling. (The solidification volume change is 3% and the room temperature

Scale: 4:5 X = Pt-Pt/Rh o = Chromel-Alumel

Figure 12.

Thermocouple Placement for Casting K-189

Figure 13.
Cooling Curves for Lead Casting K-189

coefficient of thermal expansion is $16.3 \times 10^{-6}/^{\circ}\text{F.}$) This shrinkage can become a significant factor in the heat transfer across the mold/metal interface. The gap formed during solidification and cooling to room temperature was measured for this casting. This was accomplished by drilling a hole in the bottom of the mold, vacuum impregnating the mold with epoxy, sectioning the mold, photographing the epoxy-filled gap at 100X magnification, and measuring the photographed gap. The results are tabulated in Table 1.

Table 1. Gap Formed between Lead Casting K-189 and Mold

Location	Direction	Gap, inches
Riser	Radial	0.0050 to 0.0125
Rod (top)	Radial	0.0060 to 0.0100
Rod (bottom)	Radial	0.0055 to 0.0080
Bottom	Axial	0.0330

Lead Casting K-191

Run K-191 was made in an attempt to reduce the theorized heat loss during mold filling. Discussions with Dr. Lionel Sully, Southern Research Institute, Birmingham, Alabama, concerning his work in determining thermal conductance values revealed that in order for them to get valid experimental data, acetylene soot had to be applied to the metal permanent mold to reduce initial, rapid heat losses.¹⁷ Therefore, this technique was used to try to accomplish the same goal with the graphite mold.

In addition to applying soot to the interior mold surfaces, several other changes from Runs K-184 and K-189 were made. The first change was required to get a good soot coating and consisted of placing a press fit plug in the bottom of the mold. After prefiring, the mold and plug were coated separately and then the plug pressed into place. The remaining changes were made to improve the instrumentation. The six continuous plotting strip chart recorders were converted to read chromel-alumel thermocouples. To get more than eight thermocouples in the mold (only eight chromel-alumel vacuum tight pass-throughs are present in K furnace), the run was made in air. This allowed the leads to four additional thermocouples to exit through the open top of the furnace. An inverted graphite crucible with an inside diameter of

5.5 in. was placed over the mold to reduce convection currents and their accompanying heat losses.

The chromel-alumel thermocouples have three advantages. 1) They do not react with lead. 2) They are sufficiently strong to allow them to be inserted bare and eliminate the alumina spaghetti which is a potential heat sink. 3) The signal output is greater, thus allowing better resolution.

One other change was necessary. The Shops Department no longer stocks the 890S graphite used for the previous molds. Therefore, this mold was machined from the CS-312 graphite currently stocked. The primary difference between 890S and CS-312 graphite is the thermal conductivity of CS-312 which is 40-50% lower.

The thermocouples were positioned to give readings at four axial locations. Two thermocouples were placed in the riser 1.12 in. above the rod/riser intersection (centerline and 0.37 in. from the centerline) and one centerline thermocouple was located 0.5 in. below the rod/riser intersection. Five were located 3.0 in. below the intersection (centerline, 0.37 in. from centerline, rod/mold interface, 0.06 in. from the interface into the mold and 0.37 in. into the mold). The remaining three were located at 5.5 in. from the rod/riser interface. These three thermocouples were 0.5 in. from the rod bottom and were located at the centerline, 0.37 in. from the centerline and at the interface. One thermocouple was positioned in the melt at the crucible center to measure the metal temperature at pour.

All thermocouples except those at the interface locations were the standard welded thermocouples. Those at the interface were intrinsic thermocouples. That is, they were unwelded couples whose emf output is generated by the combined results of chromel-lead and lead-alumel couples. The net result is the same emf output as a welded chromel-alumel thermocouple. All thermocouples were calibrated in the same manner as described above.

The results of Run K-191 at the rod's axial center are presented in Figure 14. The data from these and the remaining thermocouples are tabulated

Figure 14.
Cooling Curves at Axial Center for Lead Casting K-191

in Appendix E. The major item noted in these results is the relatively long solidification times compared to Run K-189. Also of interest is the slow cooling at the lead interface. Cooling followed by a short heating cycle is noted in both intrinsic thermocouples.

VII. COMPUTER SIMULATION RESULTS

Attempts have been made to calculate the time- temperature-location history for a solidifying casting in three different metal/mold systems. The first two are those described in Chapter VI -- uranium and lead rods 1.5 in. diam by 6 in. long cast into a graphite mold. The third system consists of a mild steel, L-shaped section cast in molding sand.

Uranium Casting Simulation

The first attempts made in this work to simulate casting solidification involved a two-dimensional model for the 1.5 in. diam by 6 in. long uranium rod cast experimentally as casting K-184. The model consisted of a 2⁰ wedge with an axis of symmetry located at the radial center of the rod. The model is shown schematically in Figure 15. Node sizes are listed in Appendix G and the thermal properties used are given in Appendix F.

The model contained the following assumptions:

- 1) Instantaneous mold filling.
- 2) Uniform molten metal temperature at the start of the simulation.
- 3) Intimate contact between metal and mold at all times (infinite thermal conductance).
- 4) No heat transfer out the bottom of the mold.
- 5) The top and sides of the mold and top of the metal lost heat by radiation only.
- 6) The uranium melted (solidified) over a 1⁰ F temperature range (2070-2071⁰ F).

The thermal properties used in these calculations are tabulated in Appendix F. All uranium properties were entered as temperature varying properties. This was also true for the specific heat of graphite. However, single, average values were used for the with- and across-grain thermal conductivity of graphite. Two values were required since graphite is an

Figure 15.
Two-dimensional Model for 1.5 in. Diam by 6 in. Long Rod

anisotropic material. This is the result of the graphite developing a preferred crystallographic orientation during fabrication such that the thermal conductivity parallel to the working direction can be approximately twice that of the thermal conductivity in the across-grain direction. In the case of the mold material, the radial heat transfer corresponds to across-grain orientation and axial heat transfer is with-grain.

The thermal conductivity of the 890S graphite was determined by P. Wagner of LASL using flash diffusivity techniques.¹⁸ The values are higher than expected, but standard samples run at the same time gave correct values and repeated determinations for the 890S gave the same results. Both book and experimental thermal conductivity values were used in simulation runs with the best results obtained using the experimental data. The book values increased the total solidification times about 20%.

The calculated results for the centerline nodes are presented in Figure 16. The general relationship between the top and center cooling curves is the same for the calculated and experimental data (Figure 11). Both follow each other closely with the center curve completing solidification sooner than the top. The short thermal arrest present in the calculated data for the bottom location as compared to the experimental data can be attributed to a gap being formed between the rod and mold and prolonging the experimental solidification time. With a mold of this type, the casting rests against the graphite shoulder at the rod/riser interface. Therefore all axial shrinkage in the 6.0 in. rod length resulted in movement of the rod end away from the bottom. Once intimate contact has ceased to exist, the rapid cooling also is curtailed.

The single experimental mold temperature recorded during the time span simulated agreed closely with the calculated temperature. However, it does not appear this relationship remains this exact since the simulated data shows the graphite to have cooled to a lower temperature (1537°F) in 45 seconds than the experimental value recorded at 75 seconds (1553°F).

Color film plots of the simulated uranium casting solidification were

Figure 16.
Calculated Cooling Curves for Uranium Rod K-184

made. Color slides showing the solidification history are shown in Appendix H.

Two discrepancies which were noted throughout this work related to casting in graphite molds are present in this data. First, during the first part of the simulation, the calculated data is higher than the experimental data. Second, after the metal has solidified, the experimental data is higher than the calculated values. This relationship will be more thoroughly discussed when presenting the data for lead rods.

Lead Casting Simulation

The same model described above was used for the initial lead casting simulation. The results are plotted in Figure 17 and tabulated in Appendix G. The thermal properties used in verifying LQSLTR were used for this run (see Appendix F).

Comparing Figures 13 and 17 shows there is very little agreement between the calculated and experimental data for the riser and center locations. The experimental curve for 0.37 in. from the centerline coincides with the centerline values. However, the calculations show the location cools approximately twice as fast. Interestingly, there is reasonably good, though not perfect, agreement between the calculated and experimental results for the bottom thermocouples. The centerline position agrees reasonably well while the calculated values show the location 0.37 in. from the centerline cooling faster than experimentally determined.

This suggests that the controlling solidification/heat transfer factors may change during the solidification of the casting. Cooling at the bottom of the mold is characterized by rapid cooling of the metal. This mode appears to be compatible with the model used and its assumptions.

The second mode is characterized by rapid initial cooling of the melt followed by slow cooling once the centerline thermocouples reach the melting point of the lead. This mode is present at the upper thermocouples and is not

Figure 17.

Calculated Temperatures for Lead Casting K-189

accurately described by the model and its accompanying assumptions.

The solidification pattern for the lead rod is presented in Figure 18; the color codes and their respective temperature ranges are given in Table 2. The mold thermocouple data suggested that the riser was a dominant factor affecting the mold temperature. This is also shown in Figure 18. Initially, the mold heats up in the riser area as indicated by the light blue to magenta transition. As the mold temperature increases near the riser, the heat flows toward the bottom. This occurs more rapidly than does the heat flow in the radial direction because of the thermal conductivity being approximately twice as high in the with-grain (axial) direction as in the across-grain (radial) direction. Color slides showing the simulated solidification history for the lead casting are presented in Appendix H.

Table 2. Color Codes and Their Temperature Ranges
for Lead Casting Simulation Plots

Code	Color	Temperature Range, °F
101	Blue	100 and below
202	Light blue	>100 to 225
112	Deep magenta	>225 to 350
001	Magenta	>350 to 450
012	Medium magenta	>450 to 550
002	Light magenta	>550 to 620.5
121	Dark yellow	>620.5 to 650
021	Yellow	>650 to 725
020	Light yellow	>725 to 800
111	White	>800

The lack of agreement between experimental and calculated results suggested four possible areas which could cause the disagreement. These were:

- 1) The thermocouple assemblies were acting as a major heat sink.
- 2) The instantaneous mold filling/uniform melt temperature assumptions were not applicable to solidification simulation of parts cast under these conditions.
- 3) The assumption of intimate mold/casting contact during the entire cooling cycle was not applicable to this system.
- 4) The thermal properties were not correct.

Figure 18.

Calculated Temperature-Location Plots for Lead Casting

Models could be constructed and calculations made to evaluate all assumptions. However, non-uniform melt temperatures resulting from other than instantaneous mold filling could be modeled, but experimental data could not be obtained to give the model meaning. Modeling the mold filling would require using codes which incorporate fluid dynamic calculations and is not within the scope of this work.

The first model constructed was a three dimensional section two inches long through the center of the rod/mold (Figure 19a). The following assumptions were made:

- 1) Instantaneous mold filling.
- 2) Uniform initial metal temperature.
- 3) Intimate contact between the thermocouple assembly components and the thermocouple assembly with the metal and graphite.
- 4) No heat loss through either end of the model.

The gap between the mold and rod was incorporated into the model. This was accomplished by placing arithmetic nodes (nodes which are not capable of retaining heat) at the interface. The appropriate lead and graphite arithmetic nodes are then joined with conductors whose multiplication factor is equal to the surface area of the lead node and whose thermal conductance varies with temperature.

The model is divided into eight 45° wide columns, eight rows of lead nodes, and 4 rows of graphite (see Appendix G). The thermocouple assembly was incorporated into the model by placing it between two rows in one column as shown in Figure 19b. Node sizes are tabulated in Appendix G. Thermal property data of pure Al_2O_3 was used for the insulator and pure platinum for the thermocouple. The cross-sectional area of the single wire in the model is equivalent to the combined cross-sectional areas of the 0.016 in. diam platinum and platinum-10% rhodium thermocouples wires.

Computer runs were first made without the thermocouple in the model. These runs were used as a reference for future calculations and served to

Figure 19.
Three-Dimensional Model for Lead Cylinder

debug the model. Complete axial and radial symmetry was present in the results.

Figures 20 and 21 compare the reference data with the calculated results for the model with the thermocouple in place. The results show that initially, the thermocouple assembly decreases the relative temperature of the melt near the thermocouple. However, as solidification progresses, the assembly becomes a heat source, not a heat sink. This is not unreasonable when the thermal diffusivities of the materials involved are compared (Table 3). The thermal diffusivity of 890S graphite in the across-grain direction is approximately 10 times greater than for alumina. Thus, the radial heat flow in the area occupied by the thermocouple is significantly reduced. This requires that a portion of the heat be extracted through the liquid lead and causes about a 50% increase in solidification time for the center node.

Table 3. Thermal Diffusivities of Materials
in Three-Dimensional Model

Material	Thermal Diffusivity, ft^2/hr
Liquid Lead	0.27
Solid Lead	0.78
890S Graphite	1.4
Alumina	0.14
Platinum	0.98

The "heat source" also affects the slope of the time-temperature curve in the solid metal temperature region. This effect was noted for nodes with an outside radius up to 0.47 in., but was not noted in nodes closer to the mold/metal interface (0.75 in. from the centerline). While these results cannot be directly compared to the experimental results because of the no axial heat flow assumption, they do show that the thermocouples are a contributing factor in the variation between the calculated and experimental slopes of the cooling curves.

The above data show the perturbation caused by thermocouple assemblies 180° from the thermocouple. The instrumented mold had two thermocouple assemblies in the same radial plane at three different axial locations. The thermocouples were 90° apart with the beads located at the centerline and

Figure 20.

Calculated Cooling Curves Using Three-Dimensional Model

Figure 21.

Effect of Thermocouple on Calculated Results 0.31 in. from Centerline

0.37 in. from the centerline. Therefore, the bead of the thermocouple located at 0.5r was 0.53 in. from the ceramic insulator of the centerline thermocouple. The calculations show that the circumferential effect is similar to the radial effect at 0.031 in. from the centerline. Therefore, the temperature of the thermocouple located 0.5r would be expected to be affected not only by the thermocouple assembly measuring the temperature at that point, but also by the centerline thermocouple.

Comparison of run time for the three-dimensional models with and without the thermocouple assembly is an excellent example of the importance of the lumped parameter ratio. These runs were made using the explicit forward differencing routine CNFRWD. The routine calculates the time step based on the stability criteria for that calculational step. The more uniform the model (the smaller the lumped parameter ratio), the closer the time step can be to the maximum allowable. The ratio for the model without the thermocouple was 11 and the ratio for the model with the thermocouple was 134. The without- thermocouple model simulated 1 minute casting time in 230 seconds of CDC 7600 central processor time. Adding 23 nodes (3% increase) and their associated conductors increased the running time so that only 38 seconds simulation time could be run in 603 seconds central processor time. This is a 37% decrease in simulation time with an accompanying 162% increase in computer time.

The model used for the verification of LQSLTR was modified slightly and used to evaluate the effect of the thermal diffusivity of liquid and solid lead. The thermal conductance for lead cast in permanent molds is a function of the surface temperature of the lead¹⁷ and is not significantly affected by the surface temperature of the mold. Therefore, the arithmetic nodes were eliminated from the model and the center of the interface nodes moved to the interface. These nodes were then connected using a conductor whose conductance was calculated using the lead temperature only. [CINDA-3G allows the user to define how the temperature varying properties are to be calculated. If the multiplication factor (cross-sectional area/path length) is positive, the thermal conduction for the conductor is calculated using the average temperature of the connecting nodes. If the factor is negative, only

the temperature of the first node input for the conductor is used.]

Using the modified one-dimensional model, the thermal conductivity of the liquid lead and the specific heat of the solid lead were arbitrarily increased in separate runs to increase the thermal diffusivity of the liquid and decrease the diffusivity of the solid. The results indicate that the thermal diffusivity of the liquid lead had to be increased by a factor of 8 and the thermal diffusivity of the solid decreased by a factor of 4 to force the upper and lower portions of the curve to come into agreement with the experimental results. When the two changes were combined in one run, the slopes of the calculated and experimental curves were in agreement, but the length of the thermal arrest had doubled.

The thermal properties of lead are well defined. Variations in the thermal conductivity of liquid lead of about 40% were reported by the Thermophysical Properties Research Center at Purdue University.¹⁹ However, an 800% variation was required to get experimental/calculated agreement. This is too large a discrepancy to be attributed to thermal properties.

The reported values for the specific heat of solid lead showed excellent agreement (10%) with more scatter evident in the specific heat of liquid lead (40%). Again, this scatter is not of the same order of magnitude that would account for the experimental/calculated differences. Therefore, it must be concluded that factors other than erroneous thermal properties are present and are dominant.

Both the one-dimensional and three-dimensional models incorporated a gap between the metal and mold. Various conductance values were used in the one-dimensional model to determine the effect of thermal conductance on the cooling curves. Figure 22 presents the data from three computer runs using the thermal conductivity of 890S graphite. Three different models were used. The first was made using a one-dimensional slice of the two-dimensional model (3 rows of lead nodes joining 1 row of graphite). The model assumed no axial heat flow and intimate contact between the lead and graphite at all times. The second curve is replotted from Figure 6 and represents simulated

Figure 22.

Cooling Curves for Lead Casting K-189 Calculated Using One-Dimensional Model

solidification with near-intimate contact above the melting point and thermal conductance values extrapolated from data reported by Durham and Berry.³ The third curve uses thermal conductance values generated by Sully for soot-coated steel molds.¹⁷ All conductance values are tabulated in Table 4.

These runs can be compared to the data from casting K-189 (Figure 13) and with the two-dimensional lead calculations (Figure 17). Comparing the experimental and one-dimensional calculations show that the inclusion of a gap in the model does not bring the results into closer agreement. The net effect of the gap is to shift the curve to the right, i. e., increases the solidification time, but it does not appreciably affect the slopes of the curves. The higher conductance values (less severe insulation or gap effect) are of the proper magnitude for the uninsulated, 890S graphite mold used in run K-189.

The one-dimensional and two-dimensional runs assuming intimate contact show good agreement. The 2-D data is for the location 2.12 in. below the riser. The heating effect of the riser is reflected by the two-dimensional run requiring slightly longer to solidify than the one-dimensional run, despite the fact that the starting temperature of the 1-D run was 25°F higher.

Table 4. Thermal Conductance Values

Temp., °F	Intimate contact	Uncoated with gap ¹	Soot insulated ²
32	∞	.08	.008
550	∞	.08	.008
600	∞	.08	.021
620	∞	.08	-----
621	∞	1.00	1.000
1000	∞	100.00	100.000

1. Based on data reported by Durham and Berry.³

2. Sully data for soot-coated steel mold.¹⁶

Figure 23 shows the calculated time-temperature data generated using a one-dimensional model and CS-312 graphite thermal properties. Conductance values for lead cast into soot-coated steel were used for lead temperatures below the melting point (see Table 3). Five of the curves in Figure 23 correspond to the experimental locations monitored in lead casting K-191. The

Figure 23.

Cooling Curves for Lead Casting K-191 Calculated Using One-Dimensional Model

sixth curve is for the third node from the centerline.

Several interesting correlations were noted when comparing Figure 23 with Figure 14. For the mold data, the slopes of the calculated and experimental cooling curves are the same after 20 seconds. The major discrepancy is in the rapid heating at the start of the simulation. This rapid heat transfer to the mold causes a related rapid decrease in the lead interface temperature. For the lead interface temperature, the calculated temperatures are much lower than the experimental data. However, if the curves for the interface nodes are shifted and superimposed, the slopes are identical, including the changes in slope, below the melting point.

The shift required for the calculated curve is 10 seconds to the right and 40° F up. By using the lateral (time) shift only, the curves for the location 0.5r also agree below 600° F.

The inclusion of thermal conductance across the gap has the effect of increasing the calculated solidification time. Thus the possibility exists that by varying the conductance values, closer agreement could be reached. The data also indicates that the conductance above the melting point must be decreased as the calculations indicate there is no rapid heat transfer across the gap in the first seconds of solidification. (While thermocouple response time must be considered, the fact that the calculated curve is $60-80^{\circ}$ F hotter at all times indicates that response time is not a significant factor.) This decreased heat transfer should increase the solidification time, the same effect as shifting the calculated curves to the right.

The agreement of slopes below the melting point does not exist for the experimental and calculated centerline data. However, it was noted that the slope of the calculated curve for the third node from the centerline approached that of the experimental data for the centerline location. (The third node has an inside radius of 0.100 in., an outside radius of 0.140 in., and the center is 0.122 in. from the centerline.) The trend of relatively good correlation 2 nodes from the centerline and out, and poor correlation at the center, suggests that the model be examined.

Of particular concern is the method used for calculating the conductor between the center and adjacent node. The conductor factor for cylindrical geometries is calculated using the equation

$$f = A/\ln(r_2/r_1)$$

where A is the cross-sectional area and r_1 and r_2 are the node centers. For the conductor in question, $r_1=0$ and the above equation cannot be calculated. Therefore, the factor was calculated using the following approximation

$$f' = A/[or_c + \ln(r_2/or_c)]$$

where or_c is the outside radius of the center node. By making this assumption, more heat is being conducted through this conductor than theoretically possible. This could have the effect of causing the slope to be too large for the cooling curve of this node, particularly when heat removal is not affected by the release of the heat of fusion of this or adjacent nodes.

The upper portion (temperatures above the melting point) of the lead cooling curves again are not in agreement. The experimental data show a rapid temperature drop (melt temperature vs. maximum recorded temperature) at the start of the cooling process. This is followed by rapid cooling to some temperature just above the melting point and then a relatively long time period where the metal is at or just above the melting point.

Unfortunately, the changes required to bring the curves into agreement is the opposite of the corrections noted for the mold and interface temperatures. The experimental results indicate that a significant amount of heat was lost during mold filling. However, there is not an accompanying increase in mold temperature to show this occurred. In fact, the experimental mold temperatures are lower than calculated at all times. Therefore, the proposed decrease in thermal conductance across the gap at temperatures above the melting point of lead would be expected to further prevent the rapid decrease in temperature, not increase the temperature drop.

Steel Cast in Sand

A major portion of the lack of agreement between calculated and

experimental data for lead and uranium cast into graphite molds is believed caused by heat loss during mold filling. Therefore, the L-shaped steel casting reported by Bishop, Brandt, and Pellini was modeled in an attempt to calculate the results reported by them.²⁰ This work has been simulated using numerical techniques by Pehlke, et. al., who reported excellent agreement between experimental and calculated results.²¹

Bishop, Brandt, and Pellini cast mild steel into a sand mold. Sand is an excellent insulator and there should be little heat loss during mold filling. Figure 24 shows the model used to simulate the solidification of this L-shaped steel casting; all nodes are 0.4 in. square. The thermal properties for the steel and the specific heat of the sand were used as reported by Pehlke.

The thermal conductivity of molding sand is very difficult to define because of density variations resulting from the hand ramming process, variations in clay and moisture content, vaporization of water during casting solidification, and the structural variations in the SiO_2 sand particles. Pehlke reported several values for the thermal conductivity of sand and presented four equations used to define the thermal conductivity as a function of temperature. Using a trial-and-error procedure, Pehlke reported the best agreement between experimental and calculated results were obtained when the thermal conductivity was described by the equation

$$k = 0.1634 - 5.3969 \times 10^{-5}T + 2.4113 \times 10^{-8}T^2.$$

Experimentally, the steel casting was found to solidify in 62 minutes. Using the model shown in Figure 24 and the thermal conductivity for sand described by the above equation, less than 25% of the casting had solidified in 80 minutes when simulated using CINDA-3G and LQSLTR. Additional runs made using the other equations reported by Pehlke bracketed the proper solidification time when equations yielding the two highest thermal conductivities (C and D in Table 5) were used.

These results are summarized in Figure 25 with the color slides showing these plus intermediate results presented in Appendix H. An additional

Figure 24.
Two-Dimensional Model for L-Shaped Steel Casting

Curve "c"

Curve "e"

Curve "f"

Curve "d"

Figure 25.
Effect of Thermal Conductivity of Sand on the Solid Interface
Location after 60 Minutes

feature was added to the plotting routine which allowed the experimental results obtained by Bishop, Brandt and Pellini to be plotted over the calculated results. This was accomplished by inserting a call to a subroutine which read the 4020 plotting coordinates for the line to be generated and then plotted the resulting line. This proved to be an easy method for comparing the experimental and calculated results.

Table 5. Thermal Conductivity of Sand

Temp., °F	k, BTU/ft-°F					
	A*	B	C	D	E	F
32	.162	.297	.513	.380	.470	.338
932	.134	.227	.331	.439	.360	.396
1832	.145	.328	.419	.722	.518	.626
2732	.196	.600	.777	1.23	.850	1.08
2900	.210	.669	.874	1.35	.864	1.12

* Data sets A through D per Phelke¹⁹; data sets E and F per Atterton²⁰.

As Figure 25 and Appendix H show, the exact thermal conductivity/temperature relationship has not been reached. While the correct solidification time has been reproduced, the exact thermal history has not. Figure 26 shows the temperatures along the axis of symmetry at simulation times of 10, 30, and 60 seconds and Figure 27 shows the cooling curves near the centerline of the casting at the axis of symmetry. Also plotted are the experimental temperatures as determined from graphs presented by Bishop, et. al. The agreement with the reference data is within 1.5%. Pehlke reported "excellent" agreement, but quantitative data was not presented.

At least four possible sources of error exist. The pour temperature for this casting was given as 2750-2900° F; the simulation used a pour (initial liquid metal temperature) of 2875° F. Also in question is the temperature at which solidification is assumed complete. A melting range of 2723-2763° F was used in these calculations. Bishop, et. al., did not give the melting range of the particular steel used and Phelke used a 2640-2730° F for the temperature range used to liberate the heat of fusion. This temperature range plus the fact that the latent heat of fusion was included in the specific heat

Figure 26.

Temperature Gradient Along Axis of Symmetry
for L-Shaped Steel Casting

Figure 27.

Cooling Curve for Steel Casting at the Axial Center/Axis
of Symmetry Intersection

contributed to the differences in results when the thermal conductivity used by Pehike was used with this model and method of handling the heat of fusion.

It is felt that the major factor causing the disagreement is the way the mold/air interface was handled. Pehike assumed the surface remained at 80° F while the model used in this work allowed the temperature to increase, but did not allow heat to be transferred from the mold to the surroundings. This model imposed a less severe temperature gradient upon the system with the result that a higher thermal conductivity was required.

VIII. SUMMARY AND CONCLUSIONS

Film plotting routines were developed which allowed color microfilm to be output when using the CINDA-3G general purpose heat transfer code as the simulating program. The additional information required to plot the film was input using the CINDA-3G input sequence, thus eliminating the need for preparing large amounts of data as a separate entity. The combined effect of using the existing input sequence and obtaining color microfilm output demonstrates the feasibility of the stated objective of obtaining the computer results in an easy-to-interpret form.

The SINDA subroutine LQSLTR for handling the heat of fusion was verified by comparing the results of calculations using LQSLTR and those obtained using the specific heat method for handling the heat of fusion. The agreement was excellent when a 50° F melting range was used. The results from LQSLTR using a 1° F temperature range were similar and must be considered more accurate. This statement is based on the fact that the model used to calculate the solidification history more closely describes the system being simulated.

Verification of the CINDA-3G computer simulation of uranium and lead cast in graphite molds was not achieved. Experimental problems were encountered with the uranium/graphite system which precluded its use as a verifying medium. The problems with instrumentation is further evidence of the potential value of computer simulation of uranium casting solidification.

Disagreements between the calculated and experimental lead results are believed caused by the assumptions made in the model not being valid in a system using fast cooling, permanent-type molds. The model assumed instantaneous mold filling with a uniform melt temperature at the start of the solidification time. The fact that the maximum temperatures recorded by the thermocouples were 23° to 109° F lower than the pouring temperature indicates significant heat was lost prior to the mold being filled. Measured pour times for the experimental castings were about 1.5 seconds while riser thermocouples indicate the pour time may have been as long as 3 seconds. This is 8 to 16%

of the total solidification time and can not be considered "instantaneous."

The initial model also assumed intimate metal/mold contact during the entire solidification time. For a material such as lead with its relatively large solidification shrinkage, this is not applicable when casting into a chill mold. Using soot as a mold coating to reduce the experimental metal-mold heat transfer rate and thermal conductance values for lead cast into steel molds in the calculations, the solidification time and post-solidification cooling curves were brought into closer agreement.

The thermocouple assembly used with the Pt-Pt/Rh thermocouples was shown to be a significant perturbation in the system. This was demonstrated using a three-dimensional model. The use of this model also demonstrates the applicability and usefulness of this technique in general casting solidification work. Three dimensional models significantly increase the computational time required, but can be easily modeled and input using CINDA-3G.

A slow cooling L-shaped steel casting cast in sand was modeled. The solidification pattern determined by Bishop, Brandt, and Pelliini was approached using literature values for the thermal conductivity of molding sand. However, no two molding sands are exactly alike and the thermal conductivity/temperature relationship required to give precise agreement was not determined.

IX. FUTURE WORK

This study suggests several areas which require additional work in order to completely understand the problem and to take full advantage of computer graphics. There are also modifications to CINDA-3G which have the potential of increased efficiency and greater versatility.

Experimental

The two systems used to compare experimental data and simulation results from CINDA-3G can not be accurately defined by the techniques used in this work. Therefore, additional experimental data must be generated using a low thermal diffusivity mold material whose thermal properties are well defined. Two such systems are ceramic molds and plaster molds. Both these mold material systems should reduce the initial heat loss resulting from the high thermal diffusivity of the graphite and also have a thermal conductivity which can be more accurately measured than the conductivity of molding sand.

A cylinder cast in such a system should also be poured in an instrumented mold and the system modeled and simulated. The mold should have a hollow core so that the assumption used to calculate the conductor for the centerline node would be eliminated.

The combined use of fluid dynamics and heat transfer codes should be pursued as a technique to define the initial temperature gradient in the molten metal. Once the initial heat loss has been defined, better heat transfer simulation results should be obtainable.

Computer simulation of the L-shaped steel casting should be continued to bring the experimental and calculated data into closer agreement. While this would in part duplicate the above work, it is necessary if the applicability of simulation using CINDA-3G is to be demonstrated to the foundry industry. The referenced work of Bishop, Brandt and Pellini is highly regarded within the industry. Failure to simulate this work could negate the potential

beneficial results which can be realized through computer simulation using a general purpose heat transfer code such as CINDA-3G.

Computer Graphics

The software written for this work limits the axis of symmetry for plotting to symmetry about the y axis. The program should be modified to allow 180° symmetry about other axes as well as symmetry at angles other than 180° such as encountered in the sand casting simulation.

The current program uses color microfilm to display the time-temperature-location results. The use of video tapes has the potential of yielding faster turn-around, editing capabilities (both as a post-processor and interactively while the program is executing), and improved resolution.

Currently, the only color video computer capability which exists at LASL is the commercial VIDS (Video Information Display System). This unit is a stand-alone system driven by a PDP-11 computer. Its primary purpose is the display and manipulation (rotation, modification, etc.,) of color images. It requires a special input file and is not compatible with magnetic tape generated by the CDC 7600 computers for the FR-80/SC-4020 film plotters.

Commercial video systems have a resolution of 512 points per line. While this is half the addressable resolution obtainable on the 4020-type film plots, it should not be detrimental to solidification simulation work as now being performed. The program used to generate the microfilm output for this work plots the equivalent of 205 points per line, which is within the resolution of the commercial video systems.

Future plans being discussed at LASL include linking this equipment to the computer center computers. How this will be accomplished has not been defined, but it could take several forms. The simplest form would involve software which would translate the 4020 commands written to magnetic tape by the host (6600 and 7600) computers into commands compatible with VIDS. The user could view the color output anytime after the computer run generating the

data has been completed. The information to be plotted on film could then be reviewed and a decision made as to whether or not the film should be plotted. This would reduce the turn-around time during debugging color film routines from 24-48 hr to 1-12 hr depending on computer turn-around time for the parent job.

A more sophisticated system would allow editing of the magnetic tape. In this manner, unwanted frames could be removed or modified depending on the amount and type of information present on the magnetic tape and stored in the VIDS system. Carrying this one step further would allow true interactive graphics where the user would have the capability of conversing with the host computers. The forms this could take are many. It could range from viewing the plots frame-by-frame as they are generated with the user controlling whether or not the program proceeds further to allowing the user to modify the plot, indicate if the plot is to be transferred to magnetic tape for plotting on film, and then directing the computer to restart the program with modified (corrected) data, continue execution, or stop.

In addition to the computer graphics aspect, the interactive capability implies that the basic heat transfer code, i. e., CINDA-3G, must be modified to allow this type of interaction. For the go/no-go type interaction, the modification could well be minimal. However, if the full meaning of interactive programming is implemented, the changes could be major.

The improved resolution possible through the use of video tape is accomplished by converting the magnetic tape (digital data) to video tape (analog data). By using non-standard frequency modulation, a 2048 x 2048 point matrix can be plotted through the use of lasers. This would result in better resolution, a broader color spectrum, and faster film plotting. At this point, additional resolution is not needed. However, future applications could demand this increased capability.

In the near future at LASL, film routines should be available which will allow the user to use the full capabilities of the FR-80 film plotter. Currently, the resident mini-computer in each FR-80 converts the 4020 plotting

commands to FR-80 commands. However, this does not allow the user to have complete color control. For instance, calling red will position the magenta and yellow filters in front of the lens. Or, a user can plot red with one strike using the magenta filter and one strike using the yellow filter; both strikes would use the same spot size and source intensity. However, with the proposed system, different spot sizes and/or intensities could be used with each filter to give varying shades of red.

Modification of CINDA-3G

Any upgrading of CINDA-3G should only be done if a SINDA version will not be available at LASL in the near future. The main item which needs to be improved is the calling sequence for calculating the temperature varying properties. Currently, one call per conductor and one call per node is made each iteration. Besides being inefficient, the procedure limits the size (number of nodes) of the model which may be input. CINDA-3G is capable of handling models containing up to 4000 nodes. However, when the steel/sand model previously discussed was input using 1200 nodes and all nodes and conductors having temperature varying properties, the subroutine calls in VARBL1 exhausted the available core space and the problem could not be run. By passing array addresses instead of single point addresses, the number of calls could be significantly reduced. This would be particularly easy when input using the SINDA SIM and DIM commands.

The means by which VARGSM and VARCSM calculate the temperature varying properties of the conductors and nodes, respectively, can also be changed to significantly improve the efficiency of CINDA-3G. The only difference between the two routines is six statements which are inserted at the start of VARGSM to determine the temperature at which the thermal property is to be calculated. Therefore, VARCSM will be used as an example which applies to both subroutines.

Table 6 lists the sequence used to calculate the variable value. For each call from VARBL1, two additional calls are made which could easily be eliminated by combining D1D1WM and D1DEG1 with VARCSM. This would result in a

considerable time savings. A FORTRAN call requires about 50 machine cycles. For a problem with 3000 temperature dependent values to be calculated, this saves 300,000 cycles per time step. For a problem with 10,000 time steps, this is 3.0×10^9 cycles or a savings of 82 seconds computational time. Assuming such a run can be made in 5 minutes, this represents a 27% savings in computer time.

A second area which should be modified is the coding which requires that the Fahrenheit temperature scale be used when radiation conductors are present in the model. This requirement stems from the fact that 460.0 is added to the temperature to get the absolute temperature used in calculating radiation heat transfer. The code should be modified to allow either the Celsius or Fahrenheit temperature scales to be used. The changes are trivial, but care must be taken to see that all the required changes are made.

The third area of work is eliminating some of the nuisance items involved with LASL's version of CINDA-3G. Two of particular concern are the problem associated with the comment location on data cards and the fact that columns 71 and 72 are lost when FORTRAN statements are input into the four subroutines formed by the CINDA-3G preprocessor.

Table 6. Calling Sequence for Calculating
Variable Properties

```

Subroutine VARBL1
.
.
Call VARCSM (T(n),C(m),A(i),Factor)
.
End

Subroutine VARCSM (T,C,A,F)
Call D1D1WM (T,A,F,C)
Return
End

Subroutine D1D1WM (X,A,Z,Y)
Call D1DEG1 (X,A,W)
Y = W * Z
Return
End

Subroutine D1DEG1 (X,A,Y)
.
.
21 FORTRAN statements
.
End

```

X. THE USE OF COMPUTERS

The previous discussions have described the results of scientific computer calculations. These calculations were performed on the CDC 7600 computers operating under the CROS operating system. The computer was also used in preparing this manuscript. This was accomplished using the CDC 6600 computer operating under the KRONOS operating system in the interactive mode.

Text was entered via the terminal using either the edit or text mode. Interspersed with the text were commands to the program FORM. FORM is a LASL library routine which formats text according to the commands input with the text. By changing these commands' parameters, margins, vertical spacing, paging, captions, and paragraphs can be varied without re-entering the text. FORM also allows use of superscripts and subscripts when the text is output on the GSI Model 300 terminal.

Using FORM and its companion program PHINT to convert letters from lower to upper case, rough drafts can be printed on the line printer in upper-case letters only. Changes can then be made using the KRONOS text editor. When preliminary editing has been completed, the text can again be formatted by FORM and then listed on the GSI terminal. This procedure allows preliminary drafts to be printed on the high speed printers with final drafts and the report itself being output on the slower, but more versatile, GSI terminal.

The GSI terminal is an impact printer similar to an IBM Selectric typewriter manufactured by Gen-Com Systems, Inc., Foster City, California. It can be used as a computer terminal or an off-line typewriter. The terminal is capable of printing superscripts and subscript characters and can be used for plotting when the plot mode is entered.

The use of digital computers for word processing is particularly effective when multiple intermediate drafts of a report are required during its preparation. This feature was particularly useful in preparing this manuscript as the text was typed only once. This resulted in a considerable

savings in time and helped reduce typographical errors in the final copy.

When computers first came into being, computer hardware was the dominant cost factor. However, with the advances in solid state technology, this is no longer true. Today, software (manpower) is the prime cost factor. This reversal in cost factors has changed where the emphasis should be placed in computer projects. No longer is minimum (optimum) execution time the prime factor. This allows the use of higher level languages such as FORTRAN instead of assembly language programming. While this may result in less-than-optimum utilization of core and computer time, it can result in reduced manpower costs if good programming techniques are used. Techniques such as structured programming, internal and external documentation, and modular design will not only reduce the development costs, but minimize maintenance and modification costs.

The relative cost factors are borne out by data generated during this project. Four hundred ninety computer runs were required to this work. These runs required 8 hr 35 min central processor time at a cost of \$3800. While man-hours spent on the project were not logged, allowing 2 man-hours per computer run would give a total time of 818 hr. Using LASL estimates of 2080 working hrs per year and a cost of \$53,000 per year for a staff member, the cost for 818 hours is \$20,845.

ACKNOWLEDGMENTS

Many people contributed to this work. My particular thanks go to Marcial Lujan and Robert Vigil for their help in obtaining the experimental data; my advisor Prof. A. V. Houghton and Lara Baker of my graduate committee for their help and advice during this work; James Dickinson for his thought-provoking discussions; Paul Wagner for determining the thermal conductivity of the graphites used; Larry Ebaugh and Andrew Martinez for preparing the illustrations; and Ann Solem for upgrading FORM so it could be used to prepare this manuscript. I also wish to acknowledge work done by James Bradberry, Edwin Eaton, Henry Abeyta, Lawrence Manker, and Ramon Martinez. This work could not have been performed without the approval and support of the University of California, Los Alamos Scientific Laboratory and the Materials Technology Group's management.

The work was performed under the auspices of the Energy Research and Development Administration (formerly Atomic Energy Commission).

IX. REFERENCES

1. Chalmers, B., Principles of Solidification, John Wiley and Sons, New York, 1964.
2. Flemings, M. C., "The Solidification of Castings", Scientific American, Vol 231, 1974.
3. Durham, D. R., and Berry, J. T., "The Role of the Mold-Metal Interface During Solidification in Chill Molds", AFS Transactions, Vol 82, 1974.
4. Cole, G. S., "Transport Processes and Fluid Flow in Solidification", Solidification, American Society for Metals, Metals Park, Ohio, 1971.
5. Langford, D., "The Solidification of Spheres", Int. J. Heat Mass Transfer, Vol 9, 1966.
6. Ruddle, R. W., The Solidification of Castings, The Institute of Metals, London, 1957.
7. Smith, G. D., Numerical Solutions of Partial Differential Equations, Oxford University Press, New York, 1965.
8. Pehike, R. D., Kirt, M. J., Marrone, R. E., and Cook, D. J., "Numerical Simulation of Casting Solidification", AFS Cast Metals Research Journal, Vol 9, 1973.
9. Evers, N. R., Hartee, D. R., Ingham, J., Jackson, R., Sargent, R. J., and Wagstaff, J. B., "The Calculation of Variable Heat Flow in Solids," Phil. Trans. Roy. Soc., Vol 240, Series A, 1946.
10. Gaske, J. D., Fink, L. C., and Ishimoto, T., "Systems Improved Numerical Differencing Analyzer Users Manual", TRW Systems, Redondo Beach, Calif., 1970.
11. Edwards, A. L., "TRUMP: A Computer Program for Transient and Steady-State Temperature Distribution in Multidimensional Systems", Report UCRL 14754-Rev. 3, University of California, Livermore, 1969.
12. Lazaridis, A., "A Numerical Solution of the Solidification (or Melting) Problem in Multidimensional Space," PhD Dissertation, Columbia University, 1969.
13. Erickson, W. C., "The Use of Computer-Generated Color Movies in Casting Solidification," AFS Transactions, Vol 82, 1974.
14. Lewis, P. R., Baski, J. D., Thompson, L. R., "Chrysler Improved Numerical Differencing Analyser for 3rd Generation Computers," Chrysler Corporation Space Division, New Orleans, TN-AP-67-287, 1967.
15. "Adaptation of Chrysler Improved Numerical Differencing Analyzer to CDC 6000 Series Computers," adapted by Connor, R. J., Kannady, R. E., Jr., and Aimaza, J. E., Martin Marietta Corporation, Denver, M-68-22 (Rev. 1), 1968.
16. Defretin, G., Cottin, R., Ach, J., "Casting and Solidification of Uranium in Ceramic Molds", Founderie, Vol 320, 1973.
17. Sully, L. J. D., private communication.
18. Wagner, P., private communication.
19. Thermophysical Properties of Matter, Touloukian, Y. S., editor, IFI/Plenum, New York, 1970.
20. Brandt, F. A., Bishop, H. F., and Pellini, W. S., "Solidification of Corner and Core Positions", Trans. AFS, Vol 61, 1953.
22. Atterton, D. V., "The Apparent Thermal Conductivity of Moulding Materials at High Temperatures", J. Iron and Steel Institute, Vol 174, 1953.

APPENDIX A

SINDA SUBROUTINE LQSLTR

SUBROUTINE LQSLTR(N,TL,TH,S,C,TO,TN)

SUBROUTINE LQSLTR IS PART OF THE SINDA PROGRAM LIBRARY AND ALLOWS THE LATENT HEAT OF FUSION TO BE ACCOUNTED FOR.

ARGUMENTS--

N NUMBER OF SEQUENTIAL NODES TO BE OPERATED ON
TL LOW TEMPERATURE OF MELTING RANGE
TH HIGH TEMPERATURE OF RANGE (TH-TL MUST BE GE 1)
S(DV) FIRST SERIES VALUE OF THE PHASE CHANGE ENERGY RATE
C(DV) FIRST SERIES VALUE OF THE NODAL CAPACITANCES
TO(DV) FIRST SERIES VALUE OF THE OLD TEMPERATURES. VALUE MUST BE SAVED BY USER IN VARIABLES 1.
TN(DV) FIRST SERIES VALUE OF THE NEW TEMPERATURES

SUBROUTINE CALLS

TO LQSLTR -- VARBL2
BY LQSLTR -- NONE

DIMENSION S(1),C(1),TO(1),TN(1)
DO 100 I=1,N

CHECK IF NODE ENTERED AND/OR EXITED MELTING TEMPERATURE RANGE.
IF NODE HAS NOT CHECK NEXT NODE.

IF (TO(I) .LE. TL .AND. TN(I) .LE. TL) GO TO 100
IF (TO(I) .GE. TH .AND. TN(I) .GE. TH) GO TO 100

CALCULATE HEAT CONTENT BASED ON SPECIFIC HEAT ONLY (Q) AND HEAT LOSE RATE THROUGH MELTING TEMPERATURE RANGE (SC) BASED ON HEAT OF FUSION AND SPECIFIC HEAT.

Q = C(I) * (TN(I) - TO(I))
SC = S(I) + C(I)
IF (Q) 5,100,20
5 IF (TO(I) .LE. TH) GO TO 15

NODE ENTERED MELTING RANGE DURING THIS ITERATION AND HAS LOST HEAT THIS ITERATION. FIRST SUBTRACT SPECIFIC HEAT ACCOUNTED FOR IN PREVIOUS ITERATION FROM Q. THEN ADJUST TEMPERATURE (TN) BASED ON HEAT CONTENT IN MELTING RANGE. IF NODE STILL WITHIN MELTING RANGE, GO TO NEXT NODE.

Q = Q - C(I) * (TH - TO(I))
TN(I) = Q / SC + TH
IF (TN(I) .GE. TL) GO TO 100

NODE HAS COOLED BELOW MELTING RANGE DURING THIS ITERATION. CORRECT TEMPERATURE BASED ON SPECIFIC HEAT OF SOLID.

Q = Q - SC * (TL - TH)
TN(I) = Q / C(I) + TL
GO TO 100

NODE WAS IN MELTING RANGE AFTER LAST ITERATION AND IT LOST HEAT DURING THIS ITERATION.

15 TN(I) = Q / SC + TO(I)
IF (TN(I) .GE. TL) GO TO 100

```

      Q = Q - SC * (TL - TO(I))
      TN(I) = Q / C(I) + TL
      GO TO 100
C
C  NODE GAINED HEAT DURING THIS ITERATION AND WAS BELOW SOLIDUS TEMPERATURE
C  AFTER LAST ITERATION.
C
20 IF (TO(I) .GE. TL) GO TO 30
   Q = Q + C(I) * (TO(I) - TL)
   TN(I) = Q / SC + TL
   IF (TN(I) .LE. TH) GO TO 100
   Q = Q + SC * (TL - TH)
   TN(I) = Q / C(I) + TH
   GO TO 100
C
C  NODE GAINED HEAT DURING THIS ITERATION AND WAS IN THE MELTING RANGE
C  AFTER LAST ITERATION.
C
30 TN(I) = Q / SC + TO(I)
   IF (TN(I) .LE. TH) GO TO 100
   Q = Q + SC * (TO(I) - TH)
   TN(I) = Q / C(I) + TH
100 CONTINUE
   RETURN
   ENDD

```

APPENDIX B

PREPROCESSOR PRECIND

```

C      PROGRAM PRECIND (INP,FSET60=INP,OUT,FSET55,FSET65)
C
C      PREPROCESSOR FOR CINDA3G DOES TWO THINGS -- 1) ESTABLISHES NODE
C      LOCATIONS IN MATRIX FOR PLOTTING COLOR CROSS-SECTIONAL VIEWS, AND
C      2) ALLOWS USE OF SINDA SEQUENCE GENERATING COMMANDS FOR INPUTTING
C      NODES AND CONDUCTORS.
C
C      TWO SINDA MNEMONIC CODES ARE PROCESSED BY THE PREPROCESSOR. THEY ARE
C      SIM AND DIM. SIM WILL ALLOW GENERATION OF A SERIES OF NODES OR
C      CONDUCTORS OF A SINGLE MATERIAL WITH TEMPERATURE VARYING PROPERTIES.
C      DIM DOES THE SAME FOR A TWO-MATERIAL NODE OR CONDUCTOR. THIS IS
C      ACCOMPLISHED BY GENERATING THE REQUIRED NUMBER OF CGS OR CGD CARDS ON
C      FSET55. ALL CINDA-3G CARDS ARE COPIED TO THIS FILE SET AS THEY ARE
C      INPUT. SUBSEQUENT CONTROL CARDS WHICH RESULT IN THE INPUT FILE BEING
C      READ MUST HAVE (INP=FSET55) INSERTED IN THE ARGUMENT LIST.
C
C      IT IS ASSUMED THEAT THE INPUT DECK BEING READ BY PRECIND IS OF THE
C      FOLLOWING STRUCTURE--
C
C          CINDA INPUT DATA
C          BCD 3END OF DATA
C          FORTRAN SUBROUTINES (IF ANY)
C          $FM.
C          DATA TO BE READ BY SUBROUTINES (IF ANY)
C          $FM.
C          $EJ.
C
C      A TYPICAL CROS CONTROL CARD STREAM FOR RUNNING CINDA-3G AT LASL FOLLOWS
C
C      CARD NO.          CONTROL CARD
C      1  $JOB(NAME=WERICKSON,CL=U,AC=C11JFK,UA=76500000,LC=400000,PL=50,TL=2M)
C      2  $OPERM(FS=C11CIND,OAC=CMBDOT,FSI=CMBFSI)
C      3  $COPYF(I=C11CIND,O=FILE)
C      4  $COPYF(I=C11CIND,O=ERROR)
C      5  $COPYF(I=C11CIND,O=LIBRARY)
C      6  $REWIND(FILE)
C      7  $REWIND(ERROR)
C      8  $REWIND(LIBRARY)
C      9  $WHEN(TRUE=ERRORFLG)
C     10  $OPEN(FS=FSET13,SCT=5000)
C     11  $LDGO(I=FILE,SET=0)
C     12  $RUN(C=S,I=FSET13)
C     13  $RUN(C=S)
C     14  $COPYF(I=INP,O=FSET5)
C     15  $REWIND(FSET5)
C     16  $ULCR(I=LIBRARY,L=RUNLIB)
C     17  $LOAD(L=RUNLIB)
C     18  $LDGO(I=DUM)
C     19  $STOP.
C     20  $LABEL(ERRORFLG)
C     21  $LDGO(I=ERROR,OUT=DUMMY)
C     22  $STOP.
C
C      THE FOLLOWING CARDS SHOULD BE MODIFIED AS FOLLOWS TO USE THE PRE-
C      PROCESSOR.
C
C      INSERT AFTER CARD NO. 1
C          $RUN(C=S)
C          $LDGO.

```



```

C  REPLACE NO. 11 WITH $LDGO(I=FILE,SET=0,INP=FSET55)
C  REPLACE NO. 13 WITH $RUN(C=S,I=FSET55)
C  REPLACE NO. 14 WITH $COPYF(I=FSET55,O=FSET5)
C
C  SUBROUTINE CALLS
C  TO PRECIND -- NONE
C  BY PRECIND -- PRCFILM, PSINDA, STRIP, TRANSFM
C
      COMMON IMAGE(8),MNEMON
      KFLAG = 0
      PRINT 5
5     FORMAT (*1      THE FOLLOING INFORMATION WAS INPUT TO PRECIND PREPR
10    ESSOR.  COLUMNS 1, 8, AND 12 ARE MARKED BY 1, 8, AND T RESPECTIV
20    ELY.*,//,5X,*1      8      T *,//)
7     READ (60,10) (IMAGE(J),J=1,8)
      IF (EOF,60) 50,12
10    FORMAT (8A10)
C
C  ECHO PRINT INPUT
C
12   PRINT 15,(IMAGE(J),J=1,8)
15   FORMAT (5X,8A10)
      DECODE (13,20,IMAGE(1)) KOL1,MNEMON,KOL1213
20   FORMAT (A1,6X,A3,1X,A2)
      IF (KOL1.EQ.1HF) WRITE (55,10) (IMAGE(J),J=1,8)
      IF (KOL1.EQ.1HF) GO TO 7
      IF (MNEMON.EQ.3HREM .AND. KOL1213.EQ.2H$$ ) CALL PRCFILM (KFLAG)
      IF (MNEMON.NE.3HREM) CALL STRIP (IMAGE)
      IF (MNEMON.EQ.3HSIM .OR. MNEMON.EQ.3HDIM) CALL PSINDA
C
C  IMAGE WRITTEN TO FSET55 FOR INPUT TO CINDA-3G
C
      WRITE (55,10) (IMAGE(J),J=1,8)
      IF (IMAGE(2).NE.10H 3END OF D .OR. IMAGE(3).NE.3HATA ) GO TO 7
C
C  COPY ANY FORTRAN SUBROUTINES TO FSET55
C
22   READ (60,10) (IMAGE(J),J=1,8)
      IF (EOF,60) 50,25
25   WRITE (55,10) (IMAGE(J),J=1,8)
      PRINT 15,(IMAGE(J),J=1,8)
      GO TO 22
50   END FILE 55
C
C  COPY ANY DATA TO BE READ BY SUBROUTINES TO FSET55
C  IF FILM MATRIX DATA IS PRESENT (KFLAG=1), THIS DATA IS COPIED TO
C  FSET55 AHEAD OF OTHER DATA BY SUBROUTINE TRANSFM. AFTER THIS HAS BEEN
C  DONE, ANY OTHER INFORMATION ON FSET60 IS COPIED.
C
      IF (KFLAG.EQ.1) CALL TRANSFM
55   READ (60,10) (IMAGE(J),J=1,8)
      IF (EOF,60) 70,60
60   WRITE (55,10) (IMAGE(J),J=1,8)
      PRINT 15,(IMAGE(J),J=1,8)
      GO TO 55
70   END FILE 55
C
C  REWIND FSET55 SO IT IS READY TO BE READ BY CINDA-3G
C
      REWIND 55
      STOP
      END

```

```

      SUBROUTINE STRIP (IMAGE)
C
C SUBROUTINE STRIP CHECKS THE CARDS TO SEE IF A REMARK IS PRESENT AS
C INDICATED BY A $ AFTER THE DATA. IF IT IS PRESENT, THE REMARK IS
C STRIPPED FROM THE CARD IMAGE.
C
C THIS ACTION IS REQUIRED BECAUSE OF THE UNPREDICTABLE NATURE OF THE
C LASL VERSION OF CINDA-3G IN PROCESSING CGS AND CGD CARDS.
C
C SUBROUTINE CALLS
C TO STRIP -- FILMULT, PRCFILM, PRECIND
C BY STRIP -- NONE
C
      DIMENSION IALPHA(70),IMAGE(8)
      DECODE (70,30,IMAGE(2)) (IALPHA(J),J=1,70)
30  FORMAT (70A1)
      DO 50 K=1,70
        J = K
        IF (IALPHA(J).EQ.1H$ ) GO TO 70
-50  CONTINUE
      RETURN
70  DO 100 I=J,70
      IALPHA(I) = 1H
100  CONTINUE
      ENCODE (70,30,IMAGE(2)) (IALPHA(J),J=1,70)
      RETURN
      END

```

```

      SUBROUTINE TRANSFM
C
C  SUBROUTINE TRANSFM TRANSFERS DATA FROM FSET65 TO FSET55
C
C  SUBROUTINE CALLS
C  TO TRANSFM --  PRECIND
C  BY TRANSFM --  NONE
C
      END FILE 65
      REWIND 65
C
C  COPY DATA ON FSET 65 TO FSET55
C
      5 READ (65,10) I,J
      10 FORMAT (1X,A1,14)
      WRITE (55,10) I,J
      12 READ (65,15) I,J,K
      15 FORMAT (3I5)
      IF (EOF,65) 100,20
      20 WRITE (55,15) I,J,K
      GO TO 12
      100 END FILE 55
C
C  DOUBLE END OF FILE INDICATES NO MORE FILM DATA
C
      READ (65,15) I,J,K
      IF (EOF,65) 200,5
      200 RETURN
      END

```

```

C      SUBROUTINE PSINDA
C
C      SUBROUTINE PSINDA CONVERTS SIM AND DIM SINDA STATEMENTS INTO N CGS OR
C      CGD CINDA-3G STATEMENTS.  THE GENERAL FORMAT OF SIM AND DIM FOR NODES IS
C
C      DIM N1,N2,IN,T,A1,F1,A2,F2      $ THIS IS A DIM CARD INPUT
C
C      WHERE N1=NUMBER OF FIRST NODE, N2=NUMBER OF NODES TO BE GENERATED,
C      IN=INCREMENT FOR NODE NUMBERS,T=TEMPERATURE, A1 AND A2=ARRAY
C      NAMES FOR TEMPERATURE VARYING PROPERTIES, F1 AND F2=FACTORS.
C
C      FOR CONDUCTORS THE FORMAT IS
C
C      DIM G1,G2,IG,NA,INA,NB,INB,A1,F1,A2,F2  $ INPUT FOR DIM
C
C      WHERE G1=NUMBER FIRST CONDUCTOR, G2=NUMBER OF CONDUCTORS, IG=INCREMENT,
C      NA=NUMBER OF FIRST NODE, INA=INCREMENT FOR FIRST NODE,
C      NB=NUMBER OF SECOND NODE, INB=INCREMENT FOR SECOND NODE,
C      A1 AND A2 ARE TEMPERATURE VARYING PROPERTIES, AND F1 AND F2=
C      MULTIPLICATION FACTORS.
C
C      IN EACH CASE, A1 AND F1 ARE FOR THE FIRST MATERIAL AND A2 AND F2 ARE
C      FOR THE SECOND MATERIAL.  SIM DOES NOT USE A2 AND F2.
C
C      SUBROUTINE CALLS
C      TO PSINDA -- FILMULT, PRECIND
C      BY PSINDA -- CONDUCT, NODE
C
C      COMMON IMAGE(8),MNEMON
C      COMMON /FILM/ N2,IAXIS,NAXIS,ICHAR(70),NUMREL,KOMMA
C      DIMENSION IALPHA(70),WHERECM(15),FMAT(9),IREST(8)
C      INTEGER WHERECM,FMAT
C
C      INITIALIZE VARIABLES
C
C      DO 2 I=1,8
C      IREST(I) = 10H
C      2 FMAT(I) = 10H
C      FMAT(9) = 10H
C      DO 4 I=1,15
C      4 WHERECM(I) = 0
C
C      STORE 1 CHARACTER / WORD TO ENABLE SEARCH FOR COMMA AND FURTHER
C      PROCESSING
C
C      DECODE (70,10,IMAGE(2)) (IALPHA(J),J=1,70)
C      10 FORMAT (70A1)
C
C      COUNT THE COMMAS
C
C      NUMCOMA=0
C      DO 20 I=1,70
C      IF (IALPHA(I).EQ.1H$ ) GO TO 30
C      IF (IALPHA(I).NE.1H, ) GO TO 20
C      NUMCOMA=NUMCOMA + 1
C      WHERECM(NUMCOMA)=I
C      20 CONTINUE
C
C      NODE DATA SHOULD HAVE 5 OR 7 COMMAS AND CONDUCTOR DATA SHOULD HAVE
C      8 OR 10 COMMAS.  ANY OTHER NUMBER IS BAD INPUT.
C
C      30 IFLAG=0
C      KOMMA = 1
C      IF (NUMCOMA.EQ.7 .OR. NUMCOMA.EQ.5) CALL NODE (IALPHA,
C      +WHERECM,IFLAG)
C      IF (NUMCOMA.EQ.8 .OR. NUMCOMA.EQ.10) CALL CONDUCT (IALPHA,
C      +WHERECM,IFLAG)

```

```

C
C PRINT ERROR MESSAGE IF IFLAG=0
C
      IF (IFLAG.EQ.0) PRINT 500
500 FORMAT (10H0 *****,*IMPROPER FORMAT FOR SINDA CARD. CARD WAS
+COPIED TO CINDA-3G INPUT FILE AS LISTED ABOVE.*,5H***** ,/)
      RETURN
      END

```

```

      SUBROUTINE NODE (IALPHA,WHERECM,IFLAG)
C
C  SUBROUTINE NODE CONVERTS SIM AND DIM CARDS FOR NODES TO CGS AND CGD CARDS.
C
C  SUBROUTINE CALLS
C  TO NODE -- PSINDA
C  BY NODE -- NONE
C
      COMMON IMAGE(8),MNEMON
      COMMON /FILM/ NN2,IAXIS,NAXIS,ICHAR(70),NUMREL,KOMMA
      DIMENSION IALPHA(70),WHERECM(15),FMAT(9),IREST(8)
      INTEGER WHERECM,FMAT
      DO 50 I=1,8
50  IREST(I) = 10H
      IFLAG=1
C
C  PROCESS NODE DATA.  DETERMINE NUMBER OF CHARACTERS IN EACH NUMBER.
C
      ISUM=0
      IPOINT=2
      DO 100 I=1,3
        WHERECM(I)=WHERECM(I) - IPOINT
        IPOINT = IPOINT + WHERECM(I) + 1
        ISUM = ISUM + WHERECM(I)
100  CONTINUE
C
C  DETERMINE NUMBER OF CHARACTERS REMAINING IN CARD
C  CONSTRUCT VARIABLE FORMAT (FMAT) AND REREAD COL 11-80.
C
      IREMAIN = 71 - IPOINT
      IREM=IREMAIN - (10*INT(FLOAT(IREMAIN)/10.))
      IREMAIN = INT(FLOAT(IREMAIN)/10.) - 1
      ENCODE (47,120,FMAT) (WHERECM(J),J=1,3),IREMAIN,IREM
120  FORMAT (*('1X,*,3(*I*,I5,*,1X,*)',I5,*,A10,A*,I2,*)*)
      ISUM = ISUM + 71 - IPOINT
      IF (IREMAIN.EQ.0) IREMAIN=1
      IK = IREMAIN
      IF (IREM.NE.0) IK=IK + 1
      DECODE (ISUM,FMAT,IMAGE(2)) N1,N2,IN,(IREST(J),J=1,IK)
      NN2=N2
C
C  PUNCH N-1 NODE CARDS FOR CINDA-3G
C
      NSIGN=N1
      N1=IABS(N1)
      N2 = N1 + (N2 - 2)*IN
      IF (MNEMON.EQ.3HDIM ) MNEMON = 3HCGD
      IF (MNEMON.EQ.3HSIM ) MNEMON = 3HCGS
      DO 200 I=N1,N2,IN
        IS=ISIGN(I,NSIGN)
        WRITE (55,250) MNEMON,IS,(IREST(J),J=1,IK)
200  CONTINUE
250  FORMAT (7X,A3,1X,I6,*,*,6A10,A2)
C
C  CONSTRUCT LAST CARD IMAGE IN IMAGE FOR PUNCHING BY CALLING PROGRAM
C
      N1 = N2 + IN
      N1=ISIGN(N1,NSIGN)
      ENCODE (80,250,IMAGE(1)) MNEMON,N1,(IREST(J),J=1,7)
      RETURN
      END

```

```

      SUBROUTINE CONDUCT (IALPHA,WHERECM,IFLAG)
C
C SUBROUTINE CONDUCT CONVERTS SIM AND DIM CONDUCTOR INPUT TO CGS AND
C CGD CARD IMAGES.
C
C SUBROUTINE CALLS
C TO CONDUCT -- PSINDA
C BY CONDUCT -- NONE
C
      DIMENSION IALPHA(70),WHERECM(15),FMAT(9),IREST(8)
      COMMON IMAGE(8),MNEMON
      INTEGER WHERECM,FMAT
      DO 500 I=1,8
500  IREST(I) = '10H
      IFLAG=1
C
C PROCESS CONDUCTOR DATA
C FIRST DETERMINE NUMBER OF CHARACTERS IN EACH NUMBER
C
      ISUM=0
      IPOINT=2
      DO 550 I=1,7
      WHERECM(I)=WHERECM(I) - IPOINT
      IPOINT = IPOINT + WHERECM(I) + 1
      ISUM = ISUM + WHERECM(I)
550  CONTINUE
C
C DETERMINE NUMBER OF CHARACTERS IN CARD AND CONSTRUCT VARIABLE FORMAT.
C REREAD COLUMNS 11 - 80.
C
      IREMAIN = 71 - IPOINT
      IREM=IREMAIN - (10*INT(FLOAT(IREMAIN)/10.))
      IREMAIN = INT(FLOAT(IREMAIN)/10.) - 1
      ENCODE (87,590,FMAT) (WHERECM(J),J=1,7),IREMAIN,IREM
590  FORMAT (*('1X',*,7(*('1X',I5,*,1X,*),I5,*,A10,A*,I2,*))*
      ISUM = ISUM + 71 - IPOINT
      IF (IREMAIN.EQ.0) IREMAIN=1
      IK = IREMAIN
      IF (IREM.NE.0) IK=IK + 1
      DECODE (ISUM,FMAT,IMAGE(2)) NG1,NG2,IG,NA,INA,NB,INB,
      +      (IREST(J),J=1,IK)
C
C PUNCH N-1 APPROPRIATE CINDA-3G CONDUCTOR CARDS
C
      NSIGN=NG1
      NG1=IABS(NG1)
      NA = NA - INA
      NB = NB - INB
      NG2 = NG1 + (NG2 - 1) * IG
      IF (MNEMON.EQ.3HDIM) MNEMON = 3HCGD
      IF (MNEMON.EQ.3HSIM) MNEMON = 3HCGS
      DO 600 I=NG1,NG2,IG
      NA = NA + INA
      NB = NB + INB
      IS=ISIGN(I,NSIGN)
      NIS = 0
      NNA = 0
      NNB = 0
      NUMNIS = 1
      NUMNNA = 1
      NUMNNB = 1
C
C DETERMINE NUMBER OF DIGITS IN NUMBERS TO BE OUTPUT TO FSET55
C
      DO 595 J=1,5
      NIS = NIS * 10 + 9
      NNA = NNA * 10 + 9

```

```

      NNB = NNB * 10 + 9
      IF (I.GT.NIS) NUMNIS = NUMNIS + 1
      IF (NA.GT.NNA) NUMNNA = NUMNNA + 1
      IF (NB.GT.NNB) NUMNNB = NUMNNB + 1
595  CONTINUE
      IF (IS.LT.0) NUMNIS = NUMNIS + 1
      NSUM = 66 - NUMNIS - NUMNNA - NUMNNB
      JFIRST = INT(FLOAT(NSUM) / 10.)
      JLAST = NSUM - JFIRST * 10
      IF (JLAST.NE.0) ENCODE (60,605,FMAT) NUMNIS,NUMNNA,NUMNNB,JFIRST,
+ JLAST
      IF (JLAST.EQ.0) ENCODE (56,605,FMAT) NUMNIS,NUMNNA,NUMNNB,JFIRST
      IF (I.NE.NG2) WRITE (55,FMAT) MNEMON,IS,NA,NB,(IREST(JJ),JJ=1,IK)
600  CONTINUE
605  FORMAT(*(7X,A3,1X,I*,I6,6H,*,*,I ,I6,6H,*,*,I ,I6,5H,*,*, ,I6,
+*A10,A* I2,*)*)
606  FORMAT(*(7X,A3,1X,I*,I6,6H,*,*,I ,I6,6H,*,*,I ,I6,5H,*,*, ,I6,
+*A10)*)
C
C  CONSTRUCT LAST CARD IMAGE FOR PUNCHING BY MAIN PROGRAM
C
      IF (ILAST.NE.0) JFIRST = JFIRST + 1
      ENCODE (80,FMAT,IMAGE(1)) MNEMON,NG2,NA,NB,(IREST(JJ),JJ=1,JFIRST)
      RETURN
      END

```



```

C      SUBROUTINE PRCFILM (KFLAG)
C
C      THE FILM INPUT ALGORITHM MAKES USE OF CINDA-3G MNEMONIC CODES REM AND
C      THE $ USED TO INDICATE START OF COMMENTS. IF THE NODE MATRIX IS TO
C      BE GENERATED, A REM CARD WITH $$ IN COL 12 AND 13 IS INPUT. FOLLOWING
C      THE $$ ARE THE AXIS OF SYMMETRY AND ITS VALUE. THUS TO GENERATE A
C      FILM NODE LOCATION ARRAY WITH AN AXIS OF SYMMETRY LOCATED AT X=500,
C      THE FOLLOWING CARD IS INPUT
C
C      REM $$ X=500 FILM INITIALIZATION CARD
C
C      ALL INFORMATION AFTER THE INTEGER VALUE IS TREATED AS A COMMENT AND
C      IS OPTIONAL. THE VALUE INPUT ON THE REM CARD IS ASSUMED TO BE EQUAL
C      TO ZERO FOR INPUTING SUBSEQUENT NODE LOCATIONS.
C
C      THE LOCATION OF THE NODES IN THE PLOTTING MATRIX IS INDICATED AS A
C      COMMENT ON THE CINDA-3G INPUT CARD. AGAIN THE $$ IS USED TO SPECIFY
C      FILM OUTPUT. HOWEVER, IF THE REM $$ HAS NOT PROCEEDED IT, IT WILL
C      BE TREATED AS A REMARK. FOR THE BLANK CODE, THE X AND Y COORDINATE
C      LOCATIONS ARE INPUT AS NNN,MMM FOLLOWING THE $$ FOR THE GEN, CGS,
C      CGD, SIM, AND DIM MNEMONIC CODES, THE INPUT IS NNN, MMM, LLL, D WHERE
C      NNN=X COORDINATE, MMM=Y COORDINATE, LLL=LENGTH (IN RASTER POINTS)
C      BETWEEN NODES, AND D=DIRECTION IN WHICH THE NODES ARE INCREMENTED
C      (L=LEFT, R=RIGHT, U=UP, AND D=DOWN).
C
C      IT IS ASSUMED THAT 1) WITH THE EXCEPTION OF THE GEN, DIM, AND SIM
C      COMMANDS, ONLY INFO FOR ONE NODE PER CARD IS INPUT, 2) THE REM $$ CARD
C      IS BEFORE THE BCD 3NODE DATA CARD, 3) IF MORE THAN ONE JOB IS TO USE
C      THIS FILM OUTPUT ROUTINE, THE REM $$ MUST BE ENTERED BEFORE THE NODE
C      DATA CARD FOR EACH JOB, AND 4) THE LAST DIGIT OF THE AXIS VALUE IS
C      NOT IN COLUMN 80.
C
C      VALUES ON THE AXIS OF SYMMETRY HAVE ONLY ONE PLOTTING LOCATION.
C      ALL OTHER NODES HAVE TWO, ONE ON EACH SIDE OF THE AXIS. THE RELATIVE
C      NODE NUMBER AND ITS X AND Y COORDINATES ARE WRITTEN ON FSET65 AND THEN
C      COPIED TO FSET55 BY PRECIND
C
C      SUBROUTINE CALLS
C      TO PRCFILM -- PRECIND
C      BY PRCFILM -- FILMONE, FILMULT, STRIP
C
C      COMMON IMAGE(8),MNEMON
C      COMMON /FILM/ N2,IAXIS,NAXIS,ICHAR(70),NUMREL,KOMMA
C      DIMENSION FMAT(9)
C      INTEGER FMAT
C      DECODE (70,20,IMAGE(2)) (ICHAR(J),J=1,70)
C      20 FORMAT (70A1)
C
C      FIND AXIS VALUE NAME
C
C      DO 30 I=1,70
C      IF (ICHAR(I).EQ.1HX .OR. ICHAR(I).EQ.1HY ) GO TO 40
C      30 CONTINUE
C      PRINT 35
C      35 FORMAT (10H0 ***** ,*NO AXIS NAME WAS INPUT. FILM MATRIX VALUE
C      +S IGNORED ON REMAINING CARDS.*,5H***** )
C      RETURN
C
C      FIND FIRST DIGIT. IF NEGATIVE VALUE, NO SPACE MAY SEPARATE THE MINUM
C      SIGN AND THE FIRST DIGIT
C
C      40 IAXIS = ICHAR(I)
C      DO 50 JJ=1,70
C      IF((ICHAR(JJ).GE.1H0 .AND. ICHAR(JJ).LE.1H9 ) .OR.
C      +ICHAR(JJ).EQ.1H- ) GO TO 60
C      50 CONTINUE
C      PRINT 55
C      55 FORMAT (10H0 ***** ,*NO AXIS VALUE WAS INPUT. FILM MATRIX VALU
C      +ES ARE IGNORED ON SUBSEQUENT CARDS.*,5H***** ,/)

```

```

      RETURN
C
C   FIND SPACE AFTER LAST DIGIT
C
60 DO 70 KK=JJ,70
   IF (ICHAR(KK).EQ.1H ) GO TO 80
70 CONTINUE
   PRINT 75
75 FORMAT (10H0      ***** ,*LAST DIGIT OF AXIS VALUE WAS NOT FOUND BEF
+ORE COL 80.  FILM MATRIX VALUES ARE IGNORED ON REMAINING CARDS.*,
+5H***** ,/)
   RETURN
C
C   DETERMINE VALUE OF CENTER LINE.
C
80 CONTINUE
90 FORMAT (8A10)
   LENGTH = KK - JJ
   NUMCHAR = KK - 1
   M = JJ - 1
   ENCODE (8,100,FMAT) M,LENGTH
100 FORMAT (*(*,I2,*X,I*,I1,*))
   DECODE (NUMCHAR,FMAT,IMAGE(2)) NAXIS
   WRITE (65,105) IAXIS,NAXIS
105 FORMAT (1X,A1,I4)
   KFLAG = 1
C
C   KFLAG=1 INDICATES FILM MATRIX DATA TO BE PROCESSED
C   INITIATING REMARK CARD (REM) HAS BEEN PROCESSED. CINDA-3G AND SIM AND
C   DIM CARDS ARE NOW PROCESSED UNTIL ALL NODE CARDS HAVE BEEN INPUT.
C   CONTROL IS THEN RETURNED TO THE MAIN PROGRAM.
C
110 READ (60,90) (IMAGE(J),J=1,8)
   IF (EOF,60) 300,112
112 PRINT 115, (IMAGE(J),J=1,8)
115 FORMAT (5X,8A10)
   WRITE (55,90) (IMAGE(J),J=1,8)
   IF (IMAGE(1).NE.10H      BCD.OR. IMAGE(2).NE.10H 3NODE DAT
+ .OR. IMAGE(3).NE.2HA ) GO TO 110
C
C   NODE DATA IS STARTING TO BE INPUT. ALL CARDS MUST BE PROCESSED TO
C   DETERMINE RELATIVE NODE NUMBERS (NUMREL). THE CENTER LINE INPUT ON
C   THE CARD IS ASSUMED TO HAVE A VALUE OF 0. INFORMATION REQUIRED TO
C   GENERATE A SERIES OF NODES VIA DIM OR SIM IS PASSED TO PRCFILM VIA
C   LABELED COMMON FILM.
C
   NUMREL=0
117 READ (60,90) (IMAGE(J),J=1,8)
   PRINT 115, (IMAGE(J),J=1,8)
   DECODE (10,120,IMAGE(1)) MNEMON
120 FORMAT (7X,A3)
   IF (MNEMON.EQ.3HEND ) GO TO 200
   DECODE (70,125,IMAGE(2)) (ICHAR(J),J=1,70)
125 FORMAT (70A1)
   IF (MNEMON.EQ.3HGEN ) CALL FILMULT
   IF (MNEMON.EQ.3H ) CALL FILMONE
   IF (MNEMON.EQ.3HCGS ) CALL FILMONE
   IF (MNEMON.EQ.3HCGD ) CALL FILMONE
   IF (MNEMON.EQ.3HSIM .OR. MNEMON.EQ.3HDIM) CALL FILMULT
   CALL STRIP (IMAGE)
   WRITE (55,90) (IMAGE(J),J=1,8)
   GO TO 117
200 END FILE 65
   RETURN
300 PRINT 305
305 FORMAT (10H0      ***** ,*NO NODE DATA WAS INPUT. RUN TERMINATED BY
+ PRCFILM.*,5H***** ,/)
   END

```

```

      SUBROUTINE GETVAL (IX,IY,INC,IDIR)
C
C  SUBROUTINE GETVAL INTERPRETS THE COMMENT SECTION OF THE CARD TO OBTAIN
C  X AND Y COORDINATE POINTS FOR THE NODE.  IF THE CARD IS A GEN-TYPE,
C  GETVAL ALSO DETERMINES THE INCREMENT AND THE INCREMENTING DIRECTION.
C
C  SUBROUTINE CALLS
C  TO GETVAL -- FILMONE, FILMULT
C  BY GETVAL -- NONE
C
      COMMON IMAGE(8),MNEMON
      COMMON /FILM/ N2,IAXIS,NAXIS,ICHAR(70),NUMREL,KOMMA
      DIMENSION NUMBER(5),FMAT(9),MX(5),ID(5),WHERECM(15)
      INTEGER WHERECM,FMAT
      DO 5 I=1,15
      5 WHERECM(I) = 0
      DO 10 II=KOMMA,70
      10 IF (ICHAR(II).EQ.1H$ ) GO TO 30
      10 CONTINUE
      PRINT 15
      15 FORMAT (10H0 ***** ,*NO $ WAS FOUND ON ABOVE CARD.  NODES NOT
      +ADDED TO FILM MATRIX.*,5H***** ,/)
      IX=-1
      RETURN
C
C  LOCATE POSITION OF COMMAS
C
      30 NUMCOMA = 0
      IF (ICHAR(II+1).NE.1H$ ) GO TO 400
      II = II + 1
      DO 40 J=II,70
      IF (ICHAR(J).NE.1H, ) GO TO 40
      NUMCOMA = NUMCOMA + 1
      WHERECM(NUMCOMA) = J
      40 CONTINUE
C
C  LOCATION OF COMMAS HAS BEEN FOUND.  IF NUMBER OTHER THAN 1 OR 3,
C  FORMAT IS IMPROPER.
C
      IF (NUMCOMA.NE.1 .AND. NUMCOMA.NE.3 ) GO TO 500
      LAST = 1
      ISUM=0
      IEND = WHERECM(1)
      DO 100 M=1,3
      KOUNT = 0
      DO 50 J=II,IEND
      IF (ICHAR(J).LT.1H0 .OR. ICHAR(J).GT.1H9 ) GO TO 50
      KOUNT = KOUNT + 1
      NUMBER(KOUNT) = J
      50 CONTINUE
      MX(M) = NUMBER(1) - LAST
      ID(M) = NUMBER(KOUNT) - NUMBER(1) + 1
      ISUM = ISUM + MX(M) + ID(M)
      II = IEND + 1
      IEND = WHERECM(M+1)
      LAST = NUMBER(KOUNT) + 1
C
C  COMPUTED GO TO TRANSFERS CONTROL TO THE PROPER LOCATION WHEN PROCESSING
C  THE THREE INTEGER VALUES.  IF A SINGLE NODE, ONLY ONE COMMA IS PRESNT
C  AND DO LOOP LIMIT (IEND) MUST BE CORRECTED (STMT 60).  THE SECOND
C  PASS MUST CHECK IF ENTRY IS A SINGLE NODE ALSO.  IF IT IS, IX AND IY
C  ARE DETERMINED AND CONTROL IS RETURNED TO THE CALLING SUBROUTINE (STMT
C  70).  AFTER THE THIRD LOOP, THE LOCATION OF THE DIRECTION INDICATOR
C  MUST BE DETERMINED PRIOR TO DECODING TO OBTAIN THE VALUES (STMT 80).
C

```

```

      GO TO (60,70,80) M
60  IF (NUMCOMA.EQ.1) IEND = 70
      GO TO 100
70  IF (NUMCOMA.NE.1) GO TO 100
      ENCODE (21,75,FMAT) MX(1),ID(1),MX(2),ID(2)
75  FORMAT (*(*,I2,*X,I*,I4,**,I2,*X,I*,I4,*)*)
      DECODE (ISUM,FMAT,IMAGE(2)) IX,IY
      RETURN
80  IEND = 70
      DO 85 J=II,IEND
85  IF (ICHAR(J).NE.1H ) GO TO 90
      PRINT 87
87  FORMAT (10H0 *****,*NO DIRECTION WAS GIVEN FOR INCREMENTING NO
+DES IN FILM MATRIX.*,5H*****,/)
      IX = -1
      RETURN
90  MXL = J - LAST
      ISUM = ISUM + MXL + 1
      ENCODE (38,95,FMAT) (MX(J),ID(J),J=1,3),MXL
95  FORMAT (*(*,3(I2,*X,I*,I4,**,I2,*X,A1)*)
      DECODE (ISUM,FMAT,IMAGE(2)) IX,IY,INC,IDIR
100 CONTINUE
      RETURN
400 PRINT 405
405 FORMAT (10H0 *****,*DOUBLE $ NOT FOUND ON ABOVE CARD.  NODE LO
+CATION(S) NOT ENTERED IN FILM MATRIX.*,5H*****,/)
      IX = -1
      RETURN
500 PRINT 505
505 FORMAT (10H0 *****,*IMPROPER FILM INPUT FORMAT.  NODE(S) NOT E
+NTERED IN FILM MATRIX.*,5H*****,/ )
      IX = -1
      RETURN
      END

```

```

      SUBROUTINE FILMULT
C
C SUBROUTINE FILMULT GENERATES THE PLOTTING POINTS FOR THE NODES AND
C WRITES THIS INFORMATION TO FSET65. ONLY CARDS WHICH GENERATE MORE
C THAN ONE NODE ARE PROCESSED BY THIS ROUTINE.
C
C SUBROUTINE CALLS
C TO FILMULT -- PRCFILM
C BY FILMULT -- GENCARD, GETVAL, PSINDA, STRIP
C
      COMMON IMAGE(8), MNEMON
      COMMON /FILM/ N2, IAXIS, NAXIS, ICHAR(70), NUMREL, KOMMA
      DIMENSION IMSAVE(8), ISAVE(8)
      DO 2 I=1,8
2    IMSAVE(I) = IMAGE(I)
      CALL STRIP (IMAGE)
      IF (MNEMON.EQ.3HSIM .OR. MNEMON.EQ.3HDIM ) CALL PSINDA
      DO 4 I=1,8
      ISAVE(I) = IMAGE(I)
4    IMAGE(I) = IMSAVE(I)
      IF (MNEMON.EQ.3HCGS .OR. MNEMON.EQ.3HCGD)
      +   DECODE (70,5,IMAGE(2)) (ICAR(J),J=1,70)
5    FORMAT (70A1)
      IF (MNEMON.EQ.3HGEN ) CALL GENCARD
      CALL GETVAL (IX,IY,INC,IDIR)
      IF (IX.LT.0) GO TO 800
      IF (IDIR.EQ.1HL .OR. IDIR.EQ.1HU ) INC = -INC
      IFIRST = NUMREL + 1
      NUMREL = NUMREL + N2
      IF (IAXIS.EQ.1HY ) GO TO 500
      IF (IDIR.EQ.1HU .OR. IDIR.EQ.1HD ) GO TO 200
C
C   NODES ARE BEING GENERATED IN A HORIZONTAL ROW ABOUT A VERTICAL AXIS.
C
      DO 10 I=IFIRST,NUMREL
      WRITE (65,15) IX,IY,I
15    FORMAT (3I5)
      IXR = IXR + INC
      IX = IX + INC
10    CONTINUE
      GO TO 800
C
C   NODES ARE BEING GENERATED ABOUT A VERTICAL AXIS IN THE VERTICAL DIRECTION.
C
200  DO 210 I=IFIRST,NUMREL
      WRITE (65,15) IX,IY,I
      IY = IY + INC
210  CONTINUE
      GO TO 800
C
C   THE FOLLOWING CODE GENERATES NODES ABOUT A HORIZONTAL AXIS.
C
500  IF (IDIR.EQ.1HU .OR. IDIR.EQ.1HD ) GO TO 700
C
C   NODES GENERATED HORIZONTALLY
C
      DO 550 I=IFIRST,NUMREL
      WRITE (65,15) IX,IY,I
      IX = IX + INC
550  CONTINUE
      GO TO 800

```

```

C
C  GENERATE NODES VERTICALLY
C
700 DO 750 I=IFIRST,NUMREL
    WRITE (65,15) IX,IY,I
    IY = IY + INC
750 CONTINUE
800 DO 810 I=1,8
810 IMAGE(I) = ISAVE(I)
    RETURN
    END

```

```

      SUBROUTINE FILMONE
C
C  SUBROUTINE FILMONE DETERMINES THE PLOTTING LOCATION FOR NODES WHICH
C  ARE INPUT 1 NODE/CARD.
C
C  SUBROUTINE CALLS
C  TO FILMONE -- PRCFILM
C  BY FILMONE -- GETVAL
C
      COMMON IMAGE(8),MNEMON
      COMMON /FILM/ N2,IAXIS,NAXIS,ICHAR(70),NUMREL,KOMMA
      KOMMA = 1
      CALL GETVAL (IX,IY,INC,IDIR)
      IF (IX.LT.0) RETURN
      NUMREL = NUMREL + 1
      WRITE (65,15) IX,IY,NUMREL
15  FORMAT (3I5)
      RETURN
      END

```

```

      SUBROUTINE GENCARD
C
C  GENCARD DETERMINES THE NUMBER OF NODES TO BE GENERATED BY THE GEN
C  COMMAND AND RETURNS THIS NUMBER IN N2 TO FILMULT.
C
C  SUBROUTINE CALLS
C  TO GENCARD -- FILMULT
C  BY GENCARD -- NONE
C
      COMMON IMAGE(8),MNEMON
      COMMON /FILM/ N2,IAXIS,NAXIS,ICHAR(70),NUMREL,KOMMA
      DIMENSION WHERECM(10)
      INTEGER WHERECM,FMAT
      DO 5 I=1,10
5  WHERECM(I) = 0
      NUMCOMA = 0
C
C  DETERMINE LOCATION AND NUMBER OF COMMAS
C
      DO 10 I=1,70
      IF (ICHAR(I).EQ.1H$ ) GO TO 50
      IF (ICHAR(I).NE.1H, ) GO TO 10
      NUMCOMA = NUMCOMA + 1
      WHERECM(NUMCOMA) = I
10  CONTINUE
50  K = WHERECM(2) - WHERECM(1) - 1
C
C  OBTAIN VALUE OF NUMBER OF NODES TO GE GENERATED.
C
      ENCODE (9,60,FMAT) WHERECM(1),K
60  FORMAT (*(*,I2,*X,I*,I2,*))
      K = WHERECM(1) + K
      DECODE (K,FMAT,IMAGE(2)) N2
      RETURN
      END

```


APPENDIX C

SUBROUTINES FOR GENERATING COLOR PLOTS

SUBROUTINE FILMINT

FILMINT READS FILM PLOTTING COORDINATES AND STORES THEM USING
SUBROUTINES PCL AND PMN. IT THEN CALLS CCL AND CMN TO FILL IN THE
MATRIX. IT ALSO READS IN TEMPERATURE RANGES AND COLOR CODES USED IN
PLOTTING CROSS-SECTIONS.

SUBROUTINE CALLS

TO FILMINT -- EXECTN

BY FILMINT -- CCL, CMN, PCL, PMN

COMMON /FILMTRX/ TEMP(40,163),NODEPCL(125),FPCL(125),MPCLY(125),
+FCTCCL(125),FCL(125),MCCLY(125),ICHAR,
+KOFFSET,LENX,LENY,DBN,LBN,
+NODCCL1(125),NODCCL2(125),KTPCL,KTPMN,KTCCL,KTCMN,NOCCL,KTCMNF

COMMON /COLRANG/ TOPRANG(11),KOLOR(11)
COMMON /BORDER/ IVAL(26)
INTEGER FPCL,FCL

FILM MATRIX DATA WAS COPIED ONTO FSET12 AS FILE NO. 2. HOWEVER,
INPUT (CINDA-3G RUNLIB) DOES NOT READ END OF FILE. THEREFORE, MUST
DO DUMMY READ TO BY-PASS END OF FILE AND ALLOW FILM MATRIX DATA
TO BE READ.

1 READ (12,3) A

IF (EOF,12) 2,1

2 CONTINUE

3 FORMAT (A1)

IF ARRAY LENGTH OF TEMP IS CHANGED, DO 5 PARAMETERS MUST ALSO BE CHANGED
ALSO CHANGE LENY AND LENX

KTPCL = LENGTH PRIMARY CENTER LINE NODE ARRAY

KTPMN = LENGTH PRIMARY MATRIX NODE ARRAY

KTCCL = LENGTH CALCULATED CENTER LINE PLOTTING LOCATION ARRAY

KTCMN = LENGTH CALCULATED MATRIX PLOTTING LOCATION ARRAY

KTCMNF = NUMBER FOR FACTOR USED IN CALCULATING TEMP AT PLOTTING

POINT. POINTER TO CMFACT.

NOCCL = COLUMN LOCATION OF INNER MOST NODE

LENX = LENGTH X AXIS OF TEMP ARRAY

LENY = LENGTH Y AXIS OF TEMP ARRAY

DBN = FACTOR FOR CONVERTING TEMP ARRAY COORDINATES TO 4020 COORDINATES

LBN = FACTOR FOR CONVERTING TEMP ARRAY COORDINATES TO 4020 COORDINATES

KOFFSET = FACTOR TO ADJUST Y 4020 LOCATION INTO LENGTH OF TEMP ARRAY

ICHAR = PLOTTING CHARACTER (44 = *)

KKK = 6520

KTPCL = 0

KTPMN = -4

KTCCL = 0

KTCMN = -7

KTCMNF = 0

NOCCL = 1

LENY = 163

LENX = 40

DBN = 5

LBN = 5

KOFFSET = 22

ICHAR = 44

```

      ICHAR = SHIFT(ICHAR,12)
C
C   INITIALIZE TEMP ARRAY, READ LOCATION OF AXIS OF SYMMETRY (ASSUME
C   VALUE LOCATED ON X AXIS) AND THEN READ X AND Y COORDINATES FOR EACH NODE
C
      DO 5 I=1,KKK
        5 TEMP(I) = -1.
        READ (12,10) IAXIS
      10 FORMAT (2X,I4)
        IHOLD = IAXIS
        IF (IAxis.LT.0) IAXIS = IAXIS *(-1)
      15 READ (12,20) IX,IY,NODE
      20 FORMAT (3I5)
        IF (EOF,12) 30,25
      25 IF (IX.EQ.0) CALL PCL (IAXIS,IY,NODE)
        IF (IX.NE.0) CALL PMN (IAXIS,IX,IY,NODE)
        GO TO 15
      30 CONTINUE
        CALL CCL (IAXIS)
        IF (NOCL.EQ.1) GO TO 45
C
C   IF NOCL NOT EQUAL 1, THERE ARE NO NODES ON THE CENTERLINE. FOLLOWING
C   DETERMINES WHICH COLUMN HAS FIRST DATA.
C   SUBROUTINE CMN THEN FILLS IN MATRIX AND ESTABLISHED ORDER FOR CALCULAT-
C   ING TEMPERATURES.
C
      DO 40 I=2,LENX
        DO 35 J=1,LENY
          IF (TEMP(I,J).LT.0.) GO TO 35
          NOCL = I
          GO TO 45
        35 CONTINUE
      40 CONTINUE
      45 CALL CMN (IAXIS)
C
C   IF NEGATIVE INPUT FOR AXIS, SET NOCL = 0 TO SURPRESS PLOTTING OF SECOND PT.
C
      IF (IHOLD.LT.0) NOCL = 0
C
C   THE FOLLOWING 3 READS INPUT THE UPPER TEMPERATURES FOR THE TEMPERATURE
C   RANGES PLOTTED, THE COLOR CODES USED FOR THESE TEMPERATURES, AND THE
C   PLOTTING SEQUENCE (4020 X,Y COORDINATES) TO DRAW THE CASTING AND MOLD
C   OUTLINE, RESPECTIVELY.
C
C   THE COLOR CODES ARE INPUT AS 3 DIGIT OCTAL FIGURES. FROM LEFT TO RIGHT,
C   THE COLUMNS REPRESENT THE NUMBER OF STRIKES WITH THE CYAN, YELLOW,
C   AND MAGENTA FILTERS. THE VALUE OF THE DIGIT IS THE NUMBER OF STRIKES
C   WITH THAT FILTER AND SHOULD BE LESS THEAN 3. THIS CODE IS THAT USED
C   BY SUBROUTINE COLORO BY R. E. ELLIOT.
C
      READ (12,70) (TOPRANG(J),J=1,11)
      70 FORMAT (11F7.1)
      PRINT 72, (TOPRANG(J),J=1,11)
      72 FORMAT (*0 MAXIMUM TEMPERATURES FOR TEMPERATURE RANGES ARE*,/,
        + 11F10.1)
      READ (12,75) (KOLOR(J),J=1,11)
      75 FORMAT (11(1X,O3))
      PRINT 77, (KOLOR(J),J=1,11)
      77 FORMAT (*0 COLOR CODES FOR THESE TEMPERATURE RANGES ARE*,/,
        +5X,11(O3,5X))
      READ (12,80) (IVAL(J),J=1,26)
      80 FORMAT (16I5)
      RETURN
      END

```

```

      SUBROUTINE PCL (IAXIS,IY,NODE)
C
C   PCL STORES DATA FOR PRIMARY CENTERLINE NODES AND SETS FILM MATRIX
C   VALUES TO INDICATE LOCATION HAS KNOWN TEMPERATURE VALUE.
C
C   SUBROUTINE CALLS
C   TO PCL -- FILMINT
C   BY PCL -- NONE
C
      COMMON /FILMTRX/ TEMP(40,163),NODEPCL(125),FPCL(125),MPCLY(125),
      +FCTCCL(125),FCCL(125),MCCLY(125),ICHAR,
      +KOFFSET,LENX,LENY,DBN,LBN,
      +NODCCL1(125),NODCCL2(125),KTPCL,KTPMN,KTCCL,KTCMN,NOCL,KTCMNF
      INTEGER FPCL,FCCL
C
C   KTPCL = NUMBER PRIMARY CENTERLINE NODES (LENGTH ARRAY)
C   NODEPCL = CINDA-3G NODE NUMBER
C   FPCL = 4020 PLOTTING COMMAND
C   MPCLY = TEMP MATRIX LOCATION
C
      MY = (IY * DBN) - KOFFSET
      KTPCL = KTPCL + 1
      NODEPCL(KTPCL) = NODE
      IX = SHIFT(IAXIS,18)
      FPCL(KTPCL) = IX + ICHAR + IY
      MPCLY(KTPCL) = MY
      TEMP(1,MY) = 1.
      RETURN
      END

```

```

      SUBROUTINE CCL (IAXIS)
C
C  SUBROUTINE CCL FILLS IN PLOTTING LOCATIONS BETWEEN PRIMARY NODES ON
C  THE CENTER LINE AND STORES THE INFORMATION IN APPROPRIATE SCM ARRAYS.
C
C  SUBROUTINE CALLS
C  TO CCL -- FILMINT
C  BY CCL -- NONE
C  COMMON /FILMTRX/ TEMP(40,163),NODEPCL(125),FPCL(125),MPCLY(125),
C  +FCTCCL(125),FCCL(125),MCCLY(125),ICHR,
C  +KOFFSET,LENX,LARRAY,DBN,LBN,
C  +NODCCL1(125),NODCCL2(125),KTPCL,KTPMN,KTCCL,KTCMN,NOCL,KTCMNF
C  INTEGER FPCL,FCCL
C
C  FIND LOCATION OF FIRST KNOWN NODE ON CENTER LINE
C
C  DO 10 I=1,LARRAY
C  IF (TEMP(I,I).GT.0.) GO TO 20
10  CONTINUE
C  PRINT 15
15  FORMAT (10H      ***** ,*NO CENTERLINE NODES ENTERED.*,5H***** ,/)
C  NOCL = -1
C  RETURN
C
C  DO 30 FINDS POSITION OF NEXT NODE WHOSE TEMPERATURE IS NOT DEFINED.
C  IF FALLS THROUGH, ALL NODES DEFINED.
C
20  IFIRST = I
C  DO 30 II=IFIRST,LARRAY
C  IF (TEMP(I,II).LT.0.) GO TO 35
30  CONTINUE
C  RETURN
C
C  DO 40 FINDS POSITION OF NEXT NODE WITH KNOWN TEMPERATURE. IF FALLS
C  THROUGH, ALL NODES DEFINED.
C
35  CT = 0.
C  DO 40 JJ=II,LARRAY
C  IF (TEMP(I,JJ).GT.0) GO TO 50
C  CT = CT + 1.
40  CONTINUE
C  RETURN
C
C  FACT IS LINEAR INCREMENT FOR TEMPERATURES BETWEEN KNOWN NODES AND IS
C  USED AS FOLLOWS. FACTOR = FACTOR + FACT
C
C  NODCCL1 + FACTOR(NODCCL2 - NODCCL1)
C
50  FACT = 1./(CT + 1.)
C  IJ = II + CT - 1
C  FACTOR = FACT
C  JJJ = JJ
C  JJ = II - 1
C
C  KTCCL = NUMBER OF CALCULATED CENTERLINE PLOTTING POINTS
C  FCTCCL = PROPORTIONAL FACTOR FOR CALC TEMPERATURE AT PLOTTING POINT
C  FCCL = 4020 PLOTTING COMMAND
C  MCCLY = Y COORDINATE FOR TEMP
C  NODCCL1, NODCCL2 = Y COORDINATE (TEMP) OF NODES TO BE USED IN
C  CALCULATING TEMPERATURES OF POINT TO BE PLOTTED.
C
C  DO 60 K=II,IJ
C  JJ = JJ + 1
C  KTCCL = KTCCL + 1
C  FCTCCL(KTCCL) = FACTOR
C  IX = SHIFT(IAXIS,18)
C  FCCL(KTCCL) = IX + ICHR + ((JJ + KOFFSET) * LBN)

```

```

        MCCLY(KTCCL) = JJ
        TEMP(1, JJ) = 1.
        NODCCL1(KTCCL) = II - 1
        NODCCL2(KTCCL) = JJJ
        FACTOR = FACTOR + FACT
60 CONTINUE
C
C SET LOCATION TO START SEARCH FOR UNDEFINED NODES AND CYCLE BACK TO
C STMT 20
C
        I = IJ
        GO TO 20
END

```

```

C      SUBROUTINE PMN (NAXIS,IX,IY,NODE)
C
C      STORE PRIMARY NODES IN LCM ARRAY PMNODES.  ARRAY SINGULARLY DIMENSIONED
C      WITH DATA STORED IN BLOCKS OF 5.  LENGTH OF ARRAY USED (KTPMN)
C      POINTS TO START OF LAST BLOCK OF 5.  ISTORE USED AS TEMPORARY SCM
C      STORAGE WHICH IS THEN TRANSFERED TO LCM.  VALUES OF ISTORE ARE AS
C      FOLLOWS-- 1) RELATIVE NODE NUMBER, 2) 4020 COMMAND FOR POINT 1,
C      3) 4020 COMMAND FOR POINT 2, 4) AND 5) ARE FILM MATRIX COORDINATES.
C      DBN IS FACTOR FOR CONVERTING 4020 COORDINATES TO FILM MATRIX COORDINATES
C
C      SUBROUTINE CALLS
C      TO PMN -- FILMINT
C      BY PMN -- NONE
C
C      COMMON /FILMTRX/ TEMP(40,163),NODEPCL(125),FPCL(125),MPCLY(125),
C      +FCTCCL(125),FCCL(125),MCCLY(125),ICHAR,
C      +KOFFSET,LENX,LENY,DBN,LBN,
C      +NODCCL1(125),NODCCL2(125),KTPCL,KTPMN,KTCCL,KTCMN,NOCL,KTCMNF
C      INTEGER FPCL,FCCL
C
C      *****NONSTANDARD FORTRAN STATEMENT FOLLOWS*****
C
C      LCM /MATRIX/ PMNODES(7700),CMNODES(110000),CMFACT(10000)
C      INTEGER PMNODES, CMNODES
C      DIMENSION ISTORE(5)
C
C      KTPMN = LOCATION OF FIRST WORD OF BLOCK OF 5 USED TO STORE PRIMARY
C      MATRIX NODE DATA IN LCM
C
C      KTPMN = KTPMN + 5
C      MY = (IY * DBN) - KOFFSET
C      MX = IX * DBN
C      ISTORE(1) = NODE
C      ISTORE(2) = NAXIS + IX
C      IIX = NAXIS + IX
C      III = NAXIS - IX
C      IIX = SHIFT(IIX,18)
C      III = SHIFT(III,18)
C      ISTORE(2) = IIX + ICHAR + IY
C      ISTORE(3) = III + ICHAR + IY
C      ISTORE(4) = MX + 1
C      ISTORE(5) = MY
C
C      *****NONSTANDARD FORTRAN STATEMENT FOLLOWS*****
C
C      SMALL OUT (ISTORE,PMNODES(KTPMN),5)
C      TEMP(MX+1,MY) = 1.
C      RETURN
C      END

```

```

C      SUBROUTINE CMN (NAXIS)
C
C      SUBROUTINE CMN COMPLETES THE FILM MATRIX AND SPECIFIES HOW THE
C      TEMPERATURES ARE TO BE CALCULATED FOR THE REMAINING PLOTTING POINTS.
C      THE ROWS ARE FIRST TRAVERSED HORIZONTALLY TO FILL IN THE MISSING POINTS.
C      LOCATIONS WITH MORE THAN 4 MISSING LOCATIONS BETWEEN PRIMARY NODES
C      ARE NOT FILLED AT THIS TIME. AFTER ALL ROWS HAVE BEEN TRAVERSED,
C      THE COLUMNS ARE TREATED IN THE SAME MANNER STARTING WITH COLUMN 2.
C      COLUMN 1 WAS CALCULATED IN SUBROUTINE CCL.
C
C      EACH FILM MATRIX LOCATION REQUIRES 9 PIECES OF INFORMATION. THE
C      FACTOR USED TO CALCULATE THE TEMPERATURE IS STORED IN CMFACT (LCM).
C      THE REMAINING 8 ITEMS ARE STORED IN CMNODES (LCM) IN THE FOLLOWING
C      ORDER -- 1) AND 2) ARE X AND Y COORDINATES FOR VALUE ONE USED IN
C      CALCULATING TEMPERATURE, 3) AND 4) ARE SAME FOR VALUE TWO, 5) IS
C      4020 COMMAND FOR POINT ONE, 6) IS SAME FOR POINT TWO, AND 7) AND 8)
C      ARE FILM MATRIX (TEMP) LOCATION OF POINT.
C
C      SUBROUTINE CALLS
C      TO CMN -- FILMINT
C      BY CMN -- NONE
C
C      COMMON /FILMTRX/ TEMP(40,163),NODEPCL(125),FPCL(125),MPCLY(125),
C      +FCTCCL(125),FCCL(125),MCCLY(125),ICHAR,
C      +KOFFSET,IXLAST,IYLAST,DBN,LBN,
C      +NODCCL1(125),NODCCL2(125),KTPCL,KTPMN,KTCCL,KTCMN,NOCL,KTCMNF
C      INTEGER FPCL,FCCL
C
C      *****NONSTANDARD FORTRAN STATEMENT FOLLOWS*****
C
C      LCM /MATRIX/ PMNODES(7700),CMNODES(110000),CMFACT(10000)
C      INTEGER PMNODES, CMNODES
C      DIMENSION ISTORE(8)
C
C      FIRST DETERMINE ROW IN WHICH FIRST PRIMARY NODE OR CENTER LINE NODE
C      OCCURS.
C
C      DO 10 KY=1,IYLAST
C      DO 5 KX=NOCL,IXLAST
C      IF (TEMP(KX,KY).GT.0.) GO TO 15
C      5 CONTINUE
C      10 CONTINUE
C      PRINT 12
C      12 FORMAT (10H          ***** ,*FILM MATRIX EMPTY. RUN TERMINATED.*,
C      +5H***** ,/)
C      STOP
C
C      ROW WITH FIRST VALUE KNOWN (IFIRST). START FILLING IN ROWS.
C      DO 200 ALLOWS MATRIX FILLING PROCEDURE TO CYCLE TWICE.
C      DO 50 CONTROLS ROW BEING PROCESSED.
C      DO 16 FINDS FIRST LOCATION OF KNOWN TEMPERATURE (TEMP GR 0)
C      DO 20 FINDS FIRST EMPTY LOCATION (TEMP=-1).
C      DO 30 FIND FIRST FULL LOCATION (TEMP=+1) AFTER EMPTY LOCATION.
C
C      15 IFIRST = KY
C      DO 200 MMM=1,2
C      KYSAVE = KY
C      DO 50 KY=IFIRST,IYLAST
C      DO 16 JKT = NOCL,IXLAST
C      JJ = JKT
C      16 IF (TEMP(JKT,KY).GT.0.) GO TO 17
C      17 DO 20 KX=JJ,IXLAST
C      IF (TEMP(KX,KY).LT.0.) GO TO 25
C      20 CONTINUE
C      GO TO 50
C      25 CT = 0.
C      DO 30 KX1=KX,IXLAST

```

```

      IF (TEMP(KX1,KY).GT.0.) GO TO 35
      CT = CT + 1.
30  CONTINUE
      GO TO 50
C   IF MORE THAN 5 CONSECUTIVE LOCATIONS ARE EMPTY, THE LOCATIONS ARE NOT
C   FILLED AND THE PROGRAM SEARCHES FOR THE NEXT EMPTY SPOT VIA DO 20. WHEN
C   LESS THAN 6 SPACES, CALCULATE MULTIPLICATION FACTOR (FACTOR) ASSUMING
C   LINEAR VARIATION BETWEEN POINTS.
C
35  IF (CT.GT.5.) JJ = KX1
      IF (CT.GT.5.) GO TO 17
      FACT = 1./(CT + 1.)
      FACTOR = FACT
      II = KX + CT - 1
      IKY = (KY + KOFFSET) * LBN
C
C   DO 40 STORES FACTOR IN LCM VARIABLE CMFACT USING IMPLICIT TRANSFER.
C   THE TEMPERATURE LOCATIONS USED TO CALCULATE THE TEMPERATURE OF THIS
C   LOCATION, 4020 COMMANDS FOR POINTS ON BOTH SIDES OF AXIS OF
C   SYMMETRY, AND FILM MATRIX LOCATION ARE STORED USING ISTORE AND
C   EXPLICITLY TRANSFERRED TO LCM.
C
C   KTCMNF = POINTER FOR FACTOR (CMFACT) ARRAY
C   KTCMN = POINTER TO LCM ARRAY CMNODES
C
      DO 40 I=KX,II
      TEMP(I,KY) = 1.
      KTCMNF = KTCMNF + 1
      CMFACT(KTCMNF) = FACTOR
      IKX = (I - 1) * LBN
      ISTORE(1) = KX - 1
      ISTORE(2) = KY
      ISTORE(3) = KX1
      ISTORE(4) = KY
      IIX = IKX + NAXIS
      III = NAXIS - IKX
      IIX = SHIFT(IIX,18)
      III = SHIFT(III,18)
      ISTORE(5) = IIX + ICHAR + IKY
      ISTORE(6) = III + ICHAR + IKY
      ISTORE(7) = I
      ISTORE(8) = KY
      KTCMN = KTCMN + 8
C
C   *****NONSTANDARD FORTRAN STATEMENT FOLLOWS*****
C
      SMALL OUT (ISTORE,CMNODES(KTCMN),8)
      FACTOR = FACTOR + FACT
40  CONTINUE
      JJ = KX1
      GO TO 17
50  CONTINUE
C
C   TRAVERSE COLUMNS.
C
      IFLAG = 0
      DO 100 KX=NOCL,IXLAST
      DO 53 JKT=IFIRST,IYLAST
      JJ = JKT
53  IF (TEMP(KX,JKT).GT.0.) GO TO 55
C
C   FIND AN EMPTY LOCATION
C
55  DO 60 KY=JJ,IYLAST
      IF (TEMP(KX,KY).LT.0.) GO TO 65
60  CONTINUE
      GO TO 100

```



```

C
C FIND NEXT FULL LOCATION
C
65 CT = 0.
   DO 70 KY1=KY,IYLAST
   IF (TEMP(KX,KY1).GT.0.) GO TO 75
   CT = CT + 1.
70 CONTINUE
   GO TO 100
75 IF (CT.LT.6.) GO TO 80
   IFLAG = 1
   JJ = KY1
   GO TO 55
80 FACT = 1./(CT + 1.)
   FACTOR = FACT
   II = CT + KY - 1
   IKX = (KX - 1) * LBN
C
C STORE INFORMATION FOR NODES WHICH MUST HAVE TEMPERATURE CALCULATED
C
   DO 90 I=KY,II
   IKY = (I + KOFFSET) * LBN
   TEMP(KX,I) = 1.
   KTCMNF = KTCMNF + 1
   CMFACT(KTCMNF) = FACTOR
   ISTORE(1) = KX
   ISTORE(2) = KY - 1
   ISTORE(3) = KX
   ISTORE(4) = KY1
   IIX = IKX + NAXIS
   III = NAXIS - IKX
   IIX = SHIFT(IIX,18)
   III = SHIFT(III,18)
   ISTORE(5) = IIX + ICHAR + IKY
   ISTORE(6) = III + ICHAR + IKY
   ISTORE(7) = KX
   ISTORE(8) = I
   KTCMN = KTCMN + 8
C
C *****NONSTANDARD FORTRAN STATEMENT FOLLOWS*****
C
   SMALL OUT (ISTORE,CMNODES(KTCMN),8)
   FACTOR = FACTOR + FACT
90 CONTINUE
   JJ = KY1
   GO TO 55
100 CONTINUE
   KY = KYSAVE
   IF (IFLAG.EQ.0) RETURN
200 CONTINUE
   IF (IFLAG.EQ.0) RETURN
   PRINT 300,((TEMP(I,J),I=1,40),J=1,160)
300 FORMAT (10H *****,*FILM MATRIX NOT FILLED IN TWO PASSES.*,
+5H***** ,/,160(40F3.0,/))
   RETURN
   END

```

```

      SUBROUTINE TMPPLOT
C
C   TEMPERATURES. AFTER INITIALIZING VARIABLES, SUBROUTINE DETERMINES
C   SUBROUTINE TMPPLOT IS CALLED FROM OUTCAL (CINDA-3G) TO PLOT THE
C   TEMPERATURE RANGE FOR POINT AND THEN CALLS COLSTOR TO DETERMINE
C   COLOR AND STORE INFORMATION. IT THEN CALLS ROUTINES TO PLOT POINTS.
C   THIS DUAL PROCEDURE IS REQUIRED TO PREVENT EXCESSIVE FILTER SWITCHING
C   DURING PLOTTING.
C
C   SUBROUTINE CALLS
C   TO TMPPLOT -- OUTCAL
C   BY TMPPLOT -- ADV, COLOR, COLSTOR, DRV, PLOTIT, PLTTIME, TOLCM
C
      COMMON /COLOR/ KTR,KTRED,KTBL,KTBLUE,KTM,KTMAG,KTG,KTGREEN,KTW,
+KTWHITE,KTC,KTCYAN,KTY,KTYEL,RED(64),GREEN(64),YELLOW(64),BLUE(64)
+ ,WHITE(64),CYAN(64),MAGENTA(64)
      COMMON /FILMTRX/ TEMP(40,163),NODEPCL(125),FPCL(125),MPCLY(125),
+FCTCCL(125),FCCL(125),MCCLY(125),ICHAR,
+KOFFSET,LENX,LENY,DBN,LBN,
+NODCCL1(125),NODCCL2(125),KTPCL,KTPMN,KTCCL,KTCMN,NOCL,KTCMNF
      COMMON /COLRANG/ TOPRANG(11),KOLOR(11)
      COMMON /FIXCON/ A(13),TIMEM,B(34)
      COMMON /TEMP/ T
      COMMON /BORDER/ IVAL(26)
      LCM /PRIMARY/ LCRED(30000),LCGREEN(30000),LCBLUE(30000)
      LCM /COMPLEM/ LCYEL(30000),LCCYAN(30000),LCMAG(30000)
      LCM /MATRIX/ PMNODES(7700),CMNODES(110000),CMFACT(10000)
      INTEGER PMNODES, CMNODES
      INTEGER FPCL,FCCL,RED,BLUE,GREEN,WHITE,CYAN,YELLOW
      DIMENSION T(1),IP(10)
C
C   KT- = POINTER FOR SCM BUFFER FOR COLOR STORAGE ARRAYS
C   MC = NUMBER OF WORDS OF IP ARRAY TO BE STORED BY COLSTOR
C   TOPRANG = MAX TEMPERATURE FOR COLOR INDICATED BY KOLOR
C
      CALL ADV(2)
      KTR = 0
      KTBL = 0
      KTG = 0
      KTM = 0
      KTW = 0
      KTC = 0
      KTY = 0
      KTRED = 1
      KTBLUE = 1
      KTGREEN = 1
      KTMAG = 1
      KTWHITE = 1
      KTCYAN = 1
      KTYEL = 1
      MC = 1
C
C   PLOT PRIMARY CENTER LINE NODES
C
      IF (KTPCL.LE.0) GO TO 110
      DO 100 I=1,KTPCL
      M = MPCLY(I)
      N = NODEPCL(I)
      TEMP(1,M) = T(N)
      IP(1) = FPCL(I)
      DO 90 J=1,11
      IF (TEMP(1,M).GT.TOPRANG(J)) GO TO 90
      CALL COLSTOR (IP,KOLOR(J),MC)
      GO TO 100
      90 CONTINUE
      100 CONTINUE
C

```

```

C  PLOT CALCULATED CENTERLINE NODES
C
110 IF (KTCCL.LE.0) GO TO 250
    DO 210 I=1,KTCCL
        M = MCCLY(I)
        N1 = NODCCL1(I)
        N2 = NODCCL2(I)
        IP(1) = FCCL(I)
        TEMP(1,M) = TEMP(1,N1) + ((TEMP(1,N2) - TEMP(1,N1)) * FCTCCL(I))
        DO 200 J=1,11
            IF (TEMP(1,M).GT.TOPRANG(J)) GO TO 200
        CALL COLSTOR (IP,KOLOR(J),MC)
        GO TO 210
    200 CONTINUE
    210 CONTINUE

C
C  PLOT PRIMARY MATRIX NODES
C
C  NOCL = 0 INDICATES SECOND WORD STORED IS NOT TO BE PLOTTED
C
250 MC = 2
    IF (NOCL.EQ.0) MC = 1
    DO 350 I=1,KTPMN,5

C
C      *****NONSTANDARD FORTRAN STATEMENT FOLLOWS*****
C
        SMALL IN (IP,PMNODES(I),5)
        ND = IP(1)
        M = IP(4)
        N = IP(5)
        TEMP(M,N) = T(ND)
        DO 325 J=1,11
            IF (TEMP(M,N).GT.TOPRANG(J)) GO TO 325
        CALL COLSTOR (IP(2),KOLOR(J),MC)
        GO TO 350
    325 CONTINUE
    350 CONTINUE

C
C  PLOT CALCULATED MATRIX NODES
C
        KT = 0
        DO 450 I=1,KTCMN,8
            KT = KT + 1
            FACT = CMFACT(KT)

C
C      *****NONSTANDARD FORTRAN STATEMENT FOLLOWS*****
C
            SMALL IN (IP,CMNODES(I),8)
            MX = IP(7)
            MY = IP(8)
            NX1 = IP(1)
            NY1 = IP(2)
            NX2 = IP(3)
            NY2 = IP(4)
            TEMP(MX,MY) = TEMP(NX1,NY1) + ((TEMP(NX2,NY2) - TEMP(NX1,NY1))
            + * FACT)
            DO 425 J=1,11
                IF (TEMP(MX,MY).GT.TOPRANG(J)) GO TO 425
            CALL COLSTOR (IP(5),KOLOR(J),MC)
            GO TO 450
        425 CONTINUE
        450 CONTINUE

C
C  COLORS HAVE BEEN DEFINED.  EMPTY BUFFER ARRAYS BEFORE PLOTTING
C
        IF (KTG.NE.0) CALL TOLCM (KTG,KTGREEN,GREEN,LCGREEN)
        IF (KTR.NE.0) CALL TOLCM (KTR,KTRED,RED,LCRED)

```

```

      IF (KTY.NE.0) CALL TOLCM (KTY,KTYEL,YELLOW,LCYEL)
      IF (KTB.NE.0) CALL TOLCM (KTB,KTBLUE,BLUE,LCBLUE)
      IF (KTM.NE.0) CALL TOLCM (KTM,KTMAg,MAGENTA,LCMAG)
      IF (KTC.NE.0) CALL TOLCM (KTC,KTCYAN,CYAN,LCCYAN)
C
C   PLOT POINTS
C
      CALL COLOR (2.5)
      CALL PLTTIME (TIMEM)
      KTCYAN = KTCYAN - 1
      IF (KTCYAN.GT.0) CALL PLOTIT (LCCYAN,KTCYAN)
      KTBLUE = KTBLUE - 1
      CALL COLOR (3.0)
      IF (KTBLUE.GT.0) CALL PLOTIT (LCBLUE,KTBLUE)
      KTMAG = KTMAG - 1
      CALL COLOR (3.5)
      IF (KTMAG.GT.0) CALL PLOTIT (LCMAG,KTMAG)
      KTRED = KTRED - 1
      CALL COLOR (1.0)
      IF (KTRED.GT.0) CALL PLOTIT (LCRED,KTRED)
      KTYEL = KTYEL - 1
      CALL COLOR (1.5)
      IF (KTYEL.GT.0) CALL PLOTIT (LCYEL,KTYEL)
      KTGREEN = KTGREEN - 1
      CALL COLOR (2.0)
      IF (KTGREEN.GT.0) CALL PLOTIT (LCGREEN,KTGREEN)
      CALL COLOR (0.0)
C
C   DRAW OUTLINE OF CASTING AND MOLD
C
625  CONTINUE
      KDRAW = 23
      DO 650 I=1,KDRAW,2
650  CALL DRV (IVAL(I),IVAL(I+1),IVAL(I+2),IVAL(I+3))
      RETURN
      END

```

```

      SUBROUTINE COLSTOR (IP,KODE1,MC)
C
C SUBROUTINE COLSTOR DECODES THE COLOR CODE INTO ONE OR TWO OF THE
C SEVEN BASE COLORS. THIS IS DONE BY FIRST ADDING 1 TO THE COLOR CODE
C AND THEN SUBTRACTING 8 A SERIES OF TIMES. THE RESULTS ARE THEN USED
C IN A COMPUTED GO TO TO TRANSFER TO THE PROPER LOCATIONS. THE CODE
C THEN STORES THE X AND Y LOCATION(S) IN SUCCESSIVE WORDS IN A TEMPORARY
C ARRAY FOR THE RESPECTIVE COLORS. WHEN THE ARRAY IS FILLED TO WITHIN
C 3 WORDS OF ITS FULL LENGTH, THE INFORMATION IS TRANSFERRED TO LCM
C USING SUBROUTINE TOLCM.
C
C SUBROUTINE CALLS
C TO COLSTOR -- TMPPLOT
C BY COLSTOR -- TOLCM
C
      IMPLICIT INTEGER (A-Z)
      COMMON /COLOR/ KTR,KTRED,KTBL,KTBLUE,KTM,KTMA,KTG,KTGREEN,KTW,
      +KTWHITE,KTC,KTCYAN,KTY,KTYEL,RED(64),GREEN(64),YELLOW(64),BLUE(64)
      +,WHITE(64),CYAN(64),MAGENTA(64)
      LCM /PRIMARY/ LCRED(30000),LCGREEN(30000),LCBLUE(30000)
      LCM /COMPLEM/ LCYEL(30000),LCCYAN(30000),LCMAG(30000)
      DIMENSION IP(4)
C
C LENGTH= (LENGTH-4) OF BUFFER ARRAYS. 1SED TO DETERMINE IF BUFFER
C MUST BE TRANSFERRED TO LCM.
C IC = COUNTER USED WHEN A POINT IS TO BE OVERSTRUCK WITH SAME COLOR.
C
      LENGTH = 60
      IC = 0
      KODE = KODE1 + 1
      IF (KODE.LT.4) GO TO (10,20,20) KODE
      KODE = KODE - 8
      IF (KODE.LT.4) GO TO (30,40,50) KODE
      KODE = KODE - 8
      IF (KODE.LT.4) GO TO (60,70,80) KODE
      KODE = KODE - 48
      IF (KODE.LT.4) GO TO (90,100,110) KODE
      KODE = KODE - 8
      IF (KODE.LT.4) GO TO (120,10,130) KODE
      KODE = KODE - 8
      IF (KODE.LT.3) GO TO (140,150) KODE
      KODE = KODE - 48
      IF (KODE.LT.4) GO TO (160,170,180) KODE
      KODE = KODE - 8
      IF (KODE.LT.3) GO TO (190,200) KODE
      IF (KODE.NE.9) GO TO 210
C
C LIGHT GREEN (220 DECIMAL 144)
C
      5 IC = IC + 1
      DO 7 I=1,MC
      KTG = KTG + 1
      7 GREEN(KTG) = IP(I)
      IF (KTG.GT.LENGTH) CALL TOLCM (KTG,KTGREEN,GREEN,LCGREEN)
      IF (IC.EQ.1) GO TO 5
      RETURN
C
C WHITE (000 AND 111 DECIMAL 0 AND 73)
C
      10 DO 15 I=1,MC
      KTR = KTR + 1
      KTC = KTC + 1
      RED(KTR) = IP(I)
      15 CYAN(KTC) = IP(I)
      IF (KTR.GT.LENGTH) CALL TOLCM (KTR,KTRED,RED,LCRED)
      IF (KTC.GT.LENGTH) CALL TOLCM (KTC,KTCYAN,CYAN,LCCYAN)
      RETURN

```

```

C
C FOR KODE=2, MAGENTA (001 DECIMAL 1 )
C FOR KODE=3, LIGHT MAGENTA (002 DECIMAL 2)
C
20 DO 25 I=1,MC
   KTM = KTM + 1
25 MAGENTA(KTM) = IP(I)
   IF (KTM.GT.LENGTH) CALL TOLCM (KTM,KTMAG,MAGENTA,LCMAG)
   IF (KODE.NE.3) RETURN
   KODE = 0
   GO TO 20
C
C YELLOW (010 DECIMAL 8)
C
30 DO 35 I=1,MC
   KTY = KTY + 1
35 YELLOW(KTY) = IP(I)
   IF (KTY.GT.LENGTH) CALL TOLCM (KTY,KTYEL,YELLOW,LCYEL)
   RETURN
C
C RED (011 DECIMAL 9)
C
40 DO 45 I=1,MC
   KTR = KTR + 1
45 RED(KTR) = IP(I)
   IF (KTR.GT.LENGTH) CALL TOLCM(KTR,KTRED,RED,LCRED)
   RETURN
C
C MEDIUM MAGENTA (012 DECIMAL 10)
C
50 DO 55 I=1,MC
   KTM = KTM + 1
   KTR = KTR + 1
   RED(KTR) = IP(I)
   MAGENTA(KTM) = IP(I)
55 CONTINUE
   IF (KTR.GT.LENGTH) CALL TOLCM (KTR,KTRED,RED,LCRED)
   IF (KTM.GT.LENGTH) CALL TOLCM (KTM,KTMAG,MAGENTA,LCMAG)
   RETURN
C
C LIGHT YELLOW (020 DECIMAL 16)
C
60 IC = IC + 1
   DO 65 I=1,MC
   KTY = KTY + 1
65 YELLOW(KTY) = IP(I)
   IF (KTY.GT.LENGTH) CALL TOLCM (KTY,KTYEL,YELLOW,LCYEL)
   IF (IC.EQ.1) GO TO 60
   RETURN
C
C DARK YELLOW (021 DECIMAL 17)
C
70 DO 75 I=1,MC
   KTY = KTY + 1
   KTR = KTR + 1
   YELLOW(KTY) = IP(I)
75 RED(KTR) = IP(I)
   IF (KTR.GT.LENGTH) CALL TOLCM (KTR,KTRED,RED,LCRED)
   IF (KTY.GT.LENGTH) CALL TOLCM (KTY,KTYEL,YELLOW,LCYEL)
   RETURN
C
C PINK (022 DECIMAL 18)
C
80 IC = IC + 1
   DO 85 I=1,MC
   KTR = KTR + 1
85 RED(KTR) = IP(I)

```

```

        IF (KTR.GT.LENGTH) CALL TOLCM (KTR,KTRRED,RED,LCRED)
        IF (IC.EQ.1) GO TO 80
        RETURN
C
C   CYAN (100 DECIMAL 64)
C
    90 DO 95 I=1,MC
        KTC = KTC + 1
    95 CYAN(KTC) = IP(I)
        IF (KTC.GT.LENGTH) CALL TOLCM (KTC,KTCYAN,CYAN,LCYAN)
        RETURN
C
C   BLUE (101 DECIMAL 65)
C
    105 BLUE(KTB) = IP(I)
        IF (KTB.GT.LENGTH) CALL TOLCM (KTB,KTBBLUE,BLUE,LCBLUE)
        RETURN
C
C   PURPLE (102 DECIMAL 66)
C
    110 DO 115 I=1,MC
        KTB = KTB + 1
        KTM = KTM + 1
        BLUE(KTB) = IP(I)
    115 MAGENTA(KTM) = IP(I)
        IF (KTB.GT.LENGTH) CALL TOLCM (KTB,KTBBLUE,BLUE,LCBLUE)
        IF (KTM.GT.LENGTH) CALL TOLCM (KTM,KTMAG,MAGENTA,LCMAG)
        RETURN
C
C   GREEN (113 DECIMAL 72)
C
    120 DO 125 I=1,MC
        KTG = KTG + 1
    125 GREEN(KTG) = IP(I)
        IF (KTG.GT.LENGTH) CALL TOLCM (KTG,KTGREEN,GREEN,LCGREEN)
        RETURN
C
C   DARK PURPLE (112 DECIMAL 74)
C
    130 DO 135 I=1,MC
        KTB = KTB + 1
        KTR = KTR + 1
        BLUE(KTB) = IP(I)
    135 RED(KTR) = IP(I)
        IF (KTB.GT.LENGTH) CALL TOLCM (KTB,KTBBLUE,BLUE,LCBLUE)
        IF (KTR.GT.LENGTH) CALL TOLCM (KTR,KTRRED,RED,LCRED)
        RETURN
C
C   LIGHT GOLD (120 DECIMAL 80)
C
    140 DO 145 I=1,MC
        KTG = KTG + 1
        KTY = KTY + 1
        GREEN(KTG) = IP(I)
    145 YELLOW(KTY) = IP(I)
        IF (KTG.GT.LENGTH) CALL TOLCM (KTG,KTGREEN,GREEN,LCGREEN)
        IF (KTY.GT.LENGTH) CALL TOLCM (KTY,KTYEL,YELLOW,LCYEL)
        RETURN
C
C   DARK GOLD (121 DECIMAL 81)
C
    150 DO 155 I=1,MC
        KTG = KTG + 1
        KTR = KTR + 1
        GREEN(KTG) = IP(I)
    155 RED(KTR) = IP(I)
        IF (KTG.GT.LENGTH) CALL TOLCM (KTG,KTGREEN,GREEN,LCGREEN)

```

```

        IF (KTR.GT.LENGTH) CALL TOLCM (KTR,KTRED,RED,LCRED)
        RETURN
C
C LIGHT CYAN (200 DECIMAL 128)
C
160 IC = IC + 1
    DO 165 I=1,MC
        KTC = KTC + 1
165 CYAN(KTC) = IP(I)
    IF (KTC.GT.LENGTH) CALL TOLCM (KTC,KTCTYAN,CYAN,LCCYAN)
    IF (IC.EQ.1) GO TO 160
    RETURN
C
C POWDER BLUE (201 DECIMAL 129)
C
170 DO 175 I=1,MC
    KTC = KTC + 1
    KTB = KTB + 1
    CYAN(KTC) = IP(I)
175 BLUE(KTB) = IP(I)
    IF (KTB.GT.LENGTH) CALL TOLCM (KTB,KTBLUE,BLUE,LCBLUE)
    IF (KTC.GT.LENGTH) CALL TOLCM (KTC,KTCTYAN,CYAN,LCCYAN)
    RETURN
C
C MEDIUM BLUE (202 DECIMAL 130)
C
180 IC = IC + 1
    DO 185 I=1,MC
        KTB = KTB + 1
185 BLUE(KTB) = IP(I)
    IF (KTB.GT.LENGTH) CALL TOLCM (KTB,KTBLUE,BLUE,LCBLUE)
    IF (IC.EQ.1) GO TO 180
    RETURN
C
C BLUE GREEN (210 DECIMAL 136)
C
190 DO 195 I=1,MC
    KTC = KTC + 1
    KTG = KTG + 1
    CYAN(KTC) = IP(I)
195 GREEN(KTG) = IP(I)
    IF (KTC.GT.LENGTH) CALL TOLCM (KTC,KTCTYAN,CYAN,LCCYAN)
    IF (KTG.GT.LENGTH) CALL TOLCM (KTG,KTGREEN,GREEN,LCGREEN)
    RETURN
C
C ICE BLUE (211 DECIMAL 137)
C
200 DO 205 I=1,MC
    KTG = KTG + 1
    KTB = KTB + 1
    GREEN(KTG) = IP(I)
205 BLUE(KTB) = IP(I)
    IF (KTG.GT.LENGTH) CALL TOLCM (KTG,KTGREEN,GREEN,LCGREEN)
    IF (KTB.GT.LENGTH) CALL TOLCM (KTB,KTBLUE,BLUE,LCBLUE)
    RETURN
210 PRINT 215,KODE1,(IP(J),J=1,10)
215 FORMAT (10H ***** ,*ILLEGAL COLOR CODE PASSED TO COLSTOR. CO
+DE AND ASSOCIATED INFO WAS*,5H***** ,/,5X,10I10,/,10X,
+*RUN TERMINATED*)
    STOP
    END

```



```

C      SUBROUTINE TOLCM (KTSCM,KTLCM,ISCM,LCM)
C      SUBROUTINE TOLCM TRANSFERS KTSCM WORDS FROM SCM ARRAY ISCM TO LCM
C      ARRAY LCM STARTING AT KTLCM
C
C      SUBROUTINE CALLS
C      TO TOLCM -- COLSTOR, TMPPLOT
C      BY TOLCM -- NONE
C
C      LARGE LCM(1)
C
C      *****NONSTANDARD FORTRAN STATEMENT FOLLOWS*****
C
C      SMALL OUT (ISCM,LCM(KTLCM),KTSCM)
C      KTLCM = KTLCM + KTSCM
C      KTSCM = 0
C      RETURN
C      END

```

```

      SUBROUTINE PLTIME (TIME)
C
C   PLTIME CONVERTS TIME INTO A SERIES OF HOLLERITH CHARACTERS STORED
C   IN ARRAY M AND PLOTS TIME AND ASSOCIATED CAPTION VERTICALLY ALONG
C   LEFT SIDE OF FRAME
C
C   SUBROUTINE CALLS
C   TO PLTIME -- TMPPLOT
C   BY PLTIME -- DLCH
C
      DIMENSION M(17)
      DATA (M(I),I=1,5)/1HT,1HI,1HM,1HE,1H /
      DATA M(9)/1R. /
      DATA (M(I),I=13,17)/1H ,1HS,1HE,1HC,1R. /
C
C   ASSUME TIME IN MINUTES
C
      TIME1 = TIME * 60.
      ENCODE (10,50,ITIME) TIME1
50  FORMAT (F10.3)
      DECODE (10,55,ITIME) (M(II),II=6,8),(M(II),II=10,12)
55  FORMAT (3X,3A1,1X,3A1)
30  IIY = 165
      IIX = 5
      DO 150 L=1,17
        J = 1
        IIY = IIY + 45
        IF (L.EQ.9 .OR. L.EQ.17) IIX = 17
        IF (L.EQ.9 .OR. L.EQ.17) J=0
        CALL DLCH (IIX,IIY,J,M(L),4)
        IIX = 5
150  CONTINUE
      RETURN
      END

```

IDENT PLOTIT

SUBROUTINE PLOTIT READS THE 4020 COMMANDS FROM LCM AND PASSES THEM TO
SYSTEM SUBROUTINE BS4020 IN REGISTER X1

ARGUMENTS

1 ARGUMNT STARTING LCM ADDRESS OF 4020 COMMANDS
2 B2 NUMBER OF 4020 COMMANDS TO BE PROCESSED

	ENTRY	PLOTIT	
	EXT	BS4020	
ARGUMT	BSS	2	
	VFD	42/OHPLOTIT,18/2	
PLOTIT	DATA	0	
	SX4	1	LOAD 1 INTO X4
	SA1	ARGUMT	MOVE STARTING LCM ADDRESS TO X1 (COMPLEMENT)
	SA2	B2	MOVE NUMBER POINTS TO X2
	EX3	-X1	COMPLEMENT LCM ADDRESS
	SX1	X1	TRUNCATE TO GET LOW ORDER 18 BITS (ARGUMENT)
	AX3	30	SHIFT BLOCK ADDRESS TO LOW ORDER 18 BITS OF X3
	IX1	X1+X3	GET ADDRESS WITHIN LCM BLOCK OF PASSED ARGUMENT
	IX7	X1+X2	X7 = ADDRESS LAST PCINT + 1
	IX6	X1-X4	X6 = STARTING LARGE CORE ADDRESS - 1
	SA7	ENDVAL	STORE CHECK VALUE IN ENDVAL
	SA6	LCM	STORE LARGE CORE ADDRESS IN LCM
LOOP	SA2	LCM	GET CURRENT LARGE CORE ADDRESS
	SA3	ENDVAL	GET CHECK VALUE
	IX7	X2+X4	INCREMENT LARGE CORE ADDRESS
	IX3	X3-X7	FORM CHECK SUM
	ZR	X3,PLOTIT	IF X3 = 0, RETURN
	RX1	X7	READ 4020 COMMAND INTO X1
	SA7	LCM	STORE CURRENT ADDRESS
	RJ	BS4020	CALL BS4020 WITH COMMAND IN X1
	SX4	1	RELOAD X4 WITH 1
	JP	LOOP	GO TO LOOP
LCM	BSSZ	1	
ENDVAL	BSSZ	1	
	END		

APPENDIX D
MODEL AND DATA FOR VERIFICATION OF LQSLTR

Table 7. Node Data for One-Dimensional Model

Node number	id (in.)	od (in.)	Volume (in ³)	Radial conductors in. x .01	Axial conductors in. x .01
1,101	0.000	0.060	0.1178	0.01037	0.1885
2,102	0.060	0.100	0.2094	0.01673	0.3351
3,103	0.100	0.140	0.3142	0.02918	0.5027
4,104	0.140	0.190	0.5400	0.04833	0.8639
5,105	0.190	0.240	0.7036	0.06315	1.126
6,106	0.240	0.310	1.260	0.08468	2.016
7,107	0.310	0.390	1.833	0.1110	2.932
8,108	0.390	0.490	2.880	0.1508	4.608
9,109	0.490	0.600	3.924	0.1832	6.278
10,110	0.600	0.750	6.627	0.4936	10.60
11,111	0.750	0.750	-----	0.04909	---
12	0.750	0.750	-----	0.9923	---
13	0.750	0.900	16.19	0.7076	---
14	0.900	1.050	19.14	0.9693	---
15	1.050	1.200	22.09	1.257	---
16	1.200	1.350	25.03	1.591	---
17	1.350	1.500	27.98	1.964	---
18	1.500	1.650	30.92	2.377	---
19	1.650	1.800	33.87	2.847	---
20	1.800	1.950	36.82	3.318	---
21	1.950	2.100	39.76	2.621	---
22	2.100	2.250	42.71	-----	---

Segment Size : 15°

Segment length graphite (nodes 12 to 22): 0.5 in.

Segment length lead (nodes 1 to 11 and 101 to 111): 0.25 in.

Table 8. Calculated Temperature of Center Line Nodes

Time, sec.	Temperature, ° F		
	Run 1	Run 2	Run 3
0	818.0	818.0	818.0
3	817.9	817.9	817.9
6	815.4	815.5	815.5
9	803.4	804.0	803.9
12	779.3	781.1	780.8
15	745.9	749.2	748.6
18	706.6	712.1	711.2
21	664.6	674.5	673.2
22	651.0	663.9	662.8
23	635.8	657.5	656.7
24	626.7	650.8	649.5
25	620.7	625.8	623.6
26	620.0	550.9	543.9
27	508.9	486.1	482.3
28	460.2	446.8	444.2
29	427.5	418.4	416.2
30	402.8	396.1	394.3

Run 1: LQSLTR with melting range 620-621° F.

Run 2: LQSLTR with melting range of 610-660° F.

Run 3: Latent heat of fusion added to specific heat over temperature range of 610-660° F.

APPENDIX E.
EXPERIMENTAL RESULTS

Table 9. Time-Temperature Data for Run K-184

Time, sec.	Temperature, ° F		
	3.37 in.*	6.12 in.	8.87 in.
5	2161	2116	2048#
6	2129	2098	2048#
7	2107	2096	2048#
8	2084	2096	2048#
9	2052	2082	2048#
10	2080	2075	2048#
11	2071	2071	2048#
12	2069	2069	2039
13	2067#	2067#	2039
14	2067#	2067#	1998
15	2067#	2067#	1980
16	2066#	2067#	1949
17	2066#	2067#	
18	2066#	2067#	
19	2066#	2067#	
20	2062	2033	
21	2057	1943	
22	2039	1907	

* Distance from top of mold

Location at melting point arrest

Table 10. Chemical Analysis of Lead
Used to Calibrate Thermocouples

Ag	3*	Co	<2	Ni	<3
Al	1.5	Cu	45	Sb	<15
B	<4	Fe	1.5	Si	20
Ba	<3	K	<60	Sn	<3
Be	<1	Li	<1	Sr	<2
Bi	6	Mg	<1	V	<2
Ca	<4	Mo	<1	Zn	<150
Cr	<1	Na	<4	Zr	<15

* Composition in parts per million; <
indicates less than limit of detection.

Table 11. Thermocouple Calibration Data

Recorder	Pt-Pt/Rh (K-189)		Chromel-Alumel (K-191)	
	Arrest, °F	Correction, °F	Arrest, °F	Correction, °F
1	619	+2	619	+2
2	594	+27	621	0
3	12	+9	617	+4
4	606	+15	621	0
5	612	+9	619	+2
6	606	+15	621	0
X-1	631	-10	622	-1
X-2	628	-7	617	+4
Multipoint	590*	+31	617	+4
Ceramics	603*	+18	603	+18
Top	---	---	619	+2
Bottom	---	---	621	0

* Chromel-alumel thermocouple

Table 12. Thermocouple Data from Run K-189

Time, sec	Temperature, °F				
	1*	2	3	4	5
1.0					710
2.0	684				669
3.0	663			637	644
3.3	---			635	635
3.7	---			626	633
4.0	645	644	696	619	626
4.5	---	---	658	615	---
5.0	641	644	638	597	576
7.0	634	640	621	554	
9.0	630	633	617	534	
11.0	629	630	617		
15.0	627	630	617		
20.0	622	626	617		
25.0	622	626	549		
30.0	622	626			
35.0	617	617			
38.0	581	576			

* Location 1: Center line, 2.25 in. from top

Location 2: 0.375 in. from centerline, 2.25 in. from top

Location 3: centerline, 5.25 in. from top

Location 4: Center line, 8.87 in. from top

Location 5: 0.375 in. from centerline, 8.87 in. from top

Table 13. Experimental Data from Casting K-191

Time, sec.	Temperature, °F										
	1	2	3	4	5	6	7	8	9	10	11
.5									748		
1.0		203	149	615	743		714	738	709		
1.5						748	743		631		
2.0	795	207	671	756	774	710	716	682	601		
3.0		394	779	757	770	710	705	720	527		
4.0		763	777	757	777	702	675	660	505		
5.0		765	768	756	748	689	642	648	493		
6.0	750	754	755	750	734	680	633	631	493		
7.0		747	752	736	730	669	628	628	501		
8.0		741	745	720	716	651	622	617	522		
9.0		716	707	703	702	631	617	604	507		
10.0		709	707	691	680	626	601	579	502	102	
12.0		694	664	662	649	608	567				
14.0	669	678	662	642	633	595	538				
15.0		669	651	635	630	595	523			104	
16.0		664	648	631	630	586					
18.0	636	640	639	626	626	612					
20.0		631	630	624	621	612				131	
22.0	617	626	626	624	621	612					
24.0		621	626	624	619	608					
26.0		619	626	624	619	604					
28.0		619	624	621	617	600					
30.0	615	619	619	619	617	600				147	135
35.0	619	617	619	612	595						
40.0	615	619	617	619	606	577			158		
45.0	615	619	617	619	594						
50.0	615	617	617	613	576				167		
55.0	615	603	576	556	545						
60.0	615	585	543	523	516				174	162	
65.0	605										
75.0	525										

Table 13 (cont.)

- * Location 1: Centerline riser, 2 in. from top of mold.
- Location 2: 0.5r in riser, 2.0 in. from top of mold.
- Location 3: Centerline rod, 3.62 in. from top of mold.
- Location 4: Centerline rod, 6.12 in. from top of mold.
- Location 5: 0.5r, 6.12 in. from top of the mold.
- Location 6: Mold/rod interface, 6.12 in. from the top of the mold.
- Location 7: Centerline rod, 8.62 in. from top of the mold.
- Location 8: 0.5r, 8.62 in. from top of the mold.
- Location 9: Mold/rod interface, 8.62 in. from top of the mold.
- Location 10: Mold, 0.06 in. from mold/rod interface, 6.12 in. from top of the mold.
- Location 11: Mold, 0.37 in. from mold/rod interface, 6.12 in. from top of the mold.

APPENDIX F
THERMAL PROPERTIES

Table 14. Thermal Conductivity

Material	Temperature, °F	k, BTU/hr-ft-°F
<hr/>		
Alumina	32	22.9
	260	15.3
	620	9.13
	980	6.01
890S Graphite		
with-grain		109.8
across-grain		57.8
CS-312 Graphite		
with-grain		50.8
across-grain		37.0
Lead	32	20.5
	260	19.5
	620	18.0
	621	9.0
	800	10.1
	900	11.0
Molding Sand	See text, page 70.	
Platinum	32	41.3
	260	41.4
	620	42.2
	980	43.6
1025 Steel	300	27.3

Table 14 (cont.)

	1600	17.2
	2723	18.7
	2763	15.2
	3000	15.2
Uranium	260	17.1
	800	21.0
	1520	26.8
	2070	32.1
	2500	32.1

Table 15. Specific Heat and Density as a Function of Temperature

Material	Temp., °F	Density, lbs./in ³	Sp. Heat, BTU/lb.-°F	Sp. Heat, BTU/in ³ -°F
Alumina	32	.1408	.1731	.02437
	392	.1408	.2438	.03433
	752	.1408	.2719	.03828
	1112	.1408	.2865	.04034
Graphite	32	.061	.10	.0061
	392	.061	.19	.012
	732	.061	.25	.015
	1832	.061	.35	.021
	2552	.061	.41	.025
Lead	68	.4093	.0305	.0125
	428	.4032	.0325	.0131
	620	.3970	.0334	.0133
	621	.3851	.0540	.0208
	968	.3730	.0475	.0177
Molding Sand	32	.0535	.173	.00926
	800	.0535	.265	.0142
	1500	.0535	.276	.0148
	2500	.0535	.290	.0155
	2800	.0535	.297	.0159
Platinum	32	.7748	.0314	.0243
	392	.7707	.0330	.0254
	572	.7686	.0336	.0259
	752	.7666	.0342	.0262
	932	.7646	.0347	.0265
1025 Steel	200	.2835	.118	.0415
	1300	.2773	.222	.0616

Table 15 (cont.)

	1800	.2745	.168	.0461
	2723	.2694	.200	.0539
	2763	.2456	.210	.0516
	3100	.2456	.210	.0516
Uranium	932	.6741	.0741	.0272
	1292	.6641	.0426	.0283
	1832	.6471	.0403	.0261
	2070	.6388	.0403	.0257
	2552	.6209	.0481	.0299

Table 16. Latent Heat of Fusion

Material	H_f , BTU/lb.
<hr/>	
Lead	10.
1025 Steel	117.
Uranium	15.

APPENDIX G
MODEL DATA AND CALCULATIONS

Table 17. Node Dimensions for Two-Dimensional Rod Model

Column	Row	Type	i.r.	o.r.	Length
1M	1-74	Metal*	0.000	0.030	0.125 ⁺
2M	1-74	Metal	0.030	0.060	0.125
3M	1-74	Metal	0.060	0.160	0.125
4M	1-74	Metal	0.160	0.260	0.125
5M	1-74	Metal	0.260	0.350	0.125
6M	1-74	Metal	0.350	0.440	0.125
7M	1-74	Metal	0.440	0.525	0.125
8M	1-74	Metal	0.525	0.610	0.125
9M	1-74	Metal	0.610	0.705	0.125
10M	1-49	Interface*	0.705	0.800	0.125
10M	50-74	Metal	0.705	0.800	0.125
11M	49-74	Metal	0.800	0.880	0.125
12M	49-74	Metal	0.880	0.960	0.125
13M	49-74	Interface	0.960	1.040	0.125
1G	1-3	Graphite	0.000	0.160	0.325 [#]
2G	1-3	Graphite	0.160	0.398	0.325
3G	1-3	Graphite	0.398	0.525	0.325
4G	1-3	Graphite	0.525	0.525	0.325
5G	1-3	Graphite	0.654	0.800	0.325
6G	1-19	Graphite	0.800	0.920	0.375 ⁻
7G	1-19	Graphite	0.920	1.040	0.375
8G	1-28	Graphite	1.040	1.167	0.375
9G	1-28	Graphite	1.167	1.294	0.375
10G	1-28	Graphite	1.294	1.421	0.375
11G	1-28	Graphite	1.421	1.548	0.375
12G	1-28	Graphite	1.548	1.675	0.375
13G	1-28	Graphite	1.675	1.802	0.375
14G	1-28	Graphite	1.802	1.929	0.375

Table 17 (cont.)

15G	1-28	Graphite	1.929	2.056	0.375
16G	1-28	Graphite	2.056	2.183	0.375
17G	1-28	Graphite	2.183	2.250	0.375

* The metal nodes also include interface nodes consisting of half metal and half graphite.

+ All nodes except boundary nodes (those nodes located along the top and bottom of the mold) are 0.125 in. long. The boundary nodes are 0.062 in. long.

Boundary nodes are 0.162 in. long.

- The bottom boundary node is 0.162 in. long, the next two nodes are 0.350 in. long, and the top boundary node is 0.187 in. long.

Table 18. Calculated Temperatures for Uranium Rod K-184

Time, sec.		Temperature, °F			
		Top	Center	Bottom	Mold
0	2309		2309	2309	1376
1	2309		2309	2186	1632
2	2308		2308	2129	1650
3	2304		2303	2071	1647
4	2292		2289	2070	1641
5	2274		2268	2037	1630
6	2252		2242	1957	1624
7	2227		2213	1941	1615
8	2202		2185	1908	1608
9	2178		2158	1857	1603
10	2155		2133	1827	1598
11	2134		2114	1785	1594
12	2117		2095	1750	1590
13	2102		2085	1715	1587
14	2089		2077	1682	1584
15	2080		2074	1650	1581
16	2076		2072	1624	1579
17	2073		2071	1599	1577
18	2072		2071	1577	1575
19	2071		2071	1557	1573
20	2071		2006	1540	1572
21	2071		1888	1524	1570
22	2070		1825	1511	1569
23	1990		1781	1499	1567
24	1911		1747	1488	1565
45	1598		1515	1420	1537

Table 19. Calculated Data for Lead Casting K-189

Time, sec.	Temperature, °F				
	1	2	3	4	5
1	818	818	750	747	
2	818	816	818	655	621
3	818	810	817	621	539
4	818	797	817	612	469
5	818	778	814	539	
7	815	732	802	503	
9	808	681	778	432	
10	802	664	762		
11	794	621	744		
12	784	621	722		
13	773	592	699		
14	761	549	679		
15	747	532	662		
16	732		646		
17		626			
18	701		621		
19	683		599		
20	668		477		
24	624				
25	621				
26	620				
27	513				

Table 20. Node Dimensions for Three-Dimension Rod Model

Coil	Type	i.r.	o.r.	Length	Volume, in. ³	
					w/o T.C.	w/T.C.
1	Lead	-----	0.060	0.25	0.00283	0.00282
2	Lead	0.060	0.120	0.25	0.00106	0.00069
3	Lead	0.120	0.180	0.25	0.00177	0.00140
4	Lead	0.180	0.250	0.25	0.00296	0.00253
5	Lead	0.250	0.360	0.25	0.00659	0.00591
6	Lead	0.360	0.470	0.25	0.00896	0.00829
7	Lead	0.470	0.600	0.25	0.0137	0.0129
8	Lead	0.600	0.750	0.25	0.0199	0.0190
8I	Interface	0.750	0.750	0.25	-----	-----
9I	Interface	0.750	0.750	0.50	-----	-----
9	Graphite	0.750	1.000	0.50	0.0859	0.0844
10	Graphite	1.000	1.250	0.50	0.110	0.109
11	Graphite	1.250	1.500	0.50	0.135	0.134
12	Graphite	1.500	1.750	0.50	0.160	0.158
13	Graphite	1.750	2.000	0.50	0.184	0.183
14	Graphite	2.000	2.250	0.50	0.207	0.207
1P	Platinum	0.000	0.250	*	-----	0.0001
2P	Platinum	0.250	0.470	*	-----	0.0001
3P	Platinum	0.470	0.750	*	-----	0.0001
4P	Platinum	0.750	1.000	*	-----	0.0001
5P	Platinum	1.000	1.250	*	-----	0.0001
6P	Platinum	1.250	1.500	*	-----	0.0001
7P	Platinum	1.500	1.750	*	-----	0.0001
8P	Platinum	1.750	2.000	*	-----	0.0001
9P	Platinum	2.000	2.250	*	-----	0.0001
10P	Platinum	2.250	2.750	*	-----	0.0002
11P	Platinum	2.750	3.250	*	-----	0.0002
1A	Alumina	0.060	0.250	**	-----	0.0022
2A	Alumina	0.250	0.470	**	-----	0.0026
3A	Alumina	0.470	0.750	**	-----	0.0033
4A	Alumina	0.750	1.000	**	-----	0.0030
5A	Alumina	1.000	1.250	**	-----	0.0030

Table 20 (cont.)

6A	Alumina	1.250	1.500	**	----	0.0030
7A	Alumina	1.500	1.750	**	----	0.0030
8A	Alumina	1.750	2.000	**	----	0.0030
9A	Alumina	2.000	2.250	**	----	0.0030
10A	Alumina	2.250	2.750	**	----	0.0059
11A	Alumina	2.750	3.250	**	----	0.0059

* Adjusted platinum wire diameter was 0.0226 in.

** Adjusted alumina dimensions were 0.125 in. od x 0.0226 in. id.

Table 21. Calculated Data Using Three-Dimensional Model

Time, sec.	Temperature °F							
	1	2	3	4	5	6	7	8
0	818	818	818	818	818	818	80	80
1	818	817	766	736	816	817	684	649
2	818	814	775	754	810	814	752	722
3	818	805	784	764	802	806	774	741
4	817	793	790	764	792	797	783	741
5	817	775	794	757	779	783	785	733
6	815	757	795	749	766	769	785	724
7	812	739	795	725	754	757	782	704
8	808	713	794	712	734	737	777	671
9	802	692	790	694	712	715	767	651
10	794	679	786	683	699	701	761	643
11	785	639	780	673	690	692	753	626
12	775	621	773	636	659	666	743	594
13	763	620	765	621	622	626	731	580
14	750	620	757	621	621	621	720	576
15	737	551	747	620	621	621	711	572
16	723	521	737	597	620	620	702	561
17	708	507	728	580	620	620	691	545
18	693	486	719	571	582	583	677	535
19	678	461	708	559	568	570	661	526
20	663	445	696	530	562	562	649	519
21	685		547	557	558	642	523	
22	634		677	544	552	553	637	509
23	625		669	541	549	550	633	506
24	621		661	534	541	542	620	502
25	593		652	527	535	536	612	496
26	496		644				607	
27	451		638				603	
28	420		633				598	
29	396		628				593	
30	376		622				587	
32			621				583	

Table 21 (cont.)

34	621	574
36	620	564
38	570	541

1. Centerline, no thermocouple.
2. 0.31 in. from centerline, no thermocouple.
3. Centerline, with thermocouple.
4. 0.31 in. from centerline against thermocouple.
5. 0.31 in. from centerline, 90° from thermocouple.
6. 0.31 in. from centerline, 180° from thermocouple.
7. Platinum at centerline.
8. Alumina at 0,31 in. from centerline.

Table 22. One-Dimensional Calculations for Lead
Cast in a 890S Graphite Mold

Time, sec.	Temperature, °F	
	1	2
0	842	818
2	842	818
4	839	817
6	828	816
8	803	811
10	765	803
12	719	792
14	671	779
16	622	765
18	621	752
20	411	738
22	345	725
24	309	713
26		701
28		690
30		680
35		658
40		641
45		629
50		622
55		620
57		537
59		504

1. One-dimensional slice of 2-D model.
2. One-dimensional model using low thermal conductance values.

Table 23. One-Dimensional Calculations for Lead Cast
in a CS-312 Graphite Mold

Time, sec.	Temperature, °F					
	*	1	2	3	4	5
0		818	818	818	818	80
2		818	818	809	590	186
4		817	816	785	585	173
6		816	811	758	582	172
8		811	801	734	582	177
10		802	788	717	582	183
12		790	774	703	579	190
14		777	760	688	563	188
16		764	746	675	562	189
18		751	732	665	562	193
20		737	719	657	559	197
22		724	707	634	551	196
24		712	695	638	548	197
26		700	684	634	547	200
28		690	675	623	545	202
30		680	666	621	536	205
32		670	657	620	528	206
34		662	650	611	523	209
36		654	644	591	516	211
38		647	738	586	508	213
40		641	633	573	502	215
42		636	629	561	495	217
44		631	625	551	487	219
46		628	623	539	479	220
48		624	621	529	472	222
50		622	619	517	464	223
52		621	579	505	456	225
54		621	559	494	448	226
56		616	542	483	440	227
58		531	514	472	432	228
60		502	491	460	424	229

Table 23 (cont.)

- * 1. Centerline (node 1).
- 2. Third node (0.122 from centerline).
- 3. 0.5r (node 7).
- 4. Metal/mold interface (node 10).
- 5. Mold 0.060 in. from interface (node 13).
- 6. Mold 0,375 in. from interface (node 15).

Table 24. Calculated Temperatures Along the Axis of Symmetry of Steel Casting

Distance, in. *	Temperature, °F		
	10 min.	30 min.	60 min.
.28	2761	2751	2736
.85	2764	2757	2725
1.41	2766	2758	2699
1.98	2766	2754	2676
2.55	2764	2738	2655
3.11	2763	2706	2633
3.68	2757	2670	2610
4.24	2723	2634	2584
4.81	2656	2596	2558
5.37	2588	2551	2525

* Distance measured from the inside corner.

Table 25. Experimental Data for Steel Casting Along Axis of Symmetry

Distance, in.	Temperature, °F		
	10 min.	30 min.	60 min.
1.0	2750	2735	2680
2.0	2750	2720	2655
3.0	2745	2685	2620
3.6			2605
4.0	2725	2650	

* Data from graph presented by Bishop, Brandt, and Peilini.²⁰

Table 26. Calculated and Experimental Data at Centerline
and Axis of Symmetry for Steel Casting

Time, min.	Temperature, °F	
	Calculated	Experimental *
0	2875	
10	2764	2745
20	2758	2725
30	2738	2705
40	2709	2680
50	2684	2660
60	2655	2635

* Data from graph presented by Bishop,
Brandt, and Pellini.²⁰

APPENDIX H
COMPUTER-GENERATED COLOR SLIDES

Table 27. Color Codes and Their Maximum Temperatures

Code	Color	Temperature, °F		
		Lead	Steel	Uranium
101	Blue	100	200	1652
202	Light blue	225	550	1742
112	Deep magenta	350	1500	1832
001	Magenta	450	2100	1922
012	Medium magenta	550	----	2012
002	Light magenta	620.5	2723	2070
011	Red	---	2763	----
121	Dark yellow	650	2800	2102
021	Yellow	725	2840	2174
020	Light yellow	800	2880	2246
111	White	>800	>2880	>2246

Table 28. Key to Slides

1. Colors plotted by COLSTOR and their codes.
2. Uranium rod after 6 seconds.
3. Uranium rod after 18 seconds.
4. Uranium rod after 27 seconds.
5. Uranium rod after 30 seconds.
6. Lead rod after 2 seconds.
7. Lead rod after 5 seconds.
8. Lead rod after 8 seconds.
9. Lead rod after 15 seconds.
10. Lead rod after 23 seconds.
11. Steel casting after 30 min. using thermal conductivity curve "c".
12. Steel casting after 60 min. using thermal conductivity curve "c".
13. Steel casting after 30 min. using thermal conductivity curve "d".
14. Steel casting after 60 min. using thermal conductivity curve "d".
15. Steel casting after 10 min. using thermal conductivity curve "f".
16. Steel casting after 20 min. using thermal conductivity curve "f".
17. Steel casting after 30 min. using thermal conductivity curve "f".
18. Steel casting after 40 min. using thermal conductivity curve "f".
19. Steel casting after 50 min. using thermal conductivity curve "f".
20. Steel casting after 60 min. using thermal conductivity curve "f".

Figure 28 consists of the twenty 35 mm
color slides listed in Table 28;
they are not reproduced for this report.

Figure 28.
Color Slides.