PHYSICAL AND NUMERICAL SIMULATION OF TURBULENT RECIRCULATING FLOWS IN MATERIALS PROCESSING OPERATIONS

by

ASHOK MURTHY
V.I

B. Tech., Indian Institute of Technology, New Delhi
(1978)
M.Sc.E., University of New Brunswick, Fredericton, Canada
(1981)

SUBMITTED TO THE DEPARTMENT OF MATERIALS SCIENCE AND ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June 1984

Signature Redacted

Signature of Author ______________________________
Department of Materials Science
and Engineering
May 4, 1984

Signature Redacted

Certified by ______________________________
Julian Szekely
Thesis Supervisor

Signature Redacted

Accepted by ______________________________
B. J. Wuensch
Chairman, Department Graduate
Committee

PHYSICAL AND NUMERICAL SIMULATION OF TURBULENT RECIRCULATING FLOWS IN MATERIALS PROCESSING OPERATIONS

by

Ashok Murthy

Submitted to the Department of Materials Science and Engineering on May 4, 1984 in partial fulfillment of the requirements for the Degree of Doctor of Science in Materials Engineering.

ABSTRACT

An experimental technique has been developed to make detailed measurements of flow and turbulence characteristics in a heated melt of molten Wood's Alloy using a completely microcomputer controlled data acquisition and processing environment.

A mathematical model has been developed to represent the flow in the melt starting with calculation of the electromagnetic force field and using the $\kappa - \epsilon$ model of turbulence.

An alternate turbulence model, the XI model, has been developed as part of this work. The new model is computationally simpler and uses fewer semi-empirical constants than the $\kappa - \epsilon$ model. Comparison of predictions made using the new model with those made using conventional models and also with the experimental measurements is made. The XI model is shown to predict mean flow and turbulence values very close to those predicted by the $\kappa - \epsilon$ model. Measured values for mean velocities and turbulent dissipation are shown to be reasonably predicted by both the $\kappa - \epsilon$ and the XI models.

A parameter, PI, is proposed for evaluation of the onset of turbulence in a melt and as an estimate of the degree of turbulence.

A theory , based on fundamental considerations of turbulence, is proposed for explaining the relationship of vessel mixing times with parameters like stirring power input and vessel size. This theory is shown to agree favourably with both experimental and in-plant measurements as quoted in references.

Thesis Supervisor : Dr. Julian Szekely
Title : Professor of Materials Engineering

TABLE OF CONTENTS

Appendices

LIST OF FIGURES

Number Page

ACKNOWLEDGEMENTS

I would like to thank Professor Julian Szekely who through his remarkable intuition and farsightedness encouraged me to seek out and pursue new and different approaches. I am also grateful to him for creating a working environment that made my MIT experience both fruitful and enjoyable.

I would also like to extend my gratitude to Mr. Taewook Kang for his invaluable help in creating the microcomputer software scheme and to Drs. Nagy El Kaddah and Manoj Choudhary for their guidance and help on innumerable occassions.

I would like to thank Professors T. B. King and T. W. Eagar for their valuable comments.

I am always indebted to my wife Banani, my daughter Amita and to my family in India for their understanding and cooperation in countless ways.

Financial support for this work from NASA is gratefully acknowledged.

1. INTRODUCTION

In recent years there has been a great increase in the efforts to move towards a reliable description of materials processing operations. This effort is driven, to a large extent, by a desire to automate processes using the now relatively inexpensive computer hardware as "intelligent" controllers. The first step towards computer automation is an accurate mathematical description of the dynamics of the flow and heat and mass transfer characteristics of the system.

This work deals with the physical and numerical modelling of an electromagnetically stirred liquid metal system. The stirring is the result of a force field created by passing a current through the melt between two concentric electrodes, one of which forms the base of the melt while the other dips into the melt from above. This configuration occurs in several processes of industrial importance such as Electro Slag Refining (ESR), Electro Slag Welding (ESW), Submerged Arc Furnace, among others. Many of the problems addressed in this work are, however, more general and consequently many of the results may be applied to a wide range of materials processing operations.

It is felt that the major contributions of this work are:

(1) For the first time, detailed and careful measurements of flow and turbulence parameters have been made in an applied current system using a liquid metal that is solid at room temperature.

(2) A new turbulence model, with an intuitive fundamental basis and a shorter computation time, has been developed and the results compared with existing models and with experiments.

(3) A theory based on fundamental turbulence principles has been developed to explain the dependence of mixing time in metallurgical systems on various operating parameters.

(4) Numerical techniques have been developed to evaluate and correct for the use of thermal anemometers in liquid metals in the presence of magnetic fields and probe surface contamination.

A final objective that this work leads to is the implementation of on-line computer based supervisory process control systems which, it is felt, will become increasingly prevalent in the metals processing industry in the years to come.

Regarding the organisation of this thesis, the material has been divided into six chapters. The Literature Survey is presented in Chapter 2 which reviews work on Thermal

Anemometry, particularly in liquid metals and some experimental and mathematical modelling work on electromagnetically agitated systems and on the theory of turbulence.

Chapter 3 describes the experimental setup and some of the considerations involved in making the measurements.

The formulation and implementation of the numerical models is presented in Chapter 4. Three different models have been used to model the turbulent recirculating fluid flow situation. The first model is a widely used version of the two equation model of turbulence called the k-epsilon model. The second is a low Reynolds number version of the same and the third is a new model developed during this work.

In Chapter 5 the experimental measurements and the numerical predictions are compared and discussed.

The concluding remarks and some suggestions for further work in this and in related areas are made in Chapter 6.

2. LITERATURE SURVEY:

In this chapter, literature pertaining to the use of thermal anemometry in liquid metals, to mathematical and physical modelling of electromagnetically driven flows and to turbulent flow phenomena will be reviewed.

2.1 Velocimetry

There are many kinds of techniques available for velocity measurement in fluids(86) . However, the nature of the experiment imposes several constraints on the types of schemes that may realistically be considered to be possible candidates. The fact that the working fluid is an opaque liquid metal rules out optical techniques like Laser Doppler Anemometry. The fact that relatively high frequency turbulence measurements are required rules out the various forms of Pitot Tube based methods.

There is one instrument whose development and application for measuring turbulent flow have far outstripped those of other instruments up to now, namely, the Thermal Anemometer. The hot wire/film anemometer has been very widely used for measuring turbulent air and water flows and is now a well developed technique for such applications. Its use in measuring liquid metal flows is, however, less well developed. Almost universally, the liquid metal has been liquid mercury at room temperature, with the exception of Murthy and Szekely(3) who used molten Wood's Alloy at 120° C. A large and comprehensive survey of literature pertaining to the use of Thermal Anemometry is presented in (107) .This listing contains use of the technique both in liquid metals as well as in more conventional fluids. No effort will be made here to review the use of anemometry in conventional fluids. A sampling of the work pertaining to its use in liquid metal flows is presented in an attempt to point out some of the problems encountered and the solutions adopted by the researchers.

An initial problem regarding the use of thermal anemometry in metals was the fact that the fluid is electrically conducting hence the sensor had to be coated with a film that was both electrically insulating as well as thermally conducting. The pioneering work in developing a practical design of such a probe was done by Sajben(12,13) who used a lacquer coated hot wire. In his work Sajben pointed out another serious problem with using hot wire probes in mercury namely one of calibration drift. This problem was acknowledged by most of the other workers in the field(1,3,7,9,11-13,15,16,18,38) . Several techniques were proposed by various researchers to overcome this problem.

(1) The drift was observed to occur when the probe crossed the metal-air interface. So the obvious remedy, and one adopted by many of the workers(1,7,12,13,16,38), was to provide for in-situ calibration in the same vessel where the experiment was to be done, so that the probe never had to be lifted out of the metal pool.

(2) In 1969 Hoff(18) suggested, based on his experiments, that coating the hot film probe with a sputtered layer of gold or copper improved its wetting characteristics and hence its drift behaviour. However Hurt and Welty(11) pointed out that it was difficult to control the uniformity of thickness of the sputter coating and in general concluded that sputter coating was not worth the effort involved because it did not yield any consistent improvement and in fact was found, in some instances, to reduce the life of the probe.

(3) The third approach has involved a heat transfer analysis of the probe, attempting to compensate the fouling of the probe by taking a reading at zero velocity and then subtracting out the effects of the fouling(1,12,13,3).

Another problem associated with liquid metal anemometry was related to the low Prandtl Number of metals. This in fact spawned two, somewhat related, problems. As Hill(10,14) pointed out, the probe in liquid mercury showed poor directional sensitivity except when the Peclet Numbers were very high. This was due to the fact that the thermal boundary layer became more or less uniform around the probe. This observation implied difficulties in the use of cross wire probes to measure flow direction.

A second problem linked to the thick boundary layer was one of reduced frequency response due to inertia effects. Lim et. al.(6,8) suggested theoretically calculated high frequency response limits beyond which use of uncompensated outputs from turbulent liquid metal systems may yield erroneous results.

Use of heated liquid metals that are solid at room temperature holds out the possibility of studying solidification problems as linked to fluid flow. Murthy and Szekely(3) used modified hot film probes in molten wood's alloy at 125° C. They too encountered the problem of calibration drift which was tackled by using mathematical transformation techniques mentioned earlier.

While most of the workers have concentrated on the problems of anemometry per se in idealised flow situations, some have employed the technique as a practical tool to investigate flows of metallurgical interest(1,16) .

Ricou and Vives(2) have employed a completely different technique using an incorporated magnet probe to measure velocities in melts up to 720° C. This technique is applicable in the absence of external magnetic fields and in general cannot be used for turbulence measurements. Also, this very innovative technique is not yet commercially available.

The question of what effect an external magnetic field would have on the performance of a hot film anemometer probe is a very relevant one and has been examined in greater detail in a subsequent section of this chapter.

2.2 Mathematical and Physical Modelling

In this section work aimed at modelling certain types of electromagnetically driven flows is reviewed. In 1970 Shercliffe(26) studied the case of a point electrode dipping into a semi-infinite pool of inviscid fluid. An analytical solution was developed for this system with a direct current being passed between the point electrode and another electrode set at infinity. Later Souzou and Pickering(27) extended the analysis for a finite sized current source in a hemispherical metal pool of finite viscosity. Both these models, though very important from a fundamental standpoint, were rather drastic oversimplifications of the real situation and results could not be extrapolated to metallurgical systems easily. The field was, in a sense, open for the numerical analysts to try and develop a more meaningful description of such systems.

In 1975, Szekely and Asai(20,21) established a fully numerical simulation of the turbulent flow and transport in the liquid phase of a continuous caster.

Later, Szekely and Chang(24,25) established a numerical calculation for turbulent flow in an induction furnace. Their results for mean velocities were compared with laboratory experiments in a Wood's Alloy system(23) . Reasonable agreement (within 30%) was found for mean velocities. Moore and Hunt(19) studied turbulent flow in an inductively stirred mercury pool and established measured profiles for turbulence and for mean velocities using a drag probe. Trakas et. al.(1) using hot film probes measured spectra in an inductively stirred vessel.

Along somewhat similar lines Szekely and Dilawari(22,30,32,33) and Schwerdtfeger and Kreyenberg(29) made calculations for temperature profiles, heat transfer and turbulent velocities for the case of the Electro Slag Refining (ESR) system. This was followed up by Choudhary(36) and Choudhary and Szekely(34,35,39) who included an automatic calculation of pool profiles in the ESR solidifying ingot on the basis of their heat transfer calculations.

Experimental verification of these calculations has been somewhat less studied. Choudhary(36) reported on comparison of measured pool profiles with his calculations. More recently, Choudhary et. al.(37,31) reported on comparisons of their calculations with measurements of mean velocity done on the meridional half plane for an applied current mercury system. High speed photography of particle motions on the surface of the mercury yielded somewhat rough measurements of mean velocity. A similar technique was used by Kompan et. al.(40) and by Butsenieks et. al. (43,41) in a mercury system of related but not identical geometry. Moshnyaga and Sharamkin(42) used pitot tube velocimetry to measure the centerline velocities and pressure field in an electromagnetically stirred vessel. Comparison of measurements with a laminar flow numerical simulation was also reported by Butsenieks et. al.(41) .

On a considerably more fundamental level, Alemany et. al.(38) reported on measurement of turbulent M H D flow spectra in a behind-grid mercury turbulence with an applied magnetic field.

2.3 Turbulence Modelling and Theory

This section relates to the modelling of turbulence phenomena in general. Even though the turbulence model would form an essential component of a mathematical model for the whole process, it has been dealt with separately since many works on turbulence address the idealised problem without much emphasis in the mechanics of the overall flow.

Beek and Miller(77) extended some preliminary concepts on the effects of turbulent transport on mixing and reaction rates in chemical reactors. Corrsin(67,68) developed expressions relating mixing time for passive scalar contaminants in a turbulent flow to microscales and thence to the stirring power input to the system for the general case of arbitrary Schmidt Number. Brodkey(69) applied this analysis to the case of turbulent mixing in a pipe flow and found good agreement between the measured intensity of contaminant segregation and the values predicted using Corrsin's analysis. Brodkey(70,89) put forward very physical and intuitively appealing description of the mechanism of contaminant dispersion due to turbulence and some calculations for parameters like mixing time and segregation based on study of the turbulence spectrum.

Numerical calculation of turbulence can be said to be comprised of the following approaches:

a) Ad-Hoc viscosity models

b) One Equation models

c) Two Equation models

d) Stress Transport models

e) Subgrid Scale Closure schemes and Spectral closures

a) and b) are the simplest schemes that are generally considered insufficient to study recirculating flow systems. Two equation models are currently the most popular for recirculating flow situations. These rely on using two parameters to define the state of turbulence in the field. Differential transport equations are written for these two parameters. The choice of parameters for the two equation calculation scheme can yield different models. Of these, the κ -w and the κ - ϵ models have been the most well known though the latter has recently been the more widely used of the two. Spalding(65) first used the κ -w model to predict concentration fluctuations in an axisymmetric gas jet. Rodi(91) , Launder and Spalding(84) and Rodi and

Spalding(64) , among others, described the use of the κ - ϵ model of turbulence in general two-dimensional flow situations. This model was in fact widely used in most of the numerical work on the simulation of metallurgical flow systems as described in a previous section(20-25,29-37,39) and this is among the best documented of all turbulence models.

The stress transport models(74,75) use a higher order closure for the turbulence problem resulting in more differential equations to be solved. These models represent the next level of sophistication of turbulence modelling but are presently not very popular because of their greater demands on computer time and general complexity.

Representing the highest level of numerical attack on the turbulence modelling problem are the Spectral Closure schemes, primarily proposed by Orszag and his associates(63,71,72) and the Subgrid Closure schemes(73,76) which use similar unsteady state schemes but only for flow fluctuations above a certain minimum cutoff eddy size. These methods rely on solving an usteady form of the Navier-Stokes equations with a very short time step of computation so as to be able to simulate the evolution of the turbulence flow patterns. These techniques are presently used primarily for meteorological and geophysical flows and represent considerable investment in terms of computer costs.

In 1945, Lotsianskii(66) evolved the mathematical concept of a "Disturbance Moment", later known as the Lotsianskii Integral or Invariant, which he showed to be a constant in a homogeneous, isotropic flow and determined by the initial disturbance given to the fluid. Proudman and Reid(61) and Batchelor and Prouodman(62) and Batchelor(95) showed that the so called invariant could not truly be a constant but rather a slowly varying function over a flow domain. Comte-Bellot et. al.(80) interpreted this property of slow variance to imply a value that was effectively a constant and applied this invariant to study the behaviour of turbulence decaying behind a grid. They showed that this assumption resulted in very reasonable decay laws for grid turbulence. Hinze(86) and, to a certain extent, Tennekes and Lumley(88) used this together with information about other ranges of a 3-D spectrum to patch together a complete picture of the shape of the isotropic spectrum of turbulence.

Part of this present work has been to use an understanding of spectral shapes, based on the Lotsianskii invariant, together with numerical techniques to evolve a new scheme of calculation which would fit somewhere between a 1-equation and a 2-equation model of turbulence.

Effects of magnetic fields on turbulence and on a related problem, namely hydrodynamic stability, have been widely studied. Since, in this thesis, the magnetic field strength was judged too weak to have much effect on the turbulence, this part of the literature survey is appended for the sake of completeness. References 44-53, 38 represent some of the work related to this topic.

2.4 Heat and Mass Transfer from a Cylinder in MHD flow

Thermal anemometer probes when used in the presence of a magnetic field, either aligned with the probe axis or orthogonal to it, experience a change in heat transfer characteristics (ie. a change in the overall Nusselt Number dependence on flow velocity)(15,38,99)

Literature surveyed in this section served as a foundation for the numerical calculations that were done in this work to evaluate the influence of a magnetic field on the heat transfer characteristics of a hot wire or hot film probe in a liquid metal flow.

Considerable experimental work has been done in relating the imposition of a magnetic field to the heat transfer from a hot wire/film probe by Lykoudis et. al.(99) Numerical and analytical type of solutions were explored since such approaches can be extrapolated to different geometries and sizes easily and also gives a more detailed insight into the mechanics of the problem. Regarding numerical solution of the problem, Kalis et. al.((96) considered the effects of a transverse magnetic field on the pressure profile and drag coefficient around a cylinder but did not consider the closely related problem of heat and mass transfer from the cylinder. In addition, as will be shown later, their use of a rather coarse grid near the surface resulted in a markedly weaker dependence of pressure profiles and drag coefficients on the magnetic field.

The analogous heat transfer problem was tackled by Blum(97) who used an approximate analytical approach transforming the results obtained for a flat plate boundary layer flow to a circular cylinder.

Chester(103) approached analytically the problem of the influence of a magnetic field on flow around a sphere but only in the Stokes' flow regime. Soundalgekar et. al.(98) have done numerical calculations for the case of a transverse magnetic field on a flat plate boundary layer. The general hydrodynamic problem of the numerical analysis of fluid flow around cylinders (albeit in the absence of magnetic fields) has been extensively studied. References 100-102 are a few of the works in this area. Ishiguro et. al.(104) and Grosh and Cess (105) have reported on experimental measurements of overall heat transfer from a cylinder as related to the flow of liquid metal orthogonal to the cylinder axis. In addition, the books by Hughes and Young(87) and by Roache(85) contain a wealth of valuable information on the subjects of magneto fluid dynamic modelling and numerical fluid dynamics respectively.

The task to be addressed in this work relates to the measurement and numerical representation of an electromagnetically driven turbulent flow. The literature reviewed here provides a basis for setting up and understanding such a system.

3. EXPERIMENTAL MEASUREMENTS

3.1 Introduction

In essence the experiment involved the setting up of a turbulent recirculating flow of liquid metal stirred by passing a direct current through it between two electrodes. The measurements consisted of taking velocity readings at various positions inside the melt and analysing the instantaneous velocity for mean and turbulent components. The anemometer itself, (the principle of operation of which is described in Appendix A) was calibrated using a calibration tow tank. The entire scheme of measurement was completely computer controlled. A brief description of the principal components of this experimental scheme follows. A listing of some of the equipment used in the experimentation, together with the manufacturers' name and location, is given in Appendix D.

3.2 The Main Flow System

The choice of the liquid metal to be used as the principal fluid constituted a major decision and one that had profound effects on the design and selection of equipment. The metal chosen was Wood's Alloy (see Appendix B) Despite the obvious problems associated with operating the vessel and the measurement schemes at higher temperatures, this choice was made because the use of a metal that was solid at room temperature opened up possibilities of studying solidification linked to fluid flow. This option would clearly not be available if mercury, the generally popular choice, were used. The other advantage was that Wood's Alloy was relatively non-toxic.

The containment vessel consisted principally of a cylindrical copper cylinder (15 cm dia., 10 cm. height) made from 1/16 inch thick copper sheet and brazed together (see Fig. 3.1). The base electrode was made from 1-1/2 inch thick copper plate that was cut and then turned to final shape. The electrode assembly itself was made in two parts, one being the body of the electrode which covered most of the base of the vessel and was fabricated from the plate as described earlier. The electrical connection was made by a threaded pin which could be screwed into the base plate. This pin provided at the other end a connection point which was turned to the specifications required to match the end of the current carrying cable of the welding generator. The inner walls of the vessel were painted over with Epo-tek H-72 epoxy to prevent current passing through the cylinder walls instead of through the melt entirely. On the outer surfaces of the vessel, a tape heater was fixed using ceramic cement. This tape heater was made in two parts- one for the cylindrical (vertical) walls of the vessel and the other for the base. The electrical power supply for the two heaters was kept under separate control in order to provide greater flexibility in controlling the melt temperature. The side wall heater was controlled by a proportional temperature controller while the base heater was manually set by potentiostat to create approximately isothermal conditions in the melt. Outside the ceramic jacket which held the electrical heaters was a layer of pyrex cotton wool. This entire assembly was placed inside a Plexiglas outer containment vessel which was held by three support rods above a heavy steel base plate. The three supports could be adjusted to keep the vessel and the metal pool horizontal. Holding the bottom of the vessel several inches above the base plate provided enough space beneath for a drain pot.

For periodic maintenance as well as for any future experiments which may need the vessel to be drained (perhaps to study solidification linked to fluid flow) the vessel was provided with a drain point. For this purpose, the bottom electrode was made hollow. Through this hole a slender pushrod was passed. A ceramic plug covered the point where the pushrod emerged from the base plate inside the melt to prevent any possible leakage. Draining could be started by tapping the bottom of the pushrod lightly to dislodge the ceramic plug which would then float to the top of the metal pool. The inner surface of the base plate was made slightly tapered towards the centrally placed drain hole to assist flow of the metal.

The depth of the metal pool was about 4 cms. The top electrode was 1.5 cms. in diameter and dipped into the melt from the Plexiglas upper plate.

Since the flow and turbulence field would be expected to vary spatially inside the melt, a system of accurate probe positioning was required. The positioning equipment was built on top of the upper plate. It consisted of a motor driven radial traverse. On the slider of the radial traverse was attached a probe holder arm which contained a manually operated axial traverse and a simple theta rotation device about the axis of the probe. Both the axial and radial traverse spindles were provided with 10 turn potentiometers which were connected to a digital display. A program was built in the micro-computer to take keyed-in display voltages and interpret them as r,x coordinates of the probe tip within it. The probe was never moved to a new location until the location was first plotted out and checked for proximity to walls or any solid object which might break the very fragile probe tip. The same program also provided a grid of potentiometer voltage settings. The operator then simply moved the traverses (radial and axial) until the approximate voltages appeared on the display.

A sketch of the data flow path for the data relating to the main flow system is given in Fig. 3.2. At the time of the experiment, no attempt was made to analyse or to process the information. The anemometer voltage outputs were continuously recorded on one of the channels of an F.M. tape recorder whose gain factors were set from experience to permit even the highest voltages encountered to be recorded faithfully. The voice memo channel on the recorder was used to make comments about the progress of the experiment and also to note the time of starting and stopping the current and so forth. Time (using a stop watch which was started exactly at the same time the current was switched on) and experiment number was noted on the memo channel at regular intervals to facilitate subsequent playback and analysis. This technique provided a permanant record of the experiment for future analysis and also freed the experimenter from the tasks of operating analysis instrumentation to concentrate on carrying out an experiment which could be potentially dangerous.

The subsequent post-processing of the data took the path as shown in Fig. 3.2. Each tape was played back at least twice, once to get the mean values of voltages for each experiment and once to do a spectral analysis. The considerations involved in making these two measurements are described in Appendix B. The structure of the micro-computer based data acquisition and information handling scheme is described in Section 3.4.

3.3 Calibration System Design

Fig.3.3 shows a sketch of the hot film anemometer calibration assembly, which in essence consisted of a shallow, externally heated, horizontal trough, containing molten Wood's Alloy at 120° C. Provision was made to drag a hot film probe, immersed in the metal, parallel with the axis of the trough at a predetermined velocity. This velocity of traverse ranged from about 1 to 10 cm/sec.

This arrangement is quite similar to that used in the calibration of hot film probes in mercury(12-14), except for the higher operating temperatures involved.

The trough was made from a semi-cylindrical section of 1-1/2 inch copper pipe brazed to two end plates. the cylindrical section of the trough was heated by a semi-cylindrical resistance heater which had an internal diameter equal to the outer diameter of the pipe. The power to this heater was regulated by a temperature controller. Temperature measurement for the proportional controller was provided by a thermocouple dipping into the pool of metal. A digital display of the temperature was also provided. Movement of the slider assembly was controlled from a remote switching unit and controller. To allow any initial transients to settle down when the probe first started to move, the measurement was started only after the slider had already moved several centimeters. Data acquisition was started (see Fig.3.4) when the slider tripped on a microswitch and stopped when the microswitch tripped off a fixed distance later. This also enabled a velocity measurement as the computer was programmed to measure the 'ON' time of the microswitch. When sampling was over, the velocity as well as the output voltage was printed out by the micro-computer. The rate of sampling could be set in the program. The logic of the data acquisition software and the information handling schemes are detailed in a subsequent section.

3.4 Microcomputer Usage Scheme

An Apple II Plus microcomputer was used extensively as a laboratory tool. The use of an in-house microcomputer to supervise the entire experimentation scheme held out several advantages, not the least of which were speed, repeatability, convenient user interfacing, convenient data handling and also the fact that analog signals needed to be digitised before they could be reduced to any usable form. Functionally, the tasks performed by the microcomputer could be classified as:

(1) Signal digitisation for both the main system signals and for calibration experiments.

(2) Preliminary computations like drift correction, scaling etc.

(3) Information handling involving data file handling and creation, mainframe communications etc.

(4) Secondary help functions like printing, plotting, interactive human interface, plot digitisation etc.

Many of the program segments performed more than one of the above functions. It was felt that looking at the scheme of microcomputer usage as a step by step sequence performed in a typical experiment would aid in clarifying a somewhat complex chain of interlinked program steps.

Fig. 3.5 shows the Apple II based program files which performed the various operations in a partly sequential manner, while Fig. 3.6 shows the data files that resulted at the end of each segment. A listing of some of the software developed to create this scheme is provided in Appendix E.

These programs are the result of the joint efforts of Mr. T. Kang and this author. The experiments can be divided into three distinct but interlinked data flow paths which are simultaneously processed except at points where data cross linking occurs, in which case a definite priority exists.

Path I- Calibration data flow

The microcomputer is connected real time with the calibration arrangement. The anemometer voltage output is led to channel 1 of the Analog-Digital Converter (ADC) and the microswitch is led to channel 5.

LEVEL 1(P1L1)

The operating environment is provided by routine DATACQ1 (in BASIC) and HOT1.OBJ (in Machine Code). HOT1.OBJ was the machine code routine which did the actual data acquisition and stored the new data into specific memory locations. It was called, as a subroutine, by the control routine, DATACQ1 which served as a user interface to input pertinent parameters (like probe location, current etc.) and to set the operating conditions within which HOT1 must operate.

LEVEL 2(P1L2)

Routine CALFUL1 (in BASIC) reads the specific memory locations where the data has been stored by the Level 1 routines and computes velocity of probe traverse and average output voltages during the run. These values are then

(1) Stored as a disk file under the generic name CALn/m where n represents the probe number and m the calibration number for that particular probe.

(2) Output to the terminal for viewing

(3) Output to printer for hard copy

The routine returns control to Level 1 (P1L1) for another run.

LEVEL 3 (P1L3)

Routine DRIFT-CALC reads the CALn/m file (once all data points are completed), performs computations to correct the calibration data for drift (described in Section 3.5.3) and stores the corrected values in disk file under the name CALn/m-*

LEVEL 4 (P1L4)

Routine POLYFIT reads the data in data file CALn/m-* and fits a fourth order polynomial through it. The coefficients are then stored under the generic name of CALn/m-*-COEF on disk file. This is the final process step in Path-1 and CALn/m-*-COEF is a data file read by Path2 for processing mean velocities. Thus P1L4 represents an outgoing transfer point.

At this level the calibration curve can be plotted out on the local plotter for viewing, though this is not a necessary step. (See Fig. 5.43)

Path 2- Main Flow System Data Path (Mean Values).

The microcomputer is used in an off-line mode and is used to read data from the taped signals recorded during the experiment.

LEVEL 1 (P2L1)

The environment here is identical to that provided for P1L1. The microswitch is now replaced by a handheld switch for switching on and off the digitisation routine. Operating parameters like rate of sampling or memory allocation may need to be changed according to the needs of the experiment.

LEVEL 2 (P2L2)

Routine SYSFUL1 reads the memory locations for raw data and stores the values of voltages, as well as pertinent data input by the user, in a data file with the generic name SYS-nx where nx stands for the experiment number.

LEVEL 3 (P2L3)

Routine INTERPRET reads the P1L4 data file (CALn/m-*-COEF) as well as the SYS-nx data file and converts the measured voltages to velocities and also obtains the gradient of the calibration curve at that velocity. This information is stored in file set SYSINT-nx. This stage serves as an outgoing transfer point for Path-2

Path 3- Main Flow System Data Path (Turbulent Values)

LEVEL 1 (P3L1)

The digitisation and spectral analysis is performed off-line from taped recordings using a HP 5423 A Spectrum Analyser. This level results in an output in the form of a spectral plot on paper.

LEVEL 2 (P3L2)

Routine SPECTSV is used to digitise the spectral plot. This is done by mounting the plot on the plotter which is then driven along the curve to be traced, by keyboard control. The coordinates of each point on the curve, along with the necessary data for scaling, are saved in file SPEC-nx

LEVEL 3 (P3L3)

Routine SPECTAN reads data file SPEC-nx and computes values corresponding to turbulence energy and dissipation by numerical integration. These values are saved in file SPECINT-nx.

LEVEL 4 (P3L4)

Routine COMBINE reads file set SPECINT-nx and also SYSINT-nx (from P2L3) to scale the values of the energies calculated in P3L3 from voltage to velocity based values. The mean velocity (from SYSINT-nx) together with the scaled final values of turbulence energy and dissipation energy along with other pertinent data are stored in data file SYSLAST-nx.

SYSLAST-nx is the final data file which contains only the completely processed experimental results. This file can be either printed locally or transferred to the mainframe computer (IBM-370) using the communication routine T1.

3.5 Hot film anemometry in liquid metals

The use of hot film probes in Wood's Alloy at 120° C posed some definite problems with the sensor itself. Some of the more important problems are described here as well as the steps taken to alleviate them.

The two most important problems were those of probe failure due to the extremely corrosive nature of the heated metal and of the calibration drift with time.

3.5.1 Probe Life

The temperature of the melt was the single most important reason for failure of the probe. A typical probe as supplied by TSI (without modifications) would survive about 5-10 minutes in the melt. Damaged probes were cleaned carefully in an ultrasonic bath(80° C, slightly acidic) and then examined using a Scanning Electron Microscope. Over a period of time two failure mechanisms were identified:

(1) The epoxy coating used to insulate the legs of the probe softened at elevated tempeatures and quickly eroded away when subjected to a flow of metal.

(ii) The solder connections at the points where the platinum film was joined to the support legs could not withstand the temperature.

After consultations with TSI, the manufacturers of the anemometry equipment, both the epoxy insulation and the solder used at the joints was changed to higher temperature types. The changes have proved extremely satisfactory and probe life extended to at least 40 hours. Even after that time, probes were discarded due to insufficient sensitivity rather than due to failure. A precaution that has been found quite valuable in this regard has been to examine the probe under an optical microscope and touch up the epoxy coating at points where the coating may appear somewhat thin.

3.5.2 Calibration Drift

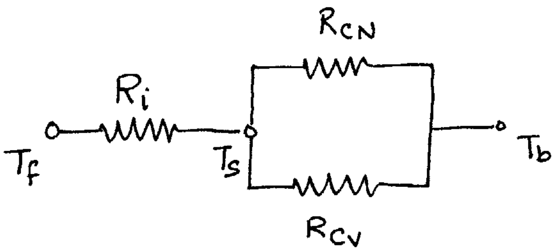
Drift in calibration characteristics was found to occur primarily when the probe was introduced into or removed from the melt. This appears to have been the experience of most other workers in this field while working with mercury. Some researchers (refer to Chapter 2 for details) have circumvented the problem by crossing the liquid metal -air interface only once during an experiment and by providing arrangements to calibrate the probe in-situ. Practical constraints with the present experimentation forbade this, so a technique was developed to correct for calibration drift using heat transfer analysis. The principal assumption of the analysis was that the drift was directly caused by the appearance of a coating of 'fouling' material (perhaps oxide) on the sensor surface, after Sajben(12,13) .

3.5.3 Drift Correction Analysis

The drift correction scheme was based on the fact that as the fouling of the sensor surface changed, this change was reflected in the output voltage produced by the anemometer at zero flow velocity (E0; with only natural convection present). Thus the scheme uses the value of E0, the zero-flow voltage, as an indicator that points to the extent of fouling. The entire scheme is computed automatically by the microcomputer and results in a new set of calibration points that are generated by resetting the E0 values for each case to a common base value.

The correction analysis involves three levels of correction, in decreasing magnitude of impact.

Heat is generated in the platinum film by resistance heating and conducted through the quartz coating and fouling material and thence convected away by the fluid. The overall heat transfer stages may be visualised by a simple thermal resistance network.



where:

T_f	is the film temperature
T_s	is the surface temperature
T_b	is the bulk temperature
R_i	is the thermal resistance of fouling material
R_{cn}	is the resistance due to natural convection
R_{cv}	is the resistance due to forced convection
Q	is the total heat flux rate
R_p	is the probe resistance
R_3	is the bridge resistance in series with the probe
E	is the voltage output of the anemometer

$\Delta T_m = T_f - T_b$

and $\Delta T_{eff} = T_s - T_b$

$$\Phi = \frac{E^2 R_P}{(R_3 + R_P)^2} = E^2 \beta \tag{3.1}$$

where

$$\beta = \frac{R_P}{(R_3 + R_P)^2} \tag{3.2}$$

Consider first the case when there is no flow. Then:

$$R_{cv} = \infty \tag{3.3}$$

and

$$\Phi_o = E_o^2 \beta \tag{3.4}$$

where E_o is the voltage output at zero flow velocity.

Now let:

$$R_{cN} \triangle \eta \, \Delta T_{eff}^{-1/4} \tag{3.5}$$

$$\Phi_o = \beta E_o^2 = \frac{\Delta T_{eff}}{R_{cN}} = \frac{\Delta T_{eff}}{\eta \, \Delta T_{eff}^{-1/4}} \tag{3.6}$$

or

$$\beta E_o^2 = \frac{\Delta T_{eff}^{5/4}}{\eta} \tag{3.7}$$

which gives:

$$\Delta T_{eff}^{5/4} = \eta \beta E_o^{2} \tag{3.8}$$

and therefore

$$\Delta T_{eff}^{1/4} = \left(\eta \beta E_o^{2}\right)^{0.2} \tag{3.9}$$

substituting in (3.5) gives

$$R_{cN} = \eta^{.8} \beta^{-.2} E_o^{-.4} \tag{3.10}$$

Now

$$\frac{\Delta T_m}{Q_o} = R_i + R_{cN} \tag{3.11}$$

or

$$R_i + \eta^{.8} \beta^{-.2} E_o^{-.4} = \frac{\Delta T_m}{\beta E_o^{2}} \tag{3.12}$$

introducing the two cases, one the active measurements (superscript (1)) and the other a reference case (superscript (ref)), it may be said that:

$$\Delta R_i = \frac{\Delta T_m}{\beta}\left(\frac{1}{E_o^{2(1)}} - \frac{1}{E_o^{2(ref)}}\right) - \eta^{.8} \beta^{-.2}\left(E_o^{-.4(1)} - E_o^{-.4(ref)}\right) \tag{3.13}$$

where

$$\Delta R_i \triangleq R_i^{(1)} - R_i^{(ref)} \tag{3.14}$$

Now consider the case when the flow velocity is non-zero. Then R_{cv} is a finite value. R_{cv} may be written as

$$R_{cv} = \chi V^{-.42} \tag{3.15}$$

where Υ is a constant of proportionality. Here

$$\frac{\Delta T_m}{\beta E_t^{2(1)}} = R_i + \frac{f_2 \cdot f_{1,1}}{f_2 + f_{1,1}} \qquad (3.16)$$

and

$$\frac{\Delta T_m}{\beta E_t^{2(ref)}} = R_i - \Delta R_i + \frac{f_2 \cdot f_{1,ref}}{f_2 + f_{1,ref}} \qquad (3.17)$$

where:

$$f_2 \triangleq \gamma \, \bar{V}^{.42} \qquad (3.18)$$

$$f_{1,1} \triangleq \eta^{.8} \beta^{-.2} E_0^{-.4(1)} \qquad (3.19)$$

$$f_{1,ref} \triangleq \eta^{.8} \beta^{-.2} E_0^{-.4(ref)} \qquad (3.20)$$

subtracting equation 3.16 from 3.17 yields:

$$\frac{1}{E_t^{2(ref)}} = \frac{1}{E_t^{2(1)}} + \frac{\beta}{\Delta T_m} \left\{ -\Delta R_i + f_2 \left(\frac{f_{1,ref}}{f_2 + f_{1,ref}} - \frac{f_{1,1}}{f_2 + f_{1,1}} \right) \right\} \qquad (3.21)$$

substituting for ΔR_i from equation 3.13 gives:

$$\frac{1}{E_t^{2}(ref)} = \frac{1}{E_t^{2}(1)} + \left(\frac{1}{E_0^{2}(ref)} - \frac{1}{E_0^{2}(1)} \right) + \frac{\beta}{\Delta T_m} \left\{ f_{1,1} - f_{1,ref} + f_2 \left(\frac{f_{1,ref}}{f_2 + f_{1,ref}} - \frac{f_{1,1}}{f_2 + f_{1,1}} \right) \right\} \qquad (3.22)$$

Equation 3.22is used to calculate values for Et $^{(ref)}$.

The voltages (E0$^{(1)}$ and Et $^{(1)}$) with superscript (1) are the values actually measured during an experiment. E0$^{(ref)}$ is the artificial value for base voltage (at zero velocity) selected for normalising the calibration curves. (In this work, $E0^{(ref)}$ has always been kept equal to 15.0 Volts.) $Et^{(ref)}$ is the set of normalised voltages which when plotted against the velocity provides a minimum drift calibration curve. This calculation is performed automatically by the computer when a new calibration is done. The resulting (normalised) calibration curve is fitted with a fourth order polynomial. This polynomial is used by other calculation stages in voltage to velocity and turbulent parameter conversions. Figs. 5.42 and 5.43 show a typical case of calibration drift before and after correction was applied.

3.6 Effect of Magnetic Fields on Calibration

A question that arose early in the experimental work was that how much does the presence of a magnetic field affect the heat transfer characteristics of the probe. Previous approaches to this problem have included calibrating the probes in the presence of magnetic fields(38), applying semi-emperical correction factors (15), or ignoring the effect of the field altogether. To get a better quantitative appreciation of the problem numerical calculations were made to predict the effect of a transverse magnetic field on the flow of a liquid metal past a heated, infinite cylinder (which represents the hot film probe). The geometry of the flow situation is shown in Fig. 5.36. Consider a circular cylinder of radius 'a' and of infinite extent in the z-direction. The cylinder surface is heated internally to a constant temperature T_f . The entire cylinder surface is subjected to a cross flow of a conducting liquid at a velocity U_o normal to its axis and a constant magnetic field of induction B_o The presence of the orthogonal magnetic field alters the velocity profile around the periphery of the cylinder and this in turn influences the heat, mass and momentum transfer characteristics, both local and averaged.

3.6.1 Mathematical Formulation

A mathematical formulation of the problem involves the following:

(1) Momentum balance (Navier Stokes) equations including a source term of electromagnetic origin.

(2) Maxwell's equations for a moving medium.

(3) Energy balance equations

The following assumptions were made:

(1) The fluid is Newtonian

(2) Property values such as viscosity, density, thermal conductivity etc. were assumed to be constant and evaluated at bulk fluid temperature T_{bulk} .

(3) The flow is considered stable and laminar

(4) The cylinder is an electrical insulator

(5) The cylinder is infinitely long.

The momentum balance equation including the Lorentz force term may be written as:

$$(\underline{U}\cdot\underline{\nabla})\,\underline{U} = \frac{\underline{\nabla} p}{\rho} + \nu \nabla^2 \underline{U} + \underline{j}\times\underline{B}$$

and the energy transport equation is given by:

$$(\underline{U}\cdot\underline{\nabla})\,T = \alpha \nabla^2 T$$

The effect of the electromagnetic field is introduced into the equation of motion through the jxB term, which as the vector product of magnetic induction and induced current represents the Lorentz force that acts on the fluid. The Lorentz force may be evaluated with the aid of Maxwell's equations, which may be written as:(87)

$$\underline{j} = \sigma(\underline{E} + \underline{U}\times\underline{B}) \qquad (3.23)$$

$$\underline{\nabla}\times\underline{B} = \mu_0 \underline{j} \qquad (3.24)$$

$$\underline{\nabla}\cdot\underline{B} = 0 \qquad (3.25)$$

$$\underline{\nabla}\times\underline{E} = -\frac{\partial \underline{B}}{\partial t} \qquad (3.26)$$

Expressing the above equations in terms of non-dimensional variables and then rewriting in terms of polar coordinates, yields the following set of equations to be solved:

The momentum balance equation

$$\frac{\partial}{r\partial r}\left(\omega\frac{\partial\psi}{\partial\theta}\right)-\frac{1}{r}\frac{\partial}{\partial\theta}\left(\omega\frac{\partial\psi}{\partial\theta}\right)=\frac{1}{R_e}\left[\frac{\partial}{\partial r}\left(r\frac{\partial\omega}{\partial r}\right)+\frac{1}{r^2}\frac{\partial^2\omega}{\partial\theta^2}\right]+X_m \tag{3.27}$$

where:

$$X_m = N\left\{\sin 2\theta\frac{\partial V_r}{\partial r}+\cos 2\theta\frac{\partial V_\theta}{\partial r}\right\}-\omega N\cos^2\theta \tag{3.28}$$

The vorticity - Stream function relationship:

$$\omega+\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\psi}{\partial r}\right)+\frac{1}{r^2}\frac{\partial^2\psi}{\partial\theta^2}=0 \tag{3.29}$$

and the energy transport equation:

$$\frac{1}{r}\frac{\partial}{\partial r}\left(T\frac{\partial\psi}{\partial\theta}\right)-\frac{1}{r}\frac{\partial}{\partial\theta}\left(T\frac{\partial\psi}{\partial\theta}\right)=\frac{1}{P_e} \tag{3.30}$$

with the following boundary conditions:

at $r=\infty$; $0\le\theta\le\pi$ $\omega=0$ $\psi=-r\sin\theta$ $T=T_{bulk}$

at $\theta=0,\pi$; $1\le r\le\infty$ $\omega=0$, $\psi=0$ $\frac{\partial T}{\partial\theta}=0$

at $r=1$; $0\le\theta\le\pi$ $\psi=0$ $T=T_W$

ω being calculated using first order vorticity generation.

In the above set of equations, the principal variables are defined as follows. Stream function ψ is defined by:

$$\underline{V} = \underline{\nabla} \times (\psi \, \underline{e}_z) \qquad (3.31)$$

where

$$\underline{V} = \underline{U} / U_0 \qquad (3.32)$$

Vorticity ω:

$$\underline{\omega} = \underline{\nabla} \times \underline{V} \qquad (3.33)$$

Interaction parameter N;

$$N = \frac{\sigma B_0^2 a}{\rho U_0} \qquad (3.34)$$

Reynold's Number, Re;

$$Re = \frac{U_0 a}{\nu} \qquad (3.35)$$

Peclet Number, Pe;

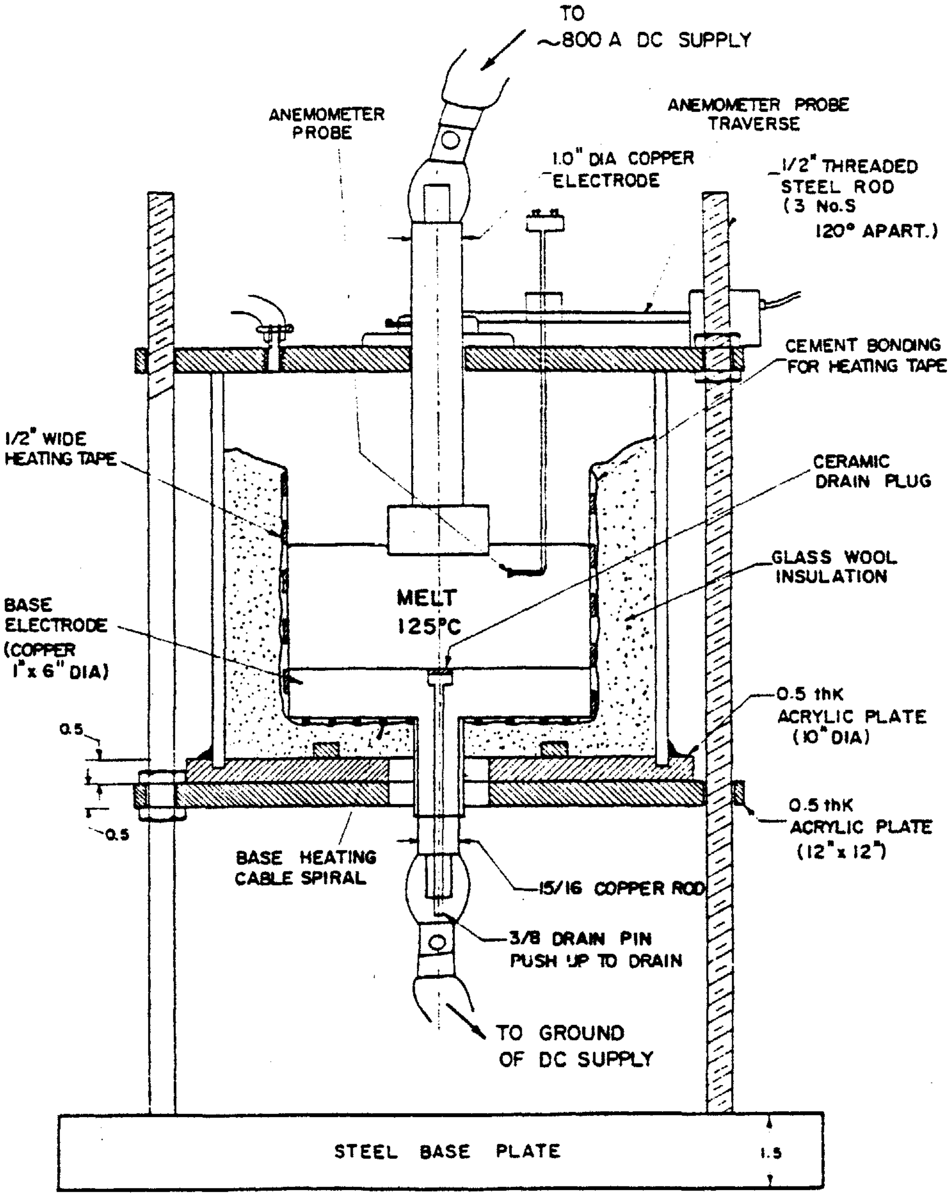
$$Pe = \frac{U_0 a}{\kappa} \qquad (3.36)$$

Dimensionless radial distance by

$$r = R/a \qquad (3.37)$$

3.6.2 Results

The results obtained from the above analysis are discussed in Chapter 5 and in Figs. 5.36 through 5.41.



EXPERIMENTAL SET UP TO STUDY ELECTOMAGNETICALLY
DRIVEN LIQUID METAL FLOW

Fig. 3.1 Experimental Set Up for Main Flow System

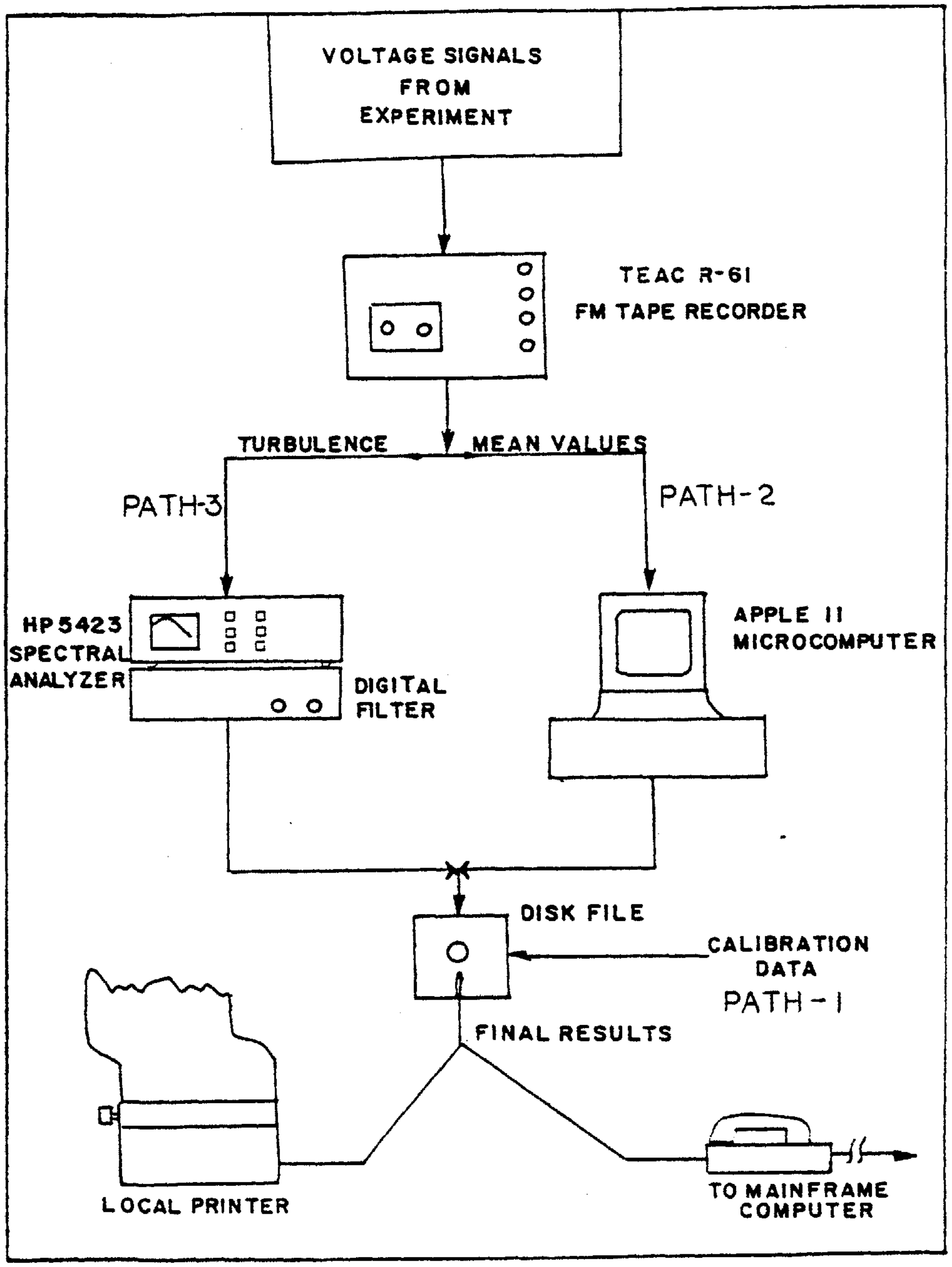


Fig. 3.2 Main Flow System Data Processing Path

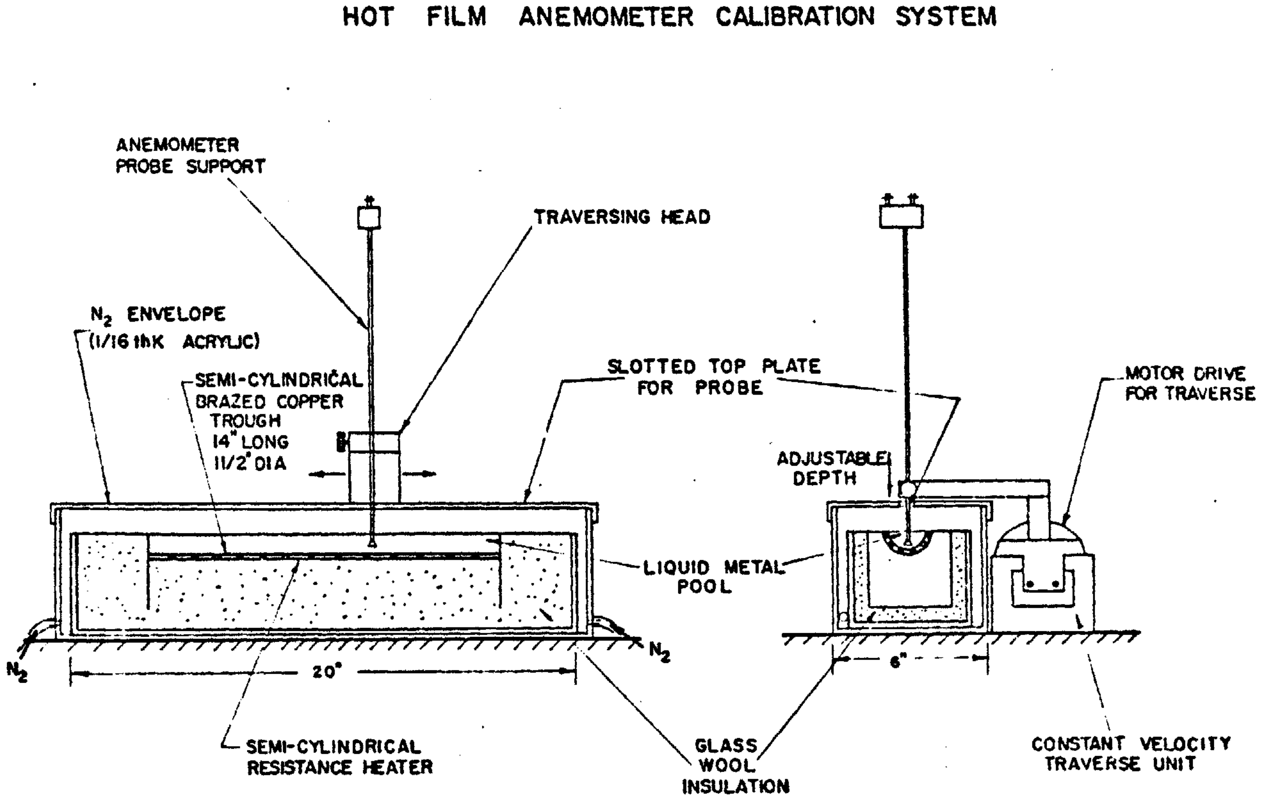


Fig. 3.3 Experimental Set Up for Calibration

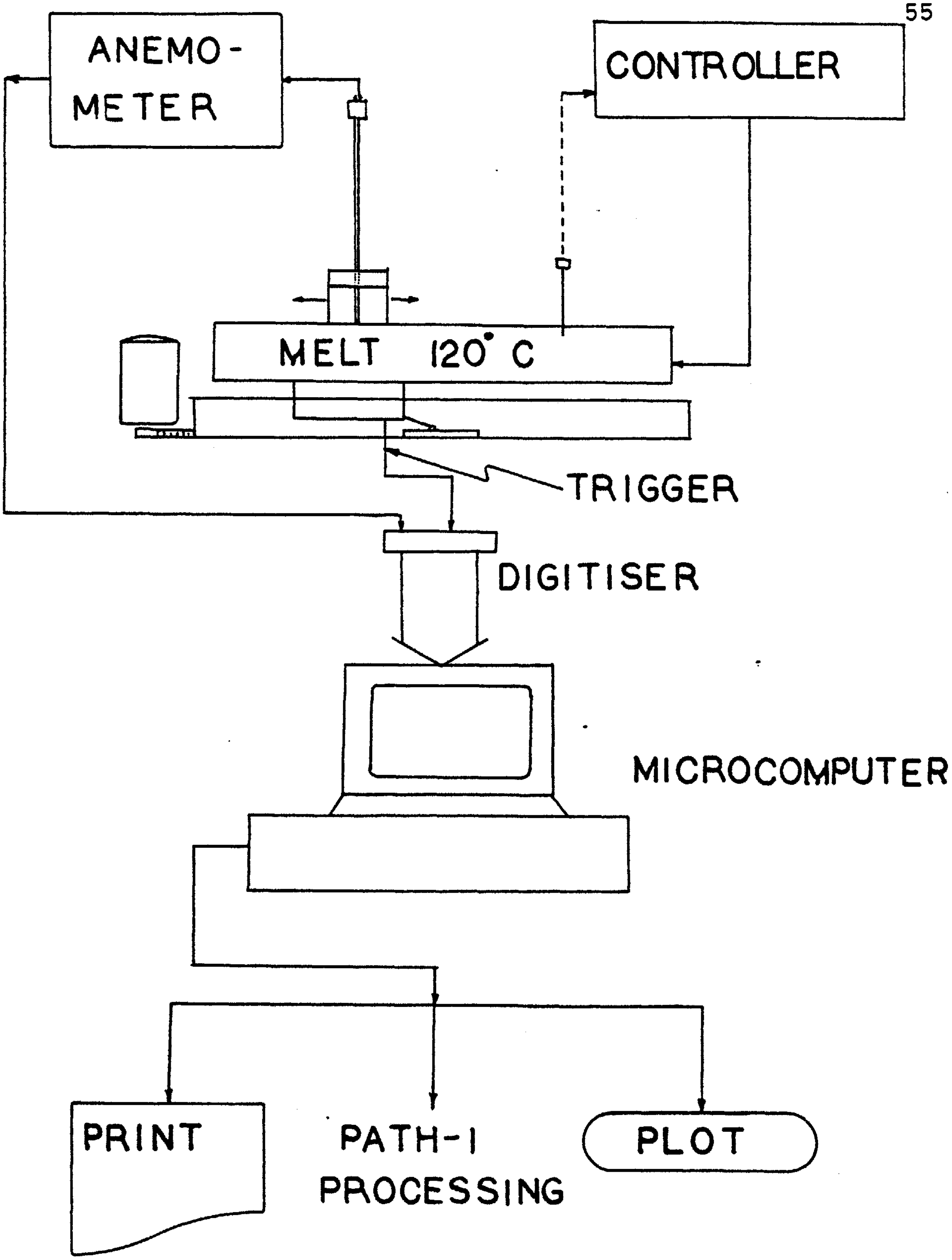


Fig. 3.4 Calibration System Data Processing

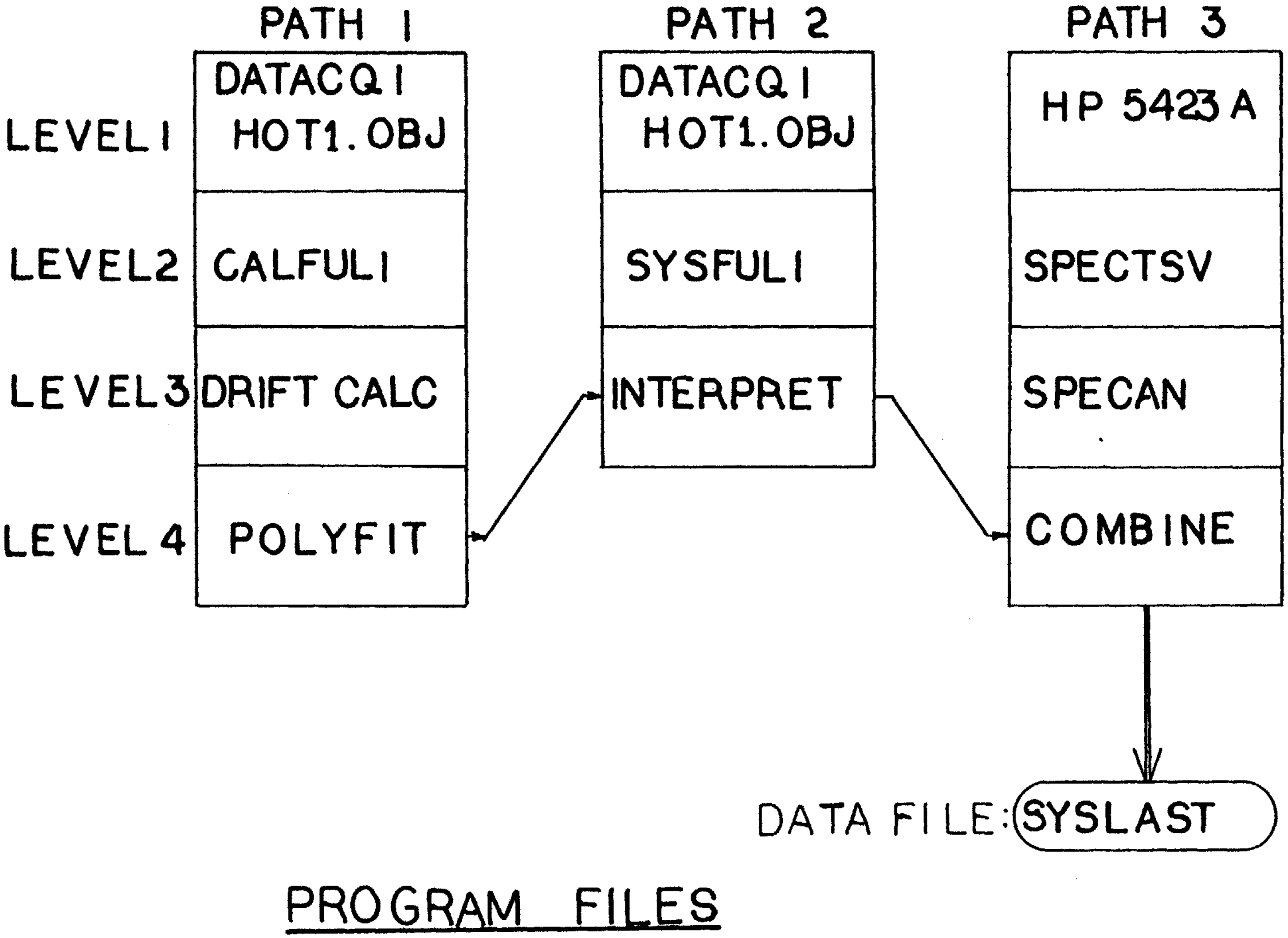


Fig. 3.5 Microcomputer Operating Environment - Program Files

	PATH 1	PATH 2	PATH 3
LEVEL1	‹RAM›	‹RAM›	‹PLOT›
LEVEL2	CAL N/M	SYS-NX	SPEC-NX
LEVEL3	CALN/M-*	SYSINT-NX	SPECINT-NX
LEVEL 4	CALN/M-*-COEF		SYSLAST-NX

DATA FILES

Fig. 3.6 ·Microcomputer Operating Environment - Data Files

4. THEORETICAL CONSIDERATIONS

4.1 Introduction

While results from experimental work possess the advantage of being direct measurements of real physical quantities, they are often restricted to the particular geometry and operating conditions of the experiment. It is therefore difficult, if not impossible, to extrapolate to other operating conditions. A mathematical formulation, coupled to a numerical solution scheme, on the other hand can be extended, almost at will, to differing operating conditions and geometries, within the limits of applicability of the model itself. It also lends itself very directly and simply, to the implementation of microprocessor based process control systems. However, a numerical scheme, needs to be evaluated and checked before acceptance and experimentation provides a convenient vehicle for making such a check.

In this chapter, a general mathematical formulation of the overall problem will first be discussed. Following this, the broad techniques used to convert this formulation to a workable numerical scheme are explained. Modelling of the turbulent flow phenomena has occupied a significant part of the effort involved in this thesis and so will be discussed in some depth. Three different numerical schemes are put forward. The first uses the conventional $\kappa - \epsilon$ model of turbulence, the second uses the Jones and Launder(79) low Reynolds number model. The third scheme uses a new basis for turbulence computation which was developed as part of this work.

4.2 Mathematical formulation of the problem

A mathematical description is sought for the flow phenomena occurring in the Wood's Alloy melt of the geometry and configuration detailed in Chapter 3. A schematic of the flow system is shown in Fig. 4.1

4.2.1 Principal Features and Assumptions

The following are the principal features of the system:

(1) The fluid is Wood's Alloy at 120° C with all the attendant property values which are listed in Appendix B.

(2) The flow is driven by applying a direct current between unequally sized electrodes. This creates a divergent current path in the melt which in turn creates a recirculating flow in the vessel.

(3) The flow is recirculating.

(4) The flow is turbulent.

(5) The fluid is essentially isothermal.

In formulating the model, the following assumptions were made:

(1) Cylindrical symmetry (no rotational flow) resulting from perfectly centered electrodes and vessel.

(2) The top electrode is assumed to be just touching the surface of the melt. (In the experiment, however, the top electrode was adjusted to be dipping into the melt a few millimetres to prevent any possibility of arcing when the current was turned on.

(3) Quasi steady state.

(4) The flow is assumed to be driven by electromagnetic forces alone, ie. the bouyancy driven component of flow was assumed to be negligibly small.

(5) Electromagnetic damping of the turbulent fluctuations was neglected. Following Alemany et. al.(38) , the interaction parameter N, based on the energy containing eddy size was estimated to be approximately

$$N = \frac{\sigma B_0^2 l}{\rho U} \sim 4 \times 10^{-5} \qquad (4.1)$$

which is small enough to have a negligible impact on the spectral energy distribution of the turbulence.

(6) The turbulence was assumed to be isotropic to allow the use of simpler models such as the two-equation model of turbulence.

4.2.2 The governing equations

Maxwell's equations

Upon applying the MHD approximation, Maxwell's equations take the following form:(78)

Faraday's Law:

$$\underline{\nabla} \times \underline{E} = -\frac{\partial \underline{B}}{\partial t} \tag{4.2}$$

Ampere's Law:

$$\underline{\nabla} \times \underline{H} = \underline{J} \tag{4.3}$$

$$\underline{\nabla} \cdot \underline{B} = 0 \tag{4.4}$$

$$\underline{\nabla} \cdot \underline{J} = 0 \tag{4.5}$$

Furthermore, we have

$$\underline{J} = \sigma(\underline{E} + \underline{U} \times \underline{B}) \tag{4.6}$$

and

$$\underline{B} = \mu_o \underline{H} \tag{4.7}$$

where σ is the electrical conductivity in mho-m, μ_o is the magnetic permeability of free space in Henry/m and U is the velocity of the fluid in m/s.

Equations 4.2 through 4.5 can be combined(78) to give:

$$\frac{\partial \underline{H}}{\partial t} = \eta \nabla^2 \underline{H} + \nabla \times (\underline{U} \times \underline{H}) \tag{4.8}$$

where

$$\eta = \frac{1}{\sigma \mu_o} \tag{4.9}$$

is called the Magnetic Diffusivity. Under conditions of low Magnetic Reynolds' Number, $R_{em}(= V_o\ L\ \sigma\ \mu_o)$, this equation can be reduced(32,36) to

$$\frac{\partial \underline{H}}{\partial t} = \eta \nabla^2 \underline{H} \tag{4.10}$$

or

$$\sigma \mu_o \frac{\partial H_\theta}{\partial t} = \frac{\partial}{\partial r}\left[\frac{1}{r}\frac{\partial}{\partial r}(r H_\theta)\right] + \frac{\partial^2 H_\theta}{\partial x^2} \tag{4.11}$$

in cylindrical coordinates with axial symmetry with the electromagnetic body force given by:

$$\underline{F}_{be} = \underline{J} \times \underline{B} = \mu_o \underline{J} \times \underline{H} \tag{4.12}$$

and the current densities being given by:

$$J_r = -\frac{\partial H_\theta}{\partial x} \qquad (4.13)$$

and

$$J_x = \frac{1}{r}\frac{\partial}{\partial r}\left(r H_\theta\right) \qquad (4.14)$$

Fluid Flow Equations

Using Reynolds' averaging applied to the turbulent fluctuating velocities, the equation of motion (Navier Stokes' equations) may be written as:

$$\rho(\underline{U}\cdot\underline{\nabla})\,\underline{U} = -\underline{\nabla}p - \underline{\nabla}\cdot[\tau] + \underline{F}_{be} \qquad (4.15)$$

where:

$$\underline{F}_{be} = \underline{J} \times \underline{B} \qquad (4.16)$$

and the mass conservation equation:

$$\underline{\nabla}\cdot\underline{U} = 0 \qquad (4.17)$$

Now introducing the vorticity ω

$$\underline{\omega} = \left[\frac{\partial U_r}{\partial x} - \frac{\partial U_x}{\partial r}\right]_\theta \qquad (4.18)$$

and the stream function, ψ

$$U_r = -\frac{1}{\rho r}\frac{\partial \psi}{\partial x} \qquad (4.19)$$

and

$$U_x = \frac{1}{\rho r}\,\frac{\partial \psi}{\partial r} \tag{4.20}$$

the equations 4.15 and 4.17 may be expressed as(74,36) :

$$r^2\left[\frac{\partial}{\partial x}\left(\frac{\omega}{r}\frac{\partial \psi}{\partial r}\right)-\frac{\partial}{\partial r}\left(\frac{\omega}{r}\frac{\partial \psi}{\partial x}\right)\right]-\frac{\partial}{\partial x}\left[r^3\frac{\partial}{\partial x}\left(\mu_{eff}\frac{\omega}{r}\right)\right]$$
$$-\frac{\partial}{\partial r}\left[r^3\frac{\partial}{\partial r}\left(\mu_{eff}\frac{\omega}{r}\right)\right]+r^2\left[\frac{\partial F_x}{\partial r}-\frac{\partial F_r}{\partial x}\right] \tag{4.21}$$

where

$$\mu_{eff} = \mu + \mu_t \tag{4.22}$$

The last term can be shown to take the following form:

$$r^2\left[\frac{\partial F_x}{\partial r}-\frac{\partial F_r}{\partial x}\right] = -\left[r\,\mu_0 H_\theta J_r\right] \tag{4.23}$$

In addition the following relationship exists between ω and ψ :

$$\omega + \frac{\partial}{\partial x}\left[\frac{1}{\rho r}\,\frac{\partial \psi}{\partial x}\right]+\frac{\partial}{\partial r}\left[\frac{1}{\rho r}\,\frac{\partial \psi}{\partial r}\right] = 0 \tag{4.24}$$

The presence of turbulent fluctuations in the flow affects the above formulation through the value for μ_{eff} . Considerations for the introduction of turbulent calculations are detailed in section 4.3

4.3 Turbulence Modelling - Conventional Models

The presence of turbulence introduces into the stress tensor τ (in the time averaged flow equations) an additional term (ρ u'v') called the Reynolds' Stress term. Following Boussinesq(74), turbulent or Reynolds' stresses can be computed using the same relationships which exist for viscous stresses in a Newtonian fluid but by replacing molecular viscosity with a scalar turbulent viscosity. This turbulent viscosity may be calculated using one of the several different methods available. The $\kappa - \epsilon$ model of turbulence is a two equation model that has been one of the most popular models for calculations of turbulent recirculating flows. In this section the $\kappa - \epsilon$ model, its low Reynolds' Number version(72) and the basis for a new model, the XI model will be examined.

4.3.1 The $\kappa - \epsilon$ Model

The $\kappa - \epsilon$ model uses two partial differential equations, one for κ and the other for ϵ. The κ (turbulence energy) equation can be easily derived by multiplying the (instantaneous) Navier Stokes equations by velocity to form the (instantaneous) Energy equation. Subtracting from this the mean energy equation which can be obtained by multiplying the Reynolds averaged Navier-Stokes equation with the mean (time averaged) velocity. For high Reynolds Numbers, this equation reads(82)

$$\overbrace{\frac{\partial \kappa}{\partial t}}^{I} + \overbrace{U_i \frac{\partial \kappa}{\partial x_i}}^{II} = \overbrace{\frac{\partial}{\partial x_i}\left[\overline{U_i\left(\frac{u'_i u'_j}{2} + \frac{p}{\rho}\right)}\right]}^{III} - \underbrace{\overline{u'_i u'_j}\frac{\partial U_i}{\partial x_j}}_{IV} - \underbrace{\epsilon}_{V} \qquad (4.25)$$

where:

Term I is the time rate of change of κ

Term II is the convective transport of κ

Term III is the diffusive transport of κ

Term IV is the generation of turbulence by mean shear (= 'g')

Term V is the dissipation of turbulence.

Terms II and III are of the redistributive type which just change the local values of κ . Introducing a gradient transport assumption for the diffusive flux of κ , gives the following form of the above equation:

$$U_i \frac{\partial \kappa}{\partial x_i} = \frac{\partial}{\partial x_i}\left(\frac{\mu_t}{\rho \sigma_K}\frac{\partial \kappa}{\partial x_i}\right) + \frac{\mu_t}{\rho}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right)\frac{\partial U_i}{\partial x_j} - \epsilon \qquad (4.26)$$

This equation alone cannot be solved because ϵ is an unknown variable. This in a sense represents the point of departure for the 1-equation, the 2-equation and the spectral models.

The $\kappa - \epsilon$ model resolves the problem by postulating another transport partial differential equation for ϵ of the form:

$$\frac{\partial \epsilon}{\partial t} + U_i \frac{\partial \epsilon}{\partial x_i} = \frac{\partial}{\partial x_i}\left(\frac{\mu_t}{\rho \sigma_e} \frac{\partial \epsilon}{\partial x_i}\right) + C_1 \frac{\epsilon}{\kappa} g - C_2 \frac{\epsilon^2}{\kappa} \qquad (4.27)$$

where

σ_e is the 'Prandtl Number' for ϵ transport

g is the generation of turbulence energy

C_1 , C_2 and σ_e are constants which were introduced when the exact form of the ϵ transport equation(79) is reduced to the gradient transport form given above. Constant C_2 is determined from experiments on the decay of behind grid turbulence while constant C_1 is determined from near wall boundary layer flows.

The following are the values suggested(82,84) for the constants in the $\kappa - \epsilon$ model.

$C_\mu = 0{\cdot}09 \qquad C_2 = 1{\cdot}92 \qquad \sigma_e = 1{\cdot}3$

$C_1 = 1{\cdot}44 \qquad \sigma_k = 1{\cdot}0$

The $\kappa - \epsilon$ model in the form described above has been applied successfully to many two dimensional wall boundary layers, duct flows, free shear flows and recirculating flows. A useful account of many interesting applications is given by Rodi(91) and by Launder and Spalding(84) .

4.3.2 The Low Reynolds Number Model

This model due to Jones and Launder[72] is an extension of the $\kappa - \epsilon$ model which was developed primarily to handle near-wall boundary layer flows for computations inside the viscous sublayer. An explanation of the principal basis of this model may be found in reference (72).

4.4 Turbulence Modelling - The XI Model

The XI model is a 1-equation turbulence model which possesses the capability to produce results very similar to, often almost exactly the same as the $\kappa - \epsilon$ model of turbulence. The technique, in essence, is to eliminate the dissipation energy transport equation altogether and to calculate the local dissipation energy directly from the local kinetic energy at that grid point. This is achieved by utilising certain information about the behaviour of idealised, decaying, isotropic turbulence and extending this to the case of a forced stationary flow. (It is shown later that this is not as strong an assumption as it may seem.) In more physical terms, the technique results in defining a certain quantity (XI) associated with large scale (low wavenumber) eddies which is:

(1) assumed to remain constant over the whole flow field.

(2) determined by the characteristic dimension of the system and by the amount of stirring power supplied to stir the flow system.

In a sense, this implies that the large scale eddies (the so called 'permanant scales') are determined by the dimensions of the flow system to a greater extent than the energy containing scales are, (the latter association being made in the case of conventional 1-equation models.)

4.4.1 Theoretical basis for the model

In 1945, Lotsianskii(66) proved that in decaying, isotropic turbulence, a certain functional, the 'Disturbance moment', (later called the 'Lotsianskii Invariant or Integral') should be a constant over the whole flow field and determined by the amount of disturbance supplied to the flow. The functional, defined as

$$I_L = \frac{8\pi}{3\pi} \int_0^\infty r^4 f(r)\, dr \tag{4.28}$$

was later shown by Proudman and Reid(54) and by Batchelor and Proudman(56) to be not truly a constant but rather a slowly varying function over the flow domain. Comte-Bellot and

Corrsin(80) showed that, except under certain conditions of transient flow, the assumption of the constancy of 'I_L' was reasonable for grid turbulence and indeed produced good results for the decay of behind-grid turbulence.

Using the Lotsianskii integral as determining the low wavenumber end of the spectrum gives a relationship of the form

$$E(k) = I_L\, k^4 \tag{4.29}$$

for the spectral power density distribution in that region(77). The spectral shape at higher wavenumber range (in the inertial subrange) has been shown(77,79) to be of the form

$$E(k) \triangleq \epsilon^{2/3}\, k^{-5/3} \tag{4.30}$$

with the two curves connected by a smooth curve, the Von-Karman patch(77) given by

$$E(k) = E(k_e)\, 2^{17/6} \frac{(k/k_e)^4}{\left[1+(k/k_e)^2\right]^{17/6}} \tag{4.31}$$

Simple algebraic manipulations of these three expressions yields the location of the energy containing scale, k_e , as

$$k_e = \left(\frac{1}{I_L}\right)^{3/17} \epsilon^{2/17} \tag{4.32}$$

$$E(k_e) = 2^{-17/6}\, I_L\, k_e^4 \tag{4.33}$$

It has also been shown(77,79) that integrating the spectral curve

$$\chi = \int_0^\infty E(k)\,dk \tag{4.34}$$

yields an expression of the form

$$E(k_e) = \frac{0.2u'^2}{k_e} \;\hat{=}\; 0.133\,\frac{\chi}{k_e} \tag{4.35}$$

substituting for $E(k_e)$ from eqn.(4.33) gives

$$\chi = \frac{2^{-17/6}}{0.133}\, I_L\, k_e^{5} \;\hat{=}\; I_L\, k_e^{5} \tag{4.36}$$

Substituting for k_e from eqn.(4.32) gives

$$\chi \;\hat{=}\; I_L^{2/17}\;\epsilon^{10/17} \tag{4.37}$$

or

$$\epsilon \;\hat{=}\; \xi^{1.7}\;\chi^{1.7} \tag{4.38}$$

where

$$\xi \triangleq (\,I_L^{-0.118}\,) \tag{4.39}$$

Equation 4.38 has the capability to yield a point value for dissipation energy, provided that the value of ξ is known. (Notice here that the assumption of the Lotsianskii 'disturbance moment' I_L being a constant is not such a strong assumption since it appears as ξ which is I_L raised to the power of -0.118 or, at worst, as $\xi^{1.7}$ which is I_L raised to the power of -0.200.)

The characteristic length scale associated with the large scale ('permanant') eddies is given by Hinze(77) as

$$\left(\frac{I_L}{\mu_t/\rho}\right)^{1/3} \sim L_P \tag{4.40}$$

Now postulating that the largest eddy sizes are constrained by the vessel dimensions and so equating L_p to the characteristic vessel dimension, 'L_v', yields an expression of the form

$$\frac{I_L}{\nu_t^2} \hat{=} L_v^3 \tag{4.41}$$

$$I_L \hat{=} \nu_t^2 L_v^3 \tag{4.42}$$

$$\xi \hat{=} \left(\nu_t^2 L_v^3\right)^{-0.118} \tag{4.43}$$

ν_T, the characteristic kinematic eddy viscosity can be determined once each iteration cycle by

$$\eta P = \nu_t \rho \iiint_{Vol} [S_{ij}]^2 \, dVol \tag{4.44}$$

or

$$\nu_t \hat{=} \frac{\eta P}{\rho \iiint_{Vol} [S_{ij}]^2 \, dVol} \tag{4.45}$$

where η is the efficiency of turbulence generation(60-63). (Also refer to Section 4.6)

Notice that in the process of eliminating the transport equation for ϵ , the constants C_1 , and C_2 and $\sigma_e$have also been eliminated. Indeed, in this model, the only constant remaining is $C_d$which is left equal to 0.09 as in the κ - ϵmodel (in addition to the Prandtl number for kinetic energy, $\sigma_k$which is generally accepted to be approximately unity.

4.5 Boundary Conditions

Boundary conditions for the three different schemes are substantially the same. For the spectral model, however, no boundary conditions for dissipation energy (ϵ) are required since there is no differential equation for this variable.

4.5.1 Boundary conditions for magnetic field calculations

The mathematical statement of the boundary conditions used with the differential equations in the preceeding sections is given below. A more detailed explanation of the considerations underlying these statements may be found in references (36) and (78).

(1) Upper Electrode Axial Surface

at $x=0$, $0 \leq r \leq R_e$

$$\frac{\partial H_\theta}{\partial x} = 0 \qquad \{ J_r = 0 \}$$

(2) Free Surface of Melt

at x=0, $R_e \leq r \leq R_m$

$$H_\theta = \frac{I}{2\pi r} \qquad \left\{ J_x = \frac{1}{r}\frac{\partial}{\partial r}(r H_\theta) = 0 \right\}$$

(3) Bottom electrode Surface

at x=X(5), $0 \leq r \leq R_m$

$$\frac{\partial H_\theta}{\partial x} = 0 \qquad \left\{ J_r = 0 \right\}$$

(4) Cylindrical wall of Vessel

at $r=R_m$, $0 \leq x \leq X(5)$

$$H_\theta = \frac{I}{2\pi R_m}$$

(5) Centerline of Vessel

at r=0, $0 \leq x \leq X(5)$

$$H_\theta = 0 \quad \text{OR} \quad \frac{\partial}{\partial r}\left(\frac{H_\theta}{r}\right) = 0$$

4.5.2 Boundary Conditions for Flow Equations

Boundary conditions are required for mean velocities and also for turbulence parameters. The conditions for the mean values should be stated in terms of the primary variables, Vorticity (ω) and Stream Function (ψ). The conditions for turbulence parameters are given in terms of expressions for kinetic energy (κ) and (except in the case of the spectral model) dissipation energy (ϵ). Only the final form of the boundary conditions are given here. A more detailed explanation may be found in references (36,74-76)

(1) Centerline of Vessel

at $r=0,\ X(1) \le x \le X(5)$

$$\psi = \frac{\partial \varkappa}{\partial r} = \frac{\partial \epsilon}{\partial r} = 0$$

$$\left(\frac{\omega}{r}\right)_0 = \frac{8}{\rho}\left[\frac{(\psi_0 - \psi_2)}{r_2^2} + \frac{(\psi_1 - \psi_0)}{r_1^2}\right] \Big/ (r_2^2 - r_1^2)$$

Where suffixes 0,1,2 denote the points on the centerline axis and the adjacent grid nodes in the r-direction respectively.

(2) Free Surface of Melt

at $x=X(1),\ R_e \le r \le R_m$

$$\psi = \frac{\omega}{r} = \frac{\partial \varkappa}{\partial x} = \frac{\partial \epsilon}{\partial x} = 0$$

(3) Top Electrode Axial Surface

at $x=X(1),\ 0 \le r \le R_e$

$$\psi = 0 \ , \ \varkappa = \epsilon = 0$$

$$\left(\frac{\omega}{r}\right)_0 = \frac{3(\psi_0 - \psi_1)}{\rho r^2 (x_1 - x_0)^2} - \frac{1}{2}\left(\frac{\omega}{r}\right)_{r_1}$$

Where suffixes 0 and 1 refer to a grid node on the boundary and to the adjacent node in the x-direction respectively.

(4) Bottom Electrode Axial Surface

at $x=X(5),\ 0 \le r \le R_m$

$$\psi = 0 \qquad \varkappa = \epsilon = 0$$

$$\left(\frac{\omega}{r}\right)_0 = \frac{3(\psi_0 - \psi_1)}{\rho r^2 (x_1 - x_0)^2} - \frac{1}{2}\left(\frac{\omega}{r}\right)_{r_1}$$

The suffixes have the same meaning as in B.C. 4 above

(5) Cylindrical Wall

at $r=R_m$, $X(1) \leq x \leq X(5)$

$$\kappa = \epsilon = 0$$

$$\left(\frac{\omega}{r}\right)_0 = \frac{3}{\rho} \cdot \frac{(\psi_0 - \psi_1)}{(r_1 - r_0)^2 r_0 r_1} - \frac{1}{2}\left(\frac{\omega}{r}\right)_1$$

Where suffixes 0 and 1 refer to a grid point on the boundary and to the adjacent node in the r-direction respectively.

4.6 Numerical Solution Scheme

The derivation of the finite difference discretization equations and the treatment of wall and neighbor nodes follows the techniques laid down in references (74) and (36) and will not be discussed here. The program itself is based, in part, on the work of Dr. M. K. Choudhary(36) .

Treatment of boundary conditions for the turbulent parameters needs special care. As detailed by Pun and Spalding(82) , a pseudo boundary condition is set up for the wall neighbor nodes. A source term for the kinetic energy equation is specified as:

$$S_k = \tau_w \frac{\partial U}{\partial n} - C_d \frac{\rho^2 \kappa^2}{\tau_w} \frac{\partial U}{\partial n} \tag{4.46}$$

at the neighbor node while the dissipation energy is calculated from :

$$\epsilon = C_d^{3/4} \kappa^{3/2} / (0.41 \delta) \tag{4.47}$$

The three models all share a large part of the program, hence it was found convenient to build the entire system as a single program set ('EMFLOW') and allow choice of the required options by reading a controlling data file. The computational steps followed under the influence of different control options is shown in Fig. 4.2. The algorithmic path taken within the program can be seen to be arranged in a cascade or 'waterfall' fashion allowing the user to exit computations at any of the levels. A flow chart of the steps followed by the program when Routing I (the Spectral model) is chosen (by setting NFIELD, NFLOW, NTURB, NSCHEM = 1 and LREYN, NCHECK =0) is shown in Fig. 4.3. A similar flow chart for the steps entailed in choosing Routing II (the $\kappa - \epsilon$ model) is shown in Fig. 4.4. It may be noted here that the XI model has only one partial differential equation to solve while the $\kappa - \epsilon$ model has two. The extra step in the XI model of computing ξ needs only the solution of an algebraic equation and that too only in an integral sense for the entire control volume.

A complete source code listing of all these routines is also provided for reference in Appendix F.

Provision is made for the program to auto-check whether the flow will indeed be turbulent or laminar by evaluating a Performance Index of Stirring (PI). This is believed to be a far more reliable estimate of the degree of turbulence than afforded by a simple estimatation of overall Reynolds Number. The basis for the calculation of PI is described in Section 4.7.

4.7 Some Useful Integral System Parameters

The objective of this section is to extend concepts of turbulence theory to compute some overall system parameters which may be beneficial in attempting to answer practical questions such as;

(1) Is the flow going to be laminar or turbulent? What is the degree of turbulence?

(2) How long is it likely to take for an added contaminant to become homogeneously distributed in the melt?

4.7.1 A Performance Index for Stirring

To sustain turbulence in a flow system, it is necessary to have a steady flow of energy through the turbulent energy cascade. In an overall sense, (integrating over the contents of the whole vessel) the total rate of generation of turbulence should equal the total rate of dissipation of turbulence.

$$G = \Phi \tag{4.48}$$

It is, however, not necessary for the entire stirring power supplied to the system by the external agency (Electromagnetic force field, Gas injection or other as the case may be) to appear as turbulence(54,67-70). Corrsin(67,68) based on Laufer's(54) measurements and later Brodkey(69,70,89) have postulated the existence of a factor of efficiency (η) governing the generation of turbulence. Thus it may be said that:

$$\eta P = G = \Phi \tag{4.49}$$

The local generation term g_{ij} can be expressed as :

$$g_{ij}(r,x) = 2\mu_t\left[\left(\frac{\partial U_x}{\partial x}\right)^2 + \left(\frac{\partial U_r}{\partial r}\right)^2 + \left(\frac{U_r}{r}\right)^2 + \frac{1}{2}\left(\frac{\partial U_r}{\partial x} + \frac{\partial U_x}{\partial r}\right)^2\right] \tag{4.50}$$

evaluated at location (r,x).

Integrating this over the whole volume gives:

$$G = \iiint_{Vol} g_{ij}(r,x)\, dVol \tag{4.51}$$

Taking a characteristic value for μ_t as a constant over the whole flow field, the expression for the total generation, G may be written as :

$$G = 2\mu_t \iiint_{Vol} \left[\left(\frac{\partial U_x}{\partial x}\right)^2 + \left(\frac{\partial U_r}{\partial r}\right)^2 + \left(\frac{U_r}{r}\right)^2 + \frac{1}{2}\left(\frac{\partial U_r}{\partial x} + \frac{\partial U_x}{\partial r}\right)^2 \right] dVol$$

$$P.I. = \frac{\mu_t}{\mu} = \eta\, P \Big/ \left\{ 2\mu \iiint_{Vol} \left[\left(\frac{\partial U_x}{\partial x}\right)^2 + \left(\frac{\partial U_r}{\partial r}\right)^2 + \left(\frac{U_r}{r}\right)^2 + \frac{1}{2}\left(\frac{\partial U_r}{\partial x} + \frac{\partial U_x}{\partial r}\right)^2 \right] dVol \right. \qquad (4.52)$$

4.7.2 Mixing Time Analysis

Following the analysis adopted by Corrsin(67) it may be shown that the time constant of mixing can be given by:

$$\tau = \frac{\ell^2}{12\, D_c} \hat{=} \frac{\lambda^2}{6\nu} \qquad (4.53)$$

Following Taylor, the dissipation rate ϕ may be expressed by

$$\phi \hat{=} \frac{10}{3}\nu \frac{\overline{q}^2}{\lambda^2} = \eta \frac{P}{M} \qquad (4.54)$$

using values averaged over the whole melt of mass M Kg.

In Isotropic turbulence at large Reynolds Numbers, Von Karman and Howarth deduced an approximate relation between dissipative scale λ and integral scale L

$$\frac{\lambda}{L} \sim \frac{1}{R_\lambda} \qquad (4.55)$$

Using the constancy factor proposed by Corrsin we may rewrite the above relation as

$$\frac{\lambda}{L} = \frac{A}{R_\lambda} \tag{4.56}$$

$$R_\lambda = \frac{u'\lambda}{\nu} = \frac{q/\sqrt{3} \quad \lambda}{\nu} \tag{4.57}$$

$$q \triangleq \frac{\sqrt{3} \quad A \quad L \quad \nu}{\lambda^2} \tag{4.58}$$

substituting this in eqn.(4.53) gives:

$$\tau \triangleq \left\{ \frac{A^2}{21{\cdot}6\eta} \quad \frac{ML^2}{P} \right\}^{1/3} \tag{4.59}$$

where $A \simeq 20$ (approx.) as measured from behind grid turbulence measurements( ) . Expanding eqn.(4.59) gives

$$\tau \triangleq \left(\frac{A^2}{21{\cdot}6\eta} \right)^{1/3} \left(L^2 \ \frac{M}{P} \right)^{0{\cdot}333} \tag{4.60}$$

Defining 'mixing time'(t_m) as the time required for concentration fluctuations to be within 5% variation from the mean, gives

$$t_m \simeq 3\,\tau$$

So

$$t_m \triangleq 3 \left(\frac{A^2}{21{\cdot}6\eta} \right)^{1/3} \left(L^2 \ \frac{M}{P} \right)^{0{\cdot}333} \tag{4.62}$$

If the same equation were written using power in Watts/Ton of melt (P_t), the expression for t_m would read:

$$t_m \triangleq \frac{80}{\eta^{1/3}} \left(L^2 / P_t \right)^{0{\cdot}333} \tag{4.63}$$

Sano and Mori(58) have found that measurements of t_m in various configurations and sizes of gas injected systems can be well correlated by an expression of the form:

$$t_m = 100 \left\{ \left(\frac{D^2}{H} \right)^2 / P_t \right\}^{0.337} \qquad (4.64)$$

Fig. 5.33 taken from Sano and Mori(58) shows this correlation, together with the experimental results of various researchers working with gas injected systems. 'A' represents the correlation stated above. Lines B1 and B2 represent the experimental results of Haida et.al.(57) for a water model without and with slag respectively. Lines C1 and C2 represent the experimental results of Hsiao et.al.(56) for 60 ton and 6 ton gas injected ladles while line D represents the results of Lehrer(55) for a water model of a gas sparged system.

> (Note here that the analysis of Sano and Mori is not being used but rather the fact that measurements quoted therein can be correlated by the expression above. The original analysis of Sano and Mori considered the time of mixing as determined by the circulation time for the mean flow in the vessel. However, as Nakanishi et. al.(59) have pointed out, mixing time in metallurgical melts appears to be dominated by diffusion (and eddy diffusion) rather than by bulk circulation time in the melt. This appears to be reasonable considering the high

Schmidt Numbers encountered in metallurgical flow systems. It should also be pointed out at this stage that the turbulence analysis followed in this work is quite general in nature and is not restricted in any way to gas injected systems. The value of turbulence efficiency would, of course, be expected to be different in different classes of systems.)

In keeping with the rather gross approximations made in this analysis, it may be written that:

$$L \simeq D \simeq H \simeq L_v$$

in which case eqn.(4.63) becomes

$$t_m \simeq \frac{80}{\eta^{1/3}} \left\{ L_v^2 / P_t \right\}^{0.333} \tag{4.66}$$

and eqn.(4.64) becomes:

$$t_m \simeq 100 \left\{ L_v^2 / P_t \right\}^{0.337} \tag{4.67}$$

Equating the two gives:

$$\eta = (0.80)^3 = 0.51 \tag{4.68}$$

Somewhat surprisingly, this value of η for turbulence generation in gas injected ladles is found to match exactly with the value of 0.50 found by Laufer for pipe flow turbulence generation and suggested by Corrsin as a first guess value for η

This value of efficiency has been calculated for the case of gas injected ladle metallurgical systems. It is expected that the efficiency of turbulence generation would be different for different classes of systems. However in the absence of data of this sort for different systems of metallurgical interest, it is recommended that η =0.50 be used as a first guess until a better estimate is forthcoming.

Nakanishi et. al.(59) have noted in their work that the data they reviewed (see Fig.5.32) does not appear to demonstrate any dependance on the linear size of the particular equipment that was studied. Indeed at first glance this would appear to be true from Fig.5.32. However a closer look at the derived dependance of mixing time on linear size (Eq.4.66) indicated a relationship of the form

$$t_m \propto L_v^{(2/3)} \qquad (4.69)$$

Now, in the data surveyed by Nakanishi et. al. the largest vessel is a 200 ton unit while the smallest is a 65 kg water model. This implies that the ratio:

(max. vol. of vessel)/(min. vol. of vessel) is in the region of 340.Very roughly, this would a indicate ratio of characteristic linear dimensions to be around $(340)^{(1/3)} = 7$

Now in eq. (4.69), L_v, the characteristic vessel dimension appears to the power of 0.66. This would mean that introducing into Fig. 5.32 a linear scale dependancy as indicated by eqs.4.66 and 4.67 would imply a change in Y-axis location of the data points by a factor of $(7)^{(2/3)} = 3.6$ or, on the Log scale used in Fig.5.32, a maximum difference of 0.5 units. This change is clearly of the same magnitude as the half decade variability of the data points themselves. It may therefore be inferred that an expression for mixing time such as the one given by eq.4.66 does not necessarily contradict the findings of Nakanishi et. al. Indeed, the additional parameters of efficiency and linear vessel size may prove to be one of the reasons contributing to the spread in data points.

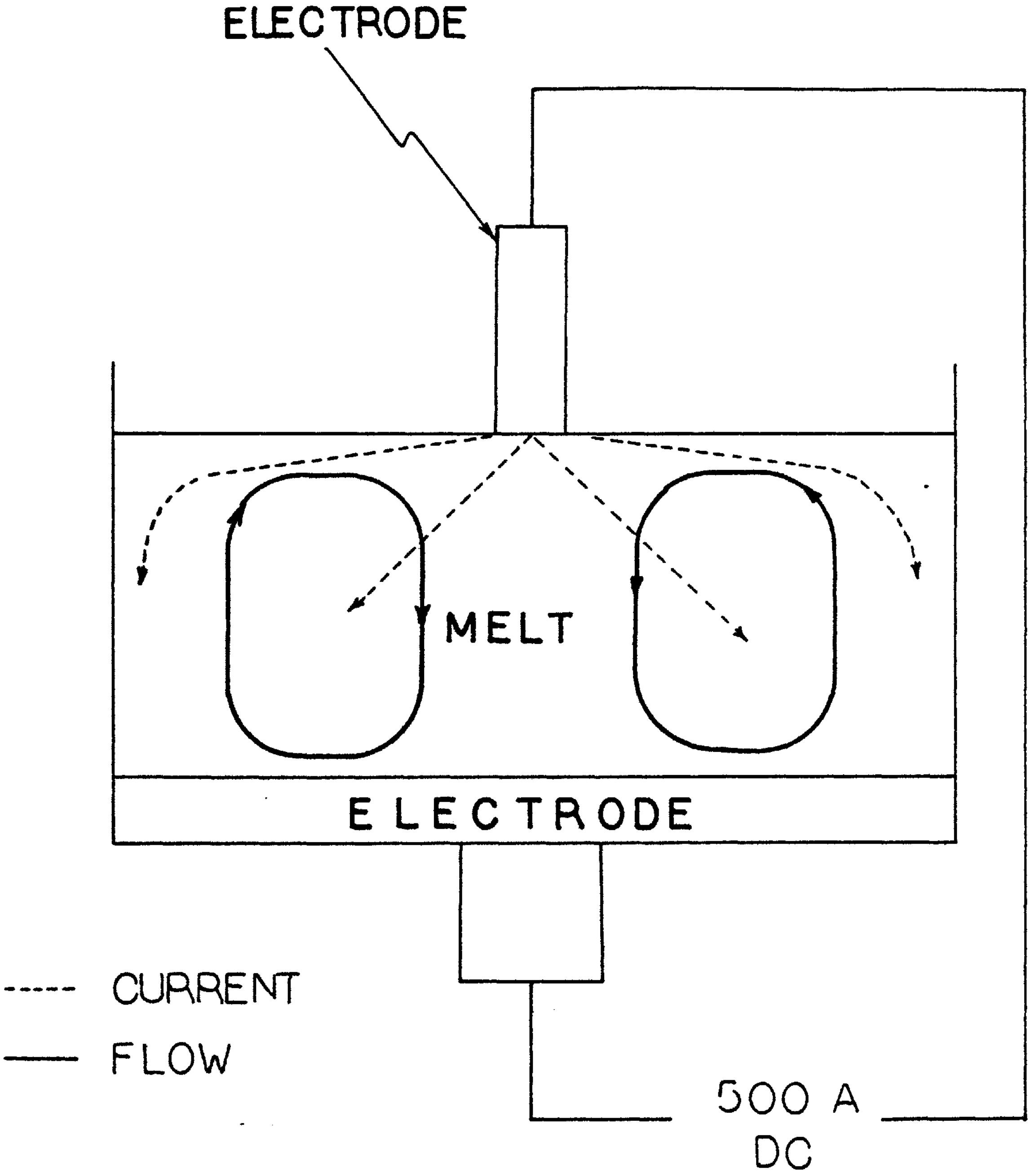


Fig. 4.1 Schematic of Flow System to be Modelled

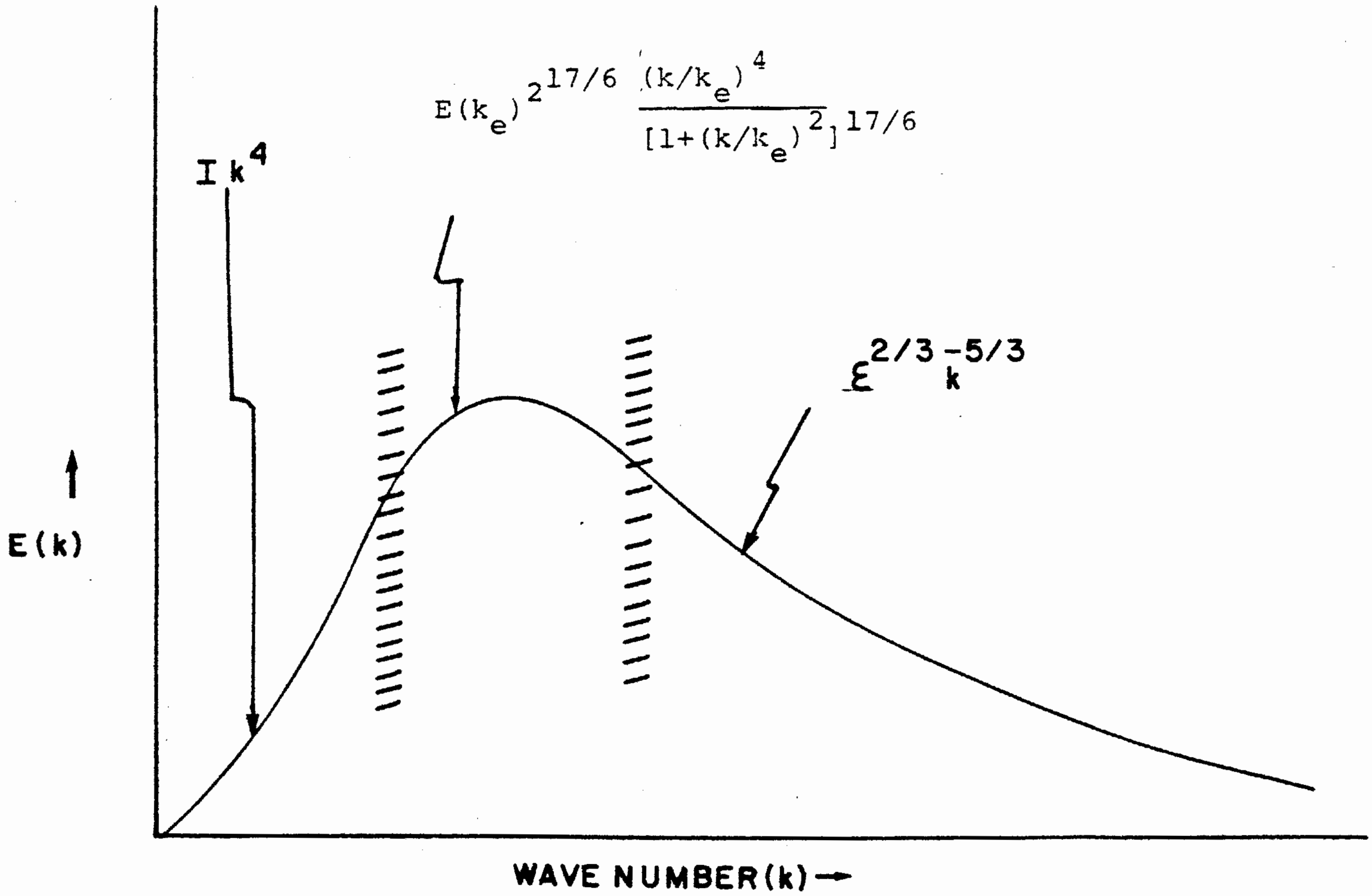


Fig. 4.2 Schematic of 3-D Turbulence Energy Spectrum

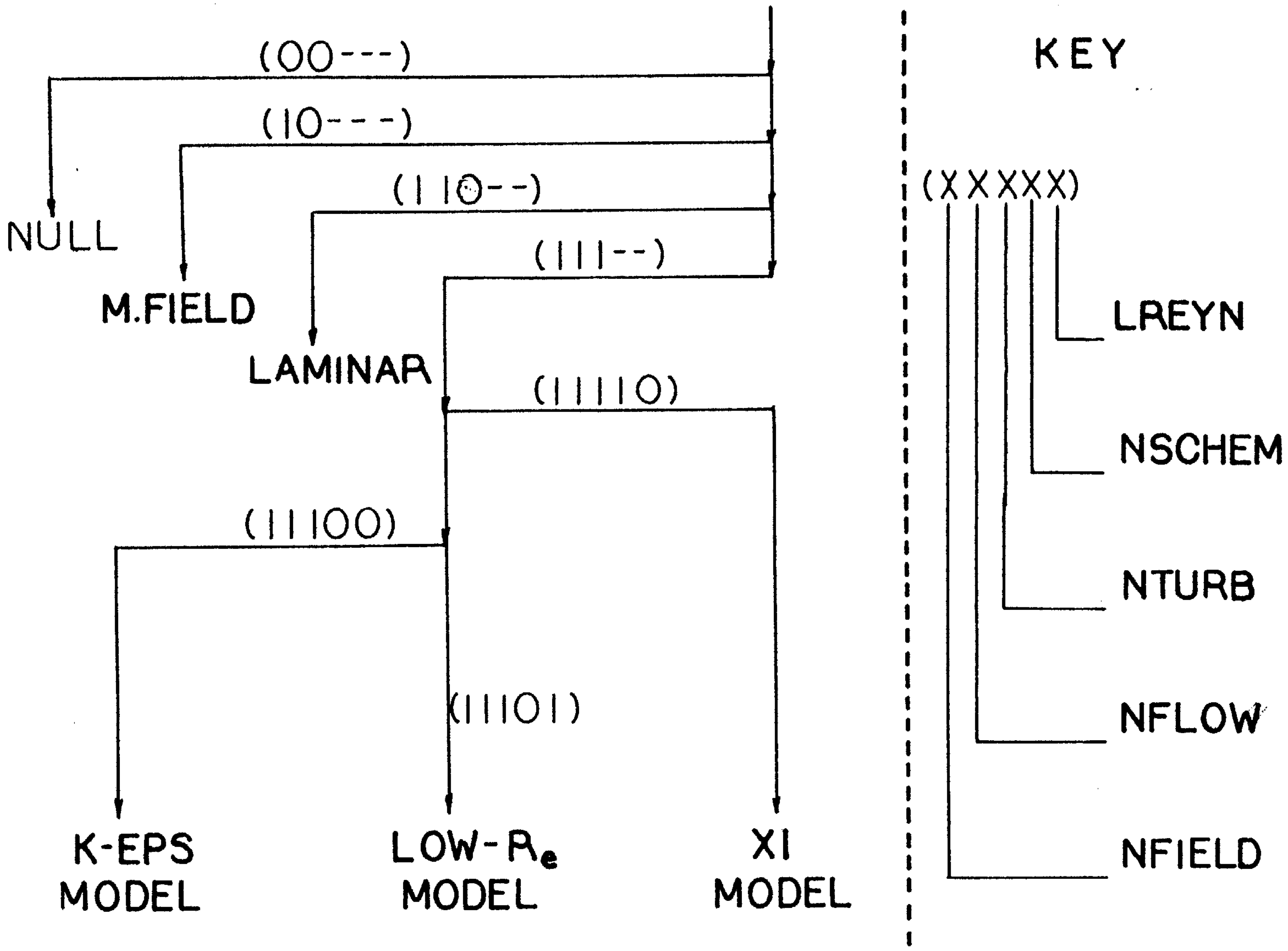


Fig. 4.3 Option Control in Computer Program

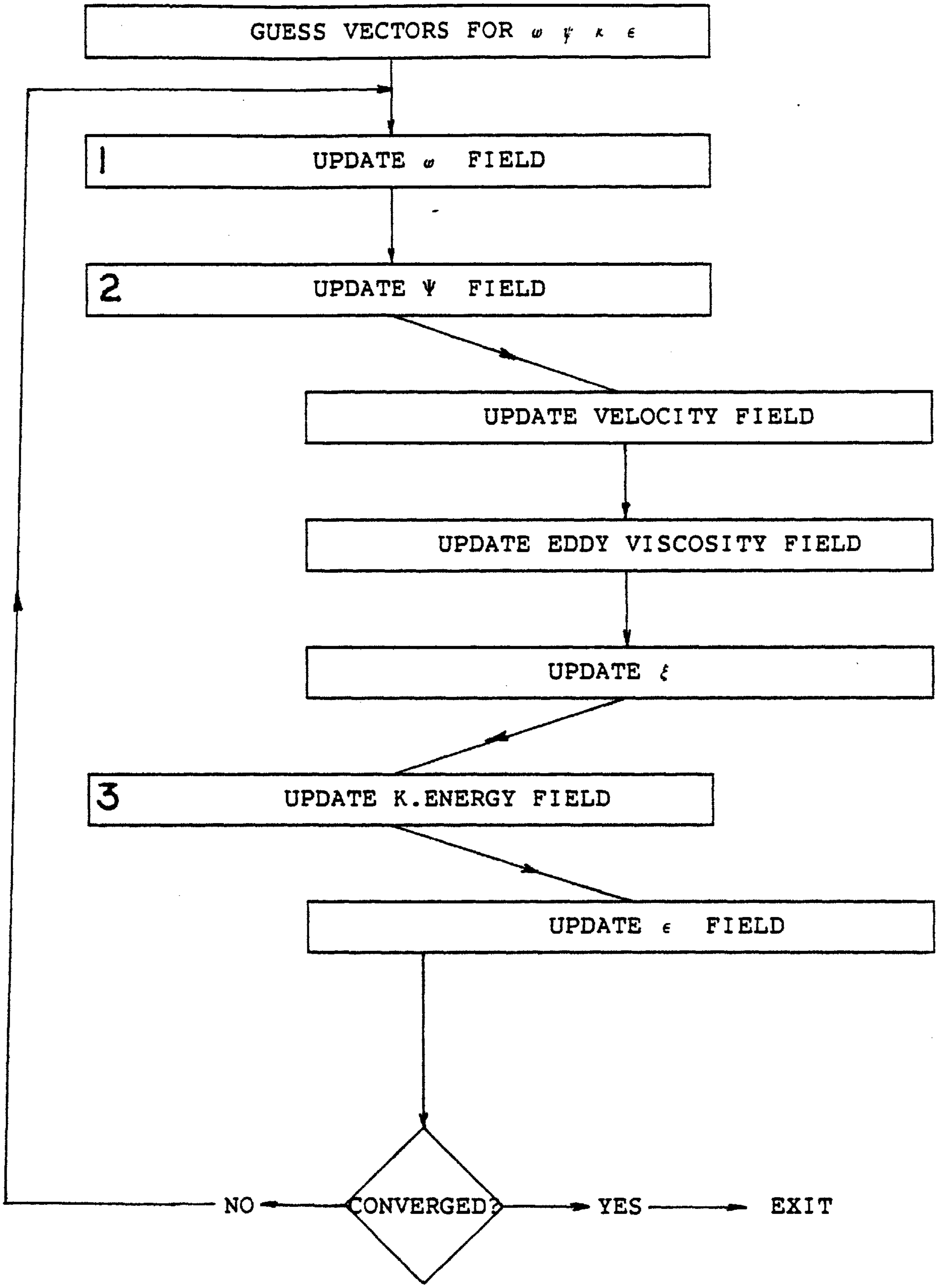


Fig. 4.4 Flow Chart of Calculations using XI Model (1 eqn. model)

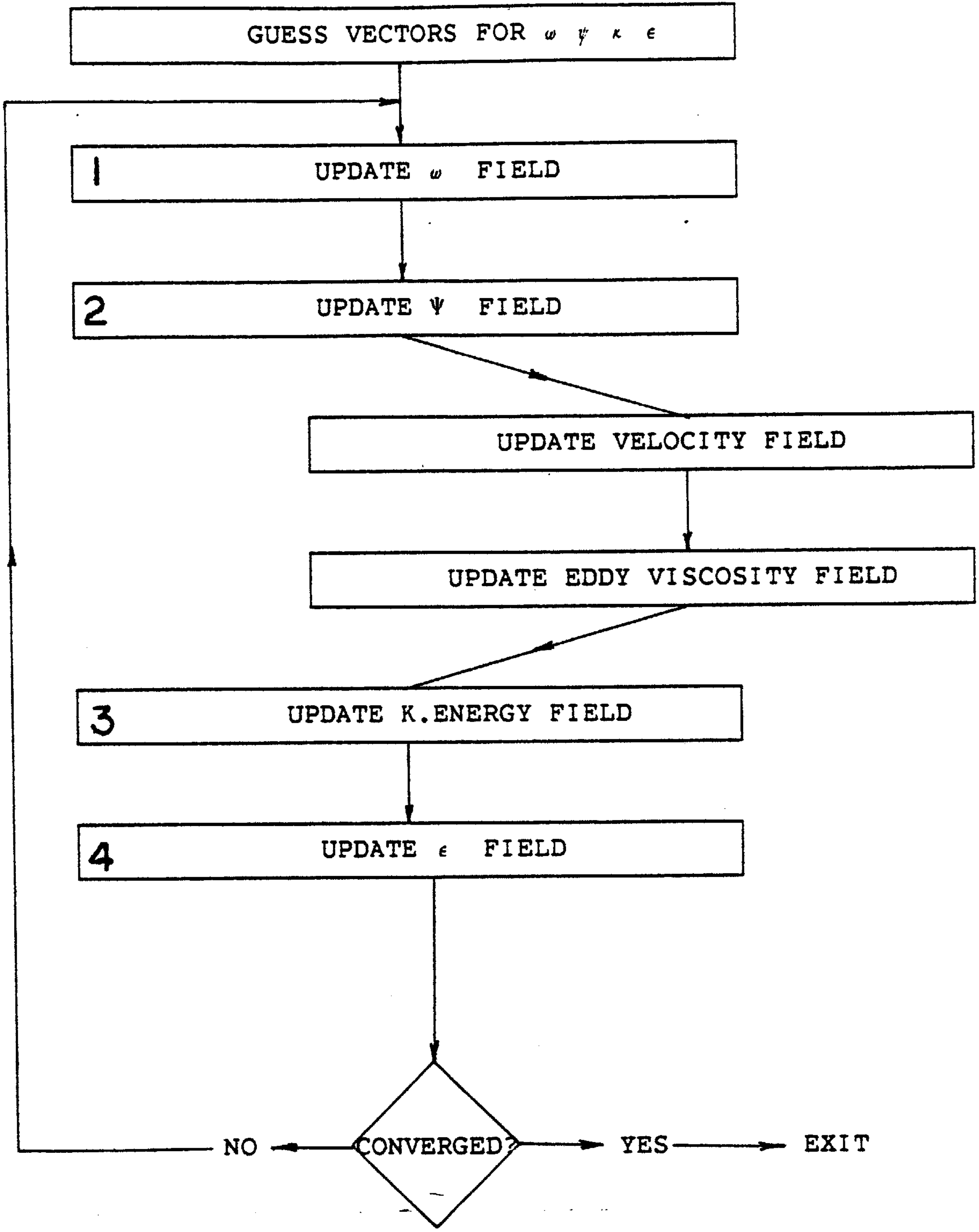


Fig. 4.5 Flow Chart of Calculations using $\kappa-\epsilon$ Model (2-eqn. model)

5 RESULTS AND DISCUSSION

In this chapter, results of experimental measurements and numerical computation will be compared and discussed. In addition the behaviour of the calculated values for certain practically relevant parameters like Performance Index (P.I. for short) and Mixing Time (t_m) and finally the results of analyses pertaining to the use of thermal anemometry in liquid metal and also in the presence of magnetic fields will be presented. Throughout this chapter, Models I, II and III refer to the $\kappa - \epsilon$ model, the Low Reynolds Number model and the XI model respectively.

5.1 Main Flow System - Measurements and Computations

Fig. 5.1 shows computed current density (J) vector plot. Two facts stand out here: (1) Most of the current passes through in a region relatively close to the smaller electrode. This means that insulation of the wall of the vessel should not play too important a role as the leakage wall current may, quite reasonably, be expected to be small. (2) The current divergence is the strongest in the immediate neighborhood of the small electrode. Hence this region is crucial in driving the overall flow.

Fig. 5.2 shows a computed velocity vector plot. The formation of one circulation loop is clearly seen. The arrows denote the velocity vector at the center of the arrow. The broken arrows, however, indicate only the local flow direction, the magnitude being greater than the maximum for the plot (3.0 cm/sec). Fig. 5.3 through 5.9 show the distribution of the local velocity across the radius at various depths. The absolute magnitude velocity was chosen as a basis for presentation since the hot film probe in the experiment actually measured the absolute resultant velocity. The good agreement of the experimentally measured velocities with the computed values is evident. Also to be noticed is the fact that all three models (I- $\kappa - \epsilon$, II- Low Reynolds Number version of $\kappa - \epsilon$, and III- XI model) predict exactly the same distribution except at the centerline. This extremely close correspondence between the calculations is partly because the flow is largely convection driven and partly because the turbulent viscosity predictions also match very closely (see Fig. 5.24). Fig. 5.8 clearly shows up the location of the eye of the vortex at about 5 cms radius. Notice also that the dip in the velocity near the vortical region is also reflected in the experimental measurements.

Fig. 5.35 shows a typical expeimentally measured spectrum of turbulence with the various pertinent operating conditions and directly computed parameters noted on it.

Scalar valued overall Dissipation Energy is plotted along the radial direction at different depths in Fig. 5.10 through 5.16. Once again two facts are to be noticed here: (1) The XI model, even though it is a one equation model (see description in chapter 4) and therefore not containing any of the constants involved in the ϵ transport equation, comes remarkably close to the predictions using the $\kappa - \epsilon$ model of turbulence. The Jones and Launder Low Reynolds Number model, however, also predicts approximately the same as the other two models being only marginally different from the $\kappa - \epsilon$ model. (2) All three models seem to predict the experimentally observed values of dissipation energy reasonably well in general. The agreement gets progressively worse as one approaches the depth at which the vortical point is located and gets better the farther away one is from the vortex.

Figures 5.17 through 5.23 show the variation of local RMS turbulence velocity in the form of radial scans at various depths from the melt surface. Once again the new XI model predicts values of the RMS fluctuations very close to those predicted by the $\kappa - \epsilon$ model. Comparison with experimentally measured values is however less encouraging. In general the numerical calculations overpredict the values of the fluctuating velocity. This may seem somewhat surprising, in view of the fact values of dissipation energy were really quite well represented by the computations. One possible reason for this apparently paradoxical behaviour may be related to the fact that the major contribution to the total dissipation energy comes from a range of frequencies towards the higher frequency end of the spectrum while the major contribution to the value of κ comes from frequencies in the lower ranges, particularly in the vicinity of the spectral density peak. Now the spectrum analysis equipment used in this experimentation (as in most cases) uses an AC coupled mode with a lower frequency cut off at 0.1 Hz (which is used by the system to distinguish the fluctuating from the mean component). This low frequency truncation would result in lower values for experimentally measured κ but would not affect the value for ϵ . Also, the higher wavenumber region of the spectrum has been shown to be closer to isotropy than the low wavenumber regions. Since the dissipation energy is a high wavenumber phenomenon it is less likely to be affected by the assumption of isotropy than the kinetic energy and RMS fluctuation, which have a major contribution from the low wavenumber range of the spectrum. It would, however, seem hard to attribute the discrepancy entirely to these effects and it is likely that there is a combination of factors at play here.

Fig. 5.24 is a representative plot of the variation of effective viscosity across the radius. As may be expected, the three models predict almost the same values of turbulence enhanced viscosity, though the XI model shows a peak at the centerline while the $\kappa - \epsilon$ model shows a peak somewhat away from the centerline. The wall neighbor node has exactly the same value with all three models since it is controlled largely by the 'Wall Function' calculation scheme which is common to all three models.

The concept of the efficiency of turbulence generation plays a central role in the formulation of the XI model. On the basis of the considerations discussed in Chapter 4, a value of η =0.5 was used in all the preceeding calculations. Fig. 5.25 and Fig. 5.26 show the effect of varying the value of η on the ϵ and RMS fluctuating velocity profiles across the radius of the melt. Decreasing the value of η lowers the calculated value of both the Kinetic and Dissipation energies all over the field, or in general, decreases the degree of turbulence in the melt.

Figures 5.27, 5.28 and 5.29 show contour plots for Kinetic Energy, Dissipation Energy and Effective Viscosity from calculations using the XI model. These plots present the same information as in the radial plots in a more compact fashion and are included here mostly to provide a 'bird's eye view' of the system.

5.2 Performance Index Computations

Fig. 5.30 shows sample voltage outputs from the anemometer when different currents are passed through the melt. The 100 Amp. trace could be said to be laminar or at least substantially so. The 250 Amp. trace shows up what is probably the onset of turbulence and is what may be characterised as 'weakly turbulent'. The 500 Amp. setting appears definitely turbulent. The transition from laminar to turbulent flow occurs somewhere between 100 A and 250 A possibly close to the 250 A end. Even a qualitative understanding of the onset of turbulence can be important because, as will be discussed later in this chapter, the rate of turbulence dissipation can be an important parameter in determining mixing, homogenisation and even reaction rates in the melt. Use of vessel Reynolds Number to predict the onset of turbulence may be misleading. For example, Reynolds Numbers based on the depth of melt and centerline velocity for the 100, 250 and 500 A cases are:

$R_{100} = 2090 \quad R_{250} = 5570 \quad R_{500} = 11\ 146$

It would be difficult, if not impossible, to judge the onset of turbulence without apriori knowledge of the dynamics of the specific flow configuration (an example would be the transition point for pipe flow which is placed around 2300 = Re_d .) This would be particularly difficult for the more complex case of a recirculating flow system.

The whole concept of Performance Index (P.I.) fits in here. The value of PI is an immediate intuitive estimate of the 'degree of turbulence' in the whole flow field and reflects the ratio of Turbulent viscosity to Molecular Viscosity in an integral sense. Hence a value of PI=0 would indicate an ideally laminar flow situation. Intuitively, it may be said that any flow with a value of PI greater than 10 can be considered turbulent. The values of PI for the three current settings are:

$PI_{100} = 5.5 \quad PI_{250} = 12.2 \quad PI_{500} = 23$

Looking at these figures one can place the transition to turbulence as somewhere between 100 A and 250 A, possibly closer to 250 A. This is similar to what may be deduced by examinining the traces in Fig. 5.30.

Fig. 5.31 shows a plot of the variation of PI with current. Such a plot could prove useful in predicting the degree of turbulence in a melt as the operating parameter (current) is changed.

5.3 Mixing Time Computations

Fig. 5.32 is the plot due to Nakanishi and Szekely (59), relating mixing time to specific power supplied to the melt (Total power/ mass of the melt) for various metallurgical systems and is excerpted from Reference (60). This represents the first attempt to relate metallurgical system mixer performance to turbulence parameters and is placed here in context of what is to follow. Fig. 5.32 is taken from Sano and Mori(58). Using a value of turbulence efficiency (η) as 0.5 and using eq.4.63 to calculate mixing time, values of t_m calculated for the experimental setup of this work (with currents = 100, 250, 500 and 2500 Amperes) have also been indicated on the figure with stars. Line A represents the correlation stated in equation 4.64. Lines B1 and B2 represent the experimental results of Haida et. al.(57) for a water model without and with slag respectively. Lines C1 and C2 represent the experimental results of Hsiao et. al.(56) for 60 ton and 6 ton gas injected ladles while line D represents the results of Lehrer(55) for a water model of a gas sparged system.

Fig. 5.34 shows the variation of overall mixing time for the current stirred melt used in this work, computed using the XI model with η = 0.5. Thus the melt could be expected to be 95% mixed in 24 seconds with a 500 Ampere current while it would take 120 seconds with a 100 A current and 4.8 seconds with a 2500 A current. Such a plot could be used in industrial practice to determine how long a blow or current stirring should be applied in order to get good mixing in the melt.

5.4 M.H.D. Flow around a cylinder

Figs. 5.36 through 5.41 relate to the behaviour of a heated cylinder subjected to a cross flow of metal and a magnetic field on the orthogonal axes. Fig. 5.36 shows the schematic of the flow situation. Figs. 5.37 and 5.38 are presented to serve as a check on the computational technique used in this work. Fig. 5.37 shows a comparison of calculated value of Drag Coefficient at zero magnetic field with corresponding values found by other workers. Fig. 5.38 compares the calculated values of Nusselt Number with the measurements of Ishiguro et. al.(95) and the older computations of Grosh and Cess(96) . The good agreement in both figures is evident.

Fig. 5.39 shows the computed variation of overall Nusselt Number with Magnetic Interaction (N). It is clear that even substantial levels of magnetic interaction would affect the heat transfer rate only marginally.

Fig. 5.40 shows, in a similar fashion, the corresponding variation of overall Sherwood Number for mass transfer from the cylinder. This shows a considerably stronger dependance on the magnetic interaction, the strongest dependency being demonstrated by the drag coefficient on magnetic interaction (Fig. 5.41). This last result would indicate the need for caution when interpreting results obtained by using drag probes to measure flow fields in the presence of magnetic fields. Also shown in this last figure are the results of Kalis et. al.(87) . The discrepancy between the two has been shown to be probably due to the use of a rather coarse computational grid near the cylinder wall.

5.5 Calibration Drift Correction

Figs. 5.42 and 5.43 relate to typical results obtained from the calibration drift correction routine described in Chapter 4. Fig. 5.42 shows a typical calibration drift if plotted using as-measured values of voltage increment against flow velocity. Fig. 5.43 shows the same basic data but after the voltages have been adjusted to correct for drift. The improved reproducibility of the corrected calibration is evident.

5.6 Summary

To summarise, it may be said that the $\kappa - \epsilon$ model as well as the XI model predict mean flow quite well and turbulence dissipation rates only reasonably well but RMS fluctuating velocities are not as well predicted. The MHD flow analysis, though restricted to orthogonal fields, indicates negligible effects of magnetic fields on anemometer behaviour in fields of the range found in this work. The PI criterion has been shown to predict the onset of turbulence reasonably well, though somewhat constrained in its use by the need to specify a value for η . The calibration drift correction scheme has been shown to improve the reproducibility of calibration curves considerably. The theory proposed for turbulent mixing in metallurgical melts appears to correlate experimental and plant scale data reasonably well.

Fig. 5.1 Computed Current Density Vector Plot

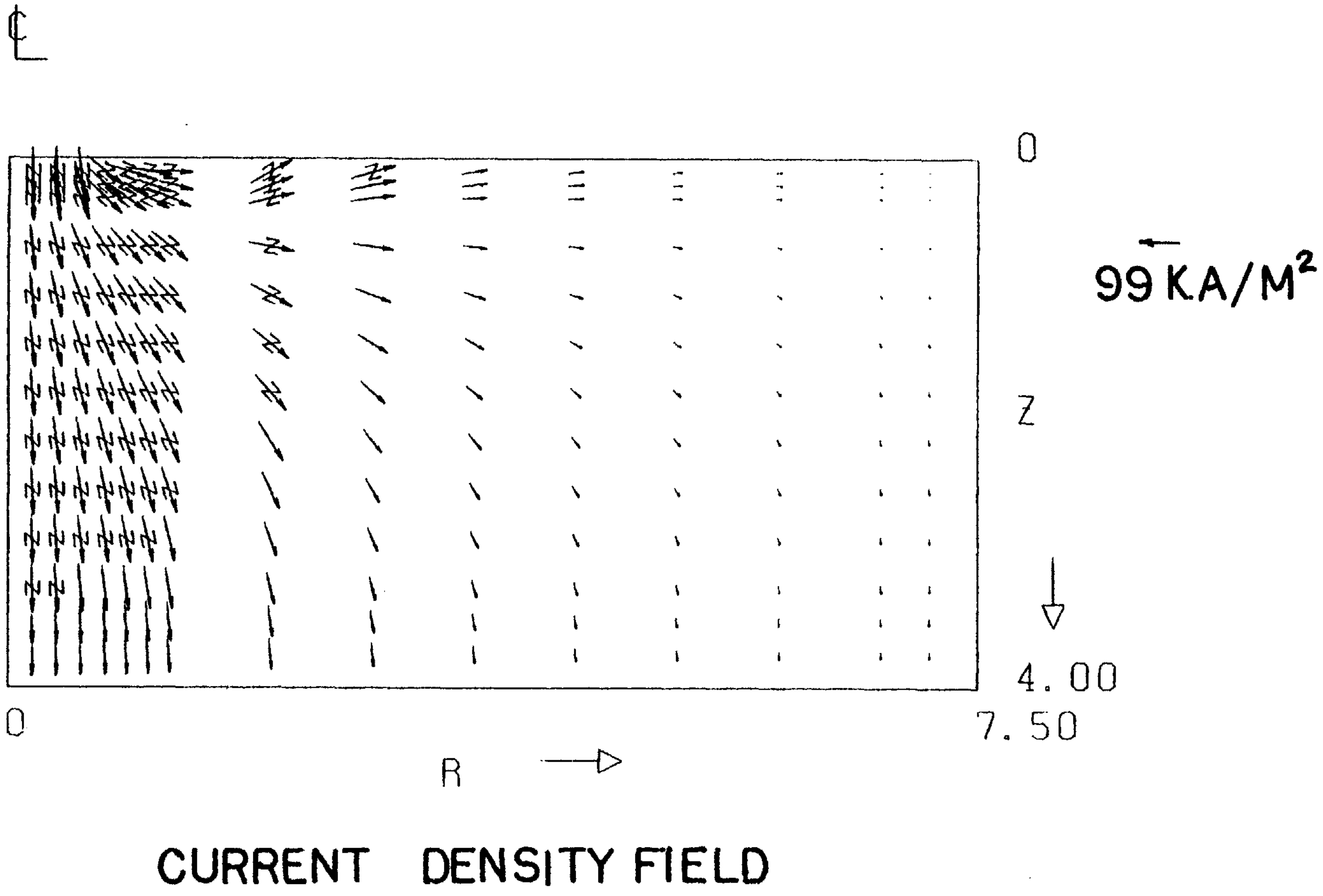
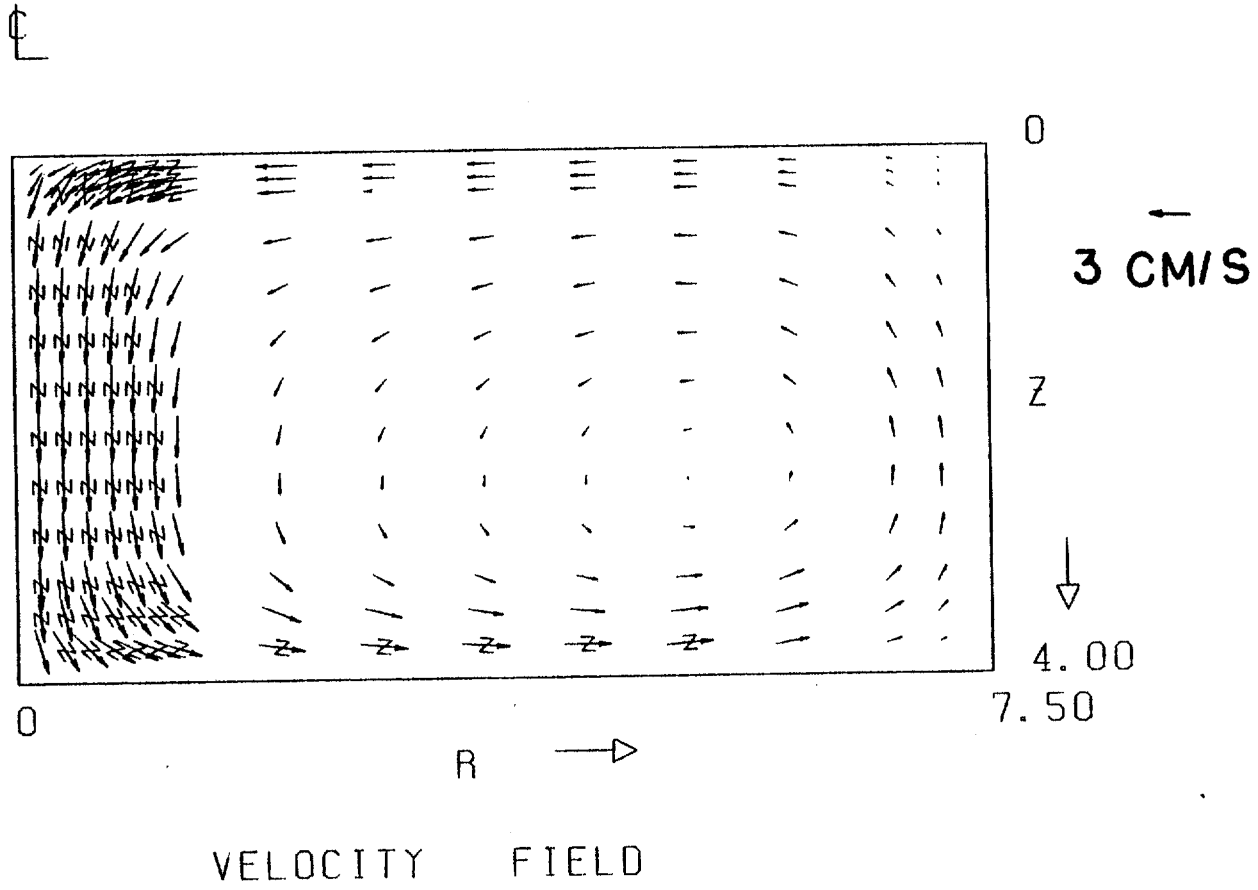


Fig. 5.2 Computed Velocity Vector Plot



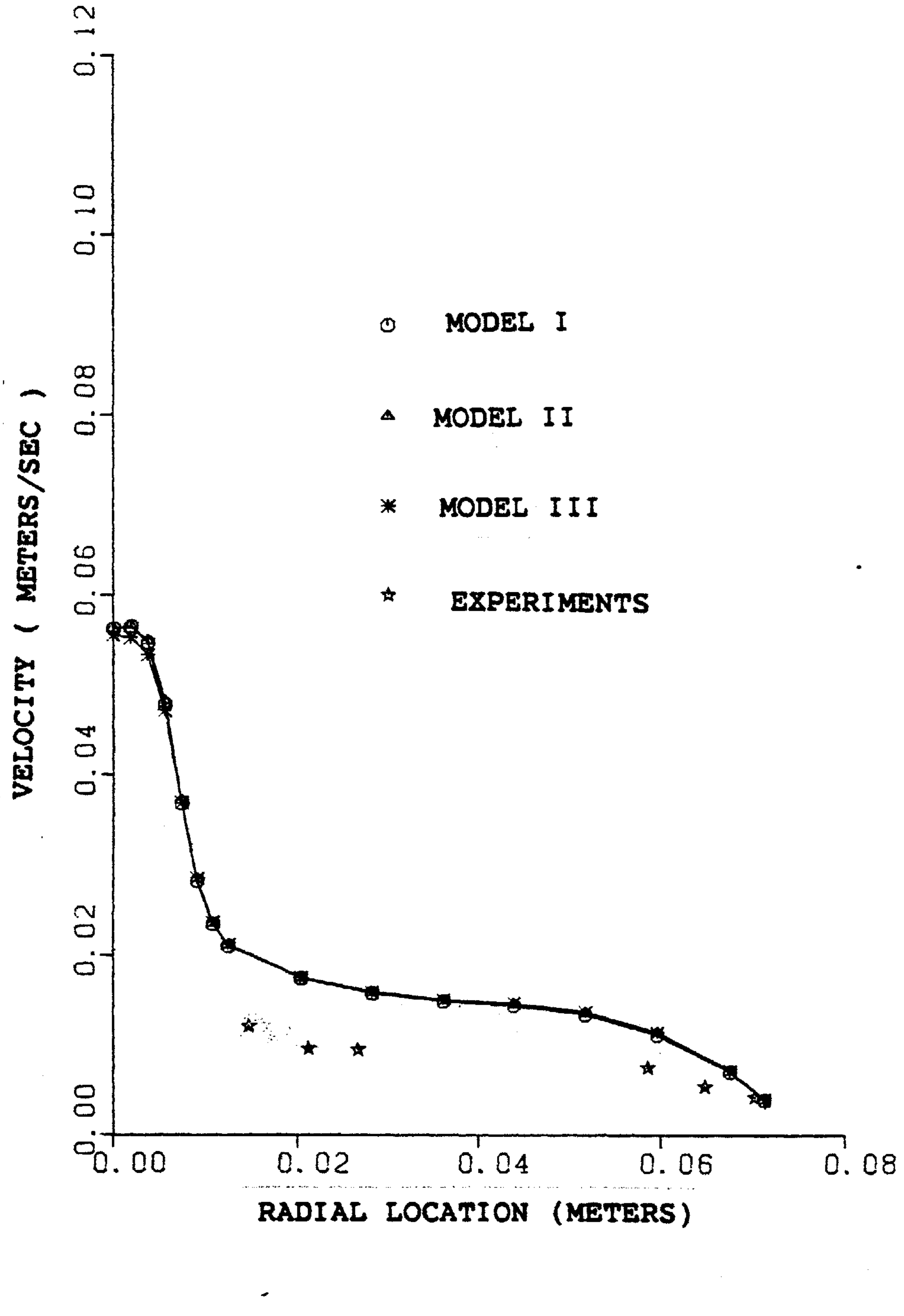


Fig. 5.3 Magnitude of Velocity, Radial Scan - Depth = 0.7 cms

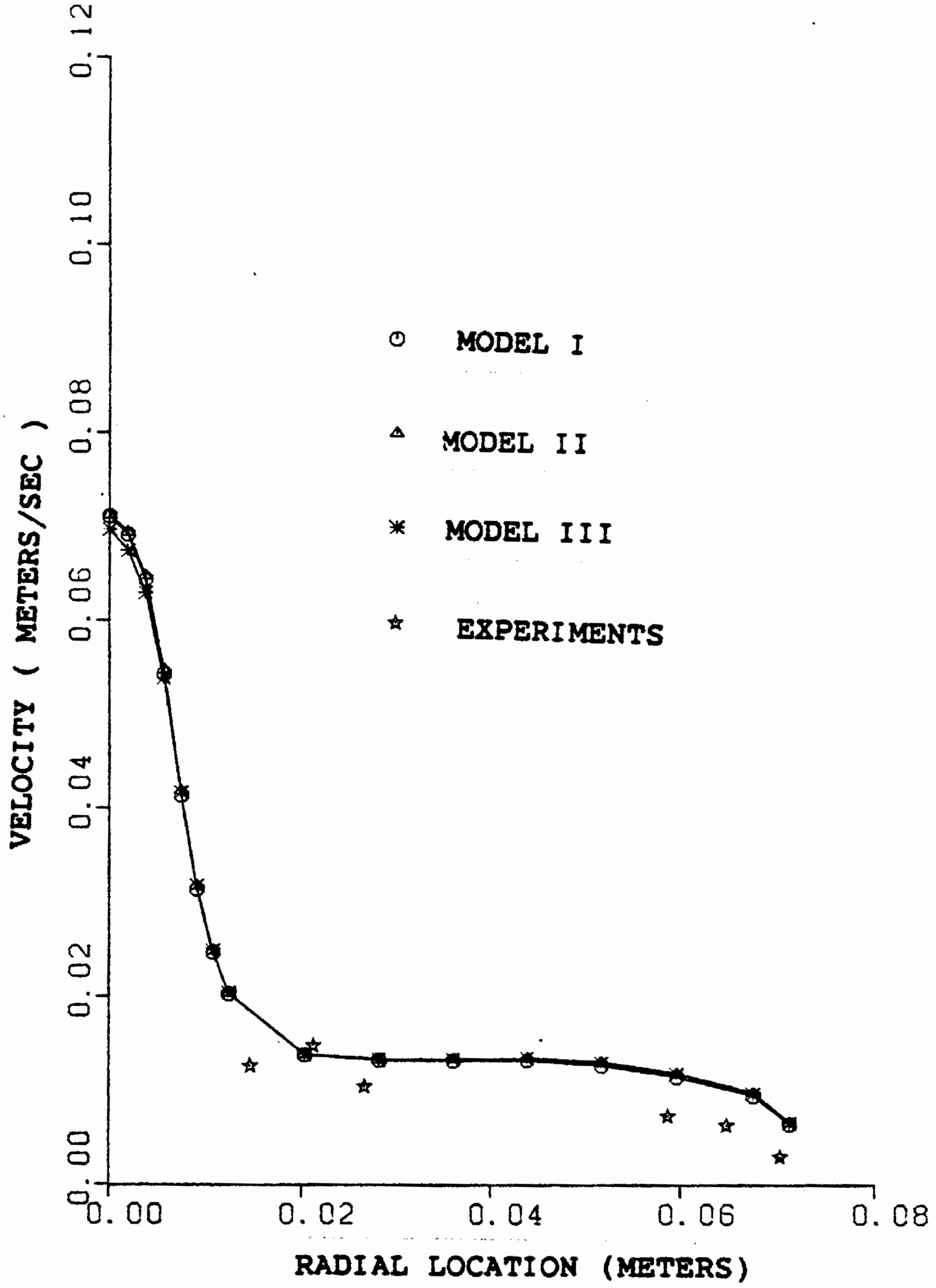


Fig. 5.4 Magnitude of Velocity, Radial Scan - Depth = 1.05 cms

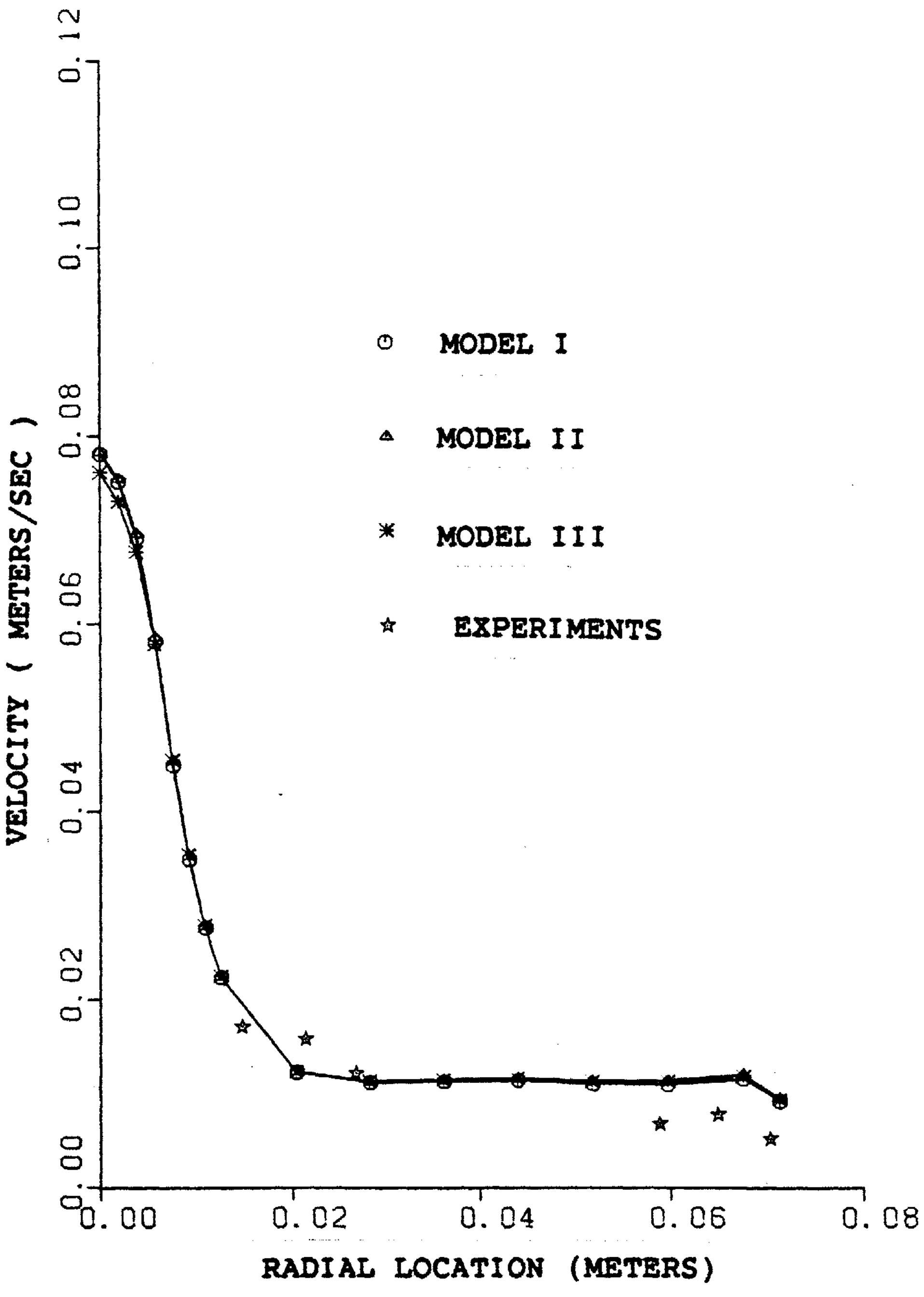


Fig. 5.5 Magnitude of Velocity, Radial Scan - Depth = 1.44 cms

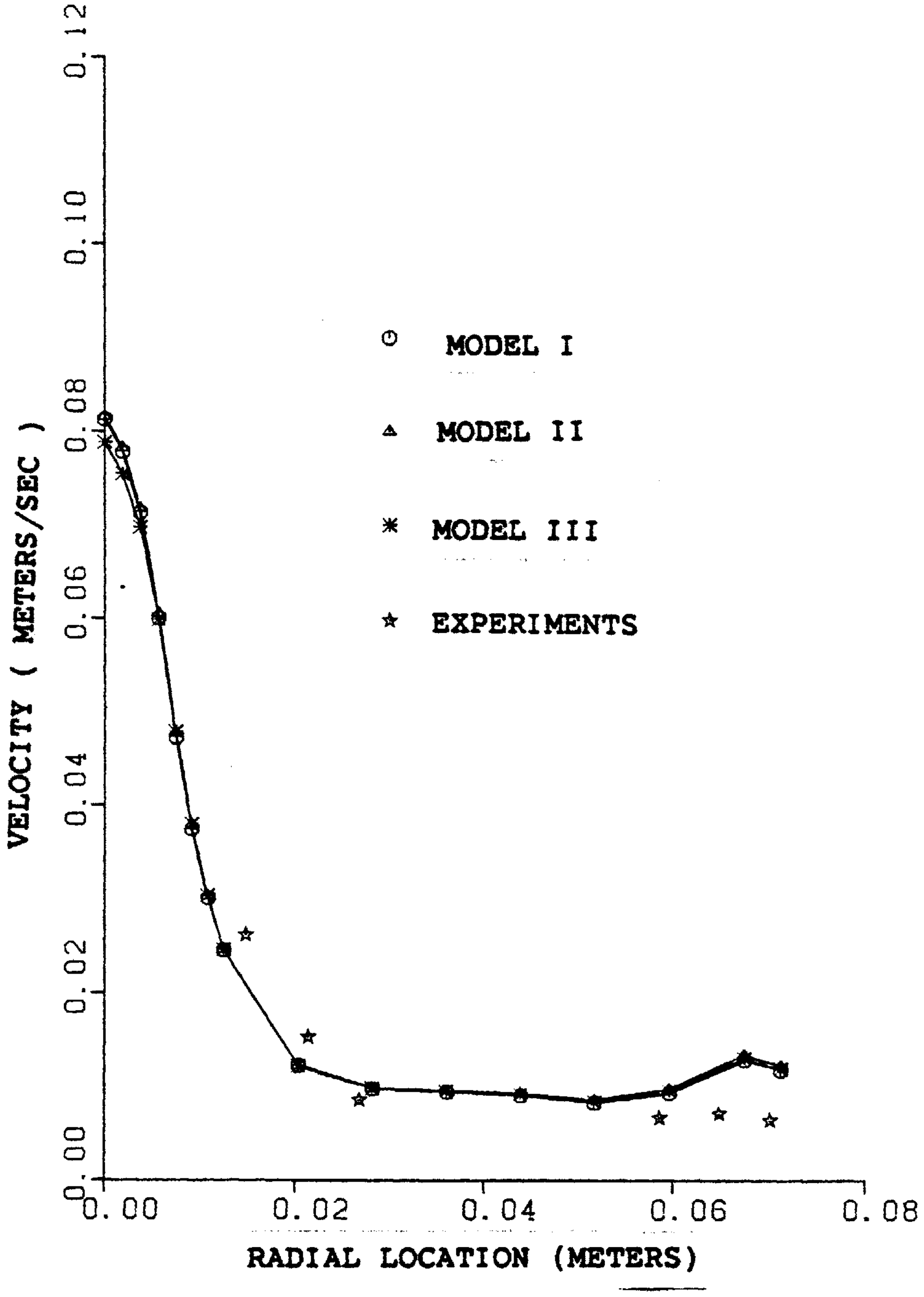


Fig. 5.6 Magnitude of Velocity, Radial Scan - Depth = 1.79 cms

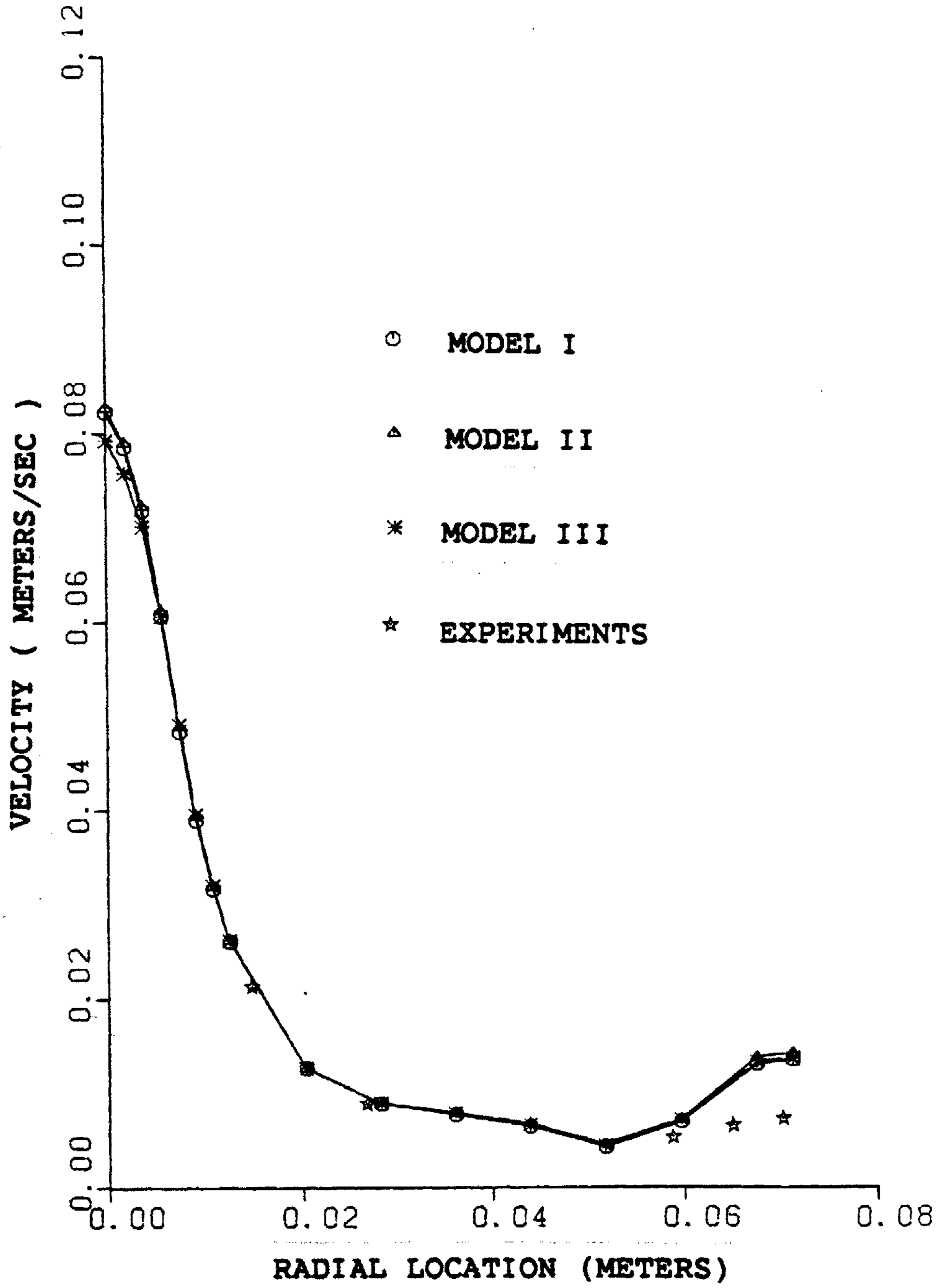


Fig. 5.7 Magnitude of Velocity, Radial Scan - Depth = 2.14 cms

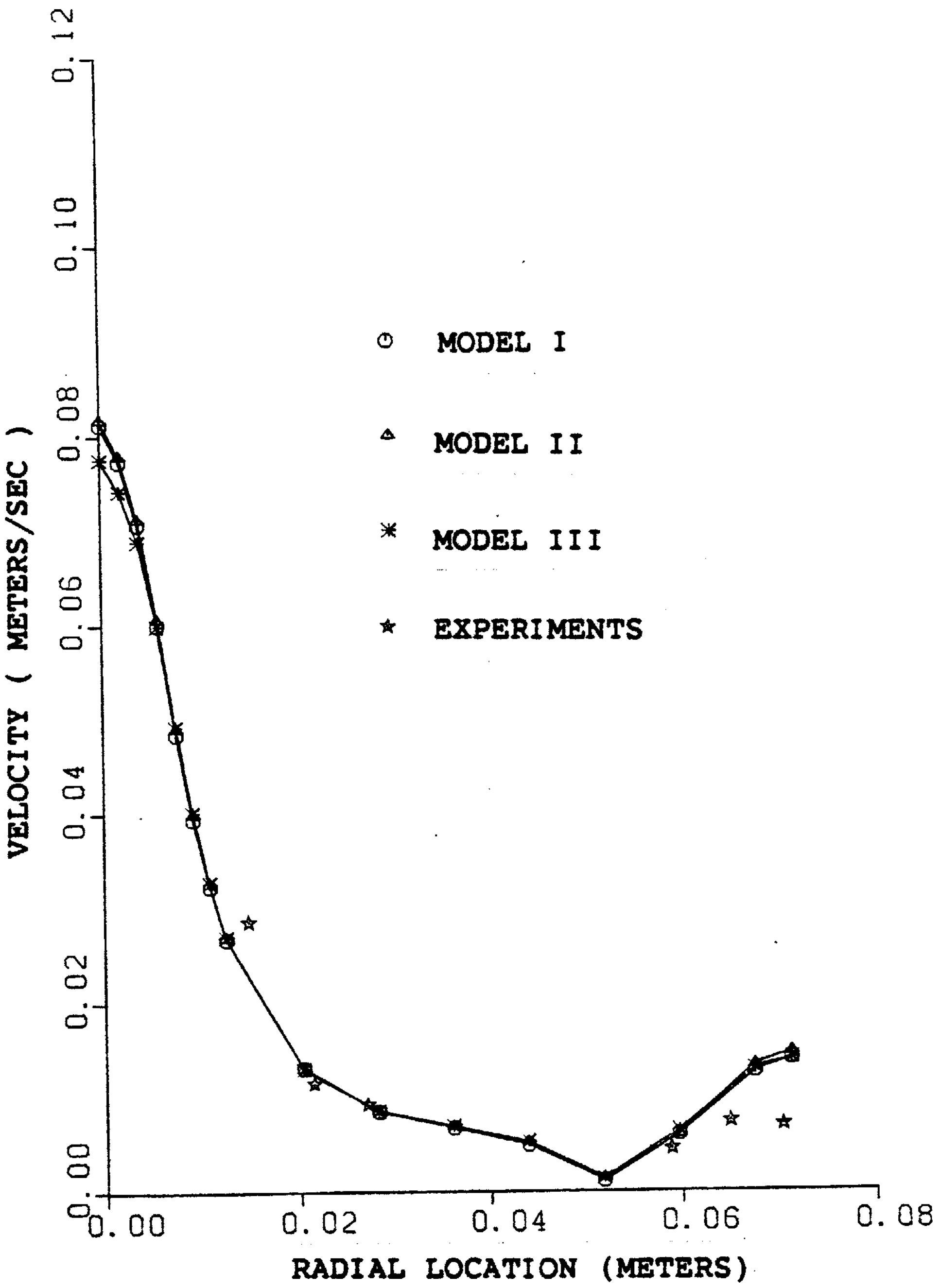


Fig. 5.8 Magnitude of velocity, Radial Scan - Depth = 2.5 cms

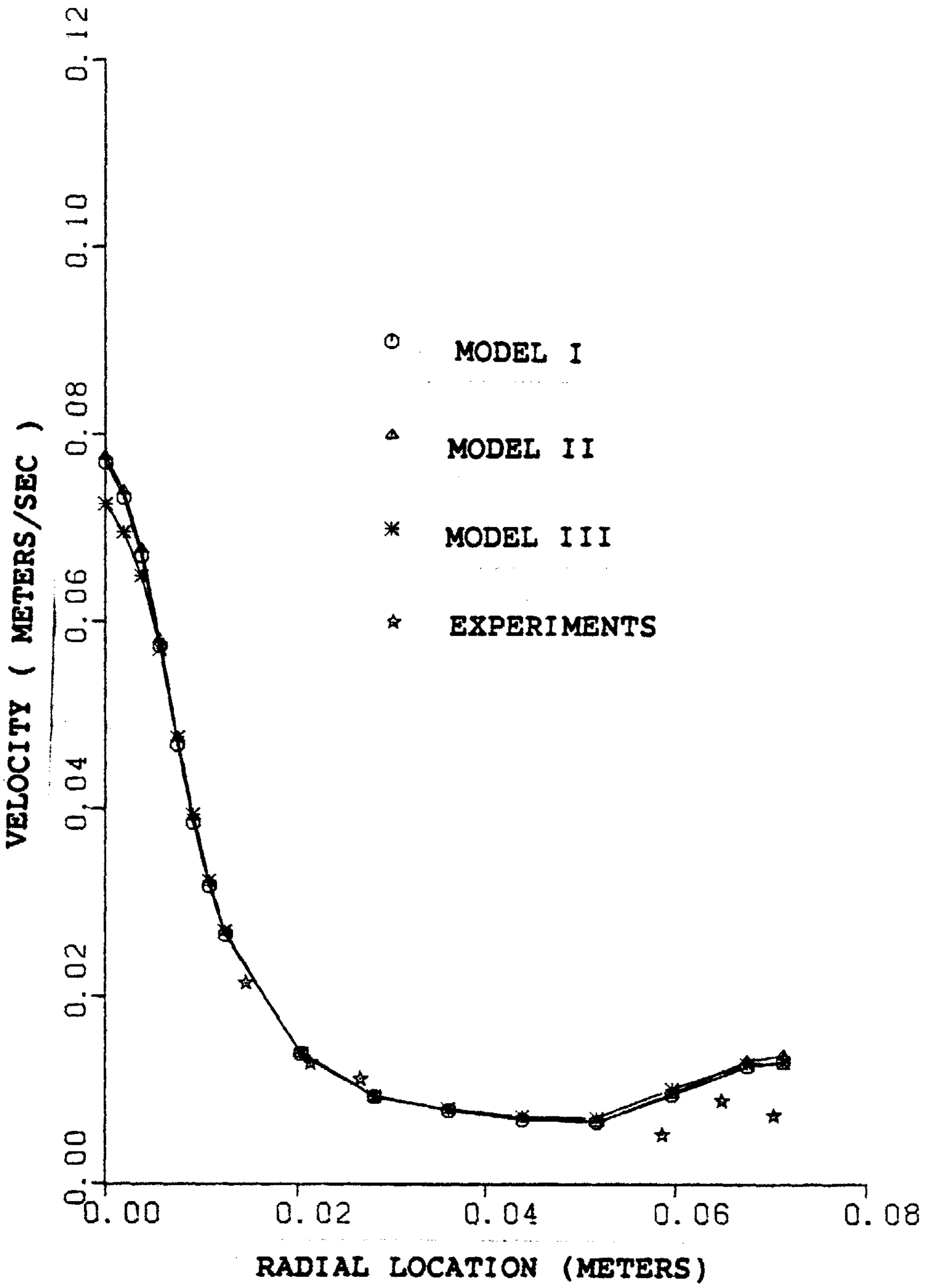


Fig. 5.9 Magnitude of velocity, Radial Scan - Depth = 2.85 cms

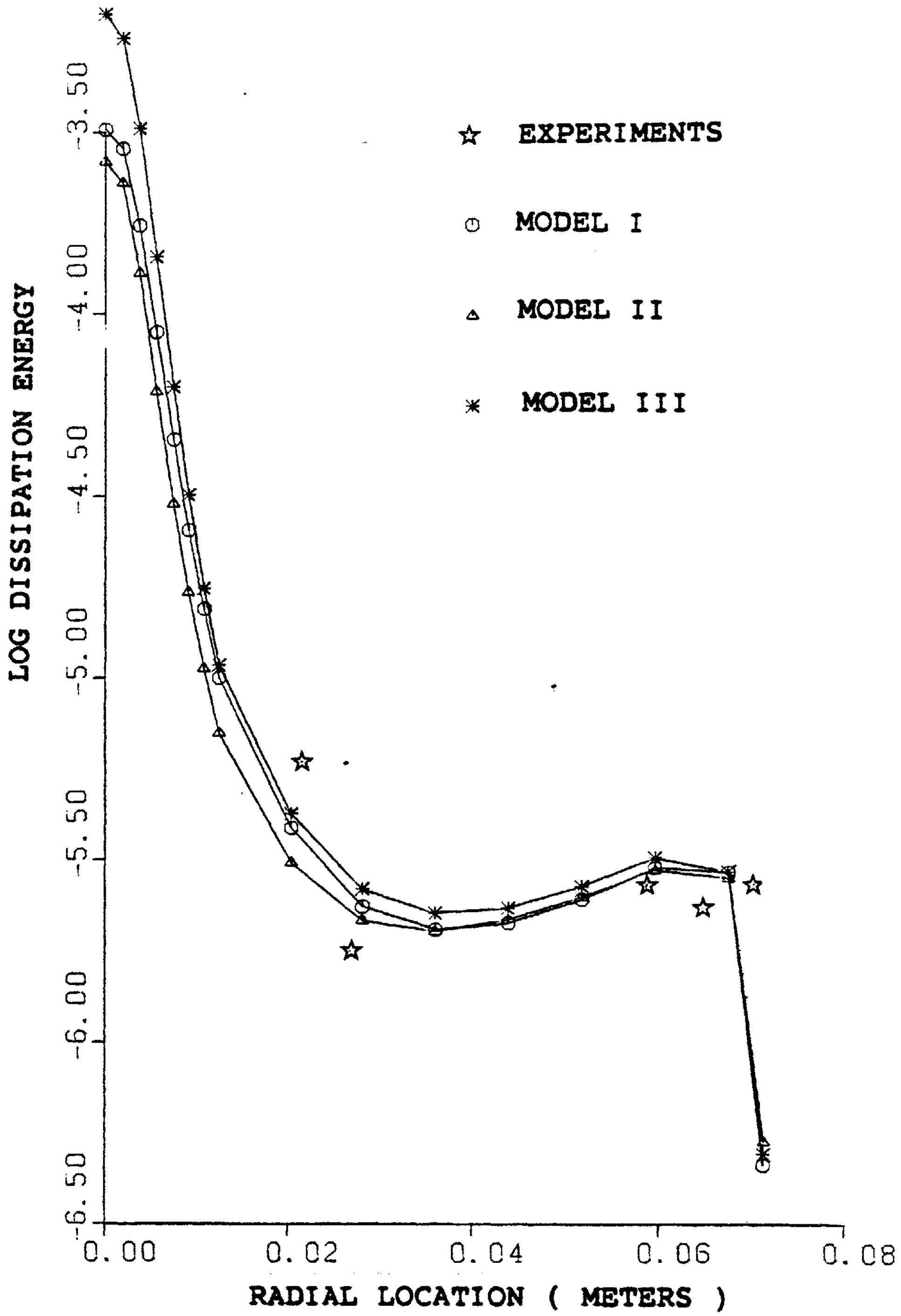


Fig. 5.10 Turbulence Dissipation Energy (W/Kg.s) Depth = 0.70 cms

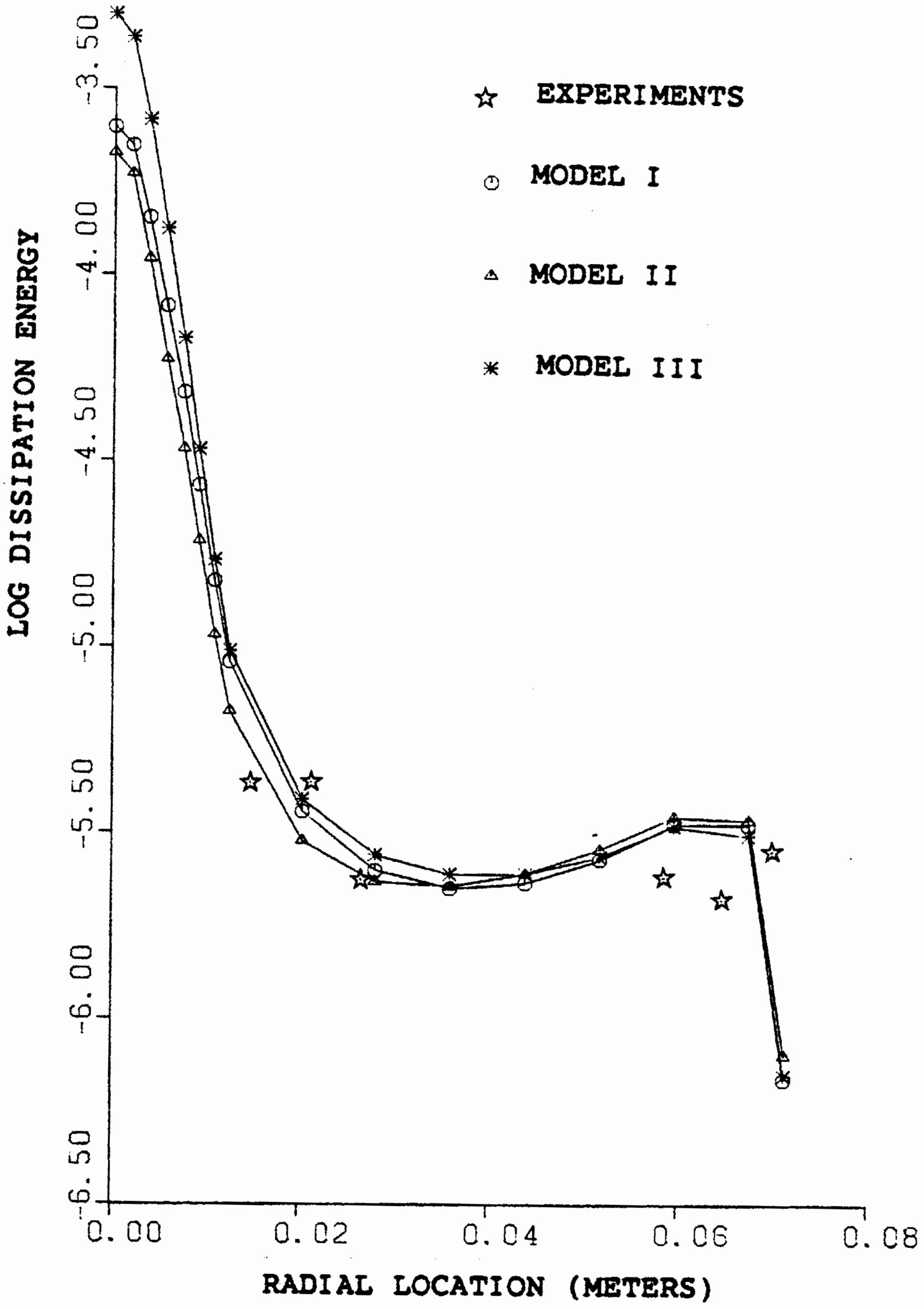


Fig. 5.11 Turbulence Dissipation Energy (W/Kg.s) Depth = 1.05 cms

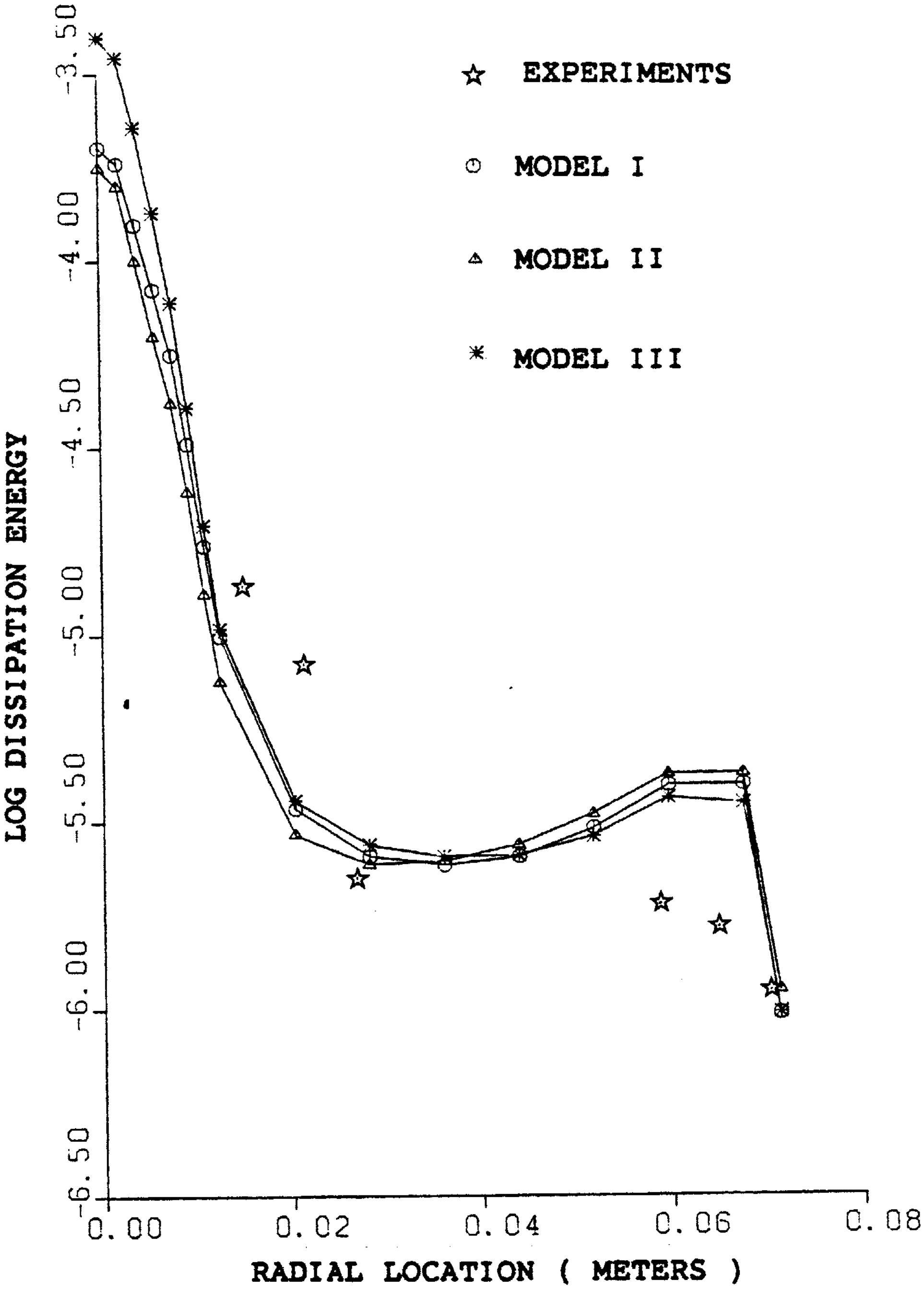


Fig. 5.12 Turbulence Dissipation Energy (W/Kg.s) Depth = 1.44 cms

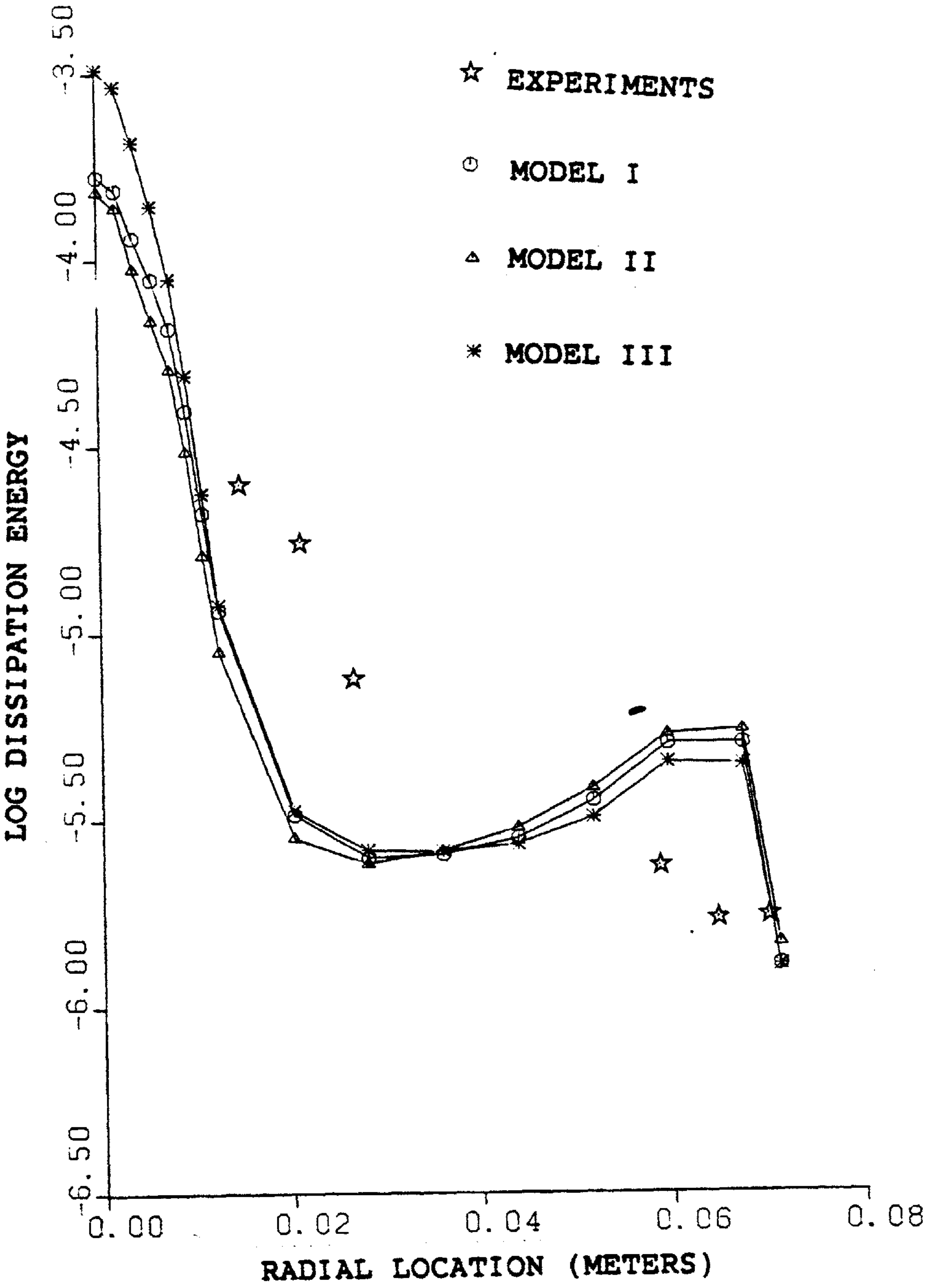


Fig. 5.13 Turbulence Dissipation Energy (W/Kg.s) Depth = 1.79 cms

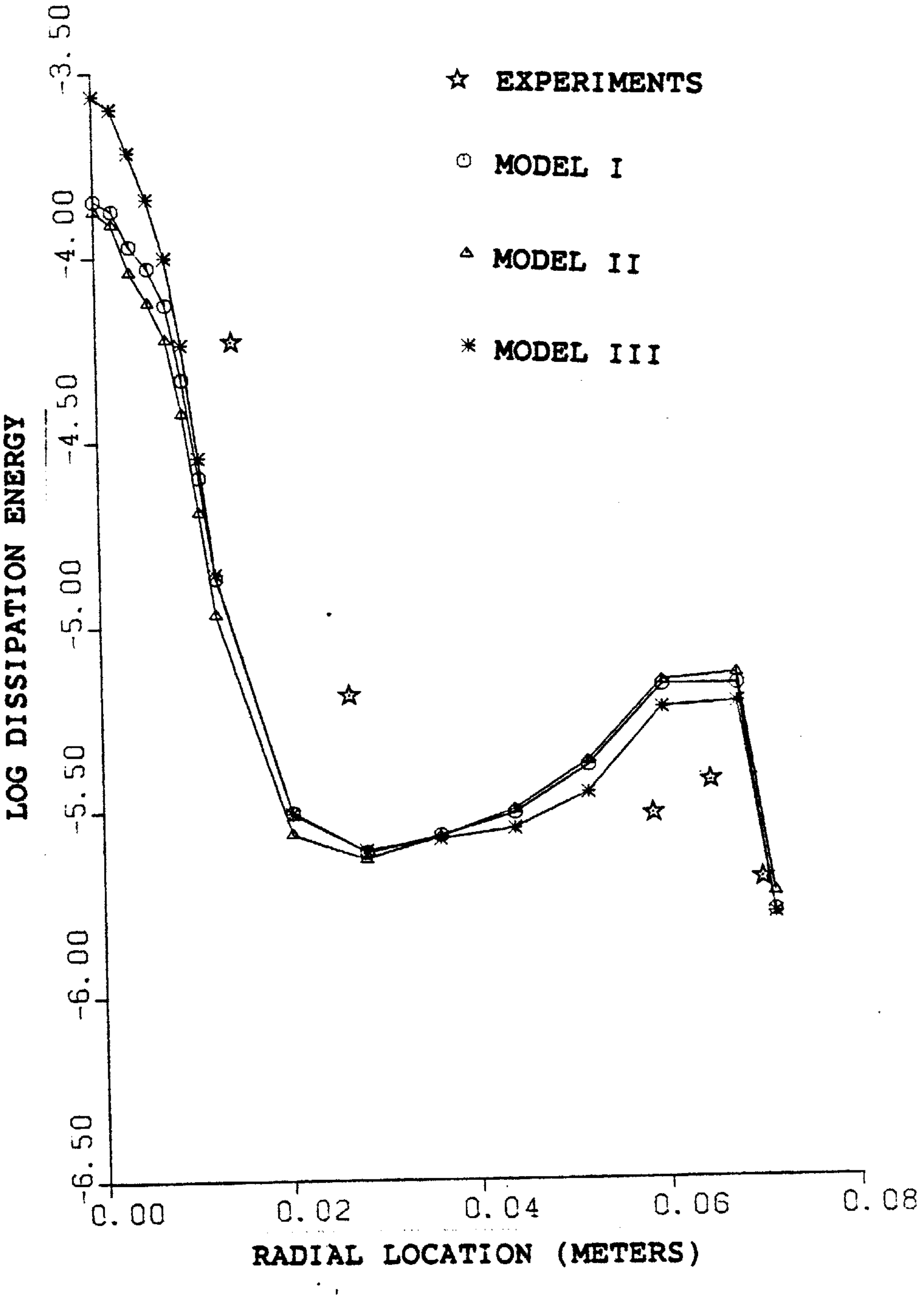


Fig. 5.14 Turbulence Dissipation Energy (W/Kg.s) Depth = 2.14 cms

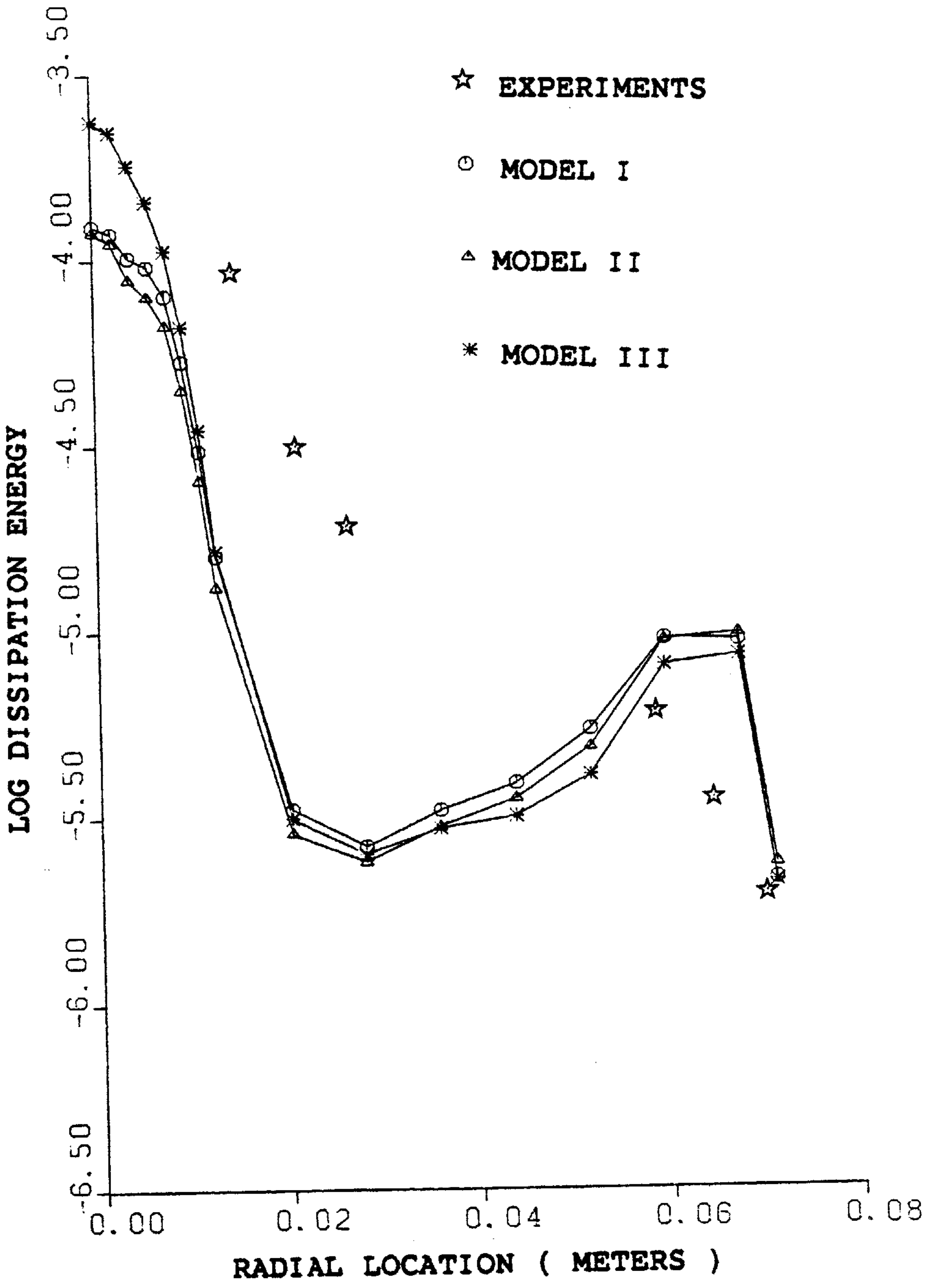


Fig. 5.15 Turbulence Dissipation Energy (W/Kg.s) Depth = 2.50 cms

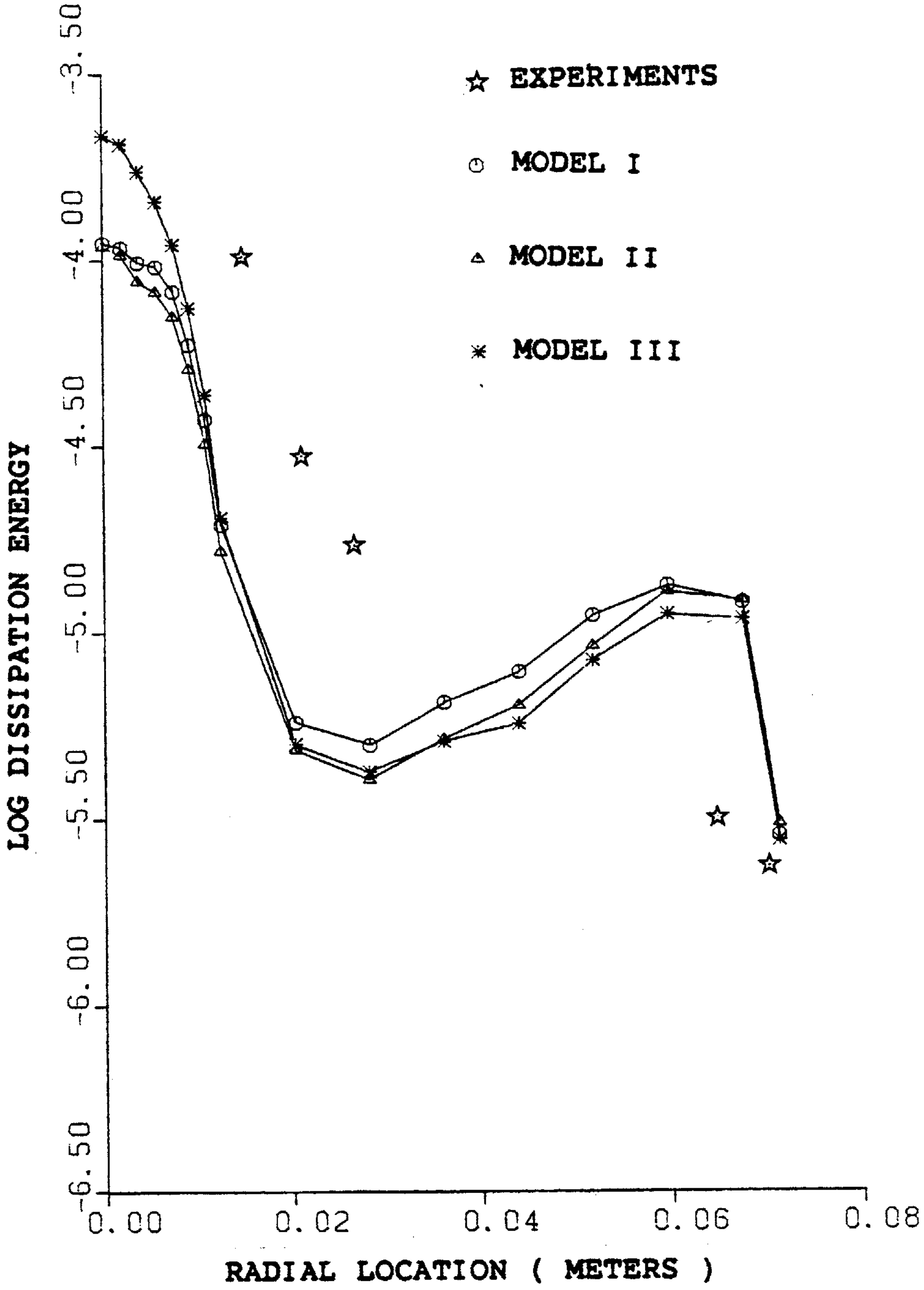


Fig. 5.16 Turbulence Dissipation Energy (W/Kg.s) Depth = 2.85 cms

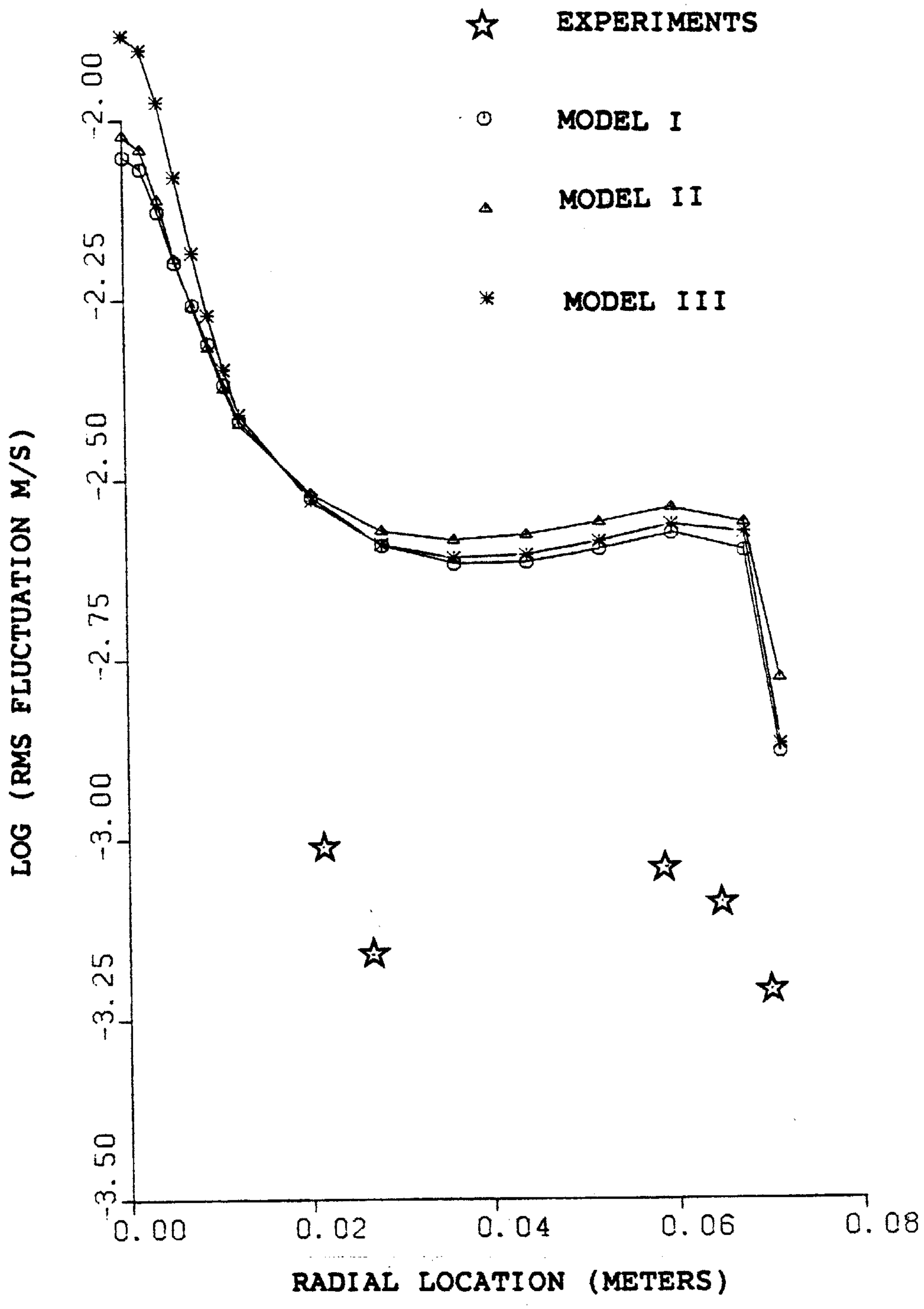


Fig. 5.17 R.M.S. Turbulent Velocity (M/sec) Depth = 0.70 cms

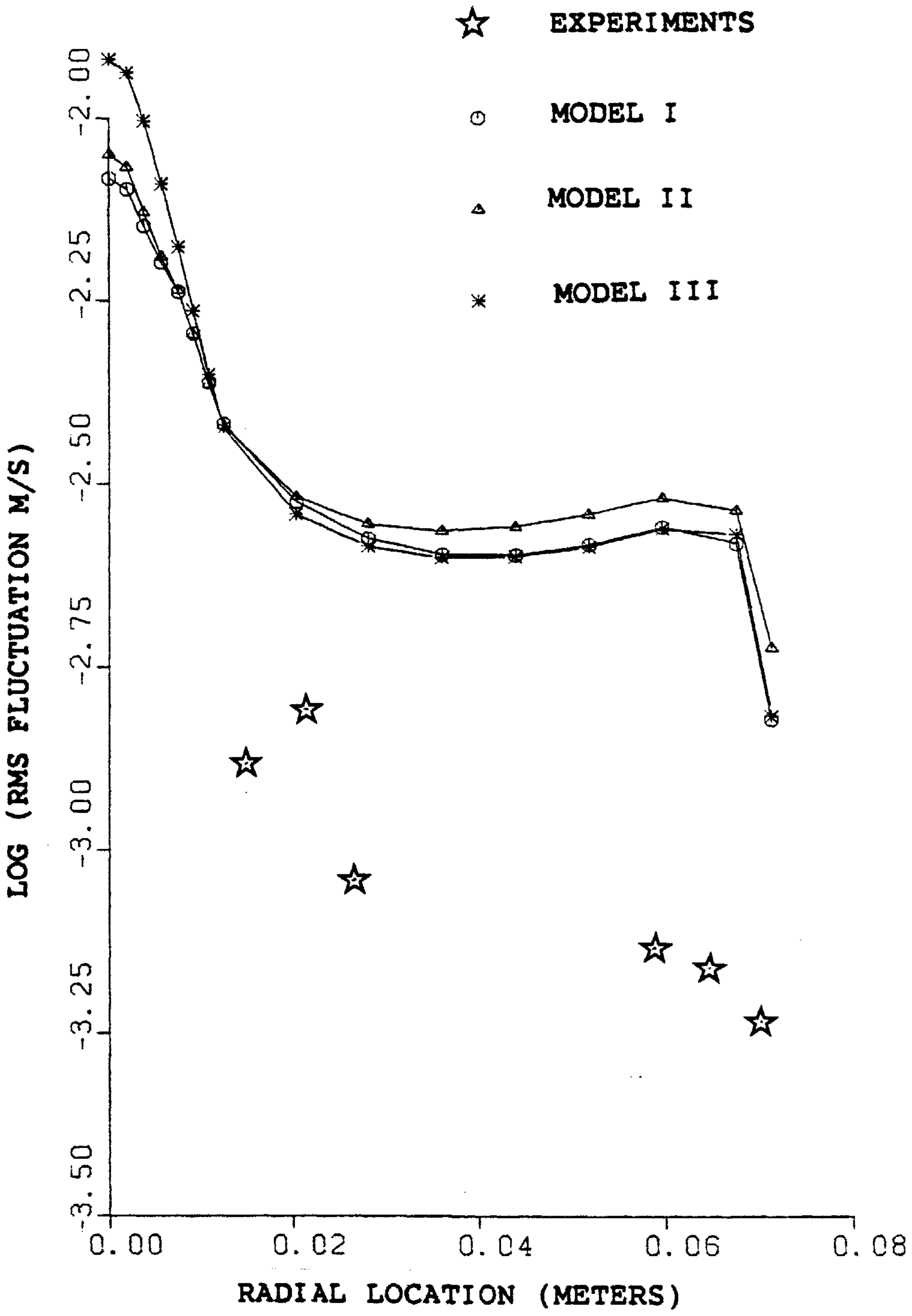


Fig. 5.18 R.M.S. Turbulent Velocity (M/sec) Depth = 1.05 cms

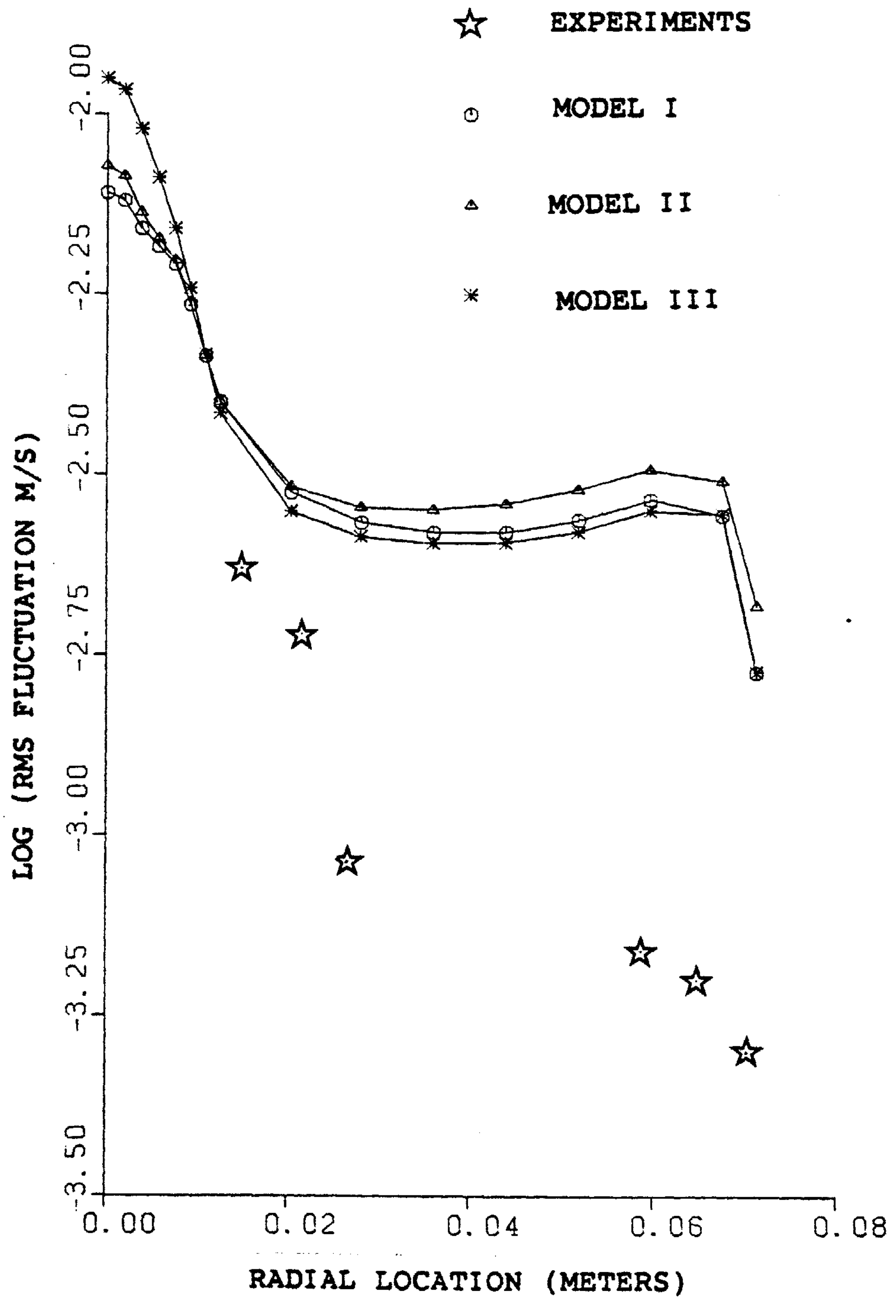


Fig. 5.19 R.M.S. Turbulent Velocity (M/sec) Depth = 1.44 cms

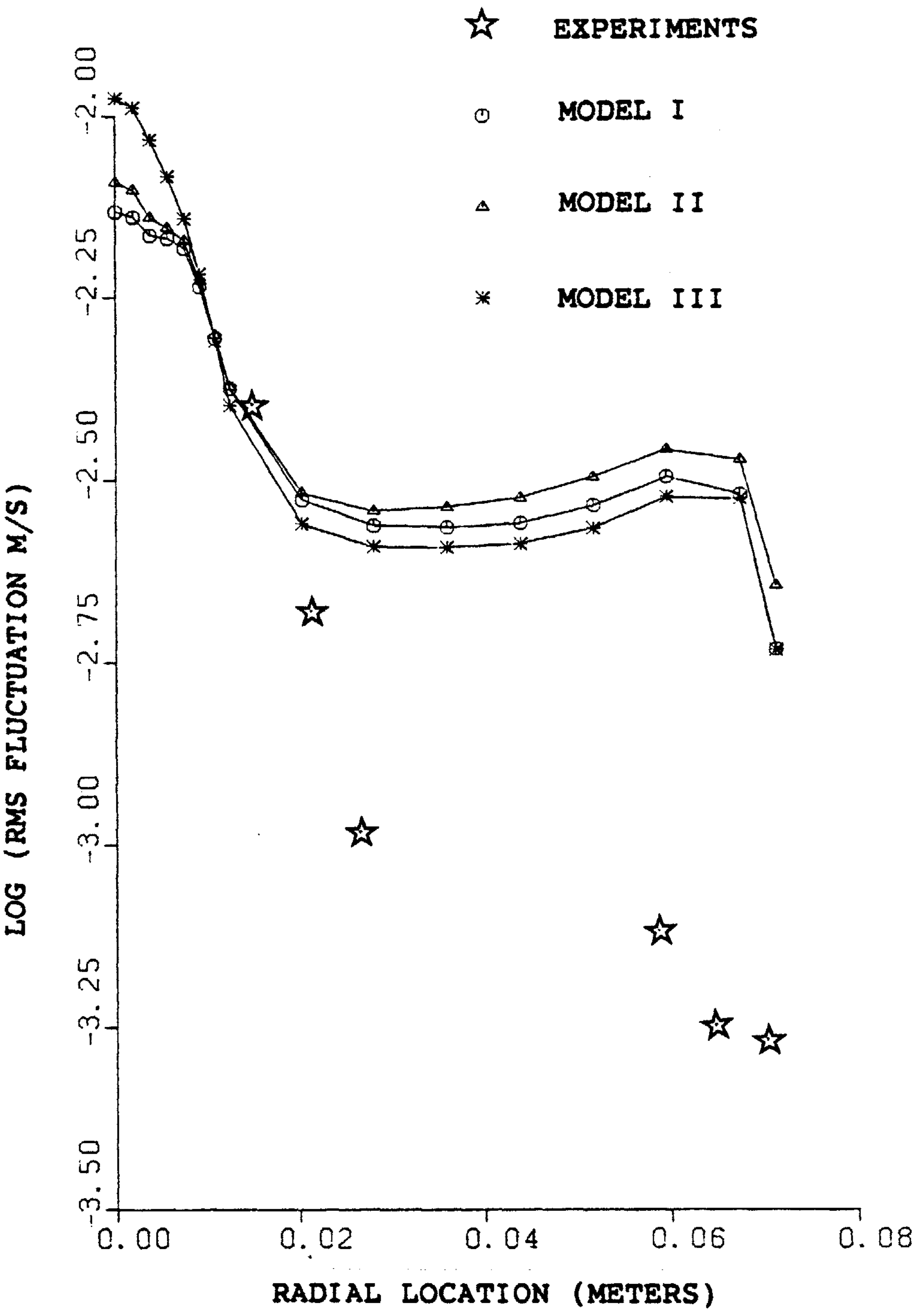


Fig. 5.20 R.M.S. Turbulent Velocity (M/sec) Depth = 1.79 cms

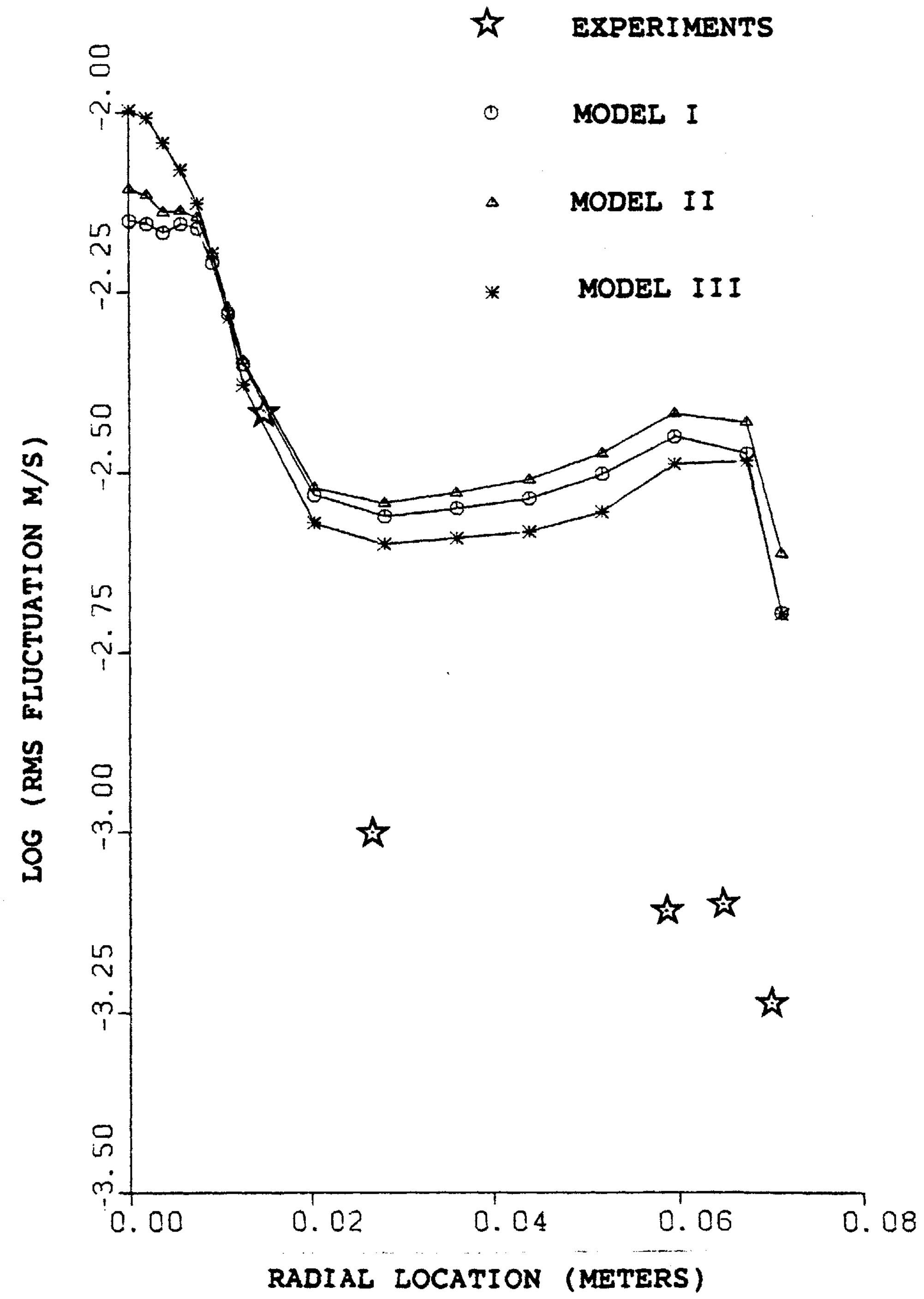


Fig. 5.21 R.M.S. Turbulent Velocity (M/sec) Depth = 2.14 cms

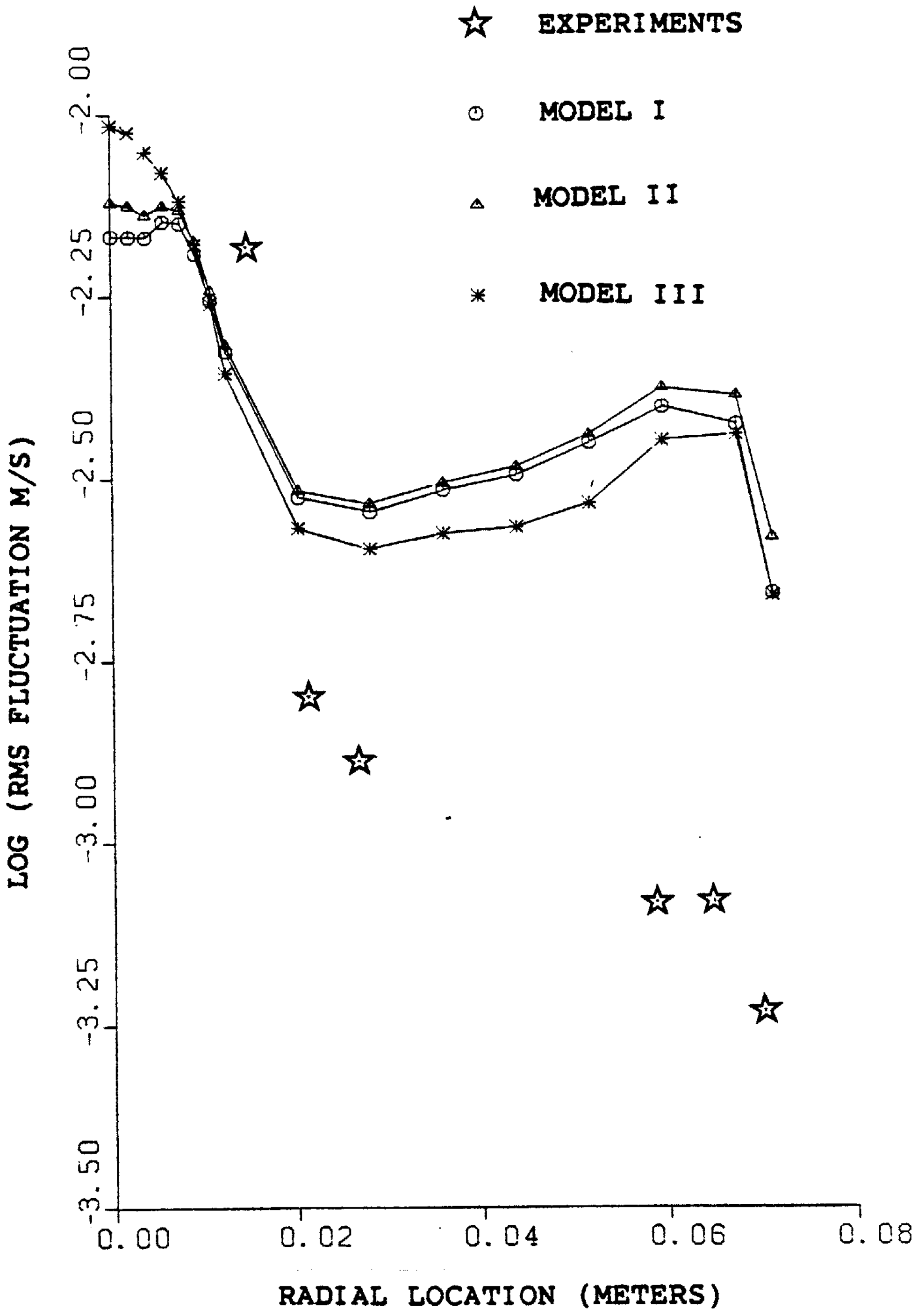


Fig. 5.22 R.M.S. Turbulent Velocity (M/sec) Depth = 2.50 cms

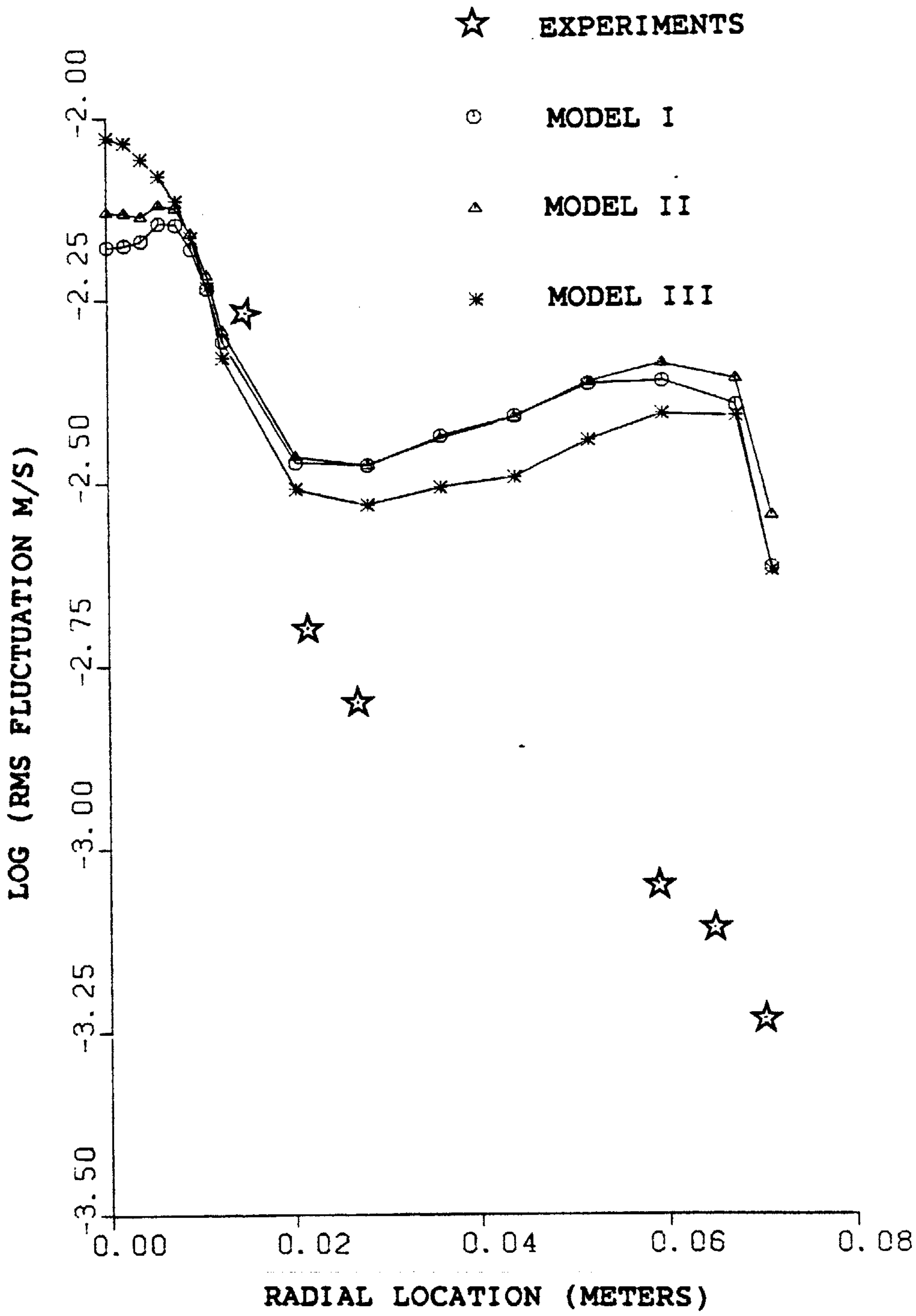


Fig. 5.23 R.M.S. Turbulent Velocity (M/sec) Depth = 2.85 cms

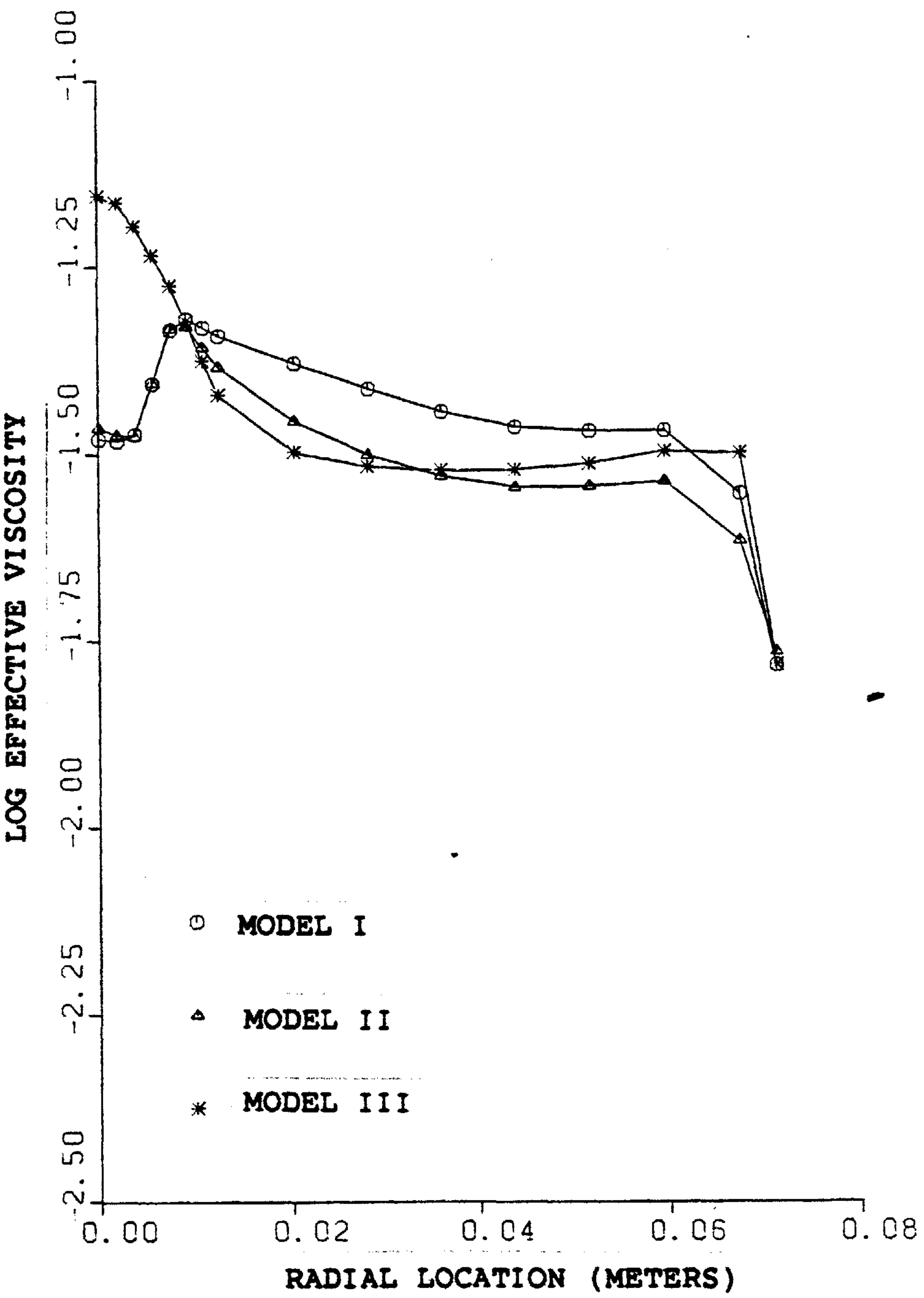


Fig. 5.24 Effective Viscosity (Kg/m-sec) Depth = 1.44 cms

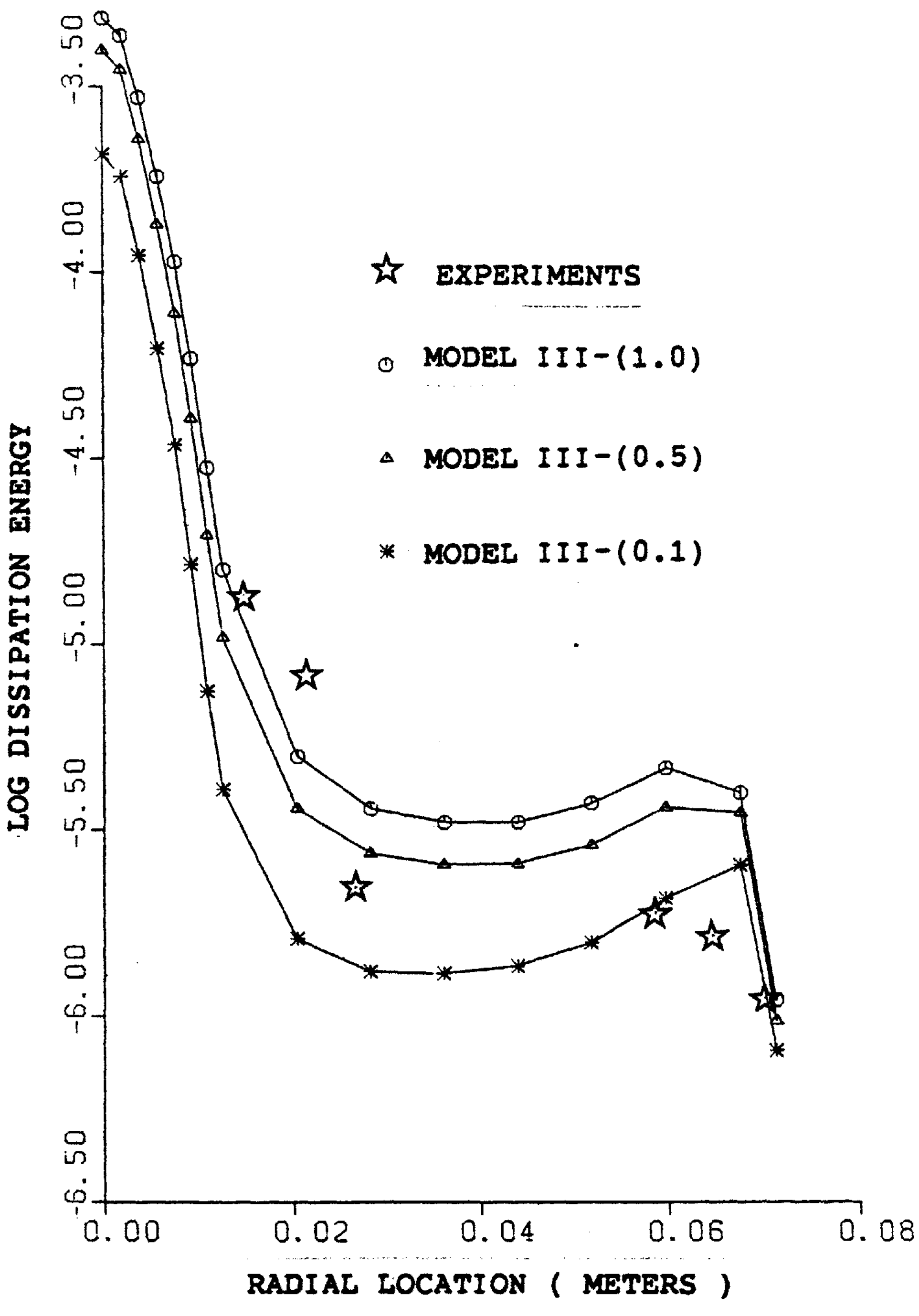


Fig. 5.25 Effect of Varying η on D.Energy Profile, Depth = 1.44 cms

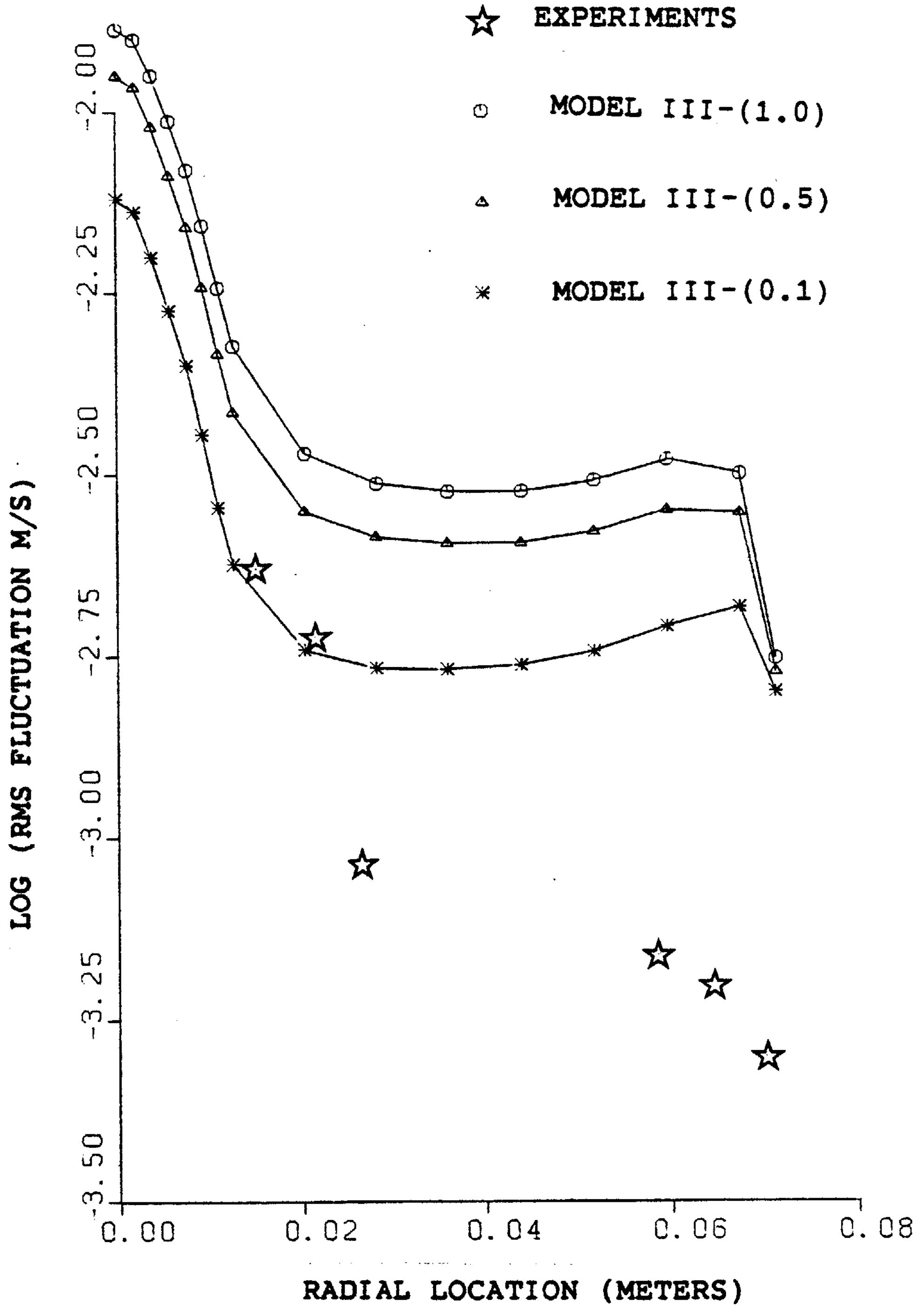


Fig. 5.26 Effect of Varying η on Fluc. Velocity, Depth = 1.44 cms

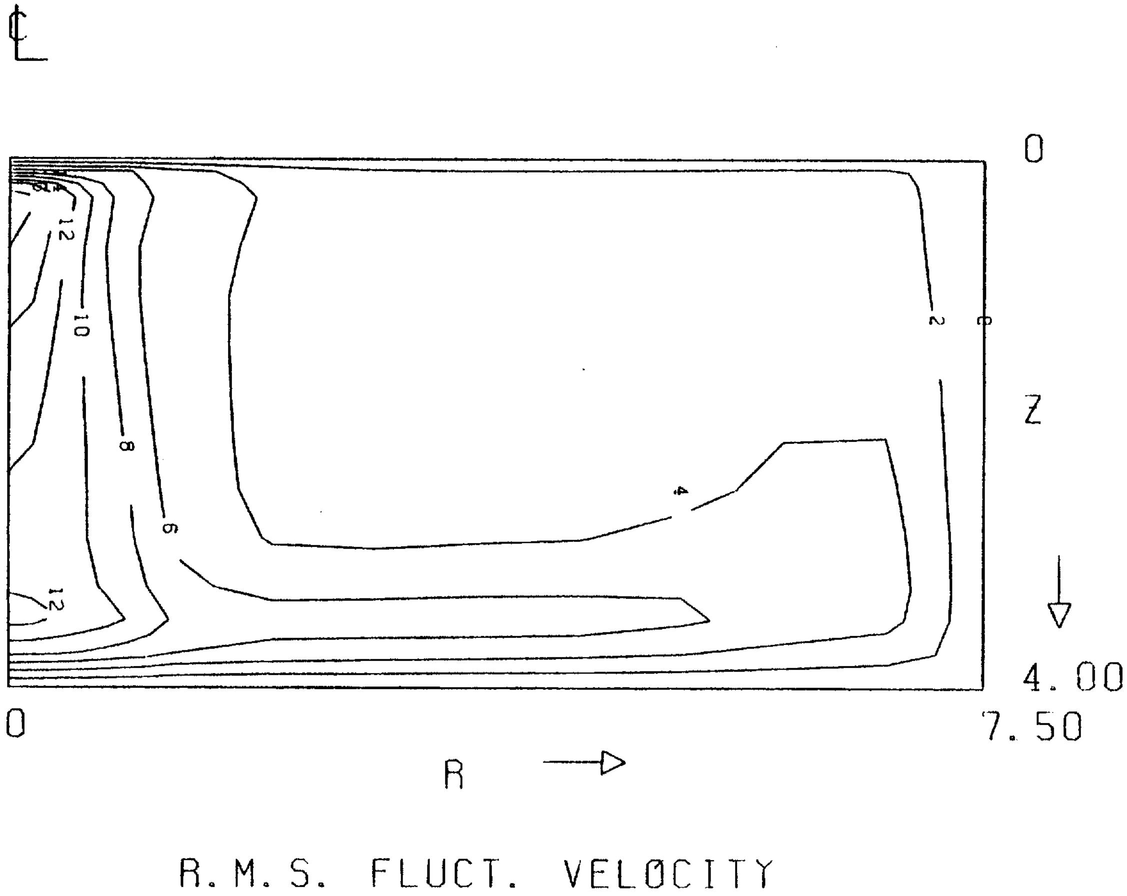


Fig. 5.27 Contour Plot of Fluc. Velocity distribution in Melt

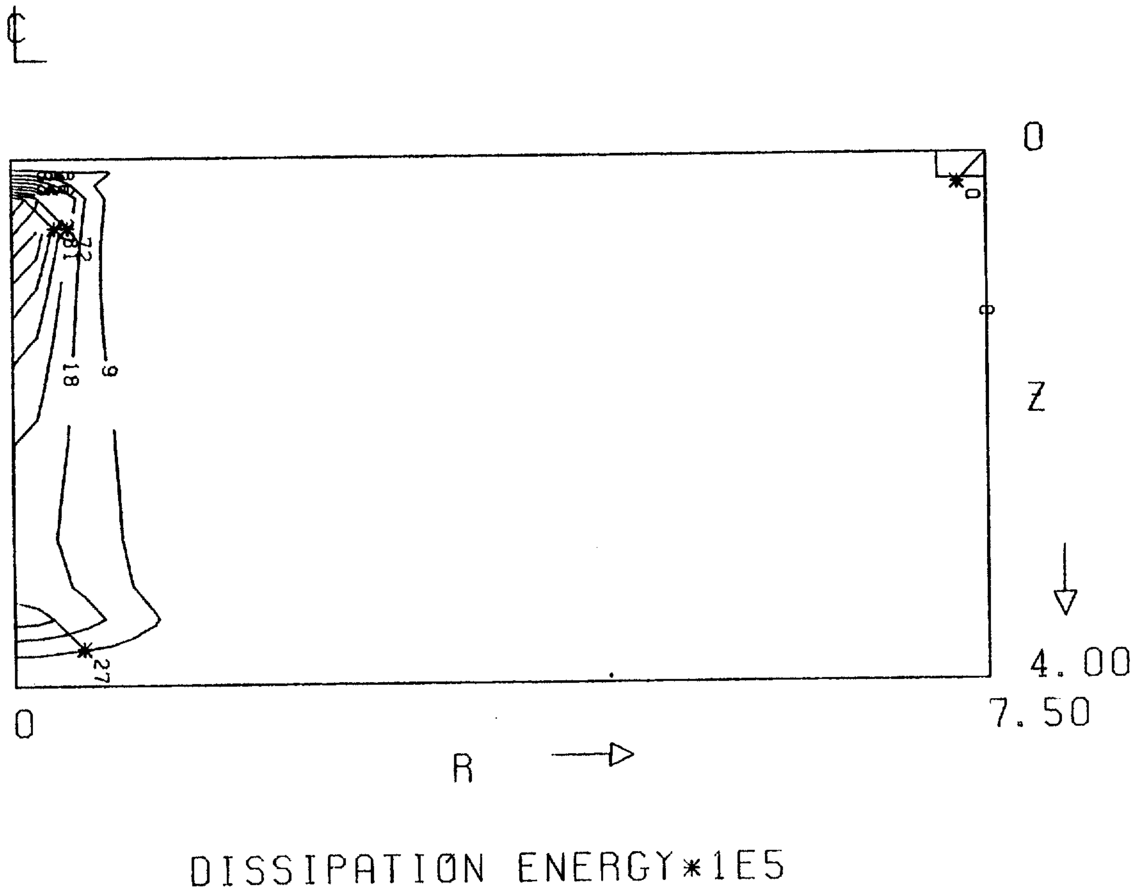


Fig. 5.28 Contour Plot of D. Energy Distribution in Melt

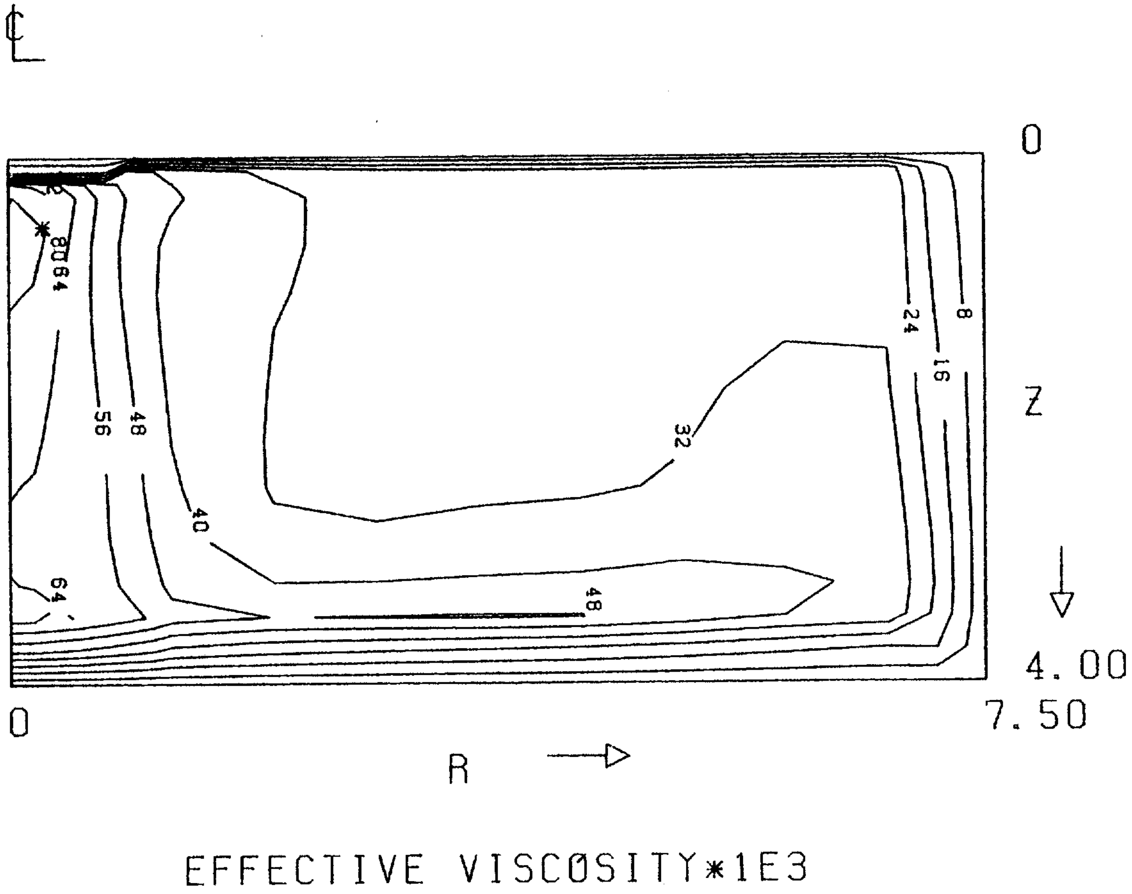


Fig. 5.29 Contour Plot of Effective Viscosity Distribution in Melt

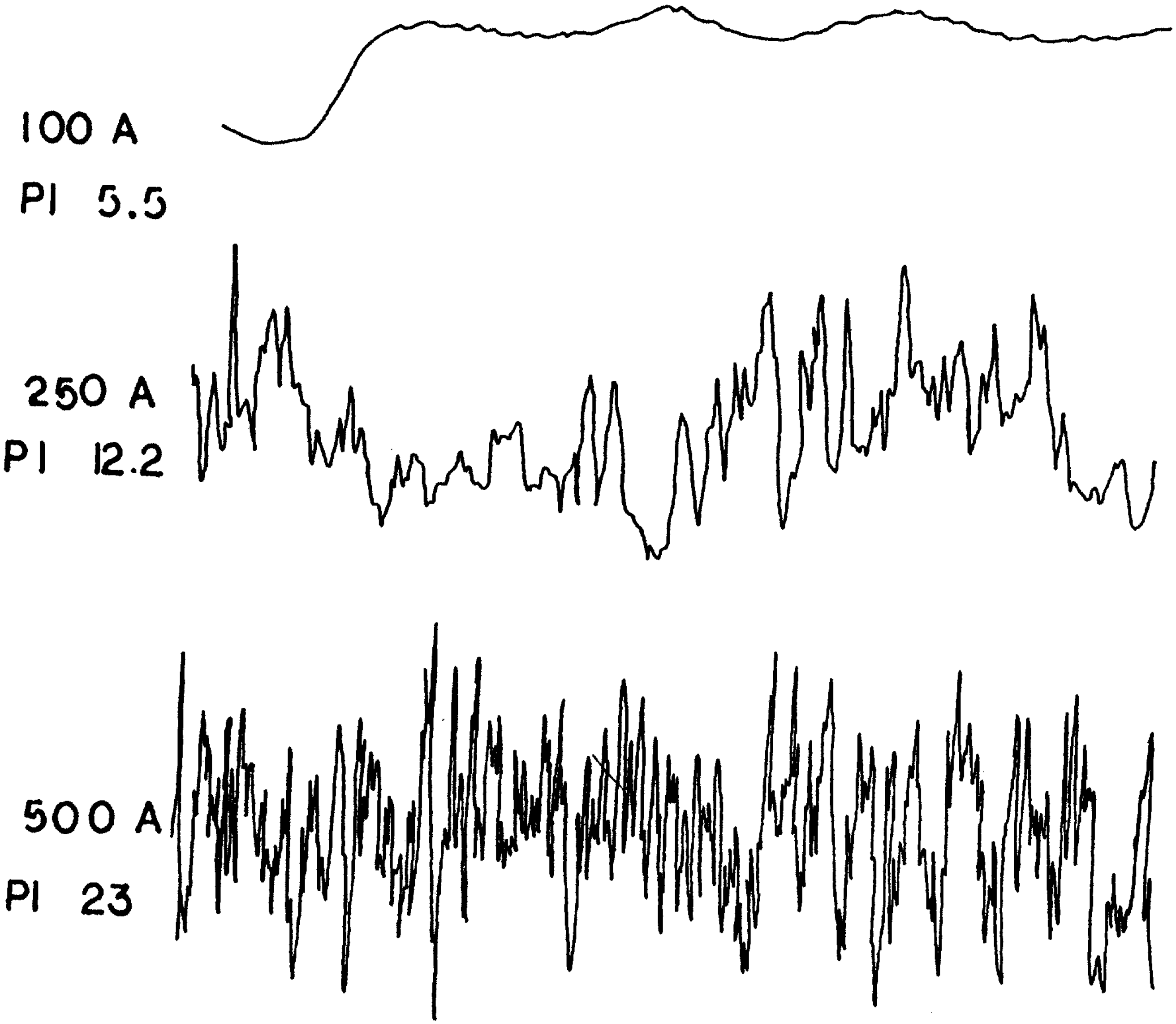


Fig. 5.30 Typical Anemometer Voltage outputs with computed values of PI

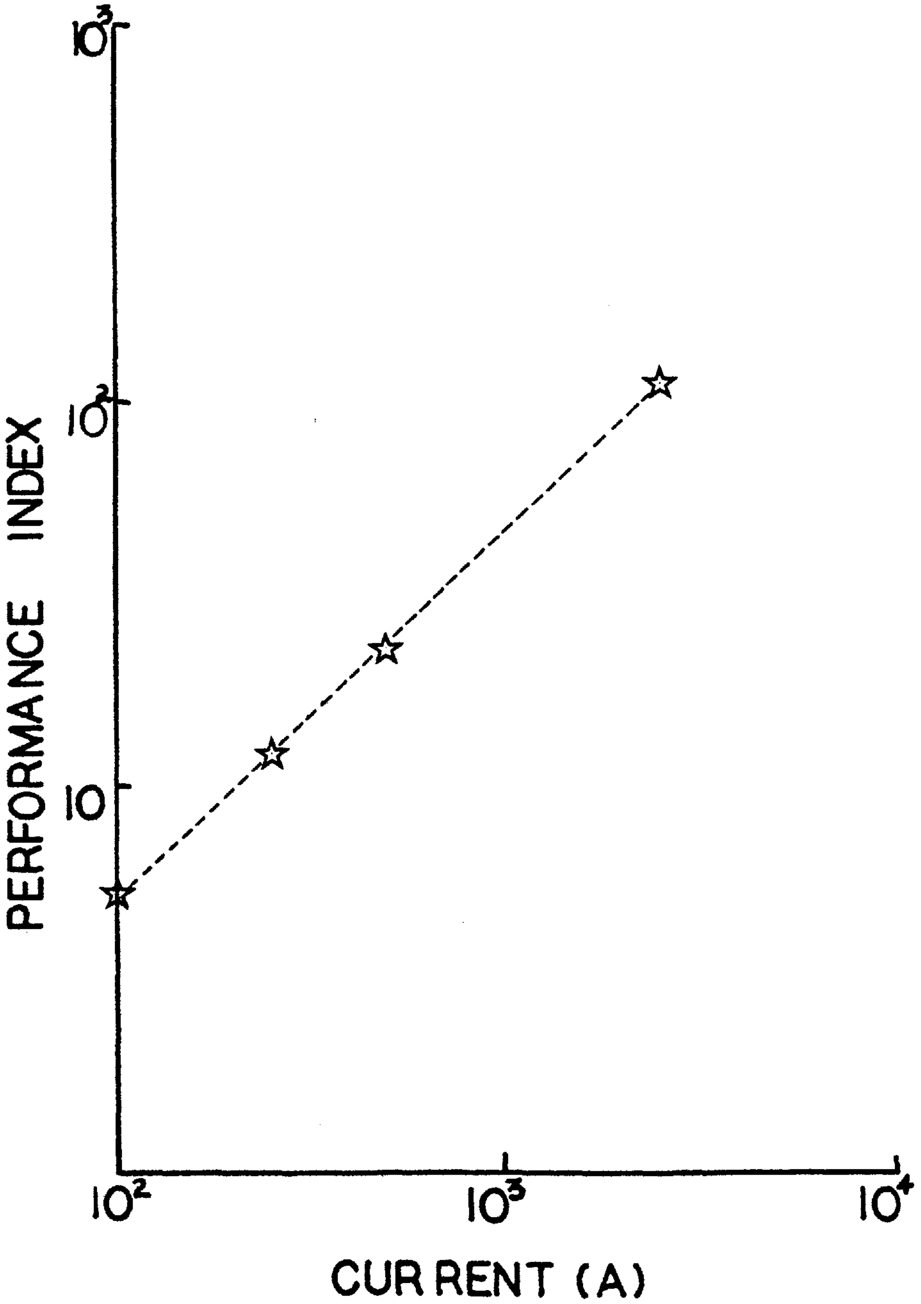


Fig. 5.31 Variation of Computed value of PI with current passed through melt

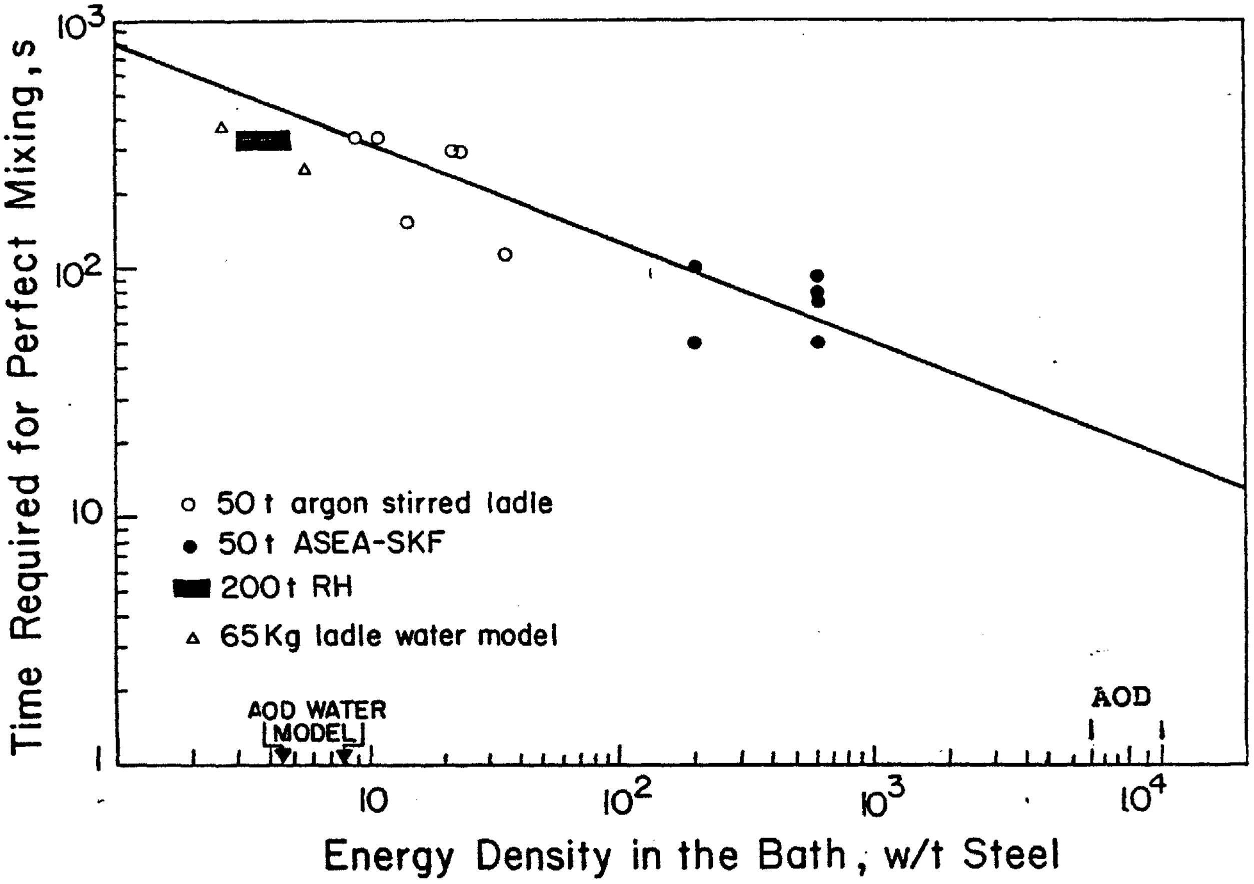


Fig. 5.32 Variation of Mixing Time with Stirring Power density in the melt (Ref. 59)

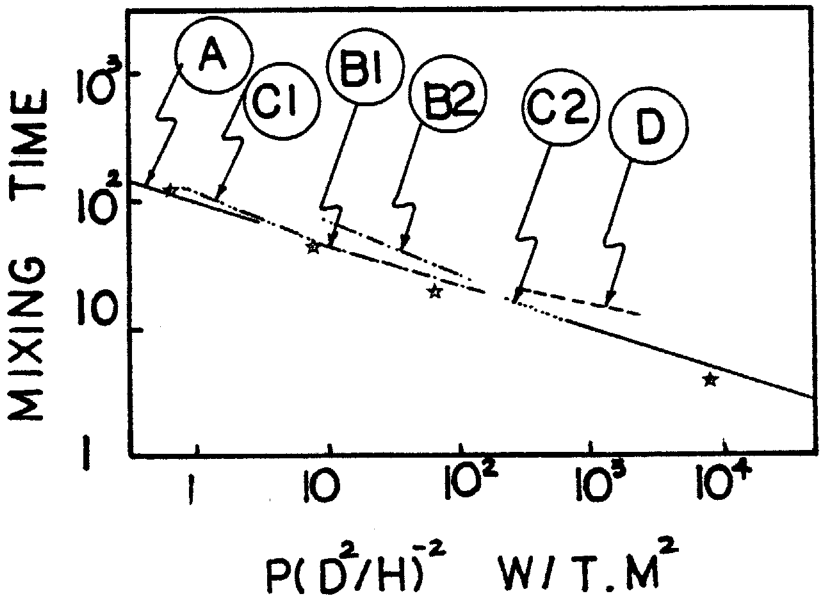


Fig. 5.33 Variation of Mixing Time with (P/D^2 /H) in the melt (Ref.58)

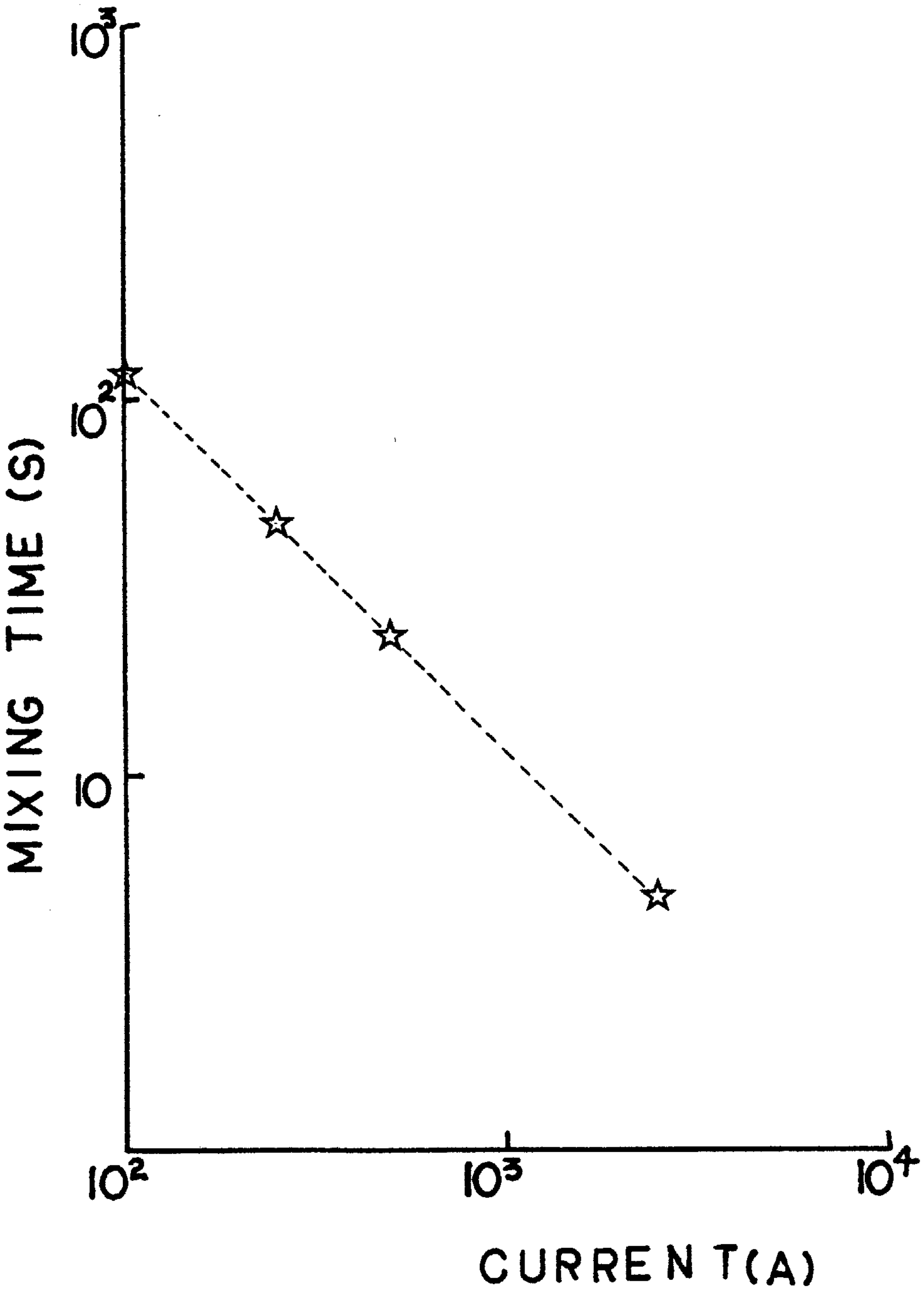
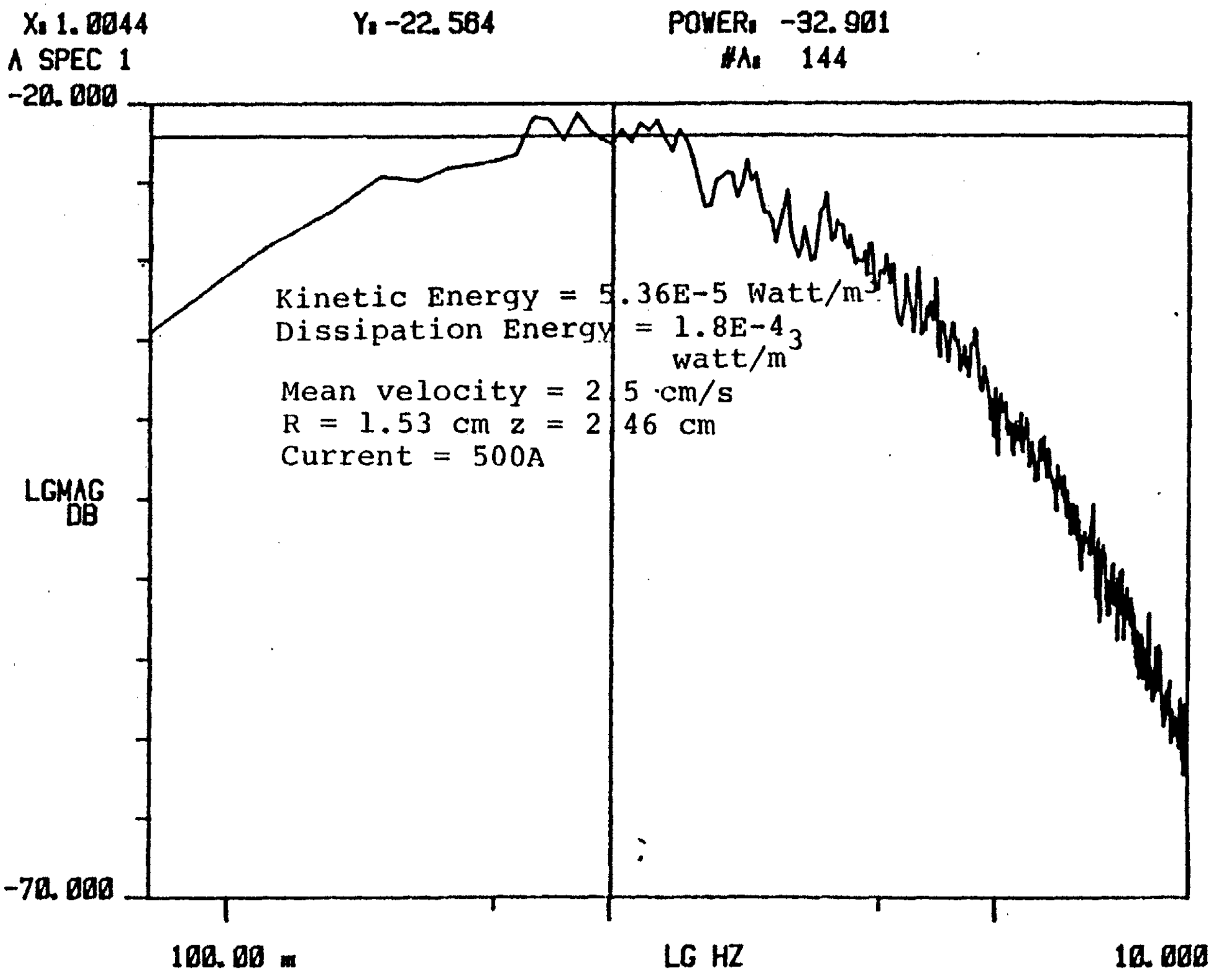


Fig. 5.34 Computed Variation of Mixing Time with Current passed through melt

Fig. 5.35 Example of Measured Turbulence Spectrum



Kinetic Energy = 5.36E-5 Watt/m^3
Dissipation Energy = 1.8E-4 watt/m^3

Mean velocity = 2.5 cm/s
R = 1.53 cm z = 2.46 cm
Current = 500A

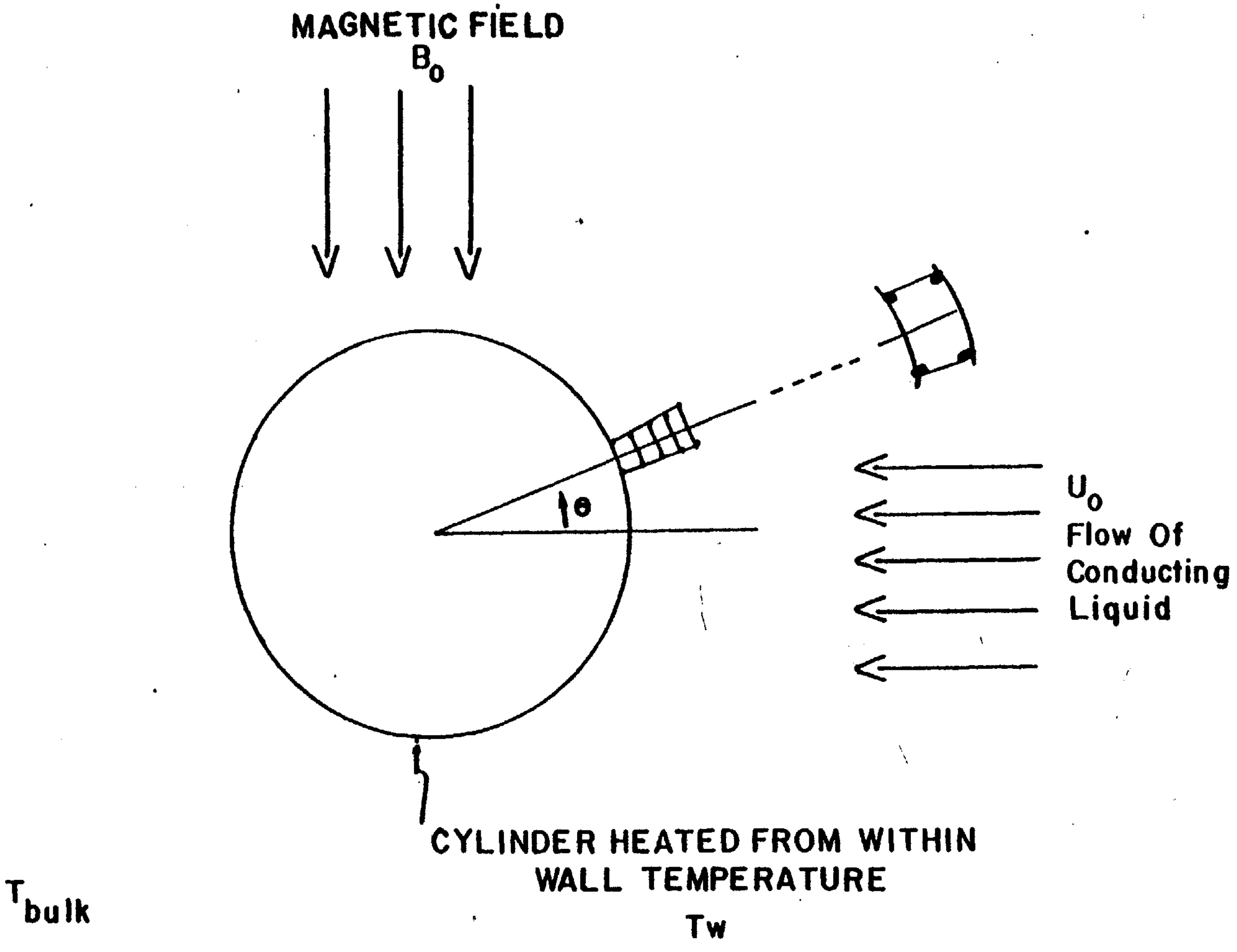


Fig. 5.36 Flow Configuration for MHD flow around cylinder

- Comparison of calculated Drag Coefficients at Red = 40

Source	C_D
Kawaguti	1.618
Apelt	1.469
Kalis, Tsinober et al.	1.645
This study	1.616

Fig. 5.37 Comparison of Drag Computed Drag Coefficient at Mag.Field = 0

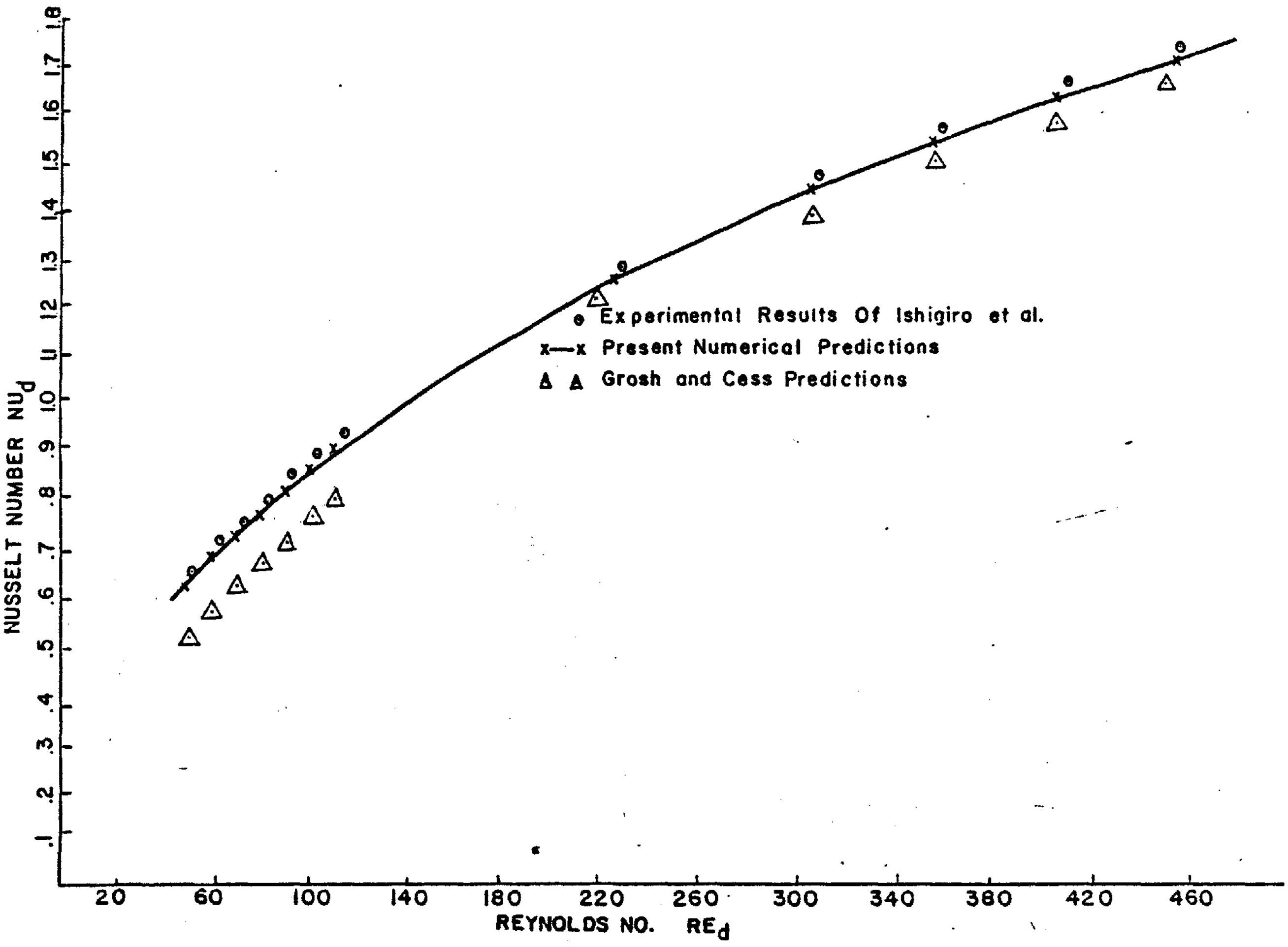


Fig. 5.38 Comparison of Averaged Nusselt Number with experiments

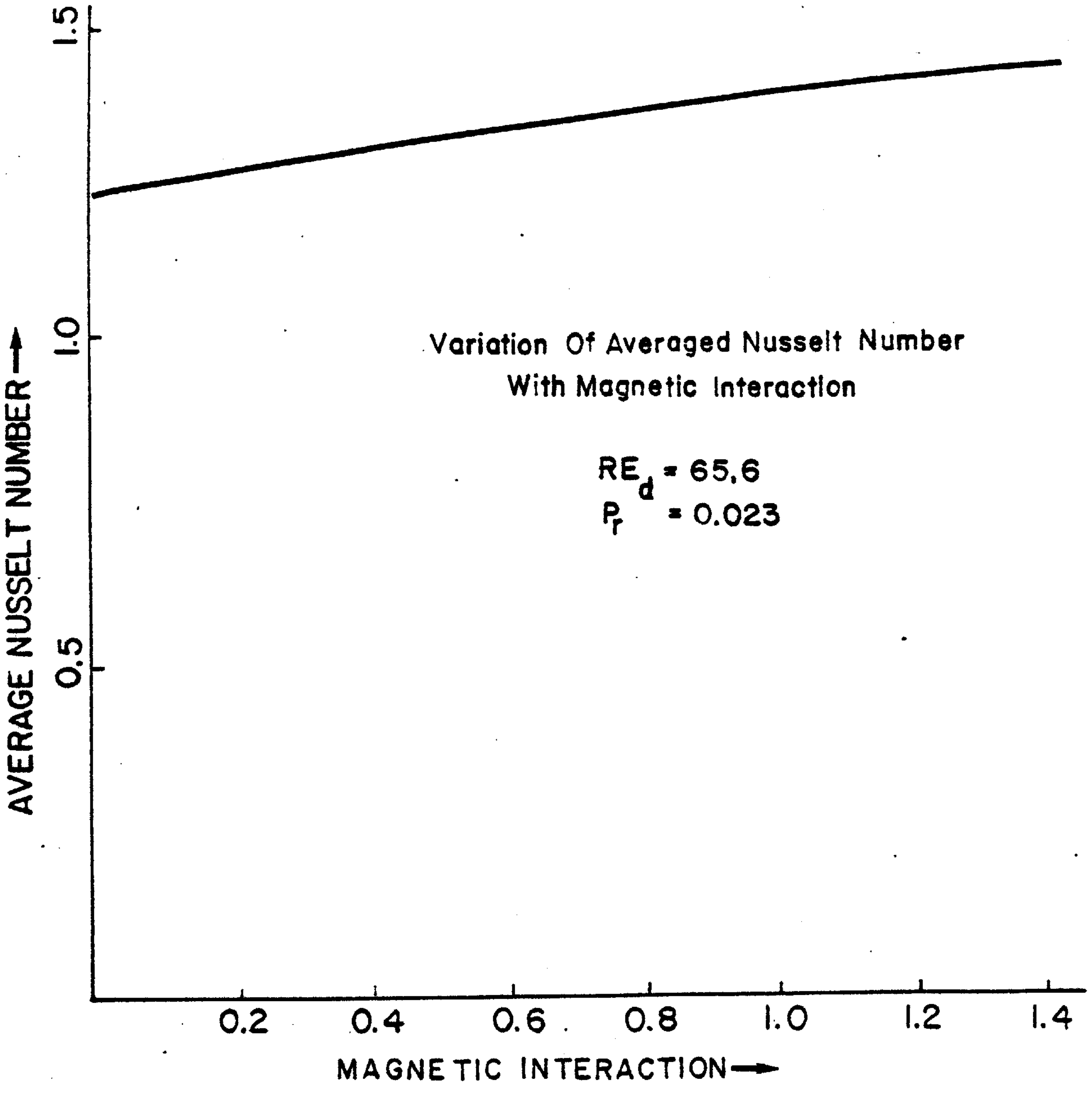


Fig. 5.39 Computed Variation of Averaged Nusselt Number with Magnetic Interaction Parameter

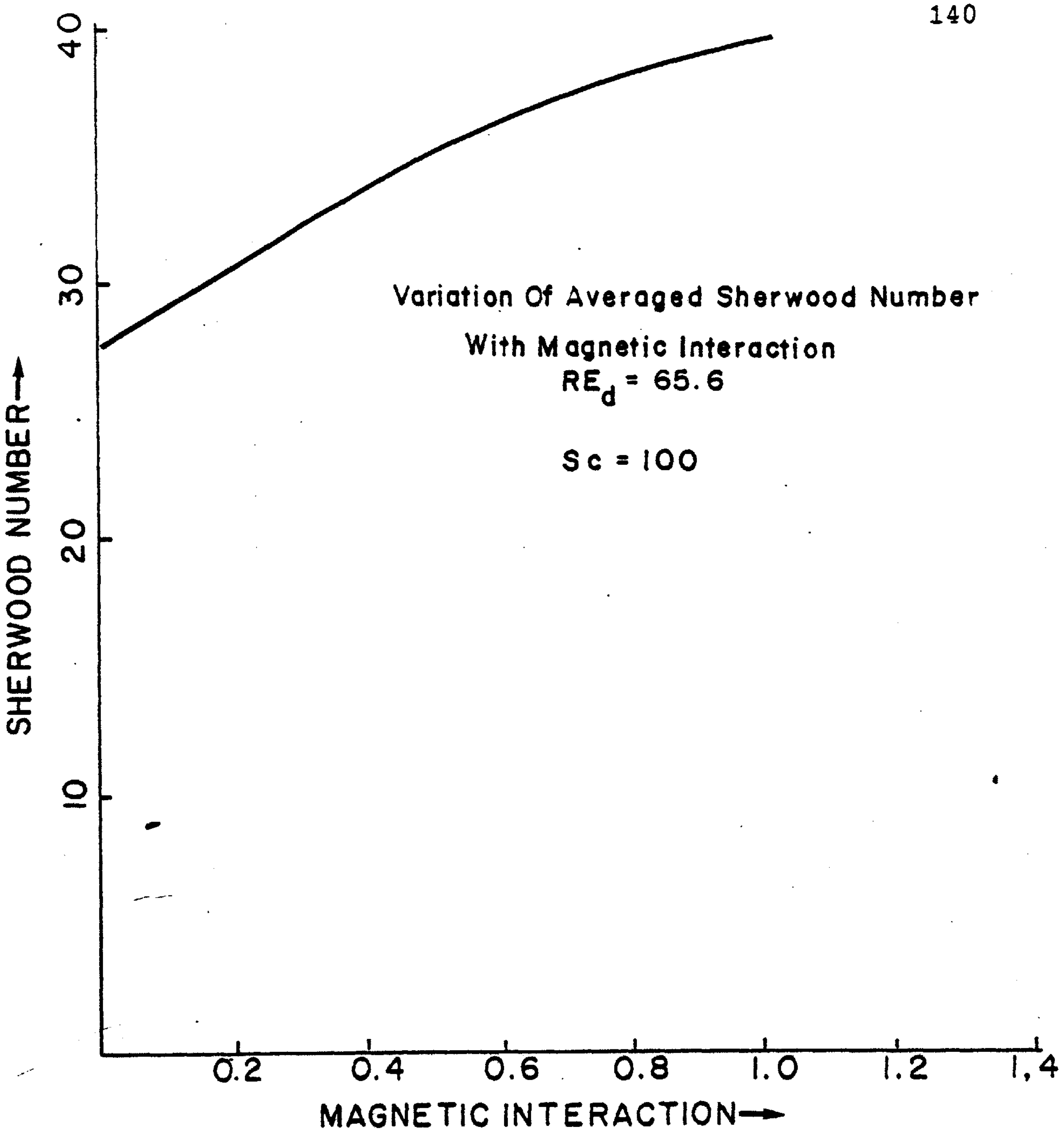


Fig. 5.40 Computed Variation of Averaged Sherwood Number with Magnetic Interaction Parameter

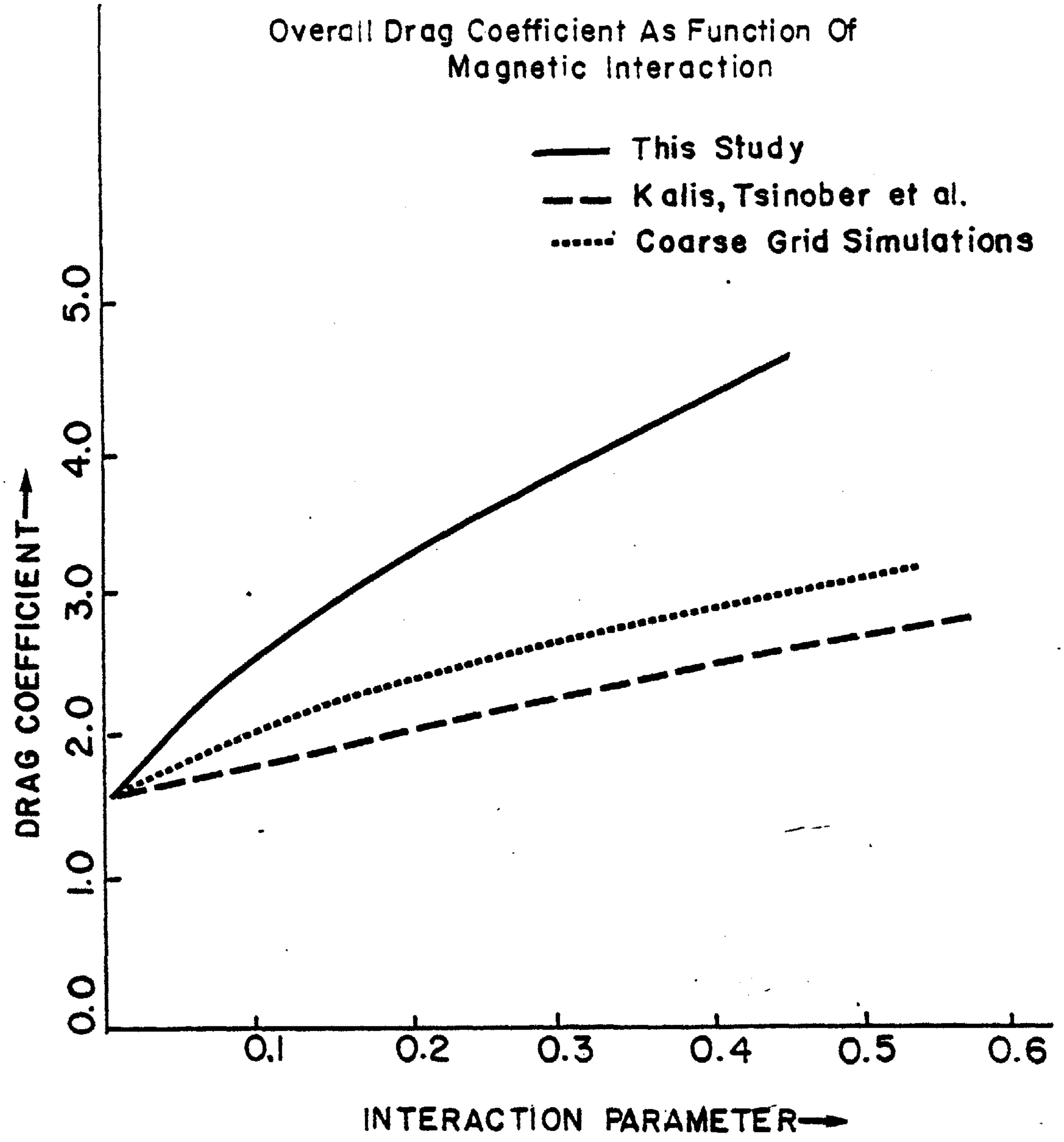


Fig. 5.41 Computed Variation of Drag Coefficient with Magnetic Interaction Parameter

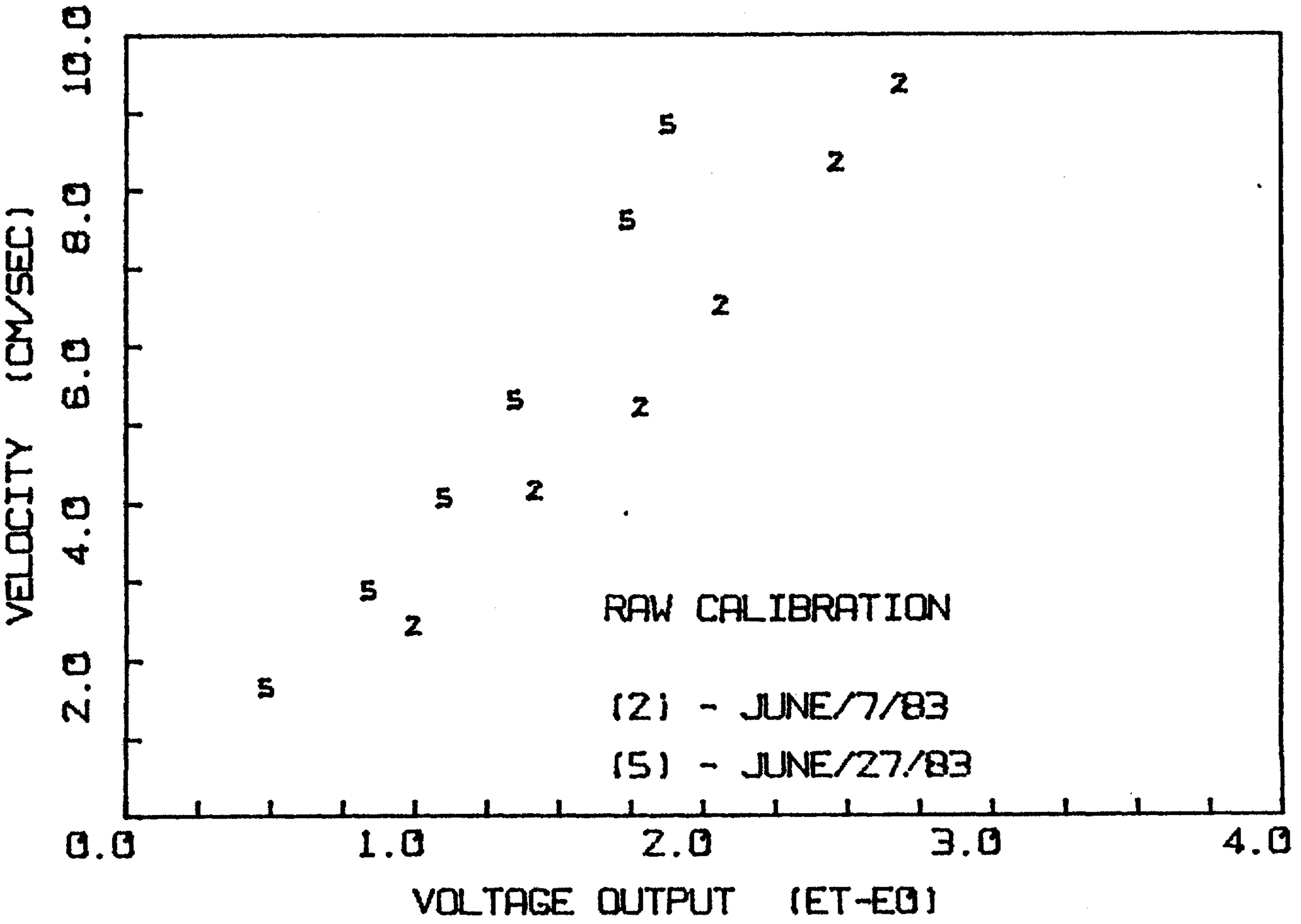


Fig. 5.42 Example of Calibration Curve for Hot Film Anemometer - before Correction Analysis

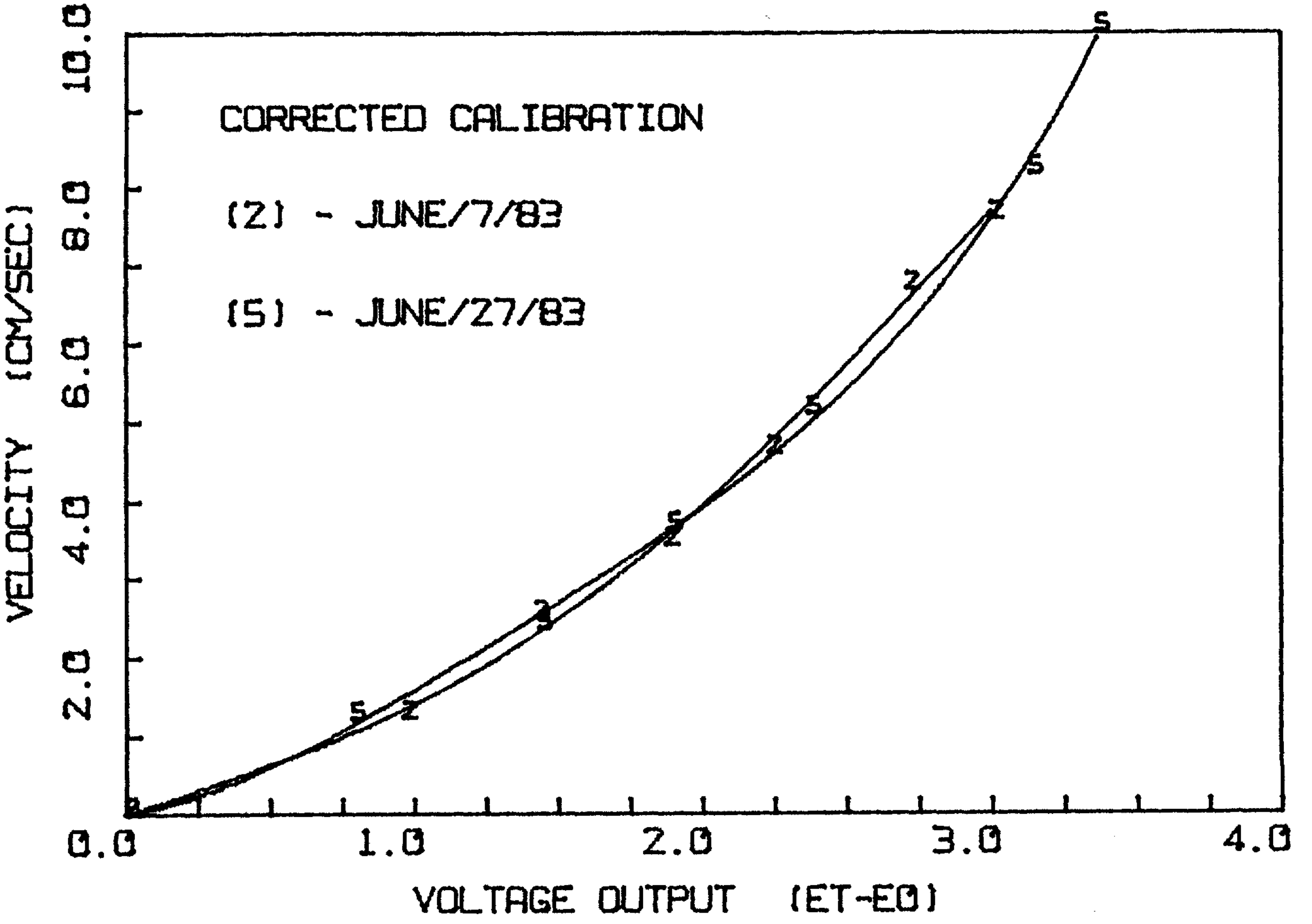


Fig. 5.43 Example of Same Calibration Curve - after correction analysis

6. CONCLUSIONS AND RECOMMENDATIONS

In this chapter some concluding remarks on the principal contributions and findings of this work are presented. Suggestions for further work along the lines of this investigation are also made.

6.1 Conclusions

Enumerated here are some of the contributions made by this work to the study of recirculating liquid metal flows in materials processing operations.

1) An experimental technique has been developed to make detailed measurements of flow and turbulence characteristics in a heated melt of liquid metal. The technique features modified probes for use in high temperature (upto 150° C) liquid metal melts, automatic calibration drift correction and a completely microcomputer controlled data acquisition and processing environment.

2) A mathematical model has been developed to represent the flow in the melt starting with calculation of the electromagnetic force field driving the flow and using the conventional $\kappa - \epsilon$ model to represent the turbulence.

3) An alternative turbulence model has been proposed. This model, called the XI model, has been entirely developed as part of this work and has the following features:

(a) It is a computationally simple model that uses 1 transport equation to represent the turbulence as opposed to 2 transport equations for the $\kappa - \epsilon$ model.

(b) It eliminates the need for most of the semi-empirical constants used by the $\kappa - \epsilon$ model.

(c) It is intuitively appealing as it uses simple concepts like specific power input to the system and characteristic vessel size as important parameters to characterise the state of turbulence.

4) Comparison of predictions using the conventional models as well as the new XI model show that:

(a) The XI model calculation of mean flow as well as turbulence parameters is very close to those predicted by the $\kappa - \epsilon$ model.

(b) Use of the Low Reynolds Number model does not result in any substantially different predictions.

(c) All three models predict the mean velocity and dissipation energy measured in the melt reasonably well. Turbulence intensity is, however, less well represented by all three models.

5) A parameter, called the 'Performance Index', (PI for short), is proposed. It is shown that this parameter can be used as a criterion for predicting the onset of turbulence in the recirculating flow in a melt more reliably than afforded by the use of a simple Reynolds Number evaluation.

6) Consideration of a 'Turbulence Efficiency' factor (η) is proposed as an important criterion in determining the performance of a metallurgical agitation system. A calculation, made to evaluate this efficiency factor for gas stirred melts, shows η =0.5 in such systems. It is expected that this value of η will be different in different classes of systems.

7) A theory is proposed for explaining the variation of mixing times in metallurgical system reactors with different levels of power input and vessel size. The predicted variation using this theory is shown to match very well with the experimentally observed relationship. The fundamental basis of this theory, it is felt, permits its use in a wide variety of metallurgical systems to evaluate mixing or reaction times.

6.2 Suggestions for further work

This section presents some thoughts on how certain aspects of the work presented in this thesis may be extended to further enhance our knowledge of mixing in metallurgical systems.

1) It is felt that pulsating the applied current passing through the melt with a variable frequency presents possibilities for an interesting investigation into the turbulence, and thence, mixer performance of metallurgical reactors. If the frequency of pulsation is adjusted to be inside the spectral range of turbulence frequencies, alteration of the spectral shape and consequently, of the turbulence parameters, may result. Also, an oscillating force field may be expected to result in a (possibly) enhanced turbulence efficiency factor, η . This would have a direct effect on the turbulence levels and mixing times for the system. This line of investigation is presently being pursued as continuing graduate research work.

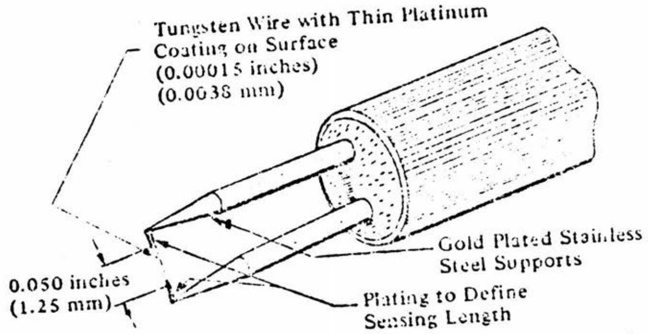
2) The XI model may be used for other turbulent recirculating flow problems and the results compared with similar computations using the $\kappa - \epsilon$ model and experimental results, if available.

3) Values for turbulence efficiency need to be evaluated for different classes of metallurgical systems like the applied current configuration (eg. Electro Slag Refining) or the inductively stirred system (eg. Induction Furnace). This could be conveniently done using the same technique that was used for gas injected systems in this work, ie. by utilising information about the overall mixing time for the system.

APPENDIX A

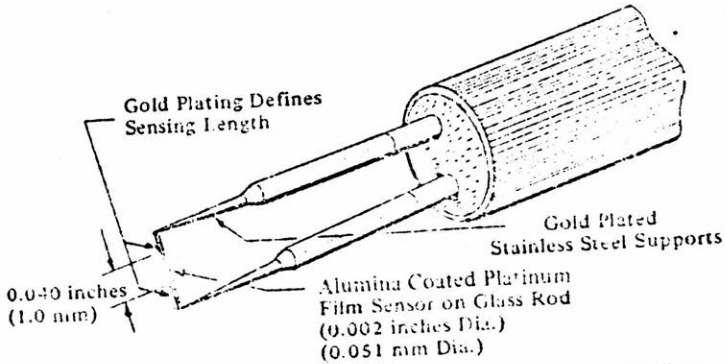
Thermal Anemometry Theory

The detecting element of a thermal anemometer consists of a very fine short metal wire (or film) which is heated by an electric current. The sensor is cooled by the flowing liquid, causing the heat transfer rate from the wire/film to increase. Typical sensors, of the hot wire and hot film type are shown below:



Tungsten Hot Wire Sensor and Support Needles-
0.00015" Dia. (0.0038 mm)

Fig. A.1



Cylindrical Hot Film Sensor and Support Needles-
0.002" Dia. (0.051 mm)

Fig. A.2

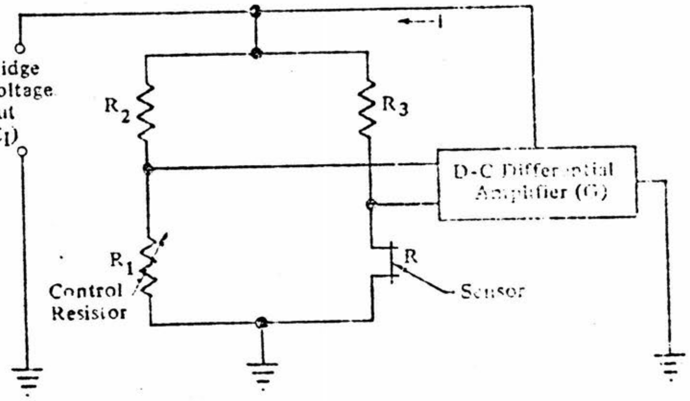
A very important part of the hot wire/film anemometer is the control circuit. There are two basic types of measurement schemes possible:

(1) The Constant Current Anemometer

and

(2) The Constant Temperature Anemometer.

The constant current type of anemometer operates by taking the voltage signal caused by wire resistance changes and compensates for frequency lag with a non-linear amplifier. The constant temperature control scheme, while certainly not a recent innovation, has gained rapidly in acceptance during the last few years. It operates by utilising feedback controlled bridge circuit to maintain the sensor at constant temperature. The figure below depicts a constant temperature system.



Schematic of Constant Temperature System

Fig. A.3

As the velocity past the sensor increases the sensor will tend to cool with a resulting decrease in the resistance. This resistance decrease will cause the voltage to decrease changing the input to the amplifier. The phase of the amplifier is such that this decrease in voltage will cause an increase in the output of the amplifier to increase the current through the sensor. If the amplifier has a sufficient gain, it will tend to keep its inputs very close to the balanced condition. Therefore any change in the sensor resistance will be immediately corrected by an increase or decrease in the current through the sensor. The output of the constant temperature system is the voltage output of the amplifier which in turn is the voltage required to drive the necessary current through the sensor. Since with feedback control the resistances in the bridge are constant, the voltage across the bridge is directly proportional to the current through the sensor and power is equal to current squared times the probe resistance. Therefore, the square of the voltage measured on top of the bridge is directly proportional to the instantaneous heat transfer between the sensor and environment.

APPENDIX B

Calculation of κ and ϵ from the Spectrum

A typical frequency domain spectrum is shown in Fig. 5.35 and represents a plot of E(n) versus n, where n is the frequency in Hertz. From this spectrum, the values of the Kinetic Energy of Turbulence, κ , and the Dissipation Rate of Turbulence, ϵ , are calculated using the following relationships:

$$\kappa = \frac{3}{2} (GD \times GF)^2 \times \int_0^\infty E(n)\, dn$$

and

$$\epsilon = \frac{60\pi^2 \nu}{U^2} (GD \times GF)^2 \times \int_0^\infty n^2 E(n)\, dn$$

where GD is the gradient of the calibration curve at the operating point specified by the mean velocity in M/sec/Volt and GF is the gain factor set on the analog data recorder input amplifiers. The other symbols have their usual meanings.

APPENDIX C

Properties of Wood's Alloy

COMPOSITION:

Bismuth	4 parts
Lead	2 part
Cadmium	1 part
Tin	1 part

PHYSICAL(87) :

Freezing Range	69 - 72 ° C
Density	8.4x10^{3} Kg/m^{3}
Electrical Conductivity	.99x10^{6} mho/m
Molecular Viscosity	2.29x10^{-3} N/s/m^{2}

APPENDIX D

Equipment Specifications:

MICROCOMPUTER AND PERIPHERICALS:

- * APPLE II Plus, 48 K RAM, MOSTEK 6502 Processor, 2 disk drives, BASIC and ASSEMBLER support
- * AI13 12 Bit A/D Converter, 16 Channel, 20 μ s conversion time, (Interactive Structures, Bala Cynwyd, Pennsylvania)
- * PL12 Flat Bed Plotter, 7" x 10" (also from Interactive Structures)
- * Micromodem II, 300 Baud, Direct Connect Modem (Hayes Microcomputer Products, Norcross, Georgia)
- * MPI - 88G Dot Matrix Printer (Microperipherals Inc., Salt Lake City, Utah)
- * 'The Clock', Real time clock with 1msec BCD bits (Mountain Computer Inc., Santa Cruz, California)

DC CURRENT GENERATOR:

Gold Star 600SS, Rated 600 Amps at 60% Duty Cycle with remote contactor control. (Miller Electric Mfg. Co., Appleton, Wisconsin)

SPECTRUM ANALYSIS:

* HP 5423 A Structural Dynamics Analyser
* HP 54470 B Digital Filter
* HP 54410 B A/D Converter
* HP 9872 S Flat Bed Plotter

(All from Hewlett Packard Co., Palo Alto, California)

THERMAL ANEMOMETRY:

* (2) Series 1050 Constant Temperature
* (2) Series 1057 Signal Conditioner
* Series 1051 Monitor and Power Supply
* 1212-60 Hg. Probes

(All from Thermo Systems Inc., St. Paul, Minnesota)

ANALOG DATA RECORDING:

Teac R 61 FM Data Recorder, 4 Channel, Single Speed, Variable Gain (TEAC Corporation, Tokyo, Japan)

APPENDIX E

Microcomputer Program Listings

```
100  TEXT : HOME : FLASH : HTAB 10: PRINT "RAW DATA ACQ": NORMAL
200  DIM A$(9),B(9),C(9),A(9)
220  GOSUB 9700: REM  HIMEM SETTING
290 K = 0
300 ARYSZ = 36609 + K:STALOC = 36611 + K:ACTSTL = 36613 + K
310 ENLOC = 36615 + K:MULTSZ = 36617 + K:MLPNM = 36619 + K
320 SLPNM = 36621 + K:INERVL = 36623 + K:NMCHN = 36624 + K
325 GSTAL = 36625 + K:GNCD = 36646 + K:TYPE = 36651 + K
330 MEXCLN = 36655 + K:TIMEIN = 36656 + K:L2AP = 36662 + K
340 R1EF = 36668 + K:R2EF = 36669 + K:SEXCLN = 36671 + K
350 L1AP = 36672 + K:S1TAL = 36678 + K:CLSYS = 36730 + K
360 G1R = 36698 + K:XPNO = 36702 + K:XCRD = 36704 + K
370 YCRD = 36706 + K:ZCRD = 36708 + K:RTN = 36710 + K
380 CURR = 36712 + K:TEMP = 36714 + K:OHR = 36716 + K:SVLTG = 36718
390 XDTE = 36732 + K:PARA = 36608 + K:HOT1 = 36736 + K
500 D$ =  CHR$ (4):R$ =  CHR$ (13):H = 256
1000  PRINT R$;D$;"BLOAD DAPARA,D1"
1010  INPUT " FIRST TIME TO THIS ROUT. (Y/N) ? ";AN$
1020  IF AN$ = "Y" THEN  POKE 34,Z1: HOME : GOSUB 2000: REM   VAR INPUT
1030  POKE 34,Z1: HOME : GOSUB 3000: REM   VAR DISPLAY
1040  PRINT : INPUT " NEED TO CHANGE VARIABLES? Y/N ";AN$
1050  IF AN$ = "Y" THEN  GOSUB 4000: GOTO 1030
1060  PRINT D$;"BLOAD HOT1.OBJO,D1"
1070  POKE 34, PEEK (37) - 1: HOME : INPUT " WHICH ROUTINE (MULT/SNGL) ";OP$
1080  POKE CHOICE, ASC ( LEFT$ (OP$,1)) + 128: POKE TYPE, ASC ( LEFT$ (OP$,1));128
1085  IF  LEFT$ (OP$,1) >  < "S" THEN  GOTO 1100
1090  POKE 36769,23: POKE 36888,0: POKE 36961,34. POKE 36989,234
1100  POKE 34, PEEK (37) - 1: HOME
1110  FLASH : PRINT "ARE YOU SURE THAT M/SW IS OFF(O V) Y/N?": GET AN$: NORMAL
1120  CALL HOT1: REM  MAIN DATA ACQ ROUT
1125  POKE 36769,128: POKE 36888,17: POKE 36961,44: POKE 36989,96
1130  PRINT R$;D$;"RUN SAVDAT1,D1"
1140  END
2000  INVERSE : PRINT "VARIABLE INPUT MODE": NORMAL : POKE 34, PEEK (37)
2010  INPUT "NO. OF CHANNELS FOR SAMPLING= ";C%: POKE NMCHN,C%
2020  FOR I = 1 TO C%
2030  PRINT I;"-CHN <GNCD, NO.> W/COMMA= ": VTAB  PEEK (37): HTAB 33: INPUT D%,E%
2040 E% = E% + D% * 16: POKE GNCD + 5 - I,E%
2050  NEXT I
2060  INPUT "MAIN SWITCH CHANNEL NUMBER= ";E%
2070  FOR J = I TO 5: POKE GNCD + 5 - J,E%: NEXT J
2080  INVERSE : PRINT "INPUT 4-HEX DIGITS PRECEDED BY '$' SIGN": NORMAL
2090  INPUT "DATA STACK SIZE=              ";A$(0)
2100  INPUT "ACT. START LOC. OF D-STACK=  ";A$(2)
2110  INPUT "START LOCATION OF D-STACK=    ";A$(1)
```

```
2120  INPUT "MAX. END LOC. OF D-STACK=     ";A$(3)
2130  INPUT "DATA SIZE OF MULT-ROUT=       ";A$(4)
2140  INPUT "LOOP NO. OF MULT-ROUT=        ";A$(5)
2150  INPUT "LOOP NO. OF SNGL-ROUT=        ";A$(6)
2160  INPUT "LOOP NO. FOR INTERVAL=        ";A$(7)
2170  FOR J = 0 TO 7
2180 A$ = A$(J): GOSUB 9800
2190  POKE ARYSZ - 1 + 2 * J,D: POKE ARYSZ + 2 * J,C
2200  NEXT J
2210  RETURN : END
3000  PRINT "A:NO.OF CHANNELS FOR SMPG="; TAB( 32); PEEK (NMCHN)
3010  FOR I = 1 TO  PEEK (NMCHN)
3020 D% =  INT ( PEEK (GNCD + 5 - I) / 16):E% =  PEEK (GNCD + 5 - I) - D% * 16
3030  PRINT "A:";I;"-CHN GNCD="; TAB( 16);D%; TAB( 21);I;"-CHN NO.="; TAB( 36);E%
3040  NEXT I
3050 E% =  PEEK (GNCD + 5 - I)
3060  PRINT "A:M/SW GNCD="; TAB( 16);"0"; TAB( 21);"M/SW CHN NO="; TAB( 36);E%: PRINT
3070  FOR J = 0 TO 7
3080 D =  PEEK (ARYSZ - 1 + 2 * J):C =  PEEK (ARYSZ + 2 * J): GOSUB 9900
3090 A$(J) = A$
3100  NEXT J
3110  PRINT "1:DATA ARY SZ="; TAB( 15);A$(0); TAB( 21);"2:START LOC="; TAB( 34);A$(1)
3120  PRINT "3:ACT.ST.LOC="; TAB( 15);A$(2); TAB( 21);"4:END LOC="; TAB( 34);A$(3)
3130  PRINT "5:MULT DAT.SZ="; TAB( 15);A$(4); TAB( 21);"6:L.NO.MULT="; TAB( 34);A$(5)
3140  PRINT "7:L.NO.SNGL="; TAB( 15);A$(6); TAB( 21);"8:L.NO.INTVL="; TAB( 34);A$(7): PRINT
3150  INVERSE : PRINT "SEE IF HIMEM =< ACTSTL, STALOC=ACTSTL-1"
3160  HTAB 7: PRINT "MULTSZ<$1000, ARYSZ<=$8F00-ACTSTL"
3170  HTAB 7: PRINT "ARYSZ > ( 4 )*NMCHN*LN(MULT/SNGL)": NORMAL
3180  RETURN : END
4000 Z2 =  PEEK (37) - 1: POKE 34,Z2: HOME
4010  INPUT "TYPE ALPHA/NO./CTRL-S TO BE CHANGED ";C$
4020  IF  LEFT$ (C$,1) =  CHR$ (19) THEN  GOTO 4180
4030  IF  ASC (C$) =  < 57 THEN  GOTO 4150
4040  INPUT "NO. OF CHANNELS FOR SAMPLING= ";C%
4050  POKE NMCHN,C%
4060  FOR I = 1 TO C%
4070  PRINT I;"-CHN <GNCD, NO.> W/COMMA= ": VTAB  PEEK (37): HTAB 33: INPUT D%,E%
4080 E% = E% + D% * 16: POKE GNCD + 5 - I,E%
4090  NEXT I
4100  INPUT "MAIN SWITCH CHANNEL NUMBER= ";E%
4110  FOR J = I TO 5
4120  POKE GNCD + 5 - J,E%
4130  NEXT J
4140  GOTO 4170
4150  INPUT "THEN VALUE(4-HEX DIG.W/'$')=? ";A$
```

```
4160  GOSUB 9800:J =  VAL (C$) - 1: POKE ARYSZ - 1 + 2 * J,D: POKE ARYSZ + 2 * J,C
4170  GOTO 4010
4180  RETURN : END
9700 C =  PEEK (116):D =  PEEK (115): GOSUB 9900
9710  VTAB 3: PRINT "PRESENT HIMEM="; TAB( 33);A$
9720  VTAB 4: HTAB 32: PRINT "=";D + C * 256:Z1 =  PEEK (37)
9730  VTAB 5: PRINT "WANT TO CHANGE HIMEM Y/N?": VTAB 5: HTAB 32: GET AN$
9740  IF AN$ >  < "Y" THEN  GOTO 9780
9750  VTAB 5: HTAB 1: PRINT "INPUT NEW HIMEM "
9755  VTAB 5: HTAB 17: INVERSE : PRINT "(4-HEX;$XXXX)=": NORMAL : VTAB 5: HTAB 32: INPUT A$
9760  GOSUB 9800:B = C * 256 + D: VTAB 6: HTAB 32: PRINT "=";B
9770  HIMEM: B:Z1 =  PEEK (37)
9780  POKE 34,Z1: HOME : RETURN : END
9800  FOR I = 2 TO 5
9810 A(I) =  ASC ( MID$ (A$,I,1))
9820  IF A(I) =  < 57 THEN A(I) = A(I) - 48: GOTO 9840
9830 A(I) = A(I) - 55
9840  NEXT I
9850 C = A(2) * 16 + A(3):D = A(4) * 16 + A(5)
9860  RETURN : END
9900 A(1) =  INT (C / 16):A(2) = C - A(1) * 16:A(3) =  INT (D / 16):A(4) = D - A(3) * 16:A$ = "$"
9910  FOR I = 1 TO 4
9920  IF A(I) < 10 THEN A$ = A$ +  STR$ (A(I)): GOTO 9940
9930 A(I) = A(I) + 55:A$ = A$ +  CHR$ (A(I))
9940  NEXT I
9950  RETURN : END
```

```
100  TEXT : HOME : FLASH : HTAB 7: PRINT "FULL CONV OF CAL-DATA": NORMAL
200  DIM A(9),B(9),C(19),D(9),E(9),G(9),GA(9)
210  GOSUB 9700: REM  HIMEM SETTING
290 K = 0
300 ARYSZ = 36609 + K:STALOC = 36611 + K:ACTSTL = 36613 + K
310 ENLOC = 36615 + K:MULTSZ = 36617 + K:MLPNM = 36619 + K
320 SLPNM = 36621 + K:INERVL = 36623 + K:NMCHN = 36624 + K
325 GSTAL = 36625 + K:GNCD = 36646 + K:TYPE = 36651 + K
330 MEXCLN = 36655 + K:TIMEIN = 36656 + K:L2AP = 36662 + K
340 R1EF = 36668 + K:R2EF = 36669 + K:SEXCLN = 36671 + K
350 L1AP = 36672 + K:S1TAL = 36678 + K:CLSYS = 36730 + K
360 G1R = 36698 + K:XPNO = 36702 + K:XCRD = 36704 + K
370 YCRD = 36706 + K:ZCRD = 36708 + K:RTN = 36710 + K
380 CURR = 36712 + K:TEMP = 36714 + K:OHR = 36716 + K:SVLTG = 36718
390 XDTE = 36732 + K:PARA = 36608 + K:HOT1 = 36736 + K
500 D$ =  CHR$ (4):R$ =  CHR$ (13):H = 256
1000  POKE 34,Z1: HOME : PRINT : INPUT " DRIVE NO.? ";D%
1010  POKE 34,2: HOME : PRINT R$;D$;"CATALOG,D";D%
1020  INPUT " DATA ACQ TYPE =? (M/S) ";TY$
1030  INPUT " EXP. NO.= ";XP
1040 F$ = "C-" +  STR$ (XP) + "-" + TY$:G$ = F$ + "-TS"
1050  PRINT R$;D$;"BLOAD ";F$;",D";D%: HOME : VTAB 2: INVERSE : PRINT F$;" ,D";D%: NORMAL : POKE 34,3
1060 NC =  PEEK (NMCHN):A% = 4:NP = 7
1070  GOSUB 6000: REM      GAIN CODE CALC;(D(I)) ,(GA(I))
1080  IF TY$ >  < "S" THEN  GOTO 1100
1090  GOSUB 3000: GOTO 1110: REM  SNGL CONV
1100  GOSUB 2000: REM    MULT CONV
1110  GOSUB 4000
1120  INPUT " WANT TO CONV TS ? (Y/N) ";AN$
1130  IF  LEFT$ (AN$,1) >  < "Y" THEN  GOTO 1150
1140  PRINT R$;D$;"BLOAD ";G$;",D";D%: VTAB 2: INVERSE : PRINT G$;" ,D";D%: NORMAL : POKE 34,3: GOSUB 4500
1150  PRINT R$;D$;"RUN MENU,D1"
1160  END
2000 DS =  PEEK (MULTSZ) * H +  PEEK (MULTSZ - 1):DAS=DS:DBS=DS
2010 NBD =  PEEK (SEXCLN) * H +  PEEK (SEXCLN - 1):NAD =  PEEK (MEXCLN) * H +  PEEK (MEXCLN - 1)
2020  FOR I = 0 TO NC - 1
2030 A(I) = 0:B(I) = 0
2040  FOR K = 0 TO 4
2050 B(I) = B(I) +  PEEK (S1TAL + 5 * I + K) * (H ^ K) / (DAS * NBD)
2060 A(I) = A(I) +  PEEK (GSTAL + 5 * I + K) * (H ^ K) / (DAS * NAD)
2070  NEXT K
2080 B(I) = D(I) * B(I) / 4095 / GA(I):A(I) = D(I) * A(I) / 4095 / GA(I)
2090  NEXT I
2100 MSB =  INT ( PEEK (L2AP + 5) / 16):MTB =  PEEK (L2AP + 5) - MSB * 16
2110 TBM =  PEEK (L2AP + 2) * 16 +  PEEK (L2AP + 3) +  PEEK (L2AP + 4) / 10 + MSB / 100 + MTB / 1000
```

```
2120 MSA =  INT ( PEEK (L1AP + 5) / 16):MTA =  PEEK (L1AP + 5) - MSA * 16
2130 TAM =  PEEK (L1AP + 2) * 16 +  PEEK (L1AP + 3) +  PEEK (L1AP + 4) / 10 + MSA / 100 + MTA / 1000
2140  RETURN : END
3000 DBS =  PEEK (MULTSZ) * H +  PEEK (MULTSZ - 1):DAS =  PEEK (SEXCLN) * H +  PEEK (SEXCLN - 1)
3005 NBD =  PEEK (MEXCLN) * H +  PEEK (MEXCLN - 1):NAD =  PEEK (SLPNM) * H +  PEEK (SLPNM - 1)
3010  FOR I = 0 TO NC - 1
3020 A(I) = 0:B(I) = 0
3030  FOR K = 0 TO 4
3040 B(I) = B(I) +  PEEK (GSTAL + 5 * I + K) * (H ^ K) / (DBS * NBD)
3050 A(I) = A(I) +  PEEK (S1TAL + 5 * I + K) * (H ^ K) / (DAS * NAD)
3060  NEXT K
3070 B(I) = D(I) * B(I) / 4095 / GA(I):A(I) = D(I) * A(I) / 4095 / GA(I)
3080  NEXT I
3090 MSB =  INT ( PEEK (L1AP + 5) / 16):MTB =  PEEK (L1AP + 5) - MSB * 16
3100 TBM =  PEEK (L1AP + 2) * 16 +  PEEK (L1AP + 3) +  PEEK (L1AP + 4) / 10 + MSB / 100 + MTB / 1000
3110 MSA =  INT ( PEEK (L2AP + 5) / 16):MTA =  PEEK (L2AP + 5) - MSA * 16
3120 TAM =  PEEK (L2AP + 2) * 16 +  PEEK (L2AP + 3) +  PEEK (L2AP + 4) / 10 + MSA / 100 + MTA / 1000
3130  RETURN : END
4000  POKE 34,3: HOME
4010  INPUT " WANT TO PRINT OUT? (Y/N) ";AN$: IF  LEFT$ (AN$,1) >  < "Y" THEN  HOME : GOTO 4030
4020  PRINT R$;D$;"PR#1": POKE 1401,80: HOME
4030 C(0) =  PEEK (XDTE + 3) +  PEEK (XDTE + 2) * H +  PEEK (XDTE + 1) * H * H +  PEEK (XDTE) * H * H * H
4040 A$ =  STR$ (C(0))
4050  FOR J = 0 TO NP - 1:C(J) =  PEEK (XCRD + 2 * J) * H +  PEEK (XCRD + 2 * J + 1): NEXT J
4060 C(0) = C(0) / 1E3:C(5) = C(5) / 10:C(6) = C(6) / 100
4065 TV = C(0) / TAM
4070  FOR I = 1 TO 39: PRINT "*";: NEXT I: PRINT : PRINT
4080 PRINT "FILE NAME & EXP NO = ";TAB(20);F$;TAB(41);"EXPERIMENT DATE = ";LEFT$ (A$,4);"/";MID$ (A$,5,2);"/";RIGHT$ (A$,2)
4090  PRINT "CURR= ";C(4);" AMP"; TAB( 21);"OP TEMP= ";C(5);" DEG"; TAB( 41);"O/H R= ";C(6)
4110  PRINT "TR= ";C(0);" CM"; TAB( 21);"TV= ";TV;" CM/SEC"
4120  FOR J = 0 TO NC - 1
4130  PRINT "GNCD:CH-";J + 1;"= ";G(J); TAB( 21);"REC GN:CH-";J + 1;"= ";GA(J); TAB( 41);"SVLTG:CH-";J + 1;"= ";E(J);" VOLT"
4140  PRINT "BASE VLTG: CH-";J + 1;"= ";B(J);" VOLT"; TAB( 41);
4150  PRINT "ACTV VLTG: CH-";J + 1;"= ";A(J);" VOLT"
4160  NEXT J
4170  PRINT "TOT SAMP TIM: BAS= ";TBM;" SEC"; TAB( 41);
4180  PRINT "TOT DAT PTS: BAS= ";DBS;"*";NBD;"= ";DBS * NBD
4190  PRINT "TOT SAMP TIM: ACT= ";TAM;" SEC"; TAB( 41);
4200  PRINT "TOT DAT PTS: ACT= ";DAS;"*";NAD;"= ";DAS * NAD
4210  PRINT : FOR I = 1 TO 39: PRINT "*";: NEXT I: PRINT
4220  PRINT R$;D$;"PR#0"
4230  RETURN : END
4500  POKE 34, PEEK (37): HOME : INPUT " WANT TO PRINT-OUT TS? (Y/N) ";AN$: IF  LEFT$ (AN$,1) >  < "Y" THEN  GOTO 4520
4510  PRINT R$;D$;"PR#1": POKE 1401,80
4520  HOME :SP =  PEEK (ACTSTL) * H +  PEEK (ACTSTL - 1)
```

```
4530  FOR J = 0 TO NAD - 1
4540 PT = SP + J * A% * NC
4550  FOR I = 0 TO NC - 1
4560 B(I) = 0:PR = PT + I * A%
4570  FOR K = 0 TO A% - 1
4580 B(I) = B(I) +  PEEK (PR + K) * (H ^ K) / DAS: NEXT K
4590 B(I) = B(I) * D(I) / 4095 / GA(I)
4600  NEXT I
4610  PRINT R$;J + 1;": ";
4620  FOR I = 0 TO NC - 1
4630 TP = I * 18 + 7:TP = TP -  INT (TP / 40) * 40: IF  PEEK (36) =  > TP THEN TP = TP + 40
4640  PRINT  TAB( TP);"C";I + 1;"=";B(I);
4650  NEXT I: NEXT J
4660  PRINT R$;D$;"PR#0": RETURN : END
6000  FOR I = 0 TO NC - 1
6010 G(I) =  INT ( PEEK (GNCD + 4 - I) / 16):C(I) =  PEEK (GNCD + 4 - I) - G(I) * 16:D(I) = G(I)
6020  IF D(I) >  = 4 THEN D(I) = D(I) - 4
6030 B% =  INT (D(I) / 2 + 0.5):C% = D(I) -  INT (D(I) / 2) * 2
6040 D(I) = 5 * ((2 ^ C%) / (10 ^ B%))
6045 GA(I) =  PEEK (G1R + I) / 100
6048 E(I) =  PEEK (SVLTG + 2 * I) * H +  PEEK (SVLTG + 2 * I + 1):E(I) = E(I) / 1E3
6050  NEXT I
6060  RETURN : END
9700 C =  PEEK (116):D =  PEEK (115): GOSUB 9900
9710  VTAB 3: PRINT "PRESENT HIMEM="; TAB( 33);A$
9720  VTAB 4: HTAB 32: PRINT "=";D + C * 256:Z1 =  PEEK (37)
9730  VTAB 5: PRINT "WANT TO CHANGE HIMEM Y/N?": VTAB 5: HTAB 32: GET AN$
9740  IF AN$ >  < "Y" THEN  GOTO 9780
9750  VTAB 5: HTAB 1: PRINT "INPUT NEW HIMEM "
9755 VTAB 5: HTAB 17: INVERSE : PRINT "(4-HEX;$XXXX)=": NORMAL : VTAB 5: HTAB 32: INPUT A$
9760  GOSUB 9800:B = C * 256 + D: VTAB 6: HTAB 32: PRINT "=";B
9770  HIMEM: B:Z1 =  PEEK (37)
9780  POKE 34,Z1: HOME : RETURN : END
9800  FOR I = 2 TO 5
9810 A(I) =  ASC ( MID$ (A$,I,1))
9820  IF A(I) =  < 57 THEN A(I) = A(I) - 48: GOTO 9840
9830 A(I) = A(I) - 55
9840  NEXT I
9850 C = A(2) * 16 + A(3):D = A(4) * 16 + A(5)
9860  RETURN : END
9900 A(1) =  INT (C / 16):A(2) = C - A(1) * 16:A(3) =  INT (D / 16):A(4) = D - A(3) * 16:A$ = "$"
9910  FOR I = 1 TO 4
9920  IF A(I) < 10 THEN A$ = A$ +  STR$ (A(I)): GOTO 9940
9930 A(I) = A(I) + 55:A$ = A$ +  CHR$ (A(I))
9940  NEXT I
```

```
9950  RETURN : END
```

```
100  TEXT : HOME : FLASH : HTAB 7: PRINT "POLY-FITTING OF CAL-CURVE": NORMAL
110  HIMEM: 36864: REM  36864=$9000
120  POKE 34,3: HOME
200  DIM X(100),Y(100),W(100),CY(100),E(100),C(10),A(10,11),XN(100)
300 D$ =  CHR$ (4):R$ =  CHR$ (13)
500  REM  PROGRAM FOR POLYNOMIAL FITTING TO A SET OF DATA
510  REM  BY LEAST-SQUARE-METHOD
520  GOSUB 4000: REM    FILE READING SUBR
530  GOSUB 4500: REM  ORIG. COEFF. MATRIX MAKING SUBR
540  PRINT : PRINT "***ORIGINAL SQUARE COEFFICIENT MATRIX***": PRINT : GOSUB 2500: REM  MATRIX PRINTING SUBR
550  GOSUB 1000: REM  LOWER/UPPER TRIANGLUAR MATRIX DECOMPOSITION
560  PRINT : PRINT "***L/U DECOMPOSED MATRIX***": PRINT : GOSUB 2500
570  REM  RESET THE RHS INTO C (COEFF)--DO THIS FOR EACH DEGREE
580  FOR I = (MS + 1) TO SF
590  PRINT : PRINT : PRINT "**********POLYNOMIAL---DEGREE ";I - 1;"**********"
600  FOR J = 1 TO I:C(J) = A(J,TF): NEXT J
610  GOSUB 2000: REM  SOLUTION SUBROUTINE
620  PRINT : PRINT "***FINAL  RESULT:COEFFICIENTS***": PRINT
630  GOSUB 2700: REM  COEFF PRINTING SUBR
640  GOSUB 5000: REM   REFERENCE OUTPUT SUBR
650  GOSUB 5500: REM  COEFF-FILE SAVE
660  NEXT I: REM   TRY NEXT DEGREE OF POLYNOMIAL
670  PRINT R$;D$;"RUN MENU,D1"
680  END
1000  REM  THIS SUBR FORMS L/U EQUIVALENT OF SQ COEFF MATRIX A
1010  REM  THE L/U MATRIX,IN COMPACT FORM, IS RETURNED IN A
1020  FOR I = 1 TO SF
1030  FOR J = 2 TO SF
1040 SUM = 0
1050  IF J > I THEN  GOTO 1110
1060  FOR K = 1 TO J - 1:SUM = SUM + A(I,K) * A(K,J): NEXT K
1070 A(I,J) = A(I,J) - SUM
1080  GOTO 1170
1110  IF (I - 1) = 0 THEN  GOTO 1150
1120  FOR K = 1 TO I - 1:SUM = SUM + A(I,K) * A(K,J): NEXT K
1130  REM  TEST FOR SMALL VALUE ON DIAGONAL
1150  IF  ABS (A(I,I) < 1.0E - 10) THEN  GOTO 1200
1160 A(I,J) = (A(I,J) - SUM) / A(I,I)
1170  NEXT J
1175  NEXT I
1180  RETURN : END
1200  PRINT "REDUCTION NOT COMPLETED, DUE TO SMALL VALUE IN DIVISOR IN ROW-";I: RETURN
1210  END
2000  REM  THIS SUBR FINDS SOLUTION TO A SET OF I-LINERAR EQUATION
2010  REM  THAT CORRESPONDS TO THE RHS VECTOR C.
```

```
2020  REM  A-MATRIX =L/U DECOMPSION EQUIV TO ORIG COEFF MATRIX
2030  REM  BY L/U REDUCTION PROCEDURE, SOLUTION VECTOR IS RETURNED IN C VECTOR
2040  REM  NOW, DO THE REDUCTION STEP
2050 C(1) = C(1) / A(1,1)
2060  FOR J = 2 TO I
2070 SUM = 0
2080  FOR K = 1 TO J - 1:SUM = SUM + A(J,K) * C(K): NEXT K
2090 C(J) = (C(J) - SUM) / A(J,J)
2100  NEXT J
2102  PRINT : PRINT "***FINALLY DECOMPOSED LAST COLUMN OF L/U MATRIX***": PRINT
2104  GOSUB 2700
2110  REM  NOW,DO BACK SUBSTITUTION,AND DIAGONAL ELEMENTS OF U-MATRIX ARE ALL 1S.
2120  FOR J = 2 TO I
2130 SUM = 0
2140  FOR K = (I - J + 2) TO I:SUM = SUM + A(I - J + 1,K) * C(K): NEXT K
2150 C(I - J + 1) = C(I - J + 1) - SUM
2155  NEXT J
2160  RETURN
2170  END
2500  REM  (N X N+1) MATRIX PRINTING SUBR
2510  FOR I = 1 TO SF
2520  PRINT "<";I;">-TH ROW ELEMENTS"
2530  FOR J = 1 TO TF
2532 TB = 20 * (J - 1) + 1:TB = TB -  INT (TB / 40) * 40
2534  IF  PEEK (36) =  > TB THEN TB = TB + 40
2540  PRINT  TAB( TB);"(";J;") ";A(I,J);
2550  NEXT J
2560  PRINT : NEXT I
2570  RETURN
2580  END
2700  REM  COEFF PRINTING SUBR
2710  FOR J = 1 TO I:TB = 20 * (J - 1) + 1:TB = TB -  INT (TB / 40) * 40
2720  IF  PEEK (36) =  > TB THEN TB = TB + 40
2730  PRINT  TAB( TB);"(";J;") ";C(J);: NEXT J: PRINT
2740  RETURN
2750  END
4000  REM  FILE READING SUBROUTINE
4010  INPUT "REF'NED CAL-ASSY F.N. = ";F$
4020  PRINT R$;D$;"OPEN ";F$,",D2"
4040  PRINT R$;D$;"READ ";F$
4045  INPUT E0,SZ
4050  FOR I = 1 TO SZ
4060  INPUT X(I),Y(I)
4065 W(I) = 1
4070  NEXT I
```

```
4080  PRINT R$;D$;"CLOSE ";F$
4090 N = SZ: PRINT "*NO. OF POINTS: N = ";N
4095 IX = EO
4098  FOR I = 1 TO N:X(I) = X(I) - IX: NEXT I
4100  PRINT : INPUT "*STARTING, FINAL DEG OF POLYNOM-APPROX:(DEG. LE. N-1 & .LE. 9) =? ";MS,MF
4110 SF = MF + 1:TF = MF + 2
4120  RETURN : END
4500  REM  ORIGINAL COEFFICIENT MATRIX MAKING SUBR(N ROWS,N+1 COL)
4510  REM  ARRAY XN HOLDS POWER OF THE EACH X VALUE
4520  FOR I = 1 TO N:XN(I) = 1: NEXT I
4530  REM  COMPUTE 1ST / N+1ST COLUMN OF A.
4540  REM  I MOVES DOWN THE ROWS, J SUMS OVER THE N VALUES
4550  FOR I = 1 TO SF
4560 A(I,1) = 0:A(I,TF) = 0
4570  FOR J = 1 TO N
4580 A(I,1) = A(I,1) + XN(J) * W(J)
4590 A(I,TF) = A(I,TF) + Y(J) * XN(J) * W(J)
4600 XN(J) = XN(J) * X(J)
4610  NEXT J
4620  NEXT I
4630  REM  COMPUTE THE LAST ROW OF A. I (COLUMN #). J SUMS OVER N VALUES
4640  FOR I = 2 TO SF
4650 A(SF,I) = 0
4660  FOR J = 1 TO N
4670 A(SF,I) = A(SF,I) + XN(J) * W(J)
4680 XN(J) = XN(J) * X(J)
4690  NEXT J
4700  NEXT I
4710  REM  FILL THE REST OF A. I-ROW,J MOVES ACROSS THE COLUMNS
4720  FOR J = 2 TO SF
4730  FOR I = 1 TO MF
4740 A(I,J) = A(I + 1,J - 1)
4750  NEXT I
4760  NEXT J
4770  RETURN : END
5000  REM   REFERENCE OUTPUT SUBROUTINE
5010 VAR = 0:ESQ = 0
5020  FOR J = 1 TO N
5030 CY(J) = 0
5040  FOR K = 2 TO I:CY(J) = (CY(J) + C(I - K + 2)) * X(J): NEXT K
5050 CY(J) = CY(J) + C(1):ER = Y(J) - CY(J)
5055  IF Y(J) = 0 THEN E(J) = 999999999: GOTO 5060
5058 E(J) = ER * 100 / Y(J)
5060 ESQ = ESQ + ER ^ 2
5070  NEXT J
```

```
5080 VAR = ESQ / (N - I)
5090  PRINT : PRINT "SUM OF ERROR SQ'ED = ";ESQ; SPC( 5);"VARIANCE OF FITTING = ";VAR: PRINT
5095  PRINT "NO. "; TAB( 10);"X"; TAB( 19);"Y"; TAB( 30);"CY"; TAB( 45);"ERROR, %"
5098 FOR ZZ=1 TO 79:PRINT "-";:NEXT ZZ:PRINT
5100  FOR J = 1 TO N
5110 J$ =  STR$ (J)
5120  IF  INT (J / 10) = 0 THEN J$ = " " + J$
5130  PRINT J$; TAB( 5);X(J); TAB( 18);Y(J); TAB( 26);CY(J); TAB( 49);E(J)
5140  NEXT J
5150  RETURN : END
5500  INPUT "CAL-COEFF FILE NAME = ";G$
5510  PRINT R$;D$;"OPEN ";G$;",D2"
5520  PRINT R$;D$;"DELETE ";G$
5530  PRINT R$;D$;"OPEN ";G$
5540  PRINT R$;D$;"WRITE ";G$
5550  PRINT C(1);",";C(2);",";C(3);",";C(4);",";C(5)
5560  PRINT EO
5570  PRINT R$;D$;"CLOSE ";G$
5580  RETURN : END
```

```
SOURCE FILE: HOT1.........
                   1 ****************************************
0000:              2 *THIS ASSEMBLY ROUTINE CONSISTS OF TWO MAIN PARTS (GETMULT
0000:              3 * & GETSNGL ), PREPARATION STEP AND OTHER SUBROUTINES.
                   4 ****************************************

0000:              6 *ZERO PAGE VARIABLES

0006:              8 ZLOC    EQU  $06      * ZERO LOCATION PTR,DUMMY,=1B,ASM
0007:              9 ZPAGE   EQU  $07      * PAGE COUNTER(DATA ARRAY),=1B,ASM
0008:             10 XSAV1   EQU  $08      * X REGISTER SAVE1,=1B,ASM
0009:             11 XSAV2   EQU  $09      * X REGISTER SAVE2,=1B,ASM
0019:             12 ZNMCHN  EQU  $19      * NO. OF CHANNELS,=1B,ASM
001A:             13 CHN1V   EQU  $1A      * 1ST CHAN VLTG(L),3B>,ASM
001D:             14 CHN2V   EQU  $1D      * 2ND CHAN VLTG(L),3B>,ASM
00CF:             15 ZLPNUM  EQU  $CF      * LOOPING NO.(H),<2B,ASM
00EB:             16 ZGNCD   EQU  $EB      *GAINCODE(G5,G4,G3,G2,G1),5B>,ASM
00F9:             17 YSAV1   EQU  $F9      * Y REGISTER SAVE1,=1B,ASM
00FA:             18 CHN3V   EQU  $FA      * 3RD CHAN VLTG(L),3B>,ASM
00FD:             19 CHN4V   EQU  $FD      * 4TH CHAN VLTG(L),3B>,ASM

0000:             21 *$8F PAGE VARIABLES

8F01:             23 ARYSZ   EQU  $8F01    **DATA ARRAY SIZE(H),<2B,BASIC
8F03:             24 STALOC  EQU  $8F03    **START LOC. OF D.ARY(H),ACT.LOC-1
8F05:             25 ACTSTL  EQU  $8F05    **ACT.START LOC OF D.ARY(H),<2B,BAS
8F07:             26 ENDLOC  EQU  $8F07    * END LOCATION OF DAT ARY(H),<2B,ASM
8F09:             27 MULTSZ  EQU  $8F09    **SIZE OF MULT DATA(H),<2B,BASIC
8F0B:             28 LNMULT  EQU  $8F0B    **LOOPING NO. OF MULT(H),<2B,BASIC
8F0D:             29 LNSNGL  EQU  $8F0D    **LOOPING NO. OF SNGL(H),<2B,BASIC
8F0F:             30 INTVL   EQU  $8F0F    **LP NO. OF INTERVAL DELAY(H),<2B
8F10:             31 NMCHN   EQU  $8F10    **NO.OF CHANNELS,=1B,BASIC
8F11:             32 GTOT1   EQU  $8F11    * GRAND TOTAL1(L),5B>,ASM
8F16:             33 GTOT2   EQU  $8F16    * GRAND TOTAL2(L),5B>,ASM
8F1B:             34 GTOT3   EQU  $8F1B    * GRAND TOTAL3(L),5B>,ASM
8F20:             35 GTOT4   EQU  $8F20    * GRAND TOTAL4(L),5B>,ASM
8F25:             36 CHOICE  EQU  $8F25    * CHOICE DISPLAY,=1B,ASM
8F26:             37 GNCD    EQU  $8F26    **GAINCODE,(G5,G4,G3,G2,G1),5B>,BASIC
8F3C:             38 REFTIM1 EQU  $8F3C    *REF. TIME1,=1B,ASM
8F3D:             39 REFTIM2 EQU  $8F3D    *REF. TIME2,=1B,ASM
8F2C:             40 DLN1    EQU  $8F2C    * DISP.L.N.(1,2 * NMCHN FOR S,M)
8F2D:             41 DLN2    EQU  $8F2D    * DISP.L.NO.(2,4 FOR S,M),=1B,ASM
8F2F:             42 EXCMLN  EQU  $8F2F    * EXEC'TED LP.NO OF MULT
8F3F:             43 EXCSLN  EQU  $8F3F    * EXEC'TED LP NO OF SNGL
8F36:             44 LAPTIM2 EQU  $8F36    * LAP TIME,6B>,ASM
```

```
8F30:            45 TIMEIN  EQU  $8F30      * TIME INITIAL,6B>,ASM
8F40:            46 LAPTIM1 EQU  $8F40      * LAP TIME,6B>,ASM
8F46:            47 STOT1   EQU  $8F46      *SUBTOTAL1(L),5B>,ASM
8F4B:            48 STOT2   EQU  $8F4B      *SUBTOTAL2(L),5B>,ASM
8F50:            49 STOT3   EQU  $8F50      *SUBTOTAL3(L),5B>,ASM
8F55:            50 STOT4   EQU  $8F55      *SUBTOTAL4(L),5B>,ASM
8F80:            51 ORIGIN  EQU  $8F80      * ORIGIN OF THIS BINARY FILE

0000:            53 *OTHER PAGE VARIABLES

COCO:            55 CLOCK   EQU  $COCO      *CLOCK, 5B>, ASM
CODO:            56 AI13    EQU  $CODO      *ANAL/DIG CONV, 3B>, ASM


                 58 ****************************************
0000:            59 *PREPARATION SECTION
                 60 ****************************************

----- NEXT OBJECT FILE NAME IS HOT1.OBJO
8F80:            62         ORG  ORIGIN
8F80:A9 00       63         LDA  #$00       *ZLOC=$00
8F82:85 06       64         STA  ZLOC
8F84:20 16 92    65         JSR  PARYCLR    *G/S-TOT & EXC(M/S)LN CLEAR
8F87:20 50 92    66         JSR  TCARYCL    *INIT/LAP TIME CHNV CLR
8F8A:AD 10 8F    67         LDA  NMCHN      *ZNMCHN=NMCHN
8F8D:85 19       68         STA  ZNMCHN

8F8F:20 7C 92    70         JSR  SCRNCLR
8F92:A9 01       71         LDA  #$01       *VTAB=01
8F94:85 22       72         STA  $22
8F96:AD 25 8F    73         LDA  CHOICE     *OPTION(MULT OR SNGL)
8F99:C9 D3       74         CMP  #'S'
8F9B:FO 03       75         BEQ  SKIP77
8F9D:4C A3 8F    76         JMP  BSVLTG
8FA0:4C 78 90    77 SKIP77  JMP  GETSNGL



                 79 ****************************************
8FA3:            80 *MULTIPLE DATA PICKING ROUTINE
                 81 ****************************************

8FA3:            83 *BASE VOLTAGE MEASURE ROUTINE
```

```
8FA3:A2 00      85 BSVLTG  LDX  #$00       *CAPTION 1 DISP & CHOICE
8FA5:20 84 92   86         JSR  HDLNCHO

8FA8:A2 05      88         LDX  #$05       *ZGNCD=GNCD
8FAA:BD 25 8F   89 LOOP72  LDA  GNCD-1,X
8FAD:95 EA      90         STA  ZGNCD-1,X
8FAF:CA         91         DEX
8FB0:D0 F8      92         BNE  LOOP72

8FB2:20 30 92   94         JSR  DARYCLR    *DATA ARRAY CLEAR
8FB5:20 F3 94   95         JSR  PTRSET

8FB8:A0 00      97         LDY  #$00       *LNMULT
8FBA:20 8E 92   98         JSR  LPLMTST    *LOOPING LIMIT SET

8FBD:20 65 92  100         JSR  MSWAIT     *MAIN SW(0/5V) WAITING & BELL

8FC0:20 D4 92  102         JSR  INITIM     *INITIAL TIME SET SUBR

8FC3:20 01 91  104 LOOP70  JSR  MULTDAT
8FC6:20 9D 92  105         JSR  STOPTM
8FC9:20 1F 93  106         JSR  STORE      *STORE DATA INTO DATA ARRAY

8FCC:AD 25 8F  108         LDA  CHOICE
8FCF:C9 D9     109         CMP  #'Y'
8FD1:D0 03     110         BNE  EXIT74
8FD3:20 0E 94  111         JSR  DISPLY

8FD6:20 6E 92  113 EXIT74  JSR  MSWCHK     *MAIN SW CHECK STEP
8FD9:F0 08     114         BEQ  EXIT75

8FDB:C6 CE     116         DEC  ZLPNUM-1   *LOOPING TIME COUNT
8FDD:D0 E4     117         BNE  LOOP70
8FDF:C6 CF     118         DEC  ZLPNUM
8FE1:10 E0     119         BPL  LOOP70

8FE3:20 9D 92  121 EXIT75  JSR  STOPTM
8FE6:20 FE 94  122         JSR  BELL
8FE9:20 C8 92  123         JSR  LAPTMV

8FEC:20 7C 92  125         JSR  SCRNCLR
8FEF:A2 22     126         LDX  #$22       *CAPTION-4 DISPLAY
8FF1:20 13 93  127         JSR  CAPTDISP
```

```
8FF4:A0 00      129          LDY  #$00       *LNMULT
8FF6:A2 10      130          LDX  #$10       *EXCSLN
8FF8:20 5E 94   131          JSR  EXLPNM     *EXEC'TED LP NO CALC

8FFB:A0 34      133          LDY  #$34       *EXCSLN
8FFD:A2 35      134          LDX  #$35       *STOT
8FFF:86 09      135          STX  XSAV2
9001:20 7A 94   136          JSR  MULTADD

9004:20 C6 93   138          JSR  ARYDISP

9007:A2 35      140          LDX  #$35       *STOT
9009:20 EE 93   141          JSR  GTOTLAPT  *STOT &LAPTIME & BELL
900C:EA EA EA   142          DFB  $EA,$EA,$EA

900F:           144 *ACTIVE VOLTAGE MEASURE ROUTINE

900F:A2 11      146 GETMULT LDX  #$11       *CAPTION 3 DISP & CHOICE
9011:20 84 92   147          JSR  HDLNCHO

9014:A2 05      149          LDX  #$05       *ZGNCD=GNCD
9016:BD 25 8F   150 LOOP12  LDA  GNCD-1,X
9019:95 EA      151          STA  ZGNCD-1,X
901B:CA         152          DEX
901C:D0 F8      153          BNE  LOOP12

901E:20 30 92   155          JSR  DARYCLR    *DATA ARRAY CLEAR
9021:20 F3 94   156          JSR  PTRSET

9024:A0 00      158          LDY  #$00       *LNMULT
9026:20 8E 92   159          JSR  LPLMTST    *LOOPING LIMIT SET SUBR

9029:20 65 92   161          JSR  MSWAIT     *MAIN SW(0/5V) WAITING & BELL

902C:20 D4 92   163          JSR  INITIM     *INITIAL TIME SET SUBR

902F:20 01 91   165 LOOP20  JSR  MULTDAT    *MULTIPLE DATA SAMPLING SUBR

9032:20 9D 92   167          JSR  STOPTM     *LAP TIME SETTING SUBR

9035:20 1F 93   169          JSR  STORE       *STORE DATA INTO DATA ARRAY

9038:AD 25 8F   171          LDA  CHOICE
903B:C9 D9      172          CMP  #'Y'
903D:D0 03      173          BNE  EXIT14
```

```
903F:20 0E 94   174          JSR  DISPLY     *DISPLAY DATA ON SCRN


9042:20 6E 92   177 EXIT14   JSR  MSWCHK     *MAIN SW CHECK STEP
9045:F0 08      178          BEQ  EXIT15

9047:C6 CE      180          DEC  ZLPNUM-1   *LOOPING TIME COUNT
9049:D0 E4      181          BNE  LOOP20
904B:C6 CF      182          DEC  ZLPNUM
904D:10 E0      183          BPL  LOOP20


904F:20 9D 92   185 EXIT15   JSR  STOPTM     *LAP TIME SETTING SUBR
9052:20 FE 94   186          JSR  BELL

9055:20 7C 92   188          JSR  SCRNCLR    *CLEAR SCRN
9058:A2 2C      189          LDX  #$2C       *CAPTION-5 DISPLAY
905A:20 13 93   190          JSR  CAPTDISP

905D:A0 00      192          LDY  #$00       *LNMULT
905F:A2 00      193          LDX  #$00       *EXCMLN
9061:20 5E 94   194          JSR  EXLPNM     *EXEC'TED LP NO CALC

9064:A0 24      196          LDY  #$24       *EXCMLN
9066:A2 00      197          LDX  #$00       *GTOT
9068:86 09      198          STX  XSAV2
906A:20 7A 94   199          JSR  MULTADD    *MULT DATA ADD SUBR

906D:20 C6 93   201          JSR  ARYDISP

9070:A2 00      203          LDX  #$00       *GTOT
9072:20 EE 93   204          JSR  GTOTLAPT   *GTOT & LAPTIME & BELL
9075:60         205          RTS
9076:EA EA      206          DFB  $EA,$EA



                208 ****************************************
9078:           209 *SINGLE DATA PICKING ROUTINE
                210 ****************************************

9078:A2 36      212 GETSNGL LDX  #$36       *CAPTION-6 DISPLAY & CHOICE
907A:20 84 92   213          JSR  HDLNCHO
```

```
907D:20 30 92  215          JSR  DARYCLR    *DATA ARRAY CLEAR SUBR

9080:A6 19     217          LDX  ZNMCHN     *ZGNCD=GNCD,DIFFERENT SEQ FROM MULT
9082:A0 04     218          LDY  #$04
9084:B9 26 8F  219 LOOP32   LDA  GNCD,Y
9087:95 EB     220          STA  ZGNCD,X
9089:88        221          DEY
908A:CA        222          DEX
908B:10 F7     223          BPL  LOOP32

908D:A0 00     225          LDY  #$00       *LNMULT
908F:20 8E 92  226          JSR  LPLMTST    *LOOPING LIMIT SET SUBR

9092:20 65 92  228          JSR  MSWAIT     *MAIN SW(0/5V) WAITING & BELL
9095:20 D4 92  229          JSR  INITIM     *INITIAL TIME SETTING SUBR
9098:29 F0     230          AND  #$F0       *4LSB-MASKING OF CLOCK+3
909A:8D 3D 8F  231          STA  REFTIM2    *1-15 SEC

909D:20 F3 94  233 LOOP41   JSR  PTRSET

90A0:AE 0D 8F  235          LDX  LNSNGL     *CHN4V+2=LNSNGL(H)
90A3:86 FF     236          STX  CHN4V+2
90A5:AE 0C 8F  237          LDX  LNSNGL-1   *CHN4V+1=LNSNGL(L)
90A8:D0 02     238          BNE  SKIP31
90AA:C6 FF     239          DEC  CHN4V+2
90AC:86 FE     240 SKIP31   STX  CHN4V+1

90AE:AE 0F 8F  242          LDX  INTVL      *XSAV2=INTVL(H)
90B1:86 09     243          STX  XSAV2
90B3:AE 0E 8F  244          LDX  INTVL-1
90B6:86 08     245          STX  XSAV1

90B8:20 C8 91  247          JSR  SNGLDAT

90BB:AD 25 8F  249          LDA  CHOICE
90BE:C9 D9     250          CMP  #'Y'
90C0:D0 03     251          BNE  EXIT33
90C2:20 B2 93  252          JSR  LNDISP     *EXEC'TED LP NO DISP

90C5:20 FF 92  254 EXIT33   JSR  TMSTEP2    *TIME STEP CHECK SUBR

90C8:20 6E 92  256          JSR  MSWCHK     *MAIN SW VLTG CHK SUBR
90CB:F0 08     257          BEQ  EXIT22
```

```
90CD:C6 CE      259          DEC  ZLPNUM-1
90CF:D0 CC      260          BNE  LOOP41
90D1:C6 CF      261          DEC  ZLPNUM
90D3:10 C8      262          BPL  LOOP41

90D5:20 9D 92   264 EXIT22   JSR  STOPTM     *LAPTIME SETTING
90D8:20 FE 94   265          JSR  BELL
90DB:20 C8 92   266          JSR  LAPTMV

90DE:20 7C 92   268          JSR  SCRNCLR
90E1:A2 47      269          LDX  #$47       *CAPTION-7 DISPLAY LOOP
90E3:20 13 93   270          JSR  CAPTDISP

90E6:A0 00      272          LDY  #$00       *LNMULT
90E8:A2 10      273          LDX  #$10       *EXCSLN
90EA:20 5E 94   274          JSR  EXLPNM     *EXEC'TED LOOPING NO CALC

90ED:A0 02      276          LDY  #$02       *LNSNGL
90EF:A2 35      277          LDX  #$35       *STOT
90F1:86 09      278          STX  XSAV2
90F3:20 7A 94   279          JSR  MULTADD    *STOT ADD SUBR

90F6:20 C6 93   281          JSR  ARYDISP    *ARRAY DISPLAY SUBR

90F9:A2 35      283          LDX  #$35       *STOT
90FB:20 EE 93   284          JSR  GTOTLAPT   *STOT & LAP TIME DISP & BELL

90FE:60         286          RTS
90FF:EA EA      287          DFB  $EA,$EA



                289 ****************************************
9101:           290 *OTHER SUBROUTINES AND CONSTANTS
                291 ****************************************


9101:           293 *MULTIPLE DATA GETTING SUBROUTINE

9101:AE 09 8F   295 MULTDAT LDX  MULTSZ     *XSAV2= MULTSZ(H)
9104:86 09      296          STX  XSAV2
9106:AE 08 8F   297          LDX  MULTSZ-1   *XSAV1=MULTSZ(L)
9109:D0 02      298          BNE  SKIP15
```

```
910B:C6 09       299           DEC  XSAV2
910D:86 08       300 SKIP15    STX  XSAV1

910F:A5 EF       302           LDA  ZGNCD+4     *1ST CHN SMPG PREP
9111:8D D0 C0    303           STA  AI13

9114:A9 00       305           LDA  #$00        *CHN VLTG CLEAR LOOP
9116:A2 06       306           LDX  #$06
9118:95 19       307 LOOP17    STA  CHN1V-1,X
911A:95 F9       308           STA  CHN3V-1,X
911C:CA          309           DEX
911D:D0 F9       310           BNE  LOOP17

911F:A6 19       312 LOOP18    LDX  ZNMCHN
9121:A5 EF       313           LDA  ZGNCD+4     *1ST CHN ACT SMPG & ADD
9123:8D D0 C0    314           STA  AI13
9126:20 0A 95    315           JSR  DELAY13
9129:18          316           CLC               *1ST CHN SMPG & ADD
912A:AC D1 C0    317           LDY  AI13+1
912D:AD D0 C0    318           LDA  AI13
9130:65 1A       319           ADC  CHN1V
9132:85 1A       320           STA  CHN1V
9134:A5 EE       321           LDA  ZGNCD+3     *2ND CHN SMPG PREPARATION
9136:8D D0 C0    322           STA  AI13        *
9139:98          323           TYA
913A:29 0F       324           AND  #$0F
913C:65 1B       325           ADC  CHN1V+1
913E:85 1B       326           STA  CHN1V+1
9140:90 02       327           BCC  SKIP16
9142:E6 1C       328           INC  CHN1V+2
9144:CA          329 SKIP16    DEX
9145:EA          330           NOP
9146:F0 6C       331           BEQ  NEXT11

9148:A5 EE       333           LDA  ZGNCD+3     *2ND CHN ACT SMPG
914A:8D D0 C0    334           STA  AI13
914D:20 0A 95    335           JSR  DELAY13
9150:18          336           CLC               *2ND CHN SMPG & ADD
9151:AC D1 C0    337           LDY  AI13+1
9154:AD D0 C0    338           LDA  AI13
9157:65 1D       339           ADC  CHN2V
9159:85 1D       340           STA  CHN2V
915B:A5 ED       341           LDA  ZGNCD+2     *3RD CHN SMPG PREP
915D:8D D0 C0    342           STA  AI13        *
9160:98          343           TYA
```

```
9161:29 OF       344            AND  #$OF
9163:65 1E       345            ADC  CHN2V+1
9165:85 1E       346            STA  CHN2V+1
9167:90 O2       347            BCC  SKIP17
9169:E6 1F       348            INC  CHN2V+2
916B:CA          349 SKIP17     DEX
916C:EA          350            NOP
916D:FO 45       351            BEQ  NEXT11

916F:A5 ED       353            LDA  ZGNCD+2   *3RD CHN ACT SMPG
9171:8D DO CO    354            STA  AI13
9174:20 OA 95    355            JSR  DELAY13
9177:18          356            CLC             *3RD CHN SMPG & ADD
9178:AC D1 CO    357            LDY  AI13+1
917B:AD DO CO    358            LDA  AI13
917E:65 FA       359            ADC  CHN3V
9180:85 FA       360            STA  CHN3V
9182:A5 EC       361            LDA  ZGNCD+1   *4TH CHN SMPG PREP
9184:8D DO CO    362            STA  AI13      *
9187:98          363            TYA
9188:29 OF       364            AND  #$OF
918A:65 FB       365            ADC  CHN3V+1
918C:85 FB       366            STA  CHN3V+1
918E:90 O2       367            BCC  SKIP18
9190:E6 FC       368            INC  CHN3V+2
9192:CA          369 SKIP18     DEX
9193:EA          370            NOP
9194:FO 1E       371            BEQ  NEXT11

9196:A5 EC       373            LDA  ZGNCD+1   *4TH CHN ACT SMPG
9198:8D DO CO    374            STA  AI13
919B:20 OA 95    375            JSR  DELAY13
919E:18          376            CLC             *4TH CHN SMPG & ADD
919F:AC D1 CO    377            LDY  AI13+1
91A2:AD DO CO    378            LDA  AI13
91A5:65 FD       379            ADC  CHN4V
91A7:85 FD       380            STA  CHN4V
91A9:98          381            TYA
91AA:29 OF       382            AND  #$OF
91AC:65 FE       383            ADC  CHN4V+1
91AE:85 FE       384            STA  CHN4V+1
91B0:90 O2       385            BCC  NEXT11
91B2:E6 FF       386            INC  CHN4V+2

91B4:A5 EF       388 NEXT11     LDA  ZGNCD+4   *1ST CHN SMPG PREP
```

```
91B6:8D D0 C0  389              STA  AI13        *
91B9:A5 06     390              LDA  ZLOC        *5 CYCLES DELAY
91BB:EA        391              NOP

91BC:C6 08     393              DEC  XSAV1
91BE:D0 05     394              BNE  SKIP29
91C0:C6 09     395              DEC  XSAV2
91C2:10 01     396              BPL  SKIP29
91C4:60        397              RTS
91C5:4C 1F 91  398 SKIP29       JMP  LOOP18



91C8:          400 *SINGLE DATA GETTING SUBROUTINE

91C8:A6 19     402 SNGLDAT LDX  ZNMCHN      *X=ZNMCHN
91CA:B5 EB     403 LOOP33  LDA  ZGNCD,X     *EACH CHN SMPG PREP
91CC:8D D0 C0  404         STA  AI13
91CF:20 07 95  405         JSR  DELAY19

91D2:B5 EB     407         LDA  ZGNCD,X     *EACH CHN ACT SMPG
91D4:8D D0 C0  408         STA  AI13
91D7:C8        409         INY              *10 CYCLE DELAY & ARRAY PTR UPDATE
91D8:F0 04     410         BEQ  SKIP32      *
91DA:84 F9     411         STY  YSAV1       *DUMMY
91DC:D0 02     412         BNE  SKIP33      *
91DE:E6 07     413 SKIP32  INC  ZPAGE       *
91E0:18        414 SKIP33  CLC              *9 CYCLES DELAY & H-BY SAVE
91E1:AD D1 C0  415         LDA  AI13+1      *
91E4:48        416         PHA              *

91E5:AD D0 C0  418         LDA  AI13        *8LSB ADD
91E8:71 06     419         ADC  (ZLOC),Y
91EA:91 06     420         STA  (ZLOC),Y
91EC:C8        421         INY
91ED:68        422         PLA               *4MSB ADD
91EE:29 0F     423         AND  #$0F
91F0:71 06     424         ADC  (ZLOC),Y
91F2:91 06     425         STA  (ZLOC),Y
91F4:C8        426         INY
91F5:A9 00     427         LDA  #$00
91F7:71 06     428         ADC  (ZLOC),Y
```

```
91F9:91 06       429          STA  (ZLOC),Y
91FB:C8          430          INY

91FC:CA          432          DEX
91FD:D0 CB       433          BNE  LOOP33

91FF:38          435          SEC            *INTERVAL DELAY(MIN 15 MICRO-SEC)
9200:A5 09       436          LDA  XSAV2     *INT
9202:A6 08       437          LDX  XSAV1     *INT
9204:F0 03       438          BEQ  SKIP34    *INT
9206:CA          439 LOOP34   DEX            *INT
9207:D0 FD       440          BNE  LOOP34    *INT
9209:E9 01       441 SKIP34   SBC  #$01      *INT
920B:10 F9       442          BPL  LOOP34    *INT

920D:C6 FE       444          DEC  CHN4V+1   *LOOPING TIME COUNT
920F:D0 B7       445          BNE  SNGLDAT
9211:C6 FF       446          DEC  CHN4V+2
9213:10 B3       447          BPL  SNGLDAT
9215:60          448          RTS



9216:            450 *ARRAY CLEAR SUBROUTINE

9216:A9 00       452 PARYCLR LDA  #$00
9218:A0 14       453         LDY  #$14      *GTOT VAR CLEAR
921A:99 10 8F    454 LOOP16  STA  GTOT1-1,Y
921D:99 45 8F    455         STA  STOT1-1,Y
9220:88          456         DEY
9221:D0 F7       457         BNE  LOOP16

9223:8D 2E 8F    459         STA  EXCMLN-1  *EXC(M/S)LN CLEAR
9226:8D 2F 8F    460         STA  EXCMLN
9229:8D 3E 8F    461         STA  EXCSLN-1
922C:8D 3F 8F    462         STA  EXCSLN
922F:60          463         RTS

9230:20 F3 94    465 DARYCLR JSR  PTRSET

9233:AE 01 8F    467         LDX  ARYSZ     *XSAV1=ARYSZ(H)
9236:86 08       468         STX  XSAV1
9238:AE 00 8F    469         LDX  ARYSZ-1   *X=ARYSZ(L)
```

```
923B:D0 02      470         BNE  SKIP11
923D:C6 08      471         DEC  XSAV1

923F:A9 00      473 SKIP11  LDA  #$00
9241:C8         474 LOOP14  INY                 *DATA ARRAY CLEAR
9242:D0 02      475         BNE  SKIP12
9244:E6 07      476         INC  ZPAGE
9246:91 06      477 SKIP12  STA  (ZLOC),Y
9248:CA         478         DEX
9249:D0 F6      479         BNE  LOOP14
924B:C6 08      480         DEC  XSAV1
924D:10 F2      481         BPL  LOOP14
924F:60         482         RTS

9250:A9 00      484 TCARYCL LDA  #$00
9252:A2 06      485         LDX  #$06       *CLOCK VAR CLEAR
9254:9D 2F 8F   486 LOOP15  STA  TIMEIN-1,X
9257:9D 3F 8F   487         STA  LAPTIM1-1,X
925A:9D 35 8F   488         STA  LAPTIM2-1,X
925D:95 19      489         STA  CHN1V-1,X *CHN VLTG CLEAR
925F:95 F9      490         STA  CHN3V-1,X
9261:CA         491         DEX
9262:D0 F0      492         BNE  LOOP15
9264:60         493         RTS


9265:           495 *MAIN SW WAITING SUBROUTINE

9265:20 6E 92   497 MSWAIT  JSR  MSWCHK
9268:F0 FB      498         BEQ  MSWAIT
926A:20 FE 94   499         JSR  BELL
926D:60         500         RTS


926E:           502 *MAIN SW CHECKING SUBR

926E:A5 EB      504 MSWCHK  LDA  ZGNCD
9270:8D D0 C0   505         STA  AI13
9273:20 0B 95   506         JSR  DELAY0
9276:AD D1 C0   507         LDA  AI13+1
9279:29 0F      508         AND  #$0F
927B:60         509         RTS


927C:           511 *SCREEN CLEAR SUBROUTINE
```

```
927C:A9 00      513 SCRNCLR LDA  #$00
927E:85 22      514         STA  $22
9280:20 58 FC   515         JSR  $FC58
9283:60         516         RTS

9284:           518 *HEADLINE DISP & CHOICE SUBR

9284:20 13 93   520 HDLNCHO JSR  CAPTDISP
9287:20 35 FD   521         JSR  $FD35      *GET A KEY F. KEYBOARD
928A:8D 25 8F   522         STA  CHOICE
928D:60         523         RTS

928E:           525 *INITIAL LOOPING LMT SET SUBR

928E:B9 0B 8F   527 LPLMTST LDA  LNMULT,Y   *ZLPNUM=(LNMULT,Y)(H)
9291:85 CF      528         STA  ZLPNUM
9293:B9 0A 8F   529         LDA  LNMULT-1,Y *ZLPNUM-1=(LNMULT-1,Y)(L)
9296:D0 02      530         BNE  SKIP14
9298:C6 CF      531         DEC  ZLPNUM     *DECREASE ZLPNUM BY 1
929A:85 CE      532 SKIP14  STA  ZLPNUM-1
929C:60         533         RTS




929D:           535 *ELAPSED TIME CALCULATION SUBR

929D:AE C4 C0   537 STOPTM  LDX  CLOCK+4    *10 TO 1 MIL-SEC(BCD FORMAT)
92A0:AD C3 C0   538         LDA  CLOCK+3    *2^3--2^0 SEC & 100 MIL-SEC(BCD)
92A3:AC C2 C0   539         LDY  CLOCK+2    *2^11--2^4 SEC
92A6:8E 45 8F   540         STX  LAPTIM1+5 *LAPTIM1+5=10 TO 1 MIL-SEC(BCD FMT)

92A9:D8         542         CLD
92AA:AA         543         TAX
92AB:29 0F      544         AND  #$0F
92AD:8D 44 8F   545         STA  LAPTIM1+4 *LAPTIM1+4=100 MIL-SEC(BCD FMT)

92B0:8A         547         TXA
92B1:29 F0      548         AND  #$F0
92B3:38         549         SEC
92B4:ED 33 8F   550         SBC  TIMEIN+3
```

```
92B7:08         551          PHP                *P-REGISTER SAVE
92B8:4A         552          LSR  A            *DIVIDE A-REG BY 16
92B9:4A         553          LSR  A
92BA:4A         554          LSR  A
92BB:4A         555          LSR  A
92BC:8D 43 8F   556          STA  LAPTIM1+3 *LAPTIM1+3=2^3--2^0 SEC

92BF:98         558          TYA
92C0:28         559          PLP                *P-REGISTER RESTORE
92C1:ED 32 8F   560          SBC  TIMEIN+2
92C4:8D 42 8F   561          STA  LAPTIM1+2 *LAPTIM1+2=2^11--2^4 SEC
92C7:60         562          RTS

92C8:           564 *LAPTIME MOVE SUBR

92C8:A2 06      566 LAPTMV  LDX  #$06
92CA:BD 3F 8F   567 LOOP71  LDA  LAPTIM1-1,X
92CD:9D 35 8F   568          STA  LAPTIM2-1,X
92D0:CA         569          DEX
92D1:D0 F7      570          BNE  LOOP71
92D3:60         571          RTS

92D4:           573 *INITIAL TIME SET SUBROUTINE

92D4:AD C6 C0   575 INITIM  LDA  CLOCK+6    *CLOCK STOP
92D7:AD C5 C0   576          LDA  CLOCK+5    *CLOCK RESTART

92DA:AD C3 C0   578          LDA  CLOCK+3    *2^3--2^0 SEC READ
92DD:AE C2 C0   579          LDX  CLOCK+2    *2^11--2^4 SEC READ
92E0:8D 33 8F   580          STA  TIMEIN+3
92E3:8E 32 8F   581          STX  TIMEIN+2
92E6:60         582          RTS

92E7:           584 *TIME STEP CHECK SUBR:500 MIL-SEC

92E7:18         586 TMSTEP1 CLC
92E8:F8         587          SED              *SET DECIMAL FLAG
92E9:AD 3C 8F   588          LDA  REFTIM1    *REFTIM1=REFTIM1+INTERVAL
92EC:69 05      589          ADC  #$05       *INTERVAL=500 MIL-SEC
92EE:29 0F      590          AND  #$0F
92F0:8D 3C 8F   591          STA  REFTIM1
92F3:D8         592          CLD
```

```
92F4:AD C3 C0   593 LOOP19   LDA  CLOCK+3    *2^3--2^0 & 100 MIL-SEC(BCD) READ
92F7:29 0F      594          AND  #$0F
92F9:CD 3C 8F   595          CMP  REFTIM1
92FC:D0 F6      596          BNE  LOOP19
92FE:60         597          RTS

92FF:18         599 TMSTEP2 CLC              *TIME STEP=1 SEC
9300:AD 3D 8F   600          LDA  REFTIM2
9303:69 10      601          ADC  #$10
9305:8D 3D 8F   602          STA  REFTIM2
9308:AD C3 C0   603 LOOP51   LDA  CLOCK+3    *SEC &100 MIL-SEC
930B:29 F0      604          AND  #$F0
930D:CD 3D 8F   605          CMP  REFTIM2
9310:D0 F6      606          BNE  LOOP51
9312:60         607          RTS

9313:           609 *CAPTION DISPLAY SUBROUTINE

9313:BD 0C 95   611 CAPTDISP LDA CAPTN1,X
9316:F0 06      612          BEQ  RETURN
9318:20 ED FD   613          JSR  $FDED
931B:E8         614          INX
931C:D0 F5      615          BNE  CAPTDISP
931E:60         616 RETURN   RTS



931F:           618 *MULTIPLE DATA STORE SUBROUTINE

931F:A4 F9      620 STORE    LDY  YSAV1      *Y=YSAV1
9321:A6 19      621          LDX  ZNMCHN     *XSAV1=ZNMCHN
9323:86 08      622          STX  XSAV1

9325:A2 00      624          LDX  #$00       *CHN1V SAVE
9327:20 42 93   625          JSR  CHNVSTO
932A:F0 13      626          BEQ  NEXT12
932C:A2 03      627          LDX  #$03       *CHN2V SAVE
932E:20 42 93   628          JSR  CHNVSTO
9331:F0 0C      629          BEQ  NEXT12
9333:A2 E0      630          LDX  #$E0       *CHN3V STORE
9335:20 42 93   631          JSR  CHNVSTO
```

```
9338:F0 05      632          BEQ  NEXT12
933A:A2 E3      633          LDX  #$E3       *CHN4V STORE
933C:20 42 93   634          JSR  CHNVSTO

933F:84 F9      636 NEXT12   STY  YSAV1
9341:60         637          RTS


9342:C8         639 CHNVSTO  INY
9343:D0 02      640          BNE  SKIP19
9345:E6 07      641          INC  ZPAGE
9347:B5 1A      642 SKIP19   LDA  CHN1V,X
9349:91 06      643          STA  (ZLOC),Y
934B:C8         644          INY
934C:B5 1B      645          LDA  CHN1V+1,X
934E:91 06      646          STA  (ZLOC),Y
9350:C8         647          INY
9351:B5 1C      648          LDA  CHN1V+2,X
9353:91 06      649          STA  (ZLOC),Y
9355:C8         650          INY
9356:C6 08      651          DEC  XSAV1
9358:60         652          RTS

9359:           654 *DIRECT ADDITION OF CHNV INTO G/S-TOT

9359:A6 19      656 ADDTN    LDX  ZNMCHN
935B:86 08      657          STX  XSAV1

935D:A6 09      659          LDX  XSAV2      *GTOT1=$00 (XSAV2) STOT1=$35
935F:A0 00      660          LDY  #$00       *CHN1V ADDITION
9361:20 89 93   661          JSR  CHNVADD
9364:F0 1C      662          BEQ  NEXT13

9366:20 83 93   664          JSR  INCX5
9369:A0 03      665          LDY  #$03        *CHN2V ADDITION
936B:20 89 93   666          JSR  CHNVADD
936E:F0 12      667          BEQ  NEXT13

9370:20 83 93   669          JSR  INCX5
9373:A0 E0      670          LDY  #$E0        *CHN3V ADDITION
9375:20 89 93   671          JSR  CHNVADD
9378:F0 08      672          BEQ  NEXT13

937A:20 83 93   674          JSR  INCX5
```

```
937D:A0 E3      675          LDY  #$E3       *CHN4V ADDITION
937F:20 89 93   676          JSR  CHNVADD

9382:60         678 NEXT13   RTS

9383:18         680 INCX5    CLC
9384:8A         681          TXA
9385:69 05      682          ADC  #$05
9387:AA         683          TAX
9388:60         684          RTS

9389:18         686 CHNVADD  CLC
938A:B9 1A 00   687          LDA  CHN1V,Y
938D:7D 11 8F   688          ADC  GTOT1,X
9390:9D 11 8F   689          STA  GTOT1,X
9393:B9 1B 00   690          LDA  CHN1V+1,Y
9396:7D 12 8F   691          ADC  GTOT1+1,X
9399:9D 12 8F   692          STA  GTOT1+1,X
939C:B9 1C 00   693          LDA  CHN1V+2,Y
939F:7D 13 8F   694          ADC  GTOT1+2,X
93A2:9D 13 8F   695          STA  GTOT1+2,X
93A5:90 08      696          BCC  SKIP23
93A7:FE 14 8F   697          INC  GTOT1+3,X
93AA:D0 03      698          BNE  SKIP23
93AC:FE 15 8F   699          INC  GTOT1+4,X
93AF:C6 08      700 SKIP23   DEC  XSAV1
93B1:60         701          RTS



93B2:           703 *LOOPING NUMBER DISPLAY SUBR

93B2:20 58 FC   705 LNDISP   JSR  $FC58
93B5:E6 FC      706          INC  CHN3V+2
93B7:D0 02      707          BNE  SKIP42
93B9:E6 FD      708          INC  CHN4V
93BB:A5 FD      709 SKIP42   LDA  CHN4V      *H-BYTE
93BD:20 DA FD   710          JSR  $FDDA
93C0:A5 FC      711          LDA  CHN3V+2    *L-BYTE
93C2:20 DA FD   712          JSR  $FDDA
93C5:60         713          RTS
```

```
93C6:           715 *ARRAY DISPLAY SUBROUTINE

93C6:A2 09      717 ARYDISP LDX  #$09      *CAPTION 2 DISPLAY
93C8:20 13 93   718         JSR  CAPTDISP
93CB:20 F3 94   719         JSR  PTRSET

93CE:A5 19      721         LDA  ZNMCHN
93D0:8D 2C 8F   722         STA  DLN1
93D3:A9 04      723         LDA  #$04
93D5:8D 2D 8F   724         STA  DLN2
93D8:A9 02      725         LDA  #$02      *VTAB=02
93DA:85 22      726         STA  $22

93DC:20 35 FD   728 DISPASK JSR  $FD35
93DF:C9 D9      729         CMP  #'Y'
93E1:D0 06      730         BNE  SKIP51
93E3:20 C3 94   731         JSR  MEMDISP
93E6:4C DC 93   732         JMP  DISPASK

93E9:A9 00      734 SKIP51  LDA  #$00      *VTAB=00
93EB:85 22      735         STA  $22
93ED:60         736         RTS


93EE:           738 *GTOT/STOT & TOTAL LAPTIME DISP & BELL

93EE:A4 19      740 GTOTLAPT LDY ZNMCHN
93F0:84 09      741         STY  XSAV2
93F2:A0 05      742 LOOP28  LDY  #$05
93F4:BD 11 8F   743 LOOP27  LDA  GTOT1,X
93F7:20 DA FD   744         JSR  $FDDA
93FA:E8         745         INX
93FB:88         746         DEY
93FC:D0 F6      747         BNE  LOOP27
93FE:A9 8D      748         LDA  #$8D
9400:20 ED FD   749         JSR  $FDED
9403:C6 09      750         DEC  XSAV2
9405:D0 EB      751         BNE  LOOP28

9407:20 2A 94   753         JSR  LAPTDISP
940A:20 FE 94   754         JSR  BELL
940D:60         755         RTS
```

```
940E:           757 *CHN VLTG & LAP TIME DISPLAY SUBROUTINE

940E:A4 19      759 DISPLY   LDY  ZNMCHN

9410:A2 00      761          LDX  #$00       *CHN1V DISP
9412:20 48 94   762          JSR  CHNVDISP
9415:F0 13      763          BEQ  LAPTDISP
9417:A2 03      764          LDX  #$03       *CHN2V DISP
9419:20 48 94   765          JSR  CHNVDISP
941C:F0 0C      766          BEQ  LAPTDISP
941E:A2 E0      767          LDX  #$E0       *CHN3V DISP
9420:20 48 94   768          JSR  CHNVDISP
9423:F0 05      769          BEQ  LAPTDISP
9425:A2 E3      770          LDX  #$E3       *CHN4V DISP
9427:20 48 94   771          JSR  CHNVDISP

942A:AD 42 8F   773 LAPTDISP LDA LAPTIM1+2 *2^11--2^4 SEC
942D:20 DA FD   774          JSR  $FDDA
9430:AD 43 8F   775          LDA  LAPTIM1+3 *2^3--0 SEC
9433:20 DA FD   776          JSR  $FDDA
9436:AD 44 8F   777          LDA  LAPTIM1+4 *100 MIL SEC
9439:20 DA FD   778          JSR  $FDDA
943C:AD 45 8F   779          LDA  LAPTIM1+5 *10--1 MIL SEC
943F:20 DA FD   780          JSR  $FDDA
9442:A9 8D      781          LDA  #$8D
9444:20 ED FD   782          JSR  $FDED
9447:60         783          RTS


9448:B5 1C      785 CHNVDISP LDA CHN1V+2,X
944A:20 DA FD   786          JSR  $FDDA
944D:B5 1B      787          LDA  CHN1V+1,X
944F:20 DA FD   788          JSR  $FDDA
9452:B5 1A      789          LDA  CHN1V,X
9454:20 DA FD   790          JSR  $FDDA
9457:A9 A0      791          LDA  #$A0
9459:20 ED FD   792          JSR  $FDED
945C:88         793          DEY
945D:60         794          RTS
```

```
945E:          796 *FINAL EXEC'TED LOOP NO CALC

945E:A5 CF     798 EXLPNM  LDA  ZLPNUM
9460:10 04     799         BPL  SKIP91    *IF ZLPNUM(H)=>$00 THEN BR
9462:E6 CF     800         INC  ZLPNUM
9464:10 02     801         BPL  SKIP98    *UNCOND BR(NOW ZLPNUM(H)=$00)
9466:C6 CE     802 SKIP91  DEC  ZLPNUM-1
9468:38        803 SKIP98  SEC
9469:B9 0A 8F  804         LDA  LNMULT-1,Y
946C:E5 CE     805         SBC  ZLPNUM-1
946E:9D 2E 8F  806         STA  EXCMLN-1,X
9471:B9 0B 8F  807         LDA  LNMULT,Y
9474:E5 CF     808         SBC  ZLPNUM
9476:9D 2F 8F  809         STA  EXCMLN,X
9479:60        810         RTS

947A:          812 *MULT/SNGL DATA ADD SUBROUTINE

947A:20 8E 92  814 MULTADD JSR  LPLMTST   *ZLPNUM=EXCMLN OR LNSNGL
947D:20 F3 94  815         JSR  PTRSET

9480:A6 09     817 LOOP98  LDX  XSAV2
9482:A5 19     818         LDA  ZNMCHN
9484:85 08     819         STA  XSAV1
9486:C8        820 LOOP99  INY
9487:D0 02     821         BNE  SKIP97
9489:E6 07     822         INC  ZPAGE
948B:18        823 SKIP97  CLC
948C:B1 06     824         LDA  (ZLOC),Y  *1ST BYTE
948E:7D 11 8F  825         ADC  GTOT1,X
9491:9D 11 8F  826         STA  GTOT1,X
9494:C8        827         INY
9495:B1 06     828         LDA  (ZLOC),Y  *2ND BYTE
9497:7D 12 8F  829         ADC  GTOT1+1,X
949A:9D 12 8F  830         STA  GTOT1+1,X
949D:C8        831         INY
949E:B1 06     832         LDA  (ZLOC),Y  *3RD BYTE
94A0:7D 13 8F  833         ADC  GTOT1+2,X
94A3:9D 13 8F  834         STA  GTOT1+2,X
94A6:C8        835         INY
94A7:90 08     836         BCC  SKIP95
94A9:FE 14 8F  837         INC  GTOT1+3,X
94AC:D0 03     838         BNE  SKIP95
```

```
94AE:FE 15 8F  839         INC  GTOT1+4,X
94B1:8A        840 SKIP95  TXA
94B2:18        841         CLC
94B3:69 05     842         ADC  #$05
94B5:AA        843         TAX
94B6:C6 08     844         DEC  XSAV1
94B8:D0 CC     845         BNE  LOOP99

94BA:C6 CE     847         DEC  ZLPNUM-1
94BC:D0 C2     848         BNE  LOOP98
94BE:C6 CF     849         DEC  ZLPNUM
94C0:10 BE     850         BPL  LOOP98
94C2:60        851         RTS




94C3:          853 *DATA ARRAY DISPLAY SUBROUTINE; 20 LINES AT A TIME
94C3:          854 *TO CONTINUE TO DISPLAY, TYPE "Y"

94C3:A4 F9     856 MEMDISP LDY  YSAV1
94C5:A9 14     857         LDA  #$14       *XSAV1=20
94C7:85 08     858         STA  XSAV1
94C9:AE 2C 8F  859 LOOP25  LDX  DLN1       *XSAV2=DLN1
94CC:86 09     860         STX  XSAV2
94CE:AE 2D 8F  861 LOOP24  LDX  DLN2       *X=DLN2

94D1:C8        863 LOOP23  INY
94D2:D0 02     864         BNE  SKIP26
94D4:E6 07     865         INC  ZPAGE
94D6:B1 06     866 SKIP26  LDA  (ZLOC),Y
94D8:20 DA FD  867         JSR  $FDDA
94DB:CA        868         DEX
94DC:D0 F3     869         BNE  LOOP23
94DE:A9 A0     870         LDA  #$A0
94E0:20 ED FD  871         JSR  $FDED

94E3:C6 09     873         DEC  XSAV2
94E5:D0 E7     874         BNE  LOOP24
94E7:A9 8D     875         LDA  #$8D
94E9:20 ED FD  876         JSR  $FDED

94EC:C6 08     878         DEC  XSAV1
94EE:D0 D9     879         BNE  LOOP25
```

```
94F0:84 F9     880         STY  YSAV1
94F2:60        881         RTS

94F3:          883 *POINTER SET SUBROUTINE

94F3:AC 03 8F  885 PTRSET  LDY  STALOC    *ZPAGE=STALOC(H)
94F6:84 07     886         STY  ZPAGE
94F8:AC 02 8F  887         LDY  STALOC-1  *YSAV1=STALOC(L)
94FB:84 F9     888         STY  YSAV1
94FD:60        889         RTS

94FE:          891 *RING-BELL SUBROUTINE

94FE:A9 87     893 BELL    LDA  #$87
9500:20 ED FD  894         JSR  $FDED
9503:20 ED FD  895         JSR  $FDED
9506:60        896         RTS

9507:          898 *TIME DELAY SUBROUTINE

9507:A5 06     900 DELAY19 LDA  ZLOC
9509:EA        901         NOP
950A:EA        902 DELAY13 NOP
950B:60        903 DELAY0  RTS

950C:          905 *DATA AND CONSTANTS

950C:C2 C1 D3  907 CAPTN1  ASC  'BASE     ' *F-O;$4F,F-N;$4E
950F:C5 A0
9511:4F 4E A0  908         DFB  $4F,$4E,$A0,$BA
9514:BA
9515:C4 BF A0  909 CAPTN2  ASC  'D?       Y/N'
9518:D9 AF CE
951B:8D 00     910         DFB  $8D,$00
951D:C1 C3 D4  911 CAPTN3  ASC  'ACTV     '
9520:D6 A0
9522:4F 4E A0  912         DFB  $4F,$4E,$A0,$BA
9525:BA
9526:C4 BF A0  913         ASC  'D?       Y/N'
9529:D9 AF CE
952C:8D 00     914         DFB  $8D,$00
```

```
952E:C2 C1 D3   915 CAPTN4   ASC   'BASE      ' *F-F;$46
9531:C5 A0
9533:4F 46 46   916          DFB   $4F,$46,$46,$8D,$00
9536:8D 00
9538:C1 C3 D4   917 CAPTN5   ASC   'ACTV      '
953B:D6 A0
953D:4F 46 46   918          DFB   $4F,$46,$46,$8D,$00
9540:8D 00
9542:D3 CE C7   919 CAPTN6   ASC   'SNGL      '
9545:CC A0
9547:4F 4E A0   920          DFB   $4F,$4E,$A0,$BA
954A:BA
954B:C4 BF A0   921          ASC   'D?        Y/N'
954E:D9 AF CE
9551:8D 00      922          DFB   $8D,$00
9553:D3 CE C7   923 CAPTN7   ASC   'SNGL      '
9556:CC A0
9558:4F 46 46   924          DFB   $4F,$46,$46,$8D,$00
955B:8D 00
955D:           925 *PROGRAMMED BY TAEWOOK KANG, MAY 5,1983
955D:           926 *REVISED BY TAEWOOK KANG, AUG.23,1983
955D:           927 *RECONSTRUCTED BY TAEWOOK KANG, SEPT. 1,1983
955D:           928 *REMODIFIED FOR HOT W/F ANNEMO BY TAEWOOK KANG, FEB. 20,1984

*** SUCCESSFUL ASSEMBLY: NO ERRORS

?8F05 ACTSTL        ?9359 ADDTN
 C0D0 AI13           93C6 ARYDISP
 8F01 ARYSZ          94FE BELL
 8FA3 BSVLTG         9313 CAPTDISP
 950C CAPTN1        ?9515 CAPTN2
?951D CAPTN3        ?952E CAPTN4
?9538 CAPTN5        ?9542 CAPTN6
?9553 CAPTN7           1A CHN1V
   1D CHN2V            FA CHN3V
   FD CHN4V          9389 CHNVADD
 9448 CHNVDISP       9342 CHNVSTO
 8F25 CHOICE         C0C0 CLOCK
 9230 DARYCLR        950B DELAY0
 950A DELAY13        9507 DELAY19
 93DC DISPASK        940E DISPLY
 8F2C DLN1           8F2D DLN2
?8F07 ENDLOC         8F2F EXCMLN
 8F3F EXCSLN         9042 EXIT14
 904F EXIT15         90D5 EXIT22
```

```
 90C5 EXIT33         8FD6 EXIT74
 8FE3 EXIT75         945E EXLPNM
?900F GETMULT        9078 GETSNGL
 8F26 GNCD           8F11 GTOT1
?8F16 GTOT2         ?8F1B GTOT3
?8F20 GTOT4          93EE GTOTLAPT
 9284 HDLNCHO        9383 INCX5
 92D4 INITIM         8F0F INTVL
 942A LAPTDISP       8F40 LAPTIM1
 8F36 LAPTIM2        92C8 LAPTMV
 93B2 LNDISP         8F0B LNMULT
 8F0D LNSNGL         9016 LOOP12
 9241 LOOP14         9254 LOOP15
 921A LOOP16         9118 LOOP17
 911F LOOP18         92F4 LOOP19
 902F LOOP20         94D1 LOOP23
 94CE LOOP24         94C9 LOOP25
 93F4 LOOP27         93F2 LOOP28
 9084 LOOP32         91CA LOOP33
 9206 LOOP34         909D LOOP41
 9308 LOOP51         8FC3 LOOP70
 92CA LOOP71         8FAA LOOP72
 9480 LOOP98         9486 LOOP99
 928E LPLMTST        94C3 MEMDISP
 9265 MSWAIT         926E MSWCHK
 947A MULTADD        9101 MULTDAT
 8F09 MULTSZ         91B4 NEXT11
 933F NEXT12         9382 NEXT13
 8F10 NMCHN          8F80 ORIGIN
 9216 PARYCLR        94F3 PTRSET
 8F3C REFTIM1        8F3D REFTIM2
 931E RETURN         927C SCRNCLR
 923F SKIP11         9246 SKIP12
 929A SKIP14         910D SKIP15
 9144 SKIP16         916B SKIP17
 9192 SKIP18         9347 SKIP19
 93AF SKIP23         94D6 SKIP26
 91C5 SKIP29         90AC SKIP31
 91DE SKIP32         91E0 SKIP33
 9209 SKIP34         93BB SKIP42
 93E9 SKIP51         8FA0 SKIP77
 9466 SKIP91         94B1 SKIP95
 948B SKIP97         9468 SKIP98
 91C8 SNGLDAT        8F03 STALOC
 929D STOPTM         931F STORE
```

```
 8F46 STOT1          ?8F4B STOT2
?8F50 STOT3          ?8F55 STOT4
 9250 TCARYCL         8F30 TIMEIN
?92E7 TMSTEP1         92FF TMSTEP2
   08 XSAV1             09 XSAV2
   F9 YSAV1             EB ZGNCD
   06 ZLOC              CF ZLPNUM
   19 ZNMCHN            07 ZPAGE

   06 ZLOC              07 ZPAGE
   08 XSAV1             09 XSAV2
   19 ZNMCHN            1A CHN1V
   1D CHN2V             CF ZLPNUM
   EB ZGNCD             F9 YSAV1
   FA CHN3V             FD CHN4V
 8F01 ARYSZ           8F03 STALOC
?8F05 ACTSTL         ?8F07 ENDLOC
 8F09 MULTSZ          8F0B LNMULT
 8F0D LNSNGL          8F0F INTVL
 8F10 NMCHN           8F11 GTOT1
?8F16 GTOT2          ?8F1B GTOT3
?8F20 GTOT4           8F25 CHOICE
 8F26 GNCD            8F2C DLN1
 8F2D DLN2            8F2F EXCMLN
 8F30 TIMEIN          8F36 LAPTIM2
 8F3C REFTIM1         8F3D REFTIM2
 8F3F EXCSLN          8F40 LAPTIM1
 8F46 STOT1          ?8F4B STOT2
?8F50 STOT3          ?8F55 STOT4
 8F80 ORIGIN          8FA0 SKIP77
 8FA3 BSVLTG          8FAA LOOP72
 8FC3 LOOP70          8FD6 EXIT74
 8FE3 EXIT75         ?900F GETMULT
 9016 LOOP12          902F LOOP20
 9042 EXIT14          904F EXIT15
 9078 GETSNGL         9084 LOOP32
 909D LOOP41          90AC SKIP31
 90C5 EXIT33          90D5 EXIT22
 9101 MULTDAT         910D SKIP15
 9118 LOOP17          911F LOOP18
 9144 SKIP16          916B SKIP17
 9192 SKIP18          91B4 NEXT11
 91C5 SKIP29          91C8 SNGLDAT
 91CA LOOP33          91DE SKIP32
 91E0 SKIP33          9206 LOOP34
```

```
  9209 SKIP34          9216 PARYCLR
  921A LOOP16          9230 DARYCLR
  923F SKIP11          9241 LOOP14
  9246 SKIP12          9250 TCARYCL
  9254 LOOP15          9265 MSWAIT
  926E MSWCHK          927C SCRNCLR
  9284 HDLNCHO         928E LPLMTST
  929A SKIP14          929D STOPTM
  92C8 LAPTMV          92CA LOOP71
  92D4 INITIM         ?92E7 TMSTEP1
  92F4 LOOP19          92FF TMSTEP2
  9308 LOOP51          9313 CAPTDISP
  931E RETURN          931F STORE
  933F NEXT12          9342 CHNVSTO
  9347 SKIP19         ?9359 ADDTN
  9382 NEXT13          9383 INCX5
  9389 CHNVADD         93AF SKIP23
  93B2 LNDISP          93BB SKIP42
  93C6 ARYDISP         93DC DISPASK
  93E9 SKIP51          93EE GTOTLAPT
  93F2 LOOP28          93F4 LOOP27
  940E DISPLY          942A LAPTDISP
  9448 CHNVDISP        945E EXLPNM
  9466 SKIP91          9468 SKIP98
  947A MULTADD         9480 LOOP98
  9486 LOOP99          948B SKIP97
  94B1 SKIP95          94C3 MEMDISP
  94C9 LOOP25          94CE LOOP24
  94D1 LOOP23          94D6 SKIP26
  94F3 PTRSET          94FE BELL
  9507 DELAY19         950A DELAY13
  950B DELAY0          950C CAPTN1
 ?9515 CAPTN2         ?951D CAPTN3
 ?952E CAPTN4         ?9538 CAPTN5
 ?9542 CAPTN6         ?9553 CAPTN7
  C0C0 CLOCK           C0D0 AI13
```

```
100  TEXT : HOME : FLASH : HTAB 7: PRINT "FULL CONV OF SYS-DATA": NORMAL
200  DIM A(9),B(9),C(19),D(9),E(9),G(9),GA(9)
210  GOSUB 9700: REM  HIMEM SETTING
290 K = 0
300 ARYSZ = 36609 + K:STALOC = 36611 + K:ACTSTL = 36613 + K
310 ENLOC = 36615 + K:MULTSZ = 36617 + K:MLPNM = 36619 + K
320 SLPNM = 36621 + K:INERVL = 36623 + K:NMCHN = 36624 + K
325 GSTAL = 36625 + K:GNCD = 36646 + K:TYPE = 36651 + K
330 MEXCLN = 36655 + K:TIMEIN = 36656 + K:L2AP = 36662 + K
340 R1EF = 36668 + K:R2EF = 36669 + K:SEXCLN = 36671 + K
350 L1AP = 36672 + K:S1TAL = 36678 + K:CLSYS = 36730 + K
360 G1R = 36698 + K:XPNO = 36702 + K:XCRD = 36704 + K
370 YCRD = 36706 + K:ZCRD = 36708 + K:RTN = 36710 + K
380 CURR = 36712 + K:TEMP = 36714 + K:OHR = 36716 + K:SVLTG = 36718
390 XDTE = 36732 + K:PARA = 36608 + K:HOT1 = 36736 + K
500 D$ =  CHR$ (4):R$ =  CHR$ (13):H = 256
1000  POKE 34,Z1: HOME : PRINT : INPUT " DRIVE NO.? ";D%
1010  POKE 34,2: HOME : PRINT R$;D$;"CATALOG,D";D%
1020  INPUT " DATA ACQ TYPE =? (M/S) ";TY$
1030  INPUT " EXP. NO.= ";XP
1040 F$ = "S-" +  STR$ (XP) + "-" + TY$:G$ = F$ + "-TS"
1050  PRINT R$;D$;"BLOAD ";F$;",D";D%: HOME : VTAB 2: INVERSE : PRINT F$;" ,D";D%: NORMAL : POKE 34,3
1060 NC =  PEEK (NMCHN):A% = 4:NP = 7
1070  GOSUB 6000: REM      GAIN CODE CALC;(D(I)) ,(GA(I))
1080  IF TY$ >  < "S" THEN  GOTO 1100
1090  GOSUB 3000: GOTO 1110: REM  SNGL CONV
1100  GOSUB 2000: REM    MULT CONV
1110  GOSUB 4000
1120  INPUT " WANT TO CONV TS ? (Y/N) ";AN$
1130  IF  LEFT$ (AN$,1) >  < "Y" THEN  GOTO 1150
1140  PRINT R$;D$;"BLOAD ";G$;",D";D%: VTAB 2: INVERSE : PRINT G$;" ,D";D%: NORMAL : POKE 34,3: GOSUB 4500
1150  PRINT R$;D$;"RUN MENU,D1"
1160  END
2000 DS =  PEEK (MULTSZ) * H +  PEEK (MULTSZ - 1):DAS=DS:DBS=DS
2010 NBD =  PEEK (SEXCLN) * H +  PEEK (SEXCLN - 1):NAD =  PEEK (MEXCLN) * H +  PEEK (MEXCLN - 1)
2020  FOR I = 0 TO NC - 1
2030 A(I) = 0:B(I) = 0
2040  FOR K = 0 TO 4
2050 B(I) = B(I) +  PEEK (S1TAL + 5 * I + K) * (H ^ K) / (DBS * NBD)
2060 A(I) = A(I) +  PEEK (GSTAL + 5 * I + K) * (H ^ K) / (DAS * NAD)
2070  NEXT K
2080 B(I) = D(I) * B(I) / 4095 / GA(I):A(I) = D(I) * A(I) / 4095 / GA(I)
2090  NEXT I
2100 MSB =  INT ( PEEK (L2AP + 5) / 16):MTB =  PEEK (L2AP + 5) - MSB * 16
2110 TBM =  PEEK (L2AP + 2) * 16 +  PEEK (L2AP + 3) +  PEEK (L2AP + 4) / 10 + MSB / 100 + MTB / 1000
```

```
2120 MSA =  INT ( PEEK (L1AP + 5) / 16):MTA =  PEEK (L1AP + 5) - MSA * 16
2130 TAM =  PEEK (L1AP + 2) * 16 +  PEEK (L1AP + 3) +  PEEK (L1AP + 4) / 10 + MSA / 100 + MTA / 1000
2140  RETURN : END
3000 DBS =  PEEK (MULTSZ) * H +  PEEK (MULTSZ - 1):DAS =  PEEK (SEXCLN) * H +  PEEK (SEXCLN - 1)
3005 NBD =  PEEK (MEXCLN) * H +  PEEK (MEXCLN - 1):NAD =  PEEK (SLPNM) * H +  PEEK (SLPNM - 1)
3010  FOR I = 0 TO NC - 1
3020 A(I) = 0:B(I) = 0
3030  FOR K = 0 TO 4
3040 B(I) = B(I) +  PEEK (GSTAL + 5 * I + K) * (H ^ K) / (DBS * NBD)
3050 A(I) = A(I) +  PEEK (S1TAL + 5 * I + K) * (H ^ K) / (DAS * NAD)
3060  NEXT K
3070 B(I) = D(I) * B(I) / 4095 / GA(I):A(I) = D(I) * A(I) / 4095 / GA(I)
3080  NEXT I
3090 MSB =  INT ( PEEK (L1AP + 5) / 16):MTB =  PEEK (L1AP + 5) - MSB * 16
3100 TBM =  PEEK (L1AP + 2) * 16 +  PEEK (L1AP + 3) +  PEEK (L1AP + 4) / 10 + MSB / 100 + MTB / 1000
3110 MSA =  INT ( PEEK (L2AP + 5) / 16):MTA =  PEEK (L2AP + 5) - MSA * 16
3120 TAM =  PEEK (L2AP + 2) * 16 +  PEEK (L2AP + 3) +  PEEK (L2AP + 4) / 10 + MSA / 100 + MTA / 1000
3130  RETURN : END
4000  POKE 34,3: HOME
4010  INPUT " WANT TO PRINT OUT? (Y/N) ";AN$: IF  LEFT$ (AN$,1) >  < "Y" THEN  HOME : GOTO 4030
4020  PRINT R$;D$;"PR#1": POKE 1401,80: HOME
4030 C(0) =  PEEK (XDTE + 3) +  PEEK (XDTE + 2) * H +  PEEK (XDTE + 1) * H * H +  PEEK (XDTE) * H * H * H
4040 A$ =  STR$ (C(0))
4050  FOR J = 0 TO NP - 1:C(J) =  PEEK (XCRD + 2 * J) * H +  PEEK (XCRD + 2 * J + 1): NEXT J
4060 C(0) = C(0) / 1E3:C(1) = C(1) / 1E3:C(2) = C(2) / 1E3:C(6) = C(6) / 100:C(5) = C(5) / 10
4070  FOR I = 1 TO 39: PRINT "*";: NEXT I: PRINT : PRINT
4080 PRINT "FILE NAME & EXP NO = ";TAB(20);F$;TAB(41);"EXPERIMENT DATE = ";LEFT$ (A$,4);"/";MID$ (A$,5,2);"/";RIGHT$ (A$,2)
4090  PRINT "(X/R)= ";C(0);" CM"; TAB( 21);"(Y/O)= ";C(1);" CM"; TAB( 41);"(Z)= ";C(2);" CM"
4100  PRINT "CURR= ";C(4);" AMP"; TAB( 21);"OP TEMP= ";C(5);" DEG"; TAB( 41);"O/H R= ";C(6)
4120  FOR J = 0 TO NC - 1
4130  PRINT "GNCD:CH-";J + 1;"= ";G(J); TAB( 21);"REC GN:CH-";J + 1;"= ";GA(J); TAB( 41);"SVLTG:CH-";J + 1;"= ";E(J);" VOLT"
4140  PRINT "BASE VLTG: CH-";J + 1;"= ";B(J);" VOLT"; TAB( 41);
4150  PRINT "ACTV VLTG: CH-";J + 1;"= ";A(J);" VOLT"
4160  NEXT J
4170  PRINT "TOT SAMP TIM: BAS= ";TBM;" SEC"; TAB( 41);
4180  PRINT "TOT DAT PTS: BAS= ";DBS;"*";NBD;"= ";DBS * NBD
4190  PRINT "TOT SAMP TIM: ACT= ";TAM;" SEC"; TAB( 41);
4200  PRINT "TOT DAT PTS: ACT= ";DAS;"*";NAD;"= ";DAS * NAD
4210  PRINT : FOR I = 1 TO 39: PRINT "*";: NEXT I: PRINT
4220  PRINT R$;D$;"PR#0"
4230  RETURN : END
4500  POKE 34, PEEK (37): HOME : INPUT " WANT TO PRINT-OUT TS? (Y/N) ";AN$: IF  LEFT$ (AN$,1) >  < "Y" THEN  GOTO 4520
4510  POKE 34,2: PRINT R$;D$;"PR#1": POKE 1401,80
4520  HOME :SP =  PEEK (ACTSTL) * H +  PEEK (ACTSTL - 1)
4530  FOR J = 0 TO NAD - 1
```

```
4540 PT = SP + J * A% * NC
4550  FOR I = 0 TO NC - 1
4560 B(I) = 0:PR = PT + I * A%
4570  FOR K = 0 TO A% - 1
4580 B(I) = B(I) +  PEEK (PR + K) * (H ^ K) / DAS: NEXT K
4590 B(I) = B(I) * D(I) / 4095 / GA(I)
4600  NEXT I
4610  PRINT R$;J + 1;": ";
4620  FOR I = 0 TO NC - 1
4630 TP = I * 18 + 7:TP = TP -  INT (TP / 40) * 40: IF  PEEK (36) =  > TP THEN TP = TP + 40
4640  PRINT  TAB( TP);"C";I + 1;"=";B(I);
4650  NEXT I: NEXT J
4660  PRINT R$;D$;"PR#0": RETURN : END
6000  FOR I = 0 TO NC - 1
6010 G(I) =  INT ( PEEK (GNCD + 4 - I) / 16):C(I) =  PEEK (GNCD + 4 - I) - G(I) * 16:D(I) = G(I)
6020  IF D(I) >  = 4 THEN D(I) = D(I) - 4
6030 B% =  INT (D(I) / 2 + 0.5):C% = D(I) -  INT (D(I) / 2) * 2
6040 D(I) = 5 * ((2 ^ C%) / (10 ^ B%))
6045 GA(I) =  PEEK (G1R + I) / 100
6048 E(I) =  PEEK (SVLTG + 2 * I) * H +  PEEK (SVLTG + 2 * I + 1):E(I) = E(I) / 1E3
6050  NEXT I
6060  RETURN : END
9700 C =  PEEK (116):D =  PEEK (115): GOSUB 9900
9710  VTAB 3: PRINT "PRESENT HIMEM="; TAB( 33);A$
9720  VTAB 4: HTAB 32: PRINT "=";D + C * 256:Z1 =  PEEK (37)
9730  VTAB 5: PRINT "WANT TO CHANGE HIMEM Y/N?": VTAB 5: HTAB 32: GET AN$
9740  IF AN$ >  < "Y" THEN  GOTO 9780
9750  VTAB 5: HTAB 1: PRINT "INPUT NEW HIMEM "
9755 VTAB 5: HTAB 17: INVERSE : PRINT "(4-HEX;$XXXX)=": NORMAL : VTAB 5: HTAB 32: INPUT A$
9760  GOSUB 9800:B = C * 256 + D: VTAB 6: HTAB 32: PRINT "=";B
9770  HIMEM: B:Z1 =  PEEK (37)
9780  POKE 34,Z1: HOME : RETURN : END
9800  FOR I = 2 TO 5
9810 A(I) =  ASC ( MID$ (A$,I,1))
9820  IF A(I) =  < 57 THEN A(I) = A(I) - 48: GOTO 9840
9830 A(I) = A(I) - 55
9840  NEXT I
9850 C = A(2) * 16 + A(3):D = A(4) * 16 + A(5)
9860  RETURN : END
9900 A(1) =  INT (C / 16):A(2) = C - A(1) * 16:A(3) =  INT (D / 16):A(4) = D - A(3) * 16:A$ = "$"
9910  FOR I = 1 TO 4
9920  IF A(I) < 10 THEN A$ = A$ +  STR$ (A(I)): GOTO 9940
9930 A(I) = A(I) + 55:A$ = A$ +  CHR$ (A(I))
9940  NEXT I
9950  RETURN : END
```

```
100  TEXT : HOME : FLASH : HTAB 7: PRINT "VOLTAGE--VELOCITY CONV": NORMAL
110  HIMEM: 36864: REM  36864=$9000
200  DIM A(9),B(9),C(19),D(9),E(9),G(9),GA(9)
205  DIM BV(255),AV(255),XR(255),YH(255),ZA(255),DAS(255),NAD(255),TAM(255),RG(255),VE(255),GT(255)
500 D$ =  CHR$ (4):R$ =  CHR$ (13)
1000  VTAB 3: INPUT "ENTER NATURAL CONV CONST = ";NT
1010  INPUT "ENTER FORCED CONV CONST = ";ZT
1020  GOSUB 4000: REM   CAL-COEFF READING
1030  GOSUB 5000: REM     OLD SYSASSY READING
1050  FOR I = 0 TO SZ - 1
1060  GOSUB 3000
1065 VE(I) = V2:GT(I) = GD
1070  GOSUB 2000: PRINT
1075 K = I
1080  INPUT "CONTINUE? (Y/N) ";AN$: IF  LEFT$ (AN$,1) >  < "Y" THEN  GOTO 1100
1090  NEXT I
1100  GOSUB 5500
1110  PRINT R$;D$;"RUN MENU,D1"
1120  END
2000  POKE 34,2: HOME : FOR J = 1 TO 39: PRINT "*";: NEXT J: PRINT
2010  PRINT "EXP NO= ";I + NO; TAB( 16);"COEFF FN= ";CF$; TAB( 41);"BV= ";BV(I);" V"; TAB( 21);"AV= ";AV(I);" V"
2020  PRINT "X/R= ";XR(I);" CM"; TAB( 21);"Y/O= ";YH(I);" CM"; TAB( 41);"Z= ";ZA(I);" CM"; TAB( 21);"REC GN= ";RG(I)
2030  PRINT "TOT DAT PTS= ";DAS(I);" * ";NAD(I);" = ";DAS(I) * NAD(I); TAB( 41);"TOT MEAS TM= ";TAM(I);" SEC"
2050  PRINT "MEAN VELOCITY= ";VE(I);" CM/SEC"; TAB( 41);"GRADIENT= ";GT(I);" (CM/SEC)/VOLT"
2060  FOR J = 1 TO 39: PRINT "*";: NEXT J: PRINT
2070  RETURN : END
3000 V1 = 1.39
3010 AL = .0016
3020 OH = .20
3030 DT = OH / AL
3040 RP = 5.01 * (1 + OH) - 0.2
3050 BT = RP / (10 + RP) ^ 2
3060 SN = (NT ^ .8) * (BT ^ ( - .2)) * EO ^ ( - .4)
3070 DR = DT / BT * (1 / BV(I) ^ 2 - 1 / EO ^ 2)
3080 SO = (NT ^ .8) * (BT ^ ( - .2)) * BV(I) ^ ( - .4)
3090 SC = SO - SN
3100 TC = SN / (ZT * V1 ^ ( - .4) + SN) - SO / (ZT * V1 ^ ( - .4) + SO)
3110 TS = TC * ZT * V1 ^ ( - .4)
3120 DC = DR - SC - TS
3130 VG = 1 / AV(I) ^ 2 - BT * DC / DT
3140 VT = (1 / VG) ^ .5
3150 VN = VT - EO
3160 V2 = C(0) + C(1) * VN + C(2) * VN ^ 2 + C(3) * VN ^ 3 + C(4) * VN ^ 4
3170 ER = (V2 - V1) / V1:ER = ER * 100
3180 EL = 1
```

```
3190  IF  ABS (ER) > EL THEN V1 = V2
3200  IF  ABS (ER) > EL THEN  GOTO 3100
3210 GD = C(1) + 2 * C(2) * VN + 3 * C(3) * VN ^ 2 + 4 * C(4) * VN ^ 3
3220  RETURN : END
4000  INPUT "COEFF FILE NAME = ";CF$
4010  PRINT R$;D$;"OPEN ";CF$;",D2"
4020  PRINT R$;D$;"READ ";CF$
4030  INPUT C(0),C(1),C(2),C(3),C(4)
4040  INPUT EO
4050  PRINT R$;D$;"CLOSE ";CF$
4060  RETURN : END
5000  INPUT "SYS-ASSY FILE NAME = ";F$
5010  PRINT R$;D$;"OPEN ";F$;",D2"
5020  PRINT R$;D$;"READ ";F$
5030  INPUT SZ,NO,N9
5040  FOR I = 0 TO SZ - 1
5050  INPUT BV(I),AV(I),XR(I),YH(I),ZA(I),DAS(I),NAD(I),TAM(I),RG(I)
5060  NEXT I
5110  PRINT R$;D$;"CLOSE ";F$
5120  RETURN : END
5500  INPUT " REFINED SYS-ASSY FILE NAME= ";G$
5510 SZ = K + 1
5520  PRINT R$;D$;"OPEN ";G$;",D2"
5530  PRINT R$;D$;"DELETE ";G$
5540  PRINT R$;D$;"OPEN ";G$
5550  PRINT R$;D$;"WRITE ";G$
5560  PRINT SZ;",";NO;",";NO + SZ - 1;",";CF$
5570  FOR I = 0 TO SZ - 1
5580  PRINT VE(I);",";GT(I);",";XR(I);",";YH(I);",";ZA(I);",";DAS(I);",";NAD(I);",";TAM(I);",";RG(I)
5590  NEXT I
5600  PRINT R$;D$;"CLOSE ";G$
5610  RETURN : END
6000  REM  INPUT OF COORDINATES FROM THE AUTO-CORRELATION SPECTRUM
6010  HOME : HTAB 5: INVERSE : PRINT "AUTOCORRELATION ENTRY MODE": NORMAL
6020  VTAB 7
6030  PRINT "1)"; TAB( 5);"TIME=";TC(1): PRINT  TAB( 5);"AUTOCORR= ";AC(1)
6040  PRINT
6050  PRINT "2)"; TAB( 5);"TIME=";TC(2): PRINT  TAB( 5);"AUTOCORR= ";AC(2)
6060  PRINT
6070  PRINT "3)"; TAB( 5);"RECORDER SCALE FACTOR= ";SF
6080  PRINT : HTAB 5: PRINT "H FOR HELP
"
6090  NORMAL : PRINT
6100  PRINT  TAB( 5);"X FOR NO CHANGE"
6110  VTAB 18
```

```
6120  HTAB 10: PRINT "WHICH? ": GET A$
6130  VTAB 20
6140  IF A$ = "1" THEN  INPUT "1)..TIME,AUTOCORR= ";TC(1),AC(1)
6150  IF A$ = "1" THEN  GOTO 6000
6160  IF A$ = "2" THEN  INPUT "2)..TIME,AUTOCORR= ";TC(2),AC(2)
6170  IF A$ = "2" THEN  GOTO 6000
6180  IF A$ = "3" THEN  INPUT "SCALE FACTOR= ";SF
6190  IF A$ = "3" THEN  GOTO 6000
6200  IF A$ = "X" THEN  HOME
6210  IF A$ = "X" THEN  RETURN
6220  IF A$ = "H" THEN  GOSUB 7500
6230  IF A$ = "H" THEN  GOTO 6000
6240  RETURN : END
6500  REM  ...PRINT OUT SECTION FOR SPECTRAL CHARACTERISTICS
6510  PRINT : PRINT
6520  PRINT "MICROSCALE= ";LM(2);" CMS"
6530  PRINT : PRINT "FLUC.VELOCITY,FROM AUTOCORR= ";TV(2);" CM/S"
6540  PRINT : PRINT "DISSIPATION ENERGY= ";EP(1);" WATTS"
6550  PRINT : PRINT "KINETIC ENERGY= ";KE(1);"  WATTS"
6560  FOR I = 1 TO 35: PRINT "=";: NEXT I: PRINT
6570  RETURN : END
7000  REM ...CALCULATION OF SPECTRAL CHARACTERISTICS
7010 FT = AC(2) / AC(1)
7020 LM(1) = TC(2) /  SQR (2 * (1 - FT))
7030 LM(2) = LM(1) * VL(1)
7040 TV(2) = (AC(1) ^ .5) * GD
7050 TV(2) = TV(2) * SF
7060 NU = 2.87E - 7
7070 EP(1) = 15 * NU * TV(2) ^ 2 / (LM(2) ^ 2)
7080 KE(1) = (TV(2) * 1E - 2) ^ 2
7090  RETURN : END
7500  REM ..AUTOCORR READ OUT HELP SEGMENT
7510  HOME
7520  INPUT "MIN. ON TIME SCALE? ";XL(1)
7530  INPUT "MAX.ON TIME SCALE? ";XL(2)
7540  INPUT "SPAN IN CMS? ";XL(3)
7550  INPUT "DIST.YOU WISH TO INTERPRET? ";XL(4)
7560  INVERSE : PRINT "END OF TIME-AXIS INPUT": NORMAL
7570  INPUT "MIN.ON AUTOCORR SCALE? ";YL(1)
7580  INPUT "MAX.ON AUTOCORR SCALE? ";YL(2)
7590  INPUT "SPAN IN CMS? ";YL(3)
7600  INPUT "DIST.YOU WISH TO INTERPRET? ";YL(4)
7610  INVERSE : PRINT "END OF INPUT ": NORMAL
7620 XL(5) = (XL(2) - XL(1)) * XL(4) / XL(3)
7630 YL(5) = (YL(2) - YL(1)) * YL(4) / YL(3)
```

```
7640 TC(1) = 0:TC(2) = XL(5)
7650 AC(2) = YL(5) + YL(1)
7660  RETURN : END
```

```
100  REM   THIS ROUTINE CONTROLS THE PLOTTER AND WRITES COORDINATE DATA TO DISK
110  TEXT : HOME : FLASH : HTAB 7: PRINT "SPECTRUM CO-OR SAVE": NORMAL
120  HIMEM: 36864: REM  36864=$9000
200  DIM SV%(1,1023),KL(255),ED(255),DD(255)
300 PTRED = 36992:CNTRL = 25
500 D$ =  CHR$ (4):R$ =  CHR$ (13):YS = 10.6
1000  PRINT R$;D$;"BLOAD PLTRED1.OBJO,D1"
1100  POKE 34,2: HOME : VTAB 3
1110  PRINT " 1.INSERT PLOT ON PLOTTER"
1120  PRINT " 2.LOCATE THE PEN AT ORIGIN"
1130  PRINT " 3.(R,D,F,C)=CTRL KEY FOR FAST PEN MOVE"
1140  PRINT " 4.(I,J,K,M)=CTRL KEY FOR SLOW PEN MOVE"
1150  PRINT " 5.(/)=STORE CO-OR INTO MEMORY"
1160  PRINT " 6.<CR>=END KEY FOR CO-OR READING"
1165  PRINT : HTAB 2: INVERSE : PRINT "1,10 HZ MK MUST BE READING #1 & #2": NORMAL
1170  PRINT : PRINT "PRESS ANY KEY TO START": GET AN$
1200  CALL PTRED
1300 TP =  PEEK (CNTRL) +  PEEK (CNTRL + 1) * 256
1310 NP = TP - 2
1320  PRINT  TAB( 2);NP;" DATA POINTS ARE TAKEN"
1400  GOSUB 5000
1410 SZ = NP: GOSUB 3000: GOSUB 2000
1500  INPUT " WANT TO CONTINUE? (Y/N) ";AN$: IF  LEFT$ (AN$,1) >  < "N" THEN  GOTO 1100
1600  PRINT R$;D$;"RUN MENU,D1"
1700  END
2000  POKE 34,2: HOME : VTAB 3
2010  INPUT " WANT TO PRINT OUT? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 2030
2020  PRINT R$;D$;"PR#1": POKE 1401,80
2030  HOME : VTAB 3
2040  FOR I = 1 TO 39: PRINT "*";: NEXT I: PRINT
2050  PRINT "EXPERIMENT NO = ";XN
2060  PRINT "KINETIC ERG(VOLT^2) = ";KE
2070  PRINT "DISSIP  ERG(VOLT^2) = ";EP
2080  FOR I = 1 TO 39: PRINT "*";: NEXT I: PRINT
2090  PRINT R$;D$;"PR#0"
2100  RETURN : END
3000  POKE 34,2: HOME : PRINT " INTEGRATING..."
3005 X0% = SV%(0,0):X1% = SV%(0,1)
3010  FOR I = 2 TO TP - 1
3020 KL(I - 1) = (SV%(0,I) - X0%) / (X1% - X0%)
3030 ED(I - 1) = E0 + SV%(1,I) * (E2 - E0) / (YS * 78.74)
3040 DD(I - 1) = 20 * KL(I) + ED(I)
3050  NEXT I
3060  REM   INEGRATE FOR KE AND EP
3070  REM   INTEGRATION USES TRAPEZOIDAL RULE
```

```
3080 KE = 0.0
3090 EP = 0.0
3100  FOR I = 1 TO NP - 1
3110 DK = 10 ^ KL(I + 1) - 10 ^ KL(I)
3120 EM = (10 ^ (ED(I + 1) / 10) + 10 ^ (ED(I) / 10)) / 2
3130 DM = (10 ^ (DD(I + 1) / 10) + 10 ^ (DD(I) / 10)) / 2
3140 KE = KE + EM * DK
3150 EP = EP + DM * DK
3160  NEXT I
3170  RETURN : END
5000  REM  SAVE COORDINATE SET TO DISK
5030  INPUT "ENTER MIN.,MAX ON DB SCALE ";EO,E2
5040  INPUT "EXPERIMENT # = ";F$
5050 F$ = "SPEC-" + F$
5060 YS = 5.5
5070  PRINT R$;D$;"OPEN ";F$;",D2"
5075  PRINT R$;D$;"DELETE ";F$
5080  PRINT R$;D$;"OPEN ";F$
5085  PRINT R$;D$;"WRITE ";F$
5090  PRINT NP
5100  PRINT EO;",";E2
5110  FOR I = 0 TO TP - 1
5120  PRINT SV%(0,I);",";SV%(1,I)
5130  NEXT I
5140  PRINT R$;D$;"CLOSE ";F$
5150  HOME : PRINT F$;",D2 SAVE COMPLETED"
5160  RETURN : END
```

```
100  TEXT : HOME : FLASH : HTAB 7: PRINT "SPECTRUM ANALYSIS": NORMAL
110  HIMEM: 36864: REM  36864=$9000
200  DIM SV%(1,255),KL(255),ED(255),DD(255),KE(255),EP(255)
500 D$ =  CHR$ (4):R$ =  CHR$ (13):YS = 10.6
1000  VTAB 3: INPUT "STARTING/FINAL EXP NO = ";NO,N9
1010  FOR XN = NO TO N9
1020 F$ = "SPEC-" +  STR$ (XN): GOSUB 5000: REM  DATA READ
1030  GOSUB 3000: REM  CALCULATION STEP
1040  GOSUB 2000: REM  RESULT PRINT
1050 K = XN - NO:KE(K) = KE:EP(K) = EP
1060  INPUT " CONTINUE? (Y/N) ";AN$: IF  LEFT$ (AN$,1) >  < "Y" THEN  GOTO 1080
1070  NEXT XN
1080  GOSUB 5500
1090  PRINT R$;D$;"RUN MENU,D1"
1100  END
2000  POKE 34,2: HOME : VTAB 3
2010  INPUT " WANT TO PRINT OUT? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 2030
2020  PRINT R$;D$;"PR#1": POKE 1401,80
2030  HOME : VTAB 3
2040  FOR I = 1 TO 39: PRINT "*";: NEXT I: PRINT
2050  PRINT "EXPERIMENT NO = ";XN
2060  PRINT "KINETIC ERG(VOLT^2) = ";KE
2070  PRINT "DISSIP  ERG(VOLT^2) = ";EP
2080  FOR I = 1 TO 39: PRINT "*";: NEXT I: PRINT
2090  PRINT R$;D$;"PR#0"
2100  RETURN : END
3000  POKE 34,2: HOME : VTAB 3: PRINT F$;"  INTEGRATING..."
3010  FOR I = 1 TO SZ
3020 KL(I) = (SV%(0,I) - X0%) / (X1% - X0%)
3030 ED(I) = E0 + SV%(1,I) * (E2 - E0) / (YS * 78.74)
3040 DD(I) = 20 * KL(I) + ED(I)
3050  NEXT I
3060  REM  INEGRATE FOR KE AND EP
3070  REM  INTEGRATION USES TRAPEZOIDAL RULE
3080 KE = 0.0
3090 EP = 0.0
3100  FOR I = 1 TO SZ - 1
3110 DK = 10 ^ KL(I + 1) - 10 ^ KL(I)
3120 EM = (10 ^ (ED(I + 1) / 10) + 10 ^ (ED(I) / 10)) / 2
3130 DM = (10 ^ (DD(I + 1) / 10) + 10 ^ (DD(I) / 10)) / 2
3140 KE = KE + EM * DK
3150 EP = EP + DM * DK
3160  NEXT I
3170  RETURN : END
5000  POKE 34,2: HOME : VTAB 3: PRINT F$;"  READING..."
```

```
5010  PRINT R$;D$;"OPEN ";F$;",D2"
5020  PRINT R$;D$;"READ ";F$
5030  INPUT SZ
5040  INPUT EO,E2
5050  INPUT XO%,YO%
5060  INPUT X1%,Y1%
5070  FOR I = 1 TO SZ
5080  INPUT SV%(O,I),SV%(1,I)
5090  NEXT I
5100  PRINT R$;D$;"CLOSE ";F$
5110  RETURN : END
5500  INPUT "SPEC-ASSY FILE NAME = ";G$
5510 SZ = K + 1
5520  PRINT R$;D$;"OPEN ";G$;",D2"
5530  PRINT R$;D$;"DELETE ";G$
5540  PRINT R$;D$;"OPEN ";G$
5550  PRINT R$;D$;"WRITE ";G$
5560  PRINT SZ;",";NO;",";NO + SZ - 1
5570  FOR I = O TO SZ - 1
5580  PRINT KE(I);",";EP(I)
5590  NEXT I
5600  PRINT R$;D$;"CLOSE ";G$
5610  RETURN : END
```

```
100  TEXT : HOME : FLASH : HTAB 7: PRINT "K.E. & D.E. CALCULATION": NORMAL
110  HIMEM: 36864: REM 36864 = $9000
200  DIM A(9),B(9),C(19),D(9),E(9),G(9),GA(9)
205  DIM VE(255),GT(255),XR(255),YH(255),ZA(255),DAS(255),NAD(255),TAM(255),RG(255),KE(255),DE(255)
500 D$ =  CHR$ (4):R$ =  CHR$ (13)
1000  VTAB 3: GOSUB 4000: REM  REF'NED SYS FILE READING
1010  GOSUB 5000: REM       OLD SPECASY READING
1020  FOR I = 0 TO SZ - 1
1030 OK = KE(I):OD = DE(I)
1040  GOSUB 3000
1050  GOSUB 2000: PRINT
1055 K = I
1060  INPUT "CONTINUE? (Y/N) ";AN$: IF  LEFT$ (AN$,1) >  < "Y" THEN  GOTO 1080
1070  NEXT I
1080  GOSUB 5500
1090  PRINT R$;D$;"RUN MENU,D1"
1100  END
2000  POKE 34,2: HOME : FOR J = 1 TO 39: PRINT "*";: NEXT J: PRINT
2010  PRINT "EXP NO= ";I + NO; TAB( 16);"COEFF FN= ";CF$
2015  PRINT "KE= ";OK;" VOLT^2"; TAB( 41);"DE= ";OD;" VOLT^2"
2020  PRINT "X/R= ";XR(I);" CM"; TAB( 21);"Y/O= ";YH(I);" CM"; TAB( 41);"Z= ";ZA(I);" CM"; TAB( 21);"REC GN= ";RG(I)
2030  PRINT "TOT DAT PTS= ";DAS(I);" * ";NAD(I);" = ";DAS(I) * NAD(I); TAB( 41);"TOT MEAS TM= ";TAM(I);" SEC"
2040  PRINT "MEAN VELOCITY= ";VE(I);" CM/SEC"; TAB( 41);"GRADIENT= ";GT(I);" (CM/SEC)/VOLT"
2050  PRINT "KINETIC ERG= ";KE(I);" WATT"; TAB( 41);"DISSIP ERG= ";DE(I);" WATT"
2060  FOR J = 1 TO 39: PRINT "*";: NEXT J: PRINT
2070  RETURN : END
3000 KE(I) = (3 / 2) * ((GT(I) / 100 / RG(I)) ^ 2) * KE(I)
3100 DE(I) = (1.54E - 4 / ((VE(I) / 100) ^ 2)) * ((GT(I) / 100 / RG(I)) ^ 2) * DE(I)
3200  RETURN : END
4000  INPUT "REF'NED SYS FILE NAME= ";F$
4010  PRINT R$;D$;"OPEN ";F$;",D2"
4020  PRINT R$;D$;"READ ";F$
4030  INPUT SZ,NO,N9,CF$
4040  FOR I = 0 TO SZ - 1
4050  INPUT VE(I),GT(I),XR(I),YH(I),ZA(I),DAS(I),NAD(I),TAM(I),RG(I)
4060  NEXT I
4070  PRINT R$;D$;"CLOSE ";F$
4080  RETURN : END
5000  INPUT "SPEC-ASSY FILE NAME = ";H$
5010  PRINT R$;D$;"OPEN ";H$;",D2"
5020  PRINT R$;D$;"READ ";H$
5030  INPUT SZ,NO,N9
5040  FOR I = 0 TO SZ - 1
5050  INPUT KE(I),DE(I)
5060  NEXT I
```

```
5110  PRINT R$;D$;"CLOSE ";F$
5120  RETURN : END
5500  INPUT " FINAL DATA FILE NAME= ";G$
5510 SZ = K + 1:SF = 1E3
5520  PRINT R$;D$;"OPEN ";G$;",D2"
5530  PRINT R$;D$;"DELETE ";G$
5540  PRINT R$;D$;"OPEN ";G$
5550  PRINT R$;D$;"WRITE ";G$
5560 Y$ = ""
5570 Z$ =  STR$ (SZ):L = 6: GOSUB 8100
5580 Z$ =  STR$ (NO):L = 6: GOSUB 8100
5590 Z$ =  STR$ (NO + SZ - 1):L = 6: GOSUB 8100
5600 Z$ =  STR$ (SF):L = 6: GOSUB 8100
5610 Z$ = CF$:L = 10: GOSUB 8100
5620  PRINT Y$
5630  FOR I = 0 TO SZ - 1
5640 Y$ = ""
5650 Z = XR(I):Z = Z * SF:L = 5: GOSUB 8000
5660 Z = YH(I):Z = Z * SF:L = 5: GOSUB 8000
5670 Z = ZA(I):Z = Z * SF:L = 5: GOSUB 8000
5680 Z = VE(I):L = 12: GOSUB 8500
5690 Z = KE(I):L = 12: GOSUB 8500
5700 Z = DE(I):L = 12: GOSUB 8500
5710 Z = DAS(I):L = 4: GOSUB 8000
5720 Z = NAD(I):L = 4: GOSUB 8000
5730 Z = TAM(I):L = 12: GOSUB 8500
5735  PRINT Y$
5740  NEXT I
5750  PRINT R$;D$;"CLOSE ";G$
5760  RETURN : END
8000 Z$ =  STR$ (Z)
8100  IF  LEN (Z$) = L THEN  GOTO 8120
8110  FOR M = 1 TO L -  LEN (Z$):Z$ = " " + Z$: NEXT M
8120 Y$ = Y$ + Z$
8130  RETURN : END
8500  IF Z = 0 THEN Z1$ = "0.":SG$ = "+":PO$ = "00":SP$ = "+": GOTO 8640
8510  IF  SGN (Z) =  - 1 THEN Z =  ABS (Z):SG$ = "-": GOTO 8530
8520 SG$ = "+"
8530 X =  LOG (Z) /  LOG (10):X1 =  ABS ( INT (X))
8540  ON  SGN (X) + 2 GOTO 8550,8560,8570
8550  FOR XX = 1 TO X1:Z = Z * 10: NEXT XX:SP$ = "-": GOTO 8590
8560 SP$ = "+": GOTO 8590
8570  IF X1 = 0 THEN  GOTO 8560
8580  FOR XX = 1 TO X1:Z = Z / 10: NEXT XX:SP$ = "+"
8590  IF  LEN ( STR$ (Z)) = 1 THEN Z$ =  STR$ (Z) + ".": GOTO 8610
```

```
8600 Z$ =  STR$ (Z)
8610  IF  INT (X1 / 10) > 0 THEN PO$ =  STR$ (X1): GOTO 8630
8620 PO$ = "O" +  STR$ (X1)
8630 Z1$ =  LEFT$ (Z$,L - 5)
8640 Z$ = SG$ + Z1$ + "E" + SP$ + PO$
8650  GOSUB 8100
8660  RETURN : END
```

```
100  TEXT : HOME : FLASH : HTAB 7: PRINT "MAKING NORMALIZED CALASSY": NORMAL
200  DIM BV(255),AV(255),VE(255),DR(255),CV(255),CE(255)
210  HIMEM: 36864: REM  36864=$9000
500 D$ =  CHR$ (4):R$ =  CHR$ (13)
1000  VTAB 3: INPUT "ENTER NATURAL CONV CONSTANT = ";NT
1010  INPUT "ENTER FORCED CONV CONSTANT = ";ZT
1020  INPUT "ENTER NEW BASE VOLTAGE = ";EO
1030  GOSUB 5000: REM  OLD CALASSY READING
1040  GOSUB 3000: POKE 34,2: HOME
1050  FOR I = 0 TO SZ - 1
1060  GOSUB 3500
1070  GOSUB 2000: PRINT
1075 K = I
1080  INPUT "CONTINUE? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1100
1090  NEXT I
1100  GOSUB 5500
1110  PRINT R$;D$;"RUN MENU,D1"
1120  END
2000  PRINT
2010  PRINT "XP= ";I + NO; TAB( 11);"BV= ";BV(I); TAB( 26);"AV= ";AV(I);
2020  PRINT TAB( 41);"EO= ";EO; TAB( 11);"CV= ";CV(I); TAB( 26);"VE= ";VE(I)
2030  RETURN : END
3000 AL = .0016
3010 OH = 0.20
3020 DT = OH / AL
3030 RP = 5.01 * (1 + OH) - .2
3040 BT = RP / (10 + RP) ^ 2
3050 SN = (NT ^ .8) * (BT ^ ( - .2)) * EO ^ ( - .4)
3060  RETURN : END
3500 DR(I) = DT / BT * (1 / BV(I) ^ 2 - 1 / EO ^ 2)
3510 SO = (NT ^ .8) * (BT ^ ( - .2)) * BV(I) ^ ( - .4)
3520 TC = SN / (ZT * VE(I) ^ ( - .4) + SN) - SO / (ZT * VE(I) ^ ( - .4) + SO)
3530 TC = TC * ZT * VE(I) ^ ( - .4)
3540 SC = SO - SN
3550 DR(I) = DR(I) - SC - TC
3560 V2 = 1 / AV(I) ^ 2 - BT * DR(I) / DT
3570 CV(I) = (1 / V2) ^ .5
3590  RETURN : END
5000  INPUT "CAL-ASSY FILE NAME = ";F$
5010  PRINT R$;D$;"OPEN ";F$;",D2"
5020  PRINT R$;D$;"READ ";F$
5030  INPUT SZ,NO,N9
5040  FOR I = 0 TO SZ - 1
5050  INPUT BV(I),AV(I),VE(I)
5060  NEXT I
```

```
5110  PRINT R$;D$;"CLOSE ";F$
5120  RETURN : END
5500  INPUT "REFINED CAL DATA F.N.= ";G$
5505 SZ = K + 1
5510  FOR I = 0 TO SZ / 2 - 1
5530 AV(I) = (CV(2 * I) + CV(2 * I + 1)) / 2
5540 CE(I) = (VE(2 * I) + VE(2 * I + 1)) / 2
5550  NEXT I
5560  PRINT R$;D$;"OPEN ";G$;",D2"
5562  PRINT R$;D$;"DELETE ";G$
5564  PRINT R$;D$;"OPEN ";G$
5570  PRINT R$;D$;"WRITE ";G$
5580  PRINT EO;",";SZ / 2 + 1
5585  PRINT EO;",";"0."
5600  FOR I = 0 TO SZ / 2 - 1
5610  PRINT AV(I);",";CE(I)
5620  NEXT I
5630  PRINT R$;D$;"CLOSE ";G$
5640  RETURN : END
```

```
100  TEXT : HOME : FLASH : HTAB 7: PRINT "PLOTTING CAL-CURVE": NORMAL
200  DIM PL(1,19),CR%(1,39),CP$(1,0),CM$(0)
210  DIM B%(9),A$(19),CV%(1,1023),A(1,15)
220  GOSUB 6700: REM  HIMEM SETTING SUBR
230 START = 36992:FRAME = 37004:ZLNDRW = 37118:MKTIC = 37221
240 CAPTIC = 37397:CURVE = 37513:DOTPLT = 37617
250 COMMENT = 37662:ENPLT = 37711:LINDRW = 37826
260 RTT = 38338:WDT = 38340:HGT = 38342
270 TCNM = 38344:TSFT = 38346:TLNG = 38348:FLAG = 38350
280 XREF = 38353:YREF = 38355:XLIM = 38357:YLIM = 38359
290 XZERO = 38361:YZERO = 38363:CRARY = 38365:CPARY = 38367
300 CMARY = 38369:CVARY = 38371:XCRD = 38373:YCRD = 38375
310 CXH = 38377:CYH = 38379:LPNMH = 38381:SYMBOL = 38383
320 UPI = 200:H = 256:R$ =  CHR$ (13):D$ =  CHR$ (4)
1000  POKE 34,6: HOME : PRINT " PLT-DEFAULTS READ & PLOT START "
1010  PRINT R$;D$;"BLOAD BPLT1.OBJO,D1"
1020  GOSUB 3000: REM    PLT-DEFAULTS READ
1030  REM       INITIALIZE PLOTTER
1040  CALL START
1100  REM       FRAME (PL(I,J :I=0,1;J=0,1))
1110  POKE 34,6: HOME : PRINT " FRAME DRAW ": VTAB 9
1120 I1 = 0:I2 = 1:J1 = 0:J2 = 1: GOSUB 4000: PRINT
1130  INPUT "WANT TO CHANGE ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1150
1140  GOSUB 4200: GOTO 1110
1150  GOSUB 7000: CALL FRAME: REM      (XREF,YREF,XLIM,YLIM)
1200  REM       ZLNDRW (PL(I,J :I=0,1;J=0,1,2,3,4)
1210  POKE 34,6: HOME : PRINT " ZERO AXIS DRAW ": VTAB 9
1220 I1 = 0:I2 = 1:J1 = 0:J2 = 4: GOSUB 4000: PRINT
1230  INPUT "WANT TO CHANGE ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1250
1240  GOSUB 4200: GOTO 1210
1250  GOSUB 7100: CALL ZLNDRW: REM      ((X/Y)ZERO,(X/Y)REF,(X/Y)LIM,FLAG(X/Y))
1300  REM       MKTIC (PL(I,J :I=0,1;J=0,1,2,3,4,5,6)
1310  POKE 34,6: HOME : PRINT " TIC MARK  DRAW ": VTAB 9
1320 I1 = 0:I2 = 1:J1 = 0:J2 = 6: GOSUB 4000: PRINT
1330  INPUT "WANT TO CHANGE ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1350
1340  GOSUB 4200: GOTO 1310
1350  GOSUB 7200: CALL MKTIC: REM      ((X/Y)ZERO,TCNM(X/Y),TSFT(X/Y),TLNG(X/Y),CRARY,CR%(J,K))
1400  REM       CAPTIC  (PL(I,J :I=0,1;J=0,2,3,4,7,8,9,10,11,12),CP$(J,0 :J=0,1))
1410  POKE 34,6: HOME : PRINT " TIC CAPT  DRAW ": VTAB 9
1420 I1 = 0:I2 = 1:J1 = 0:J2 = 0: GOSUB 4000
1430 I1 = 0:I2 = 1:J1 = 2:J2 = 4: GOSUB 4000
1440 I1 = 0:I2 = 1:J1 = 7:J2 = 12: GOSUB 4000
1450 I1 = 0:I2 = 1: GOSUB 4100: PRINT
1460  INPUT "WANT TO CHANGE PL ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1480
1470  GOSUB 4200
```

```
1480  INPUT "WANT TO CHANGE CP$ ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1500
1490  GOSUB 4300: GOTO 1410
1500  GOSUB 7400: CALL CAPTIC: REM       (RTT(X,Y),WDT(X,Y),HGT(X,Y),(X/Y)ZERO,TCNM(X/Y),(CR/CP)ARY,CP$(I,0),CR%(J,K)
1600  REM      REDATA (PL(I,J :I:0,1;J=13,14)
1610  POKE 34,6: HOME : PRINT " READ DATA ": VTAB 9
1620 I1 = 0:I2 = 1:J1 = 13:J2 = 14: GOSUB 4000: PRINT
1630  INPUT "WANT TO CHANGE ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1650
1640  GOSUB 4200: GOTO 1610
1650  GOSUB 3300
1700  REM      CURVE/DOTPLT (PL(I,J :I=0;J=2,3,13,14,15,16),PL(1,17))
1710  POKE 34,6: HOME : PRINT " CURVE/DOT PLOT ": VTAB 9
1720 I1 = 0:I2 = 1:J1 = 2:J2 = 3: GOSUB 4000
1730 I1 = 0:I2 = 1:J1 = 13:J2 = 16: GOSUB 4000: PRINT  TAB( 22);"PL(1,17) = ";PL(1,17): PRINT
1740  INPUT "WANT TO CHANGE ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1760
1750  GOSUB 4200: GOTO 1710
1760 A = 2: GOSUB 7700
1770  ON A GOTO 1780,1790
1780  CALL CURVE: GOTO 1795: REM  (LPNMH,SYMBOL,RTT,WDT,HGT,CV%(I,K),CVARY)
1790  CALL DOTPLT: REM       (LPNMH,SYMBOL,RTT,WDT,HGT,CV%(I,K),CVARY)
1795  INPUT "CONTINUE  TO  DOT-PLOT? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "Y" THEN  GOTO 1600
1800  REM        REDATA (PL(I,J :I:0,1;J=2,3,13,14)
1810  POKE 34,6: HOME : PRINT " READ DATA ": VTAB 9
1815 I1 = 0:I2 = 1:J1 = 2:J2 = 3: GOSUB 4000
1820 I1 = 0:I2 = 1:J1 = 13:J2 = 14: GOSUB 4000: PRINT
1830  INPUT "WANT TO CHANGE ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1850
1840  GOSUB 4200: GOTO 1810
1850  GOSUB 4500: REM  CAL-COEFF FILE READING & CALC
1900  REM       CURVE/DOTPLT (PL(I,J :I=0;J=2,3,13,14,15,16),PL(1,17))
1910  POKE 34,6: HOME : PRINT " CURVE/DOT PLOT ": VTAB 9
1920 I1 = 0:I2 = 1:J1 = 2:J2 = 3: GOSUB 4000
1930 I1 = 0:I2 = 1:J1 = 13:J2 = 16: GOSUB 4000: PRINT  TAB( 22);"PL(1,17) = ";PL(1,17): PRINT
1940  INPUT "WANT TO CHANGE ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1960
1950  GOSUB 4200: GOTO 1910
1960 A = 1: FLASH : PRINT "PLOTTER CO-OR'S CALCULATING": NORMAL : GOSUB 7700
1970  ON A GOTO 1980,1990
1980  CALL CURVE: GOTO 1995: REM    (LPNMH,SYMBOL,RTT,WDT,HGT,CV%(I,K),CVARY)
1990  CALL DOTPLT: REM        (LPNMH,SYMBOL,RTT,WDT,HGT,CV%(I,K),CVARY)
1995  INPUT "CONTINUE TO CURVE-PLOT? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "Y" THEN  GOTO 1800
2000  REM      COMMENT (R,W,HT,X,Y,CM$(0))
2010  POKE 34,6: HOME : PRINT " COMMENT DRAW ": VTAB 9
2020  GOSUB 4400
2030  GOSUB 8000: CALL COMMENT: REM   (RTT,WDT,HGT,(X/Y)CRD,CMARY,CM$(0))
2040  INPUT "WANT TO CONTINUE ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) >  < "N" THEN  GOTO 2020
2200  POKE 34,6: HOME : PRINT " PLOT END & PLT-DEFAULTS SAVE "
2210  CALL ENPLT
```

```
2220  GOSUB 3100: REM    PLT-DEFAULTS SAVE
2230  PRINT R$;D$;"RUN MENU,D1"
2240  END
2500  REM    GENERAL LINE DRAW
2510  POKE 34,6: HOME : PRINT " GENERAL LINE DRAW": POKE 34,8: VTAB 9
2520  INPUT " A(0,11),A(1,11),A(0,12),A(1,12):RAW VALUE = ";A(0,11),A(1,11),A(0,12),A(1,12)
2530  GOSUB 8100: CALL LINDRW
2540  INPUT "CONTINUE ? (Y/N) ";AN$
2550  IF  LEFT$ (AN$,1) >  < "N" GOTO 2520
3000 FM$ = "CALDEF"
3010  PRINT R$;D$;"OPEN ";FM$;",D1": PRINT R$;D$;"READ ";FM$
3020  FOR J = 0 TO 19
3030  INPUT PL(0,J),PL(1,J)
3040  NEXT J
3050  INPUT CP$(0,0): INPUT CP$(1,0)
3060  PRINT R$;D$;"CLOSE ";FM$
3070  RETURN : END
3100 FM$ = "CALDEF"
3110  PRINT R$;D$;"OPEN ";FM$;",D1": PRINT R$;D$;"DELETE ";FM$
3120  PRINT R$;D$;"OPEN ";FM$: PRINT R$;D$;"WRITE ";FM$
3130  FOR J = 0 TO 19: PRINT PL(0,J);",";PL(1,J): NEXT J
3140  PRINT CP$(0,0): PRINT CP$(1,0)
3150  PRINT R$;D$;"CLOSE ";FM$
3160  RETURN : END
3300  REM     REDATA (PL(I,J :I=0,1;J=13,14)
3310  POKE 34, PEEK (37) - 1: HOME
3320  INPUT "REF'NED CAL FILE NAME = ";F$
3330  PRINT R$;D$;"OPEN ";F$;",D2"
3340  PRINT R$;D$;"READ ";F$
3350  INPUT EO,SZ
3360  FOR K = 0 TO SZ - 1
3370  INPUT A(0,1),A(1,1)
3380 A(0,1) = A(0,1) - EO
3390 CV%(0,K) =  INT (A(0,1) * PL(0,14) * PL(0,13) + 0.5)
3400 CV%(1,K) =  INT (A(1,1) * PL(1,14) * PL(1,13) + 0.5)
3410  NEXT K
3420  PRINT R$;D$;"CLOSE ";F$
3430 PL(0,15) = SZ
3440  RETURN : END
4000  FOR J = J1 TO J2
4010  FOR I = I1 TO I2
4020  PRINT  TAB( 2 + I * 20);"PL(";I;",";J;") = ";PL(I,J);
4030  NEXT I: PRINT
4040  NEXT J
4050  RETURN : END
```

```
4100  FOR I = I1 TO I2: PRINT "CP$(";I;",0) = ";CP$(I,0): NEXT I: RETURN : END
4200  POKE 34, PEEK (37) - 1: HOME
4210  INPUT "J ARG OF (I,J) OR CTRL-S   = ";J$
4220  IF J$ =  CHR$ (19) THEN  RETURN : END
4230  INPUT "I ARG OF (I,J) & VALUE:I,V = ";I$,V$
4240 I =  VAL (I$):J =  VAL (J$):V =  VAL (V$)
4250 PL(I,J) = V: GOTO 4210
4260  RETURN : END
4300  POKE 34, PEEK (37) - 1: HOME
4310  INPUT "I ARG OF (I,J) OR CTRL-S   = ";I$
4320  IF I$ =  CHR$ (19) THEN  RETURN : END
4330  INPUT "THEN TYPE STRING ";V$
4340 I =  VAL (I$):CP$(I,0) = V$: GOTO 4310
4350  RETURN : END
4400  REM
4410  INPUT "ROT,WDT,HGT =  ";R,W,HT
4420  INPUT "X,Y   INCH  =  ";X,Y
4430  INPUT "CM$(0)  = ";CM$(0)
4440  RETURN : END
4500  REM        REDATA (PL(I,J :I=0,1;J=2,3,13,14)
4510  POKE 34, PEEK (37) - 1: HOME
4520  INPUT "CAL-COEFF FILE NAME = ";F$
4530  PRINT R$;D$;"OPEN ";F$;",D2"
4540  PRINT R$;D$;"READ ";F$
4550  INPUT A(0,10),A(0,11),A(0,12),A(0,13),A(0,14)
4560  INPUT E0
4570  PRINT R$;D$;"CLOSE ";F$
4580  INPUT "# OF POLY-FITTED POINTS= ";SZ
4590 IC = (PL(0,3) - PL(0,2)) / PL(0,14) / SZ
4600  FLASH : PRINT "READING VALUES...": NORMAL
4610  FOR K = 0 TO SZ
4620 A(0,1) = PL(0,3) / PL(0,13) + K * IC
4630 A(1,1) = A(0,10) + A(0,11) * A(0,1) + A(0,12) * A(0,1) ^ 2 + A(0,13) * A(0,1) ^ 3 + A(0,14) * A(0,1) ^ 4
4640 CV%(0,K) =  INT (A(0,1) * PL(0,14) * PL(0,13) + 0.5)
4650 CV%(1,K) =  INT (A(1,1) * PL(1,14) * PL(1,13) + 0.5)
4660  NEXT K
4670 PL(0,15) = SZ + 1
4680  RETURN : END
6700 C =  PEEK (116):D =  PEEK (115): GOSUB 6900
6710  VTAB 3: PRINT "PRESENT HIMEM="; TAB( 33);A$
6720  VTAB 4: HTAB 32: PRINT "=";D + C * 256:Z1 =  PEEK (37)
6730  VTAB 5: PRINT "WANT TO CHANGE HIMEM Y/N = ": VTAB 5: HTAB 28: INPUT AN$
6740  IF AN$ >  < "Y" THEN  GOTO 6780
6750  VTAB 5: HTAB 1: PRINT "INPUT NEW HIMEM "
6755  VTAB 5: HTAB 17: INVERSE : PRINT "(4-HEX;$XXXX)=": NORMAL : VTAB 5: HTAB 32: INPUT A$
```

```
6760  GOSUB 6800:B = C * 256 + D: VTAB 6: HTAB 32: PRINT "=";B
6770  HIMEM: B:Z1 =  PEEK (37)
6780  POKE 34,Z1: HOME : RETURN : END
6800  FOR I = 2 TO 5
6810 A(0,I) =  ASC ( MID$ (A$,I,1))
6820  IF A(0,I) =  < 57 THEN A(0,I) = A(0,I) - 48: GOTO 6840
6830 A(0,I) = A(0,I) - 55
6840  NEXT I
6850 C = A(0,2) * 16 + A(0,3):D = A(0,4) * 16 + A(0,5)
6860  RETURN : END
6900 A(0,1) =  INT (C / 16):A(0,2) = C - A(0,1) * 16:A(0,3) =  INT (D / 16):A(0,4) = D - A(0,3) * 16:A$ = "$"
6910  FOR I = 1 TO 4
6920  IF A(0,I) < 10 THEN A$ = A$ +  STR$ (A(0,I)): GOTO 6940
6930 A(0,I) = A(0,I) + 55:A$ = A$ +  CHR$ (A(0,I))
6940  NEXT I
6950  RETURN : END
7000  REM    SUBR FRAME (PL(I,J :I=0,1;J=0,1)
7010 B%(0) =  INT (PL(0,0) * UPI) - 1: REM    XREF
7020 B%(1) =  INT (PL(1,0) * UPI) - 1: REM    YREF
7030 B%(2) =  INT ((PL(0,0) + PL(0,1)) * UPI) - 1: REM    XLIM
7040 B%(3) =  INT ((PL(1,0) + PL(1,1)) * UPI) - 1: REM    YLIM
7050  FOR I = 0 TO 3
7060  POKE XREF + I * 2, INT (B%(I) / H): POKE XREF + I * 2 - 1,B%(I) -  INT (B%(I) / H) * H
7070  NEXT I
7080  RETURN : END
7100  REM    SUBR ZLNDRW (PL(I,J :I=0,1;J=0,1,2,3,4)
7110  FOR I = 0 TO 1
7120  IF PL(I,2) * PL(I,3) =  > 0 THEN  POKE FLAG + I,0:B%(I) =  INT (PL(I,0) * UPI) - 1: GOTO 7140
7130  POKE FLAG + I,1:B%(I) =  INT ((PL(I,0) - PL(I,2) * PL(I,1) / (PL(I,3) - PL(I,2))) * UPI) - 1
7140  POKE XZERO + I * 2, INT (B%(I) / H): POKE XZERO + I * 2 - 1,B%(I) -  INT (B%(I) / H) * H
7150  NEXT I
7160  RETURN : END
7200  REM    SUBR MKTIC (PL(I,J :I=0,1;J=0,1,2,3,4,5,6)
7202  FOR I = 0 TO 1
7204 J = 1 - I
7206  POKE TSFT + J, INT (PL(J,5) * UPI): POKE TLNG + J,H -  INT (PL(J,6) * UPI)
7208  NEXT I
7210  FOR I = 0 TO 1
7220  IF PL(I,2) * PL(I,3) =  > 0 THEN B%(I) =  INT (PL(I,0) * UPI) - 1: GOTO 7240
7230 B%(I) =  INT ((PL(I,0) - PL(I,2) * PL(I,1) / (PL(I,3) - PL(I,2))) * UPI) - 1
7240  POKE XZERO + I * 2, INT (B%(I) / H): POKE XZERO + I * 2 - 1,B%(I) -  INT (B%(I) / H) * H
7250 J = 1 - I
7260 B%(I + 2) =  INT ((PL(J,3) - PL(J,2)) / PL(J,4)): POKE TCNM + J,B%(I + 2)
7270  FOR K = 0 TO B%(I + 2)
7280 CR%(J,K) =  INT ((PL(J,0) + PL(J,1) * PL(J,4) * K /  ABS (PL(J,3) - PL(J,2))) * UPI) - 1
```

```
7290  NEXT K
7300  NEXT I
7310 B%(5) =  PEEK (107) +  PEEK (108) * H: REM    PL(1,19) ARRAY ADDR
7320 B%(5) = B%(5) + 209 + 9 - 1: REM    ADDR OF 1ST ELEMENT OF CR% - 1
7330  POKE CRARY, INT (B%(5) / H): POKE CRARY - 1,B%(5) -  INT (B%(5) / H) * H
7340  RETURN : END
7400  REM    SUBR CAPTIC (PL(I,J :I=0,1;J=0,2,3,4,7,8,9,10,11,12),CP$(J,0 :J=0,1))
7410  FOR I = 0 TO 1
7420 J = 1 - I
7430 B%(I + 2) =  INT ((PL(J,3) - PL(J,2)) / PL(J,4) / PL(J,12))
7440  POKE TCNM + J,B%(I + 2): POKE RTT + J,PL(J,7)
7450  POKE WDT + J,PL(J,8)
7460  POKE HGT + J,PL(J,9)
7470  NEXT I
7480  FOR I = 0 TO 1
7490 J = 1 - I
7500  IF PL(I,2) * PL(I,3) =  > 0 THEN B%(I) =  INT ((PL(I,0) - PL(J,10)) * UPI) - 1: GOTO 7520
7510 B%(I) =  INT ((PL(I,0) - PL(I,2) * PL(I,1) / (PL(I,3) - PL(I,2)) - PL(J,10)) * UPI) - 1
7520  POKE XZERO + I * 2, INT (B%(I) / H): POKE XZERO + I * 2 - 1,B%(I) -  INT (B%(I) / H) * H
7530  FOR K = 0 TO B%(I + 2)
7540 CR%(J,K) =  INT ((PL(J,0) + PL(J,1) * PL(J,4) * PL(J,12) * K /  ABS (PL(J,3) - PL(J,2)) - PL(J,11)) * UPI) - 1
7550  NEXT K
7560  NEXT I
7570 B%(5) =  PEEK (107) +  PEEK (108) * H: REM     PL(1,19) ARRAY ADDR
7580 B%(5) = B%(5) + 209 + 9 - 1: REM     ADDR OF 1ST ELEMENT OF CR% - 1
7590  POKE CRARY, INT (B%(5) / H): POKE CRARY - 1,B%(5) -  INT (B%(5) / H) * H
7600 B%(6) = B%(5) + 169: REM    ADDR OF 1ST ELEMENT OF CP$ -1
7610  POKE CPARY, INT (B%(6) / H): POKE CPARY - 1,B%(6) -  INT (B%(6) / H) * H
7620  RETURN : END
7700  REM    SUBR CURVE/DOTPLT (PL(I,J :I=0,1;J=2,3,13,14,15,16),PL(1,17))
7702  POKE SYMBOL,PL(0,16): POKE RTT,PL(1,15)
7704  POKE WDT,PL(1,16): POKE HGT,PL(1,16)
7710  FOR K = 0 TO PL(0,15) - 1
7720  IF (CV%(0,K) / PL(0,13)) =  > PL(0,2) THEN K1 = K: GOTO 7740
7730  NEXT K
7740  FOR K = PL(0,15) - 1 TO K1 STEP  - 1
7750  IF (CV%(0,K) / PL(0,13)) =  < PL(0,3) THEN K2 = K: GOTO 7770
7760  NEXT K
7770 B%(5) = K2 - K1 + 1
7780  POKE LPNMH, INT (B%(5) / H): POKE LPNMH - 1,B%(5) -  INT (B%(5) / H) * H
7790 B%(0) = 2000:B%(1) = 1400:B%(2) = 1999:B%(3) = 1399
7800  FOR K = K1 TO K2
7810  FOR I = 0 TO 1
7820 CV%(I,K) =  INT ((PL(I,0) + (CV%(I,K) / PL(I,13) - PL(I,2)) * PL(I,1) / (PL(I,3) - PL(I,2))) * UPI) - 1
7830  IF CV%(I,K) =  > B%(I) THEN CV%(I,K) = B%(I + 2)
```

```
7840  NEXT I
7850  NEXT K
7860 B%(5) =  PEEK (107) +  PEEK (108) * H: REM        PL(1,19) ARRAY ADDR
7870 B%(5) = B%(5) + 209 + 169 + 15 + 10 + 27 + 67 + 9: REM       ADDR OF 1ST ELEMENT OF CV%
7880 B%(6) = B%(5) + 4 * K1 - 1: REM    ADDR OF K1TH ELEMENT OF CV% -1
7890  POKE CVARY, INT (B%(6) / H): POKE CVARY - 1,B%(6) -  INT (B%(6) / H) * H
7900  RETURN : END
8000  REM      COMMENT (R,W,HT,X,Y,CM$(0))
8010  POKE RTT,R: POKE WDT,W: POKE HGT,HT
8020 B%(0) =  INT (X * UPI) - 1:B%(1) =  INT (Y * UPI) - 1
8030  FOR I = 0 TO 1
8040  POKE XCRD + 2 * I, INT (B%(I) / H): POKE XCRD + 2 * I - 1,B%(I) -  INT (B%(I) / H) * H
8050  NEXT I
8060 B%(5) =  PEEK (107) +  PEEK (108) * H: REM    PL(1,19) ARRAY ADDR
8070 B%(5) = B%(5) + 209 + 169 + 15 + 7 - 1: REM    ADDR OF 1ST ELEMENT OF CM$-1
8080  POKE CMARY, INT (B%(5) / H): POKE CMARY - 1,B%(5) -  INT (B%(5) / H) * H
8090  RETURN : END
8100  REM   LINDRW (A(I,J:I=0,1;J=11,12),PL(I,J:I=0,1;J=0,1,2,3,14))
8110  FOR K = 11 TO 12
8120  FOR I = 0 TO 1
8130 J = (K - 11) * 2 + I
8140 B%(J) =  INT ((PL(I,0) + (A(I,K) * PL(I,14) - PL(I,2)) * PL(I,1) / (PL(I,3) - PL(I,2))) * UPI) - 1
8150  NEXT I
8160  NEXT K
8170  FOR I = 0 TO 3: POKE XCRD + I * 2, INT (B%(I) / H): POKE XCRD + I * 2 - 1,B%(I) -  INT (B%(I) / H) * H: NEXT I
8180  RETURN : END
```

```
100  TEXT : HOME : FLASH : HTAB 7: PRINT "PLOTTING PROBE-LOC": NORMAL
200  DIM PL(1,19),CR%(1,39),CP$(1,0),CM$(0)
210  DIM B%(9),A$(19),CV%(1,1023),A(1,15)
220  GOSUB 6700: REM  HIMEM SETTING SUBR
230 START = 36992:FRAME = 37004:ZLNDRW = 37118:MKTIC = 37221
240 CAPTIC = 37397:CURVE = 37513:DOTPLT = 37617
250 COMMENT = 37662:ENPLT = 37711:LINDRW = 37826
260 RTT = 38338:WDT = 38340:HGT = 38342
270 TCNM = 38344:TSFT = 38346:TLNG = 38348:FLAG = 38350
280 XREF = 38353:YREF = 38355:XLIM = 38357:YLIM = 38359
290 XZERO = 38361:YZERO = 38363:CRARY = 38365:CPARY = 38367
300 CMARY = 38369:CVARY = 38371:XCRD = 38373:YCRD = 38375
310 CXH = 38377:CYH = 38379:LPNMH = 38381:SYMBOL = 38383
320 UPI = 200:H = 256:R$ =  CHR$ (13):D$ =  CHR$ (4)
500  GOSUB 4500: REM   BATH-DEFAULTS READ
510  POKE 34,6: HOME : VTAB 7: PRINT " BATH DEFUALTS": VTAB 9
530 I1 = 0:I2 = 1:J1 = 1:J2 = 6: GOSUB 4700: PRINT
540  INPUT " WANT TO CHANGE ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 560
550  GOSUB 4800: GOTO 510
560  GOSUB 5000: REM  GRID PT-->VLTG CONV
600  POKE 34,6: HOME : PRINT " PLT-DEFAULTS READ & PLOT START "
610  PRINT R$;D$;"BLOAD BPLT1.OBJO,D1"
620  GOSUB 3000: REM   PLT-DEFAULTS READ
630  GOSUB 5300: REM  A(I,J)-->PL(I,J)
1030  REM      INITIALIZE PLOTTER
1040  CALL START
1100  REM      FRAME (PL(I,J :I=0,1;J=0,1))
1110  POKE 34,6: HOME : PRINT " FRAME DRAW ": VTAB 9
1120 I1 = 0:I2 = 1:J1 = 0:J2 = 1: GOSUB 4000: PRINT
1130  INPUT "WANT TO CHANGE ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1150
1140  GOSUB 4200: GOTO 1110
1150  GOSUB 7000: CALL FRAME: REM      (XREF,YREF,XLIM,YLIM)
1300  REM      MKTIC (PL(I,J :I=0,1;J=0,1,2,3,4,5,6)
1310  POKE 34,6: HOME : PRINT " TIC MARK  DRAW ": VTAB 9
1320 I1 = 0:I2 = 1:J1 = 0:J2 = 6: GOSUB 4000: PRINT
1330  INPUT "WANT TO CHANGE ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1350
1340  GOSUB 4200: GOTO 1310
1350  GOSUB 7200: CALL MKTIC: REM       ((X/Y)ZERO,TCNM(X/Y),TSFT(X/Y),TLNG(X/Y),CRARY,CR%(J,K))
1400  REM      CAPTIC  (PL(I,J :I=0,1;J=0,2,3,4,7,8,9,10,11,12),CP$(J,0 :J=0,1))
1410  POKE 34,6: HOME : PRINT " TIC CAPT  DRAW ": VTAB 9
1420 I1 = 0:I2 = 1:J1 = 0:J2 = 0: GOSUB 4000
1430 I1 = 0:I2 = 1:J1 = 2:J2 = 4: GOSUB 4000
1440 I1 = 0:I2 = 1:J1 = 7:J2 = 12: GOSUB 4000
1450 I1 = 0:I2 = 1: GOSUB 4100: PRINT
1460  INPUT "WANT TO CHANGE PL ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1480
```

```
1470  GOSUB 4200
1480  INPUT "WANT TO CHANGE CP$ ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1500
1490  GOSUB 4300: GOTO 1410
1500  GOSUB 7400: CALL CAPTIC: REM       (RTT(X,Y),WDT(X,Y),HGT(X,Y),(X/Y)ZERO,TCNM(X/Y),(CR/CP)ARY,CP$(I,O),CR%(J,K)
1600  REM  ELECTRODE SURFACE LINE DRAW
1610  POKE 34,6: HOME : PRINT "ELECTRODE DRAW"
1620 A(0,11) = 0:A(1,11) = A(1,1) - A(1,2):A(0,12) = A(0,2) / 2:A(1,12) = A(1,11)
1630  GOSUB 8100: CALL LINDRW
1640 A(0,11) = A(0,12):A(1,11) = A(1,12):A(0,12) = A(0,12):A(1,12) = A(1,1)
1650  GOSUB 8100: CALL LINDRW
1700  REM     COMMENT (R,W,HT,X,Y,CM$(0))
1710  POKE 34,6: HOME : PRINT " COMMENT DRAW ": VTAB 9
1720  GOSUB 4400
1730  GOSUB 8000: CALL COMMENT: REM   (RTT,WDT,HGT,(X/Y)CRD,CMARY,CM$(0))
1740  INPUT "WANT TO CONTINUE ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) >  < "N" THEN  GOTO 1720
1800  REM     REDATA (PL(I,J :I:0,1;J=13,14)
1810  POKE 34,6: HOME : PRINT " READ DATA ": VTAB 9
1815  INPUT " INITIAL POINT # = ";PNR:PNR = PNR - 1
1820 I1 = 0:I2 = 1:J1 = 13:J2 = 14: GOSUB 4000: PRINT
1830  INPUT "WANT TO CHANGE ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1850
1840  GOSUB 4200: GOTO 1810
1850  GOSUB 3300
1900  REM     CURVE/DOTPLT (PL(I,J :I=0;J=2,3,13,14,15,16),PL(1,17))
1910  POKE 34,6: HOME : PRINT " CURVE/DOT PLOT ": VTAB 9
1920 I1 = 0:I2 = 1:J1 = 2:J2 = 3: GOSUB 4000
1930 I1 = 0:I2 = 1:J1 = 13:J2 = 16: GOSUB 4000
1935  PRINT  TAB( 2);"PT NO= ";PNR + 1; TAB( 22);"PL(1,17)= ";PL(1,17): PRINT
1940  INPUT "WANT TO CHANGE ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 1960
1950  GOSUB 4200: GOTO 1910
1960 A = 2: GOSUB 7700
1970  ON A GOTO 1980,1990
1980  CALL CURVE: GOTO 1700: REM (LPNMH,SYMBOL,RTT,WDT,HGT,CV%(I,K),CVARY)
1990  CALL DOTPLT: REM       (LPNMH,SYMBOL,RTT,WDT,HGT,CV%(I,K),CVARY)
2000 B%(0) = CV%(0,0) - 40:B%(1) = CV%(1,0) - 40:PNR = PNR + 1
2010  POKE RTT,8: POKE WDT,2: POKE HGT,2
2020 CM$(0) =  STR$ (PNR)
2030  GOSUB 8030: CALL COMMENT
2090  POKE 34, PEEK (37) - 1: HOME
2100  INPUT " WANT TO CONTINUE? (Y/N) ";AN$: IF  LEFT$ (AN$,1) >  < "N" THEN  GOTO 1850
2110  INPUT " CONTINUE W/OTHER INDENT? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 2200
2120  POKE 34,6: HOME : VTAB 7: PRINT " BATH DEFUALTS": VTAB 9
2130 I1 = 0:I2 = 1:J1 = 1:J2 = 6: GOSUB 4700: PRINT
2140  INPUT " WANT TO CHANGE ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) = "N" THEN  GOTO 2160
2150  GOSUB 4800: GOTO 2120
2160  GOSUB 5000: REM   GRID PT-->VLTG CONV
```

```
2170  GOTO 1800
2200  POKE 34,6: HOME : PRINT " PLOT END & PLT-DEFAULTS SAVE "
2210  CALL ENPLT
2220  GOSUB 3100: REM   PLT-DEFAULTS SAVE
2230  GOSUB 4600: REM  BATH-DEFAULTS SAVE
2240  PRINT R$;D$;"RUN MENU,D1"
2250  END
3000  PRINT R$;D$;"OPEN BHPLDEF,D1"
3010  PRINT R$;D$;"READ BHPLDEF"
3020  FOR J = 0 TO 19
3030  INPUT PL(0,J),PL(1,J)
3040  NEXT J
3050  INPUT CP$(0,0): INPUT CP$(1,0)
3060  PRINT R$;D$;"CLOSE BHPLDEF"
3070  RETURN : END
3100  PRINT R$;D$;"OPEN BHPLDEF,D1"
3110  PRINT R$;D$;"DELETE BHPLDEF"
3120  PRINT R$;D$;"OPEN BHPLDEF"
3130  PRINT R$;D$;"WRITE BHPLDEF"
3140  FOR J = 0 TO 19: PRINT PL(0,J);",";PL(1,J): NEXT J
3150  PRINT CP$(0,0): PRINT CP$(1,0)
3160  PRINT R$;D$;"CLOSE BHPLDEF"
3170  RETURN : END
3300  REM     REDATA (PL(I,J :I=0,1;J=13,14)
3310  POKE 34, PEEK (37) - 1: HOME
3320  INPUT " ENTER (1.VLTG 2.LOC): WHICH? (1/2) ";CH: ON CH GOTO 3330,3340
3330  INPUT " TYPE RADIAL/AXIAL VLTG ";RVT,ZVT: GOSUB 5200: GOTO 3350
3340  INPUT " TYPE RADIAL/AXIAL LOC ";RL,ZL: GOSUB 5400
3350  HOME : GOSUB 5500
3380 CV%(0,0) =  INT (RL * PL(0,14) * PL(0,13) + 0.5)
3390 CV%(1,0) =  INT (ZL * PL(1,14) * PL(1,13) + 0.5)
3420 PL(0,15) = 1
3430  RETURN : END
4000  FOR J = J1 TO J2
4010  FOR I = I1 TO I2
4020  PRINT  TAB( 2 + I * 20);"PL(";I;",";J;") = ";PL(I,J);
4030  NEXT I: PRINT
4040  NEXT J
4050  RETURN : END
4100  FOR I = I1 TO I2: PRINT "CP$(";I;",0) = ";CP$(I,0): NEXT I: RETURN : END
4200  POKE 34, PEEK (37) - 1: HOME
4210  INPUT "J ARG OF (I,J) OR CTRL-S    = ";J$
4220  IF J$ =  CHR$ (19) THEN  RETURN : END
4230  INPUT "I ARG OF (I,J) & VALUE:I,V = ";I$,V$
4240 I =  VAL (I$):J =  VAL (J$):V =  VAL (V$)
```

```
4250 PL(I,J) = V: GOTO 4210
4260  RETURN : END
4300  POKE 34, PEEK (37) - 1: HOME
4310  INPUT "I ARG OF (I,J) OR CTRL-S   = ";I$
4320  IF I$ =  CHR$ (19) THEN  RETURN : END
4330  INPUT "THEN TYPE STRING ";V$
4340 I =  VAL (I$):CP$(I,0) = V$: GOTO 4310
4350  RETURN : END
4400  REM
4410  INPUT "ROT,WDT,HGT =  ";R,W,HT
4420  INPUT "X,Y   INCH  =  ";X,Y
4430  INPUT "CM$(0)  = ";CM$(0)
4440  RETURN : END
4500  PRINT R$;D$;"OPEN BATHDEF,D1"
4510  PRINT R$;D$;"READ BATHDEF"
4530  FOR J = 1 TO 6: INPUT A(0,J),A(1,J): NEXT J
4540  PRINT R$;D$;"CLOSE BATHDEF"
4550  RETURN : END
4600  PRINT R$;D$;"OPEN BATHDEF,D1"
4610  PRINT R$;D$;"DELETE BATHDEF"
4620  PRINT R$;D$;"OPEN BATHDEF"
4630  PRINT R$;D$;"WRITE BATHDEF"
4650  FOR J = 1 TO 6: PRINT A(0,J);",";A(1,J): NEXT J
4660  PRINT R$;D$;"CLOSE BATHDEF"
4670  RETURN : END
4700  FOR J = J1 TO J2
4710  FOR I = I1 TO I2
4720  PRINT  TAB( 2 + I * 20);"A(";I;",";J;") = ";A(I,J);
4730  NEXT I: PRINT
4740  NEXT J
4750  RETURN : END
4800  POKE 34, PEEK (37) - 1: HOME
4810  INPUT "J ARG OF (I,J) OR CTRL-S  = ";J$
4820  IF J$ =  CHR$ (19) THEN  RETURN : END
4830  INPUT "I ARG OF (I,J) & VALUE:I,V = ";I$,V$
4840 I =  VAL (I$):J =  VAL (J$):V =  VAL (V$)
4850 A(I,J) = V: GOTO 4810
4860  RETURN : END
5000  POKE 34,6: HOME : PRINT " GRID PT-->VLTG CONV": PRINT : VTAB 9
5010  INPUT " WANT TO PRINT OUT? (Y/N) ";AN$: IF  LEFT$ (AN$,1) >  < "Y" THEN  GOTO 5025
5020  PRINT R$;D$;"PR#1": POKE 1401,80:
5025  POKE 34,8: HOME
5030  INPUT " # OF MESHES (RAD/AX)IAL = ";NR,NZ
5040 DR = (A(0,6) - A(0,5)) / NR:DZ = (A(1,6) - A(1,5)) / NZ
5048  PRINT "RI= ";A(0,3); TAB( 21);"AI= ";A(1,3); TAB( 41);"RVTO= ";A(0,4); TAB( 21);"ZVTO= ";A(1,4)
```

```
5050  FOR J = 0 TO NZ
5055  FOR K = 0 TO 39: PRINT "*";: NEXT K: PRINT
5060 ZL = A(1,5) + DZ * J
5070  FOR I = 0 TO NR
5080 RL = A(0,5) + DR * I: GOSUB 5400
5090  PRINT "ZL= ";ZL; TAB( 21);"RL= ";RL; TAB( 41);"ZVT= ";ZVT; TAB( 21);"RVT= ";RVT
5100  NEXT I
5105  PRINT : PRINT "DP= ";DP; TAB( 21);"HT= ";A(1,1)
5108  FOR K = 0 TO 39: PRINT "*";: NEXT K: PRINT : PRINT
5110  NEXT J
5120  PRINT R$;D$;"PR#0"
5130  RETURN : END
5200  REM  VOLTAGE-->LOCATION ROUT
5210 DP = (ZVT - A(1,4)) * 1.3523:ZL = A(1,1) - DP + A(1,3):DP = DP - A(1,3)
5220 RA = 9.08 - RVT * 0.646:RL = RA + A(0,3)
5230  RETURN : END
5300  REM  A(I,J)-->PL(I,J)
5310 PL(0,1) = A(0,1) / 2:PL(1,1) = A(1,1)
5320 PL(0,2) = 0:PL(1,2) = 0
5330 PL(0,3) = A(0,1) / 2:PL(1,3) = A(1,1)
5340 PL(0,4) = 0.5:PL(1,4) = 0.5
5350  RETURN : END
5400  REM  GRID PT-->POTENTIO VLTG
5410 DP = A(1,1) - ZL + A(1,3):ZVT = A(1,4) + DP / 1.3523
5420 RA = RL - A(0,3):RVT = 9.08 - RA / 0.646
5430  RETURN : END
5500  INPUT "WANT TO PRINT OUT ? (Y/N) ";AN$: IF  LEFT$ (AN$,1) >  < "Y" THEN  GOTO 5520
5510  PRINT R$;D$;"PR#1": POKE 1401,80
5520  PRINT "ZL= ";ZL; TAB( 21);"RL= ";RL; TAB( 41);"ZVT= ";ZVT; TAB( 21);"RVT= ";RVT
5530  PRINT "DP= ";DP; TAB( 21);"HT= ";A(1,1)
5540  PRINT R$;D$;"PR#0"
5550  INPUT " KEEP ON ? (Y/N) ";AN$
5560  RETURN : END
6700 C =  PEEK (116):D =  PEEK (115): GOSUB 6900
6710  VTAB 3: PRINT "PRESENT HIMEM="; TAB( 33);A$
6720  VTAB 4: HTAB 32: PRINT "=";D + C * 256:Z1 =  PEEK (37)
6730  VTAB 5: PRINT "WANT TO CHANGE HIMEM Y/N = ": VTAB 5: HTAB 28: INPUT AN$
6740  IF AN$ >  < "Y" THEN  GOTO 6780
6750  VTAB 5: HTAB 1: PRINT "INPUT NEW HIMEM "
6755  VTAB 5: HTAB 17: INVERSE : PRINT "(4-HEX;$XXXX)=": NORMAL : VTAB 5: HTAB 32: INPUT A$
6760  GOSUB 6800:B = C * 256 + D: VTAB 6: HTAB 32: PRINT "=";B
6770  HIMEM: B:Z1 =  PEEK (37)
6780  POKE 34,Z1: HOME : RETURN : END
6800  FOR I = 2 TO 5
6810 A(0,I) =  ASC ( MID$ (A$,I,1))
```

```
6820  IF A(0,I) =  < 57 THEN A(0,I) = A(0,I) - 48: GOTO 6840
6830 A(0,I) = A(0,I) - 55
6840  NEXT I
6850 C = A(0,2) * 16 + A(0,3):D = A(0,4) * 16 + A(0,5)
6860  RETURN : END
6900 A(0,1) =  INT (C / 16):A(0,2) = C - A(0,1) * 16:A(0,3) =  INT (D / 16):A(0,4) = D - A(0,3) * 16:A$ = "$"
6910  FOR I = 1 TO 4
6920  IF A(0,I) < 10 THEN A$ = A$ +  STR$ (A(0,I)): GOTO 6940
6930 A(0,I) = A(0,I) + 55:A$ = A$ +  CHR$ (A(0,I))
6940  NEXT I
6950  RETURN : END
7000  REM    SUBR FRAME (PL(I,J :I=0,1;J=0,1)
7010 B%(0) =  INT (PL(0,0) * UPI) - 1: REM    XREF
7020 B%(1) =  INT (PL(1,0) * UPI) - 1: REM    YREF
7030 B%(2) =  INT ((PL(0,0) + PL(0,1)) * UPI) - 1: REM    XLIM
7040 B%(3) =  INT ((PL(1,0) + PL(1,1)) * UPI) - 1: REM    YLIM
7050  FOR I = 0 TO 3
7060  POKE XREF + I * 2, INT (B%(I) / H): POKE XREF + I * 2 - 1,B%(I) -  INT (B%(I) / H) * H
7070  NEXT I
7080  RETURN : END
7200  REM    SUBR MKTIC (PL(I,J :I=0,1;J=0,1,2,3,4,5,6)
7202  FOR I = 0 TO 1
7204 J = 1 - I
7206  POKE TSFT + J, INT (PL(J,5) * UPI): POKE TLNG + J,H -  INT (PL(J,6) * UPI)
7208  NEXT I
7210  FOR I = 0 TO 1
7220  IF PL(I,2) * PL(I,3) =  > 0 THEN B%(I) =  INT (PL(I,0) * UPI) - 1: GOTO 7240
7230 B%(I) =  INT ((PL(I,0) - PL(I,2) * PL(I,1) / (PL(I,3) - PL(I,2))) * UPI) - 1
7240  POKE XZERO + I * 2, INT (B%(I) / H): POKE XZERO + I * 2 - 1,B%(I) -  INT (B%(I) / H) * H
7250 J = 1 - I
7260 B%(I + 2) =  INT ((PL(J,3) - PL(J,2)) / PL(J,4)): POKE TCNM + J,B%(I + 2)
7270  FOR K = 0 TO B%(I + 2)
7280 CR%(J,K) =  INT ((PL(J,0) + PL(J,1) * PL(J,4) * K /  ABS (PL(J,3) - PL(J,2))) * UPI) - 1
7290  NEXT K
7300  NEXT I
7310 B%(5) =  PEEK (107) +  PEEK (108) * H: REM    PL(1,19) ARRAY ADDR
7320 B%(5) = B%(5) + 209 + 9 - 1: REM    ADDR OF 1ST ELEMENT OF CR% - 1
7330  POKE CRARY, INT (B%(5) / H): POKE CRARY - 1,B%(5) -  INT (B%(5) / H) * H
7340  RETURN : END
7400  REM    SUBR CAPTIC (PL(I,J :I=0,1;J=0,2,3,4,7,8,9,10,11,12),CP$(J,0 :J=0,1))
7410  FOR I = 0 TO 1
7420 J = 1 - I
7430 B%(I + 2) =  INT ((PL(J,3) - PL(J,2)) / PL(J,4) / PL(J,12))
7440  POKE TCNM + J,B%(I + 2): POKE RTT + J,PL(J,7)
7450  POKE WDT + J,PL(J,8)
```

```
7460  POKE HGT + J,PL(J,9)
7470  NEXT I
7480  FOR I = 0 TO 1
7490 J = 1 - I
7500  IF PL(I,2) * PL(I,3) =  > 0 THEN B%(I) =  INT ((PL(I,0) - PL(J,10)) * UPI) - 1: GOTO 7520
7510 B%(I) =  INT ((PL(I,0) - PL(I,2) * PL(I,1) / (PL(I,3) - PL(I,2)) - PL(J,10)) * UPI) - 1
7520  POKE XZERO + I * 2, INT (B%(I) / H): POKE XZERO + I * 2 - 1,B%(I) -  INT (B%(I) / H) * H
7530  FOR K = 0 TO B%(I + 2)
7540 CR%(J,K) =  INT ((PL(J,0) + PL(J,1) * PL(J,4) * PL(J,12) * K /  ABS (PL(J,3) - PL(J,2)) - PL(J,11)) * UPI) - 1
7550  NEXT K
7560  NEXT I
7570 B%(5) =  PEEK (107) +  PEEK (108) * H: REM     PL(1,19) ARRAY ADDR
7580 B%(5) = B%(5) + 209 + 9 - 1: REM     ADDR OF 1ST ELEMENT OF CR% - 1
7590  POKE CRARY, INT (B%(5) / H): POKE CRARY - 1,B%(5) -  INT (B%(5) / H) * H
7600 B%(6) = B%(5) + 169: REM    ADDR OF 1ST ELEMENT OF CP$ -1
7610  POKE CPARY, INT (B%(6) / H): POKE CPARY - 1,B%(6) -  INT (B%(6) / H) * H
7620  RETURN : END
7700  REM    SUBR CURVE/DOTPLT (PL(I,J :I=0,1;J=2,3,13,14,15,16),PL(1,17))
7702  POKE SYMBOL,PL(0,16): POKE RTT,PL(1,15)
7704  POKE WDT,PL(1,16): POKE HGT,PL(1,16)
7710  FOR K = 0 TO PL(0,15) - 1
7720  IF (CV%(0,K) / PL(0,13)) =  > PL(0,2) THEN K1 = K: GOTO 7740
7730  NEXT K
7740  FOR K = PL(0,15) - 1 TO K1 STEP  - 1
7750  IF (CV%(0,K) / PL(0,13)) =  < PL(0,3) THEN K2 = K: GOTO 7770
7760  NEXT K
7770 B%(5) = K2 - K1 + 1
7780  POKE LPNMH, INT (B%(5) / H): POKE LPNMH - 1,B%(5) -  INT (B%(5) / H) * H
7790 B%(0) = 2000:B%(1) = 1400:B%(2) = 1999:B%(3) = 1399
7800  FOR K = K1 TO K2
7810  FOR I = 0 TO 1
7820 CV%(I,K) =  INT ((PL(I,0) + (CV%(I,K) / PL(I,13) - PL(I,2)) * PL(I,1) / (PL(I,3) - PL(I,2))) * UPI) - 1
7830  IF CV%(I,K) =  > B%(I) THEN CV%(I,K) = B%(I + 2)
7840  NEXT I
7850  NEXT K
7860 B%(5) =  PEEK (107) +  PEEK (108) * H: REM       PL(1,19) ARRAY ADDR
7870 B%(5) = B%(5) + 209 + 169 + 15 + 10 + 27 + 67 + 9: REM       ADDR OF 1ST ELEMENT OF CV%
7880 B%(6) = B%(5) + 4 * K1 - 1: REM    ADDR OF K1TH ELEMENT OF CV% -1
7890  POKE CVARY, INT (B%(6) / H): POKE CVARY - 1,B%(6) -  INT (B%(6) / H) * H
7900  RETURN : END
8000  REM       COMMENT (R,W,HT,X,Y,CM$(0))
8010  POKE RTT,R: POKE WDT,W: POKE HGT,HT
8020 B%(0) =  INT (X * UPI) - 1:B%(1) =  INT (Y * UPI) - 1
8030  FOR I = 0 TO 1
8040  POKE XCRD + 2 * I, INT (B%(I) / H): POKE XCRD + 2 * I - 1,B%(I) -  INT (B%(I) / H) * H
```

```
8050  NEXT I
8060 B%(5) =  PEEK (107) +  PEEK (108) * H: REM    PL(1,19) ARRAY ADDR
8070 B%(5) = B%(5) + 209 + 169 + 15 + 7 - 1: REM    ADDR OF 1ST ELEMENT OF CM$-1
8080  POKE CMARY, INT (B%(5) / H): POKE CMARY - 1,B%(5) -  INT (B%(5) / H) * H
8090  RETURN : END
8100  REM  LINDRW (A(I,J:I=0,1;J=11,12),PL(I,J:I=0,1;J=0,1,2,3,14))
8110  FOR K = 11 TO 12
8120  FOR I = 0 TO 1
8130 J = (K - 11) * 2 + I
8140 B%(J) =  INT ((PL(I,0) + (A(I,K) * PL(I,14) - PL(I,2)) * PL(I,1) / (PL(I,3) - PL(I,2))) * UPI) - 1
8150  NEXT I
8160  NEXT K
8170  FOR I = 0 TO 3: POKE XCRD + I * 2, INT (B%(I) / H): POKE XCRD + I * 2 - 1,B%(I) -  INT (B%(I) / H) * H: NEXT I
8180  RETURN : END
```

APPENDIX F

Mainframe Program Listings

```
      FUNCTION ADF(I,J,LX,K)
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1
      COMMON/CDIM/X(5),RE;RM
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC
      COMMON/CCONT/C1,C2,C3,CD
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)
      COMMON/CEMP/WF,P,S(4)
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW
      COMMON/CTRVL/VE,VC
      COMMON/CHEAT/HC,HW(4)
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA
      COMMON/CVF/FES(21),FEM(21),FSM,FSE
      COMMON/CDROP/TAU,HCD,D,QS,QM
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)
     1,CAPPA,E
      COMMON/CFLD/CU
      COMMON/CPI/PI
      COMMON/CTYP/LTYP
      COMMON/CANG/THETA
      COMMON/FITER/MIT,MPRINT,CP
      COMMON/CDVAR/A(31,21,20)
      COMMON/CNAME/ANAME,ASYMBL
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA
      COMMON/CLF/LF,LI
      COMMON/CSVOR/SVOR(31,21)
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)
      COMMON/CAVM/AVM(60)
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR
      COMMON/CSUR/HS(2),TS(2)
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP
      COMMON/CSORSE/SOURCE,SPRIME
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5
      COMMON/CAF/FUN(20)
      COMMON/CVOLT/VOLT
      COMMON/CAVT/TB
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
C   FUNCTION FOR EVALUATION OF FIRST DERIVATIVES
```

```
      M=1
      SIGN=1.
      IF(I.EQ.1) M=3
      IF(I.EQ.IN) M=5
      IF(J.EQ.1) M=2
      IF(J.EQ.JN)  M=4
      IF(J.EQ.J1.AND.I.LE.I1) M=4
      IF(I.EQ.I1.AND.J.GT.J1)   M=3
      GO TO (1,2,3,4,5),M
C*** M=1, FOR POINTS NOT ON ANY OF THE BOUNDARIES
    1 IF(LX.EQ.2)  GO TO 11
      PN=1.
      BENQ=A(I+1,J,K)
      BWSR=A(I-1,J,K)
      BP=A(I,J,K)
      XENQ=X1(I+1)-X1(I)
      XWSR=X1(I)-X1(I-1)
      GO TO 100
   11 PN=1.
      BENQ=A(I,J+1,K)
      BWSR=A(I,J-1,K)
      BP=A(I,J,K)
      XENQ=X2(J+1)-X2(J)
      XWSR=X2(J)-X2(J-1)
      GO TO 100
C *** FOR POINTS ON THE SYMMETRY AXIS
    2 IF(LX.EQ.1)  GO TO 1
      PN=-1.
      BENQ=A(I,J+1,K)
      BWSR=A(I,J+2,K)
      BP=A(I,J,K)
      XENQ=X2(J+1)-X2(J)
      XWSR=X2(J+2)-X2(J)
      GO TO 100
C*** M=3, FOR POINTS ON THE BOUNDARY I=I1
    3 IF(LX.EQ.2)  GO TO 11
      PN=-1.
      BENQ=A(I+1,J,K)
      BWSR=A(I+2,J,K)
      BP=A(I,J,K)
      XENQ=X1(I+1)-X1(I)
      XWSR=X1(I+2)-X1(I)
      GO TO 100
C*** M=4, FOR POINTS ON THE BOUNDARY J=JN
    4 IF(LX.EQ.1)  GO TO 1
```

```
      PN=-1.
      BENQ=A(I,J-1,K)
      BWSR=A(I,J-2,K)
      BP=A(I,J,K)
      XENQ=X2(J)-X2(J-1)
      XWSR=X2(J)-X2(J-2)
      SIGN=-1.
      GO TO 100
C*** M=5, FOR POINTS ON THE BOUNDARY I=IN
    5 IF(LX.EQ.2)  GO TO 11
      PN=-1.
      BENQ=A(I-1,J,K)
      BWSR=A(I-2,J,K)
      BP=A(I,J,K)
      XENQ=X1(I)-X1(I-1)
      XWSR=X1(I)-X1(I-2)
      SIGN=-1.
  100 CONTINUE
      ADF=((XENQ*XENQ-XWSR*XWSR)*BP+
     1XWSR*XWSR*BENQ-XENQ*XENQ*BWSR)/(XENQ*XWSR*(PN*XENQ+XWSR))
      ADF=SIGN*ADF
      RETURN
      END
```

```
      BLOCK DATA                                                        BLK00010
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$BLK00020
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT          BLK00030
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)                BLK00040
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1                  BLK00050
      COMMON/CDIM/X(5),RE,RM                                            BLK00060
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC                        BLK00070
      COMMON/CCONT/C1,C2,C3,CD                                          BLK00080
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)                              BLK00090
      COMMON/CEMP/WF,P,S(4)                                             BLK00100
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS                     BLK00110
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL            BLK00120
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW                                 BLK00130
      COMMON/CTRVL/VE,VC                                                BLK00140
      COMMON/CHEAT/HC,HW(4)                                             BLK00150
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA                       BLK00160
      COMMON/CVF/FES(21),FEM(21),FSM,FSE                                BLK00170
      COMMON/CDROP/TAU,HCD,D,QS,QM                                      BLK00180
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)    BLK00190
     1,CAPPA,E                                                          BLK00200
      COMMON/CFLD/CU                                                    BLK00210
      COMMON/CPI/PI                                                     BLK00220
      COMMON/CTYP/LTYP                                                  BLK00230
      COMMON/CANG/THETA                                                 BLK00240
      COMMON/FITER/MIT,MPRINT,CP                                        BLK00250
      COMMON/CDVAR/A(31,21,20)                                          BLK00260
      COMMON/CNAME/ANAME,ASYMBL                                         BLK00270
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA    BLK00280
      COMMON/CLF/LF,LI                                                  BLK00290
      COMMON/CSVOR/SVOR(31,21)                                          BLK00300
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI                           BLK00310
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)               BLK00320
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)                             BLK00330
      COMMON/CAVM/AVM(60)                                               BLK00340
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR                        BLK00350
      COMMON/CSUR/HS(2),TS(2)                                           BLK00360
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP                                    BLK00370
      COMMON/CSORSE/SOURCE,SPRIME                                       BLK00380
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5                              BLK00390
      COMMON/CAF/FUN(20)                                                BLK00400
      COMMON/CVOLT/VOLT                                                 BLK00410
      COMMON/CAVT/TB                                                    BLK00420
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)                BLK00430
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$BLK00440
      COMMON/CADD/JA,JAX,RA,RAX,IAX                                     BLK00450
```

```
C ............. PROGRAM AND PRINT OUT CONTROL DATA                      BLK00460
C               ----------------------------------                      BLK00470
      DATA NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,IE,IV/3,4,5,6,7,8,9,10,12,6, BLK00480
     16/                                                                BLK00490
C.......................................................................BLK00500
      DATA IP,CC/1,0.005/                                               BLK00510
     2,RP/1.5,1.2,0.2,1.2,0.5,0.75,0.5,3*1.0/                           BLK00520
     3,NTC,NSP,NMT/13,14,15/                                            BLK00530
C.......................................................................BLK00540
      DATA RSDU,SAN/10*0.0,10*0.0/                                      BLK00550
C.....................................................................  BLK00560
      DATA NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS /1,2,16,17,18,19,11,20/   BLK00570
     1,MIT,MPRINT,CP/300,25,0.0001/                                     BLK00580
C.....................................................................  BLK00590
C........................ PHYSICAL DATA                                 BLK00600
C                         -------------                                 BLK00610
      DATA ROREF/5*08.37E+03/                                           BLK00620
     1,ZMUREF/2*2.29E-02/                                               BLK00630
     2,PR/4*1.0,1.000,1.3,4*1.0/                                        BLK00640
     3,TCREF/7.0E-03,5.0E-03,7.0E-03,7.0E-03,7.0E-03/                   BLK00650
     4,SPREF,HL/0.16,0.31,0.16,1.1751,0.16,67.0/                        BLK00660
     5,TLM,TSM,TLS,TA,TM,TW/300.0,300.0,300.0,300.0,300.0,300.0/        BLK00670
     6,ES,EE,EM,SB/0.80,0.40,1.00,13.70/                                BLK00680
     7,BETA/2*0./                                                       BLK00690
     8,VE,VC,HC/0.,0.,5.0E-03/                                          BLK00700
     9,GAMA/0.9/                                                        BLK00710
     1,HW/5.0E-02,5.0E-02,4.5E-02,5.0E-02/                              BLK00720
     2,CAPPA,E/0.40,9.0/                                                BLK00730
     3,PI/3.1416/                                                       BLK00740
      DATA C1,C2,C3,CD/1.44,1.92,1.0,0.09/                              BLK00750
      DATA WF,P/0.0,1.26E-06/                                           BLK00760
     1,S/5.84E+07,0.99E+06,0.99E+06,5.84E+07/                           BLK00770
     2,CU/0.50/                                                         BLK00780
     3,LTYP/2/                                                          BLK00790
C........................................................................BLK00800
C............................... GRID DATA                              BLK00810
C                                ---------                              BLK00820
      DATA IN,JN/21,17/                                                 BLK00830
     1,X/0.01,0.010,0.0130,0.0425,0.050/                                BLK00840
     2,RE,RM,RA,RAX/0.0075,0.075,.0125,.0675/                           BLK00850
     3,I1,I2,IA,IAX,I3,J1,JA,JAX/7,7,10,18,21,5,08,15/                  BLK00860
     4,(IMIN(I),I=1,17)/05*2,12*8/                                      BLK00870
     5,(IMAX(I),I=1,17)/17*20/                                          BLK00880
     6,THETA/0.0000/                                                    BLK00890
      DATA TAU,HCD,D/0.,0.,0./                                          BLK00900
```

```
C***************************    E N D **********************************BLK00910
      END                                                               BLK00920
```

```
      SUBROUTINE BOUND                                                  BOU00010
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$BOU00020
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT          BOU00030
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)                BOU00040
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1                  BOU00050
      COMMON/CDIM/X(5),RE,RM                                            BOU00060
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC                        BOU00070
      COMMON/CCONT/C1,C2,C3,CD                                          BOU00080
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)                              BOU00090
      COMMON/CEMP/WF,P,S(4)                                             BOU00100
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS                     BOU00110
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL            BOU00120
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW                                 BOU00130
      COMMON/CTRVL/VE,VC                                                BOU00140
      COMMON/CHEAT/HC,HW(4)                                             BOU00150
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA                       BOU00160
      COMMON/CVF/FES(21),FEM(21),FSM,FSE                                BOU00170
      COMMON/CDROP/TAU,HCD,D,QS,QM                                      BOU00180
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)    BOU00190
     1,CAPPA,E                                                          BOU00200
      COMMON/CFLD/CU                                                    BOU00210
      COMMON/CPI/PI                                                     BOU00220
      COMMON/CTYP/LTYP                                                  BOU00230
      COMMON/CANG/THETA                                                 BOU00240
      COMMON/FITER/MIT,MPRINT,CP                                        BOU00250
      COMMON/CDVAR/A(31,21,20)                                          BOU00260
      COMMON/CNAME/ANAME,ASYMBL                                         BOU00270
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA    BOU00280
      COMMON/CLF/LF,LI                                                  BOU00290
      COMMON/CSVOR/SVOR(31,21)                                          BOU00300
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI                           BOU00310
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)               BOU00320
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)                             BOU00330
      COMMON/CAVM/AVM(60)                                               BOU00340
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR                        BOU00350
      COMMON/CSUR/HS(2),TS(2)                                           BOU00360
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP                                    BOU00370
      COMMON/CSORSE/SOURCE,SPRIME                                       BOU00380
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5                              BOU00390
      COMMON/CAF/FUN(20)                                                BOU00400
      COMMON/CVOLT/VOLT                                                 BOU00410
      COMMON/CAVT/TB                                                    BOU00420
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)                BOU00430
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$BOU00440
      DIMENSION WW(15)                                                  BOU00450
```

```
              J11=J1-1                                                  BOU00460
C............... AT THE CENTERLINE OF VESSEL                            BOU00470
C                ---------------------------                            BOU00480
      R3SQ =R(3)*R(3)                                                   BOU00490
      R2SQ =R(2)*R(2)                                                   BOU00500
      BB =R3SQ /(R3SQ -R2SQ )                                           BOU00510
      DO 41 I=2,INM                                                     BOU00520
                   IF(I.LT.IW(1).OR.I.GT.I3) GO TO 41                   BOU00530
      A(I,1,NW)=(A(I,1,NF)-A(I,3,NF))/R3SQ+(A(I,2,NF)-A(I,1,NF))/R2SQ   BOU00540
      A(I,1,NW) =8.*A(I,1,NW) / A(I,1,NRO)/(R3SQ-R2SQ)                  BOU00550
                   IF(NTURB.EQ.0) GO TO 41                              BOU00560
      A(I,1,NK)=BB*A(I,2,NK)+(1.-BB)*A(I,3,NK)                          BOU00570
      A(I,1,NEP)=BB*A(I,2,NEP)+(1.-BB)*A(I,3,NEP)                       BOU00580
      IF(A(I,1,NK).LE.0.0) A(I,J,NK)=0.0                                BOU00590
      IF(A(I,J,NEP).LE.0.0) A(I,J,NEP)=0.0                              BOU00600
  41               CONTINUE                                             BOU00610
C---------------------------------------------------------------------- BOU00620
C...................... AT SIDE WALL                                    BOU00630
C                       ------------                                    BOU00640
C...........VORTICITY                                                   BOU00650
C                                                                       BOU00660
              DX2=X2(JN)-X2(JNM)                                        BOU00670
                             DO 42 I=I1,I3                              BOU00680
      Z=A(I,JN,NW)                                                      BOU00690
      A(I,JN,NW)=3.*(A(I,JN,NF)-A(I,JNM,NF))/DX2/DX2/R(JN)/A(I,JN,NRO)  BOU00700
      A(I,JN,NW)=A(I,JN,NW)/R(JNM)-0.5*A(I,JNM,NW)                      BOU00710
   42                        CONTINUE                                   BOU00720
C---------------------------------------------------------------------- BOU00730
C.......................AT FREE SURFACE                                 BOU00740
C                       ---------------                                 BOU00750
              J12=J1+1                                                  BOU00760
                             DO 44 J=J12,JNM                            BOU00770
      A(I1,J,NW)=0.0                                                    BOU00780
   44                        CONTINUE                                   BOU00790
C---------------------------------------------------------------------- BOU00800
C....................AT THE BASE ELECTRODE SURFACE                      BOU00810
C                    -----------------------------                      BOU00820
              DX12=(X1(I3)-X1(I3-1))**2                                 BOU00830
                             DO 43 J=2,JNM                              BOU00840
              RSQ=R(J)*R(J)                                             BOU00850
      A(I3,J,NW)=3.*(A(I3,J,NF)-A(I3-1,J,NF))/(DX12*RSQ*ROREF(2))       BOU00860
     1-0.5*A(I3-1,J,NW)                                                 BOU00870
   43                        CONTINUE                                   BOU00880
C---------------------------------------------------------------------- BOU00890
C....................ON THE TOP ELECTRODE SURFACE                       BOU00900
```

```
C                    ----------------------------
              GFC=COS(THETA)
                             DO 40 J=2,J1
              II=IW(J)
              XS=X2(J-1)/GFC+(X2(J)-X2(J-1))*GFC
              RSQ=(XS*GFC)**2
              DXN=(X1(II+1)-X1(II))*GFC
              DX12=DXN*DXN
      WW(J)=3.*(0.-A(II+1,J,NF))/DX12/RSQ/A(II,J,NRO)
      WW(J)=WW(J)-0.5*A(II+1,J,NW)
  40                         CONTINUE
                             DO 48 J=2,J1
              II=IW(J)
              GF=SIN(THETA)/COS(THETA)
              DWF=GF/(GF+(X2(J+1)-X2(J))/(X1(II+1)-X1(II)))
      A(II,J,NW)=WW(J)
  48                         CONTINUE
C-----------------------------------------------------------------------
C .......AT SIDE SURFACE OF TOP ELECTRODE(J=J1, I1<I<I2)
C         ------------------------------------------
C        (EXIT IF ELECTRODE JUST TOUCHES THE METAL SURFACE)
                             IF(I1.EQ.I2) GO TO 47
              DX2=X2(J1+1)-X2(J1)
              DX22=DX2*DX2
              I21=I2-1
              RSQ=R(J1)*R(J1+1)
              XB=X2(J1)-X2(J11)
                             DO 45 I=I1,I21
      Z=A(I,J1,NW)
      DT=A(I,J1,NT)-A(I,J1+1,NT)
      A(I,J1,NW)=3.*(A(I,J1,NF)-A(I,J1+1,NF))/DX22/RSQ/A(I,J1,NRO)
      A(I,J1,NW)=A(I,J1,NW)-0.5*A(I,J1+1,NW)
      A(I,J1,NW)=Z+RP(NW)*(A(I,J1,NW)-Z)
      TCM=TCREF(1)
  45                         CONTINUE
  47                         CONTINUE
C-----------------------------------------------------------------------
      RETURN
      END
```

```
      SUBROUTINE CONLOT(XI,PERFIN,PCASC)
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1
      COMMON/CDIM/X(5),RE,RM
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC
      COMMON/CCONT/C1,C2,C3,CD
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)
      COMMON/CEMP/WF,P,S(4)
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW
      COMMON/CTRVL/VE,VC
      COMMON/CHEAT/HC,HW(4)
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA
      COMMON/CVF/FES(21),FEM(21),FSM,FSE
      COMMON/CDROP/TAU,HCD,D,QS,QM
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)
     1,CAPPA,E
      COMMON/CFLD/CU
      COMMON/CPI/PI
      COMMON/CTYP/LTYP
      COMMON/CANG/THETA
      COMMON/FITER/MIT,MPRINT,CP
      COMMON/CDVAR/A(31,21,20)
      COMMON/CNAME/ANAME,ASYMBL
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA
      COMMON/CLF/LF,LI
      COMMON/CSVOR/SVOR(31,21)
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)
      COMMON/CAVM/AVM(60)
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR
      COMMON/CSUR/HS(2),TS(2)
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP
      COMMON/CSORSE/SOURCE,SPRIME
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5
      COMMON/CAF/FUN(20)
      COMMON/CVOLT/VOLT
      COMMON/CAVT/TB
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
      COMMON/CPROD/GK(31,21)
      COMMON/CSHEAR/GKK(31,31)
      DATA DIMNSN,XISET,ETA/0.04,50.0,0.50/
```

```
C______________________________________________________________________
C       ROUTINE TO CALCULATE LOTSIANSKII CONSTANT,XI                  |
C       OUTPUT FILE ON LOGICAL UNIT # 17
C_____________________________________________________________________|
C
      IH=I3-1
      IL=I1+1
C... FIND THE POWER FLOW:
C   ----------------------
            PCASC=0.0
                             DO 60 J=2,JNM
              DR=R(J+1)-R(J-1)
              IL1=IL+1
                             DO 60 I=IL,IH
              DX=X1(I+1)-X1(I-1)
              DV=R(J)*DR*DX/4.0
              PCASC=PCASC-P*A(I,J,NV2)*R(J)*A(I,J,NHR)**2*1E6*DV
 60                          CONTINUE
C-----------------------------------------------------------------
C             COMPUTE POWER LOST AS VISCOUS DISSIPATION AT WALL
C             ------------------------------------------------
         PWALL = 0.0
         CONST = 11.5/(ROREF(1)**0.5)
C TOP ELECTRODE FACE:
C -----------------
                   DO 350 J=2,J1
               DR= (R(J+1)-R(J-1))/2.0
               DAREA= DR*R(J)
               PWALL = PWALL +(ABS(TAUW2(J))**1.5)*DAREA*CONST
         YPLUS = SQRT(TAUW2(J)*ROREF(1))*(X1(IL)-X1(I1))/ZMUREF(1)
         WRITE(25,999) YPLUS
 350               CONTINUE
C BOTTOM ELECTRODE FACE:
C --------------------
                   DO 360 J=2,JNM
               DR= (R(J+1)-R(J-1))/2.0
               DAREA= DR*R(J)
               PWALL = PWALL + (ABS(TAUW3(J))**1.5)*DAREA*CONST
         YPLUS = SQRT(TAUW3(J)*ROREF(1))*(X1(IN)-X1(IH))/ZMUREF(1)
         WRITE(25,999) YPLUS
360                CONTINUE
C CYLINDRICAL WALL:
C ----------------
                   DO 370 I=IL,IH
               N=I-IL+1
```

```
               DX= (X1(I+1)-X1(I-1))/2.0
               DAREA = RM*DX
               PWALL = PWALL + (ABS(TAUW1(N))**1.5)*DAREA*CONST
         YPLUS = SQRT(TAUW1(N)*ROREF(1))*(R(JN)-R(JNM))/ZMUREF(1)
         WRITE(25,999) YPLUS
999      FORMAT(' YPLUS = ',1PE10.2)
370                 CONTINUE
C             EFFICIENCY OF ENERGY CONVERSION
C             -------------------------------
               ETA2 = (PCASC-PWALL)/PCASC
               WRITE(6,400) ETA2,PWALL,PCASC
 400           FORMAT('COMPUTED EFFICIENCY = ',F5.2
     1/'VISCOUS DISSIPATION = ',1PE10.2
     2/'TOTAL POWER FLOW    = ',1PE10.2)
C-----------------------------------------------------------------------
              PINPT=ETA*PCASC
              CALL VOL(GKK,SHEAR2)
         IF(SHEAR2.EQ.0.0) GO TO 100
              AVGVIS=PINPT/(SHEAR2*ROREF(1))
              AI     =AVGVIS**2*DIMNSN**3
              XI     =1.0/(AI**0.118)
         PERFIN     =AVGVIS*ROREF(1)/ZMUREF(1)
C--------------------------------------------------------------------
         RETURN
 100     XI=XISET
         RETURN
         END
```

```
      SUBROUTINE CONVEC(AE,AW,AN,AS,I,J,K)                              CON00010
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$  CON00020
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT          CON00030
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)                CON00040
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1                  CON00050
      COMMON/CDIM/X(5),RE,RM                                            CON00060
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC                        CON00070
      COMMON/CCONT/C1,C2,C3,CD                                          CON00080
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)                              CON00090
      COMMON/CEMP/WF,P,S(4)                                             CON00100
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS                     CON00110
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL            CON00120
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW                                 CON00130
      COMMON/CTRVL/VE,VC                                                CON00140
      COMMON/CHEAT/HC,HW(4)                                             CON00150
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA                       CON00160
      COMMON/CVF/FES(21),FEM(21),FSM,FSE                                CON00170
      COMMON/CDROP/TAU,HCD,D,QS,QM                                      CON00180
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)    CON00190
     1,CAPPA,E                                                          CON00200
      COMMON/CFLD/CU                                                    CON00210
      COMMON/CPI/PI                                                     CON00220
      COMMON/CTYP/LTYP                                                  CON00230
      COMMON/CANG/THETA                                                 CON00240
      COMMON/FITER/MIT,MPRINT,CP                                        CON00250
      COMMON/CDVAR/A(31,21,20)                                          CON00260
      COMMON/CNAME/ANAME,ASYMBL                                         CON00270
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA    CON00280
      COMMON/CLF/LF,LI                                                  CON00290
      COMMON/CSVOR/SVOR(31,21)                                          CON00300
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI                           CON00310
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)               CON00320
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)                             CON00330
      COMMON/CAVM/AVM(60)                                               CON00340
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR                        CON00350
      COMMON/CSUR/HS(2),TS(2)                                           CON00360
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP                                    CON00370
      COMMON/CSORSE/SOURCE,SPRIME                                       CON00380
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5                              CON00390
      COMMON/CAF/FUN(20)                                                CON00400
      COMMON/CVOLT/VOLT                                                 CON00410
      COMMON/CAVT/TB                                                    CON00420
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)                CON00430
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$  CON00440
      COMMON/CPROD/GK(31,21)                                            CON00450
```

```
      COMMON/CSHEAR/GKK(31,31)                                          CON00460
      COMMON/CLREYN/LREYN,CDMOD(31,31)                                  CON00470
C............ROUTINE FOR CALCULATION OF AE,AW,AN,AS                     CON00480
C            --------------------------------------                     CON00490
C            --------------------------------------                     CON00500
C                                                                       CON00510
C... CALCULATION MEAN MASS FLOW RATE AT FOUR TUBES OF THE TANK          CON00520
C    ---------------------------------------------------------          CON00530
      DV=R(J)*(X1(I+1)-X1(I-1))*(X2(J+1)-X2(J-1))                       CON00540
      G1PW=(A(I,J+1,NF)-A(I,J-1,NF)+A(I-1,J+1,NF)-A(I-1,J-1,NF))/DV     CON00550
      G1PE=(A(I,J+1,NF)-A(I,J-1,NF)+A(I+1,J+1,NF)-A(I+1,J-1,NF))/DV     CON00560
      G2PS=(A(I-1,J,NF)-A(I+1,J,NF)+A(I-1,J-1,NF)-A(I+1,J-1,NF))/DV     CON00570
      G2PN=(A(I-1,J,NF)-A(I+1,J,NF)+A(I-1,J+1,NF)-A(I+1,J+1,NF))/DV     CON00580
C-------------------------------------------------------------------    CON00590
C....................ACCOUNT FOR THE WALL FUNCTION                      CON00600
C                    ----------------------------                       CON00610
                   IF(K.EQ.NHR.OR.K.EQ.NHI) GO TO 10                    CON00620
                   IF(K.EQ.NW) GO TO 10                                 CON00630
                   IF(J.NE.JNM) GO TO 7                                 CON00640
C           CYLINDRICAL WALL NEIGHBOR POINTS                            CON00650
C           --------------------------------                            CON00660
      G1PE=-(A(I,J-1,NF)+A(I,J,NF)+A(I+1,J,NF)+A(I+1,J-1,NF))/DV        CON00670
      G1PW=-(A(I,J-1,NF)+A(I,J,NF)+A(I-1,J,NF)+A(I-1,J-1,NF))/DV        CON00680
      BBE=BBE*FR1                                                       CON00690
      BBW=BBW*FR1                                                       CON00700
      BBN=0.0                                                           CON00710
      G2PN=0.                                                           CON00720
      TAUW=TAUW1(I-I1)                                                  CON00730
                   IF(K.NE.NK.AND.K.NE.NEP) GO TO 16                    CON00740
              DX=X2(JN)-X2(JNM-1)                                       CON00750
      UA=ABS(A(I,JNM,NV1)+A(I,JNM-1,NV1))/DX                            CON00760
      YP=X2(JN)-X2(JNM)                                                 CON00770
                   IF(K.EQ.NK) GO TO 20                                 CON00780
                   IF(K.EQ.NEP) GO TO 21                                CON00790
C-------------------------------------------------------------------    CON00800
  16  B1=FR1                                                            CON00810
      B2=2.*R(J+1)/R(J)/(X2(J+1)-X2(J-1))                               CON00820
                   IF(K.NE.NT) GO TO 160                                CON00830
      SNW=SN1(I-I1)                                                     CON00840
      TWALL=A(I,J+1,NT)                                                 CON00850
                   GO TO 22                                             CON00860
  160 CONTINUE                                                          CON00870
                   GO TO 24                                             CON00880
C--------------------------------------------------------------------   CON00890
C            LOWER SURFACE OF TOP ELECTRODE                             CON00900
```

```
C            ------------------------------                             CON00910
  7               IF(I.NE.(IW(J)+1)) GO TO 8                            CON00920
                  IF(J.GT.J1) GO TO 8                                   CON00930
      GF=SIN(THETA)/COS(THETA)                                          CON00940
      G2PN=-(A(I,J+1,NF)+A(I,J,NF)+A(I+1,J,NF)+A(I+1,J+1,NF))/DV        CON00950
      G2PS=-(A(I,J-1,NF)+A(I,J,NF)+A(I+1,J,NF)+A(I+1,J-1,NF))/DV        CON00960
      G1PW=0.                                                           CON00970
      FRN=FR2(J)+(X2(J+1)-X2(J))/(X1(I+1)-X1(I))*GF                     CON00980
      FRS=FR2(J)-(X2(J)-X2(J-1))/(X1(I+1)-X1(I))*GF                     CON00990
      BBN=BBN*FRN                                                       CON01000
      BBS=BBS*FRS                                                       CON01010
      BBW=0.0                                                           CON01020
      TAUW=TAUW2(J)                                                     CON01030
                  IF(K.NE.NK.AND.K.NE.NEP) GO TO 18                     CON01040
              DX=X1(I+1)-X1(I-1)                                        CON01050
      UA1=A(I,J,NV2)-A(I,J,NV1)*GF                                      CON01060
      UA2=A(I+1,J,NV2)-A(I,J,NV1)*GF                                    CON01070
      UA=ABS(UA1+UA2)/DX                                                CON01080
      YP=(X1(I)-X1(I-1))*COS(THETA)                                     CON01090
                   IF(K.EQ.NK) GO TO 20                                 CON01100
                   IF(K.EQ.NEP) GO TO 21                                CON01110
C---------------------------------------------------------------------- CON01120
  18  B1=FR2(J)                                                         CON01130
      B2=2./(X1(I+1)-X1(I-1))                                           CON01140
      IF(K.NE.NT) GO TO 180                                             CON01150
      SNW=SN2(J)                                                        CON01160
      TWALL=A(I-1,J,NT)                                                 CON01170
                   GO TO 22                                             CON01180
  180              CONTINUE                                             CON01190
                   GO TO 24                                             CON01200
C---------------------------------------------------------------------- CON01210
C                BOTTOM ELECTRODE SURFACE                               CON01220
C                ------------------------                               CON01230
  8                IF(I.NE.(I3-1)) GO TO 9                              CON01240
      G2PN= (A(I,J+1,NF)+A(I,J,NF)+A(I-1,J,NF)+A(I-1,J+1,NF))/DV        CON01250
      G2PS= (A(I,J-1,NF)+A(I,J,NF)+A(I-1,J,NF)+A(I-1,J-1,NF))/DV        CON01260
      G1PE=0.                                                           CON01270
      BBE=0.0                                                           CON01280
      BBN=BBN*FR3                                                       CON01290
      BBS=BBS*FR3                                                       CON01300
      TAUW=TAUW3(J)                                                     CON01310
                   IF(K.NE.NK.AND.K.NE.NEP) GO TO 26                    CON01320
              DX=X1(I+1)-X1(I-1)                                        CON01330
      UA=ABS(A(I3-1,J,NV2)+A(I3-2,J,NV2))/DX                            CON01340
      YP=X1(I+1)-X1(I)                                                  CON01350
```

```
                  IF(K.EQ.NK) GO TO 20
                  IF(K.EQ.NEP) GO TO 21
C-------------------------------------------------------------------
  26  B1=FR3
      B2=2./(X1(I+1)-X1(I-1))
                  IF(K.NE.NT) GO TO 261
      SNW=SN3(J)
      TWALL=A(I+1,J,NT)
                  GO TO 22
  261             CONTINUE
                  GO TO 24
C--------------------------------------------------------------------
C              SIDE SURFACE OF TOP ELECTRODE
C              ----------------------------
  9               IF(J.NE.(J1+1)) GO TO 10
                  IF(I.GT.I2) GO TO 10
      G1PE= (A(I,J+1,NF)+A(I,J,NF)+A(I+1,J,NF)+A(I+1,J+1,NF))/DV
      G1PW= (A(I,J+1,NF)+A(I,J,NF)+A(I-1,J,NF)+A(I-1,J+1,NF))/DV
      G2PS=0.
      BBS=0.
      BBE=BBE*FR4
      BBW=BBW*FR4
      TAUW=TAUW4(I-I1)
                  IF(K.NE.NK.AND.K.NE.NEP) GO TO 28
              DX=X2(J1+2)-X2(J1)
      UA=ABS(A(I,J1+1,NV1)+A(I,J1+2,NV1))/DX
      YP=X2(J1+1)-X2(J1)
                  IF(K.EQ.NK) GO TO 20
                  IF(K.EQ.NEP) GO TO 21
C--------------------------------------------------------------------
  28  B1=FR4
      B2=2.*R(J-1)/R(J)/(X2(J+1)-X2(J-1))
                  IF(K.NE.NT) GO TO 280
      SNW=SN4(I-I1)
      TWALL=A(I,J-1,NT)
                  GO TO 22
  280             CONTINUE
                  GO TO 24
C=====================================================================
  20  SOURCE=UA*TAUW
      SPRIME=UA*CDMOD(I,J)*A(I,J,NRO)**2*A(I,J,NK)/TAUW
C     SPRIME=A(I,J,NEP)*A(I,J,NRO)/A(I,J,NK)
                  GK(I,J)=SOURCE
                  GKK(I,J)=UA**2
                  GO TO 10
```

```
  21  SOURCE      =(CDMOD(I,J)**0.75)*(A(I,J,NK)**1.50)/CAPPA/YP        CON01810
C 21  SOURCE      =UA*CDMOD(I,J)*A(I,J,NRO)*A(I,J,NK)**2/TAUW           CON01820
      SPRIME=1.0                                                        CON01830
      AE=0.                                                             CON01840
      AW=0.                                                             CON01850
      AN=0.                                                             CON01860
      AS=0.                                                             CON01870
      BBE=0.                                                            CON01880
      BBW=0.                                                            CON01890
      BBN=0.                                                            CON01900
      BBS=0.                                                            CON01910
                   IF(NSCHEM.EQ.1) A(I,J,NEP)=SOURCE                    CON01920
                   GO TO 100                                            CON01930
  22  SOURCE=SOURCE*B1+B2*SNW*TWALL-B3                                  CON01940
      SPRIME=SNW*B2                                                     CON01950
                   GO TO 10                                             CON01960
  24  SOURCE=SOURCE*B1                                                  CON01970
  10               CONTINUE                                             CON01980
C---------------------------------------------------------------------- CON01990
C            COMPUTE THE CONVECTION CONSTANTS                           CON02000
C            --------------------------------                           CON02010
      APP=1.                                                            CON02020
      IF(K.EQ.NHR.OR.K.EQ.NHI) APP=P/A(I,J,NRO)                         CON02030
      IF(K.EQ.NF) APP=0.                                                CON02040
      IF(K.EQ.NW) APP=R(J)*R(J)                                         CON02050
      IF(K.EQ.NT) APP=A(I,J,NSP)                                        CON02060
      AW=0.5*APP*(ABS(G1PW)+G1PW)                                       CON02070
      AE=0.5*APP*(ABS(G1PE)-G1PE)                                       CON02080
      AN=0.5*APP*(ABS(G2PN)-G2PN)                                       CON02090
      AS=0.5*APP*(ABS(G2PS)+G2PS)                                       CON02100
  100              CONTINUE                                             CON02110
C################     END   ##########################################  CON02120
      RETURN                                                            CON02130
      END                                                               CON02140
```

```
      SUBROUTINE CORD                                                   COR00010
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$COR00020
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT          COR00030
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)                COR00040
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1                  COR00050
      COMMON/CDIM/X(5),RE,RM                                            COR00060
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC                        COR00070
      COMMON/CCONT/C1,C2,C3,CD                                          COR00080
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)                              COR00090
      COMMON/CEMP/WF,P,S(4)                                             COR00100
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS                     COR00110
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL            COR00120
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW                                 COR00130
      COMMON/CTRVL/VE,VC                                                COR00140
      COMMON/CHEAT/HC,HW(4)                                             COR00150
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA                       COR00160
      COMMON/CVF/FES(21),FEM(21),FSM,FSE                                COR00170
      COMMON/CDROP/TAU,HCD,D,QS,QM                                      COR00180
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)    COR00190
     1,CAPPA,E                                                          COR00200
      COMMON/CFLD/CU                                                    COR00210
      COMMON/CPI/PI                                                     COR00220
      COMMON/CTYP/LTYP                                                  COR00230
      COMMON/CANG/THETA                                                 COR00240
      COMMON/FITER/MIT,MPRINT,CP                                        COR00250
      COMMON/CDVAR/A(31,21,20)                                          COR00260
      COMMON/CNAME/ANAME,ASYMBL                                         COR00270
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA    COR00280
      COMMON/CLF/LF,LI                                                  COR00290
      COMMON/CSVOR/SVOR(31,21)                                          COR00300
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI                           COR00310
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)               COR00320
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)                             COR00330
      COMMON/CAVM/AVM(60)                                               COR00340
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR                        COR00350
      COMMON/CSUR/HS(2),TS(2)                                           COR00360
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP                                    COR00370
      COMMON/CSORSE/SOURCE,SPRIME                                       COR00380
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5                              COR00390
      COMMON/CAF/FUN(20)                                                COR00400
      COMMON/CVOLT/VOLT                                                 COR00410
      COMMON/CAVT/TB                                                    COR00420
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)                COR00430
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$COR00440
      COMMON/CADD/JA,JAX,RA,RAX,IAX                                     COR00450
```

```
C                  GRID GENERATION SECTION                              COR00460
C                  -----------------------                              COR00470
C                  -----------------------                              COR00480
      X1(1)=0.0                                                         COR00490
      X2(1)=0.0                                                         COR00500
      R(1)=X2(1)                                                        COR00510
                   IM=I2+J1-1                                           COR00520
                   IF(THETA.EQ.0.) IM=I2                                COR00530
                   IW(J1)=I2                                            COR00540
                   IW(1)=IM                                             COR00550
                   GF=SIN(THETA)/COS(THETA)                             COR00560
                   XIM=X(2)+RE*GF                                       COR00570
                             DO 11 J=2,J1                               COR00580
      IW(J)=IW(J-1)-1                                                   COR00590
      IF(J.GE.J1) IW(J)=I2                                              COR00600
      IF(THETA.EQ.0.) IW(J)=I2                                          COR00610
  11                        CONTINUE                                    COR00620
      DZ1=X(1)/FLOAT(I1-1)                                              COR00630
      DZ3=(X(3)-X(2))/FLOAT(IA-I2)                                      COR00640
      DZ4=(X(4)-X(3))/FLOAT(IAX-IA)                                     COR00650
      DZ5=(X(5)-X(4))/FLOAT(IN-IAX)                                     COR00660
      DO 10 I=2,IN                                                      COR00670
      IF(I.LE.I1) DX1=DZ1                                               COR00680
      IF(I.GT.I1.AND.I.LE.I2) DX1=DZ2                                   COR00690
      IF(I.GT.I2.AND.I.LE.IA) DX1=DZ3                                   COR00700
      IF(I.GT.IA.AND.I.LE.IAX) DX1=DZ4                                  COR00710
      IF(I.GT.IAX.AND.I.LE.IN) DX1=DZ5                                  COR00720
  10  X1(I)=X1(I-1)+DX1                                                 COR00730
              DR1=RE/FLOAT(J1-1)                                        COR00740
              DR2=(RA-RE)/FLOAT(JA-J1)                                  COR00750
              DR3=(RAX-RA)/FLOAT(JAX-JA)                                COR00760
              DR4=(RM-RAX)/FLOAT(JN-JAX)                                COR00770
                             DO 50 J=2,JN                               COR00780
              DX2=DR1                                                   COR00790
              IF(J.GT.J1.AND.J.LE.JA) DX2=DR2                           COR00800
              IF(J.GT.JA.AND.J.LE.JAX)DX2=DR3                           COR00810
              IF(J.GT.JAX.AND.J.LE.JN)DX2=DR4                           COR00820
              X2(J)=X2(J-1)+DX2                                         COR00830
              R(J)=X2(J)                                                COR00840
  50                         CONTINUE                                   COR00850
C---------------------------------------------------------------------- COR00860
C             CALCULATE GRID BASED CONSTANTS                            COR00870
C             ------------------------------                            COR00880
      DO 20 I=2,INM                                                     COR00890
      DX1=1./(X1(I+1)-X1(I-1))                                          COR00900
```

```
      BW(I)=DX1/(X1(I)-X1(I-1))
      BE(I)=DX1/(X1(I+1)-X1(I))
   20 CONTINUE
      DO 21 J=2,JNM
      DX2=1.0/(X2(J+1)-X2(J-1))
      A1=DX2/(X2(J+1)-X2(J))
      BN(J)=(1.+R(J+1)/R(J))*0.5*A1
      A2=DX2/(X2(J)-X2(J-1))
      BS(J)=(1.+R(J-1)/R(J))*0.5*A2
   21 CONTINUE
   23 CONTINUE
      FR1=2.-(X2(JNM)-X2(JNM-1))/(X2(JN)-X2(JNM-1))
      FR3=2.-(X1(I3-1)-X1(I3-2))/(X1(I3)-X1(I3-2))
      FR4=2.-(X2(J1+2)-X2(J1+1))/(X2(J1+2 )-X2(J1))
      FR5=2.-(X1(I1+2)-X1(I1+1))/(X1(I1+2)-X1(I1))
      DO 25 J=1,J1
      II=IW(J)
   25 FR2(J)=2.-(X1(II+2)-X1(II+1))/(X1(II+2)-X1(II))
      RETURN
      END
```

```
      SUBROUTINE DISSIP
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1
      COMMON/CDIM/X(5),RE,RM
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC
      COMMON/CCONT/C1,C2,C3,CD
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)
      COMMON/CEMP/WF,P,S(4)
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW
      COMMON/CTRVL/VE,VC
      COMMON/CHEAT/HC,HW(4)
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA
      COMMON/CVF/FES(21),FEM(21),FSM,FSE
      COMMON/CDROP/TAU,HCD,D,QS,QM
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)
     1,CAPPA,E
      COMMON/CFLD/CU
      COMMON/CPI/PI
      COMMON/CANG/THETA
      COMMON/FITER/MIT,MPRINT,CP
      COMMON/CDVAR/A(31,21,20)
      COMMON/CNAME/ANAME,ASYMBL
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA
      COMMON/CLF/LF,LI
      COMMON/CTYP/LTYP
      COMMON/CSVOR/SVOR(31,21)
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI
      COMMON/CSNW/SN1(20),SN2(15)
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)
      COMMON/CAVM/AVM(60)
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR
      COMMON/CSUR/HS(2),TS(2)
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP
      COMMON/CSORSE/SOURCE,SPRIME
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5
      COMMON/CAF/FUN(20)
      COMMON/CVOLT/VOLT
      COMMON/CAVT/TB
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
      DATA IVAR/0/
```

```
C................CALCULATE DISSIPATION ENERGY
C                ----------------------------
                   DO 31 J=1,JNM-1
      IL=I1+1
      ITOP=IL
      IF(J.LE.J1) ITOP=IL+1
      IH=I3-1
                   DO  31  I=ITOP,IH-1
                             IF(A(I,J,NK).LE.0.0) GO TO 41
      A(I,J,NEP)=(A(I,J,NK)*XI)**1.7
                   GO TO 31
41                           A(I,J,NK)=0.0
                             A(I,J,NEP)=0.0
31                 CONTINUE
C             GET DISSIPATION AT THE BOUNDARIES
C             ---------------------------------
                   DO 50 J=2,JNM
        IL=I1+1
        IH=I3-1
                   DO 50 I=IL,IH
C           TOP SURFACE
C           -----------
            IF(I.EQ.IL.AND.J.LE.J1) CALL CONVEC(D1,D2,D3,D4,I,J,NEP)
C          BOTTOM SURFACE
C          --------------
                   IF(I.EQ.IH) CALL CONVEC(D1,D2,D3,D4,I,J,NEP)
C          CYLINDRICAL SURFACE
C          -------------------
                   IF(J.EQ.JNM) CALL CONVEC(D1,D2,D3,D4,I,J,NEP)
 50                CONTINUE
      RETURN
      END
```

```
C____________________________________________________________________    EMF00010
C         E L E C T R O M A G N E T I C A L L Y      D R I V E N         EMF00020
C                     F L O W    S I M U L A T I O N                     EMF00030
C____________________________________________________________________    EMF00040
      REAL*8 ANAME(6,20),ASYMBL(14)                                      EMF00050
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$    EMF00060
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT           EMF00070
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)                 EMF00080
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1                   EMF00090
      COMMON/CDIM/X(5),RE,RM                                             EMF00100
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC                         EMF00110
      COMMON/CCONT/C1,C2,C3,CD                                           EMF00120
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)                               EMF00130
      COMMON/CEMP/WF,P,S(4)                                              EMF00140
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS                      EMF00150
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL             EMF00160
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW                                  EMF00170
      COMMON/CTRVL/VE,VC                                                 EMF00180
      COMMON/CHEAT/HC,HW(4)                                              EMF00190
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA                        EMF00200
      COMMON/CVF/FES(21),FEM(21),FSM,FSE                                 EMF00210
      COMMON/CDROP/TAU,HCD,D,QS,QM                                       EMF00220
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)     EMF00230
     1,CAPPA,E                                                           EMF00240
      COMMON/CFLD/CU                                                     EMF00250
      COMMON/CPI/PI                                                      EMF00260
      COMMON/CTYP/LTYP                                                   EMF00270
      COMMON/CANG/THETA                                                  EMF00280
      COMMON/FITER/MIT,MPRINT,CP                                         EMF00290
      COMMON/CDVAR/A(31,21,20)                                           EMF00300
      COMMON/CNAME/ANAME,ASYMBL                                          EMF00310
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA     EMF00320
      COMMON/CLF/LF,LI                                                   EMF00330
      COMMON/CSVOR/SVOR(31,21)                                           EMF00340
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI                            EMF00350
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)                EMF00360
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)                              EMF00370
      COMMON/CAVM/AVM(60)                                                EMF00380
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR                         EMF00390
      COMMON/CSUR/HS(2),TS(2)                                            EMF00400
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP                                     EMF00410
      COMMON/CSORSE/SOURCE,SPRIME                                        EMF00420
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5                               EMF00430
      COMMON/CAF/FUN(20)                                                 EMF00440
      COMMON/CVOLT/VOLT                                                  EMF00450
```

```
      COMMON/CAVT/TB                                                    EMF00460
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)                EMF00470
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$ EMF00480
      COMMON/CPROD/GK(31,21)                                            EMF00490
      COMMON/CLREYN/LREYN,CDMOD(31,31)                                  EMF00500
      DIMENSION GKK(31,21),P1(31,21),P2(31,21),RFORCE(31,21)            EMF00510
     1,ZFORCE(31,21)                                                    EMF00520
      DIMENSION XM(31),EPSIL(31,21),TIMIX(31,21)                        EMF00530
      DATA PIMIN,ETA/10.0,0.50/                                         EMF00540
C---------------------------------------------------------------------- EMF00550
C               INITIAL SET UP SECTION                                  EMF00560
C---------------------------------------------------------------------- EMF00570
C.... IF INREAD=1 INITIAL GUESS FIELD IS READ IN ON UNITS 4 AND 8       EMF00580
      INM=IN-1                                                          EMF00590
      JNM=JN-1                                                          EMF00600
C*** READ ALPHAMERIC INFORMATION FOR HEADINGS AND TITLES                EMF00610
      READ(5,200)ANAME,ASYMBL                                           EMF00620
C             READ IN INREAD,NMIN,NSCHEM,RPXI,NTURB                     EMF00630
              READ(5,205)INREAD,NMIN,NSCHEM,NMAX,NPRINT,RPXI,NTURB      EMF00640
     1,NDATA,NFIELD,NFLOW,LREYN,NCHECK                                  EMF00650
C---------------------------------------------------------------------- EMF00660
C                  DATA ECHO SEGMENT (IF NDATA=1)                       EMF00670
C                  ------------------------------                       EMF00680
                   IF(NDATA.EQ.0) GO TO 345                             EMF00690
          WRITE(6,208)                                                  EMF00700
          WRITE(6,210) CU,ZMUREF(2),ROREF(2),S(2),NMAX                  EMF00710
                   DP=X(5)-X(1)                                         EMF00720
                   XIM=X(2)-X(1)                                        EMF00730
         WRITE(6,220) RE,RM,DP,XIM                                      EMF00740
C           CALCULATION SCHEMES IN OPERATION                            EMF00750
C           --------------------------------                            EMF00760
         IF(INREAD.EQ.0) WRITE(6,213)                                   EMF00770
         IF(INREAD.EQ.1) WRITE(6,214)                                   EMF00780
         IF(LTYP.EQ.1)WRITE(6,215)                                      EMF00790
         IF(LTYP.EQ.2)WRITE(6,216)                                      EMF00800
         IF(NTURB.EQ.0) WRITE(6,217)                                    EMF00810
         IF(NSCHEM.EQ.0.AND.LREYN.EQ.1) WRITE(6,235)                    EMF00820
         IF(NTURB.EQ.1) WRITE(6,218)                                    EMF00830
                        WRITE(6,236) CD,C1,C2                           EMF00840
         IF(NSCHEM.EQ.0) WRITE(6,229)                                   EMF00850
         IF(NSCHEM.EQ.1) WRITE(6,230)                                   EMF00860
         IF(NFIELD.EQ.0) WRITE(6,231)                                   EMF00870
         IF(NFIELD.EQ.1) WRITE(6,232)                                   EMF00880
         IF(NFLOW.EQ.0) WRITE(6,233)                                    EMF00890
         IF(NFLOW.EQ.1) WRITE(6,234)                                    EMF00900
```

```
C                                                                       EMF00910
C........   COMPUTE AND PRINT COORDINATES                               EMF00920
C---------------------------------------------------------------------  EMF00930
345                          CONTINUE                                   EMF00940
      CALL  CORD                                                        EMF00950
      WRITE(6,9)                                                        EMF00960
      WRITE(6,12) I1,I2,I3,IA,IN,J1,JN                                  EMF00970
      WRITE(6,101)  (X1(I),I=1,IN)                                      EMF00980
      WRITE(6,102)  (X2(J),J=1,JN)                                      EMF00990
      WRITE(6,40)                                                       EMF01000
  350 CONTINUE                                                          EMF01010
C..........        INITIALIZE                                          EMF01020
C                  ----------                                          EMF01030
      CALL INIT                                                         EMF01040
      LV=NHR                                                            EMF01050
                IF(INREAD.EQ.1)GO TO 989                                EMF01060
C..... IF READ IN NOT DESIRED THEN USE THE VALUES GENERATED BY I N I T  EMF01070
                         GO TO 999                                      EMF01080
989             CONTINUE                                                EMF01090
C............READ IN SEGMENT; UNIT NUMBERS 4 AND 8                      EMF01100
C            ---------------                                            EMF01110
      DO 21 K=NHR,LV                                                    EMF01120
      DO 21 J=1,JN                                                      EMF01130
  21  READ(4,14) (A(I,J,K),I=1,IN)                                      EMF01140
      DO 22 K=NW,IE                                                     EMF01150
      DO 22 J=1,JN                                                      EMF01160
  22  READ(8,14) (A(I,J,K),I=I1,I3)                                     EMF01170
999                     CONTINUE                                        EMF01180
  23  CONTINUE                                                          EMF01190
      LI=1                                                              EMF01200
      CALL PROP(1)                                                      EMF01210
C..............      SOLVE MAXWELL'S EQUATIONS                          EMF01220
C                    -------------------------                          EMF01230
      IF(LI.EQ.2) GO TO 100                                             EMF01240
      IF(NFIELD.EQ.1)CALL FIELD                                         EMF01250
979   CONTINUE                                                          EMF01260
      IF(LF.EQ.2) GO TO 30                                              EMF01270
  100 CONTINUE                                                          EMF01280
C............         PERFORM FLUID FLOW CALCULATIONS                   EMF01290
C                     -------------------------------                   EMF01300
      NITER=0                                                           EMF01310
    1 CONTINUE                                                          EMF01320
C     ========                                                          EMF01330
      NITER=NITER+1                                                     EMF01340
C.............. CAUSE ONE CYCLE OF ITERATION TO BE PERFORMED            EMF01350
```

```
C
                             IF(NFLOW.EQ.0) GO TO 30
      CALL  EQN
      WRITE(15,109) XI
C..................... TEST IF PRINTOUT TO BE PRODUCED
C
      IF((NITER+NPRINT-IP)/NPRINT.NE.NITER/NPRINT)  GO TO 10
      WRITE(6,103) (ASYMBL(K),K=1,7)
      WRITE(6,104)NITER,(RSDU(K),K=1,7),NWJ,NWI,NFJ,NFI,NTJ,NTI
      WRITE(6,107) (ASYMBL(K),K=8,14)
      WRITE(6,108) NITER,(SAN(K),K=1,7),NKJ,NKI,NEJ,NEI
C     WRITE(15,108) NITER,(SAN(K),K=1,7),NKJ,NKI,NEJ,NEI
      WRITE(6,109) XI
C     WRITE(15,109) XI
   10 CONTINUE
C.......... TEST IF MAXIUM NUMBER OF ITERATONS PERFORMED
C
      IF(NITER.EQ.NMAX)  GO TO 8
      RES=0.
      DO 7 K=1,IE
      IF(ABS(RES).LT.ABS(RSDU(K))) RES=RSDU(K)
      RSDU(K)=0.
    7 CONTINUE
      SANWF=0.
      DO 3 K=1,IE
      IF(ABS(SANWF).LT.ABS(SAN(K))) SANWF=SAN(K)
      SAN(K)=0.
    3 CONTINUE
C*** TEST IF CONVERGENCE CRITERION SATISFIED;IF NOT,BACK TO(1)
      IF(ABS(SANWF).GT..001) GO TO 1
      IF (NITER.LE.NMIN) GO TO 1
C....................."CONVERGED"
C                      ----------
      WRITE(6,105) NITER
      GO TO 340
C...................... "UNCONVERGED"
C                       -----------
    8 WRITE(6,106) NITER
C                FOLLOW UP CALCULATIONS
C              **------------------------**
340    CONTINUE
       CALL CONLOT(XI,PERFIN,PCASC)
C      CALCULATE THE LOCAL MIXING TIMES
C      -------------------------------
              DEPTH=X(5)-X(1)
```

```
              DIMNSN=DEPTH                                              EMF01810
              IL=I1+1                                                   EMF01820
              IH=I3-1                                                   EMF01830
              DO 900 J=1,JNM                                            EMF01840
              DO 900 I=IL,IH                                            EMF01850
       TIMIX(I,J) = 80.0*(DIMNSN**2/(1000.*A(I,J,NEP)))**0.333          EMF01860
900           CONTINUE                                                  EMF01870
C        CALCULATE THE TOTAL MIXING TIME FOR VESSEL                     EMF01880
C        -----------------------------------------                      EMF01890
              XMASS=RM**2*DEPTH*ROREF(1)/2.0                            EMF01900
              PSPEC=PCASC/XMASS                                         EMF01910
              TOTALT=80.0*(DIMNSN**2/(ETA*1000.*PSPEC))**0.333          EMF01920
C---------------------------------------------------------------------  EMF01930
C             CALCULATE THE POWER INPUT PER TON MELT                    EMF01940
C             --------------------------------------                    EMF01950
              PTON =  PSPEC* 1000.0                                     EMF01960
C--------------------------------------------------------------------   EMF01970
C................MAIN RESULTS PRINTED OUT HERE                          EMF01980
C                -----------------------------                          EMF01990
       IF(PERFIN.GT.PIMIN)WRITE(6,250) PERFIN                           EMF02000
       IF(PERFIN.LE.PIMIN) WRITE(6,252) PERFIN                          EMF02010
       WRITE(6,254) PTON                                                EMF02020
       WRITE(6,253) TOTALT                                              EMF02030
            CALL PRINT(NW,NF,2)                                         EMF02040
            CALL PRINT(NV1,NV2,1)                                       EMF02050
        IF (NTURB.EQ.0) GO TO 355                                       EMF02060
C....TURBULENT RESULTS PRINT OUT                                        EMF02070
            CALL PRINT(NK,NEP,2)                                        EMF02080
            CALL PRINT(NMU,NMU,2)                                       EMF02090
C------------------------------------------------------------------     EMF02100
C                  WRITE OUTPUT DATA FILE                               EMF02110
C                  ----------------------                               EMF02120
355         CONTINUE                                                    EMF02130
               DO 15 K=NW,IE                                            EMF02140
               DO 16 J=1,JN                                             EMF02150
16       WRITE(7,14)(A(I,J,K),I=I1,I3)                                  EMF02160
15             CONTINUE                                                 EMF02170
C                                                                       EMF02180
              DO 11 K=NHR,LV                                            EMF02190
              DO 11 J=1,JN                                              EMF02200
         WRITE(3,14)(A(I,J,K),I=1,IN)                                   EMF02210
 11           CONTINUE                                                  EMF02220
C---------------------------------------------------------------------  EMF02230
C             CALCULATE GENERATION AND DISSIPATION ENERGY               EMF02240
C             ------------------------------------------                EMF02250
```

```
C...ENERGY GENERATION:
C   -----------------
 50                 CALL VOL(GK,PGEN)
C...DISSIPATION:
C   -----------
                    DO 52 I=1,IN
                    DO 52 J=1,JN
                    EPSIL(I,J)=0.0
 52                 CONTINUE
                         IL=I1+1
                         IH=I3-1
                         DO 400 I=IL,IH
                         DO 400 J=2,JNM
                    EPSIL(I,J)=A(I,J,NEP)
 400                     CONTINUE
               CALL VOL(EPSIL,PDISIP)
               PDISIP=PDISIP*ROREF(1)
C-------------------------------------------------------------------
C...PRINT IT:
C   --------
                    WRITE(6,410) PGEN,PDISIP,PCASC
 410      FORMAT('TOTAL GENERATION= ',1PE10.2/'TOTAL DISSIPATION= '
     1,1PE10.2/'TOTAL POWER INPUT=  ',1PE10.2)
C-----------------------------------------------------------------------
C*********************    S T O P   *********************
 969  CONTINUE
C *** SAVE DATA FOR PLOTTING (UNIT #9)
      NX=I3-I1+1
      NY=JN
      WRITE(9,1233) NX,NY,J1,NSCHEM,CU,ETA
C                COORDINATE DATA
C                ---------------
      DO 34 I=I1,I3
      L=I-I1+1
  34  XM(L)=X1(I)-X1(I1)
      WRITE(9,1234) (XM(L),L=1,NX)
      WRITE(9,1234) (X2(J),J=1,JN)
            K=NJV1
C           -----
      DO 32 J=1,NY
  32  WRITE(9,1234) (A(I,J,K),I=I1,I3)
            K=NJV2
C           -----
      DO 33 J=1,NY
  33  WRITE(9,1234)(A(I,J,K),I=I1,I3)
```

```
   30  CONTINUE
              K=NK
C             -----
       DO 121 J=1,JN
       WRITE(9,1234) (A(I,J,K),I=I1,I3)
  121  CONTINUE
              K=NEP
C             -----
       DO 122 J=1,JN
  122  WRITE(9,1234)(A(I,J,K),I=I1,I3)
              K=NMU
C             -----
       DO 123 J=1,JN
  123  WRITE(9,1234)(A(I,J,K),I=I1,I3)
C            LOCAL MIXING TIMES
C            ------------------
       DO 124 J=1,JN
  124  WRITE(9,1234)(TIMIX(I,J),I=I1,I3)
       STOP
C-----------------------------------------------------------------
C                F O R M A T T I N G
C                -------------------
    9  FORMAT(////'          C O O R D I N A T E     D A T A '//)
   12  FORMAT(4H I1=,I2,4H I2=,I2,4H I3=,I2,4H IA=,I2,4H IN=,I2
      1,4H J1=,I2,4H JN=,I2)
   14  FORMAT(7(1PE10.3))
   40  FORMAT(27H0THE ELECTRODE TIP SHAPE IS/
      125X,5HX2(J),25X,6HX1(II))
   42  FORMAT(20X,E15.5,15X,E15.5)
  101  FORMAT(25H0DISTANCES IN DIRECTION-1/(1H ,7E15.5))
  102  FORMAT(25H0DISTANCES IN DIRECTION-2/(1H ,7E15.5))
  103  FORMAT(36H0MAXIMUM RESIDUAL FOR EACH VARIABLE://
      16H0NITER,7(3X,A6),7X,'NWJ NWI NFJ NFI NTJ NTI'//)
  104  FORMAT(1H0,I3,3X,7(F9.4),5X,I2,5(I4))
  105  FORMAT(32H0THE PROCESS        CONVERGED IN,I5,13H   ITERATIONS)
  106  FORMAT(32H0THE PROCESS DID NOT CONVERGE IN,I5,13H   ITERATIONS)
  107  FORMAT(36H0MAXIMUM RESIDUAL FOR EACH VARIABLE://
      16H0NITER,7(3X,A6),7X,'NKJ NKI NEJ NEI'//)
  108  FORMAT(1H0,I3,3X,7(F9.4),5X,I2,3(I4))
  109  FORMAT(' XI=  ',1PE10.2)
  200  FORMAT( 6A6)
  205  FORMAT(5I5,F5.2,6I5)
  208  FORMAT('      I N P U T   D A T A '//)
  210      FORMAT('CURRENT= ',1PE10.2,' KILO-AMPS'
      1/'MOLECULAR VISCOSITY= ',1PE10.2,' KG/M-S'
```

```
     2/'MELT DENSITY= ',1PE10.2,' KG/(M-CUBE)'                          EMF03160
     3/'ELEC. CONDUCTIVITY OF MELT=',1PE10.2,' MHO-M'                   EMF03170
     4/'MAX.ITERATIONS FOR FLOW ',I5)                                   EMF03180
213      FORMAT('COLD START CALCULATION; DATA NOT READ IN')             EMF03190
214      FORMAT('CONTINUATION CALCULATION;PREVIOUS DATA READ')          EMF03200
215      FORMAT('CURRENT TYPE IS ==>A.C.')                              EMF03210
216      FORMAT('CURRENT TYPE IS ==>D.C.')                              EMF03220
217      FORMAT('L A M I N A R    FLOW CALCULATION')                    EMF03230
218      FORMAT('T U R B U L E N T   FLOW CALCULATION')                 EMF03240
229      FORMAT('K-EPSILON TURBULENCE MODEL')                           EMF03250
230      FORMAT('SPECRAL MODEL FOR TURBULENCE')                         EMF03260
231      FORMAT('E-M FIELD CALCULATIONS => O F F ')                     EMF03270
232      FORMAT('E-M FIELD CALCULATIONS => O N')                        EMF03280
233      FORMAT('FLOW CALCULATIONS      => O F F')                      EMF03290
234      FORMAT('FLOW CALCULATIONS      => O N')                        EMF03300
235      FORMAT('USING LOW REYNOLD NUMBER MODEL ')                      EMF03310
236      FORMAT('BASE VALUE OF CONSTANTS ARE:'/                         EMF03320
     1'CD=    ',F5.2/                                                   EMF03330
     2'C1=    ',F5.2/                                                   EMF03340
     3'C2=    ',F5.2)                                                   EMF03350
250      FORMAT(//' AUTOMATIC TURBULENCE CHECK; FLOW FOUND TO BE:'      EMF03360
     1/10X,'.....T U R B U L E N T '                                    EMF03370
     2/10X,'PERFORMANCE INDEX= ',F7.2)                                  EMF03380
252      FORMAT(' AUTOMATIC TURBULECE CHECK; FLOW FOUND TO BE:'         EMF03390
     1/10X,'.....L A M I N A R '                                        EMF03400
     2/10X,'PERFORMANCE INDEX= ',F7.2)                                  EMF03410
254      FORMAT(10X,'POWER INPUT= ',1PE10.2,' WATTS/TON OF MELT')       EMF03420
253    FORMAT(10X,'OVERALL MIXING TIME(95% MIXED)= ',F6.2,' SECS'//)    EMF03430
220      FORMAT('ELECTRODE RADIUS= ',1PE10.2,' M'                       EMF03440
     1/'VESSEL RADIUS= ',1PE10.2,' M'                                   EMF03450
     2/'MELT DEPTH= ',1PE10.2,' M'                                      EMF03460
     3/'ELECTRODE IMMERSION= ',1PE10.2,' M')                            EMF03470
  225 FORMAT(/,50H DRIVING FORCES ARE FIRST ELECTROMAG THEN BUOYANCY)   EMF03480
 1233 FORMAT(4I4,2F5.2)                                                 EMF03490
 1234 FORMAT(5E13.5)                                                    EMF03500
      END                                                               EMF03510
```

```
      SUBROUTINE EQN
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1
      COMMON/CDIM/X(5),RE,RM
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC
      COMMON/CCONT/C1,C2,C3,CD
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)
      COMMON/CEMP/WF,P,S(4)
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW
      COMMON/CTRVL/VE,VC
      COMMON/CHEAT/HC,HW(4)
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA
      COMMON/CVF/FES(21),FEM(21),FSM,FSE
      COMMON/CDROP/TAU,HCD,D,QS,QM
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)
     1,CAPPA,E
      COMMON/CFLD/CU
      COMMON/CPI/PI
      COMMON/CTYP/LTYP
      COMMON/CANG/THETA
      COMMON/FITER/MIT,MPRINT,CP
      COMMON/CDVAR/A(31,21,20)
      COMMON/CNAME/ANAME,ASYMBL
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA
      COMMON/CLF/LF,LI
      COMMON/CSVOR/SVOR(31,21)
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)
      COMMON/CAVM/AVM(60)
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR
      COMMON/CSUR/HS(2),TS(2)
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP
      COMMON/CSORSE/SOURCE,SPRIME
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5
      COMMON/CAF/FUN(20)
      COMMON/CVOLT/VOLT
      COMMON/CAVT/TB
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
      COMMON/CLREYN/LREYN,CDMOD(31,31)
```

```
  1   CONTINUE                                                          EQN00460
C                 UPDATE COEFFICIENT CD TO CDMOD(I,J) FOR               EQN00470
C                 LOW REYNOLDS NUMBER MODEL OF TURBULENCE.              EQN00480
C                 ---------------------------------------               EQN00490
                        IF(NTURB.EQ.0) GO TO 500                        EQN00500
              DO 500 J=1,JNM                                            EQN00510
         IL=I1+1                                                        EQN00520
         IH=I3-1                                                        EQN00530
              DO 500 I=IL,IH                                            EQN00540
         FMU=1.0                                                        EQN00550
              IF(LREYN.EQ.0.OR.A(I,J,NEP).EQ.0.0) GO TO 400             EQN00560
         REYLOC=A(I,J,NRO)*A(I,J,NK)**2/(ZMUREF(1)*A(I,J,NEP))          EQN00570
         FMU=EXP(-2.5/(1+REYLOC/50.0))                                  EQN00580
 400          CONTINUE                                                  EQN00590
         CDMOD(I,J)=CD*FMU                                              EQN00600
 500          CONTINUE                                                  EQN00610
C---------------------------------------------------------------------  EQN00620
C....................OBTAIN EFFECTIVE VISCOSITY                         EQN00630
C                    --------------------------                         EQN00640
      IF (NTURB.EQ.1) CALL  VISCOS                                      EQN00650
C.................... VORTICITY SUB-CYCLE                               EQN00660
C                     -------------------                               EQN00670
      GOSA=0.0                                                          EQN00680
      TOTA=0.0                                                          EQN00690
      CALL  VORITY                                                      EQN00700
      IF(GOSA.EQ.0.0.AND.TOTA.EQ.0.0) GO TO 10                          EQN00710
      SAN(NW)=GOSA/TOTA                                                 EQN00720
      GO TO 40                                                          EQN00730
   10 SAN(NW)=1.                                                        EQN00740
   40 CONTINUE                                                          EQN00750
C.................. STREAM FUNCTION SUB-CYCLE                           EQN00760
C                   -------------------------                           EQN00770
      GOSA=0.0                                                          EQN00780
      TOTA=0.0                                                          EQN00790
      CALL  STRFUN                                                      EQN00800
      IF(GOSA.EQ.0.0.AND.TOTA.EQ.0.0) GO TO 20                          EQN00810
      SAN(NF)=GOSA/TOTA                                                 EQN00820
      GO TO 50                                                          EQN00830
   20 SAN(NF)=1.                                                        EQN00840
   50 CONTINUE                                                          EQN00850
C....................  VELOCITY DISTRIBUTION                            EQN00860
C                      ---------------------                            EQN00870
      CALL  VELDIS                                                      EQN00880
          IF(NTURB.EQ.0) GO TO 120                                      EQN00890
C.............TURBULENT VARIABLE CALCULATIONS(ONLY IF NTURB=1)          EQN00900
```

```
C             -----------------------------------------------             EQN00910
C     CHOOSE WHICH SCHEME OF CALCULATION IS DESIRED.                      EQN00920
C                   NSCHEM=0  ...FOR K-EPSILON MODEL                      EQN00930
C                   NSCHEM=1  ...FOR MSR MODEL.                           EQN00940
C-------------------------------------------------------------------      EQN00950
      CALL WALL                                                           EQN00960
      NEND=NK                                                             EQN00970
      IF(NSCHEM.EQ.0) NEND=NEP                                            EQN00980
C                                                                         EQN00990
      DO 41 K=NK,NEND                                                     EQN01000
      GOSA=0.0                                                            EQN01010
      TOTA=0.0                                                            EQN01020
      IF(NSCHEM.EQ.0) CALL TURVAR(K)                                      EQN01030
      IF(NSCHEM.EQ.1) CALL TURVAR(K)                                      EQN01040
      IF(GOSA.EQ.0.0.AND.TOTA.EQ.0.0) GO TO 30                            EQN01050
      SAN(K)=GOSA/TOTA                                                    EQN01060
      GO TO 41                                                            EQN01070
   30 SAN(K)=1.                                                           EQN01080
   41 CONTINUE                                                            EQN01090
C............ INITIATE ITERATION ON BOUNDARY NODES                         EQN01100
C             ------------------------------------                        EQN01110
120   CALL  BOUND                                                         EQN01120
C.....  CALL DISSIP IF SO DESIRED BY CHOICE OF SCHEME.                    EQN01130
C       ----------------------------------------------                    EQN01140
        IF (NTURB.EQ.1.AND.NSCHEM.EQ.1)CALL DISSIP                        EQN01150
C            OUTPUT CDMOD(I,J) AS A CHECK                                 EQN01160
C            -----------------------------                                EQN01170
                   DO 550 J=1,JNM                                         EQN01180
             IL=I1+1                                                      EQN01190
             IH=I3-1                                                      EQN01200
             WRITE(18,560)(CDMOD(I,J),I=IL,IH)                            EQN01210
550                CONTINUE                                               EQN01220
560           FORMAT(10F5.2)                                              EQN01230
      RETURN                                                              EQN01240
      END                                                                 EQN01250
```

```
      SUBROUTINE FIELD                                                  FIE00010
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$  FIE00020
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT          FIE00030
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)                FIE00040
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1                  FIE00050
      COMMON/CDIM/X(5),RE,RM                                            FIE00060
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC                        FIE00070
      COMMON/CCONT/C1,C2,C3,CD                                          FIE00080
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)                              FIE00090
      COMMON/CEMP/WF,P,S(4)                                             FIE00100
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS                     FIE00110
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL            FIE00120
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW                                 FIE00130
      COMMON/CTRVL/VE,VC                                                FIE00140
      COMMON/CHEAT/HC,HW(4)                                             FIE00150
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA                       FIE00160
      COMMON/CVF/FES(21),FEM(21),FSM,FSE                                FIE00170
      COMMON/CDROP/TAU,HCD,D,QS,QM                                      FIE00180
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)    FIE00190
     1,CAPPA,E                                                          FIE00200
      COMMON/CFLD/CU                                                    FIE00210
      COMMON/CPI/PI                                                     FIE00220
      COMMON/CTYP/LTYP                                                  FIE00230
      COMMON/CANG/THETA                                                 FIE00240
      COMMON/FITER/MIT,MPRINT,CP                                        FIE00250
      COMMON/CDVAR/A(31,21,20)                                          FIE00260
      COMMON/CNAME/ANAME,ASYMBL                                          FIE00270
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA    FIE00280
      COMMON/CLF/LF,LI                                                  FIE00290
      COMMON/CSVOR/SVOR(31,21)                                          FIE00300
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI                           FIE00310
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)               FIE00320
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)                             FIE00330
      COMMON/CAVM/AVM(60)                                               FIE00340
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR                        FIE00350
      COMMON/CSUR/HS(2),TS(2)                                           FIE00360
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP                                    FIE00370
      COMMON/CSORSE/SOURCE,SPRIME                                       FIE00380
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5                              FIE00390
      COMMON/CAF/FUN(20)                                                FIE00400
      COMMON/CVOLT/VOLT                                                 FIE00410
      COMMON/CAVT/TB                                                    FIE00420
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)                FIE00430
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$  FIE00440
      NIT=0                                                             FIE00450
```

```
    1 CONTINUE
      GOSA=0.
      TOTA=0.
C        CONVECTION HAS ZERO EFFECT ON MAGNETIC DIFFUSION EQN.
                 AE=0.
                 AW=0.
                 AN=0.
                 AS=0.
C-----------------------------------------------------------------------
C          CALCULATE MAGNETIC INDUCTION AT INTERIOR POINTS
C                ------------------
      LV=NHI
      IF(LTYP.EQ.2) LV=NHR
                  DO 17 K=NHR,LV
                  DO 11 J=2,JNM
      IL=IMIN(J)
      IF(J.EQ.J1) IL=I1+1
      IH=IMAX(J)
C          BRANCH TO POINTS ON ELECTRODE FACES
C          ----------------------------------
                  DO 19 I=IL,IH
                  IF(J.EQ.J1.AND.I.LE.I2) GO TO 30
                  IF(I.EQ.IW(J).AND.J.LT.J1) GO TO 15
                  IF(I.EQ.I3) GO TO 14
      CALL SORCE(SOURCE,I,J,K)
      RISQ=1./R(J)/R(J)
      SOP=A(I,J,NS)
      BBE=4./(A(I+1,J,NS)+SOP)*RISQ*BE(I)
      IF(I.EQ.(IW(J)-1).AND.J.LT.J1) BBE=4./(SOP+SOP)*RISQ*BE(I)
      BBW=4./(A(I-1,J,NS)+SOP)*RISQ*BW(I)
      IF(I.EQ.(I3+1)) BBW=4./(SOP+SOP)*RISQ*BW(I)
      BBN=16./(A(I,J+1,NS)+SOP)/((R(J+1)+R(J))**2)*BN(J)
      IF(J.EQ.(J1-1).AND.I.LT.I2) BBN=BBN*(A(I,J+1,NS)+SOP)/(2.*SOP)
      IF(I.EQ.(IW(J)-1).AND.J.LT.J1) BBN=BBN*(A(I,J+1,NS)+SOP
     1)/(2.*SOP)
      BBS=16./(A(I,J-1,NS)+SOP)/((R(J-1)+R(J))**2)*BS(J)
      ANUM=(AE+R(J)*R(J)*BBE)*A(I+1,J,K)
     1+(AW+R(J)*R(J)*BBW)*A(I-1,J,K)
     2+(AN+R(J+1)*R(J+1)*BBN)*A(I,J+1,K)
     3+(AS+R(J-1)*R(J-1)*BBS)*A(I,J-1,K)+SOURCE
      ADNM=AE+AW+AN+AS+R(J)*R(J)*(BBE+BBW+BBN+BBS)
                  GO TO 21
C-----------------------------------------------------------------------
C              POINTS ON ELECTRODE FACES
C              ------------------------
```

```
C .......TOP ELECTRODE SIDE SURFACE(J=J1,I1<I<I2)                         FIE00910
C        ---------------------------                                      FIE00920
  30  BB=(X2(J1)-X2(J1-1))/(X2(J1+1)-X2(J1))*A(I,J1-1,NS)/A(I,J1,NS)      FIE00930
      ANUM=R(J-1)*R(J-1)*A(I,J-1,K)+BB*R(J+1)*R(J+1)*A(I,J+1,K)           FIE00940
      ADNM=R(J)*R(J)*(1.+BB)                                              FIE00950
      GO TO 21                                                            FIE00960
C.................BOTTOM ELECTRODE SURFACE(I=I3)                          FIE00970
C                 ------------------------                                FIE00980
  14  CON1=A(I3,J,NS)/S(3)                                                FIE00990
      CON=CON1                                                            FIE01000
      XQ=X1(I)-X1(I-1)                                                    FIE01010
      XR=X1(I+1)-X1(I)                                                    FIE01020
      BB=1./(1.+XQ/XR*CON)                                                FIE01030
      ANUM=BB*A(I-1,J,K)+(1.-BB)*A(I+1,J,K)                               FIE01040
      ADNM=1.                                                             FIE01050
      GO TO 21                                                            FIE01060
C................TOP ELECTRODE,AXIAL SURFACE(I=IW)                        FIE01070
C                ---------------------------                              FIE01080
  15  CON=A(I-1,J,NS)/A(I,J,NS)                                           FIE01090
      RSQ=R(J)*R(J)                                                       FIE01100
      X1Q=X1(I)-X1(I-1)                                                   FIE01110
      X1R=X1(I+1)-X1(I)                                                   FIE01120
      X2Q=X2(J)-X2(J-1)                                                   FIE01130
      X2R=X2(J+1)-X2(J)                                                   FIE01140
      BB1=CON*X1Q/X1R                                                     FIE01150
      BB2=CON*X2Q/X2R                                                     FIE01160
      BINC=X1Q/X2Q*SIN(THETA)/COS(THETA)                                  FIE01170
      ANUM=A(I-1,J,K)+BB1*A(I+1,J,K)+                                     FIE01180
     1BINC*(R(J-1)*R(J-1)*A(I,J-1,K)+BB2*R(J+1)*R(J+1)*A(I,J+1,K)         FIE01190
     2)/RSQ                                                               FIE01200
      ADNM=1.+BB1+BINC*(1.+BB2)                                           FIE01210
  21  IF(ADNM.EQ.0.) GO TO 6                                              FIE01220
      Z=A(I,J,K)                                                          FIE01230
      A(I,J,K)=ANUM/ADNM                                                  FIE01240
      IF(Z.EQ.0.0.AND.A(I,J,K).EQ.0.0) GO TO 18                           FIE01250
      RS=1.-Z/A(I,J,K)                                                    FIE01260
      GO TO 3                                                             FIE01270
  18   RS=0.                                                              FIE01280
  3   CONTINUE                                                            FIE01290
C-----------------------------------------------------------------------  FIE01300
C             RELAX THE NEW VALUES                                        FIE01310
C             --------------------                                        FIE01320
      A(I,J,K)=Z+RP(K)*(A(I,J,K)-Z)                                       FIE01330
C             ERROR CALCULATIONS                                          FIE01340
C             ------------------                                          FIE01350
```

```
              GOSA=GOSA+ABS(Z-A(I,J,K))                                 FIE01360
              TOTA=TOTA+ABS(A(I,J,K))                                   FIE01370
C ...........STORE MAXIMUM RESIDUAL                                     FIE01380
C            ---------------------                                      FIE01390
      IF(ABS(RS).LE.ABS(RSDU(K))) GO TO 6                               FIE01400
      RSDU(K)=RS                                                        FIE01410
  6                          CONTINUE                                   FIE01420
  19                         CONTINUE                                   FIE01430
  11                         CONTINUE                                   FIE01440
              IF(GOSA.EQ.0.0.AND.TOTA.EQ.0.0) GO TO 7                   FIE01450
              SAN(K)=GOSA/TOTA                                          FIE01460
                             GO TO 8                                    FIE01470
  7                          SAN(K)=1.                                  FIE01480
  8                          CONTINUE                                   FIE01490
  17                         CONTINUE                                   FIE01500
C---------------------------------------------------------------------- FIE01510
C               BOUNDARY CONDITIONS(I=1,J<J1)                           FIE01520
C               -------------------                                     FIE01530
              J11=J1-1                                                  FIE01540
              XQ=X1(2)-X1(1)                                            FIE01550
              XR=X1(3)-X1(1)                                            FIE01560
              BB=XR*XR/(XR*XR-XQ*XQ)                                    FIE01570
                             DO 10 K=NHR,NHI                            FIE01580
                             DO 10 J=2,J11                              FIE01590
  10                         A(1,J,K)=BB*A(2,J,K)+(1.-BB)*A(3,J,K)      FIE01600
C..................BOUNDARY I=IN                                        FIE01610
C                  -------------                                        FIE01620
      XQ=X1(IN)-X1(INM)                                                 FIE01630
      XR=X1(IN)-X1(INM-1)                                               FIE01640
      BB=XR*XR/(XR*XR-XQ*XQ)                                            FIE01650
                            DO 151 K=NHR,NHI                            FIE01660
                            DO 151 J=2,JNM                              FIE01670
      A(IN,J,K)=BB*A(INM,J,K)+(1.-BB)*A(INM-1,J,K)                      FIE01680
  151                       CONTINUE                                    FIE01690
C .............. J=1,  1<I<IN                                           FIE01700
C                -----------                                            FIE01710
                             DO 25 K=NHR,NHI                            FIE01720
                             DO 25 I=1,IN                               FIE01730
      A(I,1,K)=A(I,2,K)                                                 FIE01740
25                           CONTINUE                                   FIE01750
                             IF(LI.EQ.2) GO TO 9                        FIE01760
C---------------------------------------------------------------------- FIE01770
C                  CONVERGENCE CRITERION CHECK                          FIE01780
C                  ---------------------------                          FIE01790
      CF1=ABS(RSDU(NHR))                                                FIE01800
```

```
      CF2=ABS(RSDU(NHI))                                                FIE01810
      CFF=CF1                                                           FIE01820
      IF(CF1.LT.CF2) CFF=CF2                                            FIE01830
      IF(CFF.LE.CP) GO TO 9                                             FIE01840
      NIT=NIT+1                                                         FIE01850
      RSDU(NHR)=0.                                                      FIE01860
      RSDU(NHI)=0.                                                      FIE01870
C---------------------------------------------------------------------- FIE01880
C                     PRINTOUT CHECK                                    FIE01890
C                     --------------                                    FIE01900
      K1=NIT/MPRINT                                                     FIE01910
      K2=K1*MPRINT                                                      FIE01920
                            IF(NIT.NE.K2) GO TO 5                       FIE01930
      WRITE(6,104) NIT,CF1,CF2                                          FIE01940
  104 FORMAT(/,6H  NIT=,I4/,6H  CF1=,E14.8,6H  CF2=,E14.8/)             FIE01950
  5                         CONTINUE                                    FIE01960
C---------------------------------------------------------------------  FIE01970
C                    MAXIMUM ITERATION CHECK                            FIE01980
C                    -----------------------                            FIE01990
                             IF(NIT.LT.MIT) GO TO 1                     FIE02000
      WRITE(6,106) NIT                                                  FIE02010
  106 FORMAT(32HOTHE FLD.EQN DID NOT CONVERGE IN,I5,13H   ITERATIONS)   FIE02020
      LF=2                                                              FIE02030
                             GO TO 20                                   FIE02040
C---------------------------------------------------------------------  FIE02050
C           CALCULATE CURRENT DENSITY                                   FIE02060
C           -------------------------                                   FIE02070
  9                          CONTINUE                                   FIE02080
                   DO 29 J=1,JN                                         FIE02090
                   DO 191 I=1,IN                                        FIE02100
      IF(I.LT.I1.AND.J.GT.J1) GO TO 191                                 FIE02110
      IF(I.EQ.1.OR.I.EQ.IN) GO TO 192                                   FIE02120
      IF(J.EQ.J1.AND.I.LE.I1) GO TO 192                                 FIE02130
      IF(J.EQ.1.OR.J.EQ.JN) GO TO 192                                   FIE02140
      A(I,J,NJRR)=-ADF(I,J,1,NHR)*R(J)                                  FIE02150
      A(I,J,NJRI)=-ADF(I,J,1,NHI)*R(J)                                  FIE02160
      IF(I.EQ.I1.AND.J.GT.J1) GO TO 191                                 FIE02170
  192              A(I,J,NJZR)=R(J)*ADF(I,J,2,NHR)+A(I,J,NHR)*2.        FIE02180
                   A(I,J,NJZI)=R(J)*ADF(I,J,2,NHI)+A(I,J,NHI)*2.        FIE02190
  191              CONTINUE                                             FIE02200
  29                         CONTINUE                                   FIE02210
C-----------------------------------------------------------------      FIE02220
C         MODIFICATIONS FOR CURRENT AT THE BOUNDARIES                   FIE02230
C         -----------------------------------------                     FIE02240
C ................... AT I=IW ,J<J1                                     FIE02250
```

```
C                     -------------                                     FIE02260
                   DO 27 J=2,J1                                         FIE02270
      II=IW(J)                                                          FIE02280
      A(II,J,NJRR)=(A(II,J,NHR)-A(II+1,J,NHR))/(X1(II+1)-X1(II))*R(J)   FIE02290
      A(II,J,NJRI)=(A(II,J,NHI)-A(II+1,J,NHI))/(X1(II+1)-X1(II))*R(J)   FIE02300
      A(II,J,NJZR)=(R(J+1)*R(J+1)*A(II,J+1,NHR)-R(J)*R(J)*A(II,J        FIE02310
     1,NHR))/(X2(J+1)-X2(J))/R(J)                                       FIE02320
      A(II,J,NJZI)=(R(J+1)*R(J+1)*A(II,J+1,NHI)-R(J)*R(J)*A(II          FIE02330
     1,J,NHI))/(X2(J+1)-X2(J))/R(J)                                     FIE02340
  27               CONTINUE                                             FIE02350
C...................AT I=I3                                             FIE02360
C                   --------                                            FIE02370
C     DO 28 J=2,JNM                                                     FIE02380
C     A(I3,J,NJRR)=(A(I3-1,J,NHR)-A(I3,J,NHR))/(X1(I3)-X1(I3-1))*R(J)   FIE02390
C     A(I3,J,NJRI)=(A(I3-1,J,NHI)-A(I3,J,NHI))/(X1(I3)-X1(I3-1))*R(J)   FIE02400
  28  CONTINUE                                                          FIE02410
C ................. AT J=J1 ,I1<I<I2                                    FIE02420
C                   ----------------                                    FIE02430
      IF(I1.EQ.I2) GO TO 32                                             FIE02440
                             DO 31 I=I1,I2                              FIE02450
      A(I,J1,NJZR)=(R(J1+1)*R(J1+1)*A(I,J1+1,NHR)-R(J1)*R(J1)*A(I,J1,   FIE02460
     1NHR))/(X2(J1+1)-X2(J1))/R(J1)                                     FIE02470
      A(I,J1,NJZI)=(R(J1+1)*R(J1+1)*A(I,J1+1,NHI)-R(J1)*R(J1)*A(I,J1    FIE02480
     1,NHI))/(X2(J1+1)-X2(J1))/R(J1)                                    FIE02490
  31                         CONTINUE                                   FIE02500
  32                         CONTINUE                                   FIE02510
                             DO 115 I=1,IN                              FIE02520
                             DO 115 J=1,JN                              FIE02530
                             IF(J.GT.J1.AND.I.LT.I1) GO TO 115          FIE02540
      A(I,J,NJJ)=(A(I,J,NJRR)**2+A(I,J,NJRI)**2+A(I,J,NJZR)**2+A(I,J,   FIE02550
     1NJZI)**2)                                                         FIE02560
      A(I,J,NJJ)=500.*A(I,J,NJJ)/A(I,J,NS)                              FIE02570
  115                        CONTINUE                                   FIE02580
C........................CALCULATE VOLTAGE                              FIE02590
C                        -----------------                              FIE02600
                   DO 200 I=I1,I3                                       FIE02610
      N=I-I1+1                                                          FIE02620
      JL=J1                                                             FIE02630
      IF(I.GT.I2.AND.I.LE.IW(1)) JL=J1-(I-I2)                           FIE02640
      IF(I.GT.IW(1)) JL=1                                               FIE02650
                   DO 201 J=1,JN                                        FIE02660
  201              AVM(J)=R(J)*A(I,J,NJJ)                               FIE02670
  200              FUN(N)=FINT(JL,JN,2,1,1)                             FIE02680
      N3=I3-I1+1                                                        FIE02690
                   DO 202 N=1,N3                                        FIE02700
```

```
202              AVM(N)=FUN(N)                                        FIE02710
    SUMI=FINT(1,N3,1,I1,1)                                            FIE02720
    POWR=2.*PI*SUMI                                                   FIE02730
    VOLT=POWR*1.414/CU                                                FIE02740
    CURR=CU/1.414                                                     FIE02750
                 IF(LI.EQ.2) GO TO 37                                 FIE02760
    WRITE(6,110) NIT                                                  FIE02770
    CALL PRINT(NHR,LV,2)                                              FIE02780
37               CONTINUE                                             FIE02790
    WRITE(6,204) VOLT,CURR                                            FIE02800
110 FORMAT(32HOTHE FIELD EQUATION CONVERGED IN,I5,13H   ITERATIONS)   FIE02810
204 FORMAT(/,9H VOLTAGE=,F10.3,9H CURRENT=,F10.4)                     FIE02820
20  CONTINUE                                                          FIE02830
    RETURN                                                            FIE02840
    END                                                               FIE02850
```

```
      FUNCTION FINT(NL,NU,ND,NI,NJ)                                     FIN00010
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)                FIN00020
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1                  FIN00030
      COMMON/CAVM/AVM(60)                                               FIN00040
      NU1=NU-1                                                          FIN00050
      SUMA=0.                                                           FIN00060
      DO 10 N=NL,NU1                                                    FIN00070
      IF(ND.EQ.2) GO TO 1                                               FIN00080
      I=N+NI-1                                                          FIN00090
      DX=0.5*(X1(I+1)-X1(I))                                            FIN00100
      GO TO 2                                                           FIN00110
    1 J=N+NJ-1                                                          FIN00120
      DX=0.5*(X2(J+1)-X2(J))                                            FIN00130
    2 CONTINUE                                                          FIN00140
   10 SUMA=SUMA+(AVM(N)+AVM(N+1))*DX                                    FIN00150
      FINT=SUMA                                                         FIN00160
      RETURN                                                            FIN00170
      END                                                               FIN00180
```

```
      SUBROUTINE INIT
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1
      COMMON/CDIM/X(5),RE,RM
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC
      COMMON/CCONT/C1,C2,C3,CD
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)
      COMMON/CEMP/WF,P,S(4)
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW
      COMMON/CTRVL/VE,VC
      COMMON/CHEAT/HC,HW(4)
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA
      COMMON/CVF/FES(21),FEM(21),FSM,FSE
      COMMON/CDROP/TAU,HCD,D,QS,QM
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)
     1,CAPPA,E
      COMMON/CFLD/CU
      COMMON/CPI/PI
      COMMON/CTYP/LTYP
      COMMON/CANG/THETA
      COMMON/FITER/MIT,MPRINT,CP
      COMMON/CDVAR/A(31,21,20)
      COMMON/CNAME/ANAME,ASYMBL
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA
      COMMON/CLF/LF,LI
      COMMON/CSVOR/SVOR(31,21)
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)
      COMMON/CAVM/AVM(60)
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR
      COMMON/CSUR/HS(2),TS(2)
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP
      COMMON/CSORSE/SOURCE,SPRIME
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5
      COMMON/CAF/FUN(20)
      COMMON/CVOLT/VOLT
      COMMON/CAVT/TB
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
C.........INITIALISE ALL VALUES TO ZERO
```

```
C         ----------------------------                                  INI00460
                   DO 30 K=1,20                                         INI00470
                   DO 30 J=1,JN                                         INI00480
                   DO 30 I=1,IN                                         INI00490
      A(I,J,K)=0.0                                                      INI00500
  30               CONTINUE                                             INI00510
C---------------------------------------------------------------------- INI00520
C           MAGNETIC INDUCTION GUESS FIELD                              INI00530
C           -----------------------------                               INI00540
                   DO 31 J=1,J1                                         INI00550
                   DO 31 I=1,I2                                         INI00560
  31               A(I,J,NHR)=CU/(2.0*PI*RE*RE)                         INI00570
C---------------------------------------------------------------------- INI00580
C               GUESS FIELD FOR TEMPERATURE                             INI00590
C               --------------------------                              INI00600
                   DO 50 J=1,JN                                         INI00610
                   DO 50 I=1,I3                                         INI00620
                   IF(J.GT.J1.AND.I.LT.I1) GO TO 50                     INI00630
      A(I,J,NT)=300.0                                                   INI00640
      IF(J.LE.J1.AND.I.LE.I2) A(I,J,NT)=300.0                           INI00650
      IF(I.EQ.I3) A(I,J,NT)=300.0                                       INI00660
  50               CONTINUE                                             INI00670
C---------------------------------------------------------------------- INI00680
C                FIXED BOUNDARY CONDITIONS                              INI00690
C                ------------------------                               INI00700
                   DO 11 I=I1,IN                                        INI00710
  11               A(I,JN,NHR)=CU/(2.0*PI*RM*RM)                        INI00720
      J12=J1+1                                                          INI00730
C                                                                       INI00740
                   DO 12 J=J12,JN                                       INI00750
      A(I1,J,NHR)=CU/(2.0*PI*R(J)*R(J))                                 INI00760
  12               CONTINUE                                             INI00770
C                                                                       INI00780
                   DO 20 J=2,JNM                                        INI00790
      IL=I1+1                                                           INI00800
      IF(J.LE.J1) IL=IW(J)+1                                            INI00810
      IH=I3-1                                                           INI00820
                   DO  20  I=IL,IH                                      INI00830
      A(I,J,NK)=1.0E-05                                                 INI00840
      A(I,J,NEP)=1.0E-06                                                INI00850
  20               CONTINUE                                             INI00860
C                                                                       INI00870
                   DO 14 J=1,JN                                         INI00880
                   DO 14 I=1,I3                                         INI00890
  14               SVOR(I,J)=0.                                         INI00900
```

```
C                                                                       INI00910
                    DO 17 I=I1,IH                                       INI00920
   17               A(I,JN,NT)=TLS                                      INI00930
      RETURN                                                            INI00940
      END                                                               INI00950
```

```
      SUBROUTINE PRINT(NBEGIN,NTOTAL,LP)                                PRI00010
      REAL*8 ANAME(6,20),ASYMBL(14)                                     PRI00020
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$PRI00030
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT          PRI00040
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)                PRI00050
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1                  PRI00060
      COMMON/CDIM/X(5),RE,RM                                            PRI00070
      COMMON/CITER/NMAX,NPRINT;NITER,IP,IE,IV,CC                        PRI00080
      COMMON/CCONT/C1,C2,C3,CD                                          PRI00090
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)                              PRI00100
      COMMON/CEMP/WF,P,S(4)                                             PRI00110
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS                     PRI00120
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL            PRI00130
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW                                 PRI00140
      COMMON/CTRVL/VE,VC                                                PRI00150
      COMMON/CHEAT/HC,HW(4)                                             PRI00160
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA                       PRI00170
      COMMON/CVF/FES(21),FEM(21),FSM,FSE                                PRI00180
      COMMON/CDROP/TAU,HCD,D,QS,QM                                      PRI00190
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)    PRI00200
     1,CAPPA,E                                                          PRI00210
      COMMON/CFLD/CU                                                    PRI00220
      COMMON/CPI/PI                                                     PRI00230
      COMMON/CTYP/LTYP                                                  PRI00240
      COMMON/CANG/THETA                                                 PRI00250
      COMMON/FITER/MIT,MPRINT,CP                                        PRI00260
      COMMON/CDVAR/A(31,21,20)                                          PRI00270
      COMMON/CNAME/ANAME,ASYMBL                                         PRI00280
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA    PRI00290
      COMMON/CLF/LF,LI                                                  PRI00300
      COMMON/CSVOR/SVOR(31,21)                                          PRI00310
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI                           PRI00320
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)               PRI00330
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)                             PRI00340
      COMMON/CAVM/AVM(60)                                               PRI00350
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR                        PRI00360
      COMMON/CSUR/HS(2),TS(2)                                           PRI00370
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP                                    PRI00380
      COMMON/CSORSE/SOURCE,SPRIME                                       PRI00390
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5                              PRI00400
      COMMON/CAF/FUN(20)                                                PRI00410
      COMMON/CVOLT/VOLT                                                 PRI00420
      COMMON/CAVT/TB                                                    PRI00430
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)                PRI00440
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$PRI00450
```

```
      JX=JN/10                                                          PRI00460
      IF(JX.LT.1) JX=1                                                  PRI00470
      IX=IN/10                                                          PRI00480
      IF(IX.LT.1) IX=1                                                  PRI00490
      DO 10 K=NBEGIN,NTOTAL                                             PRI00500
      WRITE(6,100) (ANAME(L,K),L=1,6)                                   PRI00510
      IB=I3                                                             PRI00520
      JX=1                                                              PRI00530
      DO 2 L=1,JN,JX                                                    PRI00540
      J=JN+1-L                                                          PRI00550
      WRITE(6,103) (A(I,J,K),I=I1,IB,1)                                 PRI00560
      WRITE(6,105) J                                                    PRI00570
    2 CONTINUE                                                          PRI00580
      WRITE(6,104) (I,I=I1,IB,1)                                        PRI00590
   10 CONTINUE                                                          PRI00600
      IF(LP.EQ.2) GO TO 20                                              PRI00610
C     WRITE(6,1) HLW,HLEH,HLEV,HLS,HLD,HLM,POWR,VOLT                    PRI00620
      N1=I3-I1-1                                                        PRI00630
      J12=J1+1                                                          PRI00640
C     WRITE(6,6) (SN1(N),N=1,N1)                                        PRI00650
C     WRITE(6,7) (SN2(N),N=2,J1)                                        PRI00660
C     WRITE(6,8) (SN3(N),N=2,JNM)                                       PRI00670
      WRITE(6,15) (YP1(N),N=1,N1)                                       PRI00680
      WRITE(6,16) (YP2(N),N=2,J1)                                       PRI00690
      WRITE(6,17) (YP3(N),N=2,JNM)                                      PRI00700
      WRITE(6,19) (YP5(J),J=J12,JNM)                                    PRI00710
C     WRITE(6,212) VE,TB                                                PRI00720
   20 CONTINUE                                                          PRI00730
      RETURN                                                            PRI00740
  100 FORMAT(1H1,30X,21HTHE DISTRIBUTION OF   ,6A6/                     PRI00750
     126X,51H---------------------------------------------------/       PRI00760
     21H0,115X,1HJ/115X,3H---//)                                        PRI00770
    1 FORMAT(/,5H HLW=,E10.3,6H HLEH=,E10.3,6H HLEV=,E10.3              PRI00780
     1,5H HLS=,E10.3,5H HLD=,E10.3,5H HLM=,E10.3,6H POWR=,E10.3         PRI00790
     2/6H VOLT=,F10.3)                                                  PRI00800
  103 FORMAT(/(1H ,10(1PE10.3)))                                        PRI00810
  104 FORMAT(4H   I/(1H ,7X,9(I2,8X),I2))                               PRI00820
  105 FORMAT(105X,I2)                                                   PRI00830
  201 FORMAT(/,32H TEMP. DISTRIBUTION IN ELECTRODE)                     PRI00840
  203 FORMAT(/,10H J        =,I2,/,10(3X,F8.1))                         PRI00850
  207 FORMAT(/,4H YL=,F10.4,4H YS=,F10.4)                               PRI00860
  212 FORMAT(/,4H VE=,E12.4,4H TB=,F12.2)                               PRI00870
    6 FORMAT(25H0VALUES FOR SN1(N)    ARE/(1H ,5E14.4))                 PRI00880
    7 FORMAT(25H0VALUES FOR SN2(N)    ARE/(1H ,5E14.4))                 PRI00890
    8 FORMAT(25H0VALUES FOR SN3(N)    ARE/(1H ,5E14.4))                 PRI00900
```

```
   15 FORMAT(25HOVALUES FOR YP1(N)    ARE/(1H ,5E10.3))               PRI00910
   16 FORMAT(25HOVALUES FOR YP2(J)    ARE/(1H ,5E10.3))               PRI00920
   17 FORMAT(25HOVALUES FOR YP3(N)    ARE/(1H ,5E10.3))               PRI00930
   19 FORMAT(25HOVALUES FOR YP5(J)    ARE/(1H ,5E10.3))               PRI00940
      END                                                             PRI00950
```

```
      SUBROUTINE PROP(L)                                                PR000010
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$ PR000020
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT          PR000030
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)                PR000040
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1                  PR000050
      COMMON/CDIM/X(5),RE,RM                                            PR000060
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC                        PR000070
      COMMON/CCONT/C1,C2,C3,CD                                          PR000080
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)                              PR000090
      COMMON/CEMP/WF,P,S(4)                                             PR000100
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS                     PR000110
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL            PR000120
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW                                 PR000130
      COMMON/CTRVL/VE,VC                                                PR000140
      COMMON/CHEAT/HC,HW(4)                                             PR000150
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA                       PR000160
      COMMON/CVF/FES(21),FEM(21),FSM,FSE                                PR000170
      COMMON/CDROP/TAU,HCD,D,QS,QM                                      PR000180
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)    PR000190
     1,CAPPA,E                                                          PR000200
      COMMON/CFLD/CU                                                    PR000210
      COMMON/CPI/PI                                                     PR000220
      COMMON/CTYP/LTYP                                                  PR000230
      COMMON/CANG/THETA                                                 PR000240
      COMMON/FITER/MIT,MPRINT,CP                                        PR000250
      COMMON/CDVAR/A(31,21,20)                                          PR000260
      COMMON/CNAME/ANAME,ASYMBL                                         PR000270
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA    PR000280
      COMMON/CLF/LF,LI                                                  PR000290
      COMMON/CSVOR/SVOR(31,21)                                          PR000300
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI                           PR000310
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)               PR000320
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)                             PR000330
      COMMON/CAVM/AVM(60)                                               PR000340
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR                        PR000350
      COMMON/CSUR/HS(2),TS(2)                                           PR000360
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP                                    PR000370
      COMMON/CSORSE/SOURCE,SPRIME                                       PR000380
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5                              PR000390
      COMMON/CAF/FUN(20)                                                PR000400
      COMMON/CVOLT/VOLT                                                 PR000410
      COMMON/CAVT/TB                                                    PR000420
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)                PR000430
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$ PR000440
      IF(L.EQ.2) GO TO 1                                                PR000450
```

```
      DO 2 I=1,IN                                                       PR000460
      DO 2 J=1,JN                                                       PR000470
      IF(I.LE.I1.AND.J.LE.J1) GO TO 3                                   PR000480
      IF(I.LT.IW(J).AND.J.LT.J1) GO TO 3                                PR000490
      IF(I.GT.I3) GO TO 3                                               PR000500
      A(I,J,NRO)=ROREF(2)                                               PR000510
      A(I,J,NMU)=ZMUREF(1)                                              PR000520
      A(I,J,NTC)=TCREF(2)                                               PR000530
      A(I,J,NSP)=SPREF(2)                                               PR000540
      GO TO 2                                                           PR000550
    3 A(I,J,NRO)=ROREF(1)                                               PR000560
      A(I,J,NTC)=TCREF(1)                                               PR000570
      A(I,J,NSP)=SPREF(1)                                               PR000580
      A(I,J,NS)=S(1)                                                    PR000590
      A(I,J,NMU)=ZMUREF(1)                                              PR000600
    2 CONTINUE                                                          PR000610
    1 CONTINUE                                                          PR000620
      DO 10 I=I1,I3                                                     PR000630
      DO 10 J=1,JN                                                      PR000640
      IF(I.LT.IW(J).AND.J.LT.J1) GO TO 10                               PR000650
      A(I,J,NS)=S(2)                                                    PR000660
   10 CONTINUE                                                          PR000670
      IF(L.EQ.1.OR.L.EQ.2) GO TO 5                                      PR000680
C *** RECALCULATE VE,VC                                                 PR000690
      HRM=HLEH                                                          PR000700
      HCON=SPREF(1)*(TLM-TA)+HL                                         PR000710
      AMELT=HRM/HCON                                                    PR000720
      Z=VE                                                              PR000730
      VE=AMELT/(R(J1)*R(J1)*ROREF(1)*3.142)                             PR000740
      VC=VE*R(J1)*R(J1)/(R(JN)*R(JN))                                   PR000750
      VC=VC*ROREF(1)/ROREF(5)                                           PR000760
      HCD=HCD*VE/Z                                                      PR000770
    5 CONTINUE                                                          PR000780
      RETURN                                                            PR000790
      END                                                               PR000800
```

```
      SUBROUTINE SORCE(SOURCE,I,J,K)                                    SOR00010
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$SOR00020
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT          SOR00030
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)                SOR00040
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1                  SOR00050
      COMMON/CDIM/X(5),RE,RM                                            SOR00060
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC                        SOR00070
      COMMON/CCONT/C1,C2,C3,CD                                          SOR00080
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)                              SOR00090
      COMMON/CEMP/WF,P,S(4)                                             SOR00100
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS                     SOR00110
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL            SOR00120
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW                                 SOR00130
      COMMON/CTRVL/VE,VC                                                SOR00140
      COMMON/CHEAT/HC,HW(4)                                             SOR00150
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA                       SOR00160
      COMMON/CVF/FES(21),FEM(21),FSM,FSE                                SOR00170
      COMMON/CDROP/TAU,HCD,D,QS,QM                                      SOR00180
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)    SOR00190
     1,CAPPA,E                                                          SOR00200
      COMMON/CFLD/CU                                                    SOR00210
      COMMON/CPI/PI                                                     SOR00220
      COMMON/CTYP/LTYP                                                  SOR00230
      COMMON/CANG/THETA                                                 SOR00240
      COMMON/FITER/MIT,MPRINT,CP                                        SOR00250
      COMMON/CDVAR/A(31,21,20)                                          SOR00260
      COMMON/CNAME/ANAME,ASYMBL                                         SOR00270
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA    SOR00280
      COMMON/CLF/LF,LI                                                  SOR00290
      COMMON/CSVOR/SVOR(31,21)                                          SOR00300
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI                           SOR00310
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)               SOR00320
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)                             SOR00330
      COMMON/CAVM/AVM(60)                                               SOR00340
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR                        SOR00350
      COMMON/CSUR/HS(2),TS(2)                                           SOR00360
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP                                    SOR00370
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5                              SOR00380
      COMMON/CAF/FUN(20)                                                SOR00390
      COMMON/CVOLT/VOLT                                                 SOR00400
      COMMON/CAVT/TB                                                    SOR00410
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)                SOR00420
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$SOR00430
      COMMON/CPROD/GK(31,21)                                            SOR00440
      COMMON/CSHEAR/GKK(31,31)                                          SOR00450
```

```
      COMMON/CLREYN/LREYN,CDMOD(31,31)                                  SOR00460
C        BRANCH TO DIFFERENT SOURCE POINTS                              SOR00470
C        ---------------------------------                              SOR00480
                   GO TO (1,2,3,4,5,6,7,8,8),K                          SOR00490
C---------------------------------------------------------------------- SOR00500
C               DUMMY SOURCES                                           SOR00510
C               -------------                                           SOR00520
  8   SOURCE=0.0                                                        SOR00530
      RETURN                                                            SOR00540
C---------------------------------------------------------------------  SOR00550
C                FOR RE(H)/R                                            SOR00560
C                -----------                                            SOR00570
  1   SOURCE=WF*P*A(I,J,NHI)                                            SOR00580
      RETURN                                                            SOR00590
C .............. FOR IM(H)/R                                            SOR00600
C                -----------                                            SOR00610
  2   SOURCE=-WF*P*A(I,J,NHR)                                           SOR00620
      RETURN                                                            SOR00630
C---------------------------------------------------------------------  SOR00640
C..............FOR VORTICITY/R                                          SOR00650
C                -------------                                          SOR00660
  3   SOURCE=A(I,J,NHR)*A(I,J,NJRR)+A(I,J,NHI)*A(I,J,NJRI)              SOR00670
      SOURCE=(1.0E+06)*P*R(J)*SOURCE                                    SOR00680
      SVOR(I,J)=SOURCE                                                  SOR00690
      RETURN                                                            SOR00700
C---------------------------------------------------------------------  SOR00710
C...............STREAM FUNCTION                                         SOR00720
C               ---------------                                         SOR00730
    4 SOURCE=A(I,J,NW)                                                  SOR00740
      RETURN                                                            SOR00750
C---------------------------------------------------------------------- SOR00760
C...............TURBULENCE ENERGY                                       SOR00770
C               -----------------                                       SOR00780
    5 CONTINUE                                                          SOR00790
      B1=ADF(I,J,1,NV1)                                                 SOR00800
      B2=ADF(I,J,2,NV2)                                                 SOR00810
      B3=ADF(I,J,2,NV1)                                                 SOR00820
      B4=ADF(I,J,1,NV2)                                                 SOR00830
      B5=A(I,J,NV2)/R(J)                                                SOR00840
      SHEAR2=(2.0*(B1**2+B2**2+B5**2)+(B3+B4)**2)                       SOR00850
      GKK(I,J)=SHEAR2                                                   SOR00860
      GK(I,J)=A(I,J,NMT) * SHEAR2                                       SOR00870
      DK=A(I,J,NRO)*A(I,J,NEP)                                          SOR00880
      SOURCE=GK(I,J)-DK                                                 SOR00890
      RETURN                                                            SOR00900
```

```
C------------------------------------------------------------------
C             TURBULENT ENERGY DISSIPATION
C             -----------------------------
    6 CONTINUE
      IF(A(I,J,NK).EQ.0.0) GO TO 100
      TERM1=0.0
      TERM2=C1*GK(I,J)*A(I,J,NEP)/A(I,J,NK)
         F2=1.0
         IF(LREYN.EQ.1) REYLOC=A(I,J,NRO)*A(I,J,NK)**2/(ZMUREF(2)
     1*A(I,J,NEP))
         IF(LREYN.EQ.1.AND.REYLOC.LE.10.0) F2=1.0 -.3*EXP(-REYLOC**2)
         C2MOD=C2   *F2
      TERM3=C2MOD*A(I,J,NRO)*(A(I,J,NEP)**2)/A(I,J,NK)
      SOURCE=TERM1+TERM2-TERM3
      RETURN
  100 SOURCE=0.0
      RETURN
C---------------------------------------------------------------------
C              TEMPERATURE
C              -----------
   7  CONTINUE
      IF(J.GT.J1) GO TO 9
      Z=X1(I3)-X1(I2)
      SOURCE=A(I,J,NJJ)-QS/(3.14*R(J1)*R(J1)*Z)
      GO TO 10
    9 SOURCE=A(I,J,NJJ)
  10   CONTINUE
C---------------------------------------------------------------------
      RETURN
      END
```

```
      SUBROUTINE STRFUN
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1
      COMMON/CDIM/X(5),RE,RM
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC
      COMMON/CCONT/C1,C2,C3,CD
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)
      COMMON/CEMP/WF,P,S(4)
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW
      COMMON/CTRVL/VE,VC
      COMMON/CHEAT/HC,HW(4)
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA
      COMMON/CVF/FES(21),FEM(21),FSM,FSE
      COMMON/CDROP/TAU,HCD,D,QS,QM
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)
     1,CAPPA,E
      COMMON/CFLD/CU
      COMMON/CPI/PI
      COMMON/CTYP/LTYP
      COMMON/CANG/THETA
      COMMON/FITER/MIT,MPRINT,CP
      COMMON/CDVAR/A(31,21,20)
      COMMON/CNAME/ANAME,ASYMBL
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA
      COMMON/CLF/LF,LI
      COMMON/CSVOR/SVOR(31,21)
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)
      COMMON/CAVM/AVM(60)
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR
      COMMON/CSUR/HS(2),TS(2)
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP
      COMMON/CSORSE/SOURCE,SPRIME
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5
      COMMON/CAF/FUN(20)
      COMMON/CVOLT/VOLT
      COMMON/CAVT/TB
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
      Z2=0.
```

```
      DO 22 I=1,I3                                                      STR00460
      DO 22 J=2,JNM                                                     STR00470
      IF(ABS(Z2).LT.ABS(A(I,J,NF))) Z2=A(I,J,NF)                        STR00480
  22  CONTINUE                                                          STR00490
      DO 21 J=2,JNM                                                     STR00500
      IL=I1+1                                                           STR00510
      IF(J.LE.J1) IL=IW(J)+1                                            STR00520
      IH=I3-1                                                           STR00530
      DO 21 I=IL,IH                                                     STR00540
      RISQ=1./R(J)/R(J)                                                 STR00550
      ROP=A(I,J,NRO)                                                    STR00560
      BBE=4./(A(I+1,J,NRO)+ROP)*RISQ*BE(I)                              STR00570
      BBW=4./(A(I-1,J,NRO)+ROP)*RISQ*BW(I)                              STR00580
      BBN=16./(A(I,J+1,NRO)+ROP)/((R(J+1)+R(J))**2)*BN(J)               STR00590
      BBS=16./(A(I,J-1,NRO)+ROP)/((R(J-1)+R(J))**2)*BS(J)               STR00600
      CALL SORCE(SOURCE,I,J,NF)                                         STR00610
C     CALL CONVEC(AE,AW,AN,AS,I,J,NF)                                   STR00620
      ANUM=BBE*A(I+1,J,NF)+BBW*A(I-1,J,NF)+BBN*A(I,J+1,NF)              STR00630
     1+BBS*A(I,J-1,NF)+SOURCE                                           STR00640
      ADNM=BBE+BBW+BBN+BBS                                              STR00650
      IF(ADNM.EQ.0.)  GO TO 5                                           STR00660
      Z=A(I,J,NF)                                                       STR00670
      A(I,J,NF)=ANUM/ADNM                                               STR00680
      IF(Z.EQ.0.0.AND.A(I,J,NF).EQ.0.0)  GO TO 8                        STR00690
      RS=1.-Z/A(I,J,NF)                                                 STR00700
      GO TO 6                                                           STR00710
    8 RS=0.0                                                            STR00720
    6 CONTINUE                                                          STR00730
      GOSA=GOSA+ABS(Z-A(I,J,NF))                                        STR00740
      TOTA=TOTA+ABS(A(I,J,NF))                                          STR00750
      Z1=RP(NF)                                                         STR00760
      IF(J.GE.J1) Z1=0.2                                                STR00770
      A(I,J,NF)=Z+Z1*(A(I,J,NF)-Z)                                      STR00780
      IF(ABS(RS).LE.ABS(RSDU(NF))) GO TO 5                              STR00790
      RSDU(NF)=RS                                                       STR00800
      NFJ=J                                                             STR00810
      NFI=I                                                             STR00820
    5 CONTINUE                                                          STR00830
   21 CONTINUE                                                          STR00840
      RETURN                                                            STR00850
      END                                                               STR00860
```

```
      SUBROUTINE TURVAR(K)                                              TUR00010
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$ TUR00020
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT          TUR00030
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)                TUR00040
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1                  TUR00050
      COMMON/CDIM/X(5),RE,RM                                            TUR00060
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC                        TUR00070
      COMMON/CCONT/C1,C2,C3,CD                                          TUR00080
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)                              TUR00090
      COMMON/CEMP/WF,P,S(4)                                             TUR00100
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS                     TUR00110
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL            TUR00120
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW                                 TUR00130
      COMMON/CTRVL/VE,VC                                                TUR00140
      COMMON/CHEAT/HC,HW(4)                                             TUR00150
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA                       TUR00160
      COMMON/CVF/FES(21),FEM(21),FSM,FSE                                TUR00170
      COMMON/CDROP/TAU,HCD,D,QS,QM                                      TUR00180
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)    TUR00190
     1,CAPPA,E                                                          TUR00200
      COMMON/CFLD/CU                                                    TUR00210
      COMMON/CPI/PI                                                     TUR00220
      COMMON/CANG/THETA                                                 TUR00230
      COMMON/FITER/MIT,MPRINT,CP                                        TUR00240
      COMMON/CDVAR/A(31,21,20)                                          TUR00250
      COMMON/CNAME/ANAME,ASYMBL                                         TUR00260
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA    TUR00270
      COMMON/CLF/LF,LI                                                  TUR00280
      COMMON/CTYP/LTYP                                                  TUR00290
      COMMON/CSVOR/SVOR(31,21)                                          TUR00300
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI                           TUR00310
      COMMON/CSNW/SN1(20),SN2(15)                                       TUR00320
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)                             TUR00330
      COMMON/CAVM/AVM(60)                                               TUR00340
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR                        TUR00350
      COMMON/CSUR/HS(2),TS(2)                                           TUR00360
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP                                    TUR00370
      COMMON/CSORSE/SOURCE,SPRIME                                       TUR00380
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5                              TUR00390
      COMMON/CAF/FUN(20)                                                TUR00400
      COMMON/CVOLT/VOLT                                                 TUR00410
      COMMON/CAVT/TB                                                    TUR00420
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)                TUR00430
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$ TUR00440
      COMMON/CPROD/GK(31,21)                                            TUR00450
```

```
      COMMON/CSHEAR/GKK(31,31)
      DATA NBAL/0/
C
C.........FIRST GET THE CURRENT VALUE OF XI BASED ON OLD K.ENERGY FIELD.
C                       -------------------
                             IF(NSCHEM.EQ.0) GO TO 800
                             XIOLD=XI
                             CALL CONLOT(XI,PERFIN,PCASC)
                             XINEW=XI
                             XI=XIOLD+RPXI*(XINEW-XIOLD)
                             IF(XIOLD.EQ.0.0) XI=XINEW
C                  XI=050.00
C-----------------------------------------------------------------------
C
C..............START GRID POINT CALCULATIONS
C                    -----------------------
800                          CONTINUE
      Z2=0.
      DO 23  I=1,I3
      DO 23 J=2,JNM
      IF(ABS(Z2).LT.ABS(A(I,J,K))) Z2=A(I,J,K)
  23  CONTINUE
                             ENSUM=0.0
                  GENSUM=0.0
                 DISSUM=0.0
      DO 31 J=2,JNM
      IL=I1+1
      IF(J.LE.J1) IL=IW(J)+1
      IH=I3-1
      DO  31  I=IL,IH
      BPP=A(I,J,NMU)
      BBE=(A(I+1,J,NMU)+BPP)/PR(K)*BE(I)
      BBW=(A(I-1,J,NMU)+BPP)/PR(K)*BW(I)
      BBN=(A(I,J+1,NMU)+BPP)/PR(K)*BN(J)
      BBS=(A(I,J-1,NMU)+BPP)/PR(K)*BS(J)
36    SPRIME=0.
      CALL  SORCE(SOURCE,I,J,K)
      CALL  CONVEC(AE,AW,AN,AS,I,J,K)
      ANUM=(AE+BBE)*A(I+1,J,K)+(AW+BBW)*A(I-1,J,K)+(AN+BBN)*A(I,J+1,K)
     1+(AS+BBS)*A(I,J-1,K)+SOURCE
      ADNM=AE+AW+AN+AS+BBE+BBW+BBN+BBS+SPRIME
      IF(ADNM.EQ.0.) GO TO 6
      Z=A(I,J,K)
      A(I,J,K)=ANUM/ADNM
      IF(Z.EQ.0.0.AND.A(I,J, K).EQ.0.0)  GO TO 3
```

```
      RS=1.-Z/A(I,J,K)
      GO TO 5
    3 RS=0.0
    5 CONTINUE
      GOSA=GOSA+ABS(Z-A(I,J,K))
      TOTA=TOTA+ABS(A(I,J,K))
      Z1=RP(K)
C     IF(J.GE.10) Z1=0.2
      A(I,J,K)=Z+RP(K)*(A(I,J,K)-Z)
      IF(A(I,J, K).LE.0.0)  A(I,J, K)=0.0
                   IF(NBAL.EQ.0) GO TO 100
C....              CALCULATE ENERGY BUDGET
C                  -----------------------
C                  SPOINT=SOURCE-SPRIME*A(I,J,NK)
C                  DR=(R(J+1)-R(J-1))/2.0
C                  DX=(X1(I+1)-X1(I-1))/2.0
C                  ENSUM=ENSUM+SPOINT*R(J)*DR*DX
C                  GENSUM=GENSUM+GK(I,J)*R(J)*DR*DX
C                  DISSUM=DISSUM+A(I,J,NEP)*A(I,J,NRO)*R(J)*DR*DX
C-----------------------------------------------------------------------
 100               CONTINUE
      IF(ABS(RS).LT.ABS(RSDU(K)))  GO TO 6
      RSDU(K)=RS
      IF(K.EQ.NEP) GO TO 7
      NKI=I
      NKJ=J
      GO TO 6
 7    NEI=I
      NEJ=J
    6 CONTINUE
   31 CONTINUE
           IF(NBAL.EQ.0) RETURN
C          CALL VOL(GK,GENVOL)
C... PRINT OUT THE ENERGY BUDGET
C    ---------------------------
C                  IF(K.EQ.NK)WRITE(6,200) ENSUM,GENSUM,DISSUM,GENVOL
C200               FORMAT('ENERGY BALANCE=    ',1PE10.2
C    1/'GENERATION =  ',1PE10.2
C    2/'DISSIPATION=  ',1PE10.2
C    3/'GENERATION #2 ',1PE10.2)
C-----------------------------------------------------------------------
      RETURN
      END
```

```
      SUBROUTINE VELDIS                                                 VEL00010
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$VEL00020
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT          VEL00030
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)                VEL00040
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1                  VEL00050
      COMMON/CDIM/X(5),RE,RM                                            VEL00060
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC                        VEL00070
      COMMON/CCONT/C1,C2,C3,CD                                          VEL00080
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)                              VEL00090
      COMMON/CEMP/WF,P,S(4)                                             VEL00100
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS                     VEL00110
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL            VEL00120
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW                                 VEL00130
      COMMON/CTRVL/VE,VC                                                VEL00140
      COMMON/CHEAT/HC,HW(4)                                             VEL00150
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA                       VEL00160
      COMMON/CVF/FES(21),FEM(21),FSM,FSE                                VEL00170
      COMMON/CDROP/TAU,HCD,D,QS,QM                                      VEL00180
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)    VEL00190
     1,CAPPA,E                                                          VEL00200
      COMMON/CFLD/CU                                                    VEL00210
      COMMON/CPI/PI                                                     VEL00220
      COMMON/CTYP/LTYP                                                  VEL00230
      COMMON/CANG/THETA                                                 VEL00240
      COMMON/FITER/MIT,MPRINT,CP                                        VEL00250
      COMMON/CDVAR/A(31,21,20)                                          VEL00260
      COMMON/CNAME/ANAME,ASYMBL                                         VEL00270
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA    VEL00280
      COMMON/CLF/LF,LI                                                  VEL00290
      COMMON/CSVOR/SVOR(31,21)                                          VEL00300
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI                           VEL00310
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)               VEL00320
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)                             VEL00330
      COMMON/CAVM/AVM(60)                                               VEL00340
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR                        VEL00350
      COMMON/CSUR/HS(2),TS(2)                                           VEL00360
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP                                    VEL00370
      COMMON/CSORSE/SOURCE,SPRIME                                       VEL00380
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5                              VEL00390
      COMMON/CAF/FUN(20)                                                VEL00400
      COMMON/CVOLT/VOLT                                                 VEL00410
      COMMON/CAVT/TB                                                    VEL00420
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)                VEL00430
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$VEL00440
      DO 50 J=2,JNM                                                     VEL00450
```

```
      IL=I1+1                                                           VEL00460
      IF(J.LE.J1) IL=IW(J)+1                                            VEL00470
      IH=I3-1                                                           VEL00480
      DO 51 I=IL,IH                                                     VEL00490
      A(I,J,NV1)=ADF(I,J,2,NF)/R(J)/A(I,J,NRO)                          VEL00500
      A(I,J,NV2)=-ADF(I,J,1,NF)/R(J)/A(I,J,NRO)                         VEL00510
  51  CONTINUE                                                          VEL00520
   50 CONTINUE                                                          VEL00530
      IL=IW(1)+1                                                        VEL00540
      IH=I3-1                                                           VEL00550
      RR=R(2)*R(2)/(R(3)*R(3))                                          VEL00560
      DO 54 I=IL,IH                                                     VEL00570
      A(I,1,NV1)=2.0*(A(I,2,NF)-RR*RR*A(I,3,NF))                        VEL00580
     1/(R(2)*R(2)*(1.-RR)*A(I,1,NRO))                                   VEL00590
  54  CONTINUE                                                          VEL00600
      J12=J1+1                                                          VEL00610
      DO 40 J=J12,JNM                                                   VEL00620
      A(I1,J,NV2)=-ADF(I1,J,1,NF)/R(J)/ROREF(2)                         VEL00630
  40  CONTINUE                                                          VEL00640
      RETURN                                                            VEL00650
      END                                                               VEL00660
```

```
      SUBROUTINE VISCOS
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1
      COMMON/CDIM/X(5),RE,RM
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC
      COMMON/CCONT/C1,C2,C3,CD
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)
      COMMON/CEMP/WF,P,S(4)
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW
      COMMON/CTRVL/VE,VC
      COMMON/CHEAT/HC,HW(4)
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA
      COMMON/CVF/FES(21),FEM(21),FSM,FSE
      COMMON/CDROP/TAU,HCD,D,QS,QM
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)
     1,CAPPA,E
      COMMON/CFLD/CU
      COMMON/CPI/PI
      COMMON/CANG/THETA
      COMMON/FITER/MIT,MPRINT,CP
      COMMON/CDVAR/A(31,21,20)
      COMMON/CNAME/ANAME,ASYMBL
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA
      COMMON/CLF/LF,LI
      COMMON/CTYP/LTYP
      COMMON/CSVOR/SVOR(31,21)
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI
      COMMON/CSNW/SN1(20),SN2(15)
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)
      COMMON/CAVM/AVM(60)
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR
      COMMON/CSUR/HS(2),TS(2)
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP
      COMMON/CSORSE/SOURCE,SPRIME
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5
      COMMON/CAF/FUN(20)
      COMMON/CVOLT/VOLT
      COMMON/CAVT/TB
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
      COMMON/CLREYN/LREYN,CDMOD(31,31)
```

```
C*** CALCULATE TURBULENT VISCOSITY                                      VISOO460
      I31=I3-1                                                          VISOO470
      DO 10 J=1,JNM                                                     VISOO480
      DO  11 I=I1,I31                                                   VISOO490
      IF(J.LE.J1.AND.I.LE.IW(J)) GO TO 12                               VISOO500
      IF(A(I,J,NEP).LE.O.O.OR.A(I,J,NK).LE.O.O) GO TO 20                VISOO510
      A(I,J,NMT) =A(I,J,NRO)*CDMOD(I,J)*(A(I,J,NK)**2)/A(I,J,NEP)       VISOO520
      GO TO 21                                                          VISOO530
   20 CONTINUE                                                          VISOO540
      A(I,J,NMT)=O.                                                     VISOO550
   21 CONTINUE                                                          VISOO560
      Z=A(I,J,NMU)                                                      VISOO570
      EMU=ZMUREF(1)                                                     VISOO580
      A(I,J,NMU)=EMU    +A(I,J,NMT)                                     VISOO590
      A(I,J,NMU)=Z+RP(NMU)*(A(I,J,NMU)-Z)                               VISOO600
  12  CONTINUE                                                          VISOO610
  11  CONTINUE                                                          VISOO620
   10 CONTINUE                                                          VISOO630
      RETURN                                                            VISOO640
      END                                                               VISOO650
```

```
      SUBROUTINE VOL(P,VAL)
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1
      COMMON/CDIM/X(5),RE,RM
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC
      COMMON/CCONT/C1,C2,C3,CD
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW
      COMMON/CTRVL/VE,VC
      COMMON/CHEAT/HC,HW(4)
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA
      COMMON/CVF/FES(21),FEM(21),FSM,FSE
      COMMON/CDROP/TAU,HCD,D,QS,QM
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)
     1,CAPPA,E
      COMMON/CFLD/CU
      COMMON/CPI/PI
      COMMON/CANG/THETA
      COMMON/FITER/MIT,MPRINT,CP
      COMMON/CDVAR/A(31,21,20)
      COMMON/CNAME/ANAME,ASYMBL
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA
      COMMON/CLF/LF,LI
      COMMON/CTYP/LTYP
      COMMON/CSVOR/SVOR(31,21)
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI
      COMMON/CSNW/SN1(20),SN2(15)
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)
      COMMON/CAVM/AVM(60)
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR
      COMMON/CSUR/HS(2),TS(2)
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP
      COMMON/CSORSE/SOURCE,SPRIME
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5
      COMMON/CAF/FUN(20)
      COMMON/CVOLT/VOLT
      COMMON/CAVT/TB
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
      DIMENSION P(31,21)
C...  ROUTINE TO CALCULATE THE VOLUME INTEGRAL OF FUNCTION P
```

```
C     TO CALCULATE THE LOTSIANSKII CONSTANT
      VAL=0.0
      DO 100 J=2,JNM
      IL=IW(J)+1
      DO 100 I=IL,INM
      DR=(R(J+1)-R(J-1))/2.0
      DX=(X1(I+1)-X1(I-1))/2.0
      DV=R(J)*DR*DX
          IF(I.EQ.IL.OR.I.EQ.IH)DV=1.5*DV
          IF(J.EQ.JNM) DV=1.5*DV
      VAL=VAL+P(I,J)*DV
100   CONTINUE
                   DO 150 I=IL,IH
      VAL=VAL  + P(I,1)*R(1)**2*(X1(I+1)-X1(I-1))/4.0
150                CONTINUE
      RETURN
      END
```

```
      SUBROUTINE VORITY                                                 VCR00010
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)                             VOR00020
      COMMON/CDVAR/A(31,21,20)                                          VOR00030
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT          VOR00040
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)                VOR00050
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA    VOR00060
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)                              VOR00070
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1                  VOR00080
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL            VOR00090
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP                                    VOR00100
      COMMON/CSORSE/SOURCE,SPRIME                                       VOR00110
C----------------------------------------------------------------------|VOR00120
C           SOLUTION OF VORTICITY TRANSPORT EQUATION                   |VOR00130
C----------------------------------------------------------------------|VOR00140
      IL=I1+1                                                           VOR00150
C     IF(J.LE.J1) IL=IW(J)+1                                            VOR00160
      IH=I3-1                                                           VOR00170
      DO  11  I=IL,IH                                                   VOR00180
      DO 11 J=2,JNM                                                     VOR00190
      RSQ=R(J)*R(J)                                                     VOR00200
      BBE=2.*RSQ*BE(I)                                                  VOR00210
      BBW=2.*RSQ*BW(I)                                                  VOR00220
      BBN=(R(J+1)*R(J+1)+RSQ)*BN(J)                                     VOR00230
      BBS=(R(J-1)*R(J-1)+RSQ)*BS(J)                                     VOR00240
      SOURCE=0.0                                                        VOR00250
      CALL  CONVEC(AE,AW,AN,AS,I,J,NW)                                  VOR00260
C...................... IMPLICIT VORTICITY CALCULATIONS                 VOR00270
      TERM1=A(I+1,J,NW)                                                 VOR00280
      TERM2=A(I-1,J,NW)                                                 VOR00290
      TERM3=A(I,J-1,NW)                                                 VOR00300
      TERM4=A(I,J+1,NW)                                                 VOR00310
      TERM5=0.0                                                         VOR00320
      IF(I.NE.(I1+1)) GO TO 14                                          VOR00330
      DX12=(X1(I1+1)-X1(I1))**2                                         VOR00340
C..................ALLOW FOR COMPLETE INFLOW WALL CONDITIONS            VOR00350
      TERM2=TERM2+                                                      VOR00360
     13.*(A(I1,J,NF)-A(I1+1,J,NF))/(DX12*RSQ*A(I1,J,NRO))               VOR00370
      TERM5=0.5*(AW+A(I-1,J,NMU)*BBW)                                   VOR00380
C..................CALCULATE NEW VORTICITY...........................   VOR00390
 14                CONTINUE                                             VOR00400
      ANUM=(AE+A(I+1,J,NMU)*BBE)*TERM1                                  VOR00410
     1+(AW+A(I-1,J,NMU)*BBW)*TERM2                                      VOR00420
     2+(AS+A(I,J-1,NMU)*BBS)*TERM3                                      VOR00430
     3+(AN+A(I,J+1,NMU)*BBN)*TERM4      +SOURCE                         VOR00440
      ADNM=AE+AW+AN+AS+A(I,J,NMU)*(BBE+BBW+BBN+BBS)+TERM5               VOR00450
```

```
      IF(ADNM.EQ.0.)  GO TO 4                                           VOR00460
      Z=A(I,J,NW)                                                       VOR00470
      A(I,J,NW)=ANUM/ADNM                                               VOR00480
C................RELAXATION AND RESIDUE CALCULATIONS                    VOR00490
      IF(Z.EQ.0.0.AND.A(I,J,NW).EQ.0.0)  GO TO 1                        VOR00500
      RS=1.-Z/A(I,J,NW)                                                 VOR00510
      GO TO 3                                                           VOR00520
    1 RS=0.                                                             VOR00530
    3 CONTINUE                                                          VOR00540
      GOSA=GOSA+ABS(Z-A(I,J,NW))                                        VOR00550
      TOTA=TOTA+ABS(A(I,J,NW))                                          VOR00560
C.....................................................................   VOR00570
      Z1=RP(NW)                                                         VOR00580
C     IF(J.GE.J1) Z1=0.1                                                VOR00590
      A(I,J,NW)=Z+Z1*(A(I,J,NW)-Z)                                      VOR00600
C....................................................................    VOR00610
      IF(ABS(RS).LE.ABS(RSDU(NW))) GO TO 4                              VOR00620
      RSDU(NW)=RS                                                       VOR00630
      NWJ=J                                                             VOR00640
      NWI=I                                                             VOR00650
    4 CONTINUE                                                          VOR00660
C..............DIAGNOSTIC AIDS........................................   VOR00670
        WRITE(15,997) AE,AW,AN,AS,TERM1,TERM2,TERM3,TERM4               VOR00680
 997    FORMAT(8(1PE10.2))                                              VOR00690
         WRITE(15,998)I,J,A(I,J,NW)                                     VOR00700
 998     FORMAT(2I5,1PE10.2)                                            VOR00710
         WRITE(15,998) I1,J,A(I1,J,NF)                                  VOR00720
C....................................................................    VOR00730
   11 CONTINUE                                                          VOR00740
      RETURN                                                            VOR00750
      END                                                               VOR00760
```

```
      SUBROUTINE WALL                                                   WAL00010
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$ WAL00020
      COMMON/CNUMB/NW,NF,NK,NEP,NV1,NV2,NT,NMU,NRO,NTC,NSP,NMT          WAL00030
      COMMON/CCORD/IMIN(21),IMAX(21),X1(31),X2(21),R(21)                WAL00040
      COMMON/CGRID/IN,INM,JN,JNM,I1,I2,I3,IA,IW(21),J1                  WAL00050
      COMMON/CDIM/X(5),RE,RM                                            WAL00060
      COMMON/CITER/NMAX,NPRINT,NITER,IP,IE,IV,CC                        WAL00070
      COMMON/CCONT/C1,C2,C3,CD                                          WAL00080
      COMMON/CRLAX/RP(10),RSDU(10),SAN(10)                              WAL00090
      COMMON/CEMP/WF,P,S(4)                                             WAL00100
      COMMON/CHJ/NHR,NHI,NJRR,NJRI,NJZR,NJZI,NJJ,NS                     WAL00110
      COMMON/CTHERM/TCREF(5),SPREF(5),BETA(2),ES,EE,EM,SB,HL            WAL00120
      COMMON/CTEMP/TLM,TSM,TLS,TA,TM,TW                                 WAL00130
      COMMON/CTRVL/VE,VC                                                WAL00140
      COMMON/CHEAT/HC,HW(4)                                             WAL00150
      COMMON/CPROP/ROREF(5),ZMUREF(2),PR(10),GAMA                       WAL00160
      COMMON/CVF/FES(21),FEM(21),FSM,FSE                                WAL00170
      COMMON/CDROP/TAU,HCD,D,QS,QM                                      WAL00180
      COMMON/CTAUW/TAUW1(20),TAUW2(15),TAUW3(21),TAUW4(10),TAUW5(21)    WAL00190
     1,CAPPA,E                                                          WAL00200
      COMMON/CFLD/CU                                                    WAL00210
      COMMON/CPI/PI                                                     WAL00220
      COMMON/CTYP/LTYP                                                  WAL00230
      COMMON/CANG/THETA                                                 WAL00240
      COMMON/FITER/MIT,MPRINT,CP                                        WAL00250
      COMMON/CDVAR/A(31,21,20)                                          WAL00260
      COMMON/CNAME/ANAME,ASYMBL                                         WAL00270
      COMMON/CCRIT/NWI,NWJ,NFI,NFJ,NTI,NTJ,NKI,NKJ,NEI,NEJ,GOSA,TOTA    WAL00280
      COMMON/CLF/LF,LI                                                  WAL00290
      COMMON/CSVOR/SVOR(31,21)                                          WAL00300
      COMMON/CTURB/NTURB,NSCHEM,NVISC,RPXI,XI                           WAL00310
      COMMON/CSNW/SN1(20),SN2(15),SN3(21),SN4(20),SN5(21)               WAL00320
      COMMON/CB/BE(31),BW(31),BN(21),BS(21)                             WAL00330
      COMMON/CAVM/AVM(60)                                               WAL00340
      COMMON/CHBL/HLW,HLEH,HLEV,HLS,HLD,HLM,POWR                        WAL00350
      COMMON/CSUR/HS(2),TS(2)                                           WAL00360
      COMMON/CBB/BBE,BBW,BBN,BBS,BPP                                    WAL00370
      COMMON/CSORSE/SOURCE,SPRIME                                       WAL00380
      COMMON/CGEOM/FR1,FR2(15),FR3,FR4,FR5                              WAL00390
      COMMON/CAF/FUN(20)                                                WAL00400
      COMMON/CVOLT/VOLT                                                 WAL00410
      COMMON/CAVT/TB                                                    WAL00420
      COMMON/CYP/YP1(20),YP2(15),YP3(21),YP4(10),YP5(21)                WAL00430
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$ WAL00440
      COMMON/CLREYN/LREYN,CDMOD(31,31)                                  WAL00450
```

```
      PRL=SPREF(2)*ZMUREF(1)/TCREF(2)                                   WAL00460
      RECPRT=1./PR(NT)                                                  WAL00470
      PRRAT=PRL*RECPRT                                                  WAL00480
      PJAY=9.*(PRRAT-1.)/(PRRAT**0.25)                                  WAL00490
      ANU=ZMUREF(1)/ROREF(2)                                            WAL00500
C     ***********  AT J=JN,TAUW1    , SN1                               WAL00510
      YP=X2(JN)-X2(JNM)                                                 WAL00520
      ZL=X1(I3)-X1(I1)                                                  WAL00530
      IL=I1+1                                                           WAL00540
      IH=I3-1                                                           WAL00550
      DO 1 I=IL,IH                                                      WAL00560
      N=I-IL+1                                                          WAL00570
      A1=ROREF(2)*CDMOD(I,J)**0.25                                      WAL00580
      A2=A1*SQRT(A(I,JNM,NK))                                           WAL00590
      YPP=YP*A2/ZMUREF(1)                                               WAL00600
      UP=ABS(A(I,JNM,NV1))                                              WAL00610
      IF(YPP.GT.11.5) GO TO 2                                           WAL00620
      TAUW1(N)=ZMUREF(1)*UP/YP                                          WAL00630
      GO TO 3                                                           WAL00640
    2 TAUW1(N)=CAPPA*UP*A2/ALOG(YPP*E)                                  WAL00650
    3 CONTINUE                                                          WAL00660
C     TAUGR=ROREF(2)*BETA(1)*9.81*(A(I,JNM,NT)-A(I,JN,NT))*YP/2.        WAL00670
      TAUGR=0.                                                          WAL00680
      TAUEM=A(I,JNM,NJRR)*A(I,JNM,NHR)+A(I,JNM,NJRI)*A(I,JNM,NHI)       WAL00690
      TAUEM=TAUEM*(1.0E+06)*P*R(J)*YP/4.                                WAL00700
      PN=0.                                                             WAL00710
C     IF(A(I,JNM,NV1).LT.0.) PN=-1.                                     WAL00720
      TAUW1(N)=ABS(TAUW1(N))+ABS(TAUGR)+ABS(TAUEM)*PN                   WAL00730
      TAUW1(N)=ABS(TAUW1(N))                                            WAL00740
      FRIC=TAUW1(N)/(UP*UP*ROREF(2))                                    WAL00750
   16 CONTINUE                                                          WAL00760
      TCS=TCREF(2)+0.5*SPREF(2)*A(I,JNM,NMT)                            WAL00770
      DTG=A(I,JNM,NJJ)*YP*YP/(2.*TCS)                                   WAL00780
      F1=1.                                                             WAL00790
      STAN=FRIC*RECPRT/(1.+PJAY*SQRT(FRIC))                             WAL00800
      SN1(N)=ABS(STAN)* ROREF(2)*SPREF(2)*UP*F1                         WAL00810
      YP1(N)=YPP                                                        WAL00820
    1 CONTINUE                                                          WAL00830
C ***               AT I=IW+1,1<J<J1                                    WAL00840
      DO 4 J=2,J1                                                       WAL00850
      II=IW(J)                                                          WAL00860
      YP=(X1(II+1)-X1(II))*COS(THETA)                                   WAL00870
      A1=ROREF(2)*CDMOD(I,J)**0.25                                      WAL00880
      A2=A1*SQRT(A(II+1,J,NK))                                          WAL00890
      YPP=YP*A2/ZMUREF(1)                                               WAL00900
```

```
      UP1=A(II+1,J,NV2)*COS(THETA)-A(II+1,J,NV1)*SIN(THETA)
      UP=ABS(UP1)
      IF(YPP.GT.11.5) GO TO 5
      TAUW2(J)=ZMUREF(1)*UP/YP
      GO TO 6
    5 TAUW2(J)=CAPPA*UP*A2/ALOG(YPP*E)
    6 CONTINUE
      TAUE1=A(II+1,J,NJZR)*A(II+1,J,NHR)+A(II+1,J,NJZI)*A(II+1,
     1J,NHI)
      TAUE2=A(II+1,J,NJRR)*A(II+1,J,NHR)+A(II+1,J,NJRI)*A(II+1,
     1J,NHI)
      TAUE=TAUE1*COS(THETA)+TAUE2*SIN(THETA)
      TAUEM=ABS(TAUE)
      TAUEM=TAUEM*(1.0E+06)*P*R(J)*YP/4.
      PN=0.
C     IF(UP1.GT.0.) PN=-1.
      TAUW2(J)=ABS(TAUW2(J))+ABS(TAUEM)*PN
      TAUW2(J)=ABS(TAUW2(J))
      FRIC=TAUW2(J)/(UP*UP*ROREF(2))
      TCS=TCREF(2)+0.5*SPREF(2)*A(I2+1,J,NMT)
      DTG=A(I2+1,J,NJJ)*YP*YP/(2.*TCS)
      F2=1.
      STAN=FRIC*RECPRT/(1.+PJAY*SQRT(FRIC))
      SN2(J)=ABS(STAN)*ROREF(2)*SPREF(2)*UP*F2
      YP2(J)=YPP
   40 CONTINUE
    4 CONTINUE
C  ***              AT I=I3-1,1<J<JN , TAUW3    ,SN3
      YP=X1(I3)-X1(I3-1)
      DO 7 J=2,JNM
      A2=A1*SQRT(A(I3-1,J,NK))
      YPP=YP*A2/ZMUREF(1)
      UP=ABS(A(I3-1,J,NV2))
      IF(YPP.GT.11.5) GO TO 8
      TAUW3(J)=ZMUREF(1)*UP/YP
      GO TO 9
    8 TAUW3(J)=CAPPA*UP*A2/ALOG(YPP*E)
    9 CONTINUE
      TAUEM=A(I3-1,J,NJZR)*A(I3-1,J,NHR)+A(I3-1,J,NJZI)*A(I3-1,
     1J,NHI)
      TAUEM=TAUEM*(1.0E+06)*P*R(J)*YP/4.
      PN=0.
C     IF(A(I3-1,J,NV2).GT.0.) PN=-1.
      TAUW3(J)=ABS(TAUW3(J))+ABS(TAUEM)*PN
      TAUW3(J)=ABS(TAUW3(J))
```

```
      FRIC=TAUW3(J)/(UP*UP*ROREF(2))                                    WALO1360
      DT=A(I3-1,J,NT)-A(I3,J,NT)                                        WALO1370
      IF(DT.LE.0.) DT=0.                                                WALO1380
  18  CONTINUE                                                          WALO1390
      TCS=TCREF(2)+A(I3-1,J,NMT)*SPREF(2)*0.5                           WALO1400
      DTG=A(I3-1,J,NJJ)*YP*YP/(2.*TCS)                                  WALO1410
      F3=1.                                                             WALO1420
      STAN=FRIC*RECPRT/(1.+PJAY*SQRT(FRIC))                             WALO1430
      SN3(J)=ABS(STAN)*ROREF(2)*SPREF(2)*UP*F3                          WALO1440
      YP3(J)=YPP                                                        WALO1450
   7  CONTINUE                                                          WALO1460
C ***              AT J=J1+1,I1<I<I2                                    WALO1470
      IF(I1.EQ.I2) GO TO 15                                             WALO1480
      YP=X2(J1+1)-X2(J1)                                                WALO1490
      IL=I1+1                                                           WALO1500
      I21=I2-1                                                          WALO1510
      DO 10 I=IL,I2                                                     WALO1520
      N=I-IL+1                                                          WALO1530
      A1=ROREF(2)*CDMOD(I,J)**0.25                                      WALO1540
      A2=A1*SQRT(A(I,J1+1,NK))                                          WALO1550
      YPP=YP*A2/ZMUREF(1)                                               WALO1560
      UP=ABS(A(I,J1+1,NV1))                                             WALO1570
      IF(YPP.GT.11.5) GO TO 11                                          WALO1580
      TAUW4(N)=ZMUREF(1)*UP/YP                                          WALO1590
      GO TO 12                                                          WALO1600
  11  TAUW4(N)=CAPPA*UP*A2/ALOG(YPP*E)                                  WALO1610
  12  CONTINUE                                                          WALO1620
      TAUEM=A(I,J1+1,NJRR)*A(I,J1+1,NHR)+A(I,J1+1,NJRI)*A(I,J1+        WALO1630
     11,NHI)                                                            WALO1640
      TAUEM=TAUEM*(1.0E+06)*P*R(J)*YP/4.                                WALO1650
      PN=0.                                                             WALO1660
      TAUW4(N)=ABS(TAUW4(N))+ABS(TAUEM)*PN                              WALO1670
      FRIC=TAUW4(N)/(UP*UP*ROREF(2))                                    WALO1680
      STAN=FRIC*RECPRT/(1.+PJAY*SQRT(FRIC))                             WALO1690
      SN4(N)=ABS(STAN)*ROREF(2)*SPREF(2)*UP                             WALO1700
      YP4(N)=YPP                                                        WALO1710
  10  CONTINUE                                                          WALO1720
  15  CONTINUE                                                          WALO1730
  20  CONTINUE                                                          WALO1740
      RETURN                                                            WALO1750
      END                                                               WALO1760
```

NOMENCLATURE

a	Radius of heated cylinder (m)
B	Magnetic Flux Intensity (Tesla)
C_1, C_2, C_d	Constants in the $\kappa - \epsilon$ Turbulence Model
D_c	Diffusivity of mixing species in the melt (m^2/sec)
D	Depth of melt (m)
E(k)	3-Dimensional Turbulence Energy Density Spectrum
E	Electric Field
f(r)	Spatial Autocorrelation Function for turbulence
$\mathbf{F}_{be}$	Electromagnetic Force field
g	Local Turbulence Energy generation term
G	Total Turbulence Energy generation over entire volume of melt
H	Magnetic Induction (Amps/m)
I	Total current (Amps)
I_L	Lotsianskii Integral in XI Model of Turbulence
J	Current Density (Amps/m^2)
k	Wavenumber (m^{-1})
k_e	Wavenumber of Energy Containing Eddies
l	Energy Containing length scale
L_v	Characteristic Vessel Dimension (m)
L_p	Characteristic size of Large Scale Eddies (m)
M	Mass of melt

N	Magnetic Interaction Parameter
p	pressure in the fluid
P	Stirring Power input to the system (Watts)
P_t	Specific Power input (Watts/Ton of Melt)
PI	Stands for Performance Index used to evaluate degree of Turbulence
q	RMS Turbulent Fluctuating Velocity
R_e	Radius of electrode (m)
R_m	Radius of vessel (m)
r	Radial location in the melt (m)
t	time (secs)
t_m	Mixing time for 95% homogenisation (sec)
T	Temperature
U	Mean Velocity (m/sec)
$u'_{i,j}$	Fluctuating velocity components
V	Mean Nondimensional Velocity
x	Axial Location in melt (m)

λ	Taylor Microscale
ℓ	Length scale for concentration fluctuations corresponding to λ
ϕ	Turbulence Dissipation rate per unit mass(W/Kg-sec)
Φ	Total Turbulence Dissipation Rate(W/sec)
σ	Electrical Conductivity (mho-m)
ρ	Density of the fluid (Kg/m ^{3})

μ_o	Magnetic Permeability of Free Space
η	Magnetic Diffusivity, Efficiency of Turbulence Generation
ν	Molecular Kinematic Viscosity of Fluid (m^2/sec)
μ	Molecular Viscosity of Fluid (Kg./m-sec)
α	Thermal Diffusivity of Fluid (m^2/sec)
ω	Vorticity
ψ	Stream Function
μ_{eff}	Turbulence Enhanced Viscosity (Kg/m-sec)
μ_t	Turbulent Contribution to Viscosity (Kg/m-sec)
κ	Kinetic Energy of Turbulent Fluctuations (W/Kg)
ϵ	Local Rate of Turbulence Energy Dissipation (W/Kg-sec)
σ_e	'Prandtl Number' for ϵ Transport
σ_k	'Prandtl Number' for κ Transport
ξ	Primary Variable in the XI Model of Turbulence
τ	Time constant for Mixing (66% mixed)
τ	Shear Stress Tensor

References

1. C. Trakas, P. Tabeling and J.P. Chabrerie; Proc. IUTAM Symposium -Metallurgical Applications of M.H.D., Cambridge, Sept. 1982.

2. R. Ricou and C. Vives; Int. J. Heat Mass Transfer, Vol.25 No.10 pp1579-1588, 1982.

3. A. Murthy and J. Szekely; Met. Trans. B, Vol. 14B, Sept. 1983- 499.

4. F. N. Frenkiel and P. S. Klebanoff; Phys. Fluids, Vol. 10, Aug 1967.

5. Thermo-Systems Inc., General System Information for 1050 Series Anemometry.

6. C. A. Sleicher and G. B. Lim, Proc. Symp. Turbulence in Liquids, Missouri-Rolla, 1973, 1-14 (1975).

7. R. A. Gardner, Flow: Measurement and Control in Science and Industry, Vol. 1, Part 2, Pittsburgh, p581 (1974).

8. G. B. Lim and C. A. Sleicher, Flow: Measurement and Control in Science and Industry, Vol. 1, Part 2, Pittsburgh, p563 (1974).

9. D. G. Malcolm, J. Fluid Mech., Vol. 37, Part 4, pp701-713, (1969).

10. J. Hill and C. A. Sleicher, Rev. Sci. Instr., Vol. 42, No. 10, Oct. 1971, p1461.

11. J. C. Hurt and J. R. Welty, ASME paper 72-HT-3 (1972).

12. M. Sajben, Rev. Sci. Instr., Vol. 36, No. 7, July 1965, p945.

13. M. Sajben, Sc.D Thesis, Depart. Mech. Engg., M.I.T., Aug. 1964.

14. J. C. Hill, Ph.D Thesis, Univ. of Washington, 1968.

15. D. G. Malcolm, Transl. Magnitnaya Gidrodinamika, Vol. 6, No. 2, pp 55-64, April-June 1970.

16. T. Robinson and K. Larsson, J. Fluid Mech., Vol. 60, Part 4, p641 (1973).

17. P. Lykoudis and P. Dunn, Intl. Jrnl. Heat Mass Transfer, Vol. 16, pp 1439-1452 (1973).

18. M. Hoff, Instr. and Control Systems, March 1969, p83.

Metallurgical Operations Modelling

19. D. J. Moore and J. C. R. Hunt, Proc. Third Beer-Sheva Symposium on M.H.D. Flows and Turbulence.

20. J. Szekely and S. Asai, Trans. ISIJ, Vol. 15, p270 (1975).

21. S. Asai and J. Szekely, Ironmaking and Steelmaking (Quart.), No. 3, 205 (1975).

22. J. Szekely and A. H. Dilawari, Proc. Fifth Intl. Conf. Vaccuum Metallurgy and Electroslag Melting Processes, Munich Oct. 1976.

23. J. Szekely, C. W. Chang and R. E. Ryan, Met. Trans. B, Vol. 8B, June 1977 -333.

24. J. Szekely and C. W. Chang, Ironmaking and Steelmaking, No. 3, p196 (1977).

25. J. Szekely and C. W. Chang, Ironmaking and Steelmaking, No. 3, p190 (1977).

26. J. A. Shercliffe, J. Fluid Mech., Vol. 40, Part 2, pp 241-250, (1970).

27. C. Souzou and W. M. Pickering, Proc. R. Soc. Lond., A.362, 509-523 (1978).

28. M. Yu. Abritska, A. E. Mikelson, V. N. Moshnyaga, Yu. V. Chernov and A. I. Scherbakov, Transl. Magnitnaya Gidrodinamika, No. 3,pp 105-110, July -Sept. 1979.

29. J. Kreyenberg and K. Schwerdtfeger, Arch. Eisenhuttenwes, 50, No. 1, January 1979.

30. A. H. Dilawari and J. Szekely, Met. Trans. B, Vol. 9B, pp 77, March 1978.

31. J. Szekely, M. Choudhary, S. Woodruff, ???

32. A. H. Dilawari and J. Szekely, Met. Trans. B, Vol. 8B, June 1977, pp 227.

33. A. H. Dilawari, J. Szekely, and T. W. Eagar, Met. Trans. B, Vol. 9B, Sept. 1978, pp371.

34. M. K. Choudhary and J. Szekely, Met. Trans. B, Vol. 11B, Sept. 1980, pp 439.

35. M. K. Choudhary and J. Szekely, Ironmaking and Steelmaking, 1981, No. 5, pp 225.

36. M. K. Choudhary, A Study of Heat Transfer and Fluid Flow in the E.S.R. Process, Sc.D Thesis, M.I.T.

37. M. K. Choudhary, J. Szekely, B. I. Medovar and Yu. G. Emelyanenko, Met. Trans. B, Vol. 13B, pp 35, March 1982.

38. A. Alemany, R. Moreau, P. L. Sulem, U. Frisch, Journal de Mecanique, Vol. 18, No. 2, 1979.

39. M. K. Choudhary and J. Szekely, Trans. Instn. Min. Metall. (Sect. C : Mineral Process. Extr. Metall.), 90, Dec. 1981.

40. Ya. Yu. Kompan, V. I. Sharamkin, E. V. Scherbinin and U. A. Lasis, Avt. Svarka, 1979, No. 3, pp 13-18.

41. I. E. Butsenieks, D. E. Peterson, V. I. Sharamkin, and E. V. Scherbinin, Transl. Magnitnaya Gidrodinamika, No. 1, pp 92-97, Jan.-March, 1976.

42. V. N. Moshnyaga and V. I. Sharamkin, Transl. Magn. Gid., No. 1, pp 77-80, Jan.-March 1980.

43. I. E. Butsenieks, Ya. Yu. Kompan, V. I. Sharamkin, E. I. Shilova and E. V. Scherbinin, Transl. Mag. Gid. No. 3, pp 143-148, July-Sept. 1975.

M.H.D. Turbulence

44. Lehnert, Quart. Appl. Math. (1954), 12, 321

45. Schumann, J. Fluid Mech., (1976), Part 1, pp 31-58, 74

46. Moffat, J. Fluid Mech., (1967), Vol. 28, Part 3, pp 571-592.

47. G. Rudiger, Transl. Mag. Gid. No. 1, pp 3-14, Jan.-March 1980.

48. Gelfgat, Kit, Peterson, and Tsinober, Magneto Hydro Dynamics, July-Sept. 1971, Vol. 7, No. 3, p 319.

49. Kit and Tsinober, Magneto Hydro Dynamics, Vol. 7, No. 3, 27-34, (1971)

50. R. G. Deissler, Phys. Fluids, 6, 1250, (1963).

51. T. Tatsumi, Rev. Mod. Phys., 32, 807, (1960).

52. A. Alemany, R. Moreau, P. L. Sulem, U. Frisch, Jnl. de Mecanique, Vol. 18, No. 2, 1979.

53. J. Sommeria and R. Moreau, Submitted to J. Fluid Mechanics, Jan. 1981.

Turbulence Modelling

54. J. Laufer, N.A.C.A Rep. 1174 (1954)

55. L. H. Lehrer, IEC Process Design and Development, 7 (1968), p226

56. T. C. Hsiao, T. Lehner and K. Bjorn, Scand. J. Metallurgy, 9 (1980) p105

57. O. Haida, T. Emi, H. Bada and F. Sudo, Tetsu-to-Hagane, 66 (1980), S253

58. M. Sano and K. Mori, Scaninjet III, Part 1, No. 6, Third Intl. Conf. on Refining of Iron and Steel by powder injection, Lulea, Sweden, June 15-17, 1983

59. N. Nakanishi, T. Fujii and J. Szekely, Ironmaking and Steelmaking (Quart.), 1975, No. 3, p193

60. R. M. Figueira, Turbulent Fluid Flow Phenomena in an AOD System, Sc.D. Thesis, MIT, 1983

61. I. Proudman and W. H. Reid, Phil. Trans. A, 247, 163.

62. G. K. Batchelor and I. Proudman, Phil. Trans. A, 248, 396.

63. S. Orszag, Studies in Appl. Math., 1971

64. W. Rodi, D. B. Spalding, Warme und Stoffubertragung, 3, 85.

65. D. B. Spalding, Chem. Eng. Sci., 26, 95.

66. L. G. Lotsianskii, N.A.C.A. Tech. Mem., No. 1079, 1945.

67. S. Corrsin, A.I.Ch.E Jrnl., Vol. 3, No. 3, Sept. 1957, p 329.

68. S. Corrsin, A.I.Ch.E Jrnl., Vol. 10, No. 6, Nov. 1964, p 870.

69. P. S. Brodkey, A.I.Ch.E Jrnl., Vol. 12, No. 2, p403, March 1966

70. P. S. Brodkey (ed.), Turbulence in Mixing Operations, Chapter 2, p 49, Academic Press, New York, 1975.

71. S. Orszag and Patterson, Phys. Rev. Letters, 28, p 76, (1972).

72. S. Orszag, Phys. Fluids Suppl. II, 12, 250, (1966)

73. Deardorff, J. Fluid Mech., 41, 453-480, (1970).

74. Launder, Reece and Rodi, J. Fluid Mech., 68, 1975, 537-566.

75. Hanjalic and Launder, J. Fluid Mech., Vol. 52, Part 4, pp 609-638, (1972).

76. U. Schumann, J. Comp. Phys., 18, 376-404.

77. J. Beek and R. S. Miller, Chem. Engg. Progress Symposium Series, No. 25, Vol. 55, p23.

78. J. L. Lumley, Phys. Fluids, Vol. 8, No. 6, p 1056-1062.

79. W. P. Jones and B. E. Launder, Int. J. Heat Mass Transfer, 15, 301, (1972).

80. G. Comte-Bellot and S. Corrsin, J. Fluid Mech., 25, 1966, 657-682.

Fluid Flow and Transport Phenomena Texts.

81. R. B. Bird, W. E. Stewart and E. N. Lightfoot, Transport Phenomena, J. Wiley and Sons, New York, (1960)

82. W. M. Pun and D. B. Spalding, A General Computer Program for Two Dimensional Elliptic Flows, Report No. HTS/76/2, Heat Transfer Section, Imperial College of Science and Technology, (1976).

83. A. D. Gosman, W. M. Pun, A. K. Runchal, D. B. Spalding and M. Wolfshtein, Heat and Mass Transfer in Recirculating Flows, Academic Press, London and New York (1969).

84. B. E. Launder and D. B. Spalding, Lectures in Mathematical Modelling of Turbulence, Academic Press, London and New York (1972).

85. P. J. Roache, Computational Fluid Dyanmics, Hermosa Publishers, New Mexico (1972).

86. J. O. Hinze, Turbulence, Second Edition, McGraw Hill, NY (1975)

87. Hughes and Young, The Electromagnetodynamics of Fluids, J. Wiley and Sons,(1966).

88. Tennekes and J. L. Lumley, A First Course in Turbulence, M.I.T. Press, Cambridge, Massachusetts, (1972)

89. R. S. Brodkey, The Phenomena of Fluid Motions, Addison-Wesley, Reading, Massachusetts, (1967)

90. J. Szekely, Fluid Flow Phenomena in Metals Processing, Academic Press, New York, (1979)

91. W. Rodi, Turbulence Models and their Application in Hydraulics - A State of Art Review, I.A.H.R Section on Fundamentals of Division II, (1980).

92. S. V. Patankar, Numerical Heat Transfer and Fluid Flow, McGraw Hill, NY, (1980)

93. P. Bradshaw (ed.), Topics in Applied Physics, Vol. 12, "Turbulence", Springer-Verlag, Berlin, F.R.G., (1978)

94. G. K. Batchelor, An Introduction to Fluid Dynamics, Cambridge University Press, (1967)

95. G. K. Batchelor, Theory of Homogeneous Turbulence, Cambridge University Press, (1960)

MHD Cylinder

96. Kh. E. Kalis, A. B. Tsinober, A. G. Shtern and E. V. Scherbinin, Magnetohydrodynamics, Vol. 1, No. 1, p 11 (1965)

97. E. Ya. Blum, Magnetohydrodynamics, Vol. 6, No. 2, p 212, (1970).

98. V. M. Soundalgekar and N. V. Vighnesam, Nucl. Eng.Des., Vol. 58, No. 1, pp 109-112, (1980).

99. P. S. Lykoudis and P. F. Dunn, Intl. J. Heat Mass Transfer, Vol. 16, No. 7, p 1439, (1973).

100. A. E. Hamiliec and J. D. Raal, Phys. Fluids, 12, No. 1, pp 11-17, (1969).

101. M. Kawaguti, Jrnl. Phys. Soc. Jap., Vol. 8, No. 6, pp 747-756, (1953).

102. S. C. R. Dennis and G. Z. Chang, J. Fluid Mech., Vol. 42, Part 3, pp 471-489, (1970).

103. W. Chester, J. Fluid Mech., Vol. 3, pp 304-308, (1957).

104. R. Ishiguro, K. Sugiyama and T. Kumada, Int. J. Heat Mass Transfer, Vol. 22, No. 7, p 1041, (1980).

105. R. J. Grosh and R. D. Cess, Trans. ASME, 80-3, p667, (1958-4).

106. H. Branover and A. B. Tsinober., Magnetohydrodynamics of Incompressible Media, (in Russian), (1970).