DRD Line Item No. SE-7

DOE/JPL-954929-81/8

JPL NO. 9950-481

INVESTIGATION OF RELIABILITY ATTRIBUTES AND ACCELERATED STRESS FACTORS ON TERRESTRIAL SOLAR CELLS

(NASA-CR-164012) INVESTIGATION OF RELIABILITY ATTRIBUTES AND ACCELERATED STRESS FACTORS ON TERRESTRIAL SOLAR CELLS Annual Report (Clemson Univ.) 255 p HC A12/MF A01 N81-19568 CSCL 10A G3/44 Unclas 41672

THIRD ANNUAL REPORT

J.W. Lathrop

R.A. Hartman

C.R. Saylor

Department of Electrical and Computer Engineering

Clemson University

Clemson, SC, 29631

JANUARY 1981

PREPARED FOR
JET PROPULSION LABORATORY

PREPARED BY
CLEMSON UNIVERSITY, CLEMSON, SOUTH CAROLINA 29631

DRD Line Item No. SE-7

DOE/JPL - 954929-81/8

ENGINEERING AREA

INVESTIGATION OF RELIABILITY ATTRIBUTES AND ACCELERATED STRESS FACTORS OF TERRESTRIAL SOLAR CELLS

THIRD ANNUAL REPORT

J. W. Lathrop

R. A. Hartman

C. R. Saylor

Department of Electrical and Computer Engineering

Clemson University

Clemson, SC 29631

January 1981

The JPL Low-Cost Silicon Solar Array Project is Sponsored by the U.S. Department of Energy and forms part of the Solar Photovoltaic Conversion Program to initiate a major effort toward the development of low-cost solar arrays. This work was performed for the Jet Propulson Laboratory California, Institute of Technology by agreement between NASA and DOE.

CLEMSON PERSONNEL

Persons contributing to the work covered in this report include:

Dr. Jay W. Lathrop --	Principal Investigator
Mr. Dexter C. Hawkins --	Engineer
Mr. Robert A. Hartman --	Graduate Student (Second Quadrant Modeling)
Mr. Charles R. Saylor --	Graduate Student (Short Interval Tester)
Mr. Haskell A. Walker --	Graduate Student (Color Measurements)
Mr. J. Fury Christ --	Graduate Student (Measurement Jigging and Data Processing)
Mr. William C. Chandler, Jr. --	Undergraduate Student
Mr. C. Mark Durham --	Undergraduate Student
Mr. Daniel J. Martins --	Undergraduate Student

ACKNOWLEDGEMENT

The Jet Propulsion Laboratory Technical Manager for this work was Mr. Edward L. Royal. His numerous contributions to the program are gratefully acknowledged.

ABSTRACT

Work covered in this report represents a portion of the third year's effort of a continuing program tc determine the reliability attributes of terrestrial solar cells. Major effort during this reporting period was devoted to two tasks: 1) improvement of the electrical measurement instrumentation through the design and construction of a microcomputer controlled short interval tester, and 2) better understanding of second quadrant behavior by developing a mathematical model relating cell temperature to electrical characteristics. In addition, some preliminary work is reported on an investigation into color changes observed after stressing. While the accelerated stressing of various cell types is continuing, no new results of this activity are presented in this report and the reader is referred to earlier reports for the latest available documentation.

SUMMARY

The third year of the accelerated reliability testing program concentrated on electrical measurement instrumentation and in modeling cell behavior in the second quadrant. In addition, some preliminary work was done on correlating cell color changes with electrical degredation. Not reported are results of continuing accelerated stress tests on state of the art cells. A number of new cells were added to the program, but not in time for sufficient data to be obtained, while the older cells are undergoing extended test periods and new data is not yet available on them.

The all-digital, microprocessor controlled, short interval tester, which was designed and fabricated on the program, has replaced the manual measurement procedure formerly used. This has improved measurement accuracy and repeatability, reduced measurement time, and through coordinated data management procedures, eliminated data errors. A complete description of the tester including schematics and software is given and its operating procedures described.

A computer model, based on the thermal and electrical properties of the cells and encapsulating materials, was developed to relate cell temperature to electrical characterstics in the second quadrant. This model adequately predicted the behavior of both encapsulated and unencapsulated cells, although accurate temperature measurements on encapsulated cells were difficult to obtain. In addition, only cells of one type were used for comparison and other cell types may require different parameter values for fitting. Use of the model should permit the prediction of a cell's sensitivity to degredation in the second quadrant. The computer program is listed together with a description of its operation.

In an effort to determine the possibility of quantifying AR coating effectiveness by color measurement some preliminary work was done using an IBM 7400 spectrophotometer system.

TABLE OF CONTENTS

CLEMSON PERSONNEL ii

ACKNOWLEDGEMENT iii

ABSTRACT iv

SUMMARY v

TABLE OF CONTENTS vii

LIST OF FIGURES

LIST OF TABLES

1.0 INTRODUCTION 3

2.0 DEVELOPMENT OF A MATHEMATICAL MODEL TO ASSESS THE RELIABILITY CONSEQUENCES OF OPEARTION IN THE SECOND QUADRANT - Subtask 1 - R. A. Hartman 9

- 2.1 Introduction 9
- 2.2 Breakdown Mechanisms 14
- 2.3 Basic Equations 22
- 2.4 Temperature Dependent "Constants" 27
- 2.5 Bounday Conditions 32
- 2.6 Unencapsulated Cell Results 36
- 2.7 Encapsulated Cell Results 53
- 2.8 Measurements and Results from the Model 56

3.0 DEVELOPMENT OF IMPROVED SOLAR CELL MESUREMENT METHODS 75

- 3.1 Introduction 75
- 3.2 System Hardware 78
  - 3.2.1 Microcomputer 78
  - 3.2.2 Interface Hardware 80
  - 3.2.3 The Solar Simulator 90
  - 3.2.4 Cell Holder 92
- 3.3 System Software 95
  - 3.3.1 Introduction 95
  - 3.3.2 Main Program Module 98
  - 3.3.3 ID Procedure 98
  - 3.3.4 Get Cell Number Procedure 98
  - 3.3.5 Step Through Procedure 98
  - 3.3.6 Compute Current Procedure 99
  - 3.3.7 Scope Procedure 103
  - 3.3.8 Print Procedure 103
  - 3.3.9 Fetch Procedure 103
  - 3.3.10 Numout Procedure 103
  - 3.3.11 Tbout Procedure 104

Table of Contents (cont.)

3.3.12 X1232 Procedure 104
3.3.13 Pmax Procedure 104
3.3.14 Global Variables Procedure 104
3.4 Operating Procedure 105
3.4.1 Power Up 105
3.4.2 Start Up 105
3.4.3 Routine IV Measurement Program 105
3.4.4 Time Varying Effect Measurement Program 107

3.5 Time Varying Effects 108

3.6 Data Management 112

3.7 Cost 116

4.0 ANTIREFLECTIVE COATING EXPERIMENTATION 119

5.0 CONCLUSIONS 127

5.1 Development of a Mathematical Model to Assess the Reliability Consequences of Operation in the Second Quadrant 127
5.2 Development of Improved Solar Cell Test Methods 128
5.3 Color Development of AR Coating 129

6.0 RECOMMENDATIONS 131

6.1 Improved Test Methods 131
6.2 Second Quadrant Model 131

7.0 NEW TECHNOLOGY 133

8.0 REFERENCES 137

APPENDIX A. TECHNIQUES FOR NUMERICALLY SOLVING PARTIAL DIFFERENTIAL HEAT FLOW EQUATIONS 141

APPENDIX B. FLOW CHARTS AND PROGRAMS FOR THE MICROCOMPUTER CONTROLLED, CONSTANT TEMPERATURE, SHORT INTERVAL IV MEASAUREMENT SYSTEM ..151

APPENDIX C. ANTIREFLECTIVE COATING AND COLOR MEASUREMENT THEORY 189

APPENDIX D. COMPUTER PROGRAM LISTINGS FOR MODELLING ENCAPSULATED AND UNENCAPSULATED CELLS IN THE SECOND QUADRANT 201

LIST OF FIGURES

Figure		Page
2.1	Current-Voltage Characteristic of an Unencapsulated Solar Cell in First and Second Quadrants	11
2.2	Comparisons of Pulsed and Steady State V-I Characteristics	13
2.3	Intrinsic Temperature vs. Doping Level	17
2.4	Hole Formed in Cell as Result of Mesoplasma Formation	19
2.5	Assumed Unencapsulated Cell Geometry	23
2.6	Assumed Encapsulated Cell Geometry	25
2.7	Second Quadrnt I-V Characteristics Taken at Different Values of Temperature	28
2.8	Experimental Arrangement Used to Determine V-I Characteristics as a Function of Temperature	29
2.9	Typical Current Density vs. Temperature Curve for Second Quadrant (V = 1 volt)	30
2.10	Encapsulated Cell Space Grid	34
2.11	Measured VI Characteristics of Four Unencapsulated Cells in the Second Quadrant	37
2.12	Schematic of Second Quadrant Measurement Method	38
2.13	Phosphor Decoration Measurement Method	39
2.14	IR Scanner Thermograph of an Unencapsulated Cell in the Second Quadrant	41
2.15	Calculated VI Characteristics of an Unencapsulated Cell for $\gamma = 0.2$	42
2.16	Calculated VI CHaracteristics of an Unencapsulated Cell for $\gamma = 0.4$	43
2.17	VI Curve of Cell 1 Compared with Calculated Curve for $\beta = 1$	45
2.18	Diagram Illustrating Changes in the Physical Appearance of a Cell as Current Increases	47
2.19	Calculated Temperature Profile for $\beta = 1$ and $\gamma = 0.4$	48

List of Figures (cont.)

Figure		Page
2.20	Calculated VI Curves for Different Cell Thickness	51
2.21	Calculated Temperature Profile for Different Cell Thicknesses	52
2.22	Minimodule Construction Details	54
2.23	Thermogravimetric Analysis of EVA	55
2.24	Measured VI Curve of Glass Encapsulated Cells	58
2.25	Measured VI Curve of Masonite Encapsulated Cells	59
2.26	Minimodule with Glass Superstate Cracked	60
2.27	Result of Gas Formation on Minimodule	61
2.28	Cell C After Second Quadrant Operation	63
2.29	Cell D After Second Quadrant Operation	64
2.30	Cell E After Second Quadrant Operation	65
2.31	Calculated Vi Characteristic of a Glass Encapsulated Cell for $\gamma = 0.2$	68
2.32	Calculated VI Characteristic of a Glass Encapsulated Cell for $\gamma = 0.4$	69
2.33	Calculated VI Characteristic of a Masonite Encapsulated Cell for $\gamma = 0.2$	70
2.34	Calculated VI Characteristic of a Masonite Encapsulated Cell for $\gamma = 0.4$	71
2.35	Comparison of Clemson Model with JPL Prediction	72
3.1	Measurement System Block Diagram	77
3.2	Microcomputer System Block Diagram	79
3.3	8251 Mode Configuration	81
3.4	8251 Command Configuration	82
3.5	Port 1 Command Word	83
3.6	Port 2 Command Word	84

List of Figures (cont.)

Figure		Page
3.7	System Memory Allocations	85
3.8	Analog Interface Board Block Diagram	87
3.9	Diagram of Solar Simulator	93
3.10	Diagram of Cell Holder	94
3.11	Photograph of Cell Holder Fixture	96
3.12	Photograph of Measurement Instrumentations	97
3.13	Measured Cell characteristic Illustrating Different Compute Current Mode Regions	102
3.14	V and I as a Function of Illuminations Time for Representative Cells	109
3.15	P_m as a Function of Illumination Time for Representative Cells	110
3.16	Voltage-Current Plots for Different Illumination Times	111
3.17	Initial Method of Trensferring Measurement Data to the IBM-370 Computer	113
3.18	Present Method of Transferring Short Interval Tester Data to IBM-370 Computer	115
4.1	Visually Estimated % of AR Coating Removed vs. Mesured % Decrease in I_{sc} and P_m for Lots F-14 and F-19 after 500 Hours Pressure Cooker	120
4.2	Visually Estimated % of AR Coating Removed vs. Measured % Decrease in I_{sc} and P_m for Lots G-14 and G-19 after 500 Hours Pressure Cooker	121
4.3	Visually Estimated % of AR Coating Removed vs. Measured % Decrease in I_{sc} and P_m for Lots H-14 and H-19 after 500 Hours Pressure Cooker	122
4.4	Alignment Mask in Position on an I-Cell.	125
7.1	Proposal Solar Cell with Integrated Bypass Diode and Wrap Around Top Contact	134
7.2	Output Characteristic of 36 Cell Module with and Without Bypass Diodes	135

List of Figures (cont.)

Figure		Page
A-2	Geometric Interpretation of a Second-Order-Correct First Derivative	143
A-3	Temperature Calculations Subroutine	147-149
A-4	Structure of the Complete Program	150
B-1	Get Cell Number Procedure	151
B-2	Step Through Procedure	152
B-3	Compute Current Procedure	153
B-4	Compute Current Procedure Mode 0	154
B-5	Compute Current Procedure Mode 1	155
B-6	Compute Current Procedure Mode 2	156
B-7	Scope Procedure	157
B-8	Print Procedure	158
B-9	Fetch Procedure	159
B-10	Numout Procedure	160
B-11	Tbout Procedure	161
C.1	Reflected and Transmitted Rays at the Surface of a Solar Cell	190
C.2	Interface by Reflection from a Thin Film, Assuming an Extended Source	192
C.3	The Dimensions of Color	194
C.4	Spectral Distributions of Alluminants A & C	194
C.5	Spectral Distribution of Solar Radiation	196
C.6	1976 CIE L*A*B Color Space	197

LIST OF TABLES

Table		Page
1.1	Physical Characteristics of Cells in the Accelerated Test Program	5
2.1	Measured and Calculated Tempertures at the Hot Spot of an Unencapsulated Cell	46
2.2	Measured and Calculated Temperature at the Edge of an Unencapsulated Cell	49
2.3	Estimated Temperatures at Observation Points Compared with Calculated Temperatures for Encapsulated Cells	67
3.2	Command Word to Programmable Gain Amplifier	88
3.3	D/A Converters Memory Address Locations	89
3.4	Cost of Major Capital Equipment	116

1.0 INTRODUCTION

1.0 INTRODUCTION

This is the Third Annual Report on the Investigation of Relability Attributes and Accelerated Stress Factors on Terrestrial Solar Cells, a program being conducted by Clemson University for the Low Cost Solar Array (LSA) Project of the Jet Propulsion Laboratory. The First Annual Report was an introductory report which discussed the philosophy of accelerated testing as applied to solar cells, reported the results of applying certain test procedures to four types of state of the art solar cells, and concluded with a preliminary reliability qualification test schedule. Also included in the first year's program were the development of techniques for reproducibly measuring electrical cell parameters and the implementation of a computerized system for reliability data management and analysis.

The second year program refined and extended many of the techniques developed under the first year's program. Additional accelerated environmental testing was performed, both on the original four cell types and on three additional cell types. These additional cell types involved advanced crystal growing and assembly methods, but did not significantly differ in their metalization systems. As a result of this additional testing, several changes were made to the preliminary qualification test schedule resulting in a reduction in both the number of tests and the number of cells required per test. During the second year's effort it became evident that, in addition to studying first quadrant failure modes, the contract should spend some time characterizing the second quadrant. Results from the field were pointing to inadvertent operation in the second quadrant as the major cause of field failures. Consequently a second quadrant subtask was initiated.

PRECEDING PAGE BLANK NOT FILMED

It was also apparent during the second year of the program that it would be necessary to test numerous different cell types as the design-to-cost philosophy of the LSA program led to different and less expensive metalization systems. In order to cope with additional quantities of cells, on top of those already in the program, revision of the electrical measurement methods was required and a measurement method subtask was initiated.

Because of these two factors—second quadrant induced field failure and the need for improved measurement methods—most of the effort in the third year of the contract, and consequently the bulk of this annual report, has been devoted to these two subtasks. The statistical analysis of accelerated stress test results, which dominated the first two reports, is thus missing from this report. However, a comprehensive review of the accelerated testing program was presented in a workshop held May 1-2, 1980 at Clemson and proceedings are available as a separate JPL document*. Furthermore, a number of new cell types with radically different metalization systems were added to the program too late for any significant amount of data to be available for this report. Table 1.1 is a summary of the physical characteristics of all cell types presently in the program.

Two additional items discussed in this report are a technique for measuring the loss of antireflecting coating during stress testing, and an update on the data collection system, which has changed somewhat as a result of the improved measurement instrumentation.

*JPL Low-Cost Solar Array Project Document 5101-163.

TYPE	SIZE (INCHES)	CRYSTAL TECH	THICKNESS (MILS)	SURFACE A/R	SURFACE TEXT	JUNCTION TECH	METAL TECHNOLOGY
A	4-DIAM	CZ	24			DIFF P/N	SOLDER BASED
B	3-DIAM	CZ	19	✓		DIFF N/P	THIN FILM Ti/Pd/Ag
C	2-DIAM	CZ	20	✓		DIFF N/P	SOLDER BASED
E	3-DIAM	CZ	15		✓	DIFF N/P	THICK FILM Ag
F	3.9x0.8	EFG	13	✓		DIFF N/P	SOLDER BASED
G	3-DIAM	CZ	12	✓	✓	DIFF N/P	THIN FILM Ti/Pd/Ag + SOLDER
H	2 x 2	WACKER POLY	12	✓		IMPLANT N/P	THIN FILM Ti/Pd/Ag
I	3-DIAM	CZ	12	✓	✓	DIFF N/P	THIN FILM Ti/Pd/Ag + SOLDER
J	2 x 2	CZ	12	✓		DIFF N/P	PLATED Pd/Ni
K	3-DIAM	CZ	12	✓		DIFF N/P	PLATED Pd/Ni/Cu
L	3-DIAM	CZ	13	✓		DIFF N/P	PLATED Pd/Ni/Cu/Pb
M	2-DIAM	CZ	12	✓		DIFF N/P	PLATED Pd/Ni/Cu

Table 1.1 Physical Characteristics of Cells in the Accelerated Test Program

2.0 DEVELOPMENT OF A MATHEMATICAL MODEL TO ASSESS THE RELIABILITY CONSEQUENCES OF OPERATION IN THE SECOND QUADRANT

Subtask 1

R. A. Hartman

PRECEDING PAGE BLANK NOT FILMED

2.0 DEVELOPMENT OF A MATHEMATICAL MODEL TO ASSESS THE RELIABILITY CONSEQUENCES OF OPERATION IN THE SECOND QUADRANT -- Subtask 1 - R. A. Hartman

2.1 Introduction

It has been shown, through an extensive laboratory accelerated test program, that the application of high temperature, high humidity, and high rates of temperature change, either singly or in combination, can result in irreversible change in the electrical characteristics of solar cells. These changes often manifest themselves in a gradual degradation with time of a cell's power output. In addition, at sufficient stress levels or if stressed for a sufficient length of time, cells may display a catastrophic failure mode in which the cell becomes either open circuited or shor circuited. Although extrapolation of the observed effects in the laboratory to use conditions is difficult because of uncertainties in the acceleration factors, the best estimates of the amount of power reduction which could be expected from state of the art cells over a 20-year period gave no cause for alarm. In a number of field instllations, however, a greater than anticipatd power reduction has been observed. Subsequent investigation of the field performance of deployed arrays led to the conclusion that the problem stemmed from cells becoming reverse biased and operating in the second, rather than the first, quadrant of the VI characteristic. When this occurs the affected cell dissipated an increased amount of power and as a result increases in temperature. An examination of operating arrays that were not designed to avoid second quadrant operation, using infrared scanning instrumentation, shows dramatically that numerous cell or parts of cells are appreciably hotter than their surroundings and consequently are operating in the second quadrant. It can be recognized that when this occurs the affected cells are in effect experiencing accelerated stressing under use conditions and should experience the same

PRECEDING PAGE BLANK NOT FILMED

accelerated degradation and catastrophic failure rates experienced in the laboratory. When a cell open circuits as a result of the high stress level, power is of course lost from the entire series string and, depending on the particular module and array design, rather rapid and irreversible system power loss can result. This section of the report examines factors governing the temperature of cells operating in the second quadrant and develops a computer model to relate temperature to electrical characteristics.

A cell will operate in the second quadrant whenever the system passes current through the cell in excess of the cell's short circuit current, I_{sc}. There are a variety of reasons why this might happen: external events, such as bird droppings or fallen leaves may cause partial shading of a cell; something could be wrong with the module itself, such as interconnect failure or a cracked cell; or parametric degradation could occur resulting in a change in a cell's electrical characteristic.

When the operating point shifts from the first to the second quadrant, appreciable power is dissipated in the cell. Figure 2.1 illustrates this clearly. It shows the V-I curve of a particular type of solar cell in the first and second quadrant. Constant power hyperbola's are superimposed to permit estimation of the power which is generated, and therefore must be dissipated, at any point along the curve. While this particular solar cell can supply about 1 Watt to an external load when operating in the first quadrant, the power that is dissipated in the second quadrant can easily be 10 to 50 times that amount and in the vertical region of the curve (breakdown) will be only limited by the external circuit resistance. Solar cells are current sources and the physical configuration of series parallelling will determine the maximum current that can occur. (2.1)

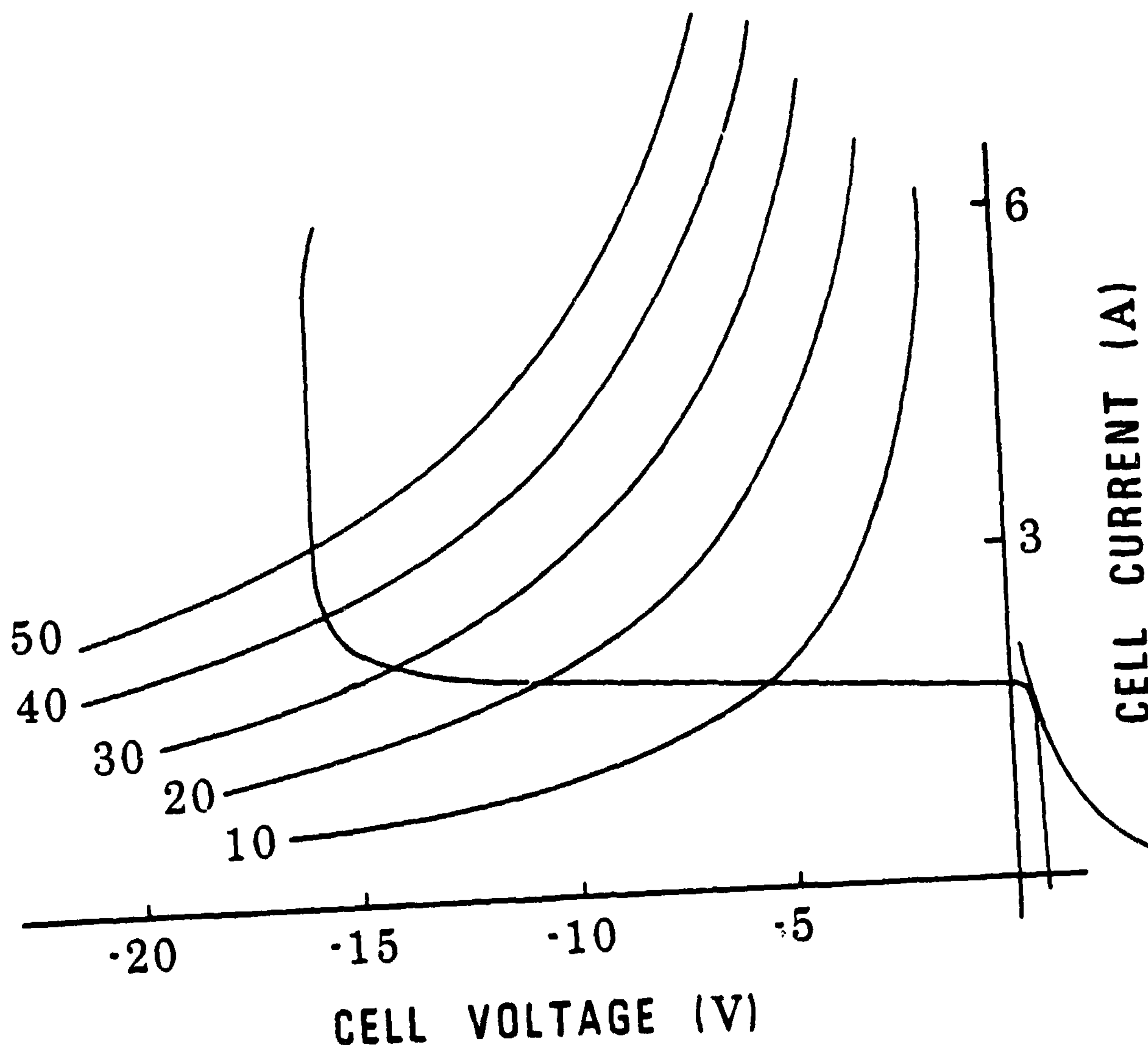


Figure 2.1 Current-Voltage Characteristic of an Unencapsulated Solar Cell in First and Second Quadrants

The thermal properties of the module determine the shape of the VI characteristic that will be obtained in the second quadrant. The cell in Figure 2.1 was suspended in free air without heat sinking. Figure 2.2 shows two characteristic traces of a cell. The top trace was obtained as in Figure 2.1 while the bottom trace was obtained using a pulsed high current fixture and curve tracer. The high current fixture permits the solar cell characteristics to be taken under pulsed conditions of up to 200 amperes peak current. The pulse width is 300 sec at a 60 Hz repetition rate, resulting in a duty cycle of less than 2%, and negligible heating. The bottom trace therefore corresponds to a cell mounted in a module having an infinite heat sink, i.e. a heat sink that would not allow a rise in temperature above ambient no matter what power was dissipated in the cell. The trace of an actual cell in a module will always be above the bottom trace, since the module is not an infinite heat sink, and below the top trace because the thermal conductivity of the encapsulant is better than the thermal conductivity of air.

Second quadrant effects were first described by F. A. Blake and K. L. Hanson in 1969. (2.2) The phenomena was "discovered" during studies of problems associated with high voltage solar arrays. They concluded that the principal danger of operation in the second quadrant was that it could amplify a small and sometimes temporary condition into a complete open circuit failure. They further noted that a small number of cells in parallel accentuate the severity of second quadrant operation and pointed out the use of by-pass diodes. R. N. Diamond and E. D. Steele (2.3) described the possibility of integrating a diode for each cell into the back of its neighbor. The article discussed the construction in detail and showed V-I curves with and without diodes. In the New Technology Section of this report is a description of a

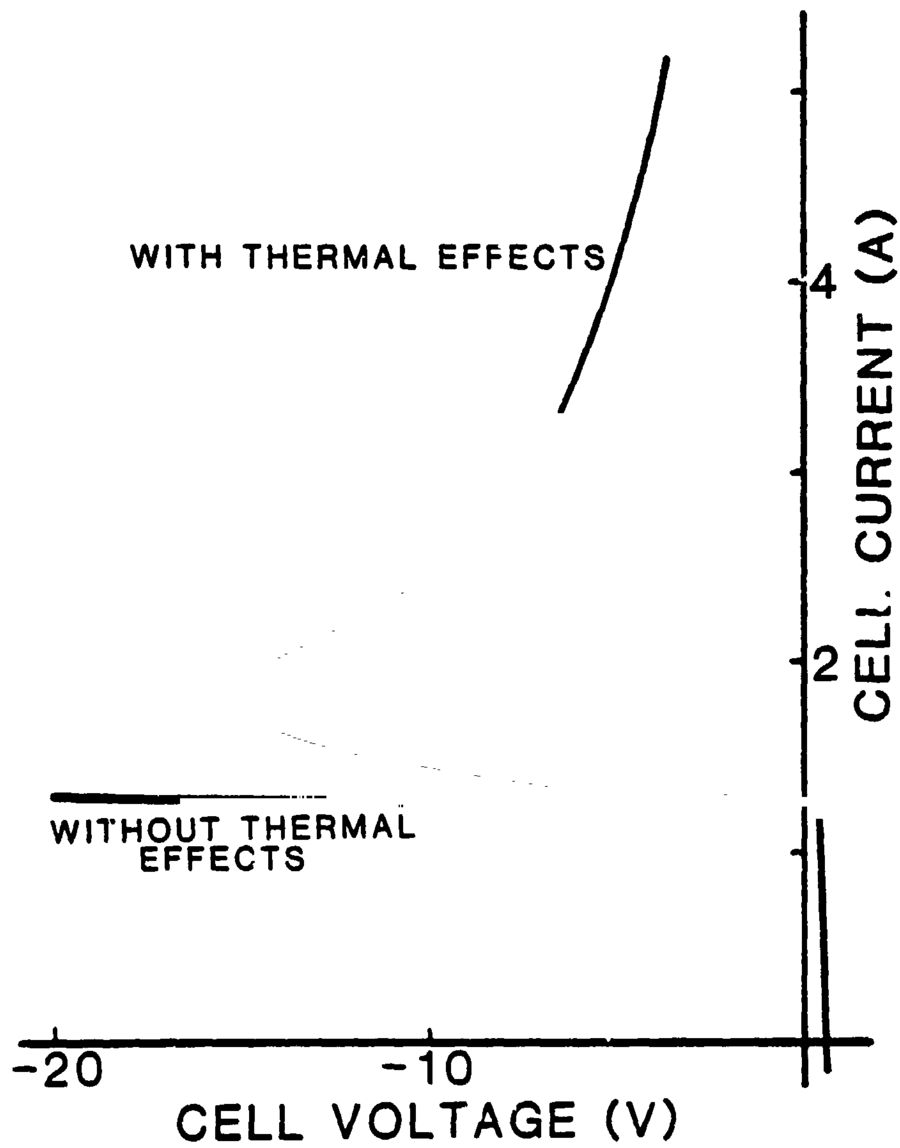


Figure 2.2 Comparisons of Pulsed and Steady State V-I Characteristics

further development of the integrated diode cell. R. N. Diamond (2.4) further described radiation effects and diode characteristics. P. L. Jett and J. L. Miller (2.5) analyzed three proposed configurations for the orbital workshop solar array for hot spot sensitivity. The configurations studied are strings of 154 by 1, 2, or 4 cells. Excessive heating was formed when three out of four parallel cells were shaded. Further analyses for skylab were made by N. B. North and D. F. Baker (2.6). N. R. Garner (2.7) applied a statistical analysis technique to determine the maximum number of failure that could be expected during a space ship maneuver that casts a shadow on the array.

H. S. Rauschenbach and E. E. Maiden (2.8) discussed breakdown phenomena in reverse biased solar cells. They found short circuit failures and occasional silicon melts. M. Sayed and L. Partain (2.9) discussed the effect of the second quadrant mode of operation on cadimum sulfide cells.

R. G. Ross (2.10) described the implications of reverse bias on module and array reliability. He pointed out that a proper design of series/parallel configurations could reduce the problem.

Three papers in the German literature by K. Schneider and W. Schultze (2.11), by J. Rath and B. Schultz (2.12) and by B. Gohrbrandt and B. Georgens (2.13) mainly reviewed the hot spot problem and its possible solutions.

The present work, described in this report, develops a mathematical model, based on heat flow considerations, which relates cell temperature and the shape of the VI characteristic curve in the second quadrant. Such a model is unique and has not been found in the literature.

2.2 Breakdown Mechanisms

Figure 2.2 shows a great difference in breakdown voltage when measured under pulsed and steady state conditions. The breakdown mechanism associated

pulsed measurement (lower tract), is avalanch multiplication. The voltage across the p-n junction increases to the point where the electric field in the space charge region is sufficient to create additional carriers by impact ionization. This type of junction breakdown has a positive temperature coefficient, i.e., the breakdown voltage increases as the temperature increases, and is the type of breakdown normally found in semiconductor diodes. Two other breakdown mechanisms are Zener and thermal. Zener breakdown is caused by carriers tunnelling through the barrier and is only found in semiconductor devices that have extremely high doping concentrations on both sides of the junction. These high levels of doping are not found in silicon solar cells. The last of the three breakdown mechanisms involves thermal instability and in discrete devices is referred to as second breakdown. Second breakdown is often observed in transistors and SCR's, but not in diodes where the pulsed and steady state breakdown values are the same because of low leakage currents and good heat sinking.

This is not the case for solar cells, however, where steady state breakdown is appreciably lower than pulsed breakdown because of the tremendous amount of power which can be dissipated in the second quadrant (reverse bias polarity) by virtue of the cell being illuminated. The power dissipated in the illuminated cells is roughly the short ciruit current, i_{sc}, times the voltage across the cell. Since the cell is much hotter in steady state, due to the higher dissipation, and yet its breakdown voltage less, the breakdown mechanism must not be avalanch multiplication which exhibits a positive temperature coefficient.

The mechanism which causes the current to increase as the reverse voltage of Figure 2.2 approaches 15 volts is the increase in leakage current due to the thermal generation of carriers.

At this point the thermally generated leakage current has become an appreciable fraction of the total current. As previously discussed, the heat which thermally generated the hole-electron pairs comes from the photon current and the reverse potential. In silicon, the thermal generation of hole-electron pairs results in an intrinsic concentration of electrons and holes per cubic centimeter given by

$$n_i = 1.69 \times 10^{19}\left(\frac{T}{300}\right)^{3/2} e^{-0.55/kT}.$$

At 300°K the intrinsic concentration is 1.38×10^{10} cm^{-3}, and is negligible compared to ordinary doping levels. Consequently any thermally generated leakage at this temperature will be orders of magnitude less than the reverse diode diffusion current derived from doping levels, which, in turn, is orders of magnitude less than the photon generated current. However, n_i increases rapidly with temperature, approximately doubling for every 11°C rise in temperature, and at high temperatures thermal generation will become the dominant process of hole-electron production.

Figure 2.3 is a plot of the temperature at which the concentration due to thermal effect, n_i, becomes equal to the original concentration of the base as a function of the base doping level. This temperature is termed the intrinsic temperature, θ_i. It can be seen that for normal base doping levels the intrinsic temperature is the order of 300°C. Below θ_i, the carrier concentration in silicon is relatively temperature independent, while above θ_i the power dissipated in the slice due to the photon and thermally generated reverse current causes the cell to heat uniformly and operation is inherently stable. As the reverse voltage is further increased, the localized

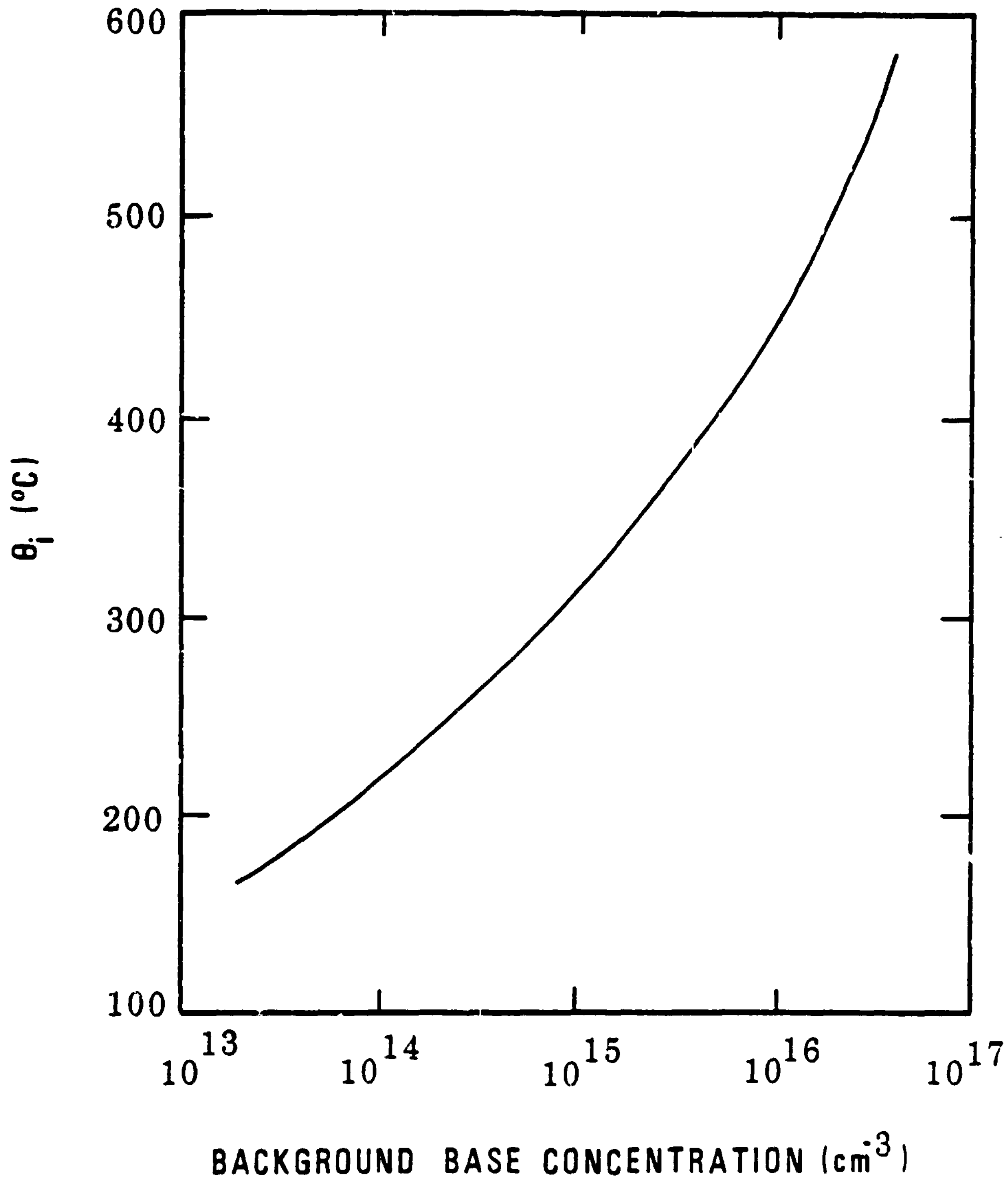


Figure 2.3 Intrinsic Temperature vs. Doping Level

temperature at some point will come close to, or exceed θ_i. When this occurs, thermally generated current becomes appreciable compared to the photon current, increasing the localized power dissipation and temperature still further, resulting in a positive feedback situation capable of producing thermal runaway with an increase in current density by 2 or 3 orders of magnitude. It is said at this point that a localized mesoplasma has formed. Any inhomogeneity that results in a localized temperature in excess of the intrinsic value can lead to mesoplasma formation and thermal runaway. This inhomogeniety could come about as a material property through structural or doping imperfections. Mesoplasma formation will depend on the ratio of the rate of power application to the thermal relaxation time of the cell. Thus a mesoplasma could be initiated if the current were suddenly increased to some value larger than I_{sc}, but at the same time, the cell could also operate satisfactorily at this same current level had it been approached gradually. Destruction of the cell in Figure 2.4 was caused by the sudden application of two times I_{sc}, whereas it had operated previously in a stable fashion at this same current level for an extended period. An encapsulated cell is less susceptible than a bare cell to mesoplasma formation because of the heat sinking properties of the encapsulating materials. As a matter of fact, in our work thermal runaway and subsequent mesoplasma formation were never observed in encapsulated cells, and even for unencapsulated cells it takes rigorous application of power to obtain it. The cell in Figure 2.4 had a voltage in excess of 60 volts suddenly applied and it took appoximately 3 seconds before the mesoplasma formed and melted the hole. Proper design should be capable of avoiding the generation of such sudden high voltages and currents within a module.

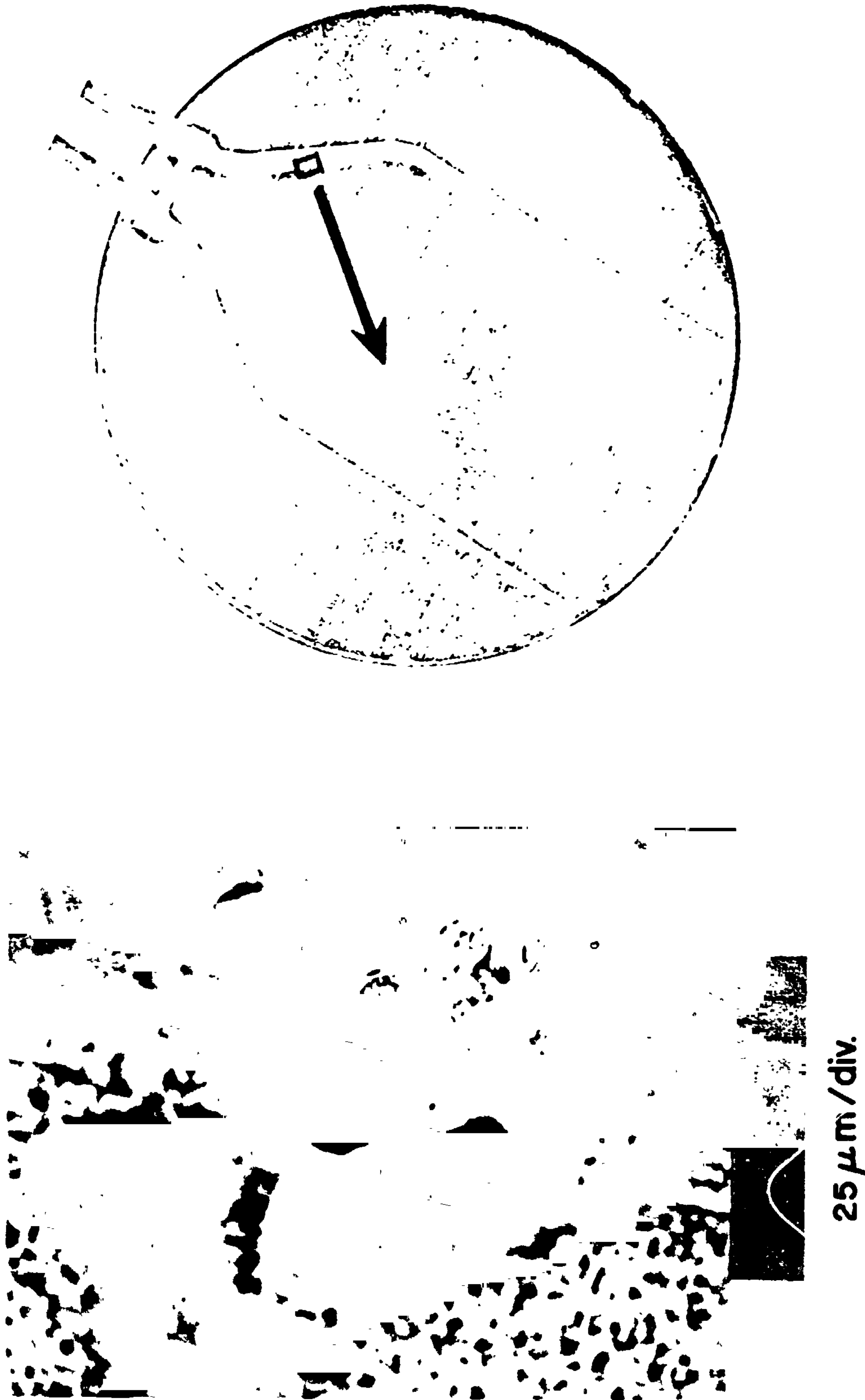


Figure 2.4 Hole Formed in Cell as a Result of Mesoplasma Formation

ORIGINAL PAGE IS
OF POOR QUALITY

On the other hand, long before mesoplasma formation with its thermal runaway is reached on the VI characteistic, non uniform heating and hot spot formation can occur. Hot spots will occur as temperatures approach $_{i}$; but where the carrier concentration does not necessarily have the strong exponential variation with temperature that it does above $_{i}$. Hence, a hot spot is inherently stable. It is in this sub-intrinsic region that most of the reverse voltage reliability problems occur as a result of localized cell temperatures of 100-200°C. The higher the temperature the stronger the carrier concentration dependence on temperature and the more localized the heating.

Because second quadrant breakdown is a thermal effect, it is important to understand the temperature dependence of the reverse current. To this end it can be assumed that the reverse current consists of several independent components, each with its own specific temperature dependence. The assumption of independence will be valid except near avalanch breakdown where multiplication should be taken into account.

The largest current component, at least at low voltages, is the photon generated current which has a positive temperature coefficient due to narrowing of the bandgap at elevated temperatures. When the bandgap becomes smaller the absorption edge shifts to longer wavelengths giving rise to increased current. In the temperature range of interest the effect will be linear

$$I_{ph} = I_o + \frac{\delta I_{ph}}{\delta T} \Delta T$$

where I_o is I_{sc} at room temperature. The actual temperature coefficient depends on the spectrum and is less than 1% of I_{sc} per ^{o}C.

Another temperature dependent component of current is the diode saturation current, also known as the diffusion current. For a one-sided $p^{+}n$

abrupt junction, similar to a shallow, heavily diffused junction, the current density is given by

$$J_s = \frac{qD_p p_{no}}{L_p}$$

where the quantities D_p, p_{no}, and $L_p = (D_p \tau_p)^{1/2}$ are all temperature dependent. Assuming that D_p/τ_p is proportional to some power of the temperature such that $D_p/\tau_p \sim T^s$ where s = constant, then,

$$J_s = \frac{qD_p P_{no}}{L_p} = q\sqrt{\frac{D_p}{\tau_p}}\frac{n_i^2}{N_d} \sim T^{3+\frac{s}{2}} \exp -\left(\frac{Eg}{kT}\right)$$

and the predominant temperature dependence will be that of the exponential term and a plot of the logarithm of current vs. 1/T should be a straight line with slope = -Eg/k.

The third temperature dependent component of current is the diode recombination-generation current. The temperature dependence of this component can be discussed in terms of a single trap located in the middle of the energy gap. For this case it can be shown that the rate of generation of electron hole pairs is

$$u = -\left[\frac{\sigma_p \sigma_n v_{th} N_r}{\sigma_n \exp\left(\frac{E_f - E_i}{kT}\right) + \sigma_p \exp\left(\frac{E_i - E_t}{kT}\right)}\right] n_i ,$$

where σ_p and σ_n are the hole and electron cross sections, respectively, v_{th} is the carrier thermal velocity, N_t is the trap density, E_t the trap energy level, E_i the intrinsic Fermi level, and n_i the intrinsic carrier density. Assuming that $\sigma_p = \sigma_n = \sigma$ and that the trap energy level is located at the intrinsic Fermi level then,

$$u = -\frac{\sigma\sqrt{3kT/m^*}\, Nt}{2} n_i \sim T^s T^{3/2} \exp\left(-\frac{E_g}{2kT}\right).$$

The recombination-generation current density is proportional to the space charge depletion width, W, so that

$$J_{rg} = \int_o^W q|u|dx = q|u|W$$

Thus a plot of the logarithm of this component vs. 1/T will also be a straight line, but with a slope = -Eg/2k. Since W is a function of the square root of voltage it should be noted that for an abrupt junction,

$$J_{rg} \sim V^{\frac{1}{2}}$$

2.3 Basic Equations

In order to derive an equation that describes the heat flow in a solar cell in the second quadrant it is necessary to make some a priori assumptions. First it is assumed that the cell is round and that the temperature distribution has radial symmetry. This means mathematically that if a hot spot exists, it will be located in the center of the cell. Secondly, it is assumed that there is no temperature difference between the top and the bottom of the cell, i.e. all vertical temperature gradients are zero. These two assumptions result in a one dimensional time dependent problem. Figure 2.5 indicates the assumed cell geometry. Equations may now be set up governing heat flow into and out of the differential ring element, as well as the heat generated within the ring as a result of current flow:

Heat conduction into the ring from the center,

$$q(\text{in at } r) = -2\pi r w \kappa \frac{\partial T}{\partial r} .$$

Heat conducted out of the ring toward the outside,

$$q(\text{out at } r + \Delta r) = 2\pi (r + \Delta r) w \kappa \left.\frac{\partial T}{\partial r}\right|_{r + \Delta r} .$$

Heat radiated to the ring from the light source,

$$q(\text{radiated}) = 2\pi r (ILL) \Delta r .$$

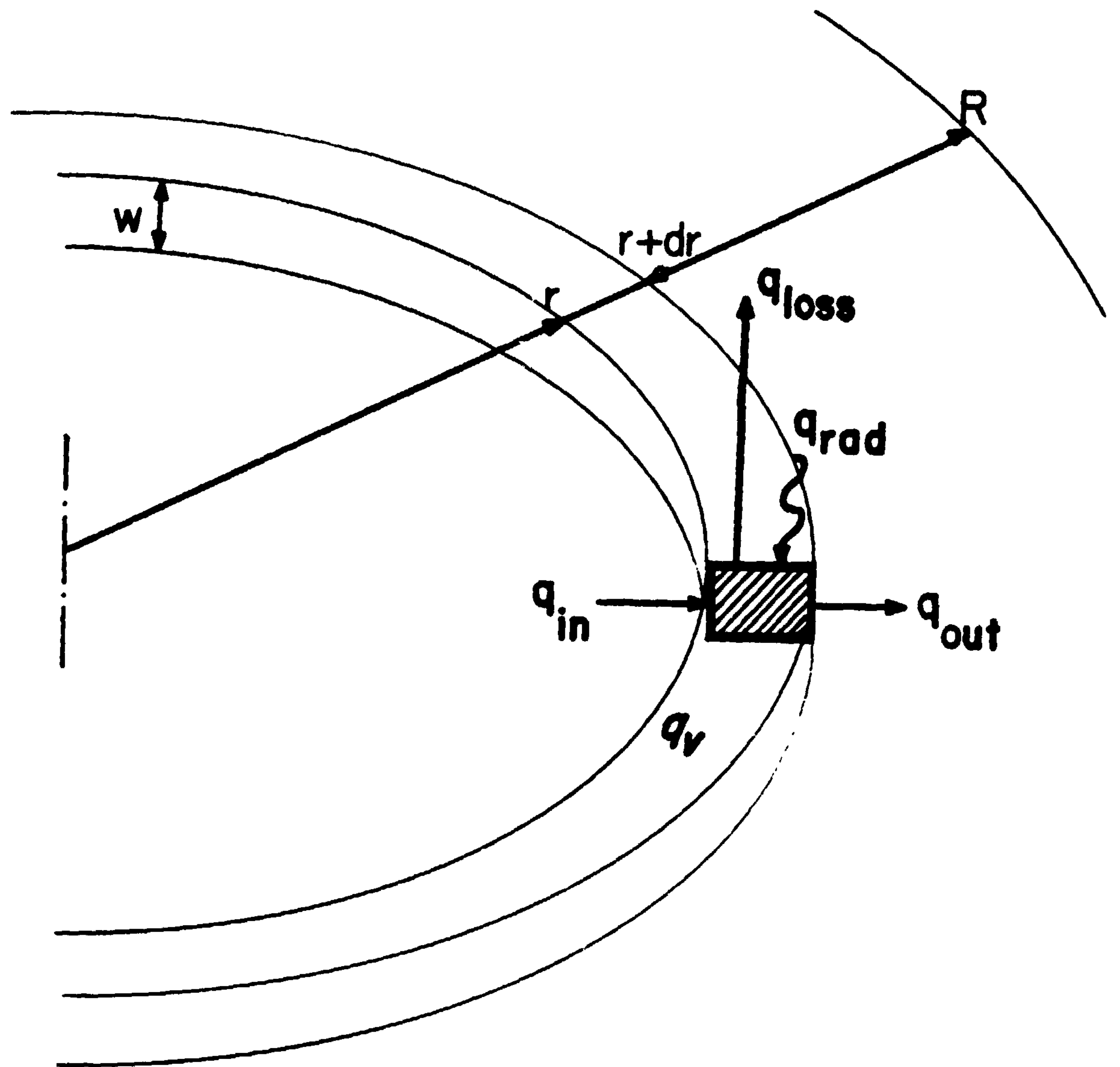


Figure 2.5 Assumed Unencapsulated Cell Geometry

Heat lost by the ring by conduction to ambient air,

$$q(\text{lost}) = -2\pi rh\ (T-T_{air})\Delta r\ .$$

Heat generated by current flowing within the ring,

$$q(\text{current}) = 2\pi rjV\Delta r\ .$$

Heat stored in the ring,

$$q(\text{stored}) = 2\pi r\Delta rw\rho c\ \frac{\partial T}{\partial t}$$

where w is the thickness of cell (cm),

κ is the thermal conductivity of the silicon (W/cm°C),

j is the current density of the current across the junction (A/cm^2),

V is the applied voltage (V), and

h is the heat exchange between cell surface and surroundings ($W/cm^2{}^\circ C$)

ILL is the incident energy flux from the sun ($W/cm^2{}^\circ C$)

ρ is the specific mass of silicon (g/cm^3)

c is the specific heat of the silicon (Ws/°Cg).

Combining all these terms into a single equation gives,

$$\frac{\partial^2 T}{\partial r^2} + \frac{1}{r}\ \frac{\partial T}{\partial r} + \frac{1}{\kappa}\ \frac{\partial \kappa}{\partial R}\left(\frac{\partial T}{\partial r}\right)^2 + \frac{jV - h(T - T_{air}) + ILL}{\kappa w} = \frac{1}{\alpha}\ \frac{\partial T}{\partial t}$$

where $\alpha = \frac{\kappa}{\rho c}$ = the thermal diffusivity.

This equation decribes the behavior of the unencapsulated solar cell at $0 < r < R$.

To include the effects of a layer of encapsulant the model is changed slightly by exchanging heat between the cell and the encapsulant instead of between the cell and the surroundings as shown in Figure 2.6. An additional set of equations is added that describes the heat transport in the encapsulant and between the encapsulant and the surroundings. These changes result in the following set of equations:

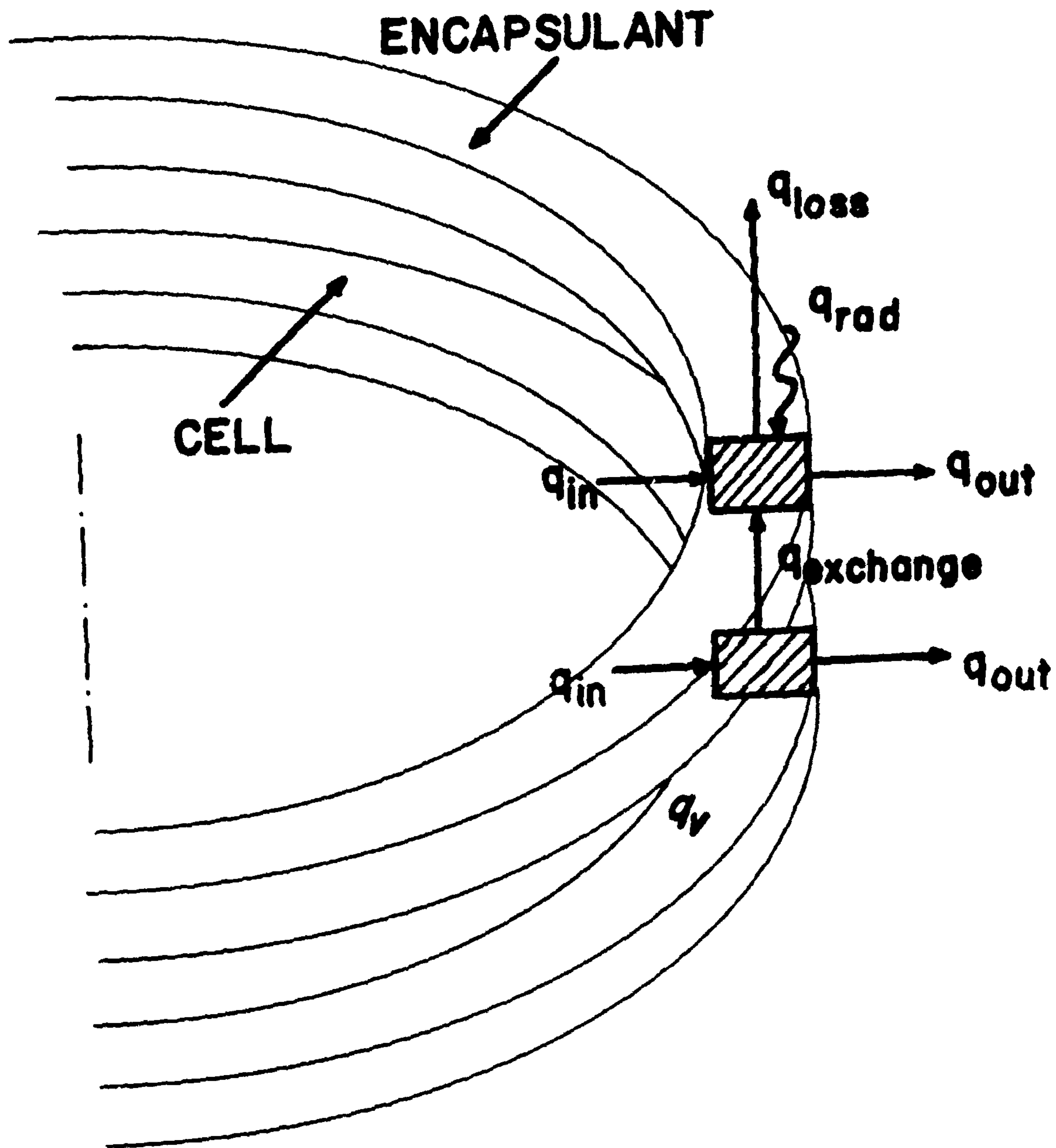


Figure 2.6 Assumed Encapsulated Cell Geometry

For the silicon slice:

$$q(\text{in at } r) = -2\pi r w_s \kappa_s \frac{\partial T_s}{\partial r}$$

$$q(\text{generated/exchanged}) = 2\pi r \Delta r \left\{ jV - h_{ce} (T_s - T_e) \right\}$$

$$q(\text{out at } r + \Delta r) = -2\pi(r + \Delta r) w_s \kappa_s \left. \frac{\partial T_s}{\partial r} \right|_{r + \Delta r}$$

$$q\ (\text{stored}) = 2\pi r \Delta r w_s \rho_s c_s \frac{\partial T_s}{\partial t}$$

For the encapsulant:

$$q(\text{in at } r) = 2\pi r w_e \kappa_e \frac{\partial T_e}{\partial r}$$

$$q(\text{exchanged}) = 2\pi r \Delta r \quad h_{ce} (T_s - T_e) - h_{ea}(T_e - T_{air}) + ILL$$

$$q(\text{out at } r + \Delta r) = -2\pi(r + \Delta r)\ w_e\ \kappa_e \ \frac{\partial T_e}{\partial r} \quad r + \Delta r$$

$$q(\text{stored}) = 2\pi r \Delta r w_e \rho_e c_e \ \frac{\partial T_e}{\partial t}$$

where the subscript s denotes the silicon cell and e the encapsulant, h_{ce} and h_{ea} stand for the heat exchange between the cell and the encapsulant and between the encapsulant and the air respectively. Setting up the heat balance as before gives:

$$\frac{\partial^2 T_s}{\partial r^2} + \frac{1}{r} \frac{\partial T_s}{\partial r} + \frac{1}{\kappa_s} \frac{\partial \kappa_s}{\partial T_s} \left(\frac{\partial T_s}{\partial r} \right)^2 + \frac{jV - h_{ce}(T_s - T_e)}{\kappa_s W_s} = \frac{1}{\alpha_s} \frac{\partial T_s}{\partial t} \qquad (3.2)$$

$$\frac{\partial^2 T_e}{\partial r^2} + \frac{1}{r} \frac{\partial T_e}{\partial r} + \frac{h_{ce}(T_s - T_e) - h_{ea}(T_e - T_{air}) + ILL}{\kappa_e w_e} = \frac{1}{\alpha_e} \frac{\partial T_e}{\partial t}$$

Note that the square of the first derivative does not occur in the encapsulant equation, since the thermal conductivity was assumed independent of temperature. For most encapsulant materials this seems to be justifiable. In

addition, the illumination term has been moved to the encapsulant equation to permit calculation of temperatures where there is encapsulant only, such as in the area directly around the cell that is studied. This might not be justified in high density modules.

The model does not allow for a variation of applied voltage across the cell due to surface resistivity. Actual variations of the order of 5% were measured between the voltage at the voltage probes, and that directly across a hot spot. This effect was neglected, since introduction of it would complicate the computations substantially. At high currents the resistive drop can shift a hot spot towards the current probe at the back contact.

The next section discusses some of the variables in the equation, like the current density and thermal conductivity, that are strong functions of temperature.

2.4 Temperature Dependent "Constants"

Several important "constants" in the equations derived in the last chapter such as the current density and the thermal conductivity, are really strongly temperature dependent.

Figure 2.7 shows the temperature dependence of the I-V plot. Clearly the current in the second quadrant is a strong function of temperature. These curves were obtained with the curve tracer and pulsed high current fixture while the cell was mounted on a hot plate equiped with vacuum hold down, voltage and current probes, and a thermocouple touching the back of the cell. The illumination was $100mW/cm^2$ from four ELH lamps. Figure 2.8 shows the construction details of the fixture used.

If in figure 2.7 a constant voltage is selected then it is possible to obtain plots of current versus temperature as shown in Figure 2.9 for a 1 volt

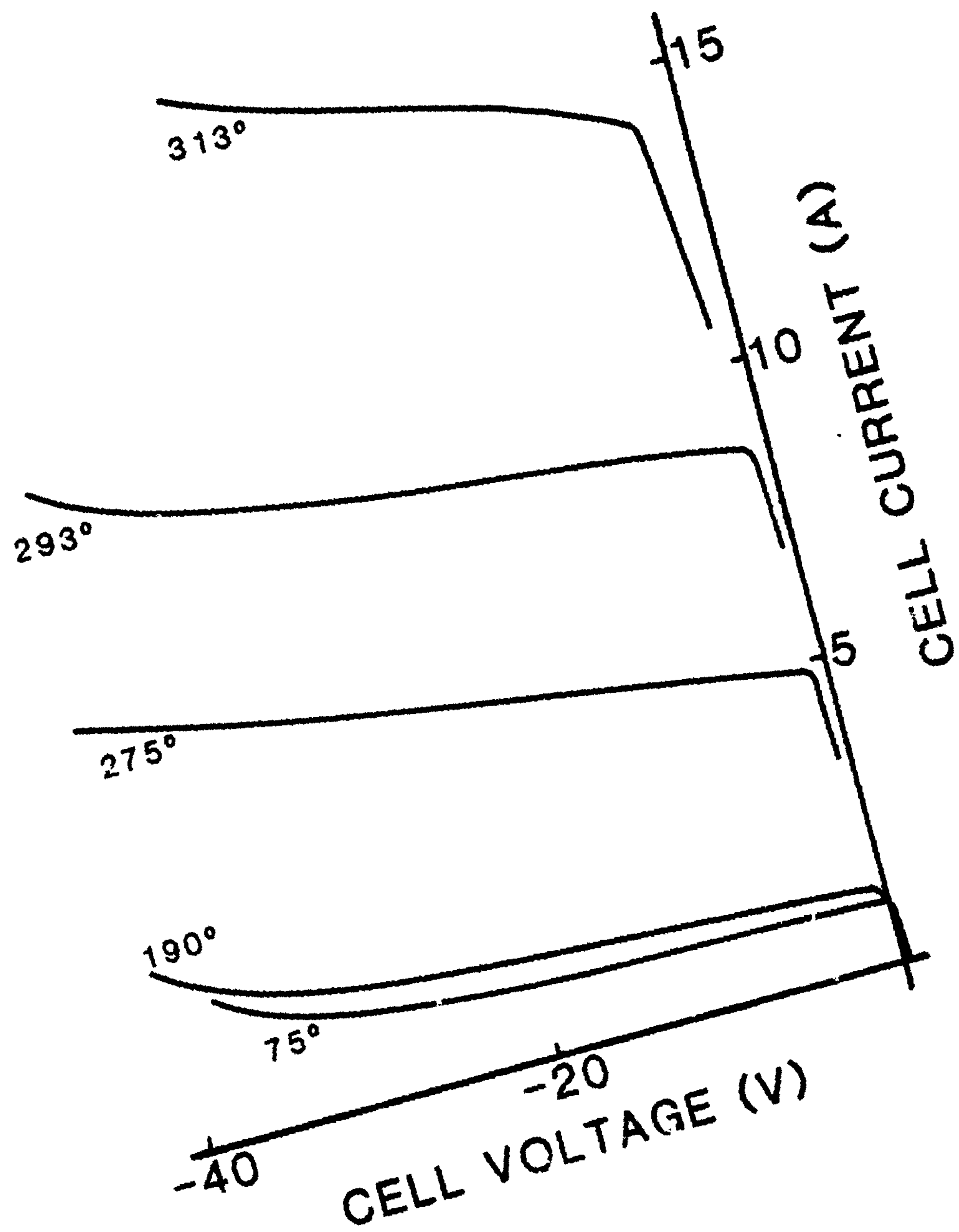


Figure 2.7 Second Quadrant I-V Characteristics Taken at Different Values of Temperature

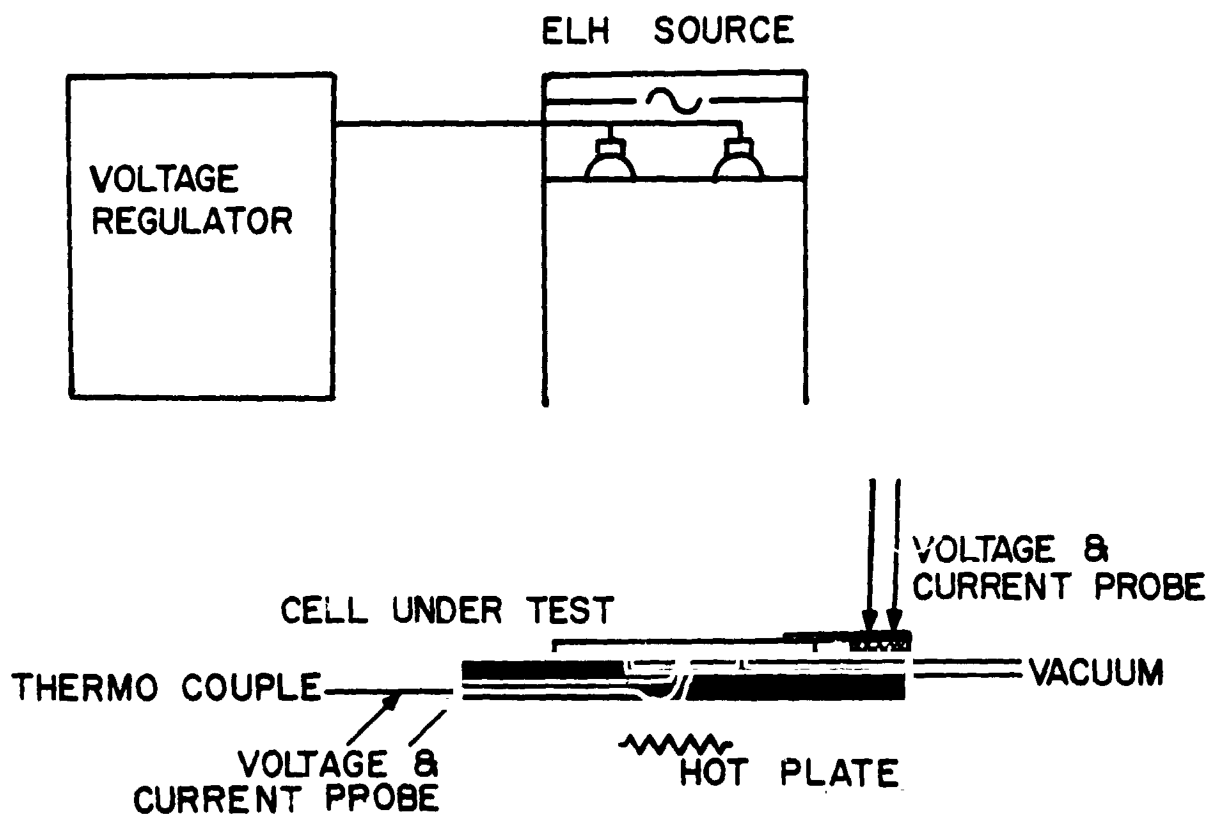


Figure 2.8 Experimental Arrangement Used to Determine V-I Characteristics as a Function of Temperature

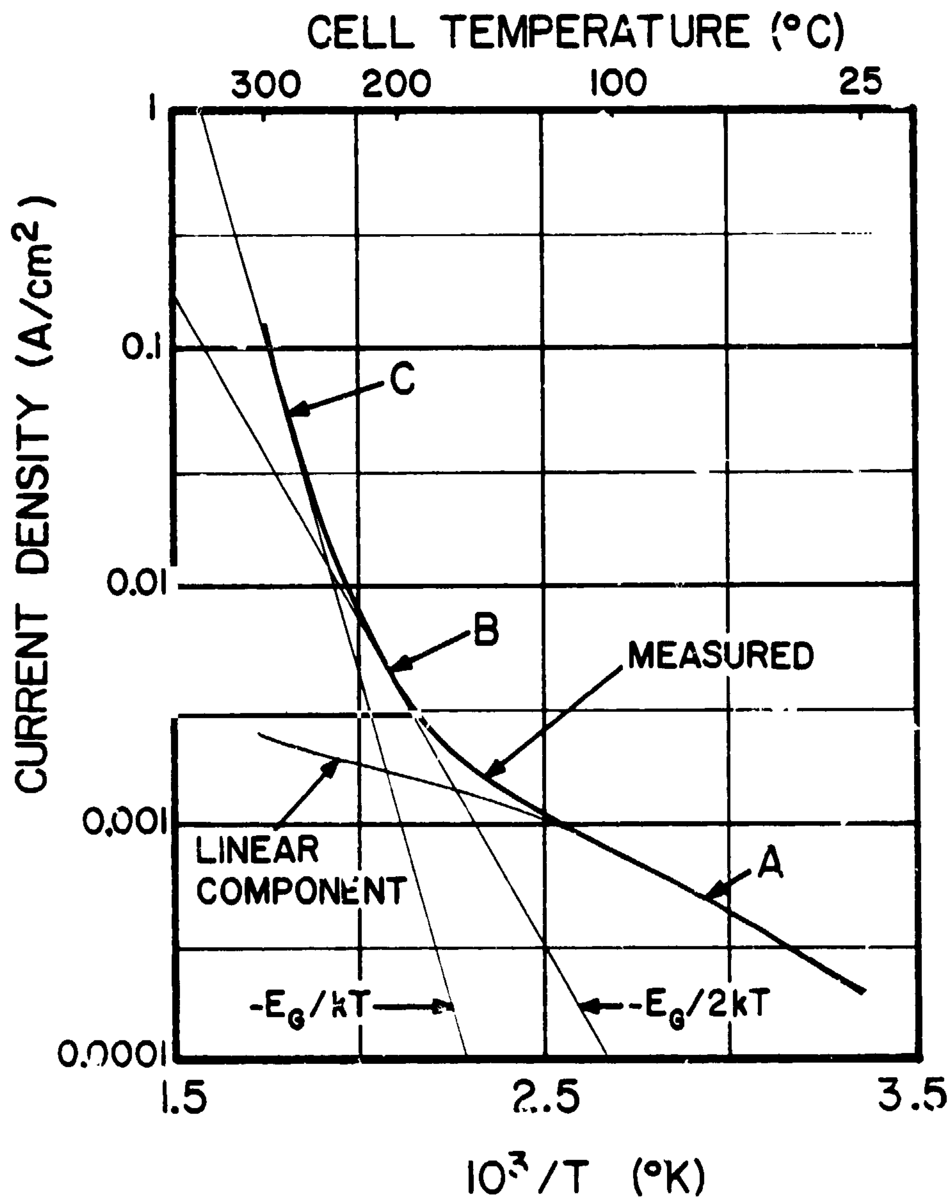


Figure 2.9 Typical Current Density vs. Temperature Curve for Second Quadrant (V = 1 volt)

reverse bias. Appropriate curve fitting techniques applied to a particular cell's data plotted in the form of Figure 2.7 and 2.9 led to an emperical equation for the current density of:

$$J = 1.933 \times 10^{-2} + 8.127 \times 10^{-6}T + 1.453 \times 10^{-7}\exp(4.585 \times 10^{-2}T) + 2.297 \times 10^{-5}V \exp(1.208 \times 10^{-2}T)$$

This expression indicates that for a given value of voltage there are three regions having different temperature dependence—a linear region and two regions of different exponential dependence. These three regions can be clearly distinguished in Figure 2.9.

Region A ($T < 180°C$): In this region the linear component of the current density dominates. The coefficient shown is 0.4%/°C which is of the order of the previously estimated 1%/°C.

Region B ($180°C < T < 250°C$): In this region the reverse current generated by traps is dominant and, as shown earlier, should have an exponential temperature dependence of $-E_g/2k$. As shown in Figure 2.9 the experimental data in this region displays this type of behavior exactly. It was also shown theoretically that the recombination-generation current for an abrupt junction, such as formed by a very thin diffused layer of high conductivity, should exhibit a $V^{1/2}$ voltage dependence. No explanation can be given at this time for the apparent linear voltage dependence observed.

Region C ($T > 250°C$): In this region saturation dominates. The lower boundary coincides with the intrinsic temperature for 10 ohm-cm material (base region). In this region the exponential slope approximate $-E_g/k$ as was theoretically predicted.

Avalanch multiplication effects were not taken into acccunt when fitting the curves of Figure 2.7, i.e. only the linear portion of these curves was used. This will affect the validity of the model near breakdown. However, in all the simulation runs to be described, voltages were kept well below breakdown. The breakdown voltage for this particular cell type was around 40 volts with a positive temperature coefficient of around 0.01 V/°C.

Values for the temperature dependence of the thermal conductivity can be obtained from the literature. In particular, E. C. Haesell (2.14) suggested the form $\kappa = \kappa_o/(T + T_o)$ with κ_o = 250 W/cm and T_o = 205°C and this was used in the simulations.

Other factors such as the specific heat, are not really independent of temperature. However these have a much weaker temperature dependence than the current density or the thermal conductivity in the region of interest. The model was only used to calculate steady state temperature. It might be desireable to include temperature dependence if the model were to be used for tranisent calculations. It should also be pointed out that the heat exchange between a solid surface and its surroundings is most likely not a linear process and may well be a function of factors other than temperature. The assumption of a linear dependent heat loss mechanism is an a priori assumption.

2.5 Boundary Conditions

For the unencapsulated cell boundary, conditions must be defined at the center and the edge of the cell. For the encapsulated cell boundary, conditions must be defined at these points and at the edge of the encapsulant.

Because rotational symmetry was assumed, the first derivative, , must equal zero at the center of the cell. This causes the deletion of the two

terms containing this derivative in the cell equation and of one term in the encapsulant equation. The rotational symmetry is a geat computational advantage as will be obvious after reviewing the Thomas algorithm, described in Appendix A.

Using the methods described in Section 2.3 Equation 2.2 may be evaluated at the boundary between the cell edge and the ambient or encapsulant. After some manipulation the following expression is obtained for the case of the encapsulated cell:

$$- \frac{1}{\Delta r} \frac{\partial T_s}{\partial r} + \frac{jV - h_{ce} (T_s - T_e)}{\kappa_s w_s}$$

$$+ \frac{(m + \frac{1}{2}) h_{ce} (1.5 T_{ms} - 0.5 T_{(m-1)s} - T_{(m+1)e}}{(m - \frac{1}{2}) \kappa_s r} = \frac{1}{\alpha_s} \frac{\partial T_s}{\partial t}$$

The equation for the unencapsulated cell is analogous. The third therm on the left is the amount of heat that leaves the cell via the cell edge. This amount is added to the next interval, the arrow in Figure 2.10 symbolizes this flow of heat.

The final boundary condition that has to be set is the end point of the encapsulant to be simulated. An extra ring is added to the encapsulant to indicate a continuation of it. Temperature was first extrapolated in the following fashion,

$$T_{N+1} = 2T_N - T_{N-1} ,$$

but this tended to introduce an oscillation at the edge of the modeled encapsulant. An obviously less accurate, but more stable solution was obtained by using

$$T_{N=1} = T_N .$$

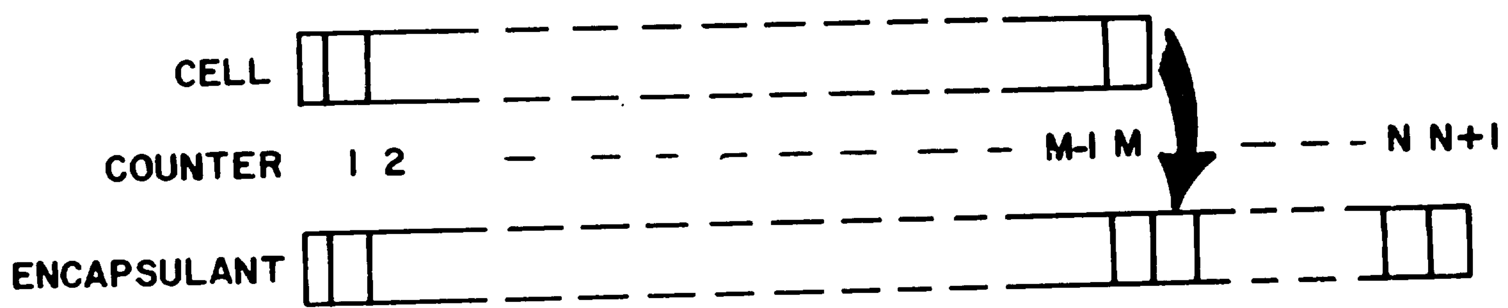


Figure 2.10 Encapsulated Cell Space Grid

In order to fit the results of the analysis to the measured temperatures and I-V curves, it was necessary to introduce a perturbation in the current density. This was done by introducing a Gaussian spatially dependent term in the exponent of the leakage current term of the form:

$$\beta + \gamma \exp\left(\frac{r^2}{\sigma^2}\right) .$$

The spread, σ , of the Gaussian was estimated from experiments to be one third of the cell radius. β and γ are arbitrary constants which can be used to fit the model to the results. The term β, if decreased, results in a cell that is less leaky that the typical cell, and consequently a cell with a higher base doping, as explained in Section 2.2. The factor γ was varied between 0 and 0.6, with a typical fit being obtained for a value of $\gamma = 0.4$. If γ is increased, a localized spot that is lighter doped, or for some other reason leakier, is introduced at the center of the cell.

Because of the assumption of circular symmetry the location of the leaky spot (defective area) is restricted to the center of the cell. This, of course, is not what is observed experimentally, but it is necessary in order to solve the equations. There is little loss of generalization, however, as a consequence of this assumption, mainly because the exact location of the hot spot does not have much effect on the cell's VI characteristic. The only possible exception might be a hot spot which formed at the cell edge. The model for encapsulated cells consists of two linear equations, and calculation of one I-V curve takes 7.5 minutes of CPU time using an IB 370/3033 computer.

Locating the hot spot anywhere else on the cell would take ten hours of CPU time by conservative estimate! In a similar fashion the model is restricted to circular cells, but this is presently the most popular form factor. As square cells proliferate their second quadrant performance will need to be studied.

The computer programs for modelling both unencapsulated cells and encapsulated cells in the second quadrant are listed in Appendix D.

2.6 Unencapsulated Cell Results

Measurement Methods--Figure 2.11 shows four experimentally measured I-V curves of cells in the second quadrant. These curves were obtained with an xy-recorder, a programmable power supply and a light source. The light source was a single ELH lamp overhead projector which gave an illumination level at the cell of 100 mW/cm^2. The power supply was programmed as a current supply and could be varied, either through the use of a variable resistor, or by means of a function generator. The slowest voltage ramp of the function generator was 5mV/sec, which was too fast to generate a steady state V-I curve, and consequently this method was used only for demonstrations.

Figure 2.12 schematically illustrates the method used. The front of the cell was placed junction side down on the $100mW/cm^2$ light source. The back of the cell was covered with a temperature sensitive phosphor which gave an indication of the temperature distribution. The phosphor used was #2090, fabricated by the U.S. Radium Corporation. Appendix C of the Second Annual Report contains information on temperature sensitive phosphors. These phosphors are not readily available in small quantities, but sample quantities for evaluation purposes may be obtained from the U.S. Radium Corporation.

Figures 2.13. (a), (b), (c), and (d) illustrates the use of phosphor coatings to detect the development of a hot spot. Figure (a) shows the cell

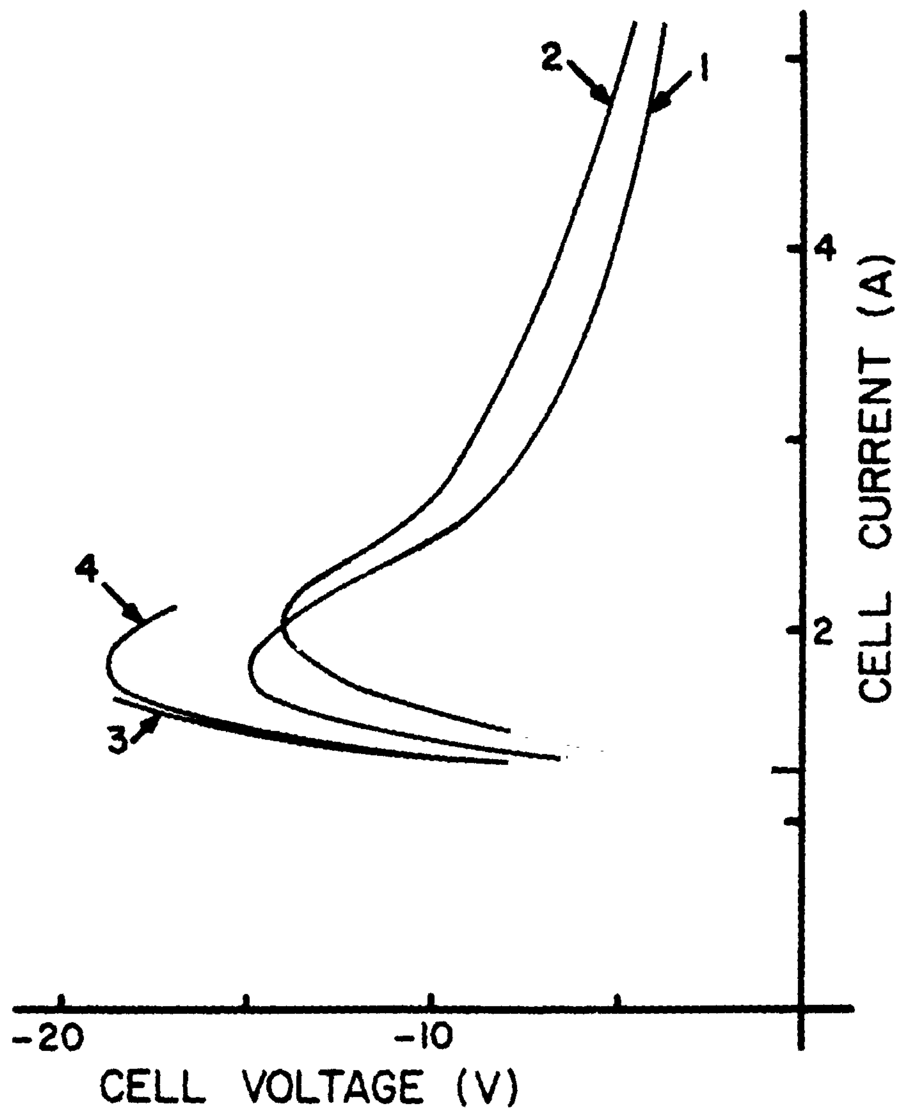


Figure 2.11 Measured VI Characteristics of Four Unencapsulated Cells in the Second Quadrant

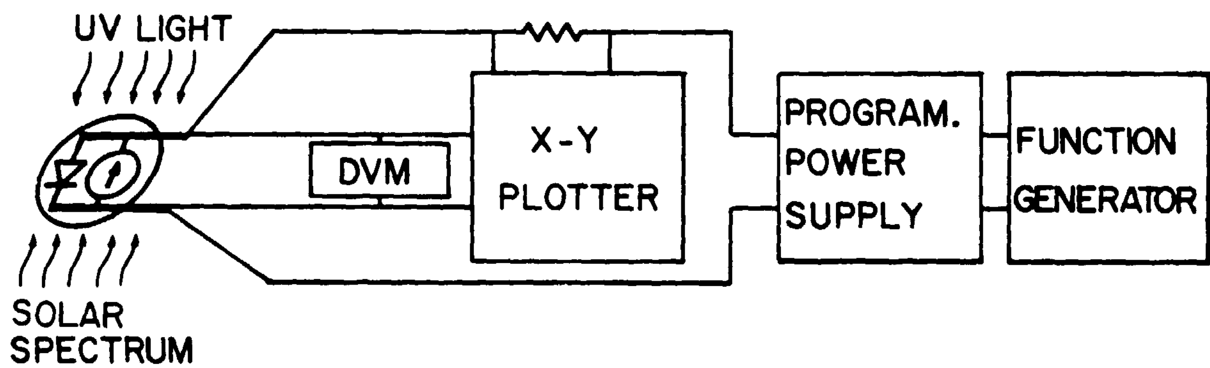


Figure 2.12 Schematic of Second Quadrant Measurement Method

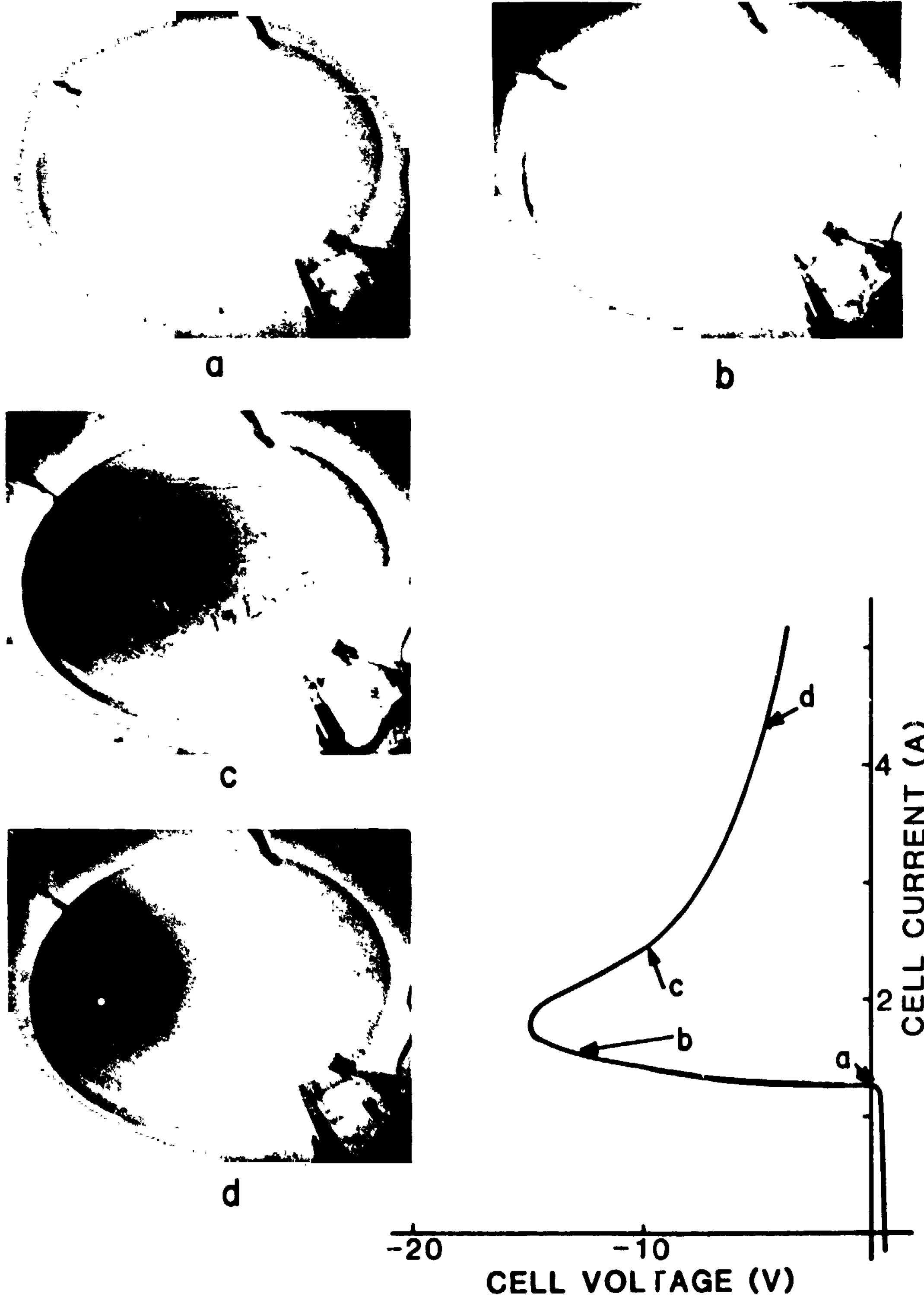


Figure 2.13 Phosphor Decoration Measurement Method

ORIGINAL PAGE IS
OF POOR QUALITY

at zero voltge, Figure (b) just before the knee in the second quadrant, Figure (c) just above the knee at moderate currents, and Figure (d) at high currents.

In some cases the peak temperature on the back of the cell was estimated using a wax temperature indicator, such as Omega's melting point standards. Good correlation was obtained between the two methods. Temperature measurements with melting point standards are very accurate, but are only available in 15°C intervals. Normally during measurement, a thermocouple was attached to the rim of the cell. Direct measurement of the hottest spot using a thermocouple was attempted, but was not successful. The problem was in maintaining good thermal contact for a prolonged period without cooling the hot spot by conduction along the thermocouple. It is possible to measure the temperature remotely using an infrared scanner as described by C. A. Lidback (2.15). There are, however, some problems with this technique. The lead on the back of the cell has emmissivity of only approximately 10%, so it is necessary to paint the back of the cell with a paint having better emmissive properties. Figure 2.14 shows a thermograph obtained in this manner.* The method is reasonably accurate ($\pm$10°C) and at the same time supplies a pictorial description of temperature variations across the cell in terms of graph tones. Drawbacks are that the positioning of this particular scanner over the light table with the cell on it is cumbersome, since both the light table and the scanner are bulky objects.

Model Prediction—As has been pointed out the model permits calculation of the IV characteristic curve. For unencapsulated cells the IV curve was

---

* The IR scanner used for this work was a model 900 manufactured by UTI and owned by North Carolina State University, Raleigh, NC.

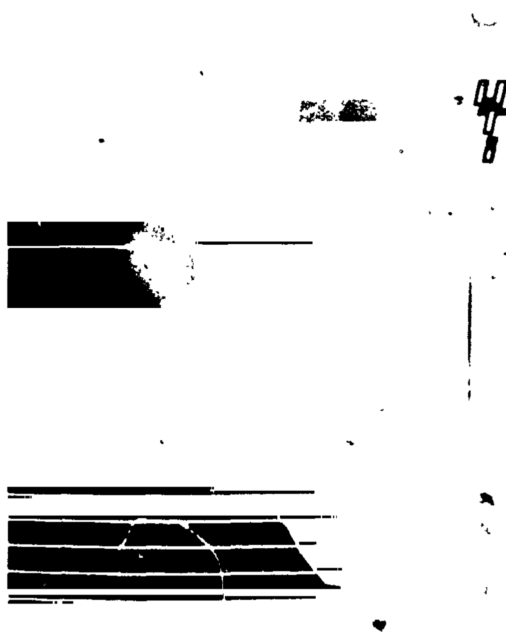


Figure 2.14 IR Scanner Thermograph of an Unencapsulated Cell in the Second Quadrant

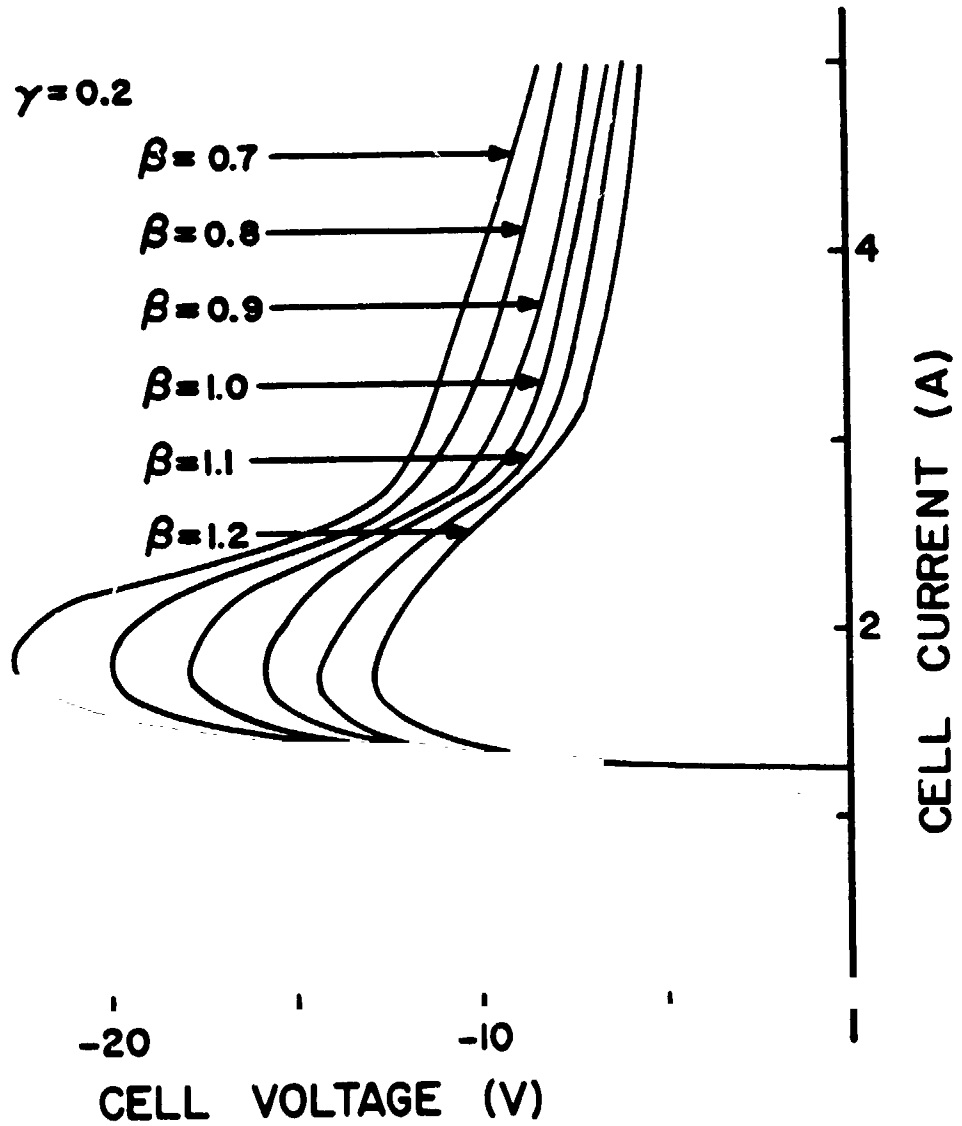


Figure 2.15 Calculated VI Characteristics of an Unencapsulated Cell for $\gamma = 0.2$

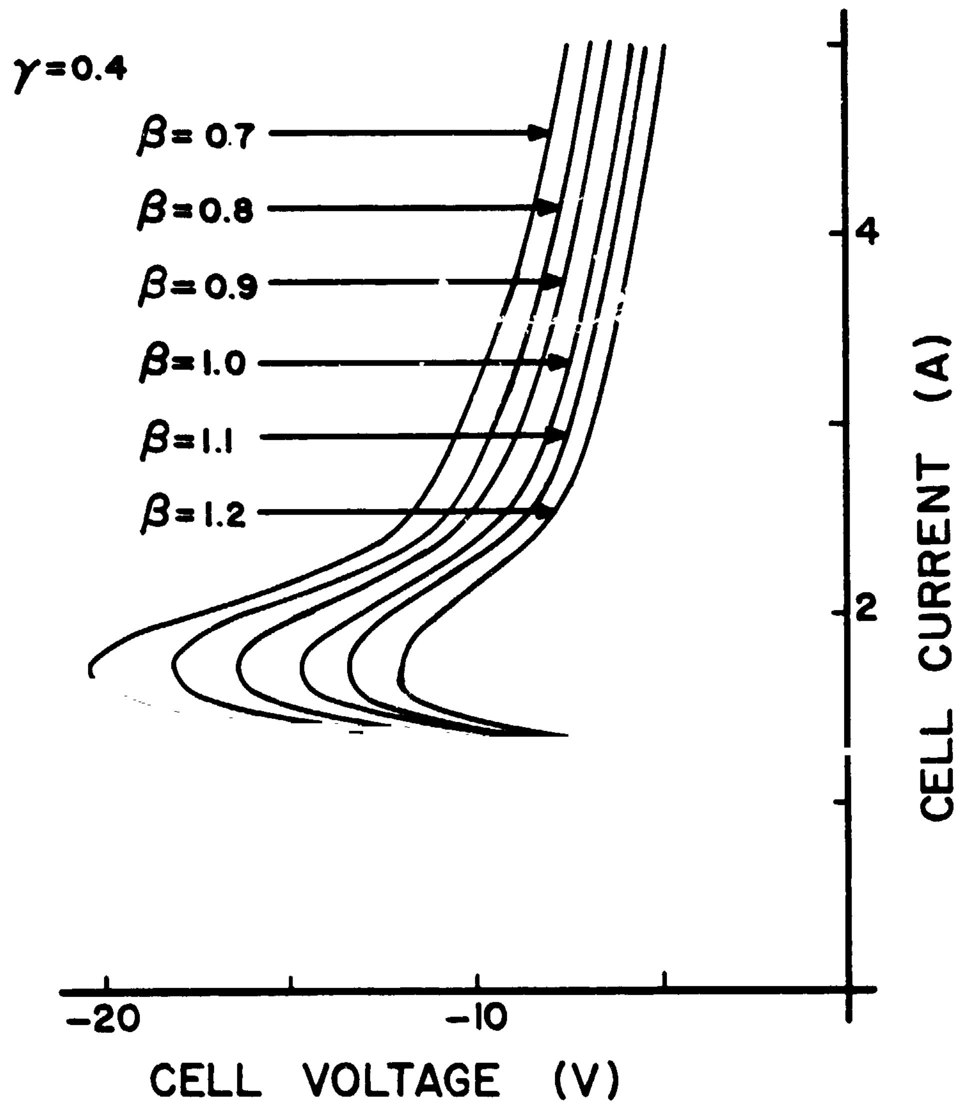


Figure 2.16 Calculated VI Characteristics of an Unencapsulated Cell for $\gamma = 0.4$

obtained from 40 points calculated at 0.1A intervals starting at a current of 1.3A. Figure 2.15 shows IV curves for which the amplitude of the Gaussian perturbation, γ, is 0.2, while the temperature dependent parameter of the leakage current, β, is varied between 0.7 and 1.2 in intervals of 0.1. Figure 2.16 shows the same but for $\gamma = 0.4$. Trace 1 in Figure 2.11 is from the cell used to model the current as function of temperature and voltage (Cell 1), so this curve should closely match a curve with $\beta = 1.0$. Figure 2.17 shows the measured curve of Cell 1 superimposed on the calculated curves for $\beta = 1$. For $\gamma = 0.4$ there is satisfactory fit between the calculated curve and the measured curve up to .5A, approximately two times I_{sc}. Figure 2.18 shows a diagram of Cell 1 with a description of the visual observations made as the current through the cell was increased. Under the UV lighting conditions used, the transition point for the phosphor was approximately 200°C. In Figure 2.18 this transition is indicated as the appearance of a hot spot. The coexistance of two hot spots at moderate current levels, followed by the disappearance of one at higher currents, is often observed. A shift towards the back current probe is not uncommon at high current loads, and probably is due to a resistive voltage drop in the base and back metalizations of the solar cell. Measurements of the resistive voltage drop gives value in the order of .2 - .3 V, an effect that was not incorporated in the model, as mentioned earlier.

Table 2.1 compares measured and calculated temperatures. The agreement between the model and the measured values is remarkable, considering this was not a cell with a hot spot centered in the middle of the cell, as was assumed in the model.

Figure 2.19 shows the calculated temperature profiles along a radius, for $\beta = 1$ and $\gamma = 0.4$, at different current levels. The effect at the edge of the

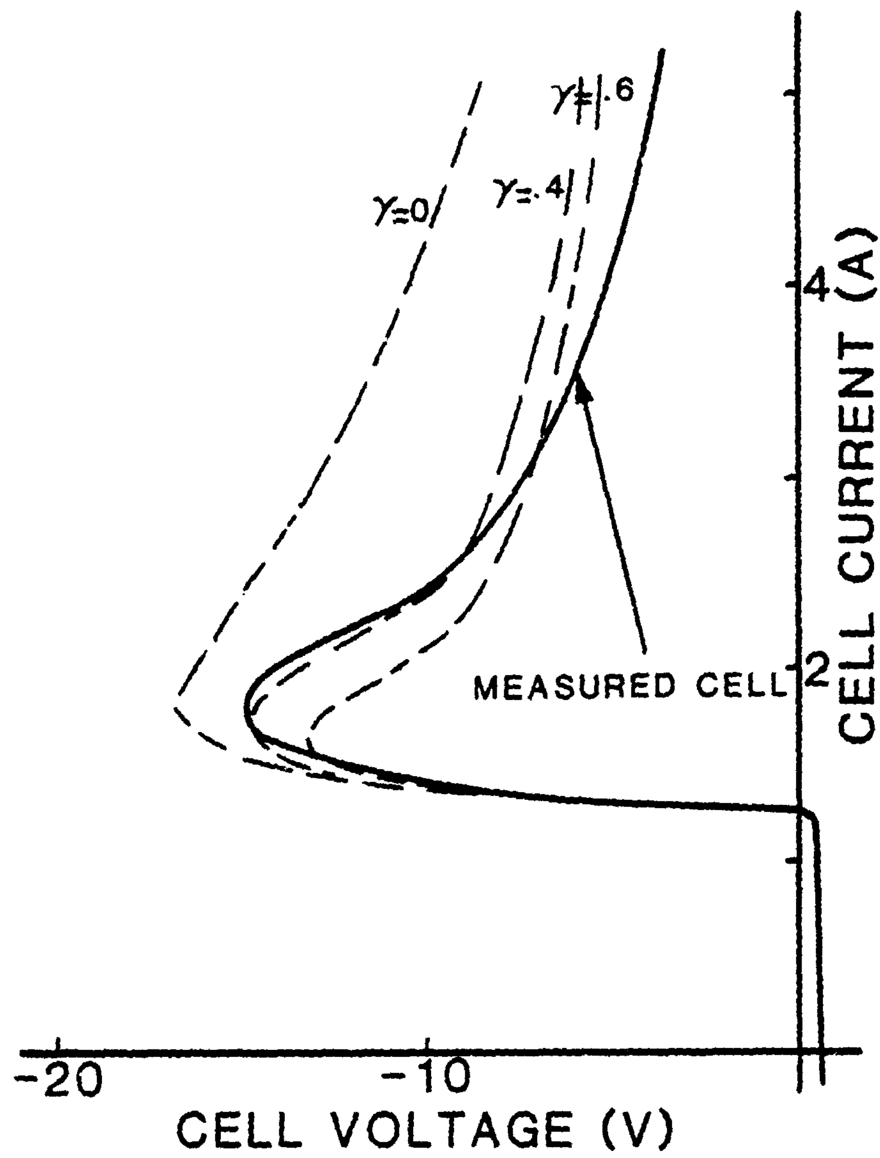
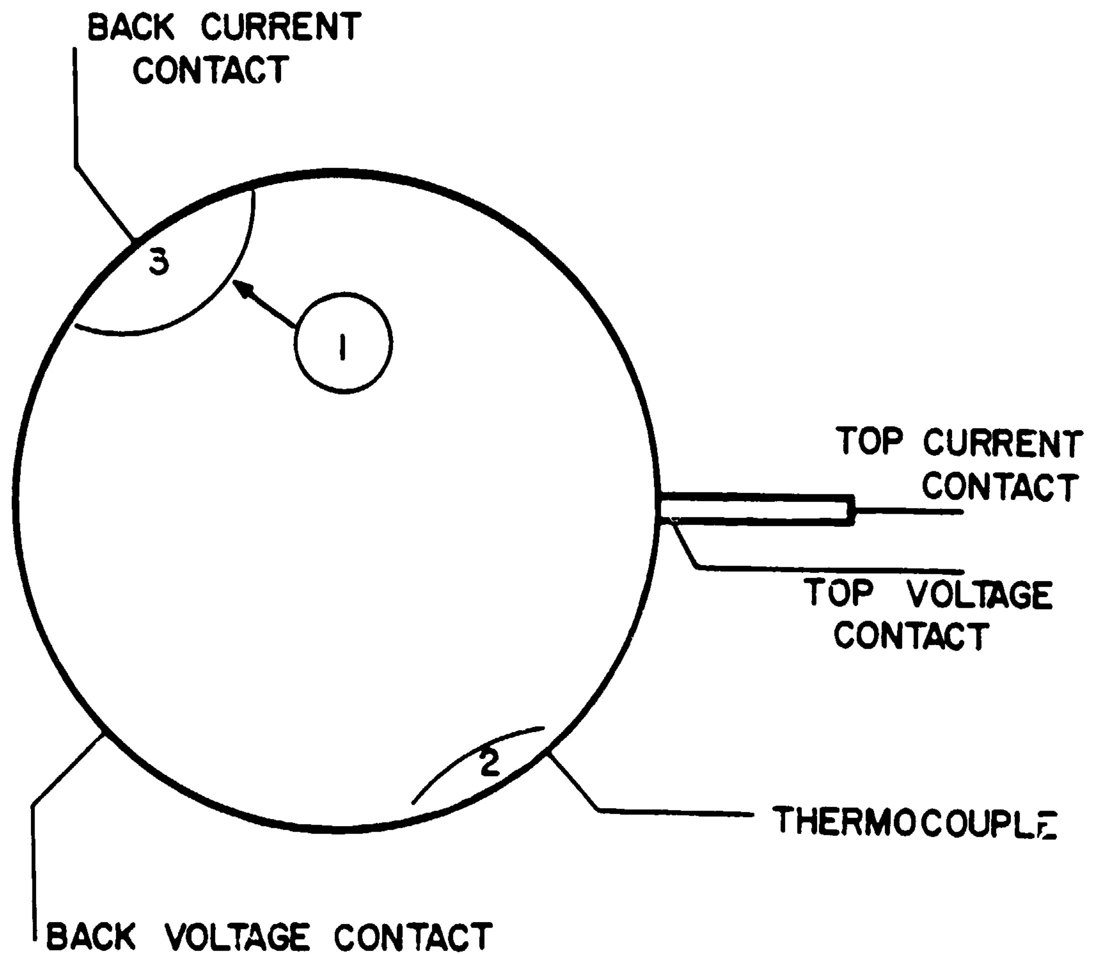


Figure 2.17 VI Curve of Cell 1 Compared with Calculated Curve for δ= 1

CURRENT (A)	TEMPERATURES (^{o}C)	
	MEASURED	MODEL
1.3	100	90
1.4	140	138
1.6	180	182
1.8	200	206
2.0	230	224
2.5	270	253
3.0	280	268
4.0	310	287
5.0	325	304

Table 2.1 Measured and Calculated temperatures at the Hot Spot of an Unencapsulated Cell



1. APPEARANCE OF HOT SPOT AT I=1.9A
2. SECONDARY SPOT AT I=1.9A
   DISAPPEARS AT I=2.1A
3. SHIFT OF SPOT AT I=3.5 A

Figure 2.18 Diagram Illustrating Changes in the Physical Appearance of a Cell as Current Increases

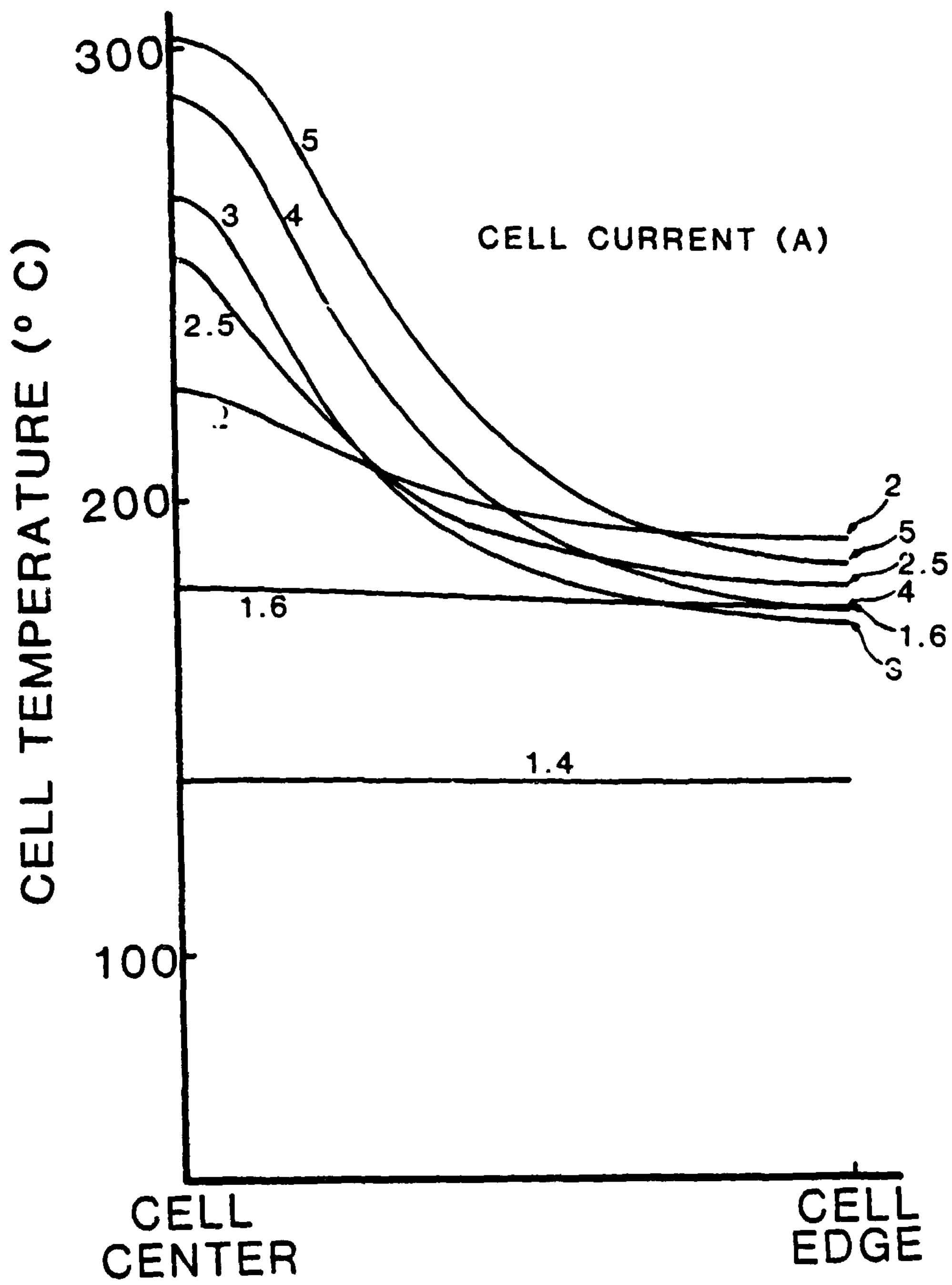


Figure 2.19 Calculated Temperature Profile for [illegible]= 1 and [illegible]= 0.[illegible]

CURRENT (A)	TEMPERATURE (^oC)	
	MEASURED	CALCULATED
1.4		136.9
1.43	149	
1.5		161.7
1.585	180	
1.6		176.5
1.7		185.3
1.767	199	
1.8		189.8
1.856	204	
1.9		190.9
1.938	200	
2.0		190.1
2.023	196	
2.1		188.2
2.295	176	
2.2		185.9
2.3		184.6
2.36	168	
2.4		182.2

Table 2.2 Measured and Calculated Temperature at the Edge of an Unencapsulated Cell

cell that had been observed experimentally can be seen in Table 2.2. The higher temperature observed at lower currents can be attributed to secondary hot spot formation in this region. The lower temperatures at higher currents, on the other hand, are probably caused by the geometry of the hot spot and how it is affected by the thermocouple.

Comparing the VI curves of Figure 2.15 and 2.16 with the remaining three traces of Figure 2.11 permits an estimation of the parameters for these curves. A reasonable fit can be obtained for cell 2 at $\beta = 1.1$, $\gamma = 0.2$ and for cell 3 and 4 at $\beta = 0.8$, $\gamma = 0.3$. Cell 3 shorted at the end point of the trace, and as far as could be determined this was caused by molten lead at the table attachment point. For cell 4 the measured temperature at 1.58A was between 206°C and 220°C, while the calculated center temperature at 1.5A was 190°C and at 1.6A was 215°C. At 1.76A the measured temperature is smaller than 258°C while the calculated temperatures at 1.7A and 1.8A are respectively 234°C and 250°C. Both these observations indicated that the current function could be modified to provide a reasonable fit of the data.

Parametric Changes -- It is possible to vary a number of parameters within the model. For example, variations could be made in light intensity, partial shading (radial symmetry only), temperature, and cell thickness. Figure 2.20 shows the influence of varying cell thickness on the VI curve. Thicknesses of 0.02, 0.04 and 0.08 mm were chosen. As the cell thickness becomes smaller, the lateral transport of heat becomes smaller, causing a lower breakdown voltage. The calculated temperature profiles at 2A, shown in Figure 2.21, illustate this. Clearly the temperature distribution becomes less uniform at smaller cell thickness. Experiments with a cell, which was artificially thickened by attaching a thin brass foil to its back, confirmed this. Breakdown voltage in this case was increased from 15V to 21V.

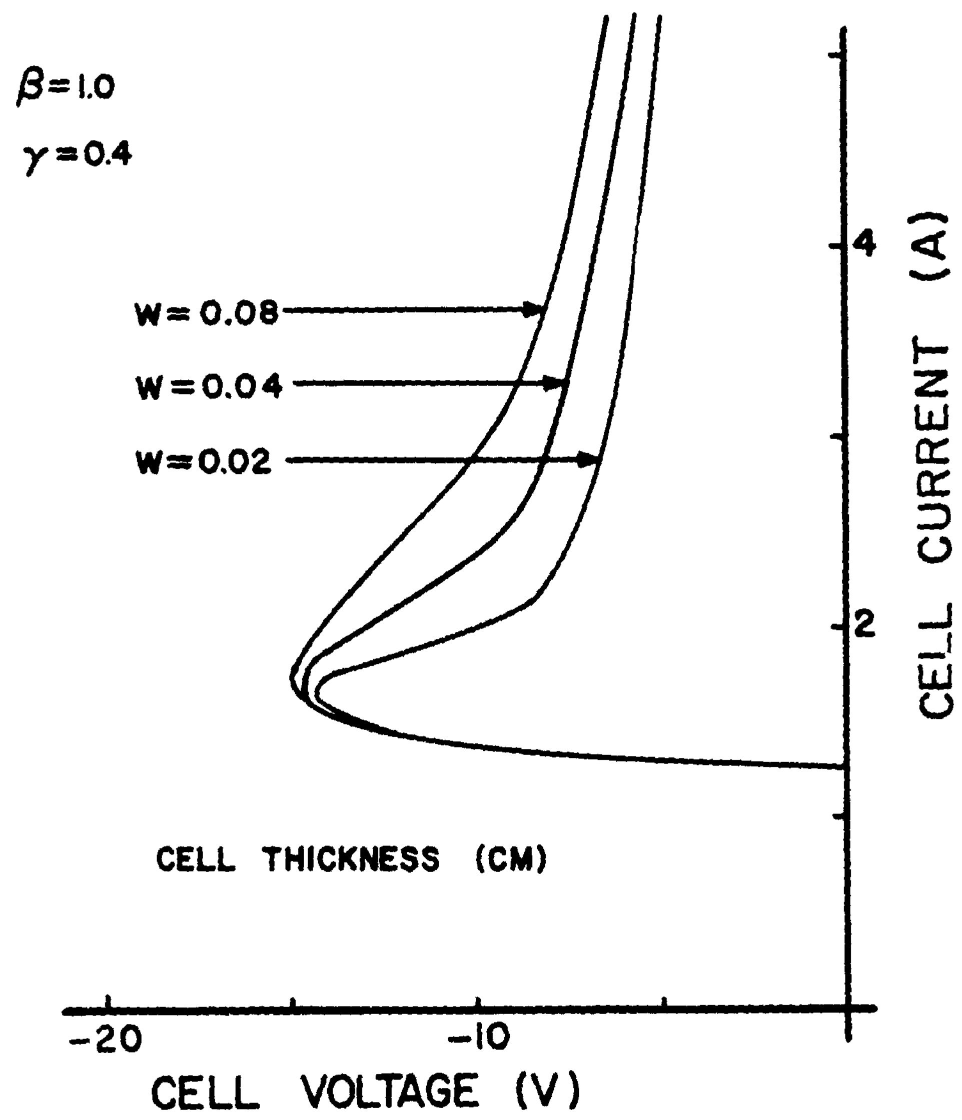


Figure 2.20 Calculated VI Curves for Different Cell Thicknesses

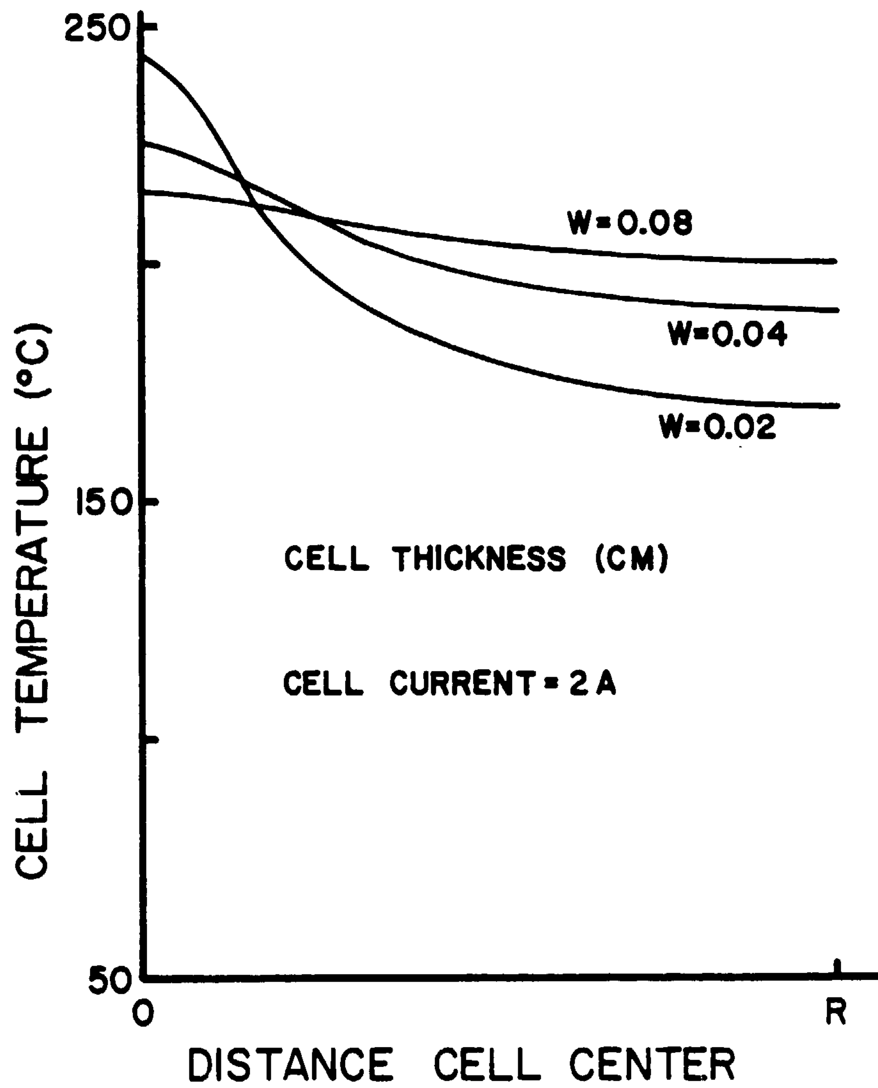


Figure 2.21 Calculated Temperature Profile for Different Cell Thicknesses

2.7 Encapsulated Cell Results

Construction — Figure 2.22 illustrates the construction of the encapsulated minimodules used in the investigation. These minimodules were made by Springborn Laboratories, Inc., for the Jet Propulsion Laboratory LSA program. The cells were manufactured by Solar Power Corp. for the JPL Block I procurement, and were of the same type that had been studied earlier by Clemson when investigating accelerated reliability testing in the first quadrant.

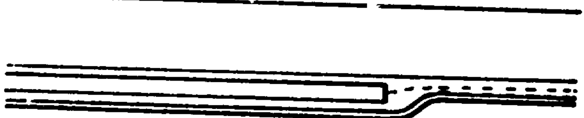
The glass superstrate minimodule consisted of the glass superstrate, a layer of clear ethylene vinyl acetate (EVA), the solar cell, a layer of EVA containing white pigment with an unwoven glass matt and covered by aluminimum foil. The glass matt is to ensure insulation between the cell and the aluminum foil and to add to the mechanical strength.

The minimodule with the masonite substrate consisted of a film of korad, a layer EVA, the cell, EVA with pigment and glass matt, a Super Dorux substrate, and an EVA layer. The korad film is a UV screen while the bottom EVA layer is to protect the masonite from the elements.

No data is available on the temperature variation of the EVA thermal conductivity. Most probably there is a change in the thermal conductivity at higher temperatures when the EVA softens and actually becomes liquid around 105°C. Another transition point is around 350°C when decomposition starts. Figure 2.23 shows a partial trace made with a thermogravimetric analyzer (TGA)* of a sample of EVA. Weight loss of the EVA began around 120°C where the sample probably started to loose water. At 225°C all the water was gone. Decomposition began around 275°C and the sample was completely decomposed at 550°C. An additional temperature effect is that the EVA becomes less clear

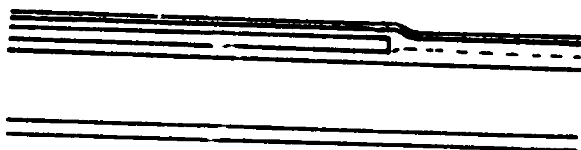
* Model 951 Thermogravimetric analyzer (Dupont)

GLASS SUPERSTRATE



	THICKNESS (MM)	THERMAL CONDUCTIVITY (W/° C.CM) X 10^{-3}
GLASS	3.175	8.37
EVA	0.445	2.62
CELL	0.495	k = k(T)
EVA ✿	0.254	2.62
ALUMINUM	0.025	2360

MASONITE SUBSTRATE



	THICKNESS (MM)	THERMAL CONDUCTIVITY (W/° C.CM) X 10^{-3}
KORAD	0.076	1.93
EVA	0.432	2.62
CELL	0.495	k = k(T)
EVA ✿	0.254	2.62
SUPER DORLUX	3.175	1.87
EVA	0.254	2.62

✿ EVA WITH GLASSMAT AND WHITE PIGMENT

Figure 2.22 Minimodule Construction Details

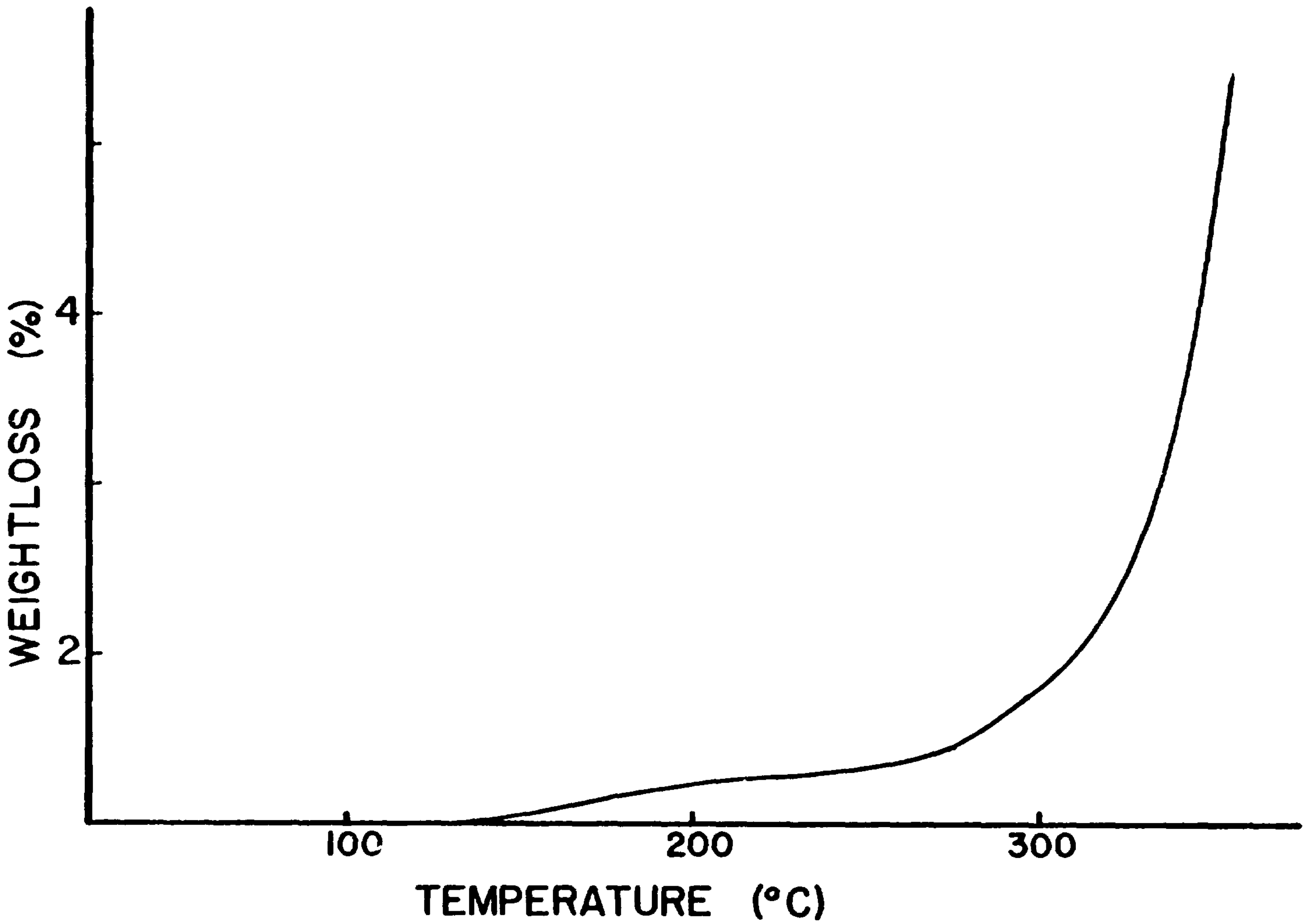


Figure 2.23 Thermogravimetric Analysis of EVA

due to carbonizing, causing a loss of transparency and a drop in the short circuit current. These effects were not taken into account in the model, however. The model was run primarily to solve steady state problems. In these cases the thermal diffusivities of the silicon and the encapsulant were set to fixed values to ensure fast convergence. This makes the value of the time increments uncertain, but for steady state problems this is not important since the solution is taken at $t = \infty$.

2.8 Measurements and Results from the Model

Figure 2.24 and 2.25 show VI-curves from the glass encapsulated cells and the masonite encapsulated cells respectively. In order to relate these curves to cell temperature, as was done for the unencapsulated cells, it is necessary to have an accurate method of measuring the cell temperature. Unfortunately this could not be easily accomplished. In some cases a thermocouple was mounted in the center of the masonite substrate touching the back of the cell. This procedure, however, infuenced the gas containment capabilities of the encapsulant and perturbed the experiments. Some thermocouple temperature measurement of the front and back surfaces were attempted, but they only gave a qualitative indication of the cell temperature. The development of more accurate temperature measuring techniques, such as using an array of very low mass thermocouples encapsulated with the cell, was not attempted. An infrared scanner is not a particularly useful tool for measuring the temperature of encapsulated cells since it measures the surface temperature of the encapsulant rather than the cell itself. Furthermore, as described in Section 2.6, the accuracy of the instrument depends on the surface emissivity and it would be necessary to paint the cell with black paint. An infrared scanner, however, is a powerful tool for locating reverse biased solar cells in arrays deployed in the field. Phosphor decoration also measures only encapsulant surface temperatures. The best indicator of cell temperature was the melting

of the solder metalization which could be observed at approximately 170°C. This effect provided a temperature check which allowed correlation with unencapsulated cell results. A total of four minimodules of each construction type were tested in the second quadrant. Two of the glass superstrate type (A, B) and three of the masonite substrate type (C, D, E) are shown in Figures 2.24 and 2.25. Two additional cells having glass superstrates shorted during testing. The first shorted when the lightsource malfunctioned while the cell was biased at 15.7 V and 1.366 A. The generated heat was sufficient to shatter the top plate of the overhead projector light source. The shock of the explosion probably caused the short to occur in the mini module. The second cell had a manufacturing defect which caused the juntion to short at approximately 7 volts reverse bias. The other masonite substrate minimodule which was tested, but not shown in Figure 2.25, was accidentally broken.

The voltage on Cell A (Figure 2.24) was gradually increased from zero. At 10.7 volts reverse bias the superstrate cracked due to thermal stress. These cracks propagated with time under continued stress. Figure 2.26 shows the minimodule two hours after the cracks first started to form. At 15 V significant delamination had occurred at the back of the cell and a gas filled bubble had formed. At 22 V the cell shorted, probably at a crack location. Delamination of the top EVA coating can also be seen in Figure 2.26. Lead melting was observed at approximtely 18 volts.

The glass superstrate of Cell B did not crack when subjected to the same reverse bias stressing. Instead the cell was pushed downward at a reverse bias of 21 volts by gas that formed between the cell and the glass. At this point the solder coating was molten. At 18 V a bubble had formed in the back EVA. The glass surface did not exceed 149°C during the test. Figure 2.27(a) and 2.27(b) show the onset of gas formation at the top of the cell. The photographs were taken a few seconds apart.

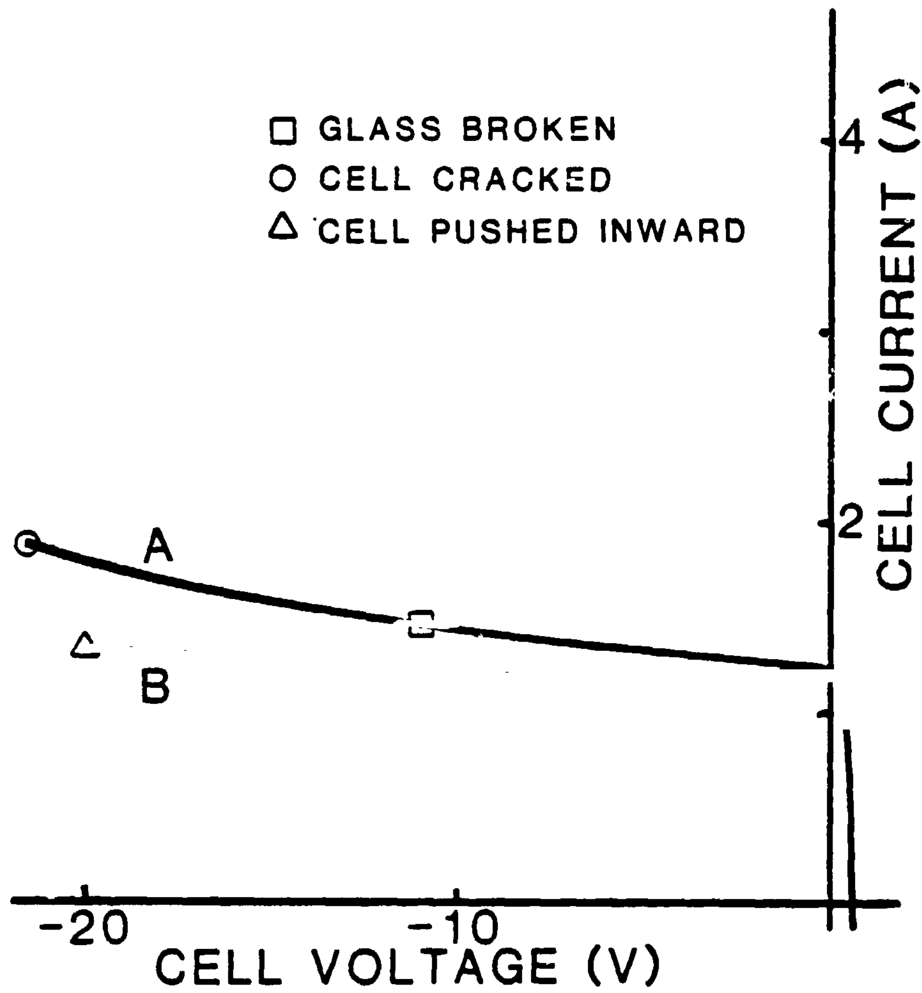


Figure 2.24 Measured VI Curve of Glass Encapsulated Cells

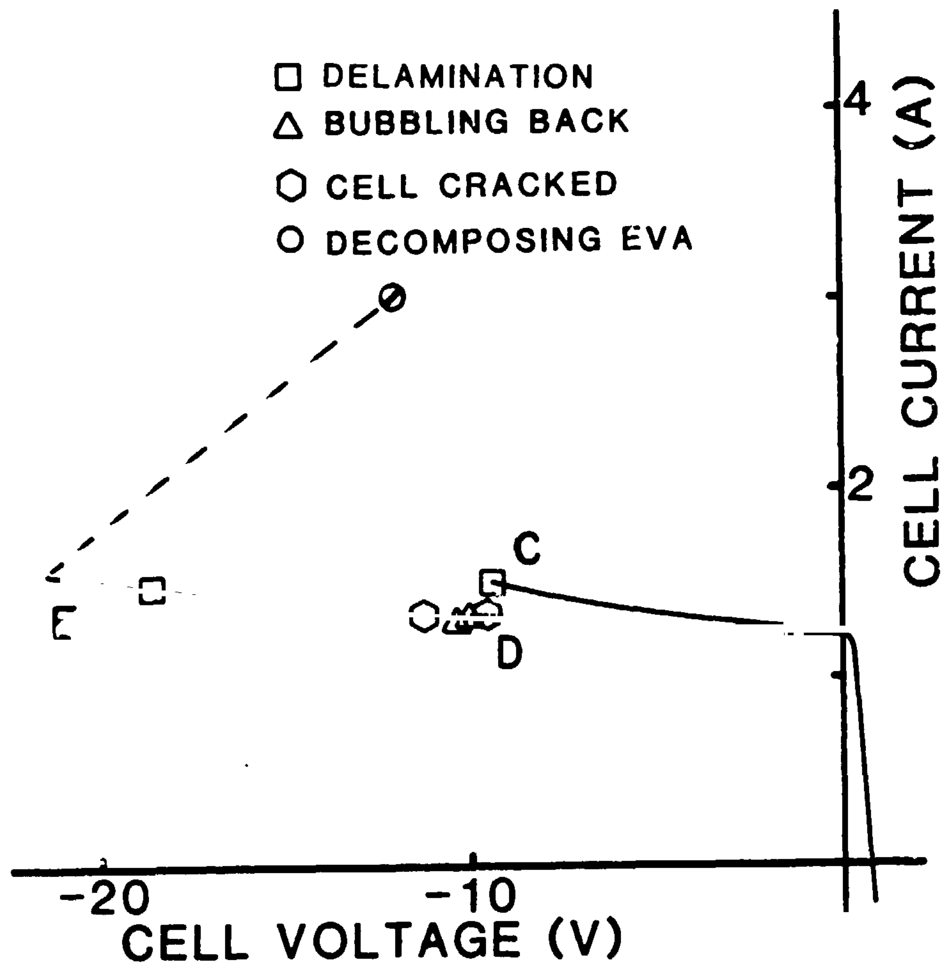


Figure 2.25 Measured VI Curve of Masonite Encapsulated Cells

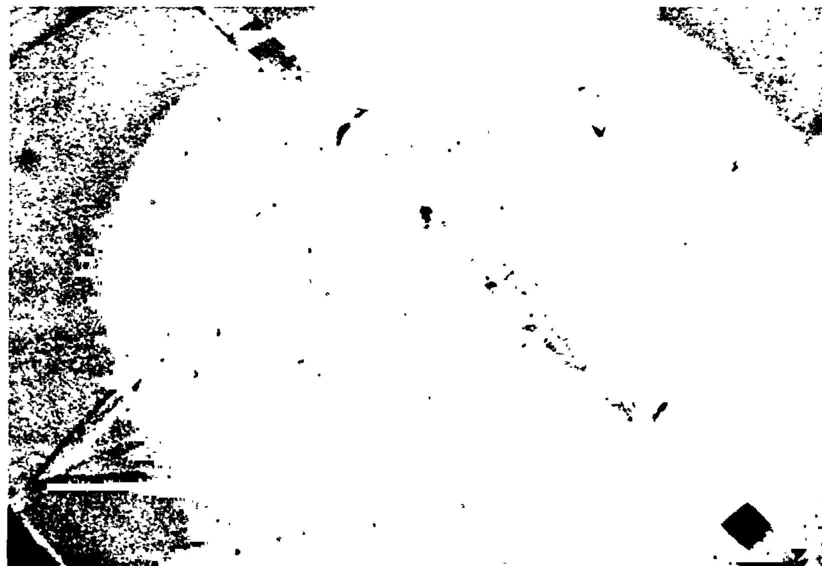
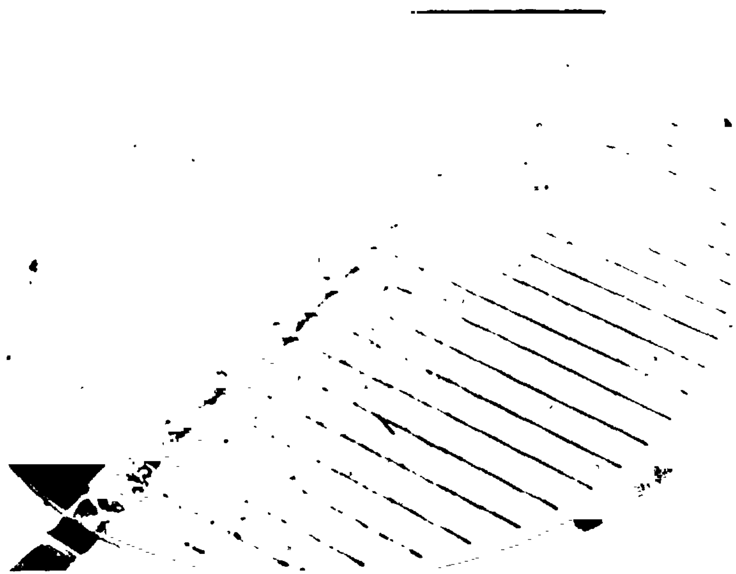
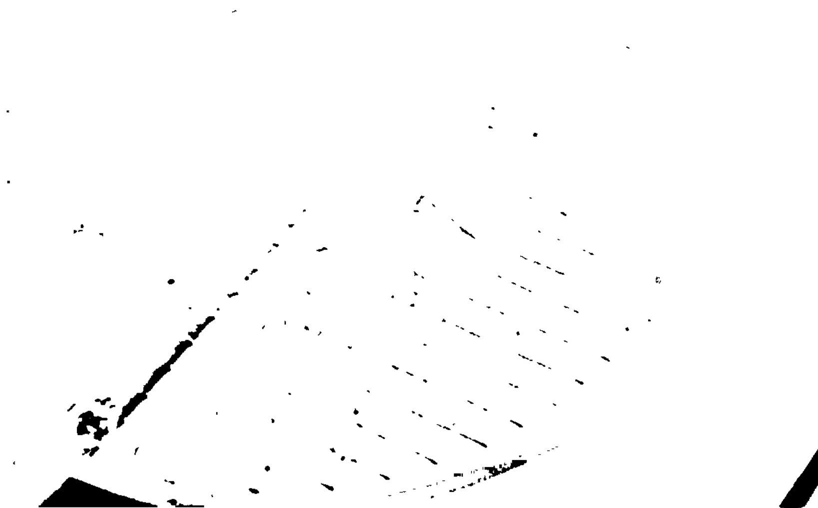


Figure 2.26 Minimodule with Glass Superstate Cracked



a) Start of Gas Formation



ORIGINAL PAGE IS
OF POOR QUALITY

b) Approximately 10 seconds later

Figure 2.27 Result of Gas Formation on Minimodule

Masonite substrate Cell C showed degradation of the back EVA at 7V reverse bias. A thermocouple, which had been inserted through a small hole in the masonite and was touching the back of the cell, registered 120°C. At 9.4 V the thermocouple indicated 146°C and delamination had occurred between the top of the cell and the top EVA coating as a result of the generation of gas. This resulted in the permanent deformation of the korad. Figure 2.28 shows the appearance of the cell after it had cooled to room temperature. Since the bubble clearly was a failure of the encapsulant, the module was not subjected to further stesses.

Masonite substrate Cell D was "lifetested" at 10 V reverse bias for 3 hours. Figure 2.29(a) shows the severely degraded back coating of EVA after this stress. The cell was cracked across the middle as can be seen from the front view in Figure 2.29(b). A possible explanation is that water evolving from the masonite caused the cell to crack.

Masonite substrate Cell E was the only encapsulated cell that remained reasonably functional after passing the knee of the IV-curve. The IV-curve was obtained by stepping in voltage rather than current, but with the power supply limited to 3A maximum current. No other adjustments were made in the power supply settings. At 10 V degradation of the back EVA was again observed. At 12 V the cell cracked and melting of the solder was observed (170°C). At 16 V small bubbles at the end of the crack appeared. At 18 V additional bubbles formed completely around the rim. A small bubble that had been formed during encapsulation got bigger, and a strong burned wood odor became noticeable. A hot spot developed at just over 21 V. At this point the smaller part of the cracked cell probably became internally disconnected, because it appeared noticeable cooler than the larger part. The EVA was observed to turn slightly brown all over the cell and dark brown at the hot spot. The EVA seemed to flow away from the hot spot and the test was

Figure 2.28 Cell C After Second Quadrant Operation

a) Back

b) Front

Figure 2.29 Cell D After Second Quadrant Operation

a) Front

b) Back

Figure 2.30 Cell E After Second Quadrant Operation

terminated for fear of fire. Figure 2.30(a) shows the front of the minimodule, with the charred hot spot clearly visible and Figure 2.30(b) shows the severely degraded back. Some EVA actually flowed from the minimodule to the support underneath.

Since only one cell survived beyond the knee of the IV-curve it was not easy to fit the model to the observations. In addition, the end points of many traces did not represent true steady state conditions since they were not in thermal equilibrium beyond breakdown. Furthermore, many cells were physically broken near the end point—a condition that would give a discontinuity in the thermal conductivity.

Figure 2.31 shows calculated IV curves for the glass superstrate module for $\gamma = 0.2$ for different values of β. Figure 2.32 shows the same for $\gamma = 0.4$. Figure 2.33 and 2.34 show the same curve for the masonite substrate module, again for $\gamma = 0.2$ and $\gamma = 0.4$.

A reasonable fit for Cell A is $\beta = 0.8$, $\gamma = 0.2$, for Cell B: $\beta = 0.7$, $\gamma = 0.2$, for Cell C: $\beta = 1.2$, $\gamma = 0.5$ and for E $\beta = 0.7$, $\gamma = 0.2$. Cell D would fit between $\beta = 0.7$ to $\beta = 1.0$ and $\gamma = 0.2$ to $\gamma = 0.4$. Table 2.3 compares the estimated temperatures with the nearest calculated ones. Agreement between the calculated and observed values was reasonably good, but not as good as for the unencapsulated cells. This can be attributed to a combination of measurement difficulties and modeling innaccuracies.

It is interesting to compare values calculated by the Clemson model with data supplied by the Jet Propulsion Laboratory at the Encapsulation Workshop held 9/23/80 in Pasadena. Figure 2.35 compares JPL calculations with values compiled from the model using glass encapsulation and $\beta = 1.0$, $\gamma = 0.4$. The two curves are in excellent agreement over the full range of JPL data. Extending the calculations to higher values of P/P_{MAX}, the Clemson model shows

CELL	OBSERVATIONS				B/	MODEL		
	Observation	Voltage (V)	Current (A)	Temperature (°C)		Voltage (V)	Current (A)	Temperature (°C)
A	Cracked glass	10.7	1.49	>97	.8/.2	11.75	1.35	124
	back delamination	15	1.55			17.52	1.45	169
	lead melting	18	1.67	>170		19.31	1.50	186
B	Lead melting	19	1.35	>170	.7/.2	19.13	1.45	180
	Severe gas formation	>21	1.45			21.22	1.50	199
C	Start degrad. back	7.0	1.39	120	1.2/.4	6.92	1.35	114
	front delamination	9.4	1.51	146		9.99	1.50	151
D	Lifetest (3 hours) back failure	10.0	1.33		.8/.2	9.37	1.35	133
E	Degradation of back	10.0	1.34	>120	.7/.2	10.02	1.35	139
	cell crack, lead melt	12.0	1.36	>170		13.06	1.40	168
	gas formation	18.0	1.43			16.82	1.50	212
	second breakdown	21.0	1.54			19.99	1.80	285
	hot spot	10.7	2.98	>300		11.58	3.00	419

Table 2.3 Estimas ed Temperatures at Observation Points Compared with Calculated Temperatures for Encapsulated Cells

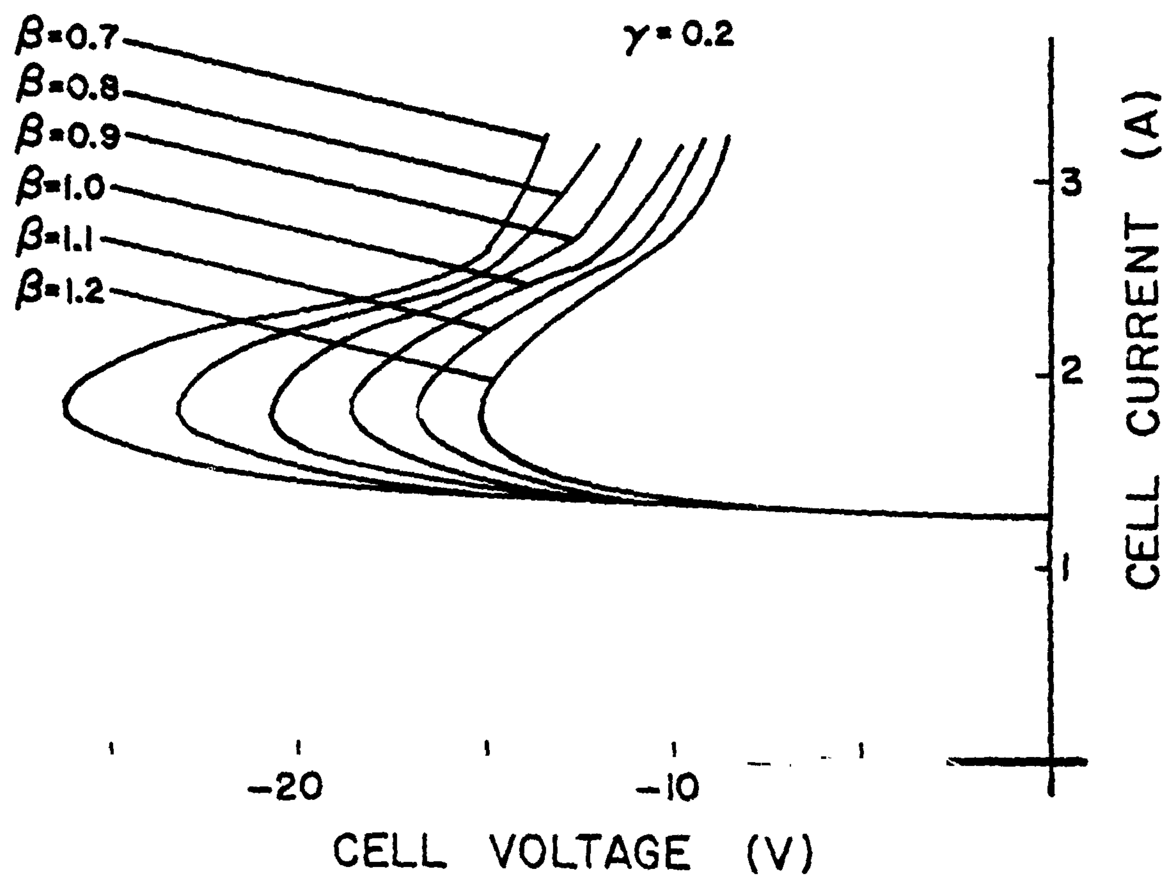


Figure 2.31 Calculated VI Characteristic of a Glass Encapsulated Cell for γ = 0.2

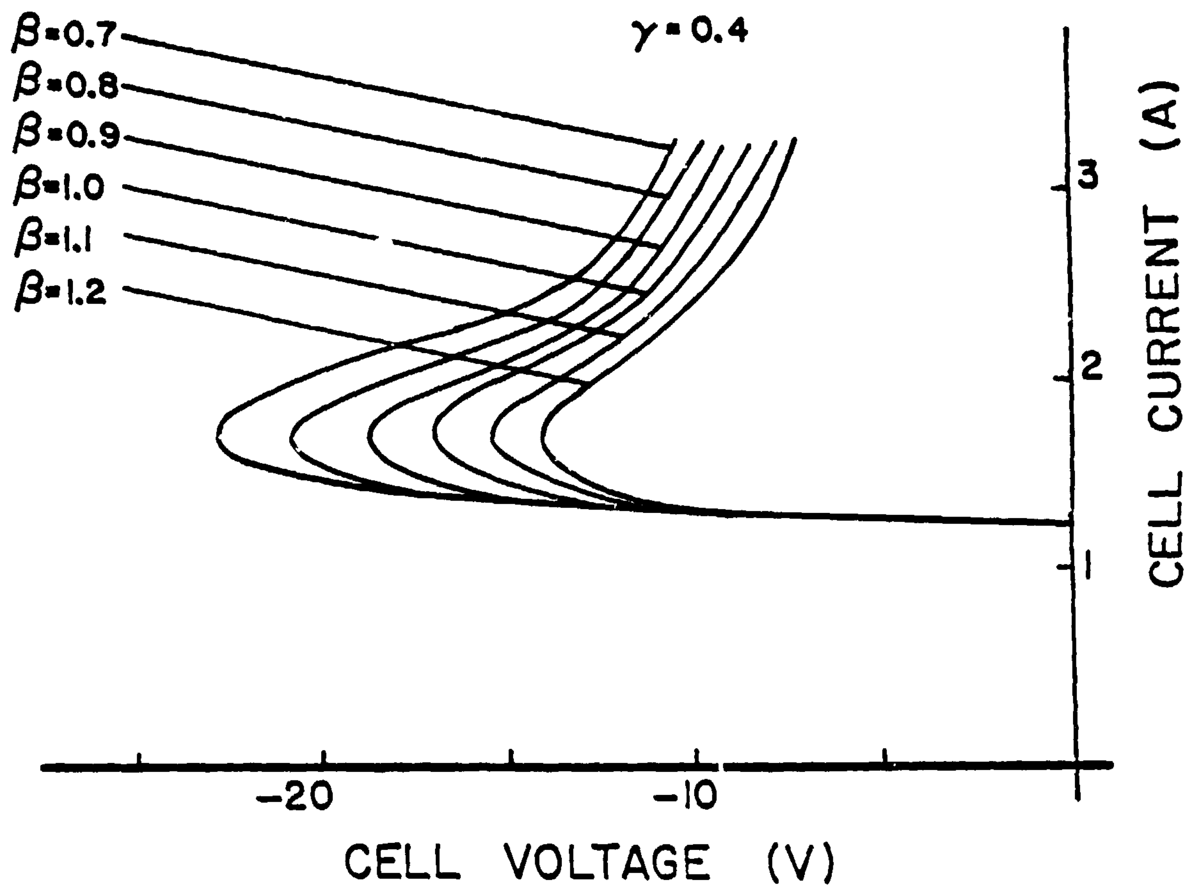


Figure 2.32 Calculated VI Characteristic of a Glass Encapsulated Cell for $\gamma = 0.4$

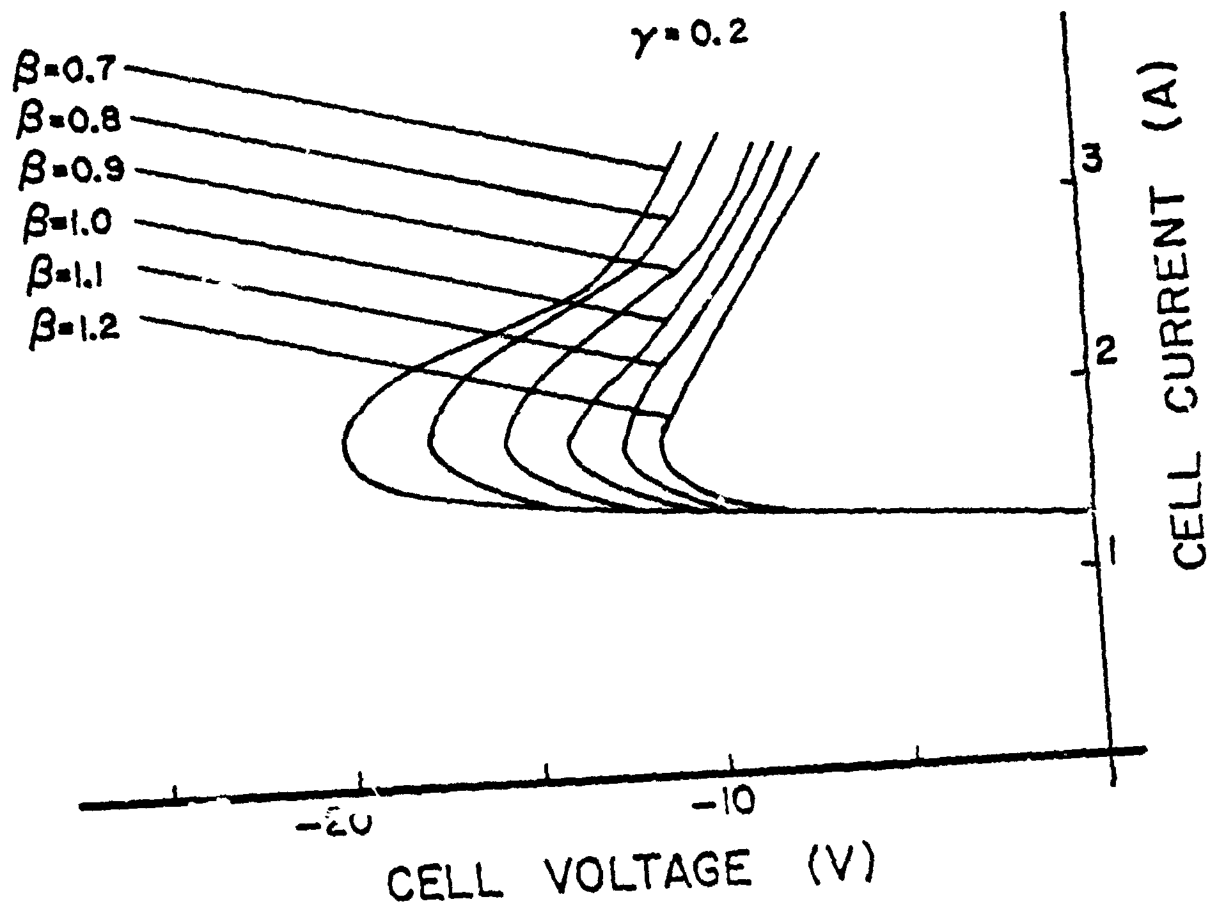


Figure 2.33 Calculated VI Characteristic of a Masonite Encapsulated Cell for = 0.2

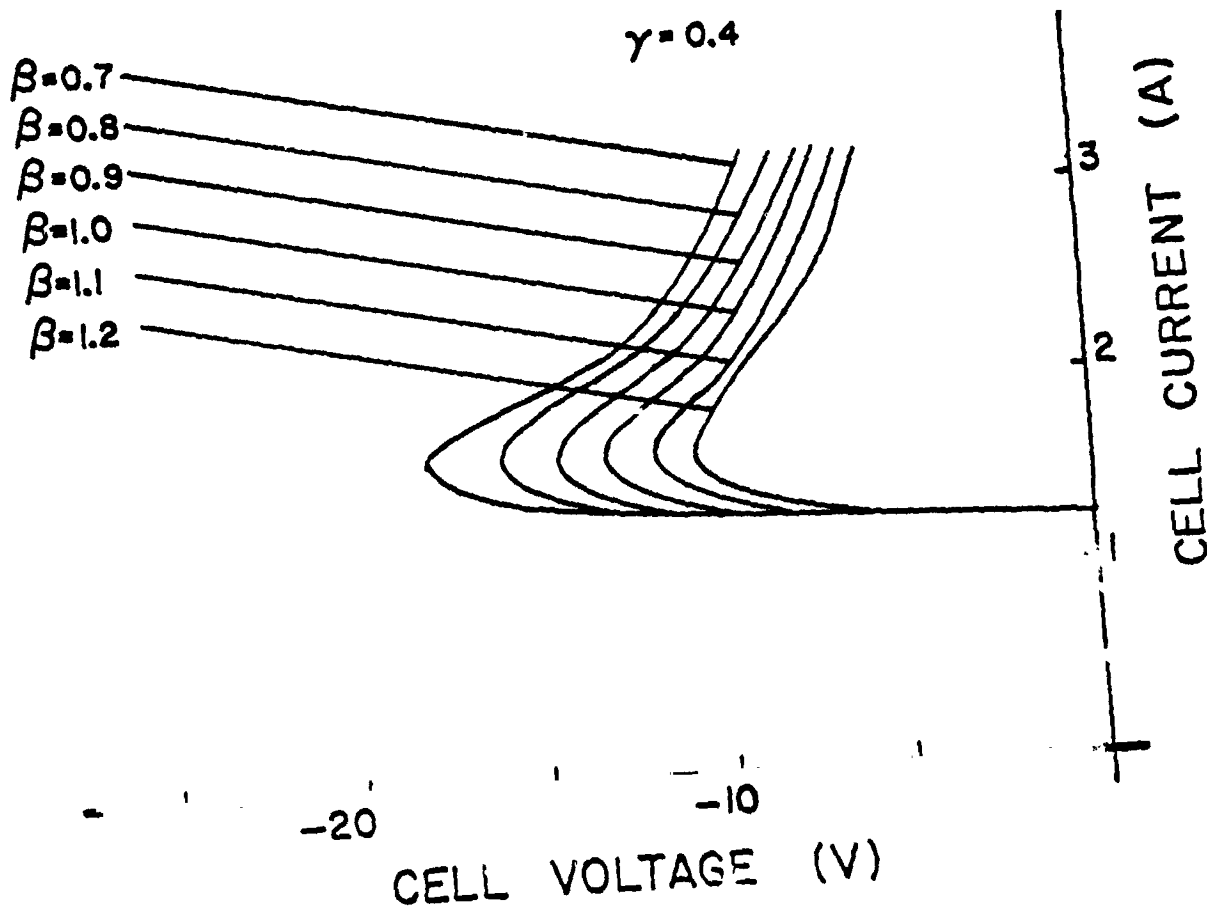


Figure 2.34 Calculated VI Characteristic of a Masonite Encapsulated Cell for γ = 0.4

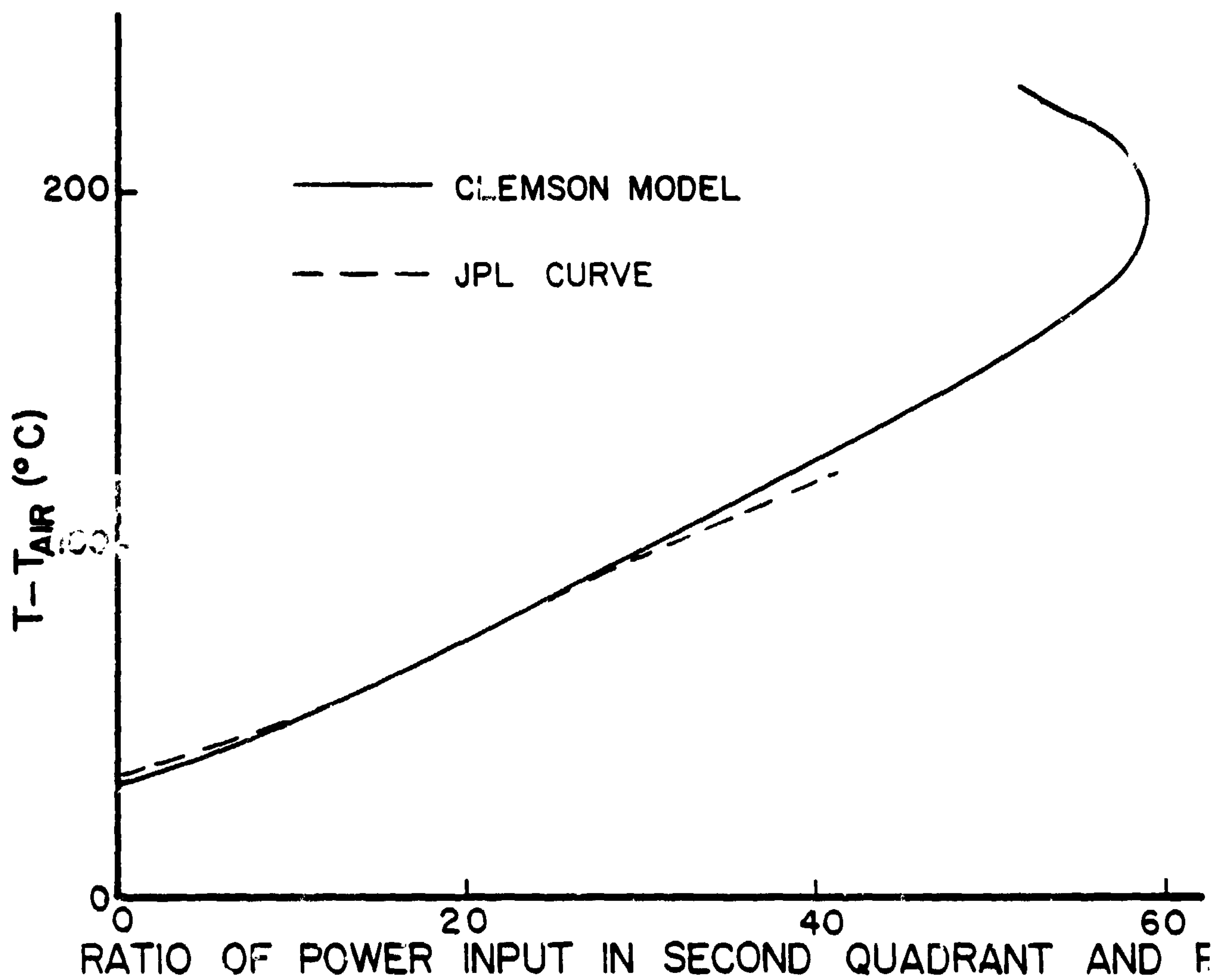


Figure 2.35 Comparison of Clemson Model with JPL Prediction

a bending back of the curve, indicating that although the hot spot temperature is continuing to increase the power into the hot spot is decreasing. This thermal runaway condition results as the silicon at the hot spot location approaches intrinsic at 250-300°C. Doubling back of Figure 2.35 occurs as the knee of the VI characteristic is approached.

3.0 DEVELOPMENT OF IMPROVED SOLAR CELL MESUREMENT METHODS

Subtask 2

C. R. Saylor

J. F. Christ

PRECEDING PAGE BLANK NOT FILMED

3.0 DEVELOPMENT OF IMPROVED SOLAR CELL MEASUREMENT METHODS - Subtask 2 - C. R. Saylor, J. F. Christ

3.1 Introduction

As the number of samples in the accelerated test program increased it became quickly evident that the electrical measurement procedure was the key to obtaining meaningful results. The measurements made before and after stressing had to be highly repeatable in order to detect the small changes which occurred as a result of stress. Furthermore, in order to perform the quantity of tests necessary to obtain statistically significant results, measurement had to be rapid. The potentiometric measurement techniques, which were used during the first two years of the program, and which were described in the First and Second Annual Contract Reports, were satisfactory as long as the volume of cells being tested was small and provided great care was used by project personnel during measurement. In order to obtain high repeatibility with a high throughput and inexperienced operators new measurement instrumentation was required. Since inexpensive instrumentation having these characteristics was not commercially available a subtask was initiated for its design and fabrication and the resulting tester is described in this section. Although originally developed for reliability work, the system is flexible and by virtue of its low cost, high repeatability, and high throughput should be useful in routine industrial measurement applications as well.

The measurement philosophy of the new instrumentation was to electrically characterize the cells rapidly enough that their temperature would not change during measurement. In practice this meant making the measurements in less than 1 second. In order to obtain satisfactory light source stability and spatial uniformity it was decided to use continuous ELH lamp illumination, but with a shutter to avoid cell heating due to illumination. To obtain the

PRECEDING PAGE BLANK NOT FILMED

desired repeatability and speed an all digital microcomputer controlled system was called for. For these reasons the tester may be termed a microcomputer controlled, constant temperature, short interval IV tester. The term "short interval" is meant to differentiate a shuttered system of this type from either a flash type tester or a steady state type tester. This approach to electrical measurement allows the cell temperature to be controlled prior to illumination by air flow rather than the massive water cooled heat sinks required previously in the steady state method. The low thermal mass of an individual cell should respond quickly to a regulated air stream. Finally, an all digital system would eliminate much human interaction and allow the data to be introduced directly to the IBM 370 system, where it could be statistically manipulated. These general philosophical guidelines led to the overall system design of Figure 3.1

When the shutter is opened a program stored in the microcomputer steps the cell through the I-V curve, storing the voltage and current values of each data point in memory. Operation is as follows: The desired current value through the solar cell is computed in the microcomputer in digital form and converted to an analog voltage using the digital controller, which in turn controls the programmable power supply. The programmable power supply electronically loads the cell to the current value given. The voltage across the cell is measured using an analog-to-digital converter whose output the microcomputer reads and stores. The process of electronically loading the cell to the given current value, measuring the voltage and storing the data, is repeated until the entire I-V curve is measured. The I-V curve data points that are stored in memory locations can be displayed on the oscilloscope by connecting two digital-to-analog converters to the X and Y inputs. Hard copy

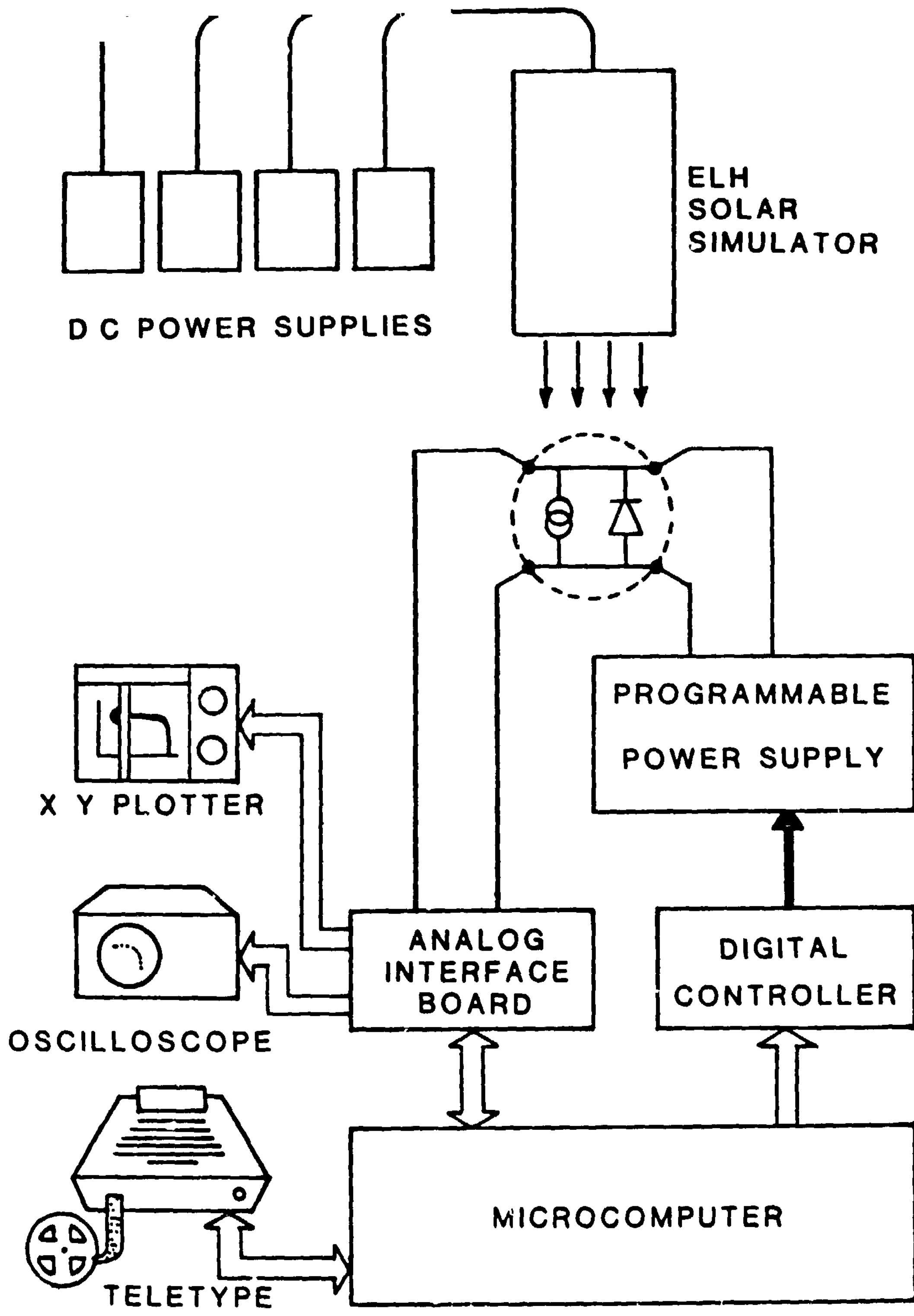


Figure 3.1 Measurement System Block Diagram

of the digitized I-V curve can also be plotted using the X-Y plotter. The individual data points and/or parameric data can be printed using the teletype and also be output on paper tape to be fed into a large digital computer for further analysis. Implementation of the system concept of Figure 3.1 into specific hardware will now be discussed.

3.2 System Hardware

The system hardware can be divided into four major catagories: microcomputer, interface hardware, solar simulator and cell holder. Each of these will be discussed in detail.

3.2.1 Microcomputer — The microcomputer used in the short interval tester is a single board computer, the SBC 80/10A, manufactured by the Intel Corporation (3.1). It is a complete computer system on a single 6.75 x 12 inch printed circuit card. As can be seen in Figure 3.2, the system is designed around three system busses which provide all necessary communication between different parts of the system.

Intel's single chip 8-bit n-channel MOS 8080A CPU is the central processor for the computer. The 8080A contains six general purpose registers and an accumulator. The six general purpose registers may be addressed individually or in pairs providing both single and double precision operators. The 16-line address bus allows addressing of up to 64k bytes of memory. An external stack, located within any portion of memory, may be used as a last-in first-out stack, to provide subroutine nesting capability. A sixteen line address bus and eight line bidirectional data bus are used as the interface to memory and input/output (I/O) devices.

The SBC 80/10A contains 48 programmable parallel I/O lines implemented using two Intel 8255 Programmable Peripheral Interface integrated circuit

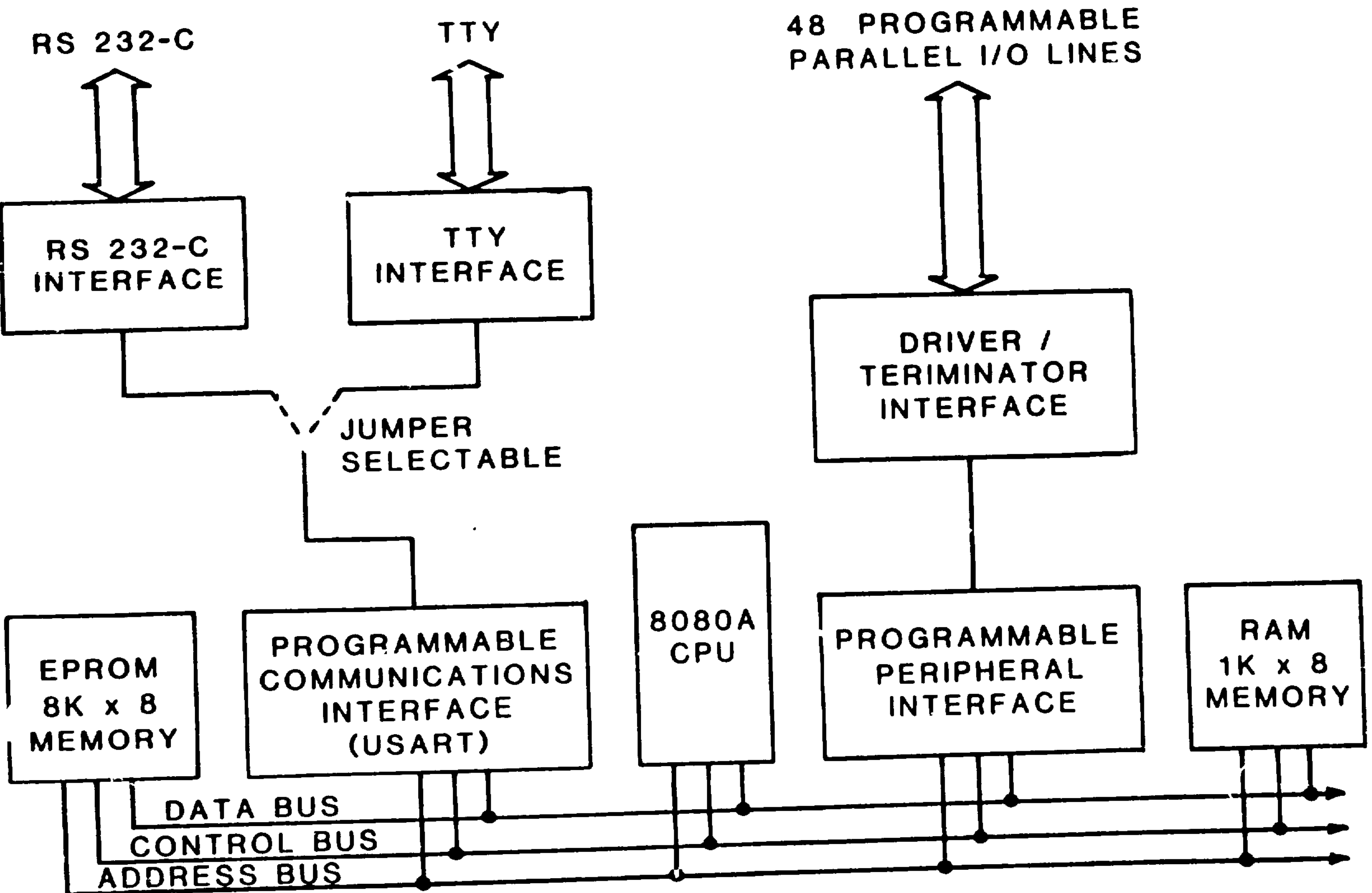


Figure 3.2 Microcomputer System Block Diagram

chips. The 8255's can be programmed in software in combinations of unidirectional/bidirectional, input/output ports.

The SBC 80/10A contains 1k (1024) 8-bit words of Random Access Memory (RAM) and sockets for up to 8k bytes of Read Only Memory (ROM).

A programmable serial communications interface using Intel's 8251 Universal Synchronous/Asynchronus Receiver/Transmitter (USART) is contained on the board. The USART can be programmed in software to interface to a number of communications devices, by programming transmission rates, control character format, data format, etc. A jumper selectable 20 ma current loop for interface to a TTY is also provided.

Both the 8251 USART and the two 8255 I/O interface chips are programmable and must be configured under software control before they are used. The 8251 is initialized by sending a mode word followed by a control word to the USART's control port. In the short interval tester, the 8251 is configured as shown in Figure 3.3 and 3.4. The 8255's are configured by sending a single command word to each of the I/O interface chips' control port and are configured as shown in Figure 3.5 and 3.6.

The measurement system required 17k bytes of RAM, and 9k bytes of TOM. Of the 17k bytes of RAM, 1k bytes were provided on the SBC 80/10A, and 16k bytes were provided by an add-on RAM board. Of the 9k bytes of ROM, 8k bytes were provided on the SBC 80/1A, and 1k bytes provided on the interface board. The memory was allocated as shown in Figure 3.7.

3.2.2 Interface Hardware — Transferring the analog data into the digital computer and the digital information from the computer to the analog outputs requires analog-to-digital converters (ADC) and digital-to-analog converters (DAC). An interface board, complete with an ADC and two DACs was

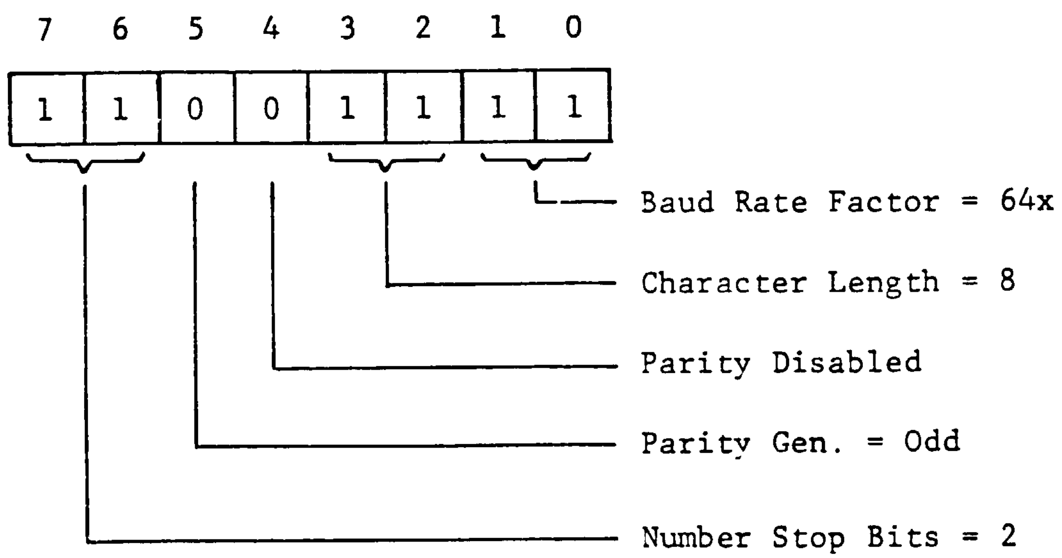


Figure 3.3 8251 Mode Configuration

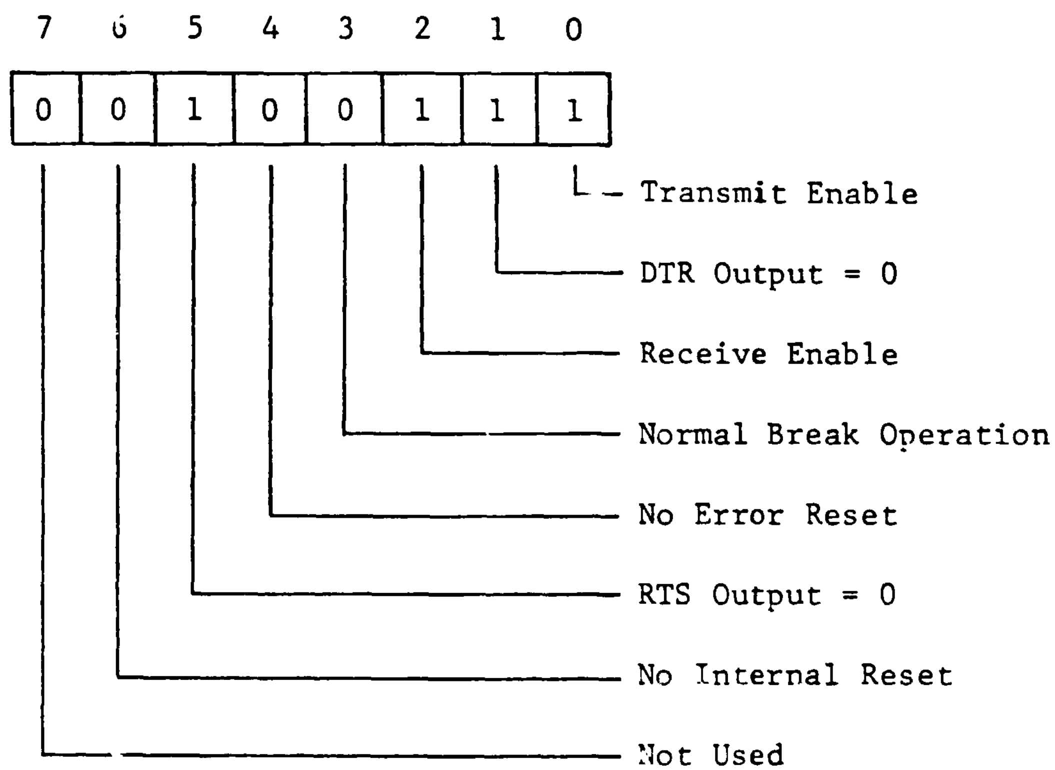


Figure 3.4 8251 Command Configuration

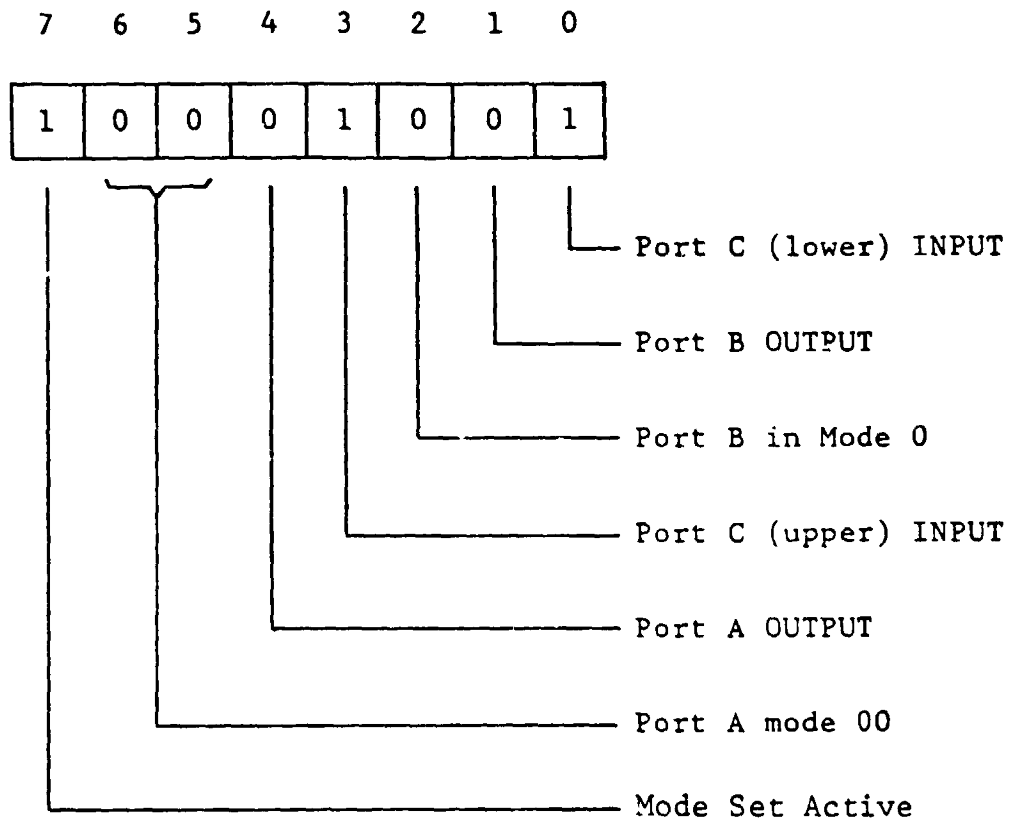


Figure 3.5 Port 1 Command Word

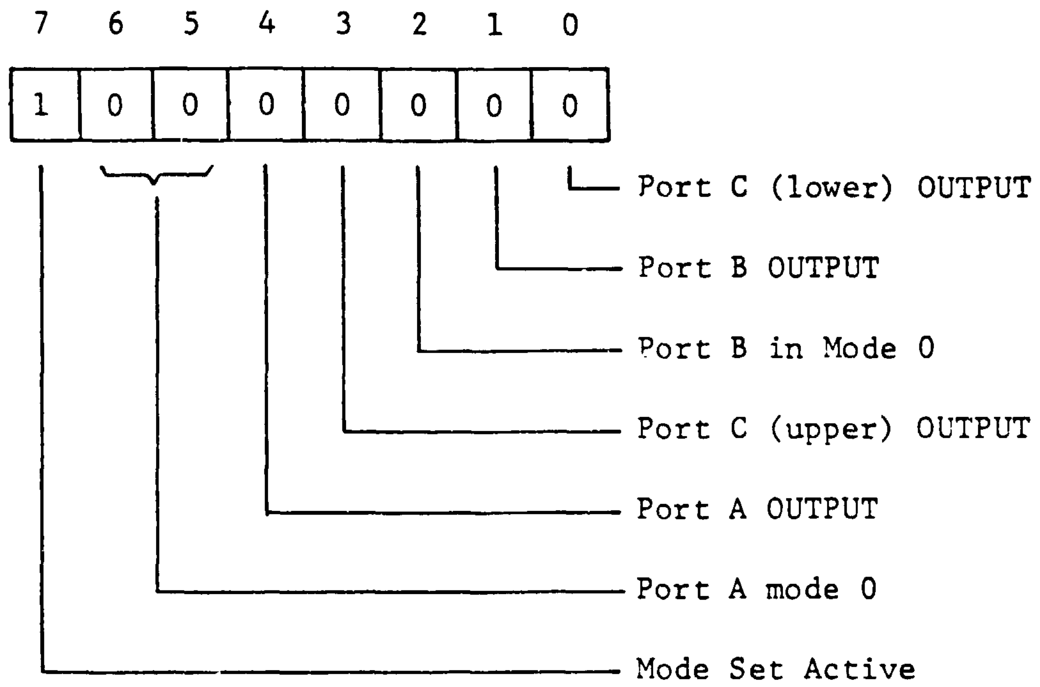


Figure 3.6 Port 2 Command Word

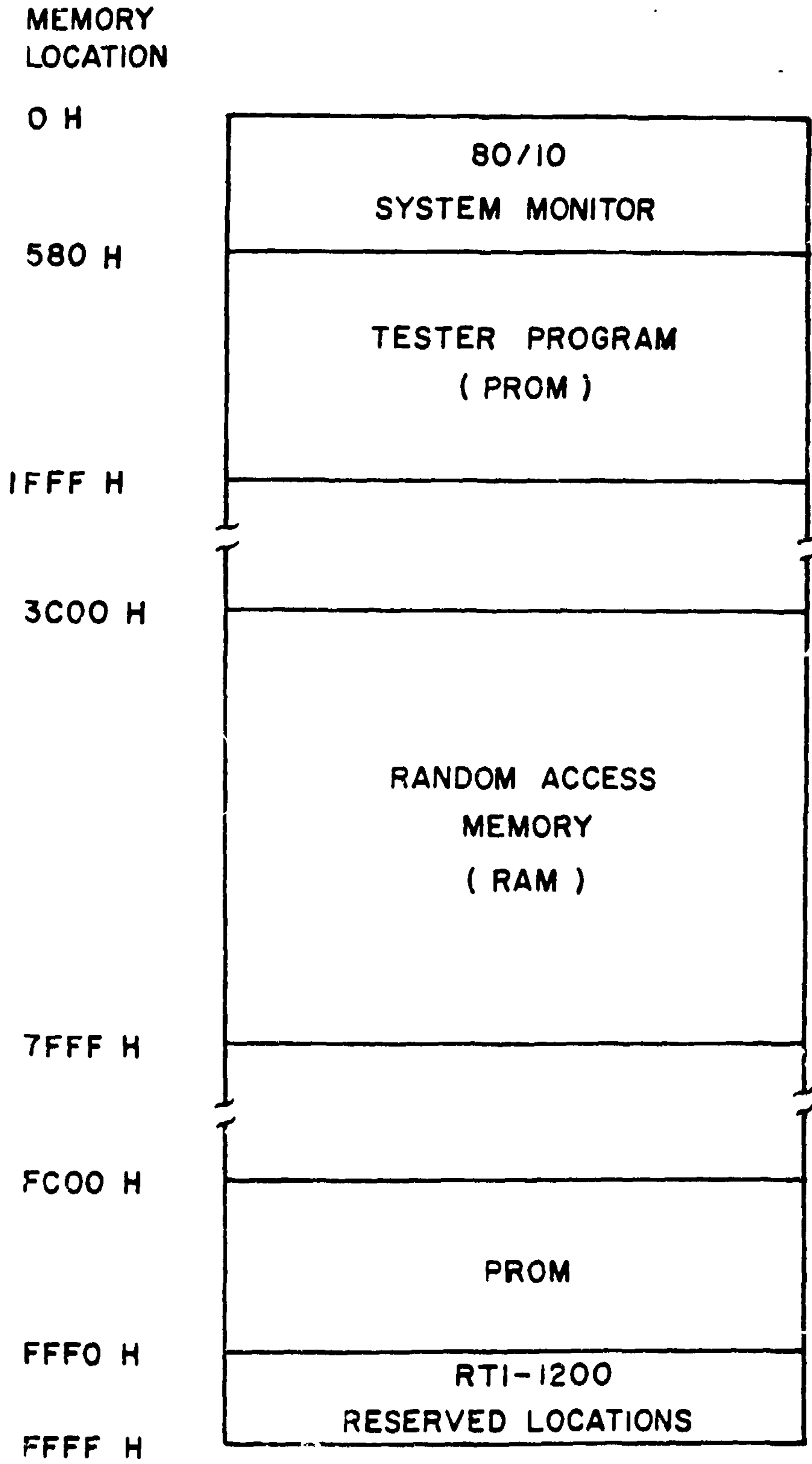


Figure 3.7 System Memory Allocations

installed in the solar cell tester. A programmable power supply and its digital controller were also installed to allow the microcomputer to step through the I-V curve during the measurement period. Each of these devices will be discussed in detail.

The Analog Interface Board The RTI-1200 interface board, manufactured by Analog Devices, Inc., is functionally, electrically, and mechanically compatible with the Intel SBC 80/10A microcomputer (3.2). All connections between the RTI-1200 and SBC 80/10A were made by plugging them into a compatible card cage. The RTI-1200 is connected to the microcomputer in a configuration commonly called memory mapped interface. The RTI-1200 appears to the microcomputer as a block of memory locations. Data and command information is transmitted to the interface board via software instructions that write into memory, and data and status information are received from the interface board via software instructions that read from memory. Ths memory mapped interface technique simplifies the software instructions needed to communicate with the interface board.

The RTI-1200 performs several functions such as data acquisition, outputting analog signals and memory expansion. The RTI-1200's most basic function is data acquisition, as can be seen from Figure 3.8. This is accomplished with an analog multiplexer than can accept up to eight differential analog input signals, a software programmable gain amplifier, a sample and hold amplifier, and a 12-bit analog to digital converter. The A/D converter can be configured for any of three input ranges using wire wrap jumpers. The input range of -5V to +5V was selected because it provided the greatest resolution. The programmable gain amplifier was used to increase the input sensitivity of the analog inputs, as shown in Table 3.1. To select the gain setting of

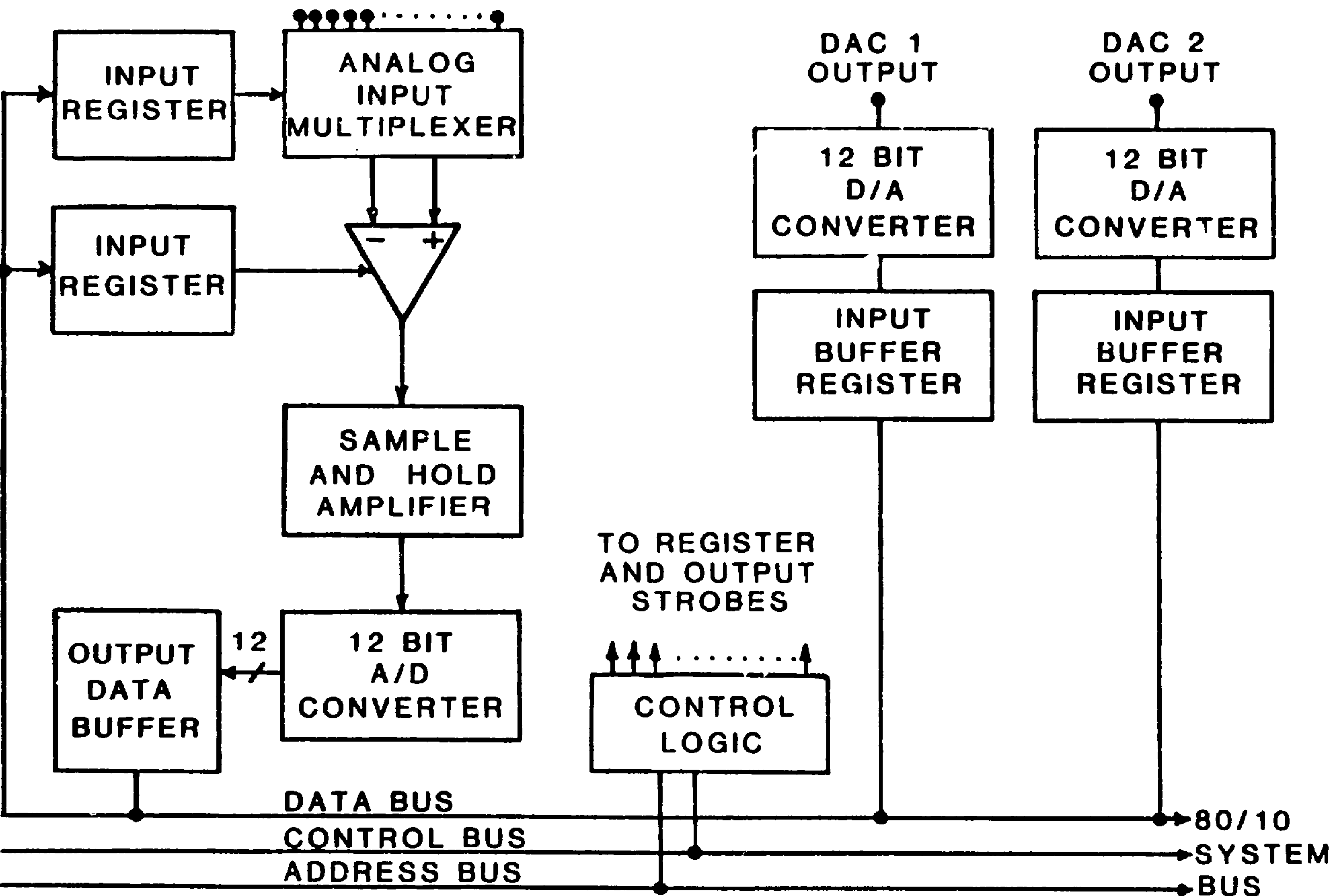


Figure 3.8 Analog Interface Board Block Diagram

the programmable gain amplifier, a command word is written to memory location address FFF9 hex. The low 2-bits of the command word establish the gain according to Table 3.2. The programmable gain amplifier also enables the software program to specify different gains for different channels of input.

Gain = 1	Gain = 2	Gain = 4	Gain = 8
-5V to +5V	-2.5V to +2.5V	-1.25V to 1.25V	-.615V to .625V

Table 3.1. Actual Input Range For Various PGA Gains.

00 - GAIN OF 1
01 - GAIN OF 2
10 - GAIN OF 4
11 - GAIN OF 8

Table 3.2. Command Word to Programmable Gain Amplifier

Selection of the desired input channel is performed by writing a command word to the memory address location FFFA hex. The bottom 3 bits of the command word are used to select one of the eight channels, 0 through 7. Once selected a channel remains selected until a different channel is selected. The selected channel's 12-bit input data is read by the microcomputer from memory address locations FFFD and FFFE hex. The coding was selected to be two's complement code.

Outputting analog signals was implemented using the digital output channels provided on the RTI-1200. each analog output channel consist of a 12-bit digital to analog converter and an operational amplifier used for selecting the output range desired. Digital data is inputted to the D/A

converters by way of memory address locations FFF4 hex through FFF7 hex, as shown in Table 3.3, using two's complement code. Since the SBC 80/10A has an

Memory Address	Format								Byte Name
FFF7	0	0	0	0	SIGN	B10	B 9	B 8	DAC 1 LO
FFF6	B 7	B 6	B 5	B 4	B 3	B 2	B 1	LSB	DAC 1 HI
FFF5	0	0	0	0	SIGN	B10	B 9	B 8	DAC 2 LO
FFF4	B 7	B 6	B 5	B 4	B 3	B 2	B 1	LSB	DAC 2 HI

Table 3.3 D/A Converters Memory Address Locations

8-bit data bus, the data has to be written to the D/A converters in two bytes. All 12-bits are simultaneously strobed into the D/A converter when the four most significant bits are loaded. This is done so that the D/A converter's analog output will move directly from an old value to the desired new value without first going to some intermediate value.

The RTI-1200 has a PROM socket for memory expansion of 1008 bytes of data using the INTEL 2708 Erasable Programmable Read Only Memory (EPROM). The bytes are served for the data, control and status functions.

Programmable Power Supply A programmable power supply was used as an electronic load to obtain the solar cell's characteristic I-V curve. This was implemented by initially setting the current value to zero and incrementing the amount of current through the cell while simulataneously taking voltage readings at each of the incremented values, until the entire I-V curve was traversed.

The ATE 6-5M programmable power supply and the SN 500-121 digital programmer, manufactured by Kepco, Inc., were used in the solar cell tester. The digital programmer's function was to convert a 12-bit unsigned number from the microcomputer into a 0 to 1 volt analog output signal. This 0 to 1 volt analog signal was fed into the input of the programmable power supply to produce a 0 to 5 ampere output current. The computer was then able to control the current through the solar cell in increments as small as one milliampere.

The digital programmer was configured to accept 12-bit parallel data and output a single polarity, 0 to 1 volt signal. The parallel input data was loaded into a temporary storage register on the negative edge of a strobe pulse. The strobe was generated in software, and outputted on port E4H, of the microcomputer. A busy signal was then outputted by the digital programmer, which remained high until the first clock transition following data transfer, indicating it was ready to accept new data. This signal was monitored by input port E5H of the microcomputer.

The programmable power supply was configured to operate as a current source. The output and main feedback capacitors were removed, thereby providing the characteristics of a wide band amplifier, ideal for current stabilization and high speed current programming.

3.2.3 The Solar Simulator -- From a spectral characteristic standpoint the most desirable light source with which to measure solar cells is natural sunlight, but because of its variability, solar simulators must be used.

Three kinds of steady state light sources are commonly used: Xenon arc lamps, quartz-halogen tungsten lamps, and ELH lamps (3.3). The Xenon arc lamps match the spectrum of natural sunlight closely, but are the most expensive of the three and tend to be somewhat unstable. The ELH lamps have

an only slightly different spectral response from the Xenon arc lamps, but have the advantage of simplicity and lower cost. The quartz-halogen tungsten lamps are a very poor choice for they supply too much infrared light. While the ELH lamp is a quartz-halogen tungsten lamp, it has dichroic-coated reflectors which transmit a significant portion of the infrared light while reflecting the visible light. The ELH lamp is commonly used in 35mm slide projectors. There also exists a pulsed type Xenon lamp that has been used with automatic test systems to measure array panels. However, pulsed simulators are expensive and not particularly stable. A steady state ELH lamp simulator, with four ELH lamps, was found quite satisfactory for reliability testing. Each of the 300 watt ELH lamps was powered by a 100 vdc power supply and cooled with forced air from a fan mounted in the top of the simulator as seen in Figure 3.9. A light shutter was used which permitted light to shine on the cell only during the one second measuring interval, minimizing the amount of heat input to the cell under test so that it remained essentially at ambient temperature during measurement.

The short circuit current of a solar cell is controlled by the light intensity and its spectral distribution. While the ELH solar simulator was capable of reproducible results, its spectral distribution did not perfectly match that of the sun. Therefore from the standpoint of absolute accuracy, where ever possible a reference cell of the same type as the cell being measured and which had been calibrated in natural sunlight was used to set the intensity. From the standpoint of repeatability it was also important that the simulator's intensity did not drift or fluctuate with time. Individual dc regulated power supplies were used for each of the four ELH lamps with a warm up period of one hour to minimize drift. A photodiode mounted in the cell

holder fixture and maintained at constant temperature was used to monitor the intensity between each measurement. The photodiode indicated fluctuation or drift in intensity or the presence of a burned out lamp that otherwise would have gone unnoticed.

3.2.4 Cell Holder — In order to position cells under the solar simulator in exactly the same place each time and to make stress free voltage and current connections, a cell holder was employed (see Figure 3.10). A stationary metal ring and an identical shape plastic ring were placed on top of each other and clamped together, sandwiching the solar cell leads between the rings. This avoided stressing the solder tab when the measuring leads were attached. Cells of different size and shape require a different pair of rings. A Kelvin four point measuring circuit was used to minimize the effect of any external contact resistance. The current contacts were made using the vacuum port supports to the back and the metal ring to the top while the voltage contacts were made using the thermocouple jacket to the back and a clip attached to the solar cell lead. Solar cell characteristics made in this fashion will thus show the effect of any contact resistance between the leads and the surface metalization and between the surface metalization and the silicon, but will be completely insensitive to any contact resistance between current carrying elements and the cell leads. Measurements will therefor reflect the intrinsic behavior of the cell with its leads, but not variations due to connecting the cell to the external circuitry. A foot pedal actuated solenoid valve activated the three vacuum ports that held the cell down to the thermocouple.

The cell holder was designed to allow air to be blown over the top and bottom surfaces of the cell for temperature control as shown in the photograph

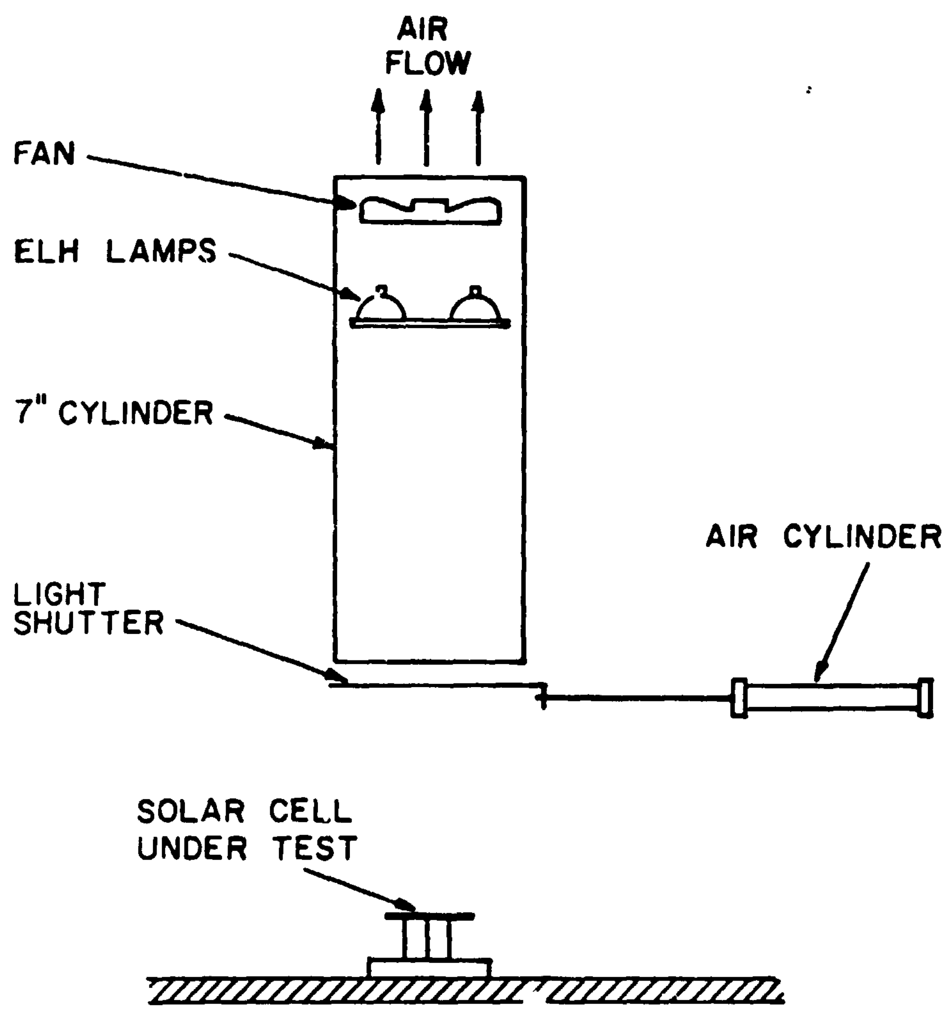


Figure 3.9 Diagram of Solar Simulator

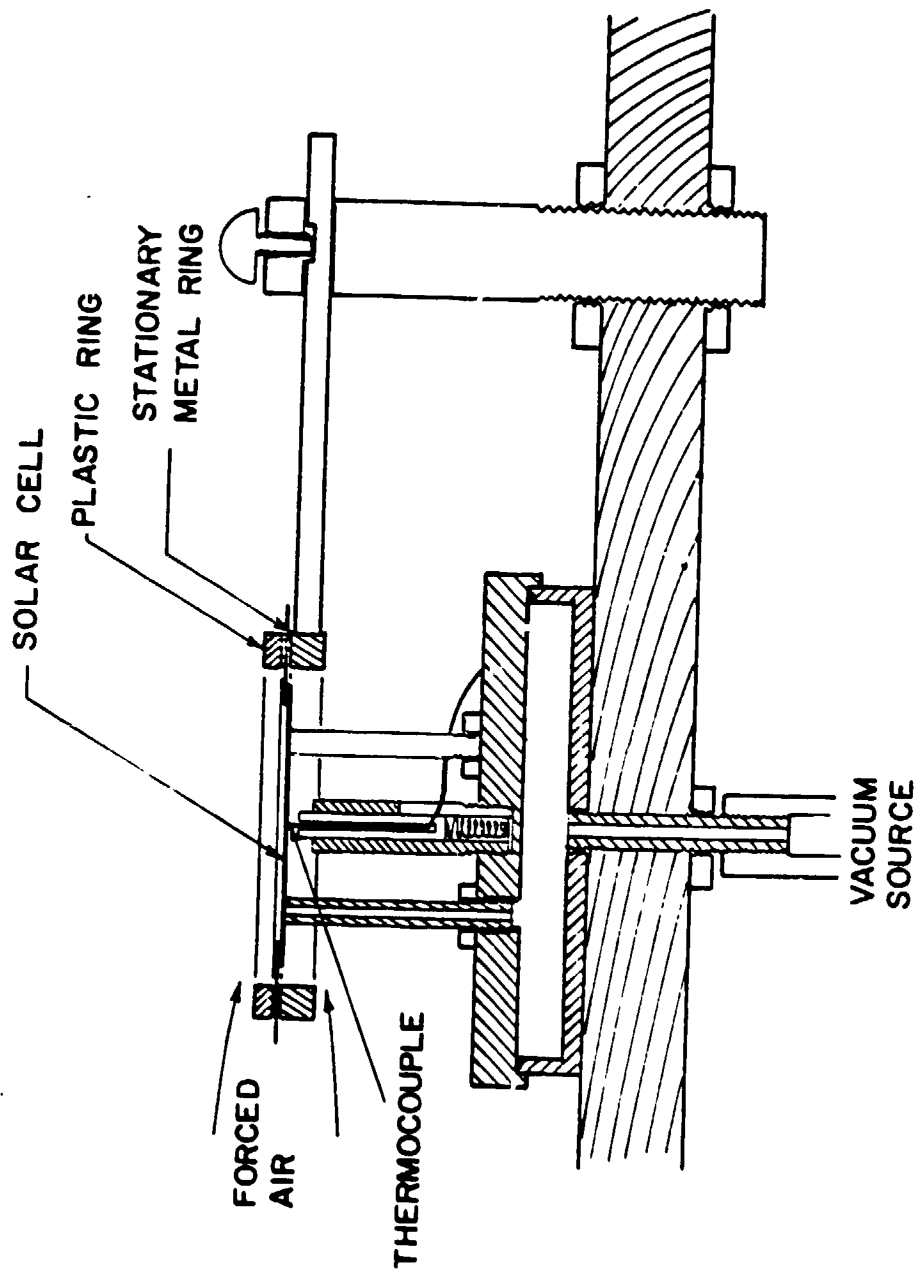


Figure 3.10 Diagram of Cell Holder

of Figure 3.11. An air mixer box was fabricated which outputs 28°C forced air. T .s approach to temperature control has proven to be a big improvement over the previously used heat sink method. Cells with irregular backs and encapsulated cells can be easily maintained at a constant temperature. Figure 3.12 is a photograph of the complete hardware system.

3.3 System Software

3.3.1 Introduction

— Software instructions for the microcomputer were written in a verson of the PL/M programming language, called PL/M-80. This is a high level language written for implementing PL/M on the Intel 8080 microcomputer. A high level language was chosen over the lower level assembly language because of the ease and speed in programming it offers. PL/M-80 programming instructions are oriented more toward the English language than toward nemonics, resulting in better documentation and a reduction of errors. While the PL/M-80 source program can not directly be executed on the microcomputer, it can be compiled to a more primitive form called assembly language, which can be assembled to create an object program that the microcomputer can execute. A program called a compiler exists in the Intel Microcomputer Development System at Clemson Unversity and which transforms a PL/M-80 source program to an object program. The development system also provides a program called the Intel Systems Implementation Supervisor, ISIS-II which gives the user the capabilities to do text editing, linking, and program locating, thus reducing the software development and debugging time.

With PL/M-80, programs should be writtin in modules. Each module can then be checked independently for errors before being linked together. This localizes errors for easier debugging. The software for the tester has one Main Program Module and a number of other modules containing only procedures.

Figure 3.11 Photograph of Cell Holder Fixture

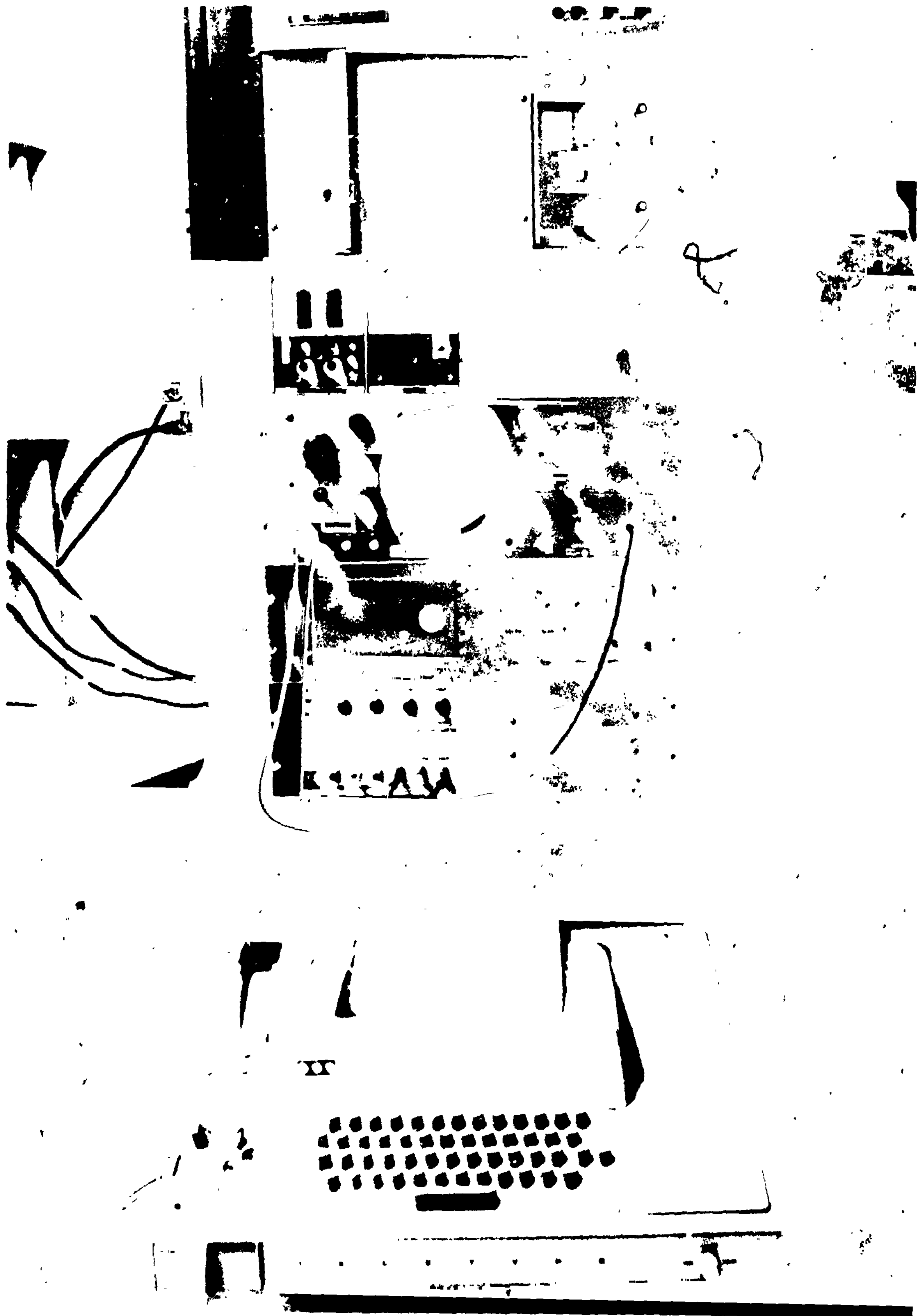


Figure 3.12 Photograph of Measurement Instrumentations

ORIGINAL PAGE IS
OF POOR QUALITY

Each of these procedures and the Main Program Module will be described in detail. Detailed flow charts and the complete PL/M-80 source program are contained in Appendix B.

3.3.2 Main Program Module — The Main Program performs a sequence of call instructions. Initially the cell identification procedure, named the ID procedure, is called to instruct the operator to enter the cell type, lot numer, and the stress level. The Main Program remains in a loop with each pass through the loop being a measurement, display, and output of the parameter data for one solar cell.

3.3.3 ID Procedure — This procedure is called from the Main Program whenever a change in cell lot is needed. The operator is instructed to type the cell type, lot number, and stress level. This information is stored to be later punched on paper tape along with the parametric data of the cell. Before the procedure returns to the Main Program it types out the table headings for the parameters.

3.3.4 Get Cell Number Procedure — This procedure is called by the Main Program. Initially the procedure prompts the operator to type in the cell number, by printing 'CELL NUMBER = '. The procedure will accept up to three characters and store them in consecutive memory locations. They will be printed out when the cells' parameters are printed. In order to correct any character that was pressed in error, only the last three numbers are saved. After entering the cell number a carriage return will return the procedure back to the main program.

3.3.5 Step Through Procedure — This procedure is called from the Main Program approximately one half second after the light shutter is activated to open. This delay, which can be varied, allows the light shutter to be fully

open before any data points are measured. It then steps through the characteristic I-V curve by repetitively calling the Compute Current Procedure which outputs the next value for current, and upon returning measures the corresponding voltage across the cell. This process is repeated until the cell has passed completely through the I-V curve. Since the programmable power supply has a finite settling time a delay of about ten milliseconds occurs between a change in output current and the time the voltage is read. The voltage is then read repetitively until two consecutive voltage readings are the same. Since the same voltage potential could be measured on the rise and fall of an oscillation the initial 10 ms time delay was included as an extra precaution. Both the delay and voltage comparison readings are used to ensure the current has settled down. If the voltage readings have not settled down after sixteen times the program stores the last value and continues to the next data point. This situation may occur in the flat region of the I-V curve, close to the short circuit current, where a small amount of ripple on the programmable power supply's output current can cause a larger ripple on the voltage. The inability to have the voltage settle down in this region of the I-V curve does not significantly affect measurement of the short circuit current parameter. The procedure returns to the main program upon a negative voltage reading, which indicates the solar cell has entered the second quadrant of the I-V curve.

3.3.6 Compute Current Procedure — This procedure is called from the Step Through Procedure. It determines the value for current of the next data point along the I-V curve, ouputs it to the programmable power supply and returns to the Step Through Procedure where the corresponding voltage value is read.

There are three regions of the I-V curve which require special concern to ensure that a sufficient number of data points are obtained in order to accurately determine cell parameters. Data points measured in the vicinity of the open circuit voltage are used to calculte the slope dV/dI at I = 0 which is proportional to the series resistance, R_s, and to determine the open circuit voltage, V_{oc}. Data points measured in the vicinity of the maximum power point need to be closely spaced in order to calculate the maximum power point, P_m, accurately. Data points in the vicinity of the short circuit current, I_{sc}, are used to determine both I_{sc} and the slope dV/dI at V = 0 which is proportional to the shunt resistance, R_{sh}. Therefore the Compute Current Procedure has separate modes of operation for each of the three regions in order to ensure that sufficient data points are measured to determine the cell parameters. Figure 3.13 illustrates the regions in which each mode is applicable.

The Compute Current Procedure is initially in mode 0, with an output current of zero amperes. In mode 0 the current is incremented by 32 milliamperes each time the procedure is called, resulting in another data point. Each time a data point is measured, the power output is calculated by computing the product of current and voltage. This value for power is compared with the value from the previous data point to determine if the maximum power point of the cell has been reached. If the power from the previous data point is greater, then the maximum power point has been passed. The procedure then goes into mode 1 where the current is decremented by 1 milliamp for each data point. The power of each data point is again calculated and compared to the previous data point. Mode 1 continues until the calculated power decreases indicating that the maximum power point has

again been passed. The previous data point is then identified and saved as the maximum power point and the compute current procedure changes to mode 2. Mode 2 initially outputs the current value that mode 0 ended with followed by a value incremented by 4 ma. The change in voltage between these two data points is used to determine the amount of increment to be added to the current for the next point. The change in voltage is again calculated and the next increment determined and so on for the remainder of the curve. This procedure is necessary rather than using a fixed increment because of the large range of possible slopes in the region around the short circuit current. Not only does the slope vary from cell type to cell type, but it also varies within cell types as a result of degradation due to stress. When measuring a high efficiency unstressed cell with a flat slope, dI/dV near zero, an increment as small as one milliampere is needed to obtain a sufficient number of data points in the vicinity of I_{SC}, where as a badly degraded cell requires larger increments for the same number of data points. Use of the 1 ma increment on the degraded cell would result in an excessively large number of data points. Mode 2 uses the voltage change between the two data points to determine the size of the increment. If the change in voltage for the 4 ma increment is between 1.2 mv and 19.5 mv then the increment remains at 4 ma. If the change in voltage is larger than 19.5 mv then the increment is reduced by 1 ma while if the change in voltage is below 1.2 mv the increment is increased by 1 ma. (1ma is the minimum increment possible). The Compute Current Procedure remains in mode 2 until the I-V curve has been completely measured (voltage becomes negative) and is reset to mode 0 for the next cell to be measured.

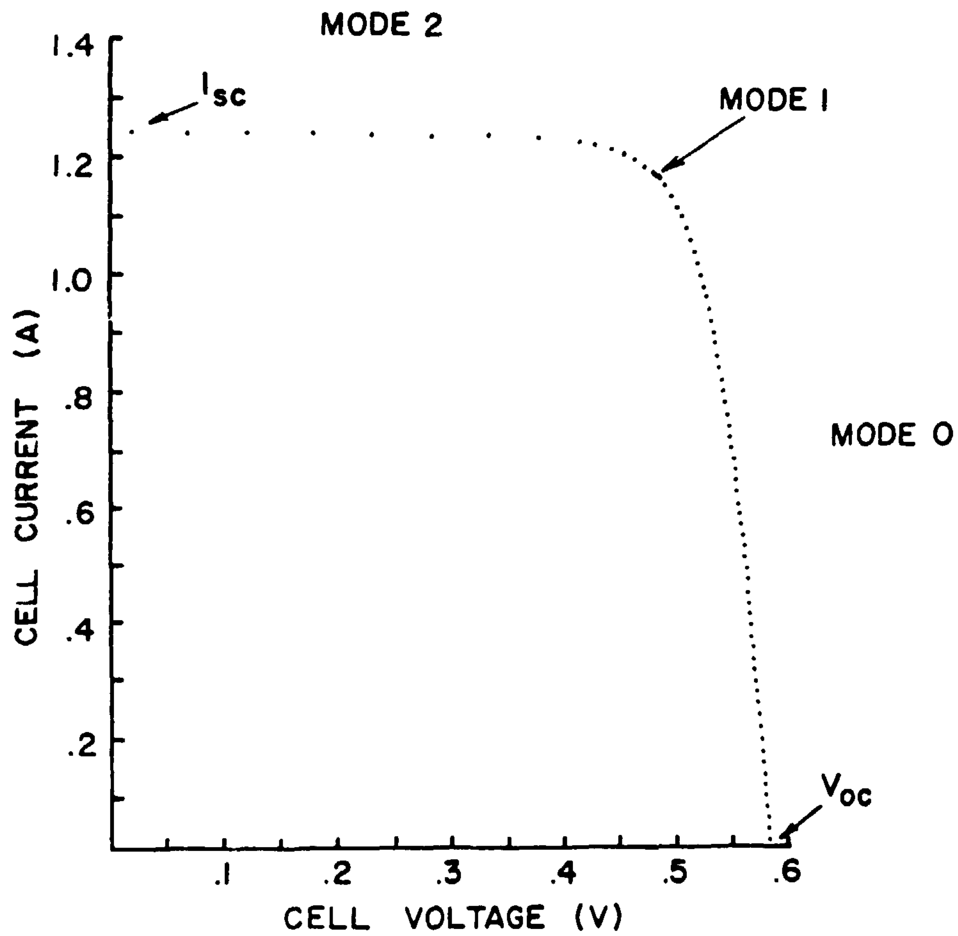


Figure 3.13 Measured Cell Characteristic Illustrating Different Compute Current Mode Regions

3.3.7 Scope Procedure -- The Scope Procedure is called from the Main Program after all I-V data points have been measured and stored in a table in memory. The scope procedure executes in a loop that outputs the voltage and current values of each data point to D/A converter one and D/A converter two respectively. The outputs of the D/A converters are connected to the X and Y inputs of an oscilloscope to display the I-V curve. Return to the Main Program occurs after any key is pressed on the TTY keyboard.

3.3.8 Print Procedure -- This procedure outputs a string of characters to the TTY. The characters need to be in consecutive memory address locations, and in their ASCII representation. The memory location of the string of characters and the number of characters to be printed are the parameters that need to be passed to the print procedure when it is called.

3.3.9 Fetch Procedure -- The Fetch Procedure inputs a string of characters from the TTY and stores them in consecutive memory address locations. The starting memory address location and the number of characters to be input need to be passed to the procedure when it is called.

3.3.10 Numout Procedure -- The numout procedure converts a number from binary form to an ASCII string suitable for printing. This procedure requires the following parameters when called.

VALUE_____The number whose printed representation is desired.

BASE______An integer from 2 to 16 specifing the base of 'value'.

LC________Leading character, either a ' ', '0', or '0'.

BUFADR____Buffer address.

WIDTH_____The number of characters desired in the printed representation.

3.3.11 Tbout Procedure -- This procedure outputs 12-bits of data to the programmable power supply's digital controller. The lower 8-bits are outputted on port 2A, and the upper 4-bits are outputted on port 2B. A strobe pulse outputted on port 1A latches the 12-bits of data into the controller. A busy line received from the controller is monitored on input port 2C until it goes low, indicating the controller is ready to accept more data. The procedure returns to the program from which it was called when the busy line goes low.

3.3.12 X1232 Procdure -- This procedure takes a 12-bt signed integer and converts it to a 32-bit signed integer in order to convert the voltage and current values read from the A/D converters to a format acceptable for the floating point arithmetic library routines. It can be called from any procedure that declares it external, provided the memory location address of the 12-bit number to be converted and the low memory location address where the 32-bit number is to be stored are given.

3.3.13 Pmax Procedure -- This procedure is called from the Step Through Procedure. It calculates the product of the voltage and current values of the last data point measured. This value for power is compared to the previous data point's power output. A flag, LT, is set to one if the last data point's power becomes less than the previous point's, otherwise the flag remains equal to zero. Before the procedure returns to the Step Through Procedure, the largest power point value is stored in memory.

3.3.14 Global Variables Procedure -- This procedure was used to declare public variables that were used in more than one procedure. The variables were declared external in each of the other procedures. This was used to simplify the bookkeeping so as to reduce the possibility of declaring a variable public in more than one procedure.

3.4 Operating Procedure

3.4.1 Power Up -- In order to prepare the system for measurement the following power up procedure is used:

1. Turn the main power switch to the ON position.

2. One at a time, set each of the four ELH lamp switches to the ON position. There is a built in delay of approximately 3 seconds and the operator should note that each lamp is operating before turning the next on. The lamps should be allowed to operate for 1 hour prior to making measurements.

3. Turn the teletype power switch to the LINE position.

3.4.2 Start Up -- In order to begin measurement taking the following start up procedure is used:

1. Press the RESET button located on the front panel. This resets the microcomputer and initiates execution of the system monitor, a program used for debugging and program start up. A message, "BLC 80P MONITOR REV.C" is printed out indicating the system is ready for a monitor command.

2. To start execution of the main tester program the user types: "G580)" ( ) indicates a carriage return). A "?" is printed out indicating that the main tester program is waiting for a command.

3. One of two valid commands may now be entered: a "Ø)" to start the routine IV characteristic measurement program or a "1)" to start the time varying effect measurement program. If an invalid command is given a "?" will be typed out and the correct command should be entered.

3.4. Routine IV Measurement Program -- Having entered "Ø)" the user is prompted to type in the cell type, cell lot number, and stress level as follows:

CELL TYPE = B

LOT # = 19

STRESS LEVEL = 3

where underline indicates information entered by user. Table headings are then typed out:

TYPE*LOT*S * VOC * I_{sc} * P_m * V_m * I_m * TEMP*

The user is then prompted to type in the cell number and temperature as follows:

CELL NUM = 35

CELL TEMP = 28.0

Ultimately it is planned to receive the temperature information directly from the thermocouple readout, but at the moment it must be manually entered. After pressing the carriage return upon entering the temperature data, the pneumatically controlled light shutter opens, the IV characteristic is measured, and the shutter closes. The sequence takes less than 1 second. The digitized IV curve is automatically displayed on the monitor oscilloscope. If the curve appears satisfactory, pressing any TTY key will cause the measured parametric data to be typed out in the previously prepared table and simultaneously punch the informaton on paper tape. If the IV curve is not satisfactory, holding down the OFF button on the paper tape while pressing any TTY key will erase the data without recording.

After the parametric data has been printed out and the paper tape punched the program will have executed one pass through a loop and returned to prompt the user to type in the next cell number. The program will remain in this loop until all the cells in a lot have been measured. The operator can then press the RESET button and begin execution of the main tester program again.

3.4.4 Time Varying Effect Measurement Program -- Having entered "1 ⟩"
the user is prompted to type in the number of measurements cell type, cell lot number, and stress level as follows:

OF MEAS = 20 ⟩

CELL TYPE = B ⟩

LOT # = 12 ⟩

STRESS LEVEL = 3 ⟩

After the last carriage, table headings are typed out as before

TYPE * LOT * S * VOC * Isc * Pm * Vm * Im * TEMP*

The user is then prompted to type in the cell number and cell temperature as follows:

CELL NUM = 25 ⟩

CELL TEMP = 28.0 ⟩

After pressing the final carriage return the pneumatic light shutter opens, the IV characteristic is measured and the digitized data displayed on the monitor. At this point the shutter does not close, but remains open. Another measurement will then be made, stored in memory, and displayed on the monitor every time a TTY key is pressed. Usually it is desired to make measurements at regular intervals. After the final measurement the parametric data is printed out in the previously prepared table and curves plotted. To measure another cell the RESET button is pushed and the process started again.

The time varying effect measurement program makes it possible to measure changes in parameters that occur as a function of illumination time. Specific examples will be discussed in the next section. It should be noted, however, that with the shutter open the temperature will rise and asymptotically approach a higher value. In this mode of operation it is necessary to record

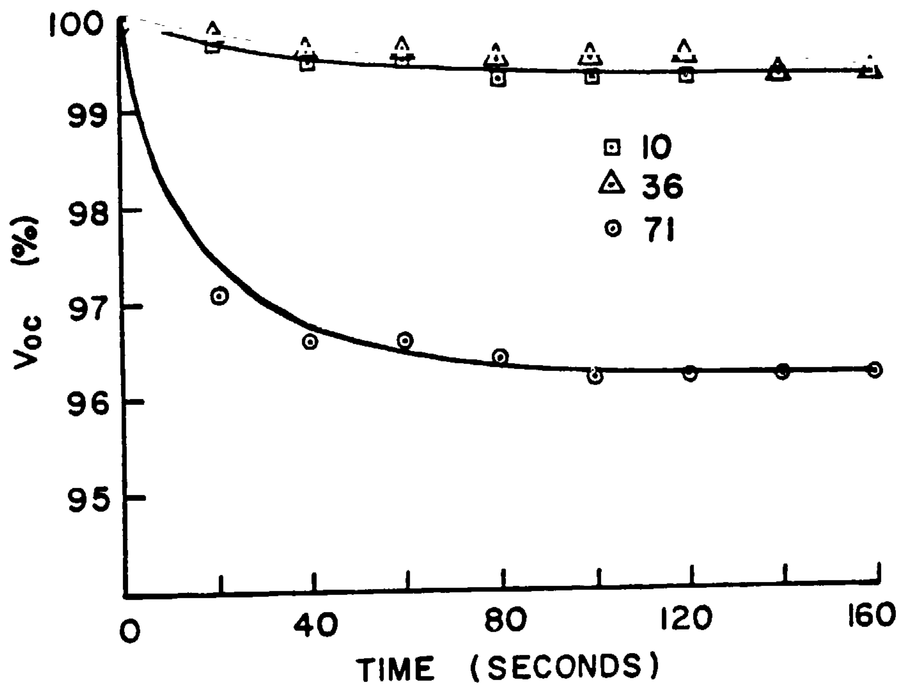
the temperature of each run and to use this value to correct the data. Presently this must be done manually, but ultimately when temperature data is directly accessed this will be automatic. Thus at present the temperature that is entered in the prompting mode and that is displayed in the data table is the starting (dark) temperature.

3.5 Time Varying Effects

Use of the short interval tester uncovered a parametric variation with time of illumination, which had gone unnoticed using conventinal test methods. Figures 3.14 and 3.15 show typical variations of V_{oc}, I_{sc}, and P_m with illumination time. This type of change has been reported previously as photon degredation (2.4). This time varying effect was observed on every cell of one n/p type cell and on a few cells of another n/p type. All cells recovered to their original performance after being in the dark for a few minutes.

Figure 3.16 shows plots from the short interval tester of one susceptible cell's output initially, after an illumination of 20 seconds, and after 10 minutes. The tester can be a useful tool for further study of this phenomenon. Digitized I-V curves can be measured as often as every second and stored in memory to be plotted and/or printed out later. Reduced parametric data can also be printed out.

While the time varying phenomenon is interesting and deserves further study, the variability of the cells pose some problems when trying to measure these cells accurately. Because of the non-repeatability when making measurements in close succession, the standard cell used for adjusting the light to one sun illumination had to be chosen from a non-time varying cell type.



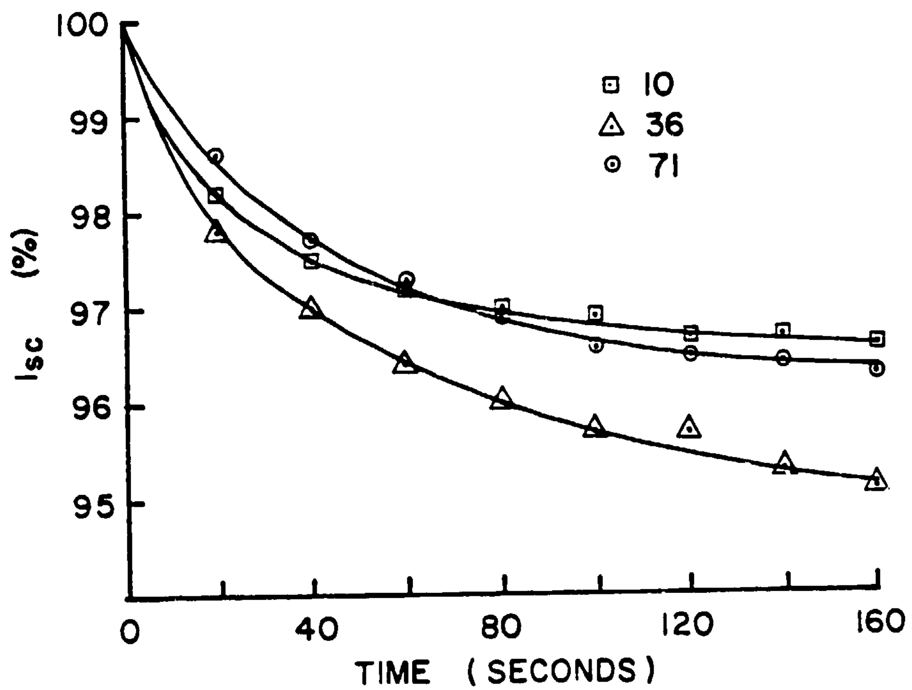


Figure 3.14 V and I as a Function of Illuminations Time for Representative Cells

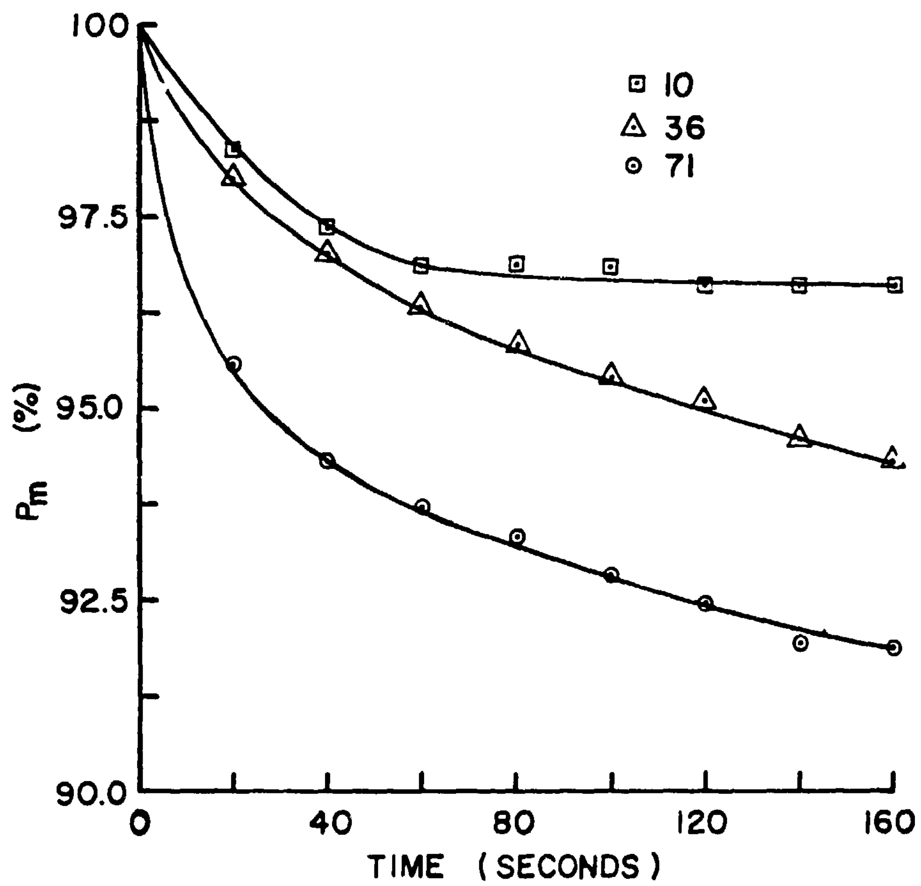


Figure 3.15 P_m as a Function of Illumination Time for Representative Cells

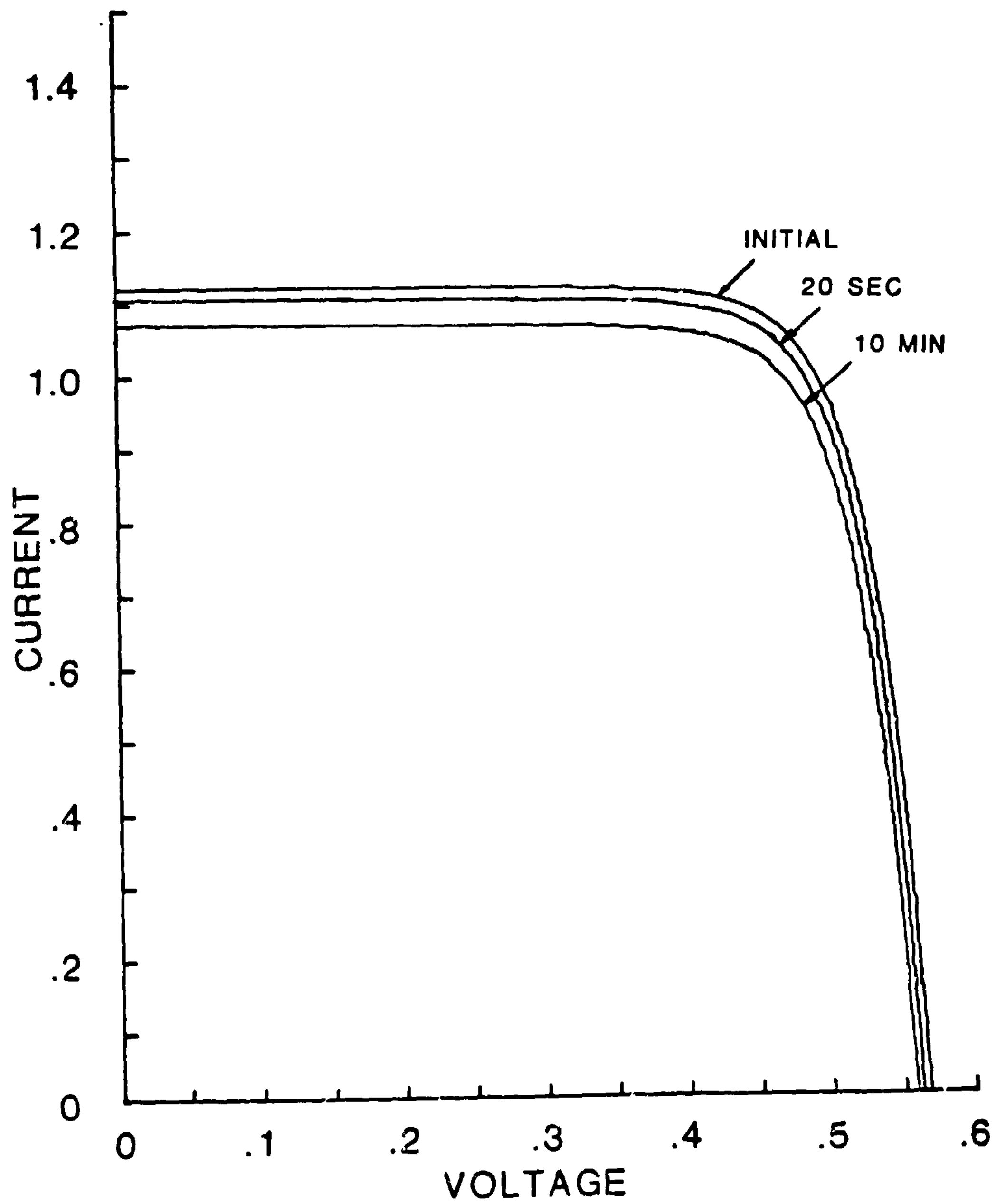


Figure 3.16 Voltage-Current Plots for Different Illumination Times

The question also arose of whether to pre-illuminate the cell and wait until the degradation occurred before measuring, or to measure immediately after illumination. It was found that differences as much as 5% in I_{sc}, 4% in V_{oc}, and 9% in P_m resulted between the two methods of measurement. Measuring immediately upon illumination was chosen because repeatability was of greater concern than absolute accuracy although the actual power tended to be overestimated in this approach. Also, a geat deal of time and effort would be required to pre-illuminate the cells before measurement.

3.6 Data Management

In order to statistically analyze the electrical data obtained on cells subjected to accelerated stressing it is necessary to enter the parameter values, cell number and type, and the stress conditions into an IBM 370 computer. The data management routine which accomplished this prior to the short interval tester, is shown in Figure 3.17. The output of the original test procedure was hard copy of the IV characteristic curve and values copied from meter readings of V_{oc}, I_{sc}, and temperature. It was necessary to first obtain the maximum power point from the characteristic using an overlay of constant power hyperbolas. The coordinates of the maximum power point together with all the other measurement parameters were then transcribed onto keypunch forms. Cards were then prepared from the information on the form and the data placed on disk for permanent storage and later manipulation. Errors were introduced in graphically obtaining the maximum power point, in transcribing the data, and in keypunching the data. Construction of the short interval tester provided an opportunity to eliminate these errors.

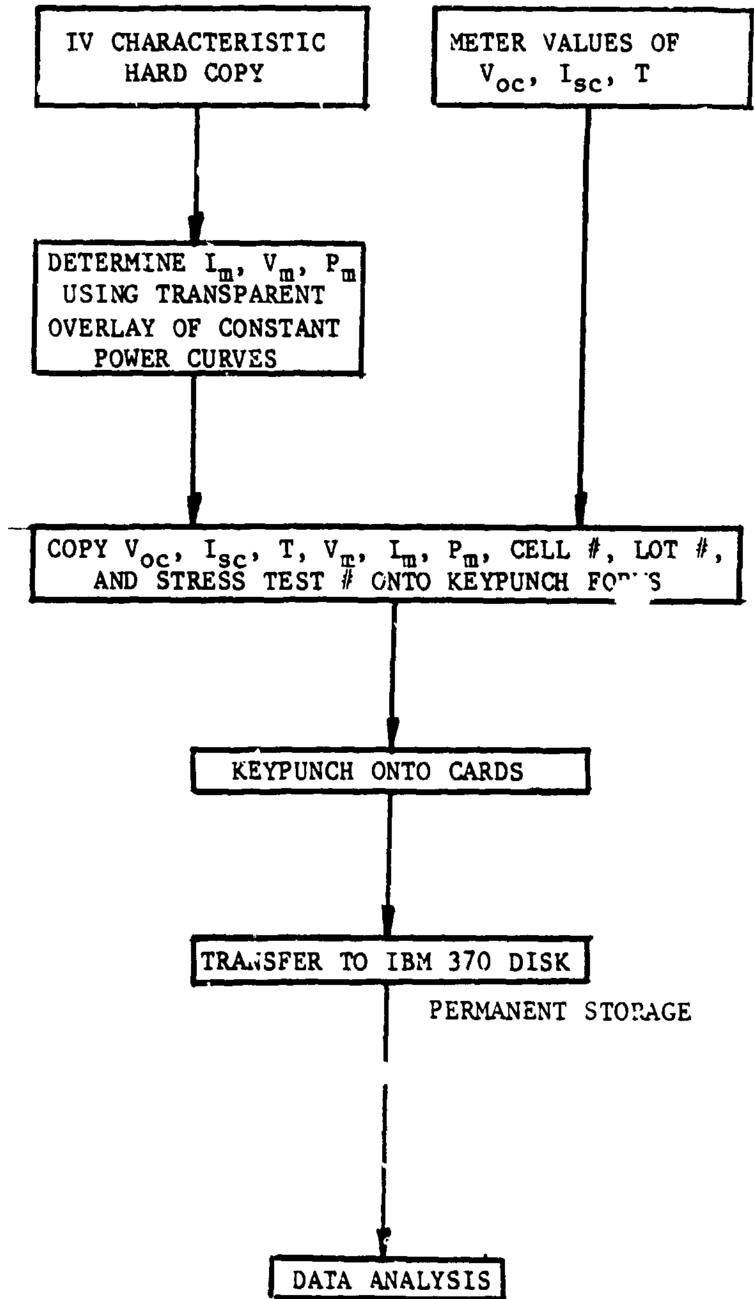


Figure 3.17. Initial Method of Transferring Measurement Data to the IBM-370 Computer

The present data management procedure used with the short interval tester is shown in Figure 3.18. The tester stores the voltage-current data pairs taken on each cell in random access memory. It also stores specific values of V_{oc}, I_{sc}, I_m, V_m, and P_m as well as the manually entered test conditions. The specific parametric values are then recorded on paper tape, but because of the bulk of this storage medium it is not practical to save actual VI data point coordinates and these are discarded. The paper tape is then processed on a PDP-15 where the parametric values are transcribed onto magnetic tape in an IBM compatible format. Use of the PDP-15 is necessary since the IBM equipment does not have paper tape reading capability. The PDP-15 generated magnetic tape can then be manually transported to the central computer facility where it is read onto an IBM-370 magnetic tape for permanent storage. Whenever data manipulation is required the information is placed on disk. Permanent storage of data on tape instead of disk results in appreciable cost savings. This particular data management system eliminates any direct human intervention, but is somewhat awkward in that several physical transport steps are involved between different computers.

Introduction of an IBM compatible floppy disk into the short interval tester system will eliminate the need for paper tape as a storage medium and consequently the need for the PDP-15 computer conversion to magnetic tape as shown by the dashed line path of Figure 3.18. Up to 300K bytes of data can be stored on a single sided diskette. Not only could they be used to store cell parametric values, but they could also store all the actual current-voltage data points so that the actual V-I characteristic curve could be reproduced later. In addition, software could also be stored on diskettes making it easier to update the system and add programs. Since almost all the PROM memory area is presently used program additions will require either an additional PROM board or a floppy disk system.

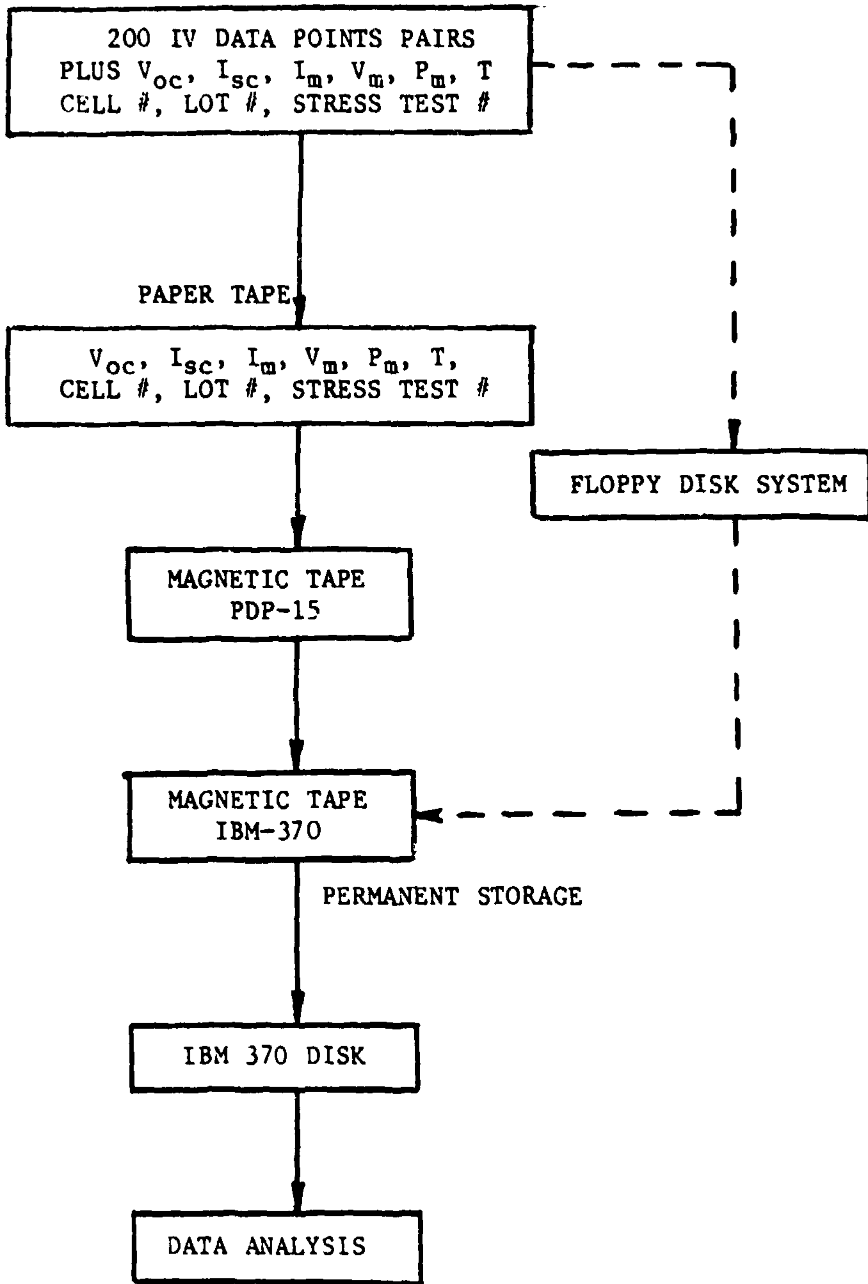


Figure 3.18. Present Method of Transferring Short Interval Tester Data to IBM-370 Computer

If the need arises in the future for a substantial number of hard copies of IV characteristics, a graphics system can be incorporated into the system replacing both the oscilloscope and the x-y plotter.

3.7 Cost

As microcomputers drop in price and improve in performance they are being used in applications where the larger minicomputers were used previously. In the case of the short interval tester a system was needed to perform dedicated functions and the lower cost microcomputer was ideal. Since the system was built in house, the total cost was reasonably low. Table 3.4 shows the cost of the major capital equipment items required to construct the system.

Item	Cost
Microcomputer	$ 1600
Analog Interface Board	$ 1000
RAM Memory Board (16k x 8)	$ 800
Programmable Power Supply	$ 600
Digital Controller	$ 400
Teletype	$ 1400
Plotter	$ 1700
Oscilloscope	$ 800
	$ 8300

Table 3.4. Cost of Major Capital Equipment

A major consideration in the design of the system was minimizing capital equipment cost. For this reason several peripheral such as the Teletype and X-Y plotter were used instead of a more satisfactory CRT, line printer and digital plotter combination. Also a floppy disk system was omitted in the original design. These items could be added to the system later as money becomes available to improve the system's capabilities.

Another major cost to a computer system is the cost of software. While the cost of the hardware has decreased due to technolgy the cost of the software has risen as labor rates increase. The cost of software can be a significant portion of the total first cost. In part the Intel 809/10 microcomputer was chosen because of the facilities at Clemson for writing software in a high level language, PLM/80, which saved both writing and debugging time. Even so, it is estimated that to reproduce the software that has been developed thus far for the tester would require in excess of $100,000 in an industrial environment.

4.0 ANTIREFLECTIVE COATING EXPERIMENTATION

H. A. Walker

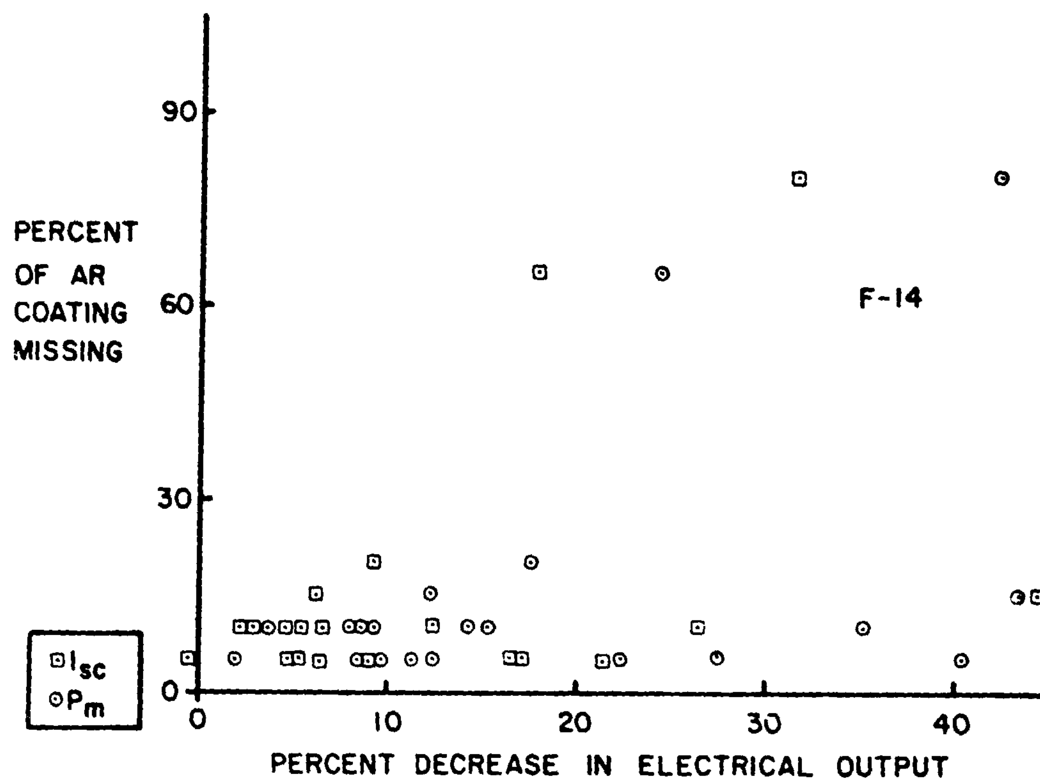
PRECEDING PAGE BLANK NOT FILMED

4.0 ANTIREFLECTIVE COATING EXPERIMENTATION — H.A. Walker

The possibility that the electrical characteristics of photovoltaic cells degrade as the antireflective (AR) coating is removed was theorized during earlier testing of F, G, and H cells. (See Second Annual Report, April 1980.) An an attempt to quantify this, new visual inspections were made of each cell which had undergone 500 hours of pressure cooker stress and an estimation made of the percentage of AR coating that had been leached away. These estimates were compared with the percent decrease of both I_{sc} and P_m. Figures 4.1-4.3 show the estimated AR coating missing vs. thepercent decrease of I_{sc} and P_m for each single cell. The plots do not all show a definite trend, but the G cells, which contained the least AR coating, seemed to exhibit a greater degree of correlation than either the F or H cells. Also, G cells are identical to I cells except for the absence of external leads. Therefore, these findings, though based on rough estimates, were significant enough to justify further research.

Mr. Edward Murphy of MIT Lincoln Labs first suggested the possibility of quantifying AR coating loss through colorspectrum analysis. We were fortunate in having available an IBM 7400 spectrophotometer which could be used for this purpose. The 7400 is a fast-scanning, high-precision, stable spectophotometer that operates in the visible wavelength range of 400 to 70 nonometers, and measures the light reflected from a sample at designated wavelength intervals (4.1). The light is filtered through a rotating circular interference filter assembly to provide continuous wavelength selection. Reflected light is fed to the color sensor by fiber optic cables from the diffuse reflectance sample holder. The sample holder contains a fixed fiber optic sensing head and a sample pedestal which can be adjusted vertically.

PRECEDING PAGE BLANK NOT FILMED



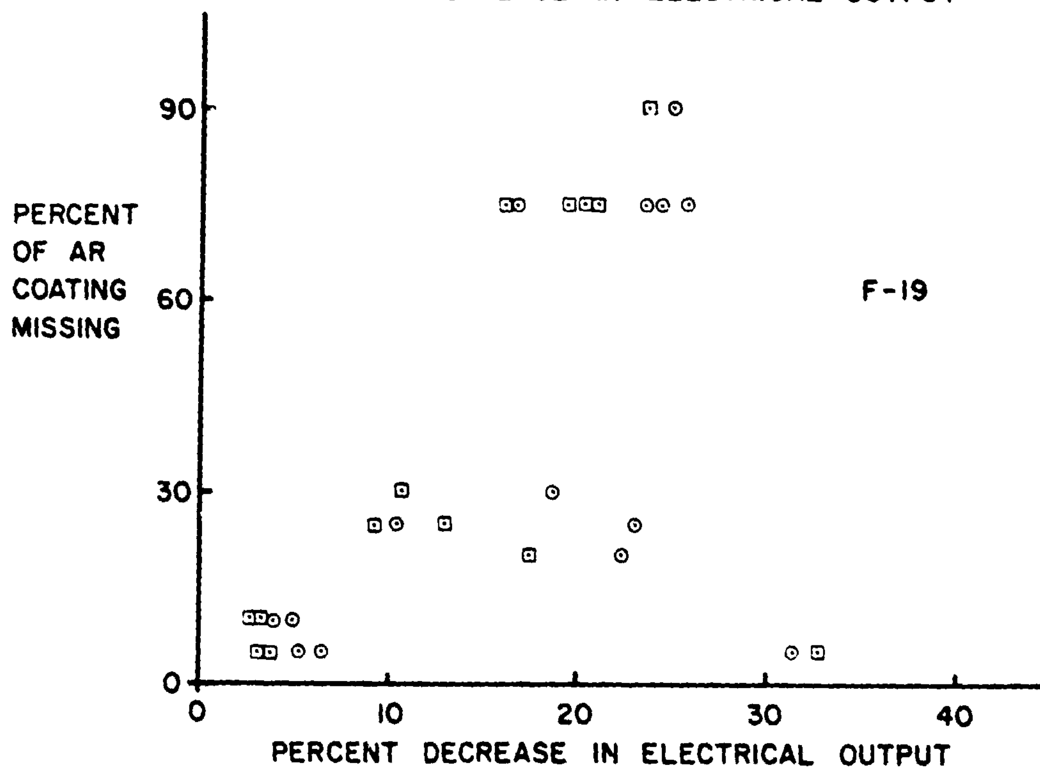
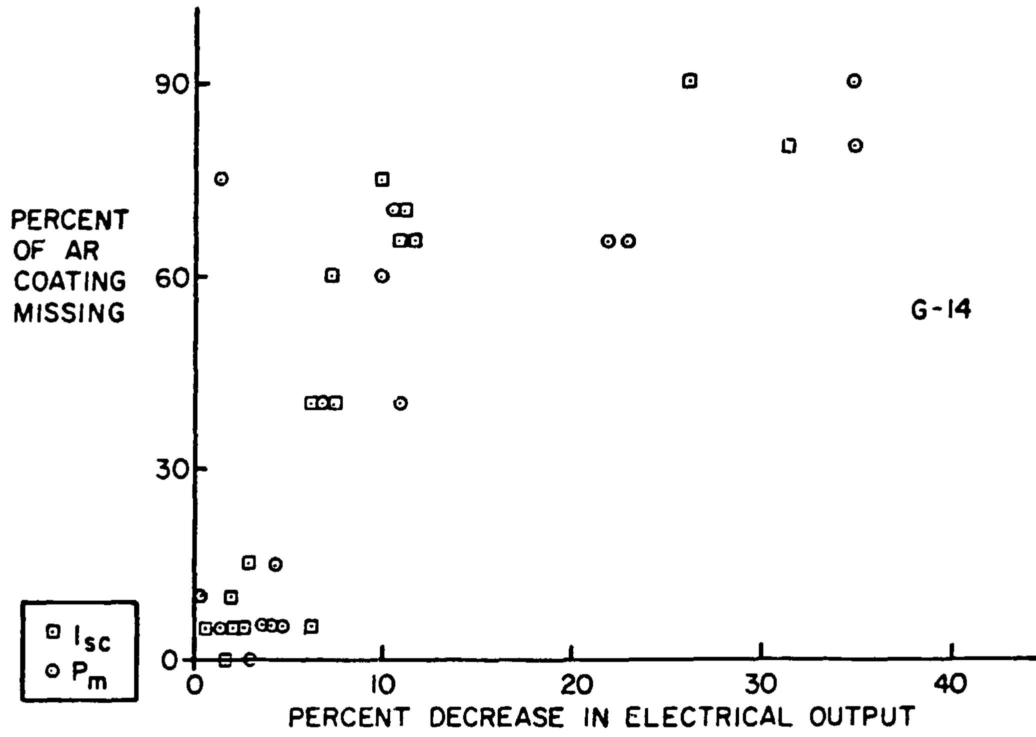


Figure 4.1 Visually Estimated % of AR Coating Removed vs. Measured % Decrease in I_{sc} and P_m for Lots F-14 and F-19 after 500 Hours Pressure Cooker



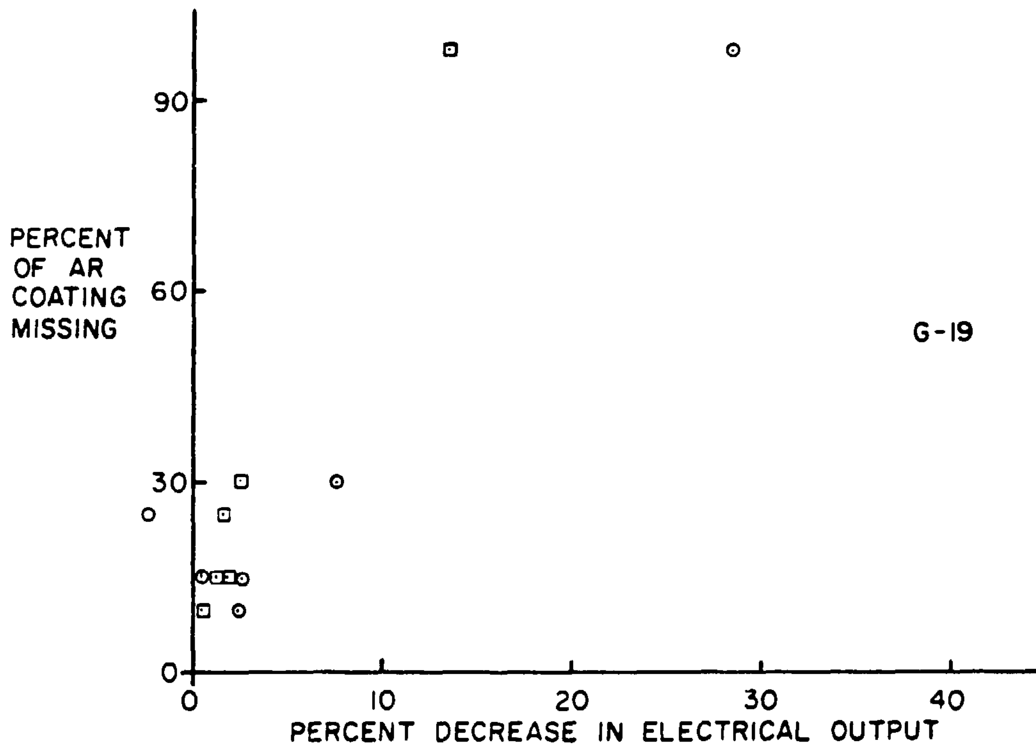
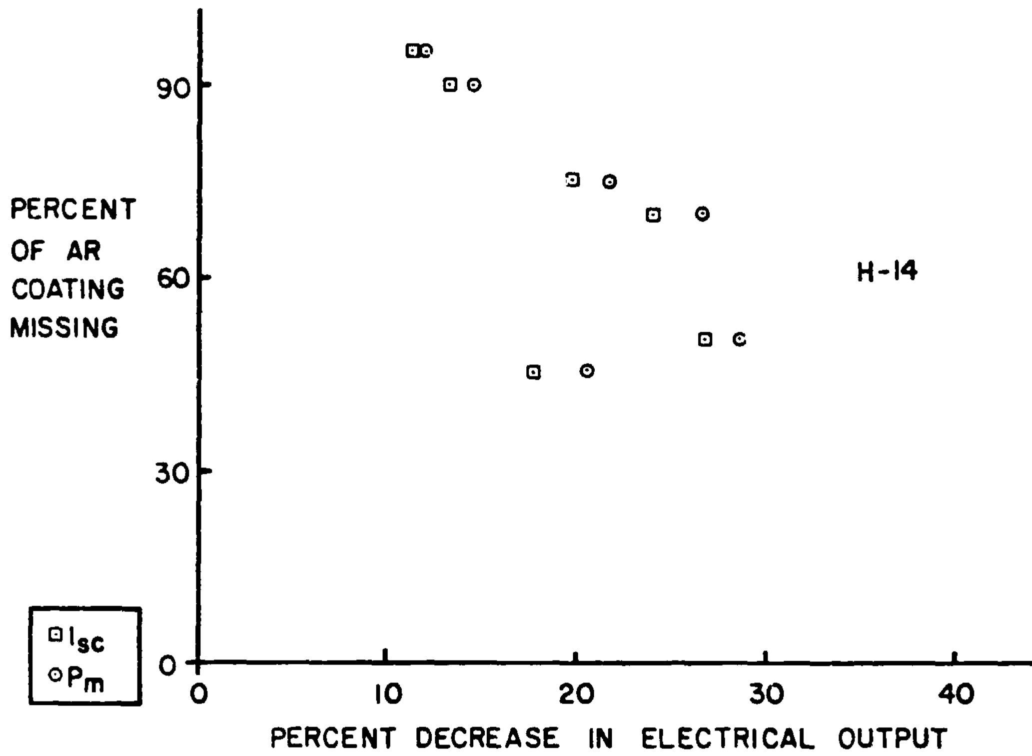


Figure 4.2 Visually Estimated % of AR Coating Removed vs. Measured % Decrease in Isc and Pm for Lots G-14 and G-19 after 500 Hours Pressure Cooker



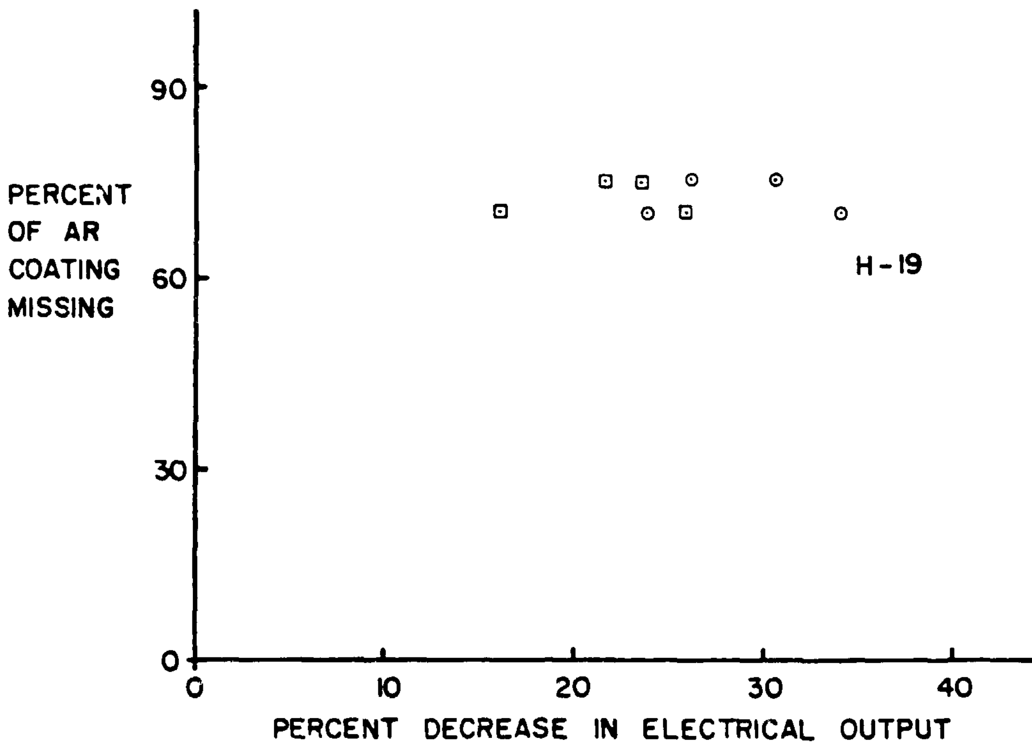


Figure 4.3 Visually Estimated % of AR Coating Removed vs. Measured % Decrease in Isc and Pm for Lots H-14 and H-19 after 500 Hours Pressure Cooker

The area measured is a circle with a diameter of 32mm. A solar cell could be placed in the pedestal and locked into position directly below the sensing head.

Control of the instrument is accomplished through a Color Sensor Analyzer which also computes reflectance data and provides hard copy color information. The spectral relfectance data obtained from the 7400 system is used to calculate color space coordinates, color difference, chromasticity coordinate values, and tristimulus values. (See Appendix C for a discussion of these terms.)

To test the feasibility of using this instrumentation another trial was held using a small group of C cells which had undergone various color changes as a result of pressure cooker stress. On some cells onlu a change in hue was observed, while others exhibited a substantial loss of AR coating. The resulting wide variety of colors suggested a good test of the IBM system's apparent visual accuracy. Various locations on each cell were analyzed and the tristimulus values and color space coordinates measured. Favorable comparisons were observed between these readings and the general visual appearance of each cell to the eye. Some concern had been expressed that the surface metallization would have a considerable effect on the measurements. Tc test this possibility, readings were taken on two cells before and after painting their metallization black. The only significant effect noted was a decrease in the overall lightness (L color space coordinate) of the cells, along with equal percentage drops in each tristimulus parameter.

The next step was to measure untested cells and to follow them through the stress testing procedure, measuring them at each down time. I-cells were used and three readings were taken for each cell. Since the I cells had not

been stressed, their readings were more uniform over a given cell. Measurement repeatability was the main concern, so several sets of measurements were made over a period of several days to determine the importance of exact cell positioning. These tests determined that it was necessary not only to measure the identical area each time, but also to place the cell on the pedestal in exactly the same orientation each time. This was accomplished by fitting an allignment mask over the cell as shown in Figure 4.4 so that only the desired area was seen by the measuring sensor. Orientation of the cell to the pedestal was maintained constant by lining up the center solder line on the cell with the edge of the sample holder. A similar mask was placed over the cell during electrical measurements so that only the areas for which the color readings were taken, would affect the output. These methods resulted in consecutive color measurements (of the same cell area on dilfferent days) that agreed within two percent.

Another factor to be considered was observer angle. The 1931 CIE standard angle referred to a 2° visual field and dealt mainly with distant viewing, whereas the 1964, 10° standard was set for relatively close observations. To investigate possible differences between the standards, two sets of readings were made for each cell. Only slight differences were noted between measurements of the same area. Therefore, the 10° angle was used for all susequent spectral analysis.

The measured I cells are currently undergoing pressure cooker tests and results will be reported in the next annual report.

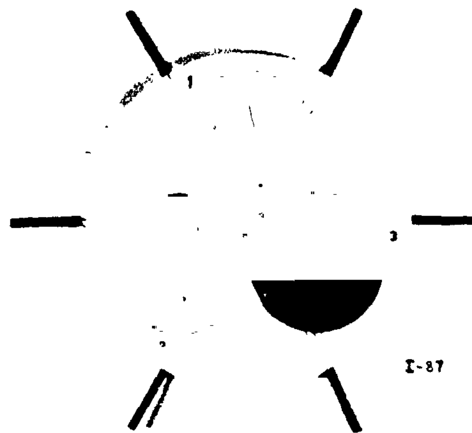


Figure 4.4 Alignment Mask in Position on an I-Cell.

5.0 CONCLUSIONS

PRECEDING PAGE BLANK NOT FILMED

5.0 CONCLUSIONS

5.1 Development of a Mathematical Model to Assess the Reliability Consequences of Operation in the Second Quadrant

A computer model based on the heat flow properties of silicon and the encapsulation materials, and on the variation of the electrical properties of silicon as a function of temperature, has been developed which relates cell temperature to cell VI characteristics in the second quadrant. This model predicts that as a cell's reverse bias voltage is increased the cell temperature will increase, uniformly at first, and then non-uniformly with subsequent hot spot formation as the voltage approaches breakdown. Agreement with experiment is excellent for unencapsulated cells and reasonable for encapsulated cells. Model parameters are less well known for encapsulated cells and the measurement problems greater, which accounts for the differences observed with encapsulated cells. Nevertheless, the model provides the only reasonable method of estimating cell temperatures in modules where cells become reverse biased. There seemed little difference between encapsulated and unencapsulated cells regarding the values of β and γ, the fitting parameters, althrough the sample size consisted of only 8 cells, 4 encapsulated and 4 unencapsulated. For most cells $\beta = 0.9 \pm 0.2$ and $\gamma = 0.3 \pm 0.1$. Additional cell types should be investigated to see if β and γ vary from structure to structure.

The ability to relate cell temperature to electrical characteristics can provide the module designer with valuable guidelines. For example, experiments with EVA encapsulation have indicated it would not withstand temperatures in excess of 140°C. The model predicts this temperature will be reached in glass superstrate modules at 15 volts and in masonite modules at 10 volts

PRECEDING PAGE BLANK NOT FILMED

and therefore under no circumstances should the voltage be allowed to exceed these values. The other components of the encapsulant and interconnection systems such as the glass, masonite and solder also have their own critical conditions. For glass the time rate of change of temperature may be the critical parameter, for masonite, the evolution of moisture, and for solder, its melting temperature. Rather than examine each component at this point a general statement may be made that cell temperatures should be kept as low as possible by restricting voltage. Based on the results of both our limited minimodule work and our extensive unencapsulated cell accelerated stress program this means cell temperatures should be kept below 100°C, which for most cell types translates to voltages less than 7 volts. This would mean bypass diodes for at least every 14 cells. More frequent bypass diodes are preferrable and the new technology section describes a design which was formulated to fabricate integral bypass diodes on every cell.

5.2 Development of Improved Solar Cell Test Methods

The microcomputer controlled, constant temperature, short interval tester described in this report is a great improvement over the previously used manual method of plotting the I-V curve, reading and recording data points. It has decreased both the measurement and data reduction time by a factor of 5 or more, along with essentially eliminating human error. At the same time the user retains a feel for the measurements taking place because of the monitor oscilloscope that displays the measured data points.

Accuracy and repeatibility have been improved by measuring cells at a more nearly constant temperature. The fraction of a second required to obtain the data enables the light source to shine on the cell for such a short time

that the heat generated by the light source has little affect on the cell's temperature.

Time varying effects may now be observed and quantified using the new measurement system. It is not clear how stressing affects a cell's sensitivity to the phenomenon, but this can now be investigated.

5.3 Color Measurement of AR Coating

Commercial color measurement instrumentation can be used to characterize AR coatings reproducibly. While this characterization is appropriate only to the visible portion of the spectrum evidence suggests that color may be related to the cell's PV output. A series of quantitative tests is underway to determine if this is the case.

6.0 RECOMMENDATIONS

PRECEDING PAGE BLANK NOT FILMED

6.0 RECOMMENDATIONS

6.1 Improved Test Methods

The capability to directly access temperature data should be incorporated in the tester. A floppy disk should be added to simplify the transfer of data to the IBM 370 system. A floppy disk would greatly increase the general versatility of the system and permit additional tasks to be performed.

6.2 Second Quandrant Model

Improved temperature measurements as a function of reverse bias need to be made on a variety of different cell types under different illumination conditions, both encapsulated and unencapsulated. This will enable the fitting parameter used in the model to be better defined which in turn will allow an assessment of the temperature rise anticipated in field installations under reverse bias conditions. Together with accelerated test data this will permit an estimate of array power degredation over life to be made.

PRECEDING PAGE BLANK NOT FILMED

7.0 NEW TECHNOLOGY

PRECEDING PAGE BLANK NOT FILMED

7.0 NEW TECHNOLOGY

Figure 7.1 illustrates a solar cell with an integrated design of bypass diode. This design features a wrap-around top contact, a back contact in the center of the cell and an integrated bypass diode. The heavy line in the schematic in Figure 7.1 represents the preformed V-shaped interconnect in the isometric view. Since it only connects to one side of the cell this metal lead could easily be attached by automated equipment.

It is anticipated that the diode and the cell junction would be manufactured in the same diffusion step. The diffused layer would then be removed from around the diode and from the back contact area. After this metalization would be applied and the interconnect preform soldered using essentially the same technology as for state of the art cells.

In order to illustrate the advantage of a one diode per cell configuration a module of 36 cells connected in series was fabricated. This module was constructed so that it could be operated either with or without bypass diodes which were fabricated using solar cell material and processing to simulate the characterisitic expected from integrated diodes. Figure 7.2 shows the VI characteristics of the module under different shading conditions. While the exact amount of power loss experienced by the module will depend on the module load it is clear that the loss will be much greater without diodes than with.

PRECEDING PAGE BLANK NOT FILMED

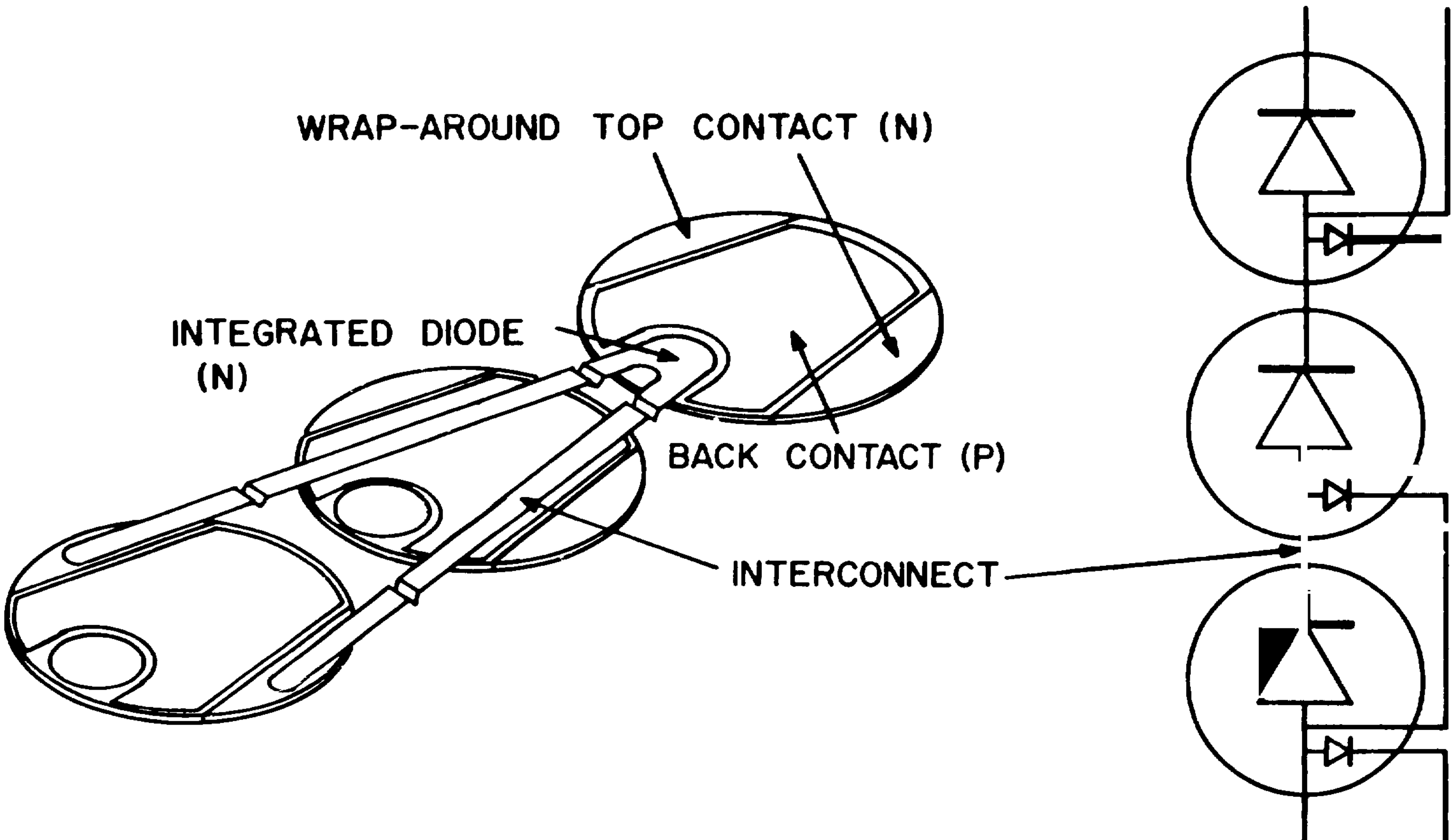


Figure 7.1 Proposal Solar Cell with Integrated Bypass Diode and Wrap Around Top Contact

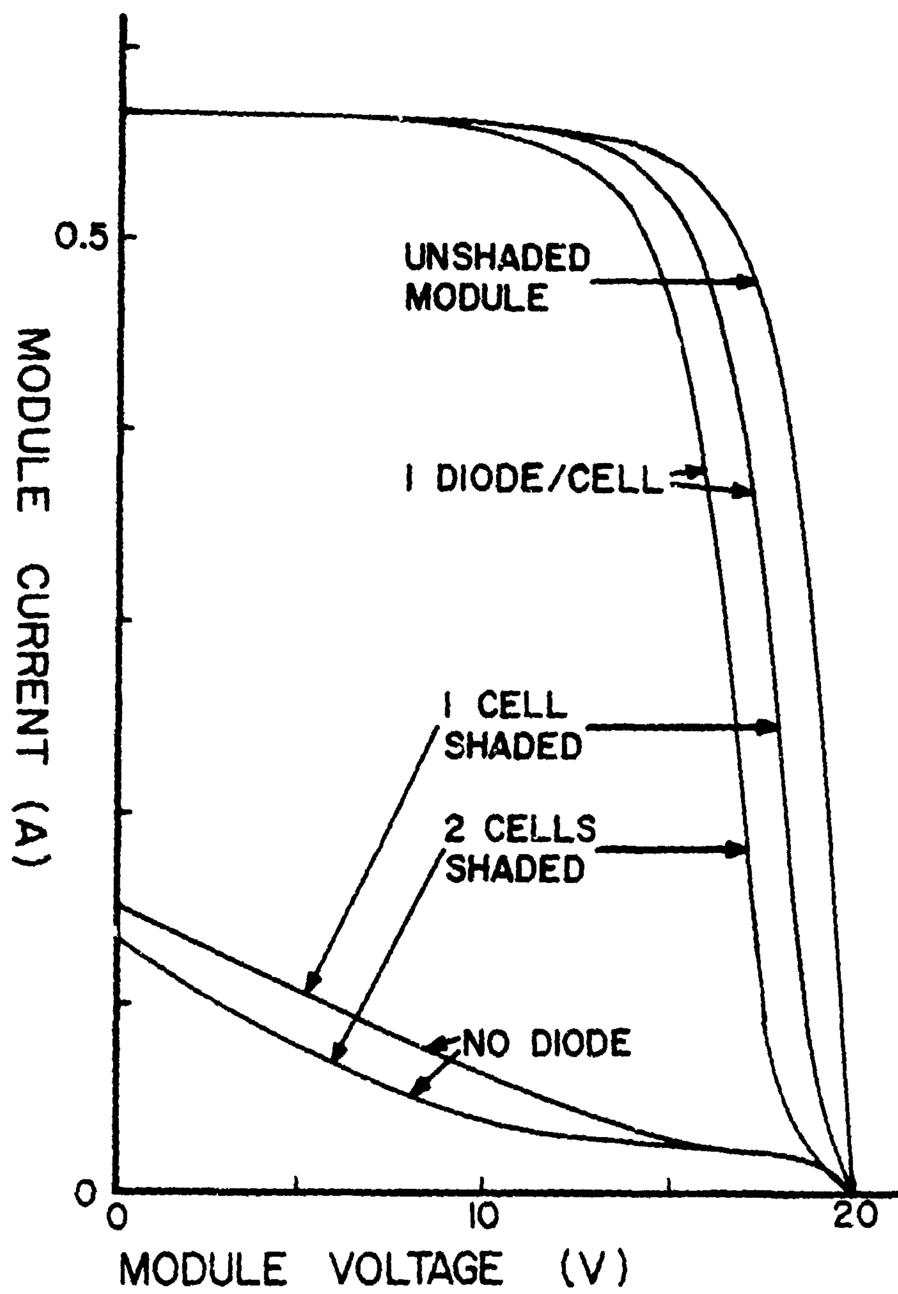


Figure 7.2 Output Characteristic of 36 Cell Module With and Witout Bypass Diodes.

8.0 REFERENCES

PRECEDING PAGE BLANK NOT FILMED

8.0 REFERENCES

2.1 Encapsulation Workshop, Jet Propulsion Laboratories, September 1980. Pasadena.

2.2 Blake F. A., Hanson K. L., "The 'Hot Spot' Failure Mode for Solar Arrays", Proceedings of the 4th Intersociety Energy Conversion Engineering Conference, September 1969, Washington, D.C.

2.3 Diamond, R. M., Steele E. D., "Solar Arrays with Integral Diodes", Proceedings of the International Colloquium on Solar Cells, July 1970, Toulouse.

2.4 Diamond R. M., "Advanced Developments of Integral Diode Solar Cells", Record of the 9th I.E.E.E. Photovoltaic Specialists Conference, May 1972, Silver Springs.

2.5 Jett P. L., Miller [illegible]. L., "Analysis of Effects of Shadowed and Open Solar Cells on Orbital Workshop Array Performance", Proceedings of the 6th Intersociety Energy Conversion Engineering Conference, August 1971, Boston.

2.6 North N. B., Baker D. F., "Solar Array System for the Skylab Orbital Workshop", Record of the 9th I.E.E.E. Photovoltaic Specialists Conference, May 1972, Silver Springs.

2.7 Garner, N. R., "Statistical Analysis Applied to Solar Cells Shorting Caused by Reverse Bias Voltage Stress", Record of the 9th I.E.E.E. Photovoltaic Specialists Conference, Mya 1972, Silver Springs.

2.8 Rauschenbach H. S., Maiden E. E., "Breakdown Phenomena in Reverse Biased Silicon Solar Cells", Record of the 9th I.E.E.E. Photovoltaic Specialists Conference, May 1972, Silver Springs.

PRECEDING PAGE BLANK NOT FILMED

2.9 Sayed M., Partain L., "Effects of Shading on CdS/Cu S Solar Cells and Optimal Solar Array Design", Energy Conversion, Vol. 14, No. 2, Pergamon Press 1975, Oxford, New York.

2.10 Ross R.G., "Interface Design Considerations for Terrestrial Solar Cell Modules", Record of the 11th I.E.E.E. Photovoltaic Specialists Conference, November 1976, Baton Rouge.

2.11 Schneider K., Schultze W., "The OTS Solar Array and the Solution of the 'Hot Spot' Problem", Proceedings of the International Conference on Solar Cells, September 1974, Hamburg.

2.12 Rath, J. Schulz B., "Formation and Control of 'Hot Spots' on Solar Arrays", Proceedings of the International Conference on Solar Cells, September 1974, Hamburg.

2.13 Gohrbrandt B., Georgens B., "The Electrical Power Interface between Solar Arrays and Power Conditioning", European Space Agency, Photovoltaic Generators in Space, November 1978, Noordwij.

2.14 Heasell E.L., "The Heat-Flow Problem in Silicon", I.E.E.E. Transactions on Electron Devices, Vol. ED-25, No. 12, December 1978, New York.

2.15 Lidback C.A., "Scanning Infrared Microscopy Techniques for Semiconductor Thermal Analysis", Proceedings of the 1979 International Reliability Physics Symposium, April 1979, Las Vegas.

2.16 Rosenburg D.U. von, "Methods for the Numerical Solution of Partial Differential Equations", American Elsevier Publishing Company, 1969, New York.

3.1 Intel System 80/10 Microcomputer Hardware Reference Manual, 98-316a, 1976.

3.2 RTI-1200 Real Time Interface User's Guide, Analog Devices Inc., 1977.

3.3 Curtis, H.B., "Errors in Short Circuit Current Measurements due to Spectral Mismatch Between Sunlight and Solar Simulators" ERDA/NASA-1022/76/10, Terrestrial Photovoltaic Mesurements II, 161-175, 1976.

3.4 Wiezer, V.G.; Brandhorst, H.W.; Broder, J.D.; Hart, R.E.; and Lamneck, J.H., "Photon Degradation Effects in Terrestrial Solar Cells", DoE Photovoltaic Conversion Program/NASA-Lewis Research Center, Project Document No. 4220-028, April 1977.

4.1 IBM 7410 Color Sensor Analyzer Produce Description and Operating Procedures, p. 1-1, F-1-F-3.

APPENDICES

PRECEDING PAGE BLANK NOT FILMED

APPENDIX A. TECHNIQUES FOR NUMERICALLY SOLVING PARTIAL DIFFERENTIAL HEAT FLOW EQUATIONS

PRECEDING PAGE BLANK NOT FILMED

APPENDIX A. MATHEMATICAL TOOLS

The thermal balance equations derived in Section 3.3 are of the form

$$\frac{\partial^2 u}{\partial x^2} + \frac{1}{x}\frac{\partial u}{\partial x} + \frac{1}{k(u)}\frac{dk}{du}\left(\frac{\partial u}{\partial x}\right)^2 + G(u) = \frac{a}{k(u)}\frac{\partial u}{\partial t}, \quad \text{(A-1)}$$

and are non-linear, parabolic partial differential equations. An excellent treatise on these and other similar types of differential equations is written by D.U. von Rosenburg (3.16), "Methods for the Numerical Solution of Partial Differential Equations".

The Crank-Nicholson technique was the method used for evaluation. In this technique all the finite differences are written about the point, x_i, $t_{n+\frac{1}{2}}$, half way between the known and the unknown time loads. In Figure A-1 this point is shown as a cross while the black dots denote known values of the dependent variable. Values of the dependent variable, u, are computed only at the points designated by open circles. The second-order-correct analog of the time derivative at the point x_i, $t_{n+\frac{1}{2}}$ is:

$$\left.\frac{\partial u}{\partial t}\right|_{i,\, n+\frac{1}{2}} = \frac{u_{i,n+\frac{1}{2}} - u_{i,n}}{t} - \left.\frac{\partial^3 n}{\partial t^3}\right|_{i,n+\frac{1}{2}} \frac{(\Delta t)^2}{24} - \cdots - \, . \quad \text{(A-2)}$$

A geometric interpretation of the analog is shown in Figure A-2. Thus the time derivative is approximately given by,

$$\left.\frac{\partial u}{\partial t}\right|_{i,n+\frac{1}{2}} \approx \frac{u_{i,n+1} - u_{i,n}}{\Delta t} \, . \quad \text{(A-3)}$$

The approximation of the second derivative, $\frac{\partial^2 u}{\partial x^2}$, without requiring the evaluation of the dependent variable at the time level for which the time

PRECEDING PAGE BLANK NOT FILMED

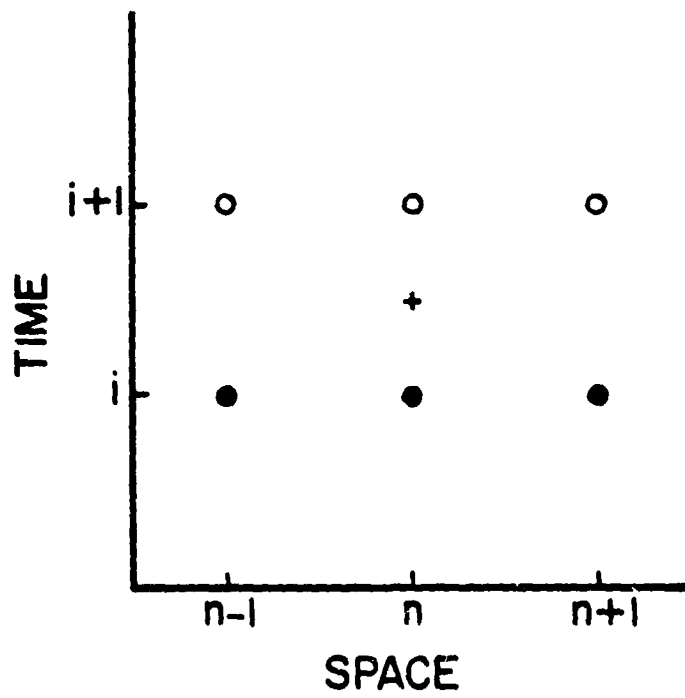


Figure A-1. Center of Analog for the Crank-Nicolson Technique

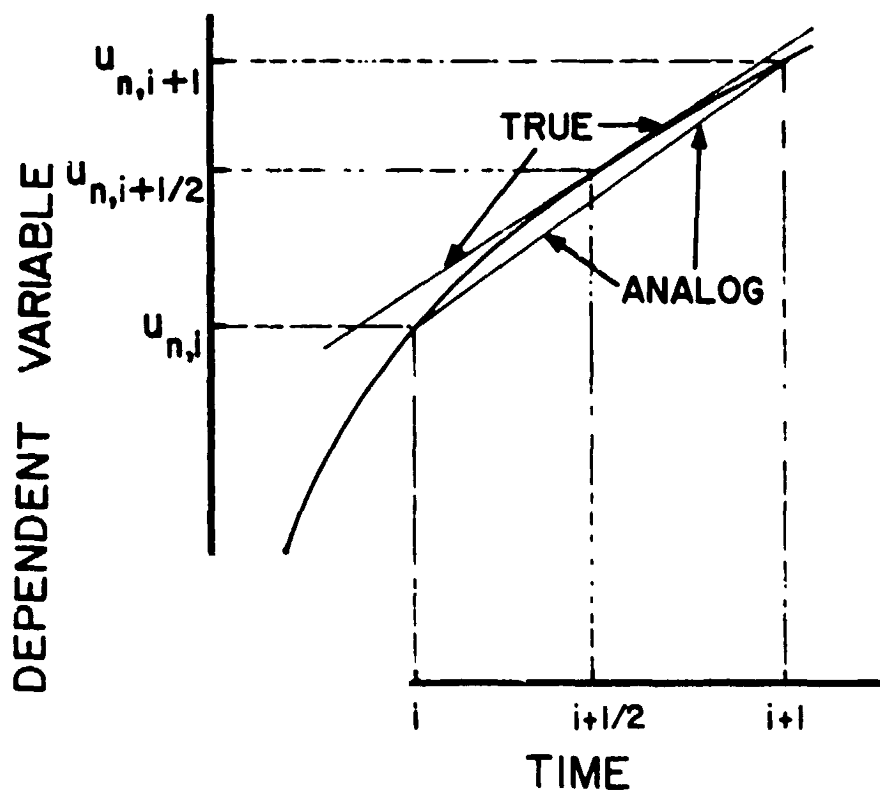


Figure A-2. Geometric Interpretation of a Second-Order-Correct First Derivative

index has the value of $n+\frac{1}{2}$, is the key to the Crank-Nicholson technique. This derivative is approximated by the artihmetic average of the finite difference analogs at the points x_i, t_n and x_i, t_{n+1}:

$$\left.\frac{\partial^2 u}{\partial x^2}\right|_{i,n+\frac{1}{2}} \approx \frac{1}{2}\left[\frac{u_{i+1,n} - 2u_{i,n} + u_{i-1,n}}{(\Delta x)^2} + \frac{u_{i+1,n+1} - 2u_{i,n+1} + u_{i-1,n+1}}{(\Delta x)^2}\right] . \quad \text{(A-}$$

The first derivative is approximated in the same manner:

$$\left.\frac{\partial u}{\partial x}\right|_{i,n+\frac{1}{2}} \approx \frac{1}{2}\left[\frac{u_{i+1,n} - u_{i-1,n}}{2\Delta x} + \frac{u_{i+1,n+1} - u_{i-1,n+1}}{2\Delta x}\right] . \quad \text{(A-5)}$$

This leaves the evaluation of the coefficients of the equation at the point x_i, $t_{n+\frac{1}{2}}$ and the problems with the term $\left(\frac{\partial u}{\partial x}\right)^2$. The coefficients can be determined by an iteration procedure in which the time dependent value is first calculated using the value of the coefficient at the point x_i, t_n. This gives a value for u at x_i, t_{n+1}, after which the equation is solved again by using values for the coefficients that are calculated by the use of the arithmetic average for u at x_i, t_n and x_i, t_{n+1}. Next the time dependent value is calculated again, and so on, until a predetermined accuracy is reached. This method was found to be unnecessary, however. Instead, the value at x_i, t_n was used and the time step was taken small to make up for the inherent danger of oscillation. In the same manner the square of the first space derivative was handled. If equation (A-5) were used it would have been necessary to solve a matrix of second order equations, which would have been very time consuming, if not impossible.

Using the described method the following set of equations was obtained:

$$b_o u_o + c_o u_1 + 0 + \ldots\ldots\ldots\ldots\ldots + 0 = d_o$$

$$a_1 u_o + b_1 u_1 + c_1 u_2 + 0 + \ldots\ldots\ldots + 0 = d_1$$

$$0 + a_2 u_1 + b_2 a_2 + c_2 u_3 + 0 + \ldots + 0 = d_2$$

$$\vdots \qquad\qquad\qquad\qquad \vdots$$

$$0 + \ldots\ldots\ldots\ldots + 0 + a_N u_{N-1} + b_N u_N = d_n$$

A set of equations of this sort, a tridiagonal matrix, is readily solved by a computer with the algorithm of Thomas. The algorithm is as follows:

first compute, $\beta_i = b_i - \frac{a_i c_{i-1}}{\beta_{i-1}}$ with $\beta_o = b_o$,

$$\gamma_i = \frac{d_i - a_i x_{i-1}}{\beta_i} \qquad \text{with} \qquad \gamma_o = \frac{d_o}{b_o} ,$$

to obtain the dependent variable with

$$u_n = x_n \frac{c_n u_{n+1}}{\beta_n}^{*} \qquad \text{and} \qquad u_i = x_i \frac{c_i u_{i+1}}{\beta_i}$$

(* from previous period.)

The solution of equation (A-1) can be approximated using the methods described, but care must be taken to limit the time step in order to suppress instabilities caused by the coefficients and by the square of the first spatial derivative. It is not necessary to take an excessive number of intervals along the spatial axis however.

The core of the program is the subroutine that calculates the temperatures at the point t_{i-1} . Figure A-3 lists the flow chart of this subroutine for the unencapsulated cell. It is largely self explanatory except for the boundary conditions which are discussed in Section 2.5. The subroutine contains two limiters: a temperature limiter and a current limiter. The temperature limiter is necessary to reduce the risk of overflow, as for example when the exponential functions in the current density equation are calculated. The current limiter is essential for physical reasons. As mentioned earlier, a solar cell is essentially a current source, and its operation in the second quadrant will typically be current limited rather than voltage limited. A secondary reason for the use of a current limiter is computational; the I-V curve of a cell in the second quadrant is single valued in voltage , but often double valued in current.

The computational current limiter works in a similar fashion to a physical current limiter. If current exceeds a preset value, the voltage is reduced until the current falls to the present value. The current limiter has a typical overshoot of about 5 computational periods. Both the temperature and current limiters set a flag when they are used.

Figure A-4 shows the structure of the complete program. As mentioned, the program solves the I-V curve by stepping along the current axis. For each current level the program solves the steady state solution with decreasing time interval for fast converquence. The program is capable in solving transient solutions, but should use a fixed, small time interval to be efective.

```
      SUBROUTINE NEWTEM
C*********************************************************************
C*********************************************************************
C**                   NEW TEMPERATURE CALCULATION                   **
C*********************************************************************
C*********************************************************************

      IA=1
      I=0
      TJ=TB(IA)
C
C     CALCULATION OF CURRENTDENSITY
C
      CALL CURDEN
C
C     CALCULATION OF THERMAL CONDUCTION
C
      WK=WSI/(.58+TJ/350)
C
C     CALCULATION OF THERMAL DIFFUSION
C
      A=WK/(1.647*WSI)
C
C     CALCULATION OF FIRST SET OF MATRIX ELEMENTS
C
      AI(IA)=0
      BI(IA)=-2/DR**2-1/(A*DT)
      CI(IA)=2/DR**2
      DI(IA)=(2/DR**2-1/(A*DT))*TJ-2*TB(IA+1)/DR**2
     C-(CD*V-HL*TJ+HL*TINF+IRR)/WK
C     WRITE(6,10)AI(IA),BI(IA),CI(IA),DI(IA)
      DO 300 I=1,74,1
      IA=I+1
      TJ=TB(IA)
C
C     CALCULATION OF CURRENTDENSITY
C
      CALL CURDEN
C
C     CALCULATION OF THERMAL CONDUCTION
C
      WK=WSI/(.58+TJ/350)
C
C     CALCULATION OF THERMAL DIFFUSION
C
      A=WK/(1.647*WSI)
C
C     CALCULATION OF THE REMAINDER OF THE MATRIX ELEMENTS
C
      AI(IA)=(I-1)/((2*I-1)*DR**2)
      BI(IA)=-1/DR**2-1/(A*DT)
      CI(IA)=I/((2*I-1)*DR**2)
      DI(IA)=-AI(IA)*TB(IA-1)+(1/DR**2-1/(A*DT))*TJ
     C-CI(IA)*TB(IA+1)-(CD*V-HL*TJ+HL*TINF+IRR)/WK
     C+(TB(IA+1)-TB(IA-1))**2/((205+TJ)*4*DR**2)
C     WRITE(6,10)AI(IA),BI(IA),CI(IA),DI(IA)
  300 CONTINUE
```

Figure A-3. Temperature Calculations Subroutine

```
      I=75
      IA=I+1
      TJ=TB(IA)
C
C     CALCULATION OF CURRENTDENSITY
C
      CALL CURDEN
C
C     CALCULATION OF THERMAL CONDUCTION
C
      WK=WSI/(.58+TJ/350)
C
C     CALCULATION OF THERMAL DIFFUSION
C
      A=WK/(1.647*WSI)
C
C     CALCULATION OF THE 76TH MATRIX ELEMENTS
C
      AI(IA)=1/(2*DR**2)
      BI(IA)=-1/(2*DR**2)-1/(A*DT)
      CI(IA)=0.0
      DI(IA)=-AI(IA)*TB(IA-1)-(BI(IA)+2/(A*DT))*TJ
     C-(CD*V-HL*TJ+HL*TINF+IRR)/WK
     C+75.5*HL*WSI*(1.5*TB(IA)-.5*TB(IA-1)-TINF)/(74.5*WK*DR)
C     WRITE(6,10)AI(IA),BI(IA),CI(IA),DI(IA)
C
C     CALCULATION OF TA
C
      IA=1
      BETA(IA)=BI(IA)
      GAMMA(IA)=DI(IA)/BI(IA)
      DO 380 IA=2,76,1
      BETA(IA)=BI(IA)-(AI(IA)*CI(IA-1))/BETA(IA-1)
      GAMMA(IA)=(DI(IA)-AI(IA)*GAMMA(IA-1))/BETA(IA)
  330 CONTINUE
      TA(76)=GAMMA(76)
      DO 430 M=1,75,1
      IA=76-M
      TA(IA)=GAMMA(IA)-(CI(IA)*TA(IA+1))/BETA(IA)
  430 CONTINUE
C
C     TEMPERATURE LIMITER
C
      DO 440 I=1,76,1
      IF(TA(I).LT.1000) GO TO 440
      II=1
      TA(I)=1000
  440 CONTINUE
C
C     CALCULATION OF TOTAL CURRENT
C*************************************************************************
C
      IA=1
      I=0
      ITOT=0
      TJ=TA(IA)
```

Figure A-3. Continued.

```
C
C      CALCULATION OF CURRENTDENSITY
C
       CALL CURDEN
       ITOT=(CD*PI*DR**2)/4
       DO 450 I=1,75,1
       IA=I+1
       TJ=TA(IA)
C
C      CALCULATION OF CURRENTDENSITY
C
       CALL CURDEN
       ITOT=ITOT+CD*I*PI*2*DR**2
   450 CONTINUE
       TC(L)=TA(1)
       TT(L)=ITOT
       TV(L)=V
C
C      CURRENT LIMITER
C*********************************************************************
C
       D=1.00000001*IMAX
       E=.99999999*IMAX
       IF (ITOT.GT.D) GO TO 490
       IF (ITOT.LT.E.AND.IF.EQ.1) GO TO 495
       GO TO 500
   490 V=(V*IMAX)/ITOT
       IF =1
       GO TO 500
   495 V=(V*IMAX)/ITOT
       IF(V.LT.VMAX) GO TO 500
       V=VMAX
       IF=0
   500 CONTINUE
       IM=IM+1
       DO 540 I=1,76,1
       TB(I)=TA(I)
   540 CONTINUE
       RETURN
```

Figure A-3. Continued.

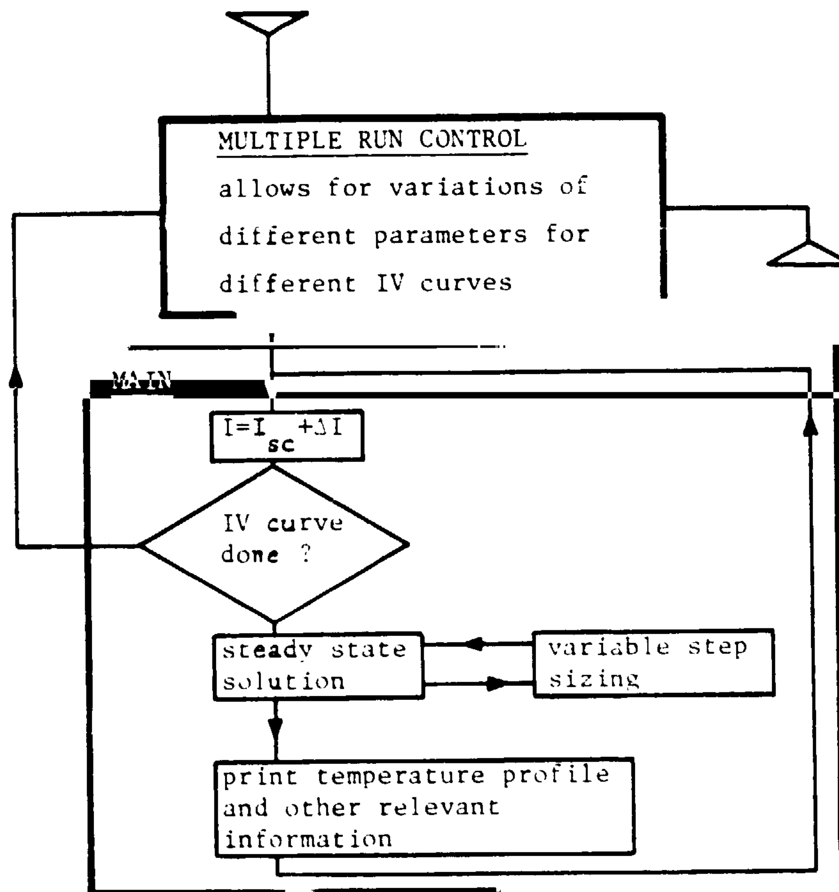


Figure A-4. Structure of the Complete Program

APPENDIX B. FLOW CHARTS AND PROGRAMS FOR THE MICROCOMPUTER CONTROLLED, CONSTANT TEMPERATURE, SHORT INTERVAL IV MEASUREMENT SYSTEM

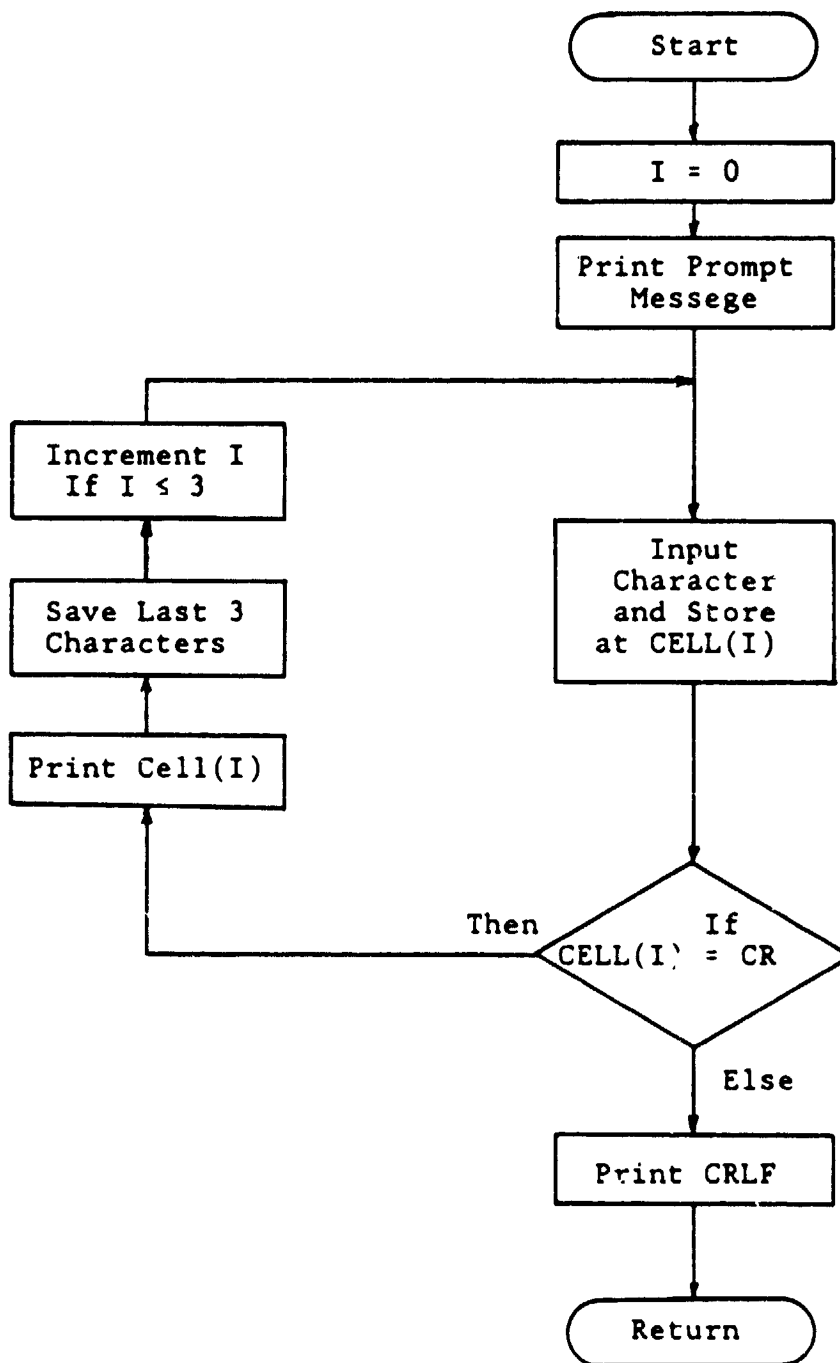


Figure B-1. Get Cell Number Procedure

PRECEDING PAGE BLANK NOT FILMED

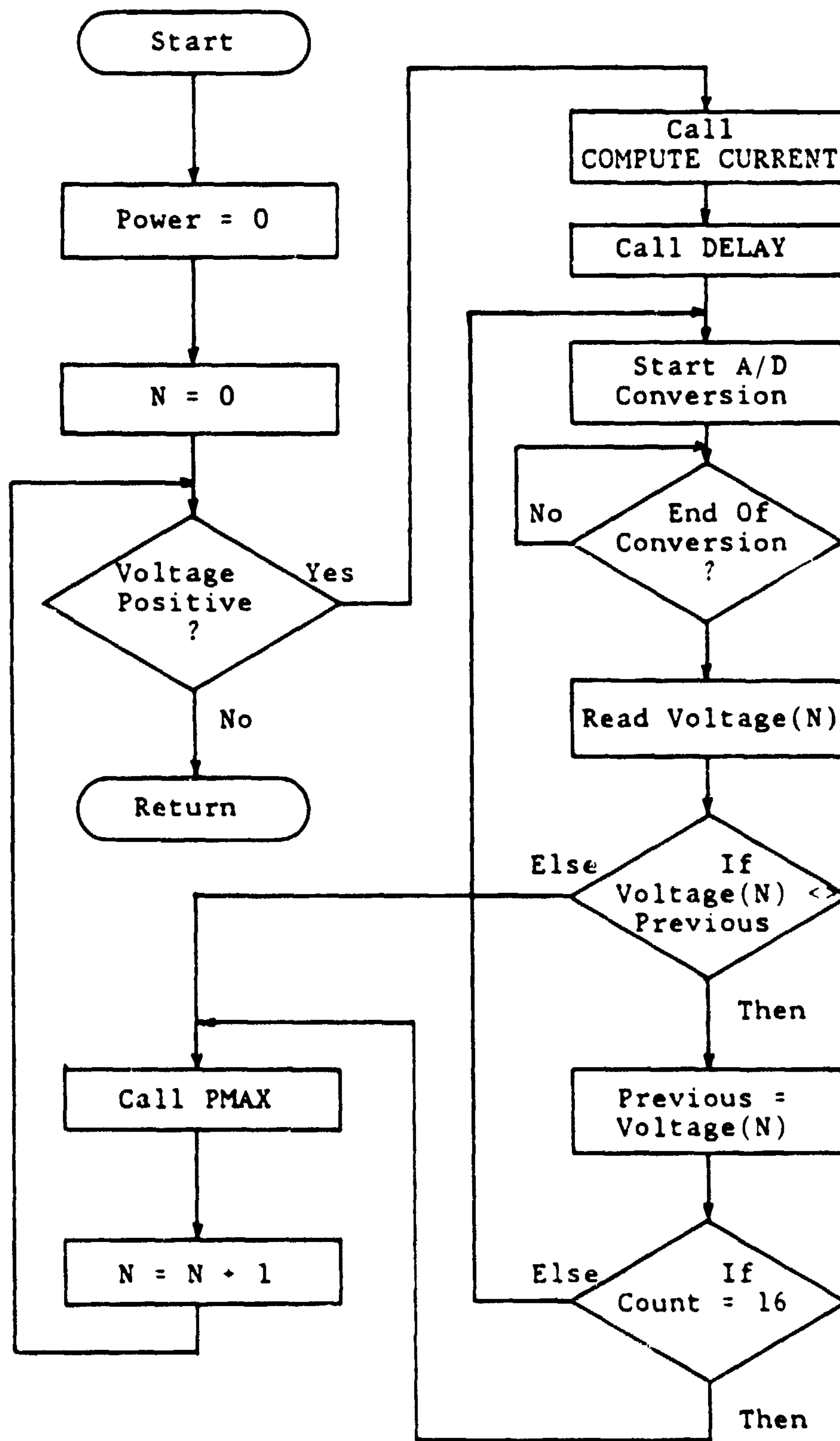


Figure B-2. Step Through Procedure

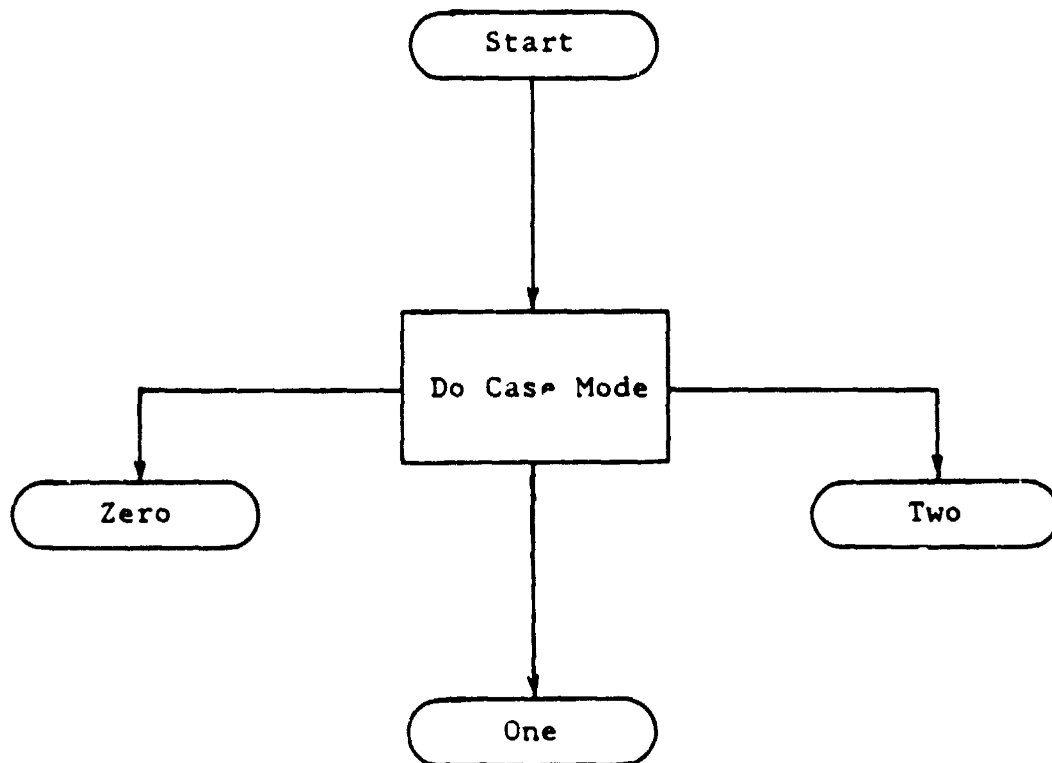


Figure B-3. Compute Current Procedure

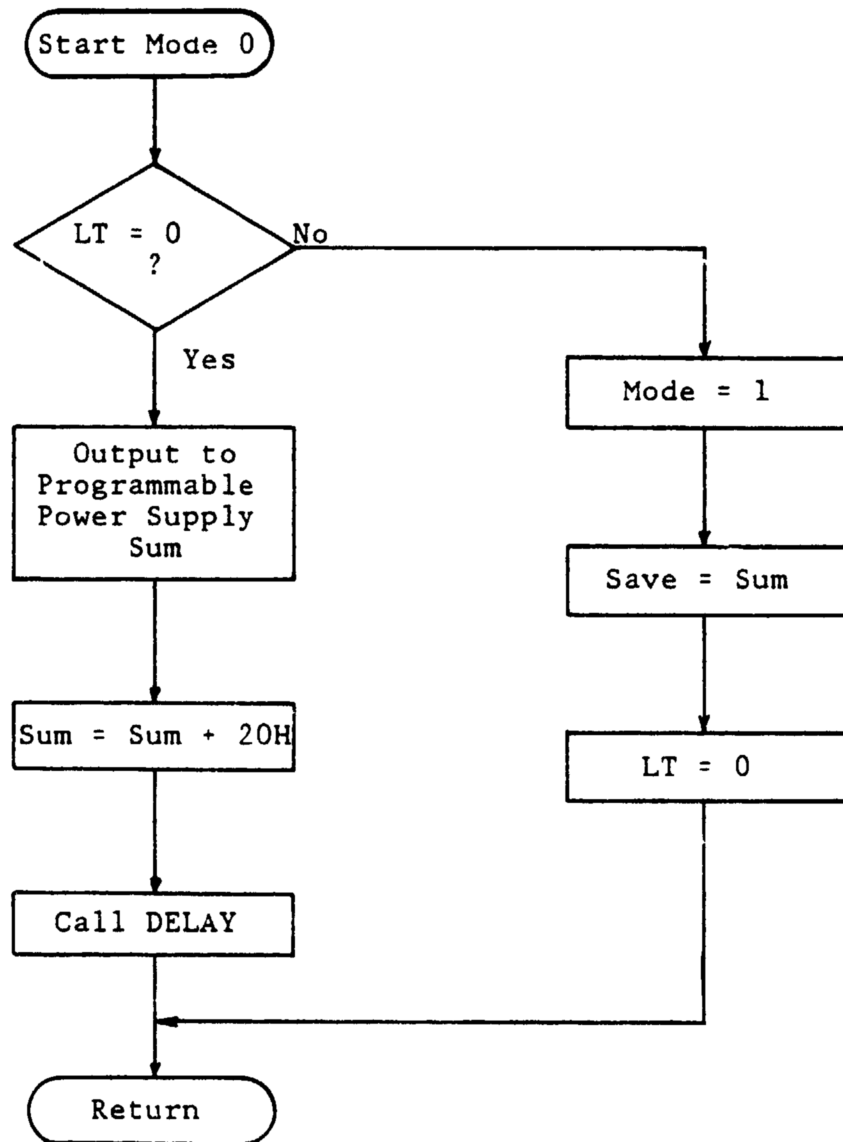


Figure B-4. Compute Current Procedure Mode 0

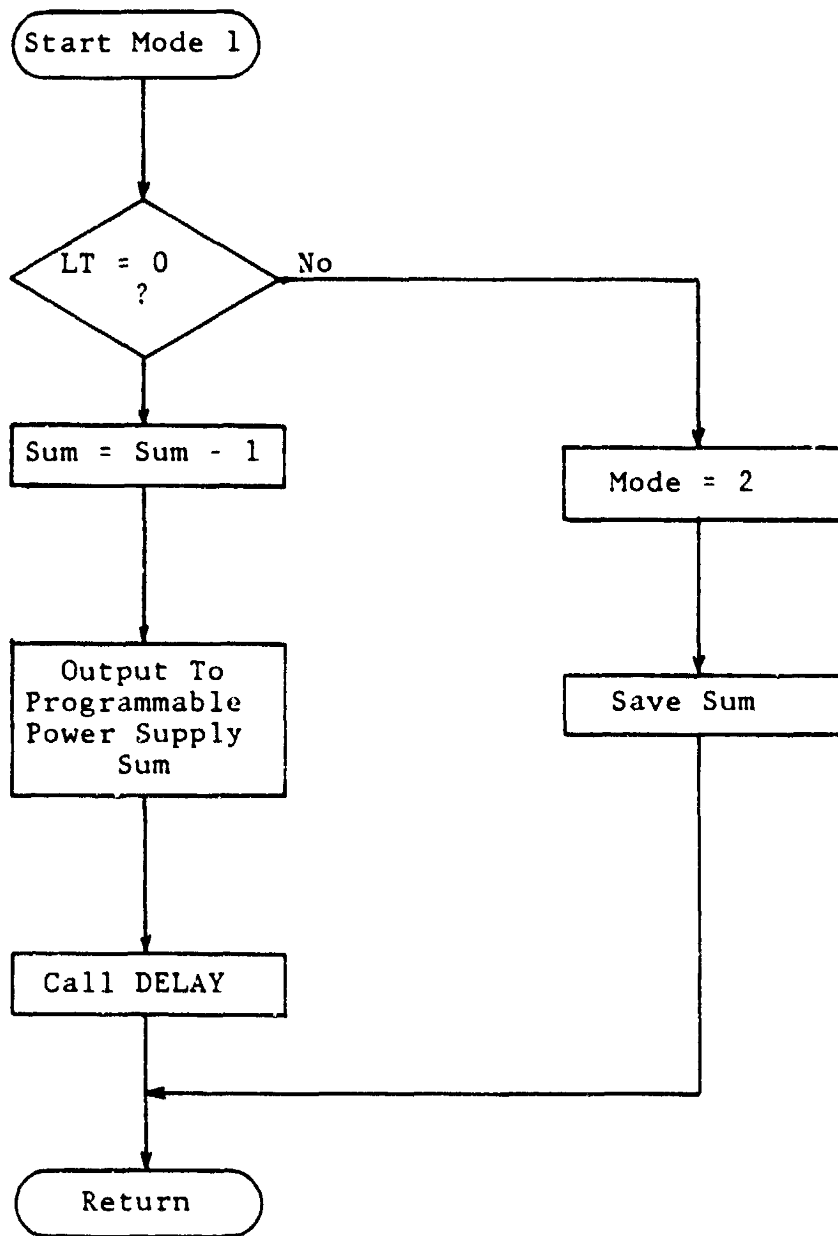


Figure B-5. Compute Current Procedure Mode 1

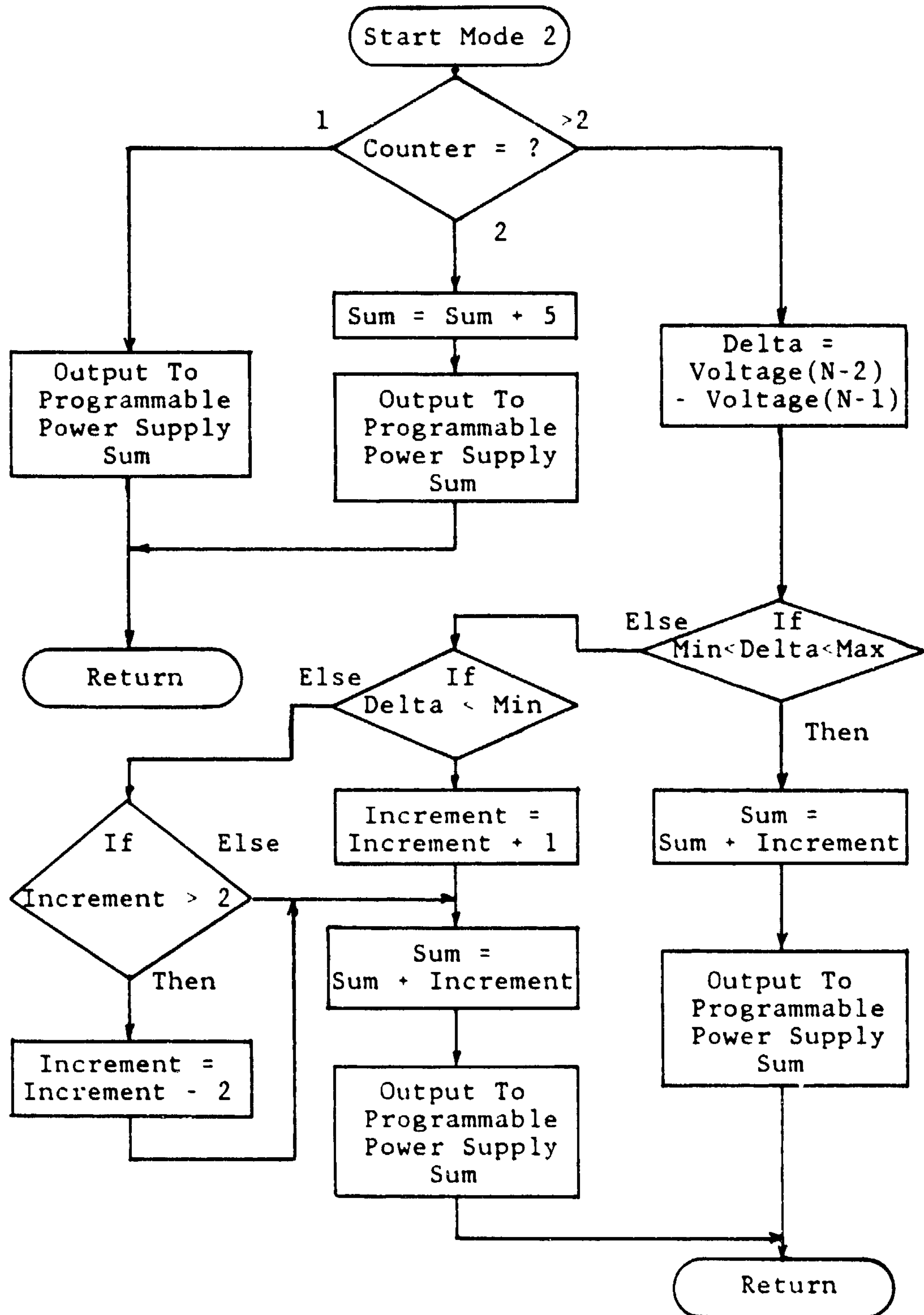


Figure B-6. Compute Current Procedure Mode 2

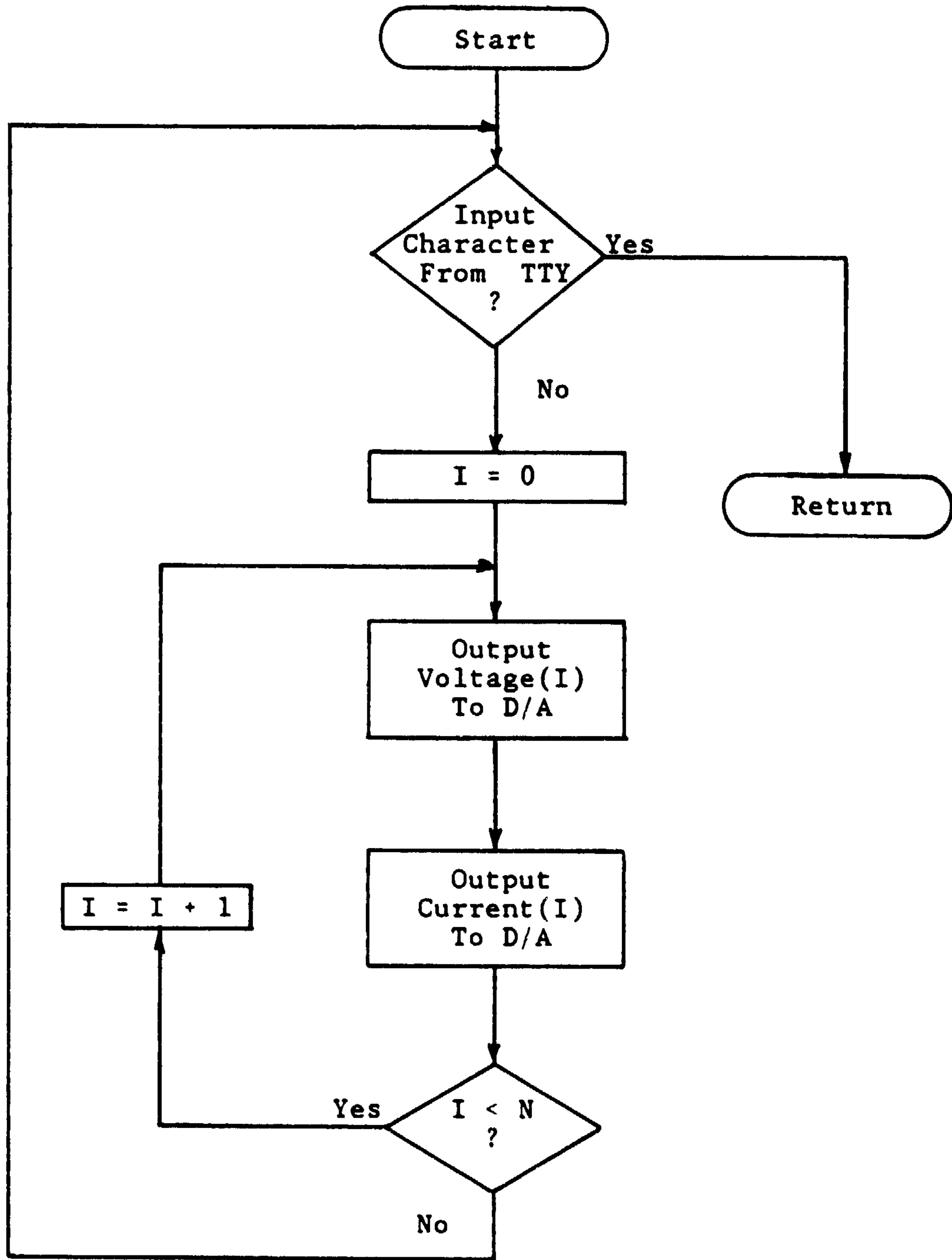


Figure B-7. Scope Procedure

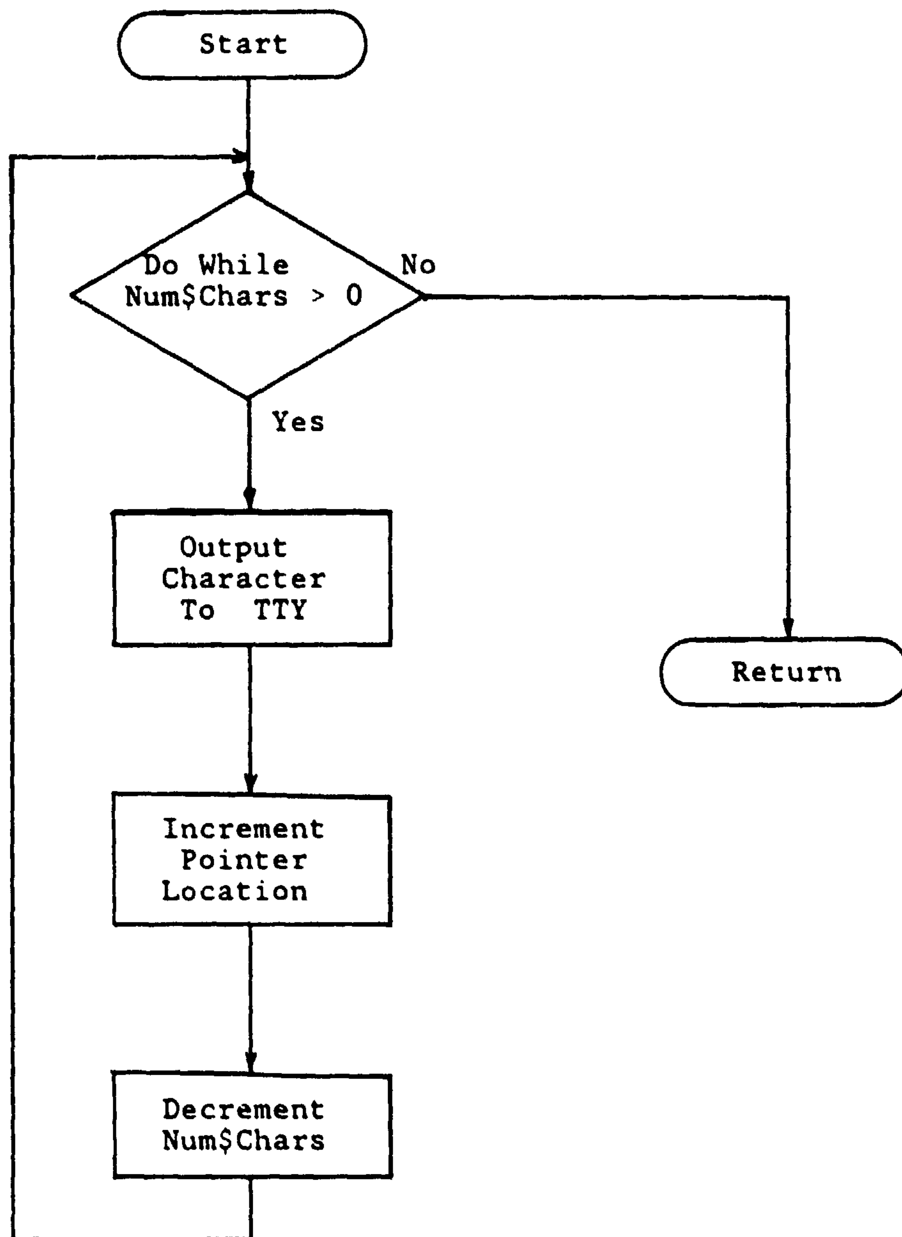


Figure B-8. Print Procedure

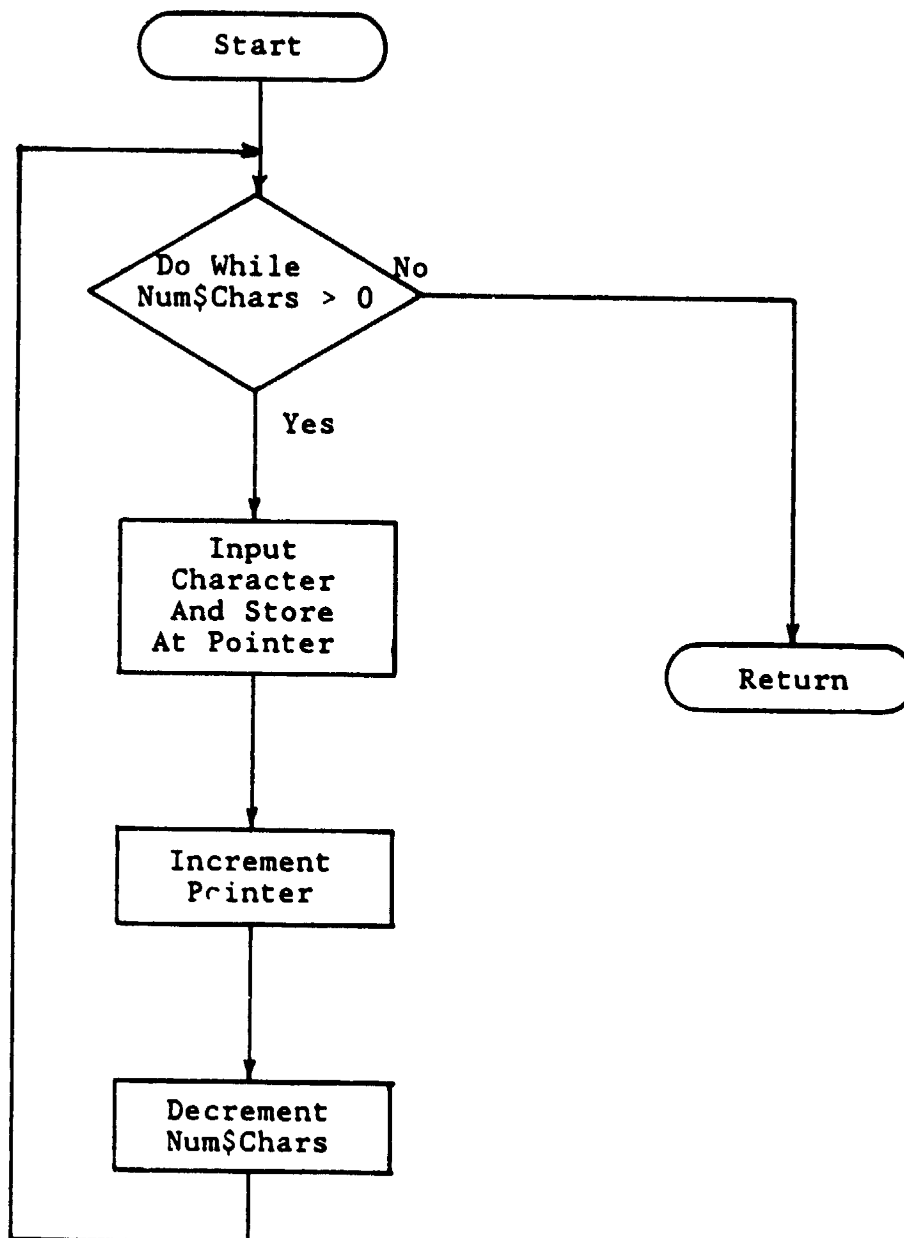


Figure B-9. Fetch Procedure

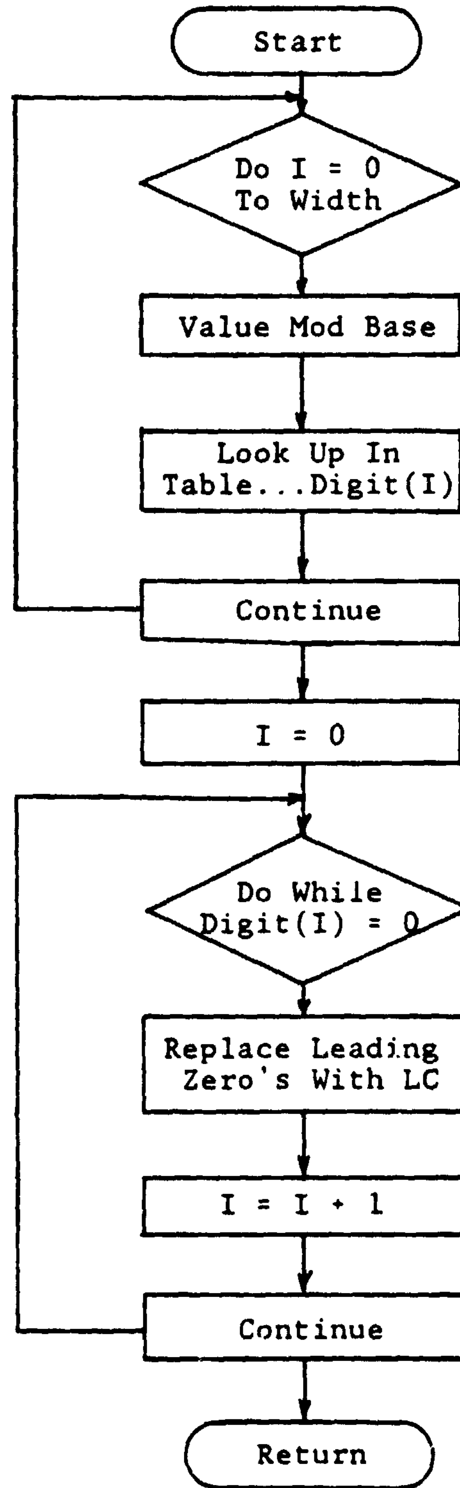


Figure B-10. Numout Procedure

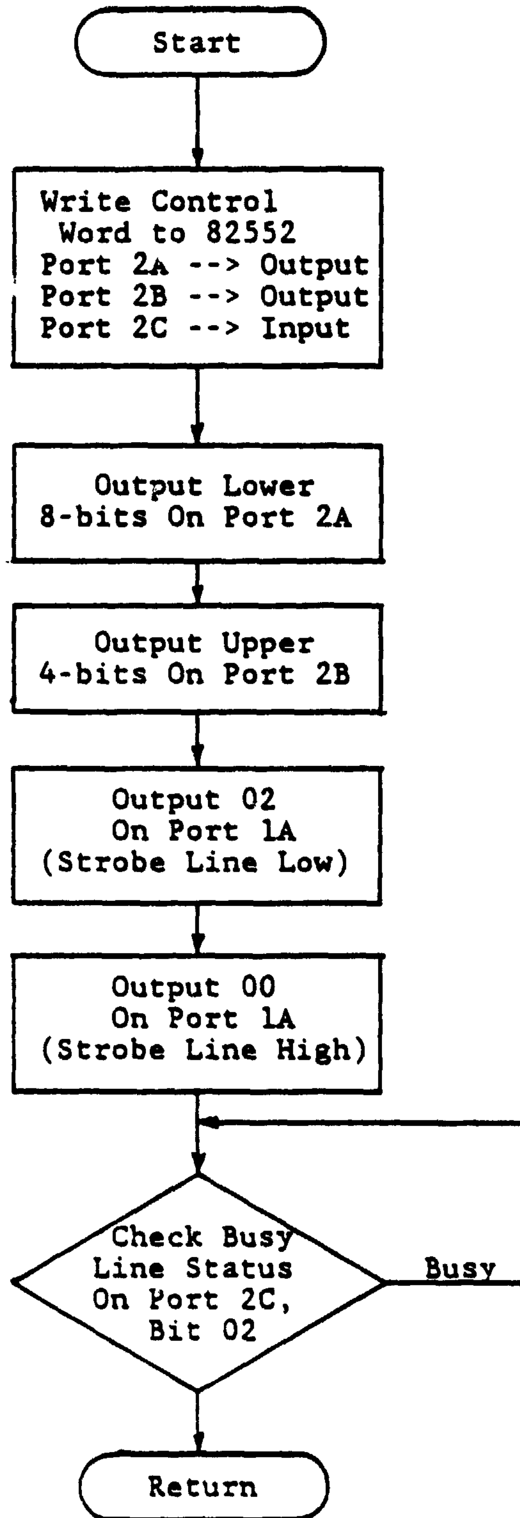


Figure B-11. Tbout Procedure

```
ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE GLOBALVARIABLESMODULE
OBJECT MODULE PLACED IN :F1:GLOBE.OBJ
COMPILER INVOKED BY:  PLM80 :F1:GLOBE.SRC



              /*****************************************************/
              /*                                                   */
              /*                  GLOBAL VARIABLES                 */
              /*                                                   */
              /*****************************************************/
  1           GLOBAL$VARIABLES$MODULE:
              DO;
  2   1          DECLARE POINT(250) STRUCTURE(
                    VOLTAGE(250) ADDRESS,
                    CURRENT(250) ADDRESS) PUBLIC;
  3   1          DECLARE (LT,MODE,H) BYTE PUBLIC;
  4   1          DECLARE INPUT$CHAR BYTE PUBLIC;
  5   1          DECLARE INCREMENT BYTE PUBLIC;
  6   1          DECLARE OUTPUT$CHAR BYTE PUBLIC;
  7   1          DECLARE COUNT BYTE PUBLIC;
  8   1          DECLARE N BYTE PUBLIC;
  9   1          DECLARE COMMAND(2) BYTE PUBLIC;
 10   1          DECLARE MEASUREMENT(250) STRUCTURE(
                    POWER(4) BYTE) PUBLIC;
 11   1          DECLARE MEAS BYTE PUBLIC;
 12   1          DECLARE MP(250) BYTE PUBLIC;
 13   1          DECLARE NSAVE(250) BYTE PUBLIC;
 14   1          DECLARE SUM ADDRESS PUBLIC;
 15   1          DECLARE STRESS(2) BYTE PUBLIC;
 16   1          DECLARE LOT(3) BYTE PUBLIC;
 17   1          DECLARE TYPE(2) BYTE PUBLIC;
 18   1          DECLARE TAPE$ON BYTE PUBLIC DATA(12H);
 19   1          DECLARE TAPE$OFF BYTE PUBLIC DATA(14H);
 20   1          DECLARE PERIOD BYTE PUBLIC DATA(2EH);
 21   1          DECLARE TEMP(5) BYTE PUBLIC;
 22   1          DECLARE CRLF(2) BYTE PUBLIC DATA(0DH,0AH);
 23   1          DECLARE CELL(5) BYTE PUBLIC;
 24   1          DECLARE CONTROL STRUCTURE(
                     SIGN BYTE,
                     SCALE ADDRESS,
                     SLENGTH BYTE,
                     STRING$PTR ADDRESS) STRING(5) BYTE PUBLIC;
 25   1       END;


MODULE INFORMATION

     CODE AREA SIZE     = 0000H      0D
     VARIABLE AREA SIZE = [illegible]  [illegible]
     MAXIMUM STACK SIZE = 0000H      0D
     [illegible] LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

ORIGINAL PAGE IS
OF POOR QUALITY

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE MAINPROGRAMMODULE
OBJECT MODULE PLACED IN :F1:PMAIN.OBJ
COMPILER INVOKED BY: PLM80 :F1:PMAIN.SRC

```
              /*********************************************/
              /*                                           */
              /*         MAIN PROGRAM MODULE               */
              /*                                           */
              /*********************************************/
  1           MAIN$PROGRAM$MODULE:
              DO;
  2   1       TYE     PROCEDURE EXTERNAL;
  3   2          END TYE;
  4   1       GETCM:  PROCEDURE EXTERNAL;
  5   2          END GETCM;
  6   1       ID
                 PROCEDURE EXTERNAL;
  7   2          END ID;
  8   1       GET$CELL$NUM:
                 PROCEDURE EXTERNAL;
  9   2          END GET$CELL$NUM;
 10   1       STEP$THROUGH:
                 PROCEDURE EXTERNAL;
 11   2       END STEP$THROUGH;
 12   1       SCOPE:
                 PROCEDURE EXTERNAL;
 13   2       END SCOPE;
 14   1       PMAX:   PROCEDURE EXTERNAL;
 15   2          END PMAX;
 16   1       POUT    PROCEDURE EXTERNAL;
 17   2          END POUT;
 18   1          DECLARE MEAS BYTE EXTERNAL;
 19   1          DECLARE COMMAND BYTE EXTERNAL;
 20   1          DECLARE T BYTE;
 21   1       BEGIN:  CALL GETCM;
 22   1          DO CASE COMMAND;
 23   2             GO TO IDENT;
 24   2             DO;
 25   3             CALL TYE;
 26   3             GO TO BEGIN;
 27   3             END;
 28   2          END;
 29   1       IDENT:  CALL ID;
 30   1       START:  CALL GET$CELL$NUM;
               ***************************************
               *         CONTROL WORD TO [illegible]      *
 31   1          OUTPUT([illegible]) = [illegible]
               ***************************************
               *                                     *
               *         OPEN LIGHT SHUTTER          *
 32   1          OUTPUT([illegible]) = [illegible]
               ***************************************
               *         TIME DELAY FOR LIGHT SHUTTER *
 33   1       DELAY:  DO T = 0 TO [illegible]
 34   2          CALL TIME([illegible])
```

PL/M-80 COMPILER

```
 35   2           END DELAY;
              /*                                                    */
 36   1           REAS = 0;
 37   1       CALL STEP*THROUGH;
              /****************************************************/
              /*              CLOSE LIGHT SHUTTER                   */
 38   1       OUTPUT(0E4H) = 0FFH;
              /****************************************************/
 39   1       CALL SCOPE;
 40   1       CALL POUT;
 41   1           GO TO START;
 42   1       END MAIN*PROGRAM*MODULE;
```

```
MODULE INFORMATION:

     CODE AREA SIZE     = 0065H    101D
     VARIABLE AREA SIZE = 0001H      1D
     MAXIMUM STACK SIZE = 0002H      2D
     63 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE COMMANDMODULE
OBJECT MODULE PLACED IN :F1:CMD.OBJ
COMPILER INVOKED BY:  PLM80 :F1:CMD.SRC


                /************************************************/
                /*                                              */
                /*              GET COMMAND MODULE              */
                /*                                              */
                /************************************************/
  1             COMMANDMODULE:
                DO;
  2    1           DECLARE CRLF(2) BYTE EXTERNAL;
  3    1           DECLARE COMMAND(2) BYTE EXTERNAL;
  4    1        PRINT:  PROCEDURE(LOCATION,NUMCHARS) EXTERNAL;
  5    2           DECLARE LOCATION ADDRESS;
  6    2           DECLARE NUMCHARS BYTE;
  7    2        END PRINT;
  8    1        FETCH:  PROCEDURE(LOCATION,NUMCHARS) EXTERNAL;
  9    2           DECLARE LOCATION ADDRESS;
 10    2           DECLARE NUMCHARS BYTE;
 11    2        END FETCH;
 12    1        GETCM:  PROCEDURE PUBLIC;
 13    2           DECLARE C(*) BYTE DATA(0CH,0AH,'?');
 14    2        START:  CALL PRINT(.C(0),03H);
 15    2           CALL FETCH(.COMMAND(0),2);
 16    2           CALL PRINT(.CRLF,2);
 17    2           IF COMMAND(0) < 30H THEN
 18    2           GO TO START;
                   ELSE
 19    2           IF COMMAND(0) > 31H THEN
 20    2           GO TO START;
                   ELSE
 21    2           COMMAND(0) = COMMAND(0) - 30H;
 22    2        END GETCM;
 23    1        END COMMANDMODULE;


MODULE INFORMATION:

     CODE AREA SIZE     = [illegible]H     [illegible]D
     VARIABLE AREA SIZE = [illegible]H     [illegible]D
     MAXIMUM STACK SIZE = [illegible]H     [illegible]D
     23 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE CELLIDMODULE
OBJECT MODULE PLACED IN :F1:ID.OBJ
COMPILER INVOKED BY: PLM80 :F1:ID.SRC

```
              /*****************************************************/
              /*                                                   */
              /*              SOLAR CELL IDENTIFICATION            */
              /*                                                   */
              /*****************************************************/
  1           CELL$ID$MODULE:
              DO;
  2   1           DECLARE STRESS(2) BYTE EXTERNAL;
  3   1           DECLARE LOT(2) BYTE EXTERNAL;
  4   1           DECLARE TYPE(2) BYTE EXTERNAL;
  5   1           DECLARE CRLF(2) BYTE EXTERNAL;
  6   1       PRINT: PROCEDURE(LOCATION,NUMCHARS) EXTERNAL;
  7   2           DECLARE LOCATION ADDRESS;
  8   2           DECLARE NUMCHARS BYTE;
  9   2       END PRINT;
 10   1       FETCH: PROCEDURE(LOCATION,NUMCHARS) EXTERNAL;
 11   2           DECLARE LOCATION ADDRESS;
 12   2           DECLARE NUMCHARS BYTE;
 13   2       END FETCH;
              /*                                                   */
              /*                                                   */
 14   1       ID: PROCEDURE PUBLIC;
 15   2           DECLARE HEAD(*) BYTE DATA('#LOT#S#CELL#
                    VOC  #   ISC   #    PM   #   VM   #
                      IM   #   TEMP   #');
 16   2           DECLARE T(*) BYTE DATA('CELL TYPE = ');
 17   2           DECLARE L(*) BYTE DATA('LOT # = ');
 18   2           DECLARE S(*) BYTE DATA('STRESS LEVEL = ');
              /*                                                   */
              /*                  CELL TYPE                        */
              /*                                                   */
 19   2           CALL PRINT(.T(0),0CH);
 20   2           CALL FETCH(.TYPE(0),1);
 21   2       AGAIN: CALL FETCH(.TYPE(1),1);
 22   2           IF (TYPE(1) <> 0DH) THEN
 23   2               DO;
 24   3               TYPE(0) = TYPE(1);
 25   3               GO TO AGAIN;
 26   3               END;
 27   2           ELSE
 28   2               TYPE(1) = 020H;
 29   2           CALL PRINT(.CRLF,2);
              /*                                                   */
              /*                  LOT NUMBER                       */
              /*                                                   */
 30   2           CALL PRINT(.L(0),08H);
 31   2           CALL FETCH(.LOT(0),1);
 32   2       LAST: CALL FETCH(.LOT(1),1);
 33   2           IF (LOT(1) <> 0DH) THEN
 34   2               DO;
 35   3               LOT(0) = LOT(1);
```

```
 36   3              LOT(1) = LOT(2);
 37   3              GO TO LAST;
 38   3              END;
 39   2          LOT(2) = 020H;
 40   2          CALL PRINT(.CRLF,2);
            /*                                        */
            /*                STRESS LEVEL            */
            /*                                        */
 41   2          CALL PRINT(.S(0),0FH);
 42   2          CALL FETCH(.STRESS(0),1);
 43   2     THRU:    CALL FETCH(.STRESS(1),1);
 44   2          IF STRESS(1) <> 0DH THEN
 45   2              DO;
 46   3              STRESS(0) = STRESS(1);
 47   3              GO TO THRU;
 48   3              END;
 49   2          STRESS(1) = 020H;
 50   2          CALL PRINT(.CRLF,2);
            /*                                        */
            /*             TABLE HEADING              */
            /*                                        */
 51   2          CALL PRINT(.CRLF,2);
 52   2          CALL PRINT(.HEAD,71D);
 53   2          CALL PRINT(.CRLF,2);
 54   2          END ID;
 55   1     END CELL$ID$MODULE;
```

```
MODULE INFORMATION:

     CODE AREA SIZE     = 0130H    304D
     VARIABLE AREA SIZE = 0000H      0D
     MAXIMUM STACK SIZE = 0002H      2D
     77 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE STEPTHROUGHMODULE
OBJECT MODULE PLACED IN :F1:STEP.OBJ
COMPILER INVOKED BY:  PLM80 :F1:STEP.SRC


                /************************************************/
                /*                                              */
                /*           STEP THROUGH MODULE                */
                /*                                              */
                /************************************************/
  1             STEP$THROUGH$MODULE:
                DO;
  2   1             DECLARE POINT(255) STRUCTURE(
                        VOLTAGE(25D) ADDRESS,
                        CURRENT(25D) ADDRESS) EXTERNAL;
  3   1             DECLARE SUM ADDRESS EXTERNAL;
  4   1             DECLARE INCREMENT BYTE EXTERNAL;
  5   1             DECLARE (LT,MODE,H) BYTE EXTERNAL;
  6   1             DECLARE N BYTE EXTERNAL;
  7   1             DECLARE MEAS BYTE EXTERNAL;
  8   1             DECLARE NSAVE(25D) BYTE EXTERNAL;
  9   1             DECLARE MEASUREMENT(25D) STRUCTURE(
                        POWER(4) BYTE) EXTERNAL;
                /*                                              */
 10   1         DELAY:  PROCEDURE EXTERNAL;
 11   2             END DELAY;
                /*                                              */
 12   1         PMAX:   PROCEDURE EXTERNAL;
 13   2             END PMAX;
                /*                                              */
 14   1         TBOUT:
                    PROCEDURE (VALUE) EXTERNAL;
 15   2             DECLARE VALUE ADDRESS;
 16   2             END TBOUT;
                /*                                              */
 17   1         COMPUTE$CURRENT:
                PROCEDURE EXTERNAL;
 18   2         END COMPUTE$CURRENT;
                /*                                              */
 19   1         STEP$THROUGH:
                PROCEDURE PUBLIC;
 20   2             DECLARE VOLTAGE$POINTER ADDRESS;
 21   2             DECLARE VLTGE BASED VOLTAGE$POINTER ADDRESS;
 22   2             DECLARE START$CONV BYTE AT(0FFF0H);
 23   2             DECLARE STATUS$POINTER ADDRESS;
 24   2             DECLARE STATUS BASED STATUS$POINTER BYTE;
 25   2             DECLARE POS BYTE;
 26   2             DECLARE SETUP BYTE AT(0FFF0H);
 27   2             DECLARE INSEL BYTE AT(0FFF1H);
 28   2             DECLARE MUXADR BYTE AT(0FFF4H);
 29   2             DECLARE COUNTER BYTE;
 30   2             DECLARE PREVIOUS ADDRESS;
                *                                              *
 31   2             MEASUREMENT(MEAS).POWER(0),MEASUREMENT(MEAS).POWER(1),
                    MEASUREMENT(MEAS).POWER(2),MEASUREMENT(MEAS).POWER(3)=0;
```

```
32   2          PREVIOUS=0;
33   2          CALL TBOUT(PREVIOUS);
              /*                                                    */
34   2          MODE = 00H;
35   2          LT = 00H;
36   2          H = 00H;
37   2          SUM = 00H;
38   2          SETUP = 06H;
39   2          GNSEL = 02H;
40   2          VOLTAGE#POINTER = 0FFFDH;
41   2          STATUS#POINTER = 0FFFCH;
42   2          INCREMENT = 03H;
43   2          N=0;
44   2          POS = 0;
45   2          DO WHILE POS = 0;
46   3          CALL COMPUTE#CURRENT;
47   3          CALL DELAY;
48   3          MUXADR = 00H;
49   3       READ:   START#CONV = 0FFH;
50   3                WAIT:
                      DO WHILE(STATUS AND 30H) = 0;
51   4                END WAIT;
52   3          POINT(N).VOLTAGE(MEAS) = VLTGE AND 07FFH;
53   3          POS = SHR(VLTGE AND 800H,11D);
54   3          IF ((PREVIOUS AND 0FFEH)<>(POINT(N).VOLTAGE(MEAS) AND 0FFEH)) THEN
55   3                DO;
56   4                PREVIOUS = POINT(N).VOLTAGE(MEAS);
57   4                IF COUNTER > 10H THEN
58   4                    GO TO QUIT;
                      ELSE
59   4                    COUNTER = COUNTER + 1;
60   4                GO TO READ;
61   4                END;
62   3       QUIT    COUNTER = 00H;
63   3           IF MODE <> 02 THEN
64   3           CALL PMAX;
65   3           N = N + 1;
66   3           PREVIOUS = 00H;
67   3       END;
68   2           NSAVE(MEAS)=N;
69   2           CALL TBOUT(PREVIOUS);
70   2       END STEP#THROUGH;
71   1       END STEP#THROUGH#MODULE;
```

ORIGINAL PAGE IS OF POOR QUALITY

```
MODULE INFORMATION

     CODE AREA SIZE     = 0176H    374D
     VARIABLE AREA SIZE = 000[illegible]H     [illegible]D
     MAXIMUM STACK SIZE = 0028H     40D
     [illegible] LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE GETCELLNUMMODULE
OBJECT MODULE PLACED IN :F1:GCELL.OBJ
COMPILER INVOKED BY:  PLM80 :F1:GCELL.SRC



  1             GET$CELL$NUM$MODULE:
                DO;
  2   1             PRINT:PROCEDURE(LOCATION,NUMCHARS)EXTERNAL;
  3   2                 DECLARE LOCATION ADDRESS;
  4   2                 DECLARE NUMCHARS BYTE;
  5   2             END PRINT;
                /*                                                    */
  6   1             FETCH:PROCEDURE(LOCATION,NUM$CHARS)EXTERNAL;
  7   2                 DECLARE LOCATION ADDRESS;
  8   2                 DECLARE NUM$CHARS BYTE;
  9   2             END FETCH;
                /*                                                  */
 10   1             DECLARE CRLF(2) BYTE EXTERNAL;
 11   1             DECLARE CELL(5) BYTE EXTERNAL;
 12   1             DECLARE I BYTE;
 13   1             DECLARE X(*) BYTE DATA('CELL NUMBER = ');
 14   1             DECLARE T(*) BYTE DATA ('CELL TEMP = ');
 15   1             DECLARE TEMP(5) BYTE EXTERNAL;
 16   1         GET$CELL$NUM:
                    PROCEDURE PUBLIC;
 17   2             CELL(0),CELL(1),CELL(2),CELL(3),CELL(4) = 020H;
 18   2             I = 0;
 19   2             CALL PRINT(.X(0),0EH);
 20   2             CALL FETCH(.CELL(0),2);
 21   2             IF (CELL(1) = 0DH) THEN
 22   2                 DO;
 23   3                 CELL(2) = CELL(0);
 24   3                 CELL(0),CELL(1) = 030H;
 25   3                 GO TO FET;
 26   3                 END;
 27   2             CALL FETCH(.CELL(2),1);
 28   2             IF (CELL(2) = 0DH) THEN
 29   2                 DO;
 30   3                 CELL(2) = CELL(1);
 31   3                 CELL(1) = CELL(0);
 32   3                 CELL(0) = 030H;
 33   3                 GO TO FET;
 34   3                 END;
 35   2         AIR:    CALL FETCH(.CELL(3),1);
 36   2             IF (CELL(3) [illegible] 0DH) THEN
 37   2                 DO;
 38   3                 CELL(0) = CELL(1);
 39   3                 CELL(1) = CELL(2);
 40   3                 CELL(2) = CELL(3);
 41   3                 GO TO AIR;
 42   3                 END;
                  /*                                              */
                  /*                  [illegible]                 */
                  /*                                              */
 43   2         FET:    [illegible]
 44   2             CALL PRINT(.CRLF(0) [illegible]
```

```
  45   2           CALL PRINT(.T(0),0CH);
  46   2           CALL FETCH(.TEMP(0),5);
  47   2           CALL PRINT(.CRLF,2);
  48   2           END;
  49   1       END GETACFL$NUM$MODULE;
```

MODULE INFORMATION

```
      CODE AREA SIZE     = 00C3H    200D
      VARIABLE AREA SIZE = 0001H      1D
      MAXIMUM STACK SIZE = 0002H      2D
      36 LINES READ
      0 PROGRAM ERROR(S)
```

END OF PL/M-80 COMPILATION

PL/M-80 COMPILER

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE COMPUTECURRENTMODULE
OBJECT MODULE PLACED IN :F1:CCRNT.OBJ
COMPILER INVOKED BY: PLM80 :F1:CCRNT.SRC

```
              /**************************************************/
              /*                                                */
              /*             COMPUTE CURRENT MODULE             */
              /*                                                */
              /**************************************************/
  1           COMPUTE$CURRENT$MODULE:
              DO;
              /*                                                */
  2   1       TBOUT:  PROCEDURE (VALUE) EXTERNAL;
  3   2           DECLARE VALUE ADDRESS;
  4   2           END TBOUT;
              /*                                                */
  5   1           DECLARE POINT(255) STRUCTURE(
                      VOLTAGE(250) ADDRESS,
                      CURRENT(250) ADDRESS) EXTERNAL;
  6   1           DECLARE MEAS BYTE EXTERNAL;
  7   1           DECLARE (LT,MODE,H) BYTE EXTERNAL;
  8   1           DECLARE INCREMENT BYTE EXTERNAL;
  9   1           DECLARE SUM ADDRESS EXTERNAL;
 10   1           DECLARE N BYTE EXTERNAL;
 11   1       DELAY:  PROCEDURE PUBLIC;
 12   2           DO;
 13   3           DECLARE P BYTE;
 14   3           DO WHILE (P<0FFH);
 15   4           P = P + 1;
 16   4           END;
 17   3           END;
 18   2           END DELAY;
 19   1       COMPUTE$CURRENT:
              PROCEDURE PUBLIC;
 20   2           DECLARE (MIN,MAX) BYTE;
 21   2           DECLARE SAVE ADDRESS;
 22   2           DECLARE DELTA BYTE;
              /*                                                */
 23   2       DO CASE MODE;
 24   3       MODE0:  DO;
 25   4           IF LT = 0 THEN
 26   4               DO;
 27   5               POINT(N).CURRENT(MEAS) = SUM;
 28   5               CALL TBOUT(SUM);
 29   5               SUM = SUM + 10H;
 30   5               END;
              ELSE
 31   4               DO;
 32   5               MODE = 01H;
 33   5               SUM = SUM -10H;
 34   5               SAVE = SUM;
 35   5               LT = 0;
 36   5               GO TO MODE1;
 37   5               END;
 38   3           END MODE0;
```

```
 39   3        MODE1   DO;
 40   4            IF LT = 0 THEN
 41   4                DO;
 42   5                SUM = SUM -01H;
 43   5                POINT(N).CURRENT(MEAS) = SUM;
 44   5                CALL TBOUT(SUM);
 45   5                END;
             ELSE
 46   4                DO;
 47   5                MODE = 02H;
 48   5                SUM = SAVE;
 49   5                GO TO MODE2;
 50   5                END;
 51   4            END MODE1;
 52   3       MODE2:  DO;
 53   4       MIN = 02H;
 54   4       MAX = 20H;
 55   4            IF M<= 1 THEN
 56   4            DO CASE M;
          /*                                                   */
 57   5                DO;
 58   6                SUM = SUM + 01;
 59   6                POINT(N).CURRENT(MEAS) = SUM;
 60   6                CALL TBOUT(SUM);
 61   6                END;
          /*                                                   */
 62   5                DO;
 63   6                SUM = SUM + 04H;
 64   6                POINT(N).CURRENT(MEAS) = SUM;
 65   6                CALL TBOUT(SUM);
 66   6                END;
 67   5           END;    /* END CASE    */
          ELSE
 68   4       DO;
 69   5       DELTA = POINT(N-2).VOLTAGE(MEAS) - POINT(N-1).VOLTAGE(MEAS);
 70   5           IF (DELTA > MIN) AND (DELTA < MAX) THEN
 71   5           DO;
 72   6           SUM = SUM + INCREMENT;
 73   6           POINT(N).CURRENT(MEAS) = SUM;
 74   6           CALL TBOUT(SUM);
 75   6           END;
             ELSE
 76   5               DO;
 77   6               IF (DELTA < MIN) THEN
 78   6               INCREMENT = INCREMENT + 01H;
                     ELSE
 79   6               DO;
 80   7               IF INCREMENT > 1 THEN
 81   7               INCREMENT = INCREMENT - 01H;
 82   7               END;
 83   6           SUM = SUM + INCREMENT;
 84   6           POINT(N).CURRENT(MEAS) = SUM;
 85   6           CALL TBOUT(SUM);
 86   6               END;
 87   5           END;
          /*                                                   */
 88   4           N = N + 01H;
```

ORIGINAL PAGE IS OF POOR QUALITY

```
 49   4      END MODE2;
 50   3      END;
 51   2      END COMPUTE$CURRENT;
 52   1      END COMPUTE$CURRENT$MODULE;
```

```
MODULE INFORMATION

     CODE AREA SIZE     = 025DH    605D
     VARIABLE AREA SIZE = 0006H      6D
     MAXIMUM STACK SIZE = 0006H      6D
     112 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
PL/M-80 COMPILER

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE SCOPEMODULE
OBJECT MODULE PLACED IN :F1:PSCOPE.OBJ
COMPILER INVOKED BY:  PLM80 :F1:PSCOPE.SRC



                   /*****************************************************
                   /*                                                   */
                   /*           OSCILLOSCOPE DISPLAY MODULE             */
                   /*                                                   */
                   /*****************************************************/
  1                SCOPE$MODULE:
                   DO;
                   /*                                                   */
  2   1            CI: PROCEDURE BYTE EXTERNAL;
  3   2                END CI;
  4   1            PLOT:   PROCEDURE(AXIS) EXTERNAL;
  5   2                DECLARE AXIS BYTE;
  6   2                END PLOT;
                   /*                                                   */
  7   1                DECLARE POINT(250) STRUCTURE(
                           VOLTAGE(250) ADDRESS,
                           CURRENT(250) ADDRESS) EXTERNAL;
  8   1                DECLARE NSAVE(250) BYTE EXTERNAL;
  9   1                DECLARE MEAS BYTE EXTERNAL;
 10   1            SCOPE:
                   PROCEDURE PUBLIC;
 11   2                DECLARE I BYTE;
 12   2                DECLARE PLT BYTE;
 13   2                DECLARE (DAC2LO,DAC1LO) ADDRESS AT(0FFF0H);
 14   2            DO WHILE ((INPUT(0ECH) AND 02H) = 0);
 15   3            SWEEP:
                       DO I =0 TO (NSAVE(MEAS)-2);
 16   4                DAC1LO = POINT(I).VOLTAGE(MEAS);
 17   4                DAC2LO = POINT(I).CURRENT(MEAS);
 18   4                END SWEEP;
 19   3            END;
 20   2                PLT = CI;
 21   2                IF PLT = 'P' THEN
 22   2                    CALL PLOT(MEAS);
 23   2            END SCOPE;
 24   1            END SCOPEMODULE;



MODULE INFORMATION

     CODE AREA SIZE     = [illegible]
     VARIABLE AREA SIZE = [illegible]
     MAXIMUM STACK SIZE = [illegible]
     [illegible] LINES READ
     0 PROGRAM ERRORS

END OF PL/M-80 COMPILATION
```

PL/M-80 COMPILER

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE GETPARAMETERSMODULE
OBJECT MODULE PLACED IN :F1:GPAR.OBJ
COMPILER INVOKED BY: PLM80 :F1:GPAR.SRC

```
  1           GET$PARAMETERS$MODULE:
              DO;
  2   1       DECLARE POINT(250) STRUCTURE(
                  VOLTAGE(250) ADDRESS,
                  CURRENT(250) ADDRESS) EXTERNAL;
  3   1           DECLARE LT BYTE EXTERNAL;
  4   1           DECLARE CELL(5) BYTE EXTERNAL;
  5   1           DECLARE TAPE$ON BYTE EXTERNAL;
  6   1           DECLARE TAPE$OFF BYTE EXTERNAL;
  7   1           DECLARE STRESS(2) BYTE EXTERNAL;
  8   1           DECLARE LOT(3) BYTE EXTERNAL;
  9   1           DECLARE TYPE(2) BYTE EXTERNAL;
 10   1           DECLARE PERIOD BYTE EXTERNAL;
 11   1           DECLARE TEMP(5) BYTE EXTERNAL;
 12   1           DECLARE N BYTE EXTERNAL;
 13   1           DECLARE CRLF(2) BYTE EXTERNAL;
 14   1           DECLARE MEAS BYTE EXTERNAL;
 15   1           DECLARE NSAVE(250) BYTE EXTERNAL;
 16   1           DECLARE MEASUREMENT(250) STRUCTURE(
                      POWER(4) BYTE) EXTERNAL;
 17   1           DECLARE MP(250) BYTE EXTERNAL;
 18   1           DECLARE ISC(4) BYTE;
 19   1           DECLARE VOC(4) BYTE;
 20   1           DECLARE KV(4) BYTE DATA (02H,14H,20H,3AH);
 21   1           DECLARE KI(4) BYTE DATA (00H,0AH,0A0H,3AH);
 22   1           DECLARE CONTROL STRUCTURE(
                      SIGN BYTE,
                      SCALE ADDRESS,
                      SLENGTH BYTE,
                      STRING$PTR ADDRESS),STRING(8) BYTE EXTERNAL;
 23   1           DECLARE FPR(18) BYTE;
 24   1           DECLARE (MULTIPLICAND,MULTIPLIER)(4) BYTE;
              /*                                                        */
 25   1       PRINT PROCEDURE(LOCATION,NUM$CHARS) EXTERNAL;
 26   2           DECLARE LOCATION ADDRESS;
 27   2           DECLARE NUM$CHARS BYTE;
 28   2       END PRINT;
               *                                                        *
 29   1       NUMOUT PROCEDURE(VALUE,BASE,LC,BUFADR,WIDTH) EXTERNAL;
 30   2           DECLARE (VALUE,BUFADR) ADDRESS;
 31   2           DECLARE (BASE,LC,WIDTH) BYTE;
 32   2       END NUMOUT;
               *                                                        *
 33   1       FPREAD PROCEDURE(FA,[illegible]) EXTERNAL;
 34   2           DECLARE (FA,[illegible]) ADDRESS;
 35   2       END FPREAD;
               *                                                        *
 36   1       FSET   PROCEDURE(FA,[illegible]) EXTERNAL;
 37   2           DECLARE(FA,[illegible]) ADDRESS;
 38   2           END FSET;
 39   1       FSTAT  PROCEDURE (FA) BYTE EXTERNAL;
```

```
 40   2          DECLARE FA ADDRESS;
 41   2          END FSTAT;
 42   1      FMUL:   PROCEDURE (FA,OA) EXTERNAL;
 43   2          DECLARE (FA,OA) ADDRESS;
 44   2          END FMUL;
 45   1      FLOAD:  PROCEDURE (FA,OA) EXTERNAL;
 46   2          DECLARE (FA,OA) ADDRESS;
 47   2          END FLOAD;
 48   1      FSTOR:  PROCEDURE (FA,OA)EXTERNAL;
 49   2          DECLARE (FA,OA) ADDRESS;
 50   2          END FSTOR;
             /*      FCMPR - FP # COMPARISON         */
 51   1      FCMPR:  PROCEDURE (FA,OA) BYTE EXTERNAL;
 52   2          DECLARE (FA,OA) ADDRESS;
 53   2          END FCMPR;
             /*      FLTDS - INTEGER -> FP CONVERSION      */
 54   1      FLTDS:  PROCEDURE (FA,OA) EXTERNAL;
 55   2          DECLARE (FA,OA) ADDRESS;
 56   2          END FLTDS;
 57   1      X12$TO$32BIT:
             PROCEDURE(INPUT$ADDRESS,STORE$ADDRESS)EXTERNAL;
 58   2          DECLARE (INPUT$ADDRESS,STORE$ADDRESS) ADDRESS;
 59   2      END X12$TO$32BIT;
 60   1      DEC$OUT:PROCEDURE;
 61   2          DECLARE POS(2) BYTE DATA ('E+');
 62   2          DECLARE NEG(2) BYTE DATA('E-');
 63   2          DECLARE BUFFER(4) BYTE;
             /*                                                   */
 64   2          CALL PRINT(CONTROL.STRING$PTR,CONTROL.SLENGTH);
 65   2          CALL PRINT(.PERIOD,1);
 66   2          IF ((CONTROL.SCALE AND 8000H)=0) THEN
 67   2              CALL PRINT(.POS,2);
                 ELSE
 68   2              DO;
 69   3              CALL PRINT(.NEG,2);
 70   3              CONTROL.SCALE = ((NOT CONTROL.SCALE) + 1);
 71   3              END;
 72   2          CALL NUMOUT(CONTROL.SCALE,10,'0',.BUFFER,2);
 73   2          CALL PRINT(.BUFFER,2);
 74   2      END DEC$OUT;
 75   1      PMAX:
             PROCEDURE PUBLIC;
 76   2          DECLARE (I,STATUS) BYTE;
 77   2          I = 4;
 78   2          CALL FSET(.FPR,0,0);
 79   2          CALL X12$TO$32BIT(.POINT(I).VOLTAGE$MEAS,.MULTIPLICAND);
 80   2          CALL X12$TO$32BIT(.POINT(I).CURRENT$MEAS,.MULTIPLIER);
 81   2          CALL FLTDS(.FPR,.MULTIPLIER);
 82   2          CALL FSTOR(.FPR,.MULTIPLIER);
 83   2          CALL FLTDS(.FPR,.MULTIPLICAND);
 84   2          CALL FMUL(.FPR,.MULTIPLIER);
 85   2          STATUS = FCMPR(.FPR,.MEASUREMENT.MEAS$POWER);
 86   2          IF (STATUS AND 10H) = 10H THEN
 87   2              DO;
 88   3              CALL FSTOR(.FPR,.MEASUREMENT.MEAS$POWER);
 89   3              MP$MEAS = I;
 90   3              END;
```

```
144  2        CALL PRINT(.TAPE$OFF,1);
145  2      END POUT;
146  1      END GET$PARAMETERS$MODULE;


MODULE INFORMATION:

     CODE AREA SIZE     = 0330H    829D
     VARIABLE AREA SIZE = 002EH     46D
     MAXIMUM STACK SIZE = 000AH     10D
     168 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
                   ELSE
 91   2                DO;
 92   3                CALL FSTOR(.FPR, .MEASUREMENT(MEAS).POWER(0));
 93   3                LT = 01H;
 94   3                END;
 95   2            END PMAX;
 96   1            POUT: PROCEDURE PUBLIC;
 97   2                DECLARE LEADER(*) BYTE DATA(0,0,0,0,0,0,0,0,0,0,0,
                              0,0,0,0);
 98   2                DECLARE SPACE(3) BYTE DATA('   ');
                   /*                                                   */
 99   2                CONTROL.SLENGTH = 03;
100   2                CONTROL.STRING$PTR = .STRING;
101   2                CALL PRINT(.CRLF,1);
102   2                CALL PRINT(.TAPE$ON,1);
103   2                CALL PRINT(.LEADER,15D);
104   2                CALL PRINT(.TYPE,2);
105   2                CALL PRINT(.LOT,3);
106   2                CALL PRINT(.STRESS,2);
107   2                CALL PRINT(.CELL(0),4);
108   2                CALL X12$TO$32BIT(.POINT(0).VOLTAGE(MEAS), .MULTIPLIER);
109   2                CALL FLTDS(.FPR, .MULTIPLIER);
110   2                CALL FMUL(.FPR, .KV(0));
111   2                CALL FQFR2D(.FPR, .CONTROL);
112   2                CALL DEC$OUT;
113   2                CALL FLOAD(.FPR, .MEASUREMENT(MEAS).POWER(0));
114   2                CALL FMUL(.FPR, .KV(0));
115   2                CALL FSTOR(.FPR, .MEASUREMENT(MEAS).POWER(0));
116   2                CALL PRINT(.SPACE,2);
117   2                CONTROL.SLENGTH = 04;
118   2                CALL X12$TO$32BIT(.POINT(NSAVE(MEAS) - 2).CURRENT(MEAS), .MULTIPLIER);
119   2                CALL FLTDS(.FPR, .MULTIPLIER);
120   2                CALL FMUL(.FPR, .KI(0));
121   2                CALL FQFR2D(.FPR, .CONTROL);
122   2                CALL DEC$OUT;
123   2                CALL PRINT(.SPACE,2);
124   2                CONTROL.SLENGTH = 03;
125   2                CALL FLOAD(.FPR, .MEASUREMENT(MEAS).POWER(0));
126   2                CALL FMUL(.FPR, .KI(0));
127   2                CALL FQFB2D(.FPR, .CONT[illegible]);
128   2                CALL DEC$OUT;
129   2                CALL PRINT(.SPACE,2);
130   2                CALL X12$TO$32BIT(.POINT(MP(MEAS)).VOLTAGE(MEAS), .MULTIPLICAND);
131   2                CALL FLTDS(.FPR, .MULTIPLICAND);
132   2                CALL FMUL(.FPR, .KV(0));
133   2                CALL FQFR2D(.FPR, .CONTROL);
134   2                CALL DEC$OUT;
                   *                                                    *
135   2                CALL PRINT(.SPACE,2);
136   [illegible]      CALL X12$TO$32BIT(.POINT(MP(MEAS)).CURRENT(MEAS), .MULTIPLICAND);
137   [illegible]      CALL FLTDS(.FPR, .MULTIPLICAND);
138   2                CALL FMUL(.FPR, .KI(0));
139   2                CALL FQFR2D(.FPR, .CONTROL);
140   2                CALL DEC$OUT;
141   2                CALL PRINT(.SPACE,[illegible]);
142                    CALL PRINT(.TEMP,[illegible]);
143   2                CALL PRINT(.CRLF[illegible]
```

```
PL/M-80 COMPILER

ISIS-II PL/M-80 [illegible] COMPILATION OF MODULE PLOTMODULE
OBJECT MODULE PLACED IN :F1:PLOT.OBJ
COMPILER INVOKED BY:  PLM80 :F1:PLOT.SRC



                /*************************************************************
                 *                                                           *
                 *                      PLOT ROUTINE                         */
                 *                                                           */
                /************************************************************/
  1                 PLOTMODULE:
                DO;
  2   1             DECLARE POINT (250) STRUCTURE(
                        VOLTAGE(250) ADDRESS,
                        CURRENT(250) ADDRESS) EXTERNAL;
  3   1             DECLARE NSAVE(250) BYTE EXTERNAL;
  4   1             DECLARE (DAC2LO,DAC1LO) ADDRESS AT(0FFF8H);
  5   1             DECLARE MEAS BYTE EXTERNAL;
                /*                                                           */
                /*                                                           */
  6   1         PLOT:   PROCEDURE(AXIS) PUBLIC;
  7   2             DECLARE AXIS BYTE;
  8   2             DECLARE I BYTE;
  9   2             DECLARE (XMAX,YMAX) ADDRESS;
 10   2             DECLARE (XPOSITION,YPOSITION) ADDRESS;
 11   2             DECLARE COUNTER BYTE;
 12   2             DECLARE CV ADDRESS;
 13   2             DECLARE CSCALE BYTE;
                /*                                                           */
 14   2             CSCALE = 1;
 15   2             XMAX = 3620D;
 16   2             YMAX = 2230D;
 17   2             YPOSITION = 0FFFH;
 18   2             DAC1LO = YPOSITION;
 19   2             XPOSITION = 0;
 20   2             DAC2LO = XPOSITION;
 21   2             DO I = 0 TO 255D;
 22   3             CALL TIME(150);
 23   3             END;
 24   2             CALL TIME(255);
 25   2             IF AXIS <> 0 THEN
 26   2                 GO TO NOAXIS;
                 *                                                           *
 27   2           AXIS: DO WHILE XPOSITION < XMAX;
 28   3                 DO I = 1 TO [illegible];
 29   4                 XPOSITION = XPOSITION + 1;
 30   4                 DAC2LO = XPOSITION;
 31   4                 CALL TIME([illegible]);
 32   [illegible]       END;
 33   [illegible]       CALL TIME([illegible]);
 34   [illegible]       DAC1LO = [illegible];
 35   [illegible]       CALL TIME([illegible]);
 36   [illegible]       CALL TIME([illegible]);
 37   [illegible]       DAC1LO = [illegible];
 38   [illegible]       CALL TIME([illegible]);
 39   [illegible]       CALL TIME([illegible]);
```

```
40   3              CALL TIME(255D);
41   3          END;
42   2              DO WHILE XPOSITION <> 0;
43   3              XPOSITION = XPOSITION - 1;
44   3              DAC2LO = XPOSITION;
45   3              CALL TIME(90);
46   3              END;
47   2              YPOSITION = 0;
48   2      YAXIS:  DO WHILE YPOSITION < YMAX;
49   3              DO I = 0 TO 285D;
50   4              YPOSITION = YPOSITION + 1;
51   4              CY = NOT (YPOSITION);
52   4              DAC1LO = CY;
53   4              CALL TIME(500);
54   4              END;
55   3              CALL TIME(255D);
56   3              CALL TIME(255D);
57   3              DAC2LO = 40H;
58   3              CALL TIME(255D);
59   3              CALL TIME(255D);
60   3              DAC2LO = 0;
61   3              CALL TIME(255D);
62   3              CALL TIME(255D);
63   3              CALL TIME(255D);
64   3          END;
65   2      NOAXIS      IF (POINT(NSAVE(MEAS -2) CURRENT(MEAS) > 815D) THEN
66   2                  CSCALE = CSCALE + 1;
67   2              CALL TIME(255D);
68   2              XPOSITION = (POINT(0).CURRENT(MEAS) + 5) / CSCALE
69   2              YPOSITION = (POINT(0).VOLTAGE(MEAS) + 5) / 2;
70   2              CY = NOT (YPOSITION);
71   2              DAC1LO = CY;
72   2              DAC2LO = XPOSITION;
73   2              DO I = 0 TO 260;
74   3              CALL TIME(255);
75   3              END;
76   2          DO I = 0 TO (NSAVE(MEAS)-2);
77   3              XPOSITION = (POINT(I).CURRENT(MEAS) + 5) / CSCALE
78   3              YPOSITION = (POINT(I).VOLTAGE(MEAS) + 5) / 2;
79   3              CY = NOT (YPOSITION);
80   3              DAC1LO = CY;
81   3              DAC2LO = XPOSITION;
82   3              COUNTER = 0;
83   3              DO WHILE COUNTER < 100;
84   4              COUNTER = COUNTER + 1;
85   4              CALL TIME(255D);
86   4              END;
87   3          END;
88   2              DO WHILE [illegible]
89   3              [illegible]
90   3              [illegible] NOT [illegible]
91   3              DAC1LO [illegible]
92   3              CALL TIME [illegible]
93   3              END;
94   2              [illegible] WHILE [illegible]
95   3              [illegible]
96   3              DAC2LO [illegible]
```

PL/M-80 COMPILER

```
 97   3              CALL TIME(50);
 98   2              END;
 99   2          END PLOT;
100   1      END PLOTMODULE;
```

MODULE INFORMATION:

```
     CODE AREA SIZE     = 02ECH    [illegible]D
     VARIABLE AREA SIZE = 000EH     14D
     MAXIMUM STACK SIZE = 0006H      6D
     112 LINES READ
     0 PROGRAM ERROR(S)
```

END OF PL/M-80 COMPILATION

ORIGINAL PAGE IS
OF POOR QUALITY

C-3

```
ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE TIMEEFFECTSMODULE
OBJECT MODULE PLACED IN :F1:PD.OBJ
COMPILER INVOKED BY:  PLM80 :F1:PD.SRC


  1          TIME$EFFECTS$MODULE:
             DO;
  2   1      ID: PROCEDURE EXTERNAL;
  3   2          END ID;
  4   1      GET$CELL$NUM: PROCEDURE EXTERNAL;
  5   2          END GET$CELL$NUM;
  6   1      STEP$THROUGH: PROCEDURE EXTERNAL;
  7   2          END STEP$THROUGH;
  8   1      SCOPE:   PROCEDURE EXTERNAL;
  9   2          END SCOPE;
 10   1      POUT:    PROCEDURE EXTERNAL;
 11   2          END POUT;
 12   1      PLOT:    PROCEDURE (MEASUREMENT) EXTERNAL;
 13   2          DECLARE MEASUREMENT BYTE;
 14   2          END PLOT;
 15   1      FETCH:PROCEDURE(LOCATION,NUM$CHARS)EXTERNAL;
 16   2          DECLARE LOCATION ADDRESS;
 17   2          DECLARE NUM$CHARS BYTE;
 18   2          END FETCH;
 19   1      PRINT:  PROCEDURE(LOCATION,NUM$CHARS) EXTERNAL;
 20   2          DECLARE LOCATION ADDRESS;
 21   2          DECLARE NUM$CHARS BYTE;
 22   2          END PRINT;
 23   1          DECLARE CRLF(2) BYTE EXTERNAL;
 24   1          DECLARE COMMAND BYTE EXTERNAL;
 25   1          DECLARE MEAS BYTE EXTERNAL;
 26   1      TYE:     PROCEDURE PUBLIC;
 27   2          DECLARE NUM$MEAS(2) BYTE;
 28   2          DECLARE I BYTE;
 29   2          DECLARE TIMES BYTE;
 30   2          DECLARE Q(*) BYTE DATA('# OF MEAS - [illegible]');
 31   2      START:  CALL PRINT(.Q(0),12D);
 32   2          CALL FETCH(.NUM$MEAS,2);
 33   2      LASTC:  IF NUM$MEAS(1) [illegible] 0DH THEN
 34   2              DO;
 35   3              CALL PRINT(.CRLF,2);
 36   3              IF NUM$MEAS(0) [illegible] '0' THEN
 37   3                  GO TO START;
 38   3              IF NUM$MEAS(0) [illegible] '9' THEN
 39   3                  DO;
 40   4                  NUM$MEAS(0) = NUM$MEAS(0) - 30H;
 41   4                  GO TO CONT;
 42   4                  END;
 43   3              IF NUM$MEAS(0) [illegible] 'A' THEN
 44   3                  GO TO START;
 45   3              IF NUM$MEAS(0) [illegible] 'F' THEN
 46   3                  NUM$MEAS(0) = NUM$MEAS(0) - [illegible];
 47   3              ELSE GO TO START;
 48   3              END;
           ELSE
 49   2   [illegible]
```

```
 50   3                  NUM$MEAS(0) = NUM$MEAS(1);
 51   3                  CALL FETCH(NUM$MEAS(1));
 52   3                  GO TO LASTC;
 53   3                  END;
 54   2        CONT      CALL ID;
 55   2             CALL GETACELL$NUM;
 56   2             MEAS=0;
 57   2             OUTPUT(0E7H) = 50H;
 58   2             OUTPUT(0E4H) = 20H;
 59   2             DO I = 0 TO 200;
 60   3                  CALL TIME(255D);
 61   3             END;
 62   2             DO WHILE MEAS <= NUM$MEAS(0);
 63   3                  CALL STEP$THROUGH;
 64   3                  CALL SCOPE;
 65   3                  MEAS = MEAS + 1;
 66   3             END;
 67   2             OUTPUT(0E4H) = 0FFH;
 68   2             MEAS = 0;
 69   2             DO WHILE (MEAS <= NUM$MEAS(0));
 70   3                  CALL POUT;
 71   3                  MEAS = MEAS + 1;
 72   3             END;
 73   2             MEAS = 0;
 74   2             DO WHILE(MEAS <= NUM$MEAS(0));
 75   3                  CALL PLOT(MEAS);
 76   3                  MEAS = MEAS + 1;
 77   3             END;
 78   2             END TYE;
 79   1             END TIME$EFFECTS$MODULE;
```

```
MODULE INFORMATION

     CODE AREA SIZE     = 0101H    257D
     VARIABLE AREA SIZE = 0004H      4D
     MAXIMUM STACK SIZE = 0002H      2D
     81 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE NUMOUTMODULE
OBJECT MODULE PLACED IN :F1:NMOUT.OBJ
COMPILER INVOKED BY:  PLM80 :F1:NMOUT.SRC



   1          NUMOUT$MODULE:
              DO;
              /****************************************************************/
              /*                          NUMOUT                               */
              /****************************************************************/
              /*                                                               */
              /*  NUMOUT CONVERTS A NUMBER FROM INTERNAL (BINARY) FORM TO
                  AN ASCII STRING SUITABLE FOR PRINTING


                  PARAMETERS
                  ----------
              VALUE       THE NUMBER WHOSE PRINTED REPRESENTATION IS DESIRED
              BASE        AN INTEGER BETWEEN 2 AND 16 SPECIFING THE BASE
                    VALUE  IS TO BE INTERPRETED
              LC      LEADING CHARACTER (EITHER A ' ', '0', OR '0')
              BUFADR      BUFFER ADDRESS.
              WIDTH       THE NUMBER OF CHARACTER POSITIONS DESIRED
                    IN THE PRINTED REPRESENTATION                                 */


   2   1      NUMOUT:
              PROCEDURE(VALUE,BASE,LC,BUFADR,WIDTH) PUBLIC;
   3   2         DECLARE (VALUE,BUFADR) ADDRESS;
   4   2         DECLARE (BASE,LC,WIDTH,I) BYTE;
   5   2         DECLARE (CHARS BASED BUFADR)(1) BYTE;
   6   2         DECLARE DIGITS(*) BYTE DATA('0123456789ABCDEF');
   7   2         DO I = 1 TO WIDTH;
   8   3            CHARS(WIDTH - I) = DIGITS(VALUE MOD BASE);
   9   3            VALUE = VALUE/BASE;
  10   3         END;
  11   2         I = 0;
  12   2         DO WHILE CHARS(I) = '0' AND I < WIDTH - 1;
  13   3            CHARS(I) = LC;
  14   3            I = I + 1;
  15   3         END;
  16   2      END NUMOUT;
  17   1      END NUMOUT$MODULE;


MODULE INFORMATION

     CODE AREA SIZE     = [illegible]H    [illegible]D
     VARIABLE AREA SIZE = [illegible]H    [illegible]D
     MAXIMUM STACK SIZE = [illegible]H    [illegible]D
     [illegible] LINES READ
     [illegible] PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

PL/M-80 COMPILER

ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE TBOUTMODULE
OBJECT MODULE PLACED IN :F1:TBOUT.OBJ
COMPILER INVOKED BY: PLM80 :F1:TBOUT.[illegible]

```
                /*************************************************/
                /*                                               */
                /*  TWELVE BITS OUT FOR PROG. POWER SUPPLY  */
                /*                                               */
                /*************************************************/
                /*                                               */
  1             TBOUT$MODULE:
                DO;
  2   1         TBOUT:  PROCEDURE (VALUE) PUBLIC;
  3   2             DECLARE VALUE ADDRESS;
  4   2             DECLARE BUF BYTE;
                /*                                               */
                /*************************************************/
                /*                                               */
                /*  CONTROL WORD    PORT A -> OUTPUT          */
                /*          PORT B -> OUTPUT         */
                /*          PORT C -> INPUT          */
  5   2             OUTPUT(0EBH) = 089H;
                /**                                              */
                /*************************************************/
  6   2             BUF = VALUE;
                /*                                               */
                /*      OUTPUT LOWER 8-BITS             */
  7   2             OUTPUT(0E8H) = BUF;
                /*                                               */
  8   2             BUF = SHR(VALUE,8);
                /*                                               */
                /*      OUTPUT UPPER 4-BITS             */
  9   2             OUTPUT(0E9H) = BUF;
                /*                                               */
                /*      STROBE GOES LOW                 */
 10   2             OUTPUT(0EAH) = 02H;
                /*                                               */
                /*      STROBE GOES HIGH                */
 11   2             OUTPUT(0EAH) = 00H;
                /*                                               */
 12   2             DO WHILE (INPUT(0EAH) AND 0FEH) = 01 ;
 13   3             END;
 14   2         END;
 15   1         END TBOUT$MODULE;
```

MODULE INFORMATION:

```
     CODE AREA SIZE     = [illegible]  [illegible]
     VARIABLE AREA SIZE = [illegible]  [illegible]
     MAXIMUM STACK SIZE = [illegible]  [illegible]
     [illegible] LINES READ
     [illegible] PROGRAM ERROR(S)
```

PL/M-80 COMPILER

```
ISIS-II PL/M-80 V3.1 COMPILATION OF MODULE IOMODULE
OBJECT MODULE PLACED IN :F1:IO.OBJ
COMPILER INVOKED BY:  PLM80 :F1:IO.SRC


  1          IO$MODULE:
             DO;
             /*                                                    */
  2   1      CO: PROCEDURE(CHAR) EXTERNAL;
  3   2          DECLARE CHAR BYTE;
  4   2      END CO;
             /*                                                    */
  5   1      CI: PROCEDURE BYTE EXTERNAL;
  6   2      END CI;
             /*                                                    */
  7   1      PRINT: PROCEDURE(LOCATION,NUM$CHARS) PUBLIC;
  8   2          DECLARE LOCATION ADDRESS;
  9   2          DECLARE NUM$CHARS BYTE;
 10   2          DECLARE CHAR BASED LOCATION BYTE;
 11   2              DO WHILE(NUM$CHARS > 0);
 12   3                  CALL CO(CHAR);
 13   3                  NUM$CHARS = NUM$CHARS - 1;
 14   3                  LOCATION = LOCATION + 1;
 15   3              END;
 16   2          END PRINT;
 17   1      FETCH: PROCEDURE(LOCATION,NUM$CHARS) PUBLIC;
 18   2          DECLARE LOCATION ADDRESS;
 19   2          DECLARE NUM$CHARS BYTE;
 20   2          DECLARE CHAR BASED LOCATION BYTE;
 21   2              DO WHILE(NUM$CHARS > 0);
 22   3                  CHAR = CI AND 07FH;
 23   3                  CALL CO(CHAR);
 24   3                  NUM$CHARS = NUM$CHARS - 1;
 25   3                  LOCATION = LOCATION + 1;
 26   3              END;
 27   2          END FETCH;
 28   1      END IO$MODULE;



MODULE INFORMATION

     CODE AREA SIZE     = 00C7H    [illegible]D
     VARIABLE AREA SIZE = 0006H      6D
     MAXIMUM STACK SIZE = 000[illegible]H     [illegible]D
     28 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

APPENDIX C. ANTIREFLECTIVE COATING AND COLOR MEASUREMENT THEORY

PRECEDING PAGE BLANK NOT FILMED

APPENDIX C. ANTIREFLECTIVE COATING AND COLOR MEASUREMENT THEORY

Antireflective Coating Theory

Before sunlight reaches the energy producing area of a terrestrial solar cell, it must pass through several optical media. Each boundary between media with different indices of refraction causes the light to experience reflections which are non-uniform over all angles of incidence and wavelengths. For unencapsulated cells, the media with the most significant effects are:

Atmosphere

Anit-reflective films on the cell's surface

Solar cell material

Sunlight which strikes the cell's front surface is partly reflected and partly absorbed as shown in Figure C.1. An anti-reflective film is a thin coating, applied by some manufacturers, to enable the cell to capture a larger percentage of the available energy. Ideally, the anti-reflective (AR) coating should reduce the reflection loss to a maximum of three percent, and result in a greater electrical output by the cell.

A typical AR coating is optimized for the wave length of the sun's peak spectral response (C.1). A thin film of magnesium flouride (M_9F_2) or similar material is vaccum deposited on the surface. This film should have a thickness which is an odd multiple of $\lambda_{ar}/4$ where λ_{ar} is the wavelength in the AR coating. The effectiveness of the coaing is determined by the angle of incidence at which sunlight strikes the cell, the indexes of refraction of the air and the film, and the amount of the spectrum over which reflection is to be minimized. For light passing from air into the AR coating at a right angle, the reflectance, ρ , is given by:

$$\rho = \frac{\left(n_2 - n_1\right)^2}{\left(n_2 + n_1\right)^2}$$

PRECEDING PAGE BLANK NOT FILMED

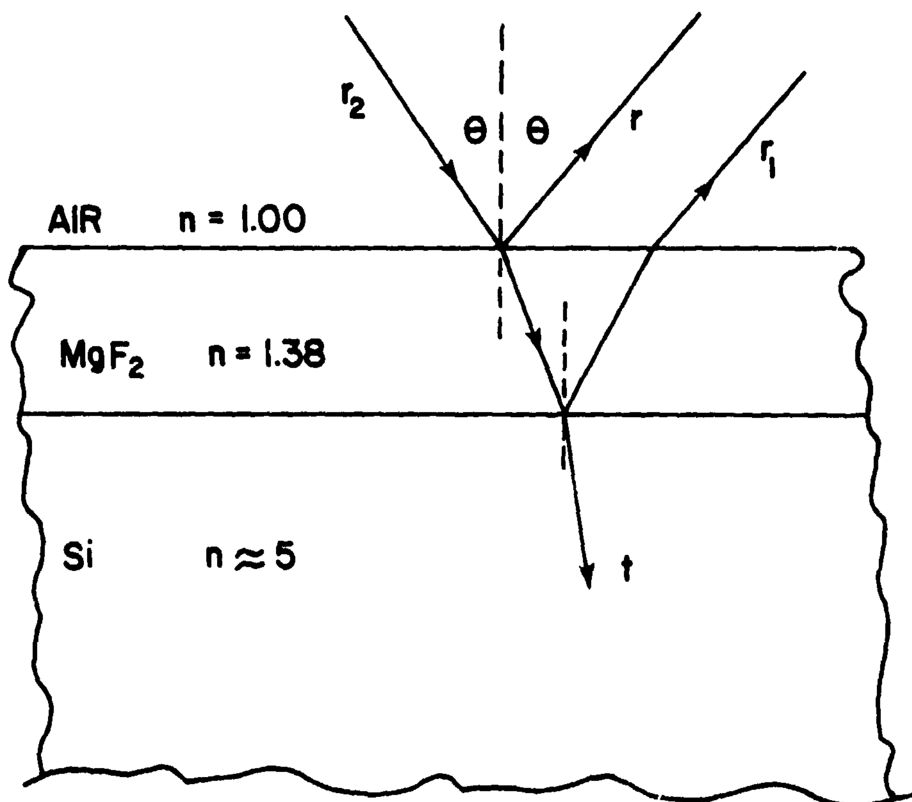


Figure C.1 Reflected and Transmitted Rays at the Surface of a Solar Cell.

where n_1 is the index of refraction of air and n_2 is the index of refraction of the coating. The value of ρ is the portion of light that is lost in the transition. If the angle of incidence is not normal to the cell surface, then ρ is modified by Fresnel's formula.

$$\rho = \frac{\tan^2(\phi_1 - \phi_2)}{2\tan^2(\phi_1 + \phi_2)} + \frac{\sin^2(\phi_1 - \phi_2)}{2\sin^2(\phi_1 + \phi_2)}$$

The angle ϕ_1 is the angle of incidence at the boundary and ϕ_2 is the angle of the refracted wave with the normal to the cell surface. These angles are related by Snell's Law:

$$n_1 \sin\phi_1 = n_2 \sin\phi_2$$

The incident ray which passes through the coating is then partly reflected again at the silicon surface. This reflected wave then returns through the AR coating to combine with the original reflected wave (Figure C.2). If the path difference between the distances that the two waves travelled is exactly $n\lambda/2$, where n is any odd integer, the waves cancel each other.

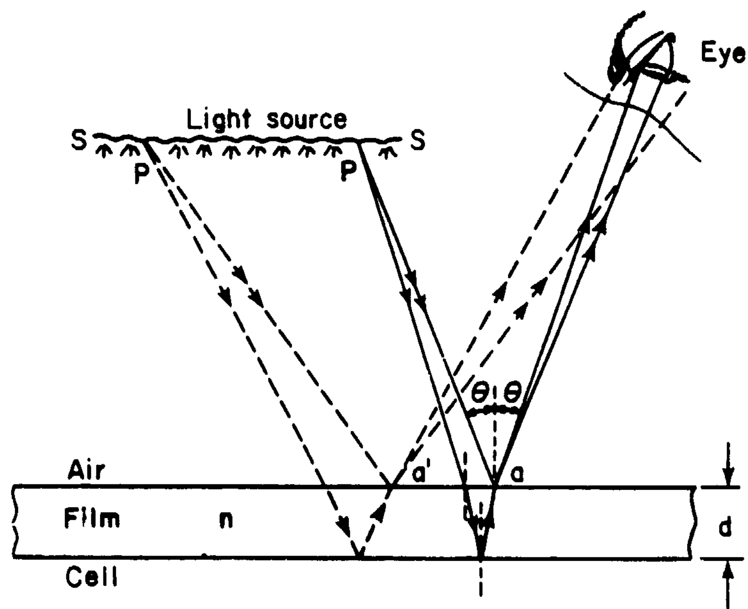


Figure C.2 Interference by Reflection from a Thin Film, Assuming an Extended Source.

Color Theory

Images are seen by stimulating the retina of the eye with light energy in the 400 to 700 nm wavelength region. Listed below are the colors of the spectrum seen by the eye at various wavelengths.

Violet	400-230 nm
Blue	430-485 nm
Green	485-560 nm
Yellow	560-585 nm
Orange	585-610 nm
Red	610-700 nm

In order to categorize one object from another by color, certain differences must be observed. These are hue, lightness, and saturation (C.2). Hue is the exact shade of color seen and is a linear function of the dominant wavelength reflected by an object. Lightness deals with lighter and darker shades of the same hue. It can range from a perfect white to black. Saturation pertains to the pureness of a color. It measures the degree of difference of a color from gray of the same lightness. The interrelationship of these three quantities can be visualized by using the diagram of Figure C.3.

Also important in color perception is the spectral distribution of the light source. The same object may appear a different color when viewed under a different type of illumination. The spectral characteristics of two common illuminants are shown in Figure C.4. Illuminant A approximates the spectral distribution produced by a tungsten lamp (indoor illumination). It produces greater energy at longer wavelengths, particularly in the infrared range. Illuminant C, however, has a more nearly flat response with a small peak at

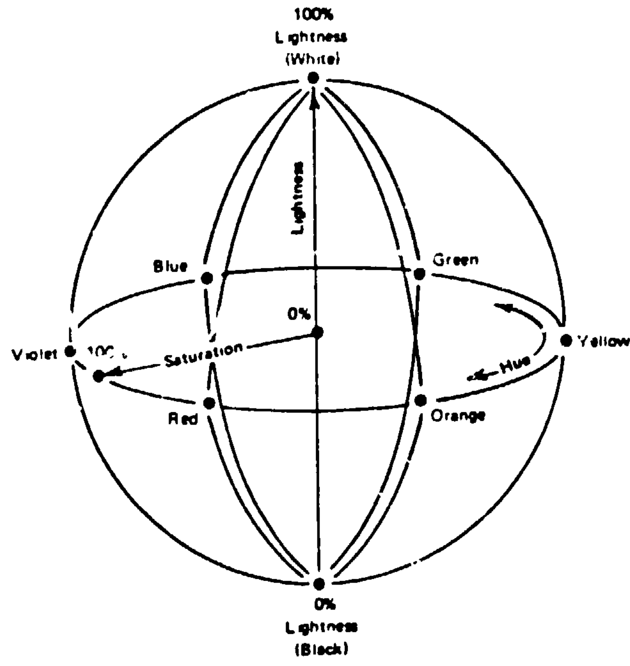


Figure C.3 The Dimensions of Color.

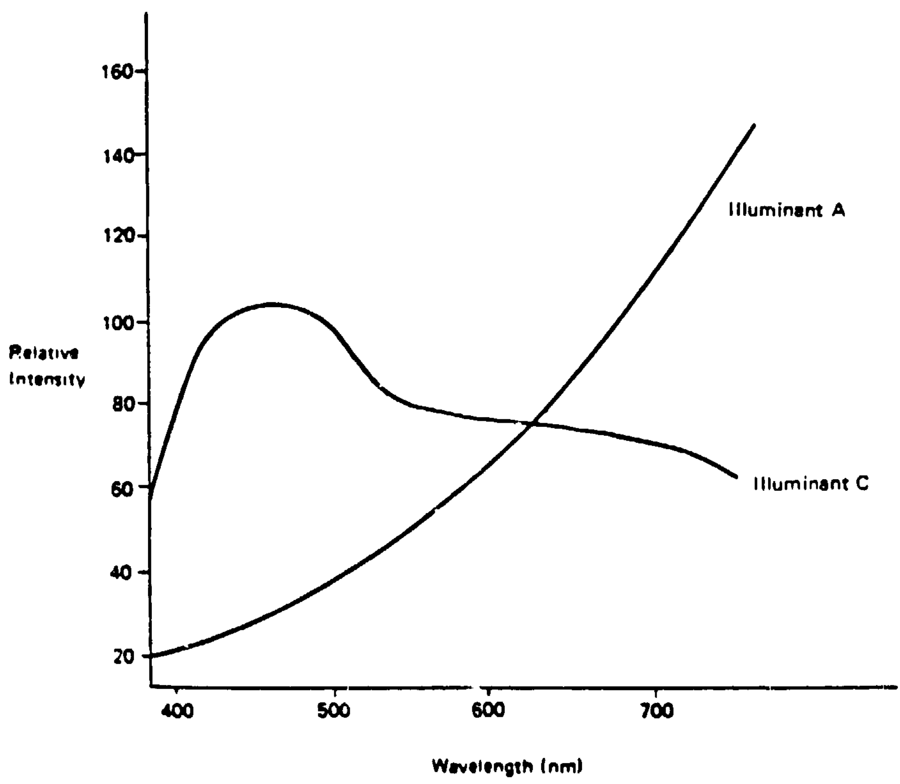


Figure C.4 Spectral Distributions of Alluminants A & C.

about 450 nm. This more nearly approximates the solar spectrum (outdoor illumination) in the visible region and produces a slightly bluish hue. The total solar spectrum is shown in Figure C.5 and it can be seen that only a limited portion is useful for photovoltaic conversion, and only a limited portion of that, can be seen with the eye.

Measurement of perceived color may be expressed in several different ways. The most straightforward approach is to state the percent spectral reflectance. This may be presented either in tabular form listing percent reflectance for specific wavelengths, or as a continuous plot over a particular range of wavelengths.

Another more advanced form allows the apparent color to be expressed as a combination of the three colors of light. These are known as tristimulus values: x for red, y for green, and z for blue. Each of these parameters range from 0 to 100; the higher the value, the more intense that particular component is. A similar representation is by chromasticity values. These coordinates, x(red) and y(green), give the per unit fractions of these two colors reflected by the smple. However, no information on light intensity is obtained.

Another widely used representation involves clor space coordinates. The most recent method accepted by the International Commission on Illumination is the 1976 CIE L*A*B formulas. The L*A*B values that are obtained can be plotted in three dimensions, as shown in Figure C.6, which aids in determining color differences. The L parameter is the overall lightness of the sample and has a range of 0 to 100 relative to the measuring device's calibration standard. The A value reveals the degree of green or red reflected by the sample. A green hue would result in a large negative A, while red is a large

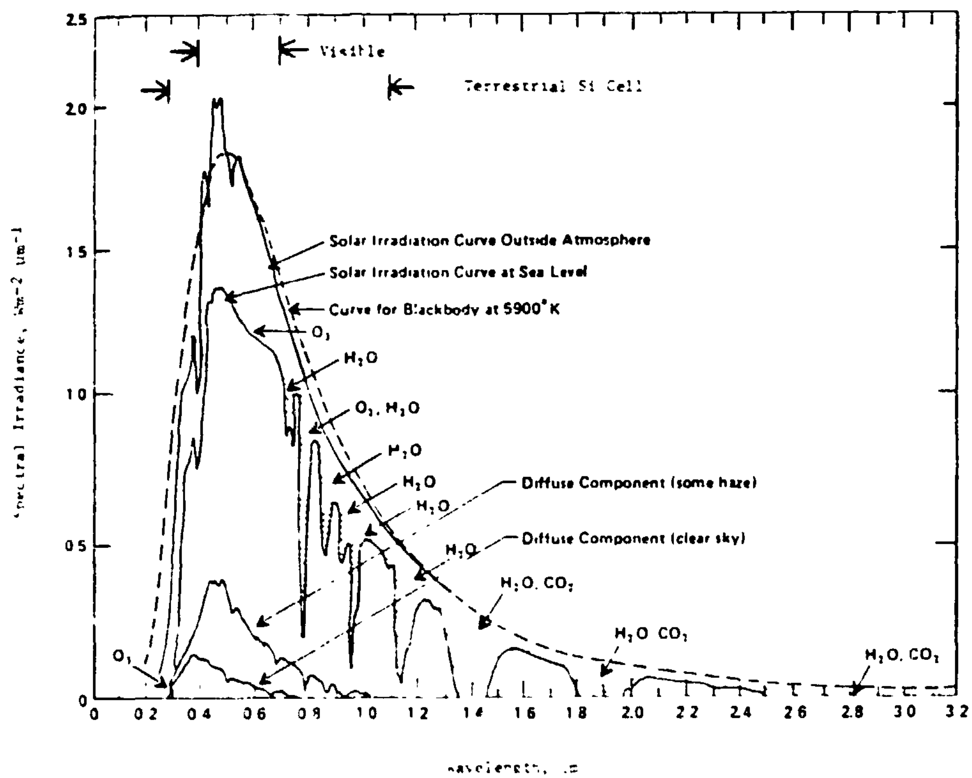


Figure C.5 Spectral Distribution of Solar Radiation.

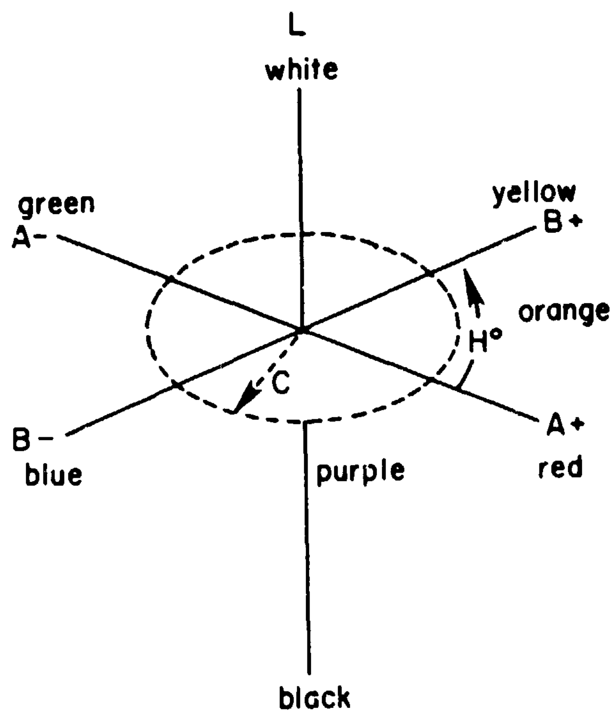


Figure C.6 1976 CIE L*A*B Color Space

positive A. Similarly, B is a meausre of the blueness or yellowness reflected, with blue in the negative scale and yellow on the positive scale.

Each type of color representaton can be obtained from the IBM measuring system. The 7409 unit divides the visible portion of the spectrum into 31 channels each of 10 nanometers width. The fiber optic detector divides the illuminated region of the sample spatially. During measurement each spatial region is samples over the 31 frequency regions using a 12-bit A/D converter. The sum of the different spatial readings for each frequency interval is stored. After all measurements are taken, this sum is divided by the number of spatial points measured to provide an average value for each frequency channel. The voltage V(I,J), for wavelength I at measurement position J, is linearly related to the percent reflectance R(I,J). The constants in this relation are determined during the device calibration procedure.

In order to calculate tristimulus values x, y, and x, the Stearn's Coefficients T (I), T (I), and T (I) must first be determined. They are found for each wavelength by multiplying the spectral tristimulus values of an equal energy spectrum for a particular observer angle by the relative spectral irradiance, E, for a specific illuminant. Tables, listing these numbers, are published by the National Bureau of Standards. The following example computes the x Stearn's coefficient at 500 nm for illuminant A with a 10° observer angle.

$$T_x (500) = x_{10} (500) \times E_A (500)$$

$$= 0.0038 \times 59.86 = 0.22747$$

The percent reflectances and the Stearn's Coefficients are then used to determine the tristimulus values using the formulas given:

$$x(J) = \sum_I (T_x(I) \times R(I,J) \times C1/\sum_I T_x(I))$$

$$y(J) = \sum_I (T_y(I) \times R(I,J) \times C2/\sum_I T_y(I))$$

$$z(J) = \sum_I (T_z(I) \times R(I,J) \times C3/\sum_I T_z(I))$$

$$C1 = 98,\ C2 = 100,\ \text{and}\ C3 = 116$$

Chromasticity values, Cx(J) and Cy(J) may be realized from the relations:

$$Cx(J) = x(J)/(x(J) + y(J) + z(J)$$

$$Cy(J) = y(J)/(x(J) + y(J) + z(J)$$

The 1976 CIE L*A*B color space coordinates are also found from the tristimulus values.

$$L(J) = 116 \times (Cx(J)/10000)^{1/3}-16$$

$$A(J) = 500 \times (Cx(J)/9800)^{1/3}-(Cy(J)/1000)^{1/3}$$

$$B(J) = 220 \times (Cy(J)/10000)^{1/3}-(Cz(J)/11600)^{1/3}$$

These parameters are used in describing the observed color changes of solar cells.

References

C.1 Solar Cell Array Design Handbook p.246-248.

C.2 Fundamentals of Physics (Halliday & Resnick) p. 715-718.

APPENDIX D. COMPUTER PROGRAM LISTINGS FOR MODELLING ENCAPSULATED AND UNENCAPSULATED CELLS IN THE SECOND QUADRANT

PRECEDING PAGE BLANK NOT FILMED

```
//RAHRTB JOB (0915-2-992-00- ,18:00,800),'ROB HARTMAN'
// EXEC FTG1CLG
//C.SYSIN DD *
C***********************************************************************
C***********************************************************************
C**                              BARE                                 **
C** THIS PROGRAM CALCULATES THE TEMPERATURE PROFILE OF A BACKWARD-    **
C** BIASED SILICON SOLAR CELL, THAT IS NOT ENCAPSULATED.              **
C** A GAUSSIAN DISTURBANCE WITH DIAMETER DEL AND AMPLITUDE LF IS A    **
C** PARAMETER THAT CAN FIT THE MODEL TO A MEASURED I-V CURVE.         **
C**                                                                   **
C***********************************************************************
C***********************************************************************
C***********************************************************************
C**                      MULTIPLE RUN CONTROL                         **
C***********************************************************************
C***********************************************************************
      INTEGER I,IF,K,L,M,IA,IL,IM
      REAL TJ,WK,CD,DR,V,ITOT,TB,TA,VMAX,TC,TT
     C,IMAX,DT,X,AI,BI,CI,DI,BETA,GAMMA,D,E
     C,HL,IRR,TINF,WSI,PI,C,A,DEL,LF,TV,VAR
      COMMON I,IF,K,L,M,IA,D,E,IL,A,IM
     C,TJ,WK,CD,DR,V,ITOT,TB(101),TA(101),VMAX,TC(100),TT(100)
     C,IMAX,DT,X,AI(100),BI(100),CI(100),DI(100),BETA(100),GAMMA(100)
     C,HL,IRR,TINF,WSI,PI,C,DEL,LF,TV(100),VAR
      DEL=25
      WSI=.04
      VAR=1.2
      LF=0.0
      CALL MN
      LF=0.2
      CALL MN
      LF=0.4
      CALL MN
      LF=0.6
      CALL MN
      END
      SUBROUTINE MN
C***********************************************************************
C***********************************************************************
C**                              MAIN                                 **
C***********************************************************************
C***********************************************************************
      INTEGER I,IF,K,L,M,IA,IL,IM
      REAL TJ,WK,CD,DR,V,ITOT,TB,TA,VMAX,TC,TT
     C,IMAX,DT,X,AI,BI,CI,DI,BETA,GAMMA,D,E
     C,HL,IRR,TINF,WSI,PI,C,A,DEL,LF,TV,VAR
      COMMON I,IF,K,L,M,IA,D,E,IL,A,IM
     C,TJ,WK,CD,DR,V,ITOT,TB(101),TA(101),VMAX,TC(100),TT(100)
     C,IMAX,DT,X,AI(100),BI(100),CI(100),DI(100),BETA(100),GAMMA(100)
     C,HL,IRR,TINF,WSI,PI,C,DEL,LF,TV(100),VAR
      IM=0
      HL=.003
      IRR=.1
      TINF=25.0
      PI=3.141592654
      A=1
      IF=0
      IL=0
      DR=.059602649
      C=IRR/HL+TINF
      DO 5 I=1,76,1
      TA(I)=C
      TB(I)=C
    5 CONTINUE
      DO 6 I=77,101,1
```

PRECEDING PAGE BLANK NOT FILMED

```
      TA(I)=TINF
      TB(I)=TINF
    6 CONTINUE
      V=0.0
      VMAX=50.0
C
C     CALCULATION OF TOTAL CURRENT
C**********************************************************************
C
      IA=1
      I=0
      ITOT=0
      TJ=TA(IA)
      CALL CURDEN
      ITOT=(CD*PI*DR**2)/4
      DO 160 I=1,75,1
      IA=I+1
      TJ=TA(IA)
      CALL CURDEN
      ITOT=ITOT+CD*I*PI*2*DR**2
  160 CONTINUE
      CALL OUTPUT
      V=25.0
      IMAX=1.2
C
C     RUNNING CONTROL
C**********************************************************************
C
      DO 200 K=1,40,1
      IMAX=IMAX+.1
      CALL RUN
      CALL FINE
      CALL OUTPUT
  200 CONTINUE
      RETURN
      END
      SUBROUTINE RUN
C**********************************************************************
C**********************************************************************
C**                NEW TEMPFRATURE RUN AND STEPSIZING                **
C**********************************************************************
C**********************************************************************
      INTEGER I,IF,K,L,M,IA,IL,IM
      REAL TJ,WK,CD,DR,V,ITOT,TB,TA,VMAX,TC,TT
     C,IMAX,DT,X,AI,BI,CI,DI,BETA,GAMMA,D,E
     C,HL,IRR,TINF,WSI,PI,C,A,DEL,LF,TV,VAR
      COMMON I,IF,K,L,M,IA,D,E,IL,A,IM
     C,TJ,WK,CD,DR,V,ITOT,TB(101),TA(101),VMAX,TC(100),TT(100)
     C,IMAX,DT,X,AI(100),BI(100),CI(100),DI(100),BETA(100),GAMMA(100)
     C,HL,IRR,TINF,WSI,PI,C,DEL,LF,TV(100),VAR
      DO 161 L=1,20,1
      DT=2
      CALL NEWTEM
  161 CONTINUE
      DO 162 L=21,40,1
      DT=1
      CALL NEWTEM
  162 CONTINUE
      DO 163 L=41,60,1
      DT=.8
      CALL NEWTEM
  163 CONTINUE
      DO 164 L=61,80,1
      DT=.6
      CALL NEWTEM
  164 CONTINUE
```

```
      DO 165 L=81,100,1
      DT=.5
      CALL NEWTEM
  165 CONTINUE
      DO 166 L=1,20,1
      DT=.4
      CALL NEWTEM
  166 CONTINUE
      DO 167 L=21,40,1
      DT=.3
      CALL NEWTEM
  167 CONTINUE
      DO 168 L=41,60,1
      DT=.2
      CALL NEWTEM
  168 CONTINUE
      DO 169 L=61,80,1
      DT=.1
      CALL NEWTEM
  169 CONTINUE
      DO 170 L=81,100,1
      DT=.08
      CALL NEWTEM
  170 CONTINUE
      DO 175 I=1,100,1
      TT(I)=0
      TC(I)=0
      TV(I)=0
  175 CONTINUE
      DO 181 L=1,20,1
      DT=.06
      CALL NEWTEM
  181 CONTINUE
      DO 182 L=21,40,1
      DT=.04
      CALL NEWTEM
  182 CONTINUE
      DO 183 L=41,60,1
      DT=.02
      CALL NEWTEM
  183 CONTINUE
      DO 184 L=61,80,1
      DT=.01
      CALL NEWTEM
  184 CONTINUE
      DO 185 L=81,100,1
      DT=.01
      CALL NEWTEM
  185 CONTINUE
      RETURN
      END
      SUBROUTINE FINE
C*****************************************************************
C*****************************************************************
C**         EXTRA FINE STEPPING ROUTINE FOR BETTER PRECISION    **
C*****************************************************************
C*****************************************************************
      INTEGER I,IF,K,L,M,IA,IL,IM
      REAL TJ,WK,CD,DR,V,ITOT,TB,TA,VMAX,TC,TT
     C,IMAX,DT,X,AI,BI,CI,DI,BETA,GAMMA,D,E
     C,HL,IRR,TINF,WSI,PI,C,A,DEL,LF,TV,VAR
      COMMON I,IF,K,L,M,IA,D,E,IL,A,IM
     C,TJ,WK,CD,DR,V,ITOT,TB(101),TA(101),VMAX,TC(100),TT(100)
     C,IMAX,DT,X,AI(100),BI(100),CI(100),DI(100),BETA(100),GAMMA(100)
     C,HL,IRR,TINF,WSI,PI,C,DEL,LF,TV(100),VAR
      DO 175 I=1,100,1
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      TT(I)=0
      TC(I)=0
      TV(I)=0
  175 CONTINUE
      DO 181 L=1,20,1
      DT=.008
      CALL NEWTEM
  181 CONTINUE
      DO 182 L=21,40,1
      DT=.006
      CALL NEWTEM
  182 CONTINUE
      DO 183 L=41,60,1
      DT=.005
      CALL NEWTEM
  183 CONTINUE
      DO 184 L=61,80,1
      DT=.003
      CALL NEWTEM
  184 CONTINUE
      DO 185 L=81,100,1
      DT=.001
      CALL NEWTEM
  185 CONTINUE
      RETURN
      END
      SUBROUTINE NEWTEM
C*********************************************************************
C*********************************************************************
C**                       NEW TEMPERATURE CALCULATION              **
C*********************************************************************
C*********************************************************************
      INTEGER I,IF,K,L,M,IA,IL,IM
      REAL TJ,WK,CD,DR,V,ITOT,TB,TA,VMAX,TC,TT
     C,IMAX,DT,X,AI,BI,CI,DI,BETA,GAMMA,D,E
     C,HL,IRR,TINF,WSI,PI,C,A,DEL,LF,TV,VAR
      COMMON I,IF,K,L,M,IA,D,E,IL,A,IM
     C,TJ,WK,CD,DR,V,ITOT,TB(101),TA(101),VMAX,TC(100),TT(100)
     C,IMAX,DT,X,AI(100),BI(100),CI(100),DI(100),BETA(100),GAMMA(100)
     C,HL,IRR,TINF,WSI,PI,C,DEL,LF,TV(100),VAR
   10 FORMAT(' ','A  ',F10.4,'  B  ',F10.4,'  C  ',F10.4,'  D  ',F10.4)
      IA=1
      I=0
      TJ=TB(IA)
C
C     CALCULATION OF CURRENTDENSITY
C
      CALL CURDEN
C
C     CALCULATION OF THERMAL CONDUCTION
C
      WK=WSI/(.58+TJ/350)
C
C     CALCULATION OF THERMAL DIFFUSION
C
      A=WK/(1.647*WSI)
C
C     CALCULATION OF FIRST SET OF MATRIX ELEMENTS
C
      AI(IA)=0
      BI(IA)=-2/DR**2-1/(A*DT)
      CI(IA)=2/DR**2
      DI(IA)=(2/DR**2-1/(A*DT))*TJ-2*TB(IA+1)/DR**2
     C-(CD*V-HL*TJ+HL*TINF+IRR)/WK
C     WRITE(6,10)AI(IA),BI(IA),CI(IA),DI(IA)
      DO 300 I=1,74,1
```

```
      IA=I+1
      TJ=TB(IA)
C
C     CALCULATION OF CURRENTDENSITY
C
      CALL CURDEN
C
C     CALCULATION OF THERMAL CONDUCTION
C
      WK=WSI/(.58+TJ/350)
C
C     CALCULATION OF THERMAL DIFFUSION
C
      A=WK/(1.647*WSI)
C
C     CALCULATION OF THE REMAINDER OF THE MATRIX ELEMENTS
C
      AI(IA)=(I-1)/((2*I-1)*DR**2)
      BI(IA)=-1/DR**2-1/(A*DT)
      CI(IA)=I/((2*I-1)*DR**2)
      DI(IA)=-AI(IA)*TB(IA-1)+(1/DR**2-1/(A*DT))*TJ
     C-CI(IA)*TB(IA+1)-(CD*V-HL*TJ+HL*TINF+IRR)/WK
     C+(TB(IA+1)-TB(IA-1))**2/((205+TJ)*4*DR**2)
C     WRITE(6,10)AI(IA),BI(IA),CI(IA),DI(IA)
  300 CONTINUE
      I=75
      IA=I+1
      TJ=TB(IA)
C
C     CALCULATION OF CURRENTDENSITY
C
      CALL CURDEN
C
C     CALCULATION OF THERMAL CONDUCTION
C
      WK=WSI/(.58+TJ/350)
C
C     CALCULATION OF THERMAL DIFFUSION
C
      A=WK/(1.647*WSI)
C
C     CALCULATION OF THE 76TH MATRIX ELEMENTS
C
      AI(IA)=1/(2*DR**2)
      BI(IA)=-1/(2*DR**2)-1/(A*DT)
      CI(IA)=0.0
      DI(IA)=-AI(IA)*TB(IA-1)-(BI(IA)+2/(A*DT))*TJ
     C-(CD*V-HL*TJ+HL*TINF+IRR)/WK
     C+75.5*HL*WSI*(1.5*TB(IA)-.5*TB(IA-1)-TINF)/(74.5*WK*DR)
C     WRITE(6,10)AI(IA),BI(IA),CI(IA),DI(IA)
C
C     CALCULATION OF TA
C
      IA=1
      BETA(IA)=BI(IA)
      GAMMA(IA)=DI(IA)/BI(IA)
      DO 380 IA=2,76,1
      BETA(IA)=BI(IA)-(AI(IA)*CI(IA-1))/BETA(IA-1)
      GAMMA(IA)=(DI(IA)-AI(IA)*GAMMA(IA-1))/BETA(IA)
  380 CONTINUE
      TA(76)=GAMMA(76)
      DO 430 M=1,75,1
      IA=76-M
      TA(IA)=GAMMA(IA)-(CI(IA)*TA(IA+1))/BETA(IA)
  430 CONTINUE
C
```

```
C     TEMPERATURE LIMITER
C
      DO 440 I=1,76,1
      IF(TA(I).LT.1000) GO TO 440
      IL=1
      TA(I)=1000
  440 CONTINUE
C
C     CALCULATION OF TOTAL CURRENT
C***********************************************************************
C
      IA=1
      I=0
      ITOT=0
      TJ=TA(IA)
C
C     CALCULATION OF CURRENTDENSITY
C
      CALL CURDEN
      ITOT=(CD*PI*DR**2)/4
      DO 450 I=1,75,1
      IA=I+1
      TJ=TA(IA)
C
C     CALCULATION OF CURRENTDENSITY
C
      CALL CURDEN
      ITOT=ITOT+CD*I*PI*2*DR**2
  450 CONTINUE
      TC(L)=TA(1)
      TT(L)=ITOT
      TV(L)=V
C
C     CURRENT LIMITER
C***********************************************************************
C
      D=1.00000001*IMAX
      E=.99999999*IMAX
      IF (ITOT.GT.D) GO TO 490
      IF (ITOT.LT.E.AND.IF.EQ.1) GO TO 495
      GO TO 500
  490 V=(V*IMAX)/ITOT
      IF =1
      GO TO 500
  495 V=(V*IMAX)/ITOT
      IF(V.LT.VMAX) GO TO 500
      V=VMAX
      IF=0
  500 CONTINUE
      IM=IM+1
      DO 540 I=1,76,1
      TB(I)=TA(I)
  540 CONTINUE
      RETURN
      END
      SUBROUTINE OUTPUT
C***********************************************************************
C***********************************************************************
C**                          OUTPUT IN TABEL FORM                     **
C***********************************************************************
C***********************************************************************
      INTEGER I,IF,K,L,M,IA,IL,IM
      REAL TJ,WK,CD,DR,V,ITOT,TB,TA,VMAX,TC,TT
     C,IMAX,DT,X,AT,BT,CT,DT,BETA,GAMMA,D,E
     C,HL,IRR,TINF,WSI,PI,C,A,DEL,LF,TV,VAR
      COMMON I,IF,K,L,M,IA,D,E,IL,A,IM
```

```
     C,TJ,WR,CD,DR,V,ITOT,IB(101),TA[illegible],VMAX,TC(100),TT(100)
     C,IMAX,DT,X,AI(100),BI(1C[illegible]),C[illegible].DI(100),BETA(100),GAMMA(100)
     C,HL,IRR,TINF,WSI,PI,C,DEL,L[illegible]00),VAR
      IF(IM.EQ.0) GO TO 1
      GO TO 9
    1 WRITE(6,2)IM,VAR
    2 FORMAT('1',4X,'TIME = [illegible]' INCREMENTS, THE VARIATION',
     C' FROM THE STANDARD [illegible]',F12.4)
      WRITE(6,3)DEL,LF
    3 FORMAT(' ',4X,'THE GA[illegible] IS '
     C,F12.4,' WIDE, AND HA[illegible] AMPLITUDE OF ',F12.4)
      GO TO 15
    9 WRITE(6,10)IM
   10 FORMAT('1',4X,'TIME = ',I12,2X,' INCREMENTS')
   15 WRITE(6,20)V
   20 FORMAT(' ',4X,'VOLTAGE = ',F12.4,'V.')
      WRITE(6,30)ITOT
   30 FORMAT(' ',4X,'CURRENT = ',F12.4,'A.')
      IF(IL.EQ.0) GO TO 50
      WRITE(6,45)
   45 FORMAT(' ',4X,'TEMPERATURE LIMITER HAS BEEN USED')
      IL=0
   50 IF(IF.EQ.0)GO TO 60
      WRITE(6,55)
   55 FORMAT(' ',4X,'CURRENT LIMITER IS IN USE')
   60 WRITE(6,65)
   65 FORMAT(' ')
      WRITE(6,66)
   66 FORMAT(' ',4X,'TEMPERATURE PROFILE')
      WRITE(6,70)
   70 FORMAT(' ',9X,'0',11X,'1',11X,'2',11X,'3',11X,'4',11X,'5',11X,
     C'6',11X,'7',11X,'8',11X,'9')
      WRITE(6,65)
      DO 100 I=1,91,10
      IA=I-1
      WRITE(6,80)IA,TA(I),TA(I+1),TA(I+2),TA(I+3),TA(I+4),TA(I+5),
     CTA(I+6),TA(I+7),TA(I+8),TA(I+9)
   80 FORMAT(' ',I2,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,
     CF10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4)
  100 CONTINUE
      DTEMP=0.0
      WRITE(6,65)
      WRITE(6,110)
  110 FORMAT(' ',4X,'CENTER TEMPERATURE AFTER EACH TIME INTERVAL')
      WRITE(6,120)
  120 FORMAT(' ',9X,'1',11X,'2',11X,'3',11X,'4',11X,'5',11X,
     C'6',11X,'7',11X,'8',11X,'9',11X,'10')
      WRITE(6,65)
      DO 140 I=1,91,10
      IA=I-1
      WRITE(6,130)IA,TC(I),TC(I+1),TC(I+2),TC(I+3),TC(I+4),TC(I+5),
     CTC(I+6),TC(I+7),TC(I+8),TC(I+9)
  130 FORMAT(' ',I2,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,
     CF10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4)
  140 CONTINUE
      WRITE(6,65)
      WRITE(6,150)
  150 FORMAT(' ',4X,'TOTAL CURRENT AFTER EACH TIME INTERVAL')
      WRITE(6,160)
  160 FORMAT(' ',9X,'1',11X,'2',11X,'3',11X,'4',11X,'5',11X,
     C'6',11X,'7',11X,'8',11X,'9',11X,'10')
      WRITE(6,65)
      DO 180 I=1,91,10
      IA=I-1
      WRITE(6,170)IA,TT(I),TT(I+1),TT(I+2),TT(I+3),TT(I+4),TT(I+5),
     CTT(I+6),TT(I+7),TT(I+8),TT(I+9)
```

```
  170 FORMAT(' ',I2,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,
     CF10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4)
  180 CONTINUE
      WRITE(6,65)
      WRITE(6,190)
  190 FORMAT(' ',4X,'VOLTAGE DURING EACH TIME INTERVAL')
      WRITE(6,200)
  200 FORMAT(' ',9X,'1',11X,'2',11X,'3',11X,'4',11X,'5',11X,
     C'6',11X,'7',11X,'8',11X,'9',11X,'10')
      WRITE(6,65)
      DO 220 I=1,91,10
      IA=I-1
      WRITE(6,210)IA,TV(I),TV(I+1),TV(I+2),TV(I+3),TV(I+4),TV(I+5),
     CTV(I+6),TV(I+7),TV(I+8),TV(I+9)
  210 FORMAT(' ',I2,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,
     CF10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4)
  220 CONTINUE
      DO 230 I=1,100,1
      TC(I)=0
      TT(I)=0
      TV(I)=0
  230 CONTINUE
      RETURN
      END
      SUBROUTINE CURDEN
C*********************************************************************
C*********************************************************************
C**                  CALCULATION OF THE CURRENTDENSITY              **
C*********************************************************************
C*********************************************************************
      INTEGER I,IF,K,L,M,IA,IL,IM
      REAL TJ,WK,CD,DR,V,ITOT,TB,TA,VMAX,TC,TT
     C,IMAX,DT,X,AI,BI,CI,DI,BETA,GAMMA,D,E
     C,HL,IRR,TINF,WSI,PI,C,A,DEL,LF,TV,VAR
      COMMON I,IF,K,L,M,IA,D,E,IL,A,IM
     C,TJ,WK,CD,DR,V,ITOT,TB(101),TA(101),VMAX,TC(100),TT(100)
     C,IMAX,DT,X,AI(100),BI(100),CI(100),DI(100),BETA(100),GAMMA(100)
     C,HL,IRR,TINF,WSI,PI,C,DEL,LF,TV(100),VAR
      D=VAR+LF*EXP(-I**2/DEL**2)
      CD=1.933E-2+8.127E-6*TJ+1.453E-7*EXP(4.585E-2*TJ*D)
     C+2.297E-5*EXP(1.208E-2*TJ*D)*V
      X=.1382*V+.023
      CD=AMIN1(CD,X)
      RETURN
      END
```

```
//RAHRTE JOB (0915-2-992-00- ,35:00,900),'ROB HARTMAN'
// EXEC FTG1CLG
//C.SYSIN DD *
C***********************************************************************
C***********************************************************************
C**                               ENCAP                               **
C** THIS PROGRAM CALCULATES THE TEMPERATURE PROFILE OF A BACKWARD-     **
C** BIASED SILICON SOLAR CELL, THAT IS ENCAPSULATED.                   **
C** A GAUSSIAN DISTURBANCE WITH DIAMETER DEL AND AMPLITUDE LF IS A     **
C** PARAMETER THAT CAN FIT THE MODEL TO A MEASURED I-V CURVE.          **
C**                                                                   **
C***********************************************************************
C***********************************************************************
C***********************************************************************
C**                      MULTIPLE RUN CONTROL                         **
C***********************************************************************
C***********************************************************************
       INTEGER I,IF,K,L,M,IA,IL,IM
       REAL TJ,WKE,WKS,CD,DR,V,ITOT,VMAX,TC,TT,TE,AS,AE
      C,TBS,TBE,TAS,TAE,AIS,BIS,CIS,DIS,BETAS,GAMS
      C,IMAX,DT,X,AIE,BIE,CIE,DIE,BETAE,GAME,D,E
      C,HECE,HEEA,IRR,TAIR,WS,PI,C,A,DEL,LF,TV,WE,KE,VAR
       COMMON I,IF,K,L,M,IA,IL,IM
      C,TJ,WKE,WKS,CD,DR,V,ITOT,VMAX,TC(100),TT(100),TE(100)
      C,TBS(101),TBE(101),TAS(101),TAE(101),AS,AE
      C,AIS(100),BIS(100),CIS(100),DIS(100),BETAS(100),GAMS(100)
      C,IMAX,DT,X,AIE(100),BIE(100),CIE(100),DIE(100)
      C,BETAE(100),GAME(100),D,E
      C,HECE,HEEA,IRR,TAIR,WS,PI,C,A,DEL,LF,TV(100),WE,KE,VAR
C      WE=.387
C      KE=7.24E-3
       WE=.4191
       KE=2.039E-3
       VAR=1.0
       LF=0.6
       WE=WE/2
       CALL MN
       WE=WE*2
       VAR=1.0
       LF=0.2
       CALL MN
       LF=0.6
       VAR=1.1
       CALL MN
       END
       SUBROUTINE MN
C***********************************************************************
C***********************************************************************
C**                                MAIN                               **
C***********************************************************************
C***********************************************************************
       INTEGER I,IF,K,L,M,IA,IL,IM
       REAL TJ,WKE,WKS,CD,DR,V,ITOT,VMAX,TC,TT,TE,AS,AE
      C,TBS,TBE,TAS,TAE,AIS,BIS,CIS,DIS,BETAS,GAMS
      C,IMAX,DT,X,AIE,BIE,CIE,DIE,BETAE,GAME,D,E
      C,HECE,HEEA,IRR,TAIR,WS,PI,C,A,DEL,LF,TV,WE,KE,VAR
       COMMON I,IF,K,L,M,IA,IL,IM
      C,TJ,WKE,WKS,CD,DR,V,ITOT,VMAX,TC(100),TT(100),TE(100)
      C,TBS(101),TBE(101),TAS(101),TAE(101),AS,AE
      C,AIS(100),BIS(100),CIS(100),DIS(100),BETAS(100),GAMS(100)
      C,IMAX,DT,X,AIE(100),BIE(100),CIE(100),DIE(100)
      C,BETAE(100),GAME(100),D,E
      C,HECE,HEEA,IRR,TAIR,WS,PI,C,A,DEL,LF,TV(100),WE,KE,VAR
       IM=0
       DEL=25
       HECE=KE/(.5*WE)
```

```
      HEEA=.003
      IRR=.1
      TAIR=25.0
      WS=.04
      PI=3.141592654
      IF=0
      IL=0
      DR=.059602649
      C=IRR/HEEA+TAIR
      DO 5 I=1,76,1
      TAS(I)=C
      TBS(I)=C
    5 CONTINUE
      DO 10 I=1,101,1
      TAE(I)=C
      TBE(I)=C
   10 CONTINUE
      V=0.0
      VMAX=50.0
C
C     CALCULATION OF TOTAL CURRENT
C*********************************************************************
C
      IA=1
      I=0
      ITOT=0
      TJ=TAS(IA)
C
C     CALCULATION OF CURRENTDENSITY FOR I=0
C
      CALL CURDEN
      ITOT=(CD*PI*DR**2)/4
      DO 160 I=1,75,1
      IA=I+1
      TJ=TAS(IA)
C
C     CALCULATION OF CURRENTDENSITY FOR 1<I<76
C
      CALL CURDEN
      ITOT=ITOT+CD*I*PI*2*DR**2
  160 CONTINUE
      CALL OUTPUT
      V=2.00
      IMAX=1.25
      AS=20
      AL=2
      IF=1
C
C     RUNNING CONTROL
C*********************************************************************
C
      DO 180 K=1,40,1
      IMAX=IMAX+.05
      CALL RUN
      CALL RUN
      CALL FINE
      CALL OUTPUT
  180 CONTINUE
      RETURN
      END
      SUBROUTINE RUN
C*********************************************************************
C*********************************************************************
C**                NEW TEMPERATURE RUN AND STEPSIZING               **
C*********************************************************************
C*********************************************************************
```

```
      INTEGER I,IF,K,L,M,IA,IL,IM
      REAL TJ,WKE,WKS,CD,DR,V,ITOT,VMAX,TC,TT,TE,AS,AE
     C,TBS,TBE,TAS,TAE,AIS,BIS,CIS,DIS,BETAS,GAMS
     C,IMAX,DT,X,AIE,BIE,CIE,DIE,BETAE,GAME,D,E
     C,HECE,HEEA,IRR,TAIR,WS,PI,C,A,DEL,LF,TV,WE,KE,VAR
      COMMON I,IF,K,L,M,IA,IL,IM
     C,TJ,WKE,WKS,CD,DR,V,ITOT,VMAX,TC(100),TT(100),TE(100)
     C,TBS(101),TBE(101),TAS(101),TAE(101),AS,AE
     C,AIS(100),BIS(100),CIS(100),DIS(100),BETAS(100),GAMS(100)
     C,IMAX,DT,X,AIE(100),BIE(100),CIE(100),DIE(100)
     C,BETAE(100),GAME(100),D,E
     C,HECE,HEEA,IRR,TAIR,WS,PI,C,A,DEL,LF,TV(100),WE,KE,VAR
      DO 161 L=1,20,1
      DT=.01
      CALL NEWTEM
C     CALL OUTPUT
  161 CONTINUE
C     CALL OUTPUT
      DO 162 L=21,40,1
      DT=.005
      CALL NEWTEM
C     CALL OUTPUT
  162 CONTINUE
C     CALL OUTPUT
      DO 163 L=41,60,1
      DT=.004
      CALL NEWTEM
  163 CONTINUE
C     CALL OUTPUT
      DO 164 L=61,80,1
      DT=.003
      CALL NEWTEM
  164 CONTINUE
C     CALL OUTPUT
      DO 165 L=81,100,1
      DT=.0025
      CALL NEWTEM
  165 CONTINUE
C     CALL OUTPUT
      DO 166 L=1,20,1
      DT=.002
      CALL NEWTEM
  166 CONTINUE
C     CALL OUTPUT
      DO 167 L=21,40,1
      DT=.0015
      CALL NEWTEM
  167 CONTINUE
C     CALL OUTPUT
      DO 168 L=41,60,1
      DT=.001
      CALL NEWTEM
  168 CONTINUE
C     CALL OUTPUT
      DO 169 L=61,80,1
      DT=.001
      CALL NEWTEM
  169 CONTINUE
C     CALL OUTPUT
      DO 170 L=81,100,1
      DT=.0008
      CALL NEWTEM
  170 CONTINUE
C     CALL OUTPUT
      DO 175 I=1,100,1
      TT(I)=0
```

```
      TC(I)=0
      TV(I)=0
  175 CONTINUE
C     CALL OUTPUT
      DO 181 L=1,20,1
      DT=.0006
      CALL NEWTEM
  181 CONTINUE
C     CALL OUTPUT
      DO 182 L=21,40,1
      DT=.0004
      CALL NEWTEM
  182 CONTINUE
C     CALL OUTPUT
      DO 183 L=41,60,1
      DT=.0002
      CALL NEWTEM
  183 CONTINUE
C     CALL OUTPUT
      DO 184 L=61,80,1
      DT=.0001
      CALL NEWTEM
  184 CONTINUE
C     CALL OUTPUT
      DO 185 L=81,100,1
      DT=.0001
      CALL NEWTEM
  185 CONTINUE
C     CALL OUTPUT
      RETURN
      END
      SUBROUTINE FINE
C*********************************************************************
C*********************************************************************
C**         EXTRA FINE STEPPING ROUTINE FOR BETTER PRECISION       **
C*********************************************************************
C*********************************************************************
      INTEGER I,IF,K,L,M,IA,IL,IM
      REAL TJ,WKE,WKS,CD,DR,V,ITOT,VMAX,TC,TT,TE,AS,AE
     C,TBS,TBE,TAS,TAE,AIS,BIS,CIS,DIS,BETAS,GAMS
     C,IMAX,DT,X,AIE,BIE,CIE,DIE,BETAE,GAME,D,E
     C,HECE,HEEA,IRR,TAIR,WS,PI,C,A,DEL,LF,TV,WE,KE,VAR
      COMMON I,IF,K,L,M,IA,IL,IM
     C,TJ,WKE,WKS,CD,DR,V,ITOT,VMAX,TC(100),TT(100),TE(100)
     C,TBS(101),TBE(101),TAS(101),TAE(101),AS,AE
     C,AIS(100),BIS(100),CIS(100),DIS(100),BETAS(100),GAMS(100)
     C,IMAX,DT,X,AIE(100),BIE(100),CIE(100),DIE(100)
     C,BETAE(100),GAME(100),D,E
     C,HECE,HEEA,IRR,TAIR,WS,PI,C,A,DEL,LF,TV(100),WE,KE,VAR
      DO 175 I=1,100,1
      TT(I)=0
      TC(I)=0
      TV(I)=0
  175 CONTINUE
      DO 181 L=1,20,1
      DT=.00008
      CALL NEWTEM
  181 CONTINUE
C     CALL OUTPUT
      DO 182 L=21,40,1
      DT=.00006
      CALL NEWTEM
  182 CONTINUE
C     CALL OUTPUT
      DO 183 L=41,60,1
      DT=.00005
```

```
      CALL NEWTEM
  183 CONTINUE
C     CALL OUTPUT
      DO 184 L=61,80,1
      DT=.00003
      CALL NEWTEM
  184 CONTINUE
C     CALL OUTPUT
      DO 185 L=81,100,1
      DT=.00001
      CALL NEWTEM
  185 CONTINUE
C     CALL OUTPUT
      RETURN
      END
      SUBROUTINE NEWTEM
C*********************************************************************
C*********************************************************************
C**                     NEW TEMPERATURE CALCULATION                 **
C*********************************************************************
C*********************************************************************
      INTEGER I,IF,K,L,M,IA,IL,IM
      REAL TJ,WKE,WKS,CD,DR,V,ITOT,VMAX,TC,TT,TE,AS,AE
     C,TBS,TBE,TAS,TAE,AIS,BIS,CIS,DIS,BETAS,GAMS
     C,IMAX,DT,X,AIE,BIE,CIE,DIE,BETAE,GAME,D,E
     C,HECE,HEEA,IRR,TAIR,WS,PI,C,A,DEL,LF,TV,WE,KF,VAR
      COMMON I,IF,K,L,M,IA,IL,IM
     C,TJ,WKE,WKS,CD,DR,V,ITOT,VMAX,TC(100),TT(100),TE(100)
     C,TBS(101),TBE(101),TAS(101),TAE(101),AS,AE
     C,AIS(100),BIS(100),CIS(100),DIS(100),BETAS(100),GAMS(100)
     C,IMAX,DT,X,AIE(100),BIE(100),CIE(100),DIE(100)
     C,BETAE(100),GAME(100),D,E
     C,HECE,HEEA,IRR,TAIR,WS,PI,C,A,DEL,LF,TV(100),WE,KE,VAR
   10 FORMAT(' ','A ',F10.4,'  B ',F10.4,'  C ',F10.4,'  D ',F10.4)
      I=0
      IA=I+1
      TJ=TBS(IA)
      TK=TBE(IA)
C
C     CALCULATION OF CURRENTDENSITY FOR I=0
C
      CALL CURDEN
C
C     CALCULATION OF THERMAL CONDUCTION
C
      WKS-WS/(.58+TJ/350)
      WKF=WF*KF
C
C     CALCULATION OF THERMAL DIFFUSION
C
C     AS=WKS/(1.647*WS)
C     AE=WKE/(1.28*WE)
C
C     CALCULATION OF FIRST MATRIX ELEMENTS ON THE CELL
C
      AIS(IA)=0
      BIS(IA)=-2/DR**2-1/(AS*DT)
      CIS(IA)=2/DR**2
      DIS(IA)=(2/DR**2-1/(AS*DT))*TJ-2*TBS(IA+1)/DR**2
     C-(CD*V-HECE*TJ+HECE*TK)/WKS
C
C     CALCULATION OF THE FIRST MATRIX ELEMENTS ON THE ENCAPSULANT
C
      AIE(IA) 0
      BIE(IA)=-2/DR**2-1/(AE*DT)
      CIE(IA)=2/DR**2
```

```
      DIE(IA)=(2/DR**2-1/(AE*DT))*TK-2*TBE(IA+1)/DR**2
     C-(HECE*TJ-HECE*TK-HEEA*TK+HEEA*TAIR+IRR)/WKE
C     WRITE(6,10)AIS(IA),BIS(IA),CIS(IA),DIS(IA)
C     WRITE(6,10)AIE(IA),BIE(IA),CIE(IA),DIE(IA)
      DO 100 I=1,74,1
      IA=I+1
      TJ=TBS(IA)
      TK=TBE(IA)
C
C     CALCULATION OF CURRENTDENSITY FOR I=1 TO I=74
C
      CALL CURDEN
C
C     CALCULATION OF THERMAL CONDUCTION
C
      WKS=WS/(.58+TJ/350)
C
C     CALCULATION OF THERMAL DIFFUSION
C
C     AS=WKS/(1.647*WS)
C
C     CALCULATION OF THE MATRIX ELEMENTS ON THE CELL
C
      AIS(IA)=(I-1)/((2*I-1)*DR**2)
      BIS(IA)=-1/DR**2-1/(AS*DT)
      CIS(IA)=I/((2*I-1)*DR**2)
      DIS(IA)=-AIS(IA)*TBS(IA-1)+(1/DR**2-1/(AS*DT))*TJ
     C-CIS(IA)*TBS(IA+1)-(CD*V-HECE*TJ+HECE*TK)/WKS
     C+(TBS(IA+1)-TBS(IA-1))**2/((205+TJ)*4*DR**2)
C
C     CALCULATION OF THE MATRIX ELEMENTS ON THE ENCAPSULANT
C
      AIE(IA)=(I-1)/((2*I-1)*DR**2)
      BIE(IA)=-1/DR**2-1/(AE*DT)
      CIE(IA)=I/((2*I-1)*DR**2)
      DIE(IA)=-AIE(IA)*TBE(IA-1)+(1/DR**2-1/(AE*DT))*TK
     C-CIE(IA)*TBE(IA+1)-(+HECE*TJ-HECE*TK-HEEA*TK+HEEA*TAIR+IRR)/WKE
C     WRITE(6,10)AIS(IA),BIS(IA),CIS(IA),DIS(IA)
C     WRITE(6,10)AIE(IA),BIE(IA),CIE(IA),DIE(IA)
  100 CONTINUE
      I=75
      IA=I+1
      TJ=TBS(IA)
      TK=TBE(IA)
C
C     CALCULATION OF CURRENTDENSITY FOR I=75
C
      CALL CURDEN
C
C     CALCULATION OF THERMAL CONDUCTION
C
      WKS=WS/(.58+TJ/350)
C
C     CALCULATION OF THERMAL DIFFUSION
C
C     AS=WKS/(1.647*WS)
C
C     CALCULATION OF MATRIX ELEMENT FOR I=75 ON THE CELL
C
      AIS(IA)=1/(2*DR**2)
      BIS(IA)=-1/(2*DR**2)-1/(AS*DT)
      CIS(IA)=0
      DIS(IA)=-AIS(IA)*TBS(IA-1)-(BIS(IA)+2/(AS*DT))*TJ
     C-(CD*V-HECE*TJ+HECE*TK)/WKS
     C+75.5*HECE*WS*(1.5*TBS(IA)-.5*TBS(IA-1)-TBE(IA+1))/(74.5*WKS*DR)
C
```

```
C     CALCULATION OF MATRIX ELEMENTS FOR I=75 ON THE ENCAPSULANT
C
      AIE(IA)=(I-1)/((2*I-1)*DR**2)
      BIE(IA)=-1/DR**2-1/(AE*DT)
      CIE(IA)=I/((2*I-1)*DR**2)
      DIE(IA)=-AIE(IA)*TBE(IA-1)+(1/DR**2-1/(AE*DT))*TK
     C-CIE(IA)*TBE(IA+1)-(+HECE*TJ-HECE*TK-HEEA*TK+HEEA*TAIR+IRR)/WKE
C     WRITE(6,10)AIS(IA),BIS(IA),CIS(IA),DIS(IA)
C     WRITE(6,10)AIE(IA),BIE(IA),CIE(IA),DIE(IA)
      I=76
      IA=I+1
C
C     CALCULATION MATRIX ELEMENT FOR I=76 ON THE ENCAPSULANT
C
      AIE(IA)=(I-1)/((2*I-1)*DR**2)
      BIE(IA)=-1/DR**2-1/(AE*DT)
      CIE(IA)=I/((2*I-1)*DR**2)
      DIE(IA)=-AIE(IA)*TBE(IA-1)+(1/DR**2-1/(AE*DT))*TBE(IA)
     C-CIE(IA)*TBE(IA+1)-(-HEEA*TBE(IA)+HEEA*TAIR+IRR)/WKE
     C-75.5*HECE*WS*(1.5*TBS(IA-1)-.5*TBS(IA-2)-TBE(IA))/(74.5*WKS*DR)
C     WRITE(6,10)AIE(IA),BIE(IA),CIE(IA),DIE(IA)
      DO 150 I=77,99,1
      IA=I+1
C
C     CALCULATION MATRIX ELEMENTS ON THE ENCAPSULANT
C
      AIE(IA)=(I-1)/((2*I-1)*DR**2)
      BIE(IA)=-1/DR**2-1/(AE*DT)
      CIE(IA)=I/((2*I-1)*DR**2)
      DIE(IA)=-AIE(IA)*TBE(IA-1)+(1/DR**2-1/(AE*DT))*TBE(IA)
     C-CIE(IA)*TBE(IA+1)-(-HEEA*TBE(IA)+HEEA*TAIR+IRR)/WKE
C     WRITE(6,10)AIE(IA),BIE(IA),CIE(IA),DIE(IA)
  150 CONTINUE
C
C     CALCULATION OF TEMPERATURE ON THE CELL AT THE END OF PERIOD
C
      IA=1
      BETAS(IA)=BIS(IA)
      GAMS(IA)=DIS(IA)/BIS(IA)
      DO 380 IA=2,76,1
      BETAS(IA)=BIS(IA)-(AIS(IA)*CIS(IA-1))/BETAS(IA-1)
      GAMS(IA)=(DIS(IA)-AIS(IA)*GAMS(IA-1))/BETAS(IA)
  380 CONTINUE
      TAS(76)=GAMS(76)
      DO 430 M=1,75,1
      IA=76-M
      TAS(IA)=GAMS(IA)-(CIS(IA)*TAS(IA+1))/BETAS(IA)
  430 CONTINUE
C
C     CALCULATION OF TEMPERATURE IN THE ENCAPSULANT AT THE END OF PERIOD
C
      IA=1
      BETAE(IA)=BIE(IA)
      GAME(IA)=DIE(IA)/BIE(IA)
      DO 435 IA=2,100,1
      BETAE(IA)=BIE(IA)-(AIE(IA)*CIE(IA-1))/BETAE(IA-1)
      GAME(IA)=(DIE(IA)-AIE(IA)*GAME(IA-1))/BETAE(IA)
  435 CONTINUE
      TAE(100)=GAME(100)-(CIE(100)*TBE(101))/BETAE(100)
      DO 436 M=1,99,1
      IA=100-M
      TAE(IA)=GAME(IA)-(CIE(IA)*TAE(IA+1))/BETAE(IA)
  436 CONTINUE
      TAE(101)=TAE(100)
C
C     TEMPERATURE LIMITER
```

```
C
      DO 440 I=1,76,1
      IF(TAS(I).LT.1000) GO TO 440
      IL=1
      TAS(I)=1000
  440 CONTINUE
C
C     CALCULATION OF TOTAL CURRENT
C*********************************************************************
C
      I=0
      ITOT=0
      TJ=TAS(1)
C
C     CALCULATION OF CURRENTDENSITY FOR I=0
C
      CALL CURDEN
      ITOT=(CD*PI*DR**2)/4
      DO 450 I=1,75,1
      IA=I+1
      TJ=TAS(IA)
C
C     CALCULATION OF CURRENTDENSITY FOR 1<I<76
C
      CALL CURDEN
      ITOT=ITOT+CD*I*PI*2*DR**2
  450 CONTINUE
      TC(L)=TAS(1)
      TE(L)=TAE(1)
      TT(L)=ITOT
      TV(L)=V
C
C     CURRENT LIMITER
C*********************************************************************
C
      D=1.00000001*IMAX
      E=.99999999*IMAX
      IF (ITOT.GT.D) GO TO 490
      IF (ITOT.LT.E.AND.IF.EQ.1) GO TO 495
      GO TO 500
  490 V=(V*IMAX)/ITOT
      IF =1
      GO TO 500
  495 V=(V*IMAX)/ITOT
      IF(V.LT.VMAX) GO TO 500
      V=VMAX
      IF=0
  500 CONTINUE
      IM=IM+1
      DO 520 I=1,76,1
      TBS(I)=TAS(I)
  520 CONTINUE
      DO 540 I=1,101,1
      TBE(I)=TAE(I)
  540 CONTINUE
C     CALL OUTPUT
      RETURN
      END
      SUBROUTINE OUTPUT
C*********************************************************************
C*********************************************************************
C**                        OUTPUT IN TABEL FORM                     **
C*********************************************************************
C*********************************************************************
      INTEGER I,IF,K,L,M,IA,II,IM
      REAL TJ,WKE,WKS,CD,DR,V,ITOT,VMAX,TC,TT,TE,AS,AL
```

```
      C,TBS,TBE,TAS,TAE,AIS,BIS,CIS,DIS,BETAS,GAMS
      C,IMAX,DT,X,AIE,BIE,CIE,DIE,BETAE,GAME,D,E
      C,HECE,HEEA,IRR,TAIR,WS,PI,C,A,DEL,LF,TV,WE,KE,VAR
       COMMON I,IF,K,L,M,IA,IL,IM
      C,TJ,WKE,WKS,CD,DR,V,ITOT,VMAX,TC(100),TT(100),TE(100)
      C,TBS(101),TBE(101),TAS(101),TAE(101),AS,AE
      C,AIS(100),BIS(100),CIS(100),DIS(100),BETAS(100),GAMS(100)
      C,IMAX,DT,X,AIE(100),BIE(100),CIE(100),DIE(100)
      C,BETAE(100),GAME(100),D,E
      C,HECE,HEEA,IRR,TAIR,WS,PI,C,A,DEL,LF,TV(100),WE,KE,VAR
C
C       PRINT STATUS
C
        IF(IM.EQ.0) GO TO 1
        GO TO 9
      1 WRITE(6,2)IM,VAR
      2 FORMAT('1',4X,'TIME = ',I12,2X,' INCREMENTS, THE VARIATION',
       C' FROM THE STANDARD CURRENT IS ',F12.4)
        WRITE(6,3)DEL,LF
      3 FORMAT(' ',4X,'THE GAUSSIAN IS ',F12.4,
       C' WIDE AND HAS AN AMPLITUDE OF ',F12.4)
        GO TO 15
      9 WRITE(6,10)IM
     10 FORMAT('1',4X,'TIME = ',I12,2X,' INCREMENTS')
     15 WRITE(6,20)V
     20 FORMAT(' ',4X,'VOLTAGE = ',F12.4,'V.')
        WRITE(6,30)ITOT
     30 FORMAT(' ',4X,'CURRENT = ',F12.4,'A.')
        IF(IL.EQ.0) GO TO 50
        WRITE(6,45)
     45 FORMAT(' ',4X,'TEMPERATURE LIMITER HAS BEEN USED')
        IL=0
     50 IF(IF.EQ.0)GO TO 60
        WRITE(6,55)
     55 FORMAT(' ',4X,'CURRENT LIMITER IS IN USE')
     60 WRITE(6,65)
     65 FORMAT(' ')
C
C       PRINT TEMPERATURE PROFILE OF THE CELL
C
        WRITE(6,66)
     66 FORMAT(' ',4X,'TEMPERATURE PROFILE OF THE CELL')
        WRITE(6,70)
     70 FORMAT(' ',9X,'0',11X,'1',11X,'2',11X,'3',11X,'4',11X,'5',11X,
       C'6',11X,'7',11X,'8',11X,'9')
        DO 100 I=1,91,10
        IA=I-1
        WRITE(6,80)IA,TAS(I),TAS(I+1),TAS(I+2),TAS(I+3),TAS(I+4),TAS(I+5),
       CTAS(I+6),TAS(I+7),TAS(I+8),TAS(I+9)
     80 FORMAT(' ',I2,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,
       CF10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4)
    100 CONTINUE
        WRITE(6,65)
C
C       PRINT TEMPERATURE PROFILE OF THE ENCAPSULANT
C
        WRITE(6,101)
    101 FORMAT(' ',4X,'TEMPERATURE PROFILE OF THE ENCAPSULANT')
        WRITE(6,102)
    102 FORMAT(' ',9X,'0',11X '1',11X,'2',11X,'3',11X,'4',11X,'5',11X,
       C'6',11X,'7',11X,'8',11X,'9')
        DO 103 I=1,91,10
        IA=I-1
        WRITE(6,104)IA,TAE(I),TAE(I+1),TAE(I+2),TAE(I+3),TAE(I+4),
       CTAE(I+5),TAE(I+6),TAE(I+7,,TAE(I+8),TAE(I+9)
    104 FORMAT(' ',I2,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,
```

```
     CF10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4)
  103 CONTINUE
      WRITE(6,65)
C
C     HISTORY OF CENTER TEMPERATURE OF THE CELL
C
C      WRITE(6,110)
  110 FORMAT(' ',4X,'CENTER TEMPERATURE OF CELL '
     C,'AFTER EACH TIME INTERVAL')
C      WRITE(6,120)
  120 FORMAT(' ',9X,'1',11X,'2',11X,'3',11X,'4',11X,'5',11X,
     C'6',11X,'7',11X,'8',11X,'9',11X,'10')
      DO 140 I=1,91,10
      IA=I-1
C      WRITE(6,130)IA,TC(I),TC(I+1),TC(I+2),TC(I+3),TC(I+4),TC(I+5),
C     CTC(I+6),TC(I+7),TC(I+8),TC(I+9)
  130 FORMAT(' ',I2,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,
     CF10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4)
  140 CONTINUE
C      WRITE(6,65)
C
C     HISTORY OF THE CENTER TEMPERATURE ON ENCAPSULANT
C
C      WRITE(6,141)
  141 FORMAT(' ',4X,'CENTER TEMPERATURE OF ENCAPSULANT '
     C,'AFTER EACH TIME INTERVAL')
C      WRITE(6,142)
  142 FORMAT(' ',9X,'1',11X,'2',11X,'3',11X,'4',11X,'5',11X,
     C'6',11X,'7',11X,'8',11X,'9',11X,'10')
      DO 144 I=1,91,10
      IA=I-1
C      WRITE(6,143)IA,TE(I),TE(I+1),TE(I+2),TE(I+3),TE(I+4),TE(I+5),
C     CTE(I+6),TE(I+7),TE(I+8),TE(I+9)
  143 FORMAT(' ',I2,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,
     CF10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4)
  144 CONTINUE
C      WRITE(6,65)
C
C     HISTORY OF TOTAL CURRENT
C
C      WRITE(6,150)
  150 FORMAT(' ',4X,'TOTAL CURRENT AFTER EACH TIME INTERVAL')
C      WRITE(6,160)
  160 FORMAT(' ',9X,'1',11X,'2',11X,'3',11X,'4',11X,'5',11X,
     C'6',11X,'7',11X,'8',11X,'9',11X,'10')
C      WRITE(6,65)
      DO 180 I=1,91,10
      IA=I-1
C      WRITE(6,170)IA,TT(I),TT(I+1),TT(I+2),TT(I+3),TT(I+4),TT(I+5),
C     CTT(I+6),TT(I+7),TT(I+8),TT(I+9)
  170 FORMAT(' ',I2,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,
     CF10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4)
  180 CONTINUE
C      WRITE(6,65)
C
C     HISTORY OF VOLTAGE
C
C      WRITE(6,190)
  190 FORMAT(' ',4X,'VOLTAGE DURING EACH TIME INTERVAL')
C      WRITE(6,200)
  200 FORMAT(' ',9X,'1',11X,'2',11X,'3',11X,'4',11X,'5',11X,
     C'6',11X,'7',11X,'8',11X,'9',11X,'10')
C      WRITE(6,65)
      DO 220 I=1,91,10
      IA=I-1
C      WRITE(6,210)IA,TV(I),TV(I+1),TV(I+2),TV(I+3),TV(I+4),TV(I+5),
```

```
C       CTV(I+6),TV(I+7),TV(I+8),TV(I+9)
  210 FORMAT(' ',I2,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,
     CF10.4,2X,F10.4,2X,F10.4,2X,F10.4,2X,F10.4)
  220 CONTINUE
      DO 230 I=1,100,1
      TC(I)=0
      TE(I)=0
      TT(I)=0
      TV(I)=0
  230 CONTINUE
      RETURN
      END
      SUBROUTINE CURDEN
C*****************************************************************
C*****************************************************************
C**              CALCULATION OF THE CURRENTDENSITY              **
C*****************************************************************
C*****************************************************************
      INTEGER I,IF,K,L,M,IA,IL,IM
      REAL TJ,WKE,WKS,CD,DR,V,ITOT,VMAX,TC,TT,TE,AS,AE
     C,TBS,TBE,TAS,TAE,AIS,BIS,CIS,DIS,BETAS,GAMS
     C,IMAX,DT,X,AIE,BIE,CIE,DIE,BETAE,GAME,D,E
     C,HECE,HEEA,IRR,TAIR,WS,PI,C,A,DEL,LF,TV,WE,KE,VAR
      COMMON I,IF,K,L,M,IA,IL,IM
     C,TJ,WKE,WKS,CD,DR,V,ITOT,VMAX,TC(100),TT(100),TE(100)
     C,TBS(101),TBE(101),TAS(101),TAE(101),AS,AE
     C,AIS(100),BIS(100),CIS(100),DIS(100),BETAS(100),GAMS(100)
     C,IMAX,DT,X,AIE(100),BIE(100),CIE(100),DIE(100)
     C,BETAE(100),GAME(100),D,E
     C,HECE,HEEA,IRR,TAIR,WS,PI,C,A,DEL,LF,TV(100),WE,KE,VAR
      D=VAR+LF*EXP(-I**2/DEL**2)
      CD=1.933E-2+8.127E-6*TJ+1.453E-7*EXP(4.585E-2*TJ*D)
     C+2.297E-5*EXP(1.208E-2*TJ*D)*V
      X=.1382*V+.023
      CD=AMIN1(CD,X)
      RETURN
      END
```