27
11/27/78
2467TIS

MASTER

BDX-613-1943 (Rev.)

ELECTRICAL COMPENSATION SYSTEM FOR
PLATED-THROUGH-HOLE TESTER

By W. B. Abel

Published October 1978

Milestone Report

Prepared for the United States Department of Energy
Under Contract Number EY-76-C-04-0613.

**Kansas City
Division**

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

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

Printed in the United States of America

Available From the National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Road, Springfield, Virginia 22161.

Price:	Microfiche	\$3.00
	Paper Copy	\$5.25

ELECTRICAL COMPENSATION SYSTEM FOR
PLATED-THROUGH-HOLE TESTER

By W. B. Abel

Published October 1978

Milestone Report
W. B. Abel, Project Leader

Project Team:
J. A. Ford
R. E. Hines

NOTICE

PORTIONS OF THIS REPORT ARE ILLEGIBLE. It
has been reproduced from the best available
copy to permit the broadest possible avail-
ability.

NOTICE

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

Communications Services

**Kansas City
Division**

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

ELECTRICAL COMPENSATION SYSTEM FOR PLATED-THROUGH-HOLE TESTER

BDX-613-1943 (Rev.), UNCLASSIFIED Milestone Report, Published October 1978

Prepared by W. B. Abel

A microprocessor system has been developed to control the acquisition and interpretation of resistance-measurement data from plated-through holes in printed wiring boards. The system provides a direct digital display of the thickness of the copper plating on the hole walls. The complex calibration for holes having different geometries is stored in the memory of the system, and the geometry factors are input through thumbwheel switches. A special probe system makes possible the determination of an average resistance value for each test hole without moving the probe contacts in relation to the hole.

WPC/OS-6

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

The Bendix Corporation
Kansas City Division
P. O. Box 1159
Kansas City, Missouri 64141

A prime contractor with the United States Department of Energy under Contract Number EY-76-C-04-0613

CONTENTS

Section	Page
SUMMARY.	6
DISCUSSION	7
SCOPE AND PURPOSE.	7
PRIOR WORK	7
ACTIVITY	7
<u>Description.</u>	7
<u>Theory of Operation.</u>	9
ACCOMPLISHMENTS.	13
FUTURE WORK.	13
APPENDICES	
A. ELECTRICAL SCHEMATICS AND DIAGRAMS OF PROTOTYPE TESTER	16
B. MEMORY-ORGANIZATION AND SOFTWARE FLOW DIAGRAMS . .	30
C. PLATED-THROUGH-HOLE PROGRAM.	33
D. CONVERSION ROUTINES.	57
DISTRIBUTION	69

ILLUSTRATIONS

Figure		Page
1	Plated-Through-Hole Illustration Showing Contactor Designation and Measurement Sequence for Probe System.	8
2	Block Diagram of Electrical Compensation System.	10
3	Diagram Showing Interaction of Operator With Tester.	12
4	Front Panel of Microprocessor System (P-98909)	14
5	Rear View of Microprocessor System Showing Internal Components (P-98910)	15
A-1	Schematic of Thumbwheel Switch Section of Electrical Compensation System.	17
A-2	Schematic of Relay Section of Electrical Compensation System	18
A-3	Schematic of Digital Display Section of Electrical Compensation System.	19
A-4	Relay Matrix.	20
A-5	Location of Integrated Circuits and Relays on Printed Circuit Board.	21
A-6	Power Supply and Control Circuits for Micro- processor	22
A-7	DLV11 Pin Connections	23
B-1	Memory-Organization Diagram	31
B-2	Software Flow Diagram	32

TABLES

Number		Page
A-1	DRV11-1 Pin Connections.	24
A-2	Pin Connections, DRV11-2 to ESI Meter.	26
A-3	Parts List for Integrated Circuits, Relays, Readouts, and Thumbwheel Switches.	27
A-4	Codes for Probe Configurations	28
A-5	Probe-Configuration-Code Example	29

SUMMARY

The goal of this project is to improve the microresistance testing of plated-through holes in printed wiring boards to provide variables data of wall thickness. A microprocessor system has been developed to control and interpret these resistance measurements. This report describes the design and functional characteristics of the system.

Used in conjunction with a probe system having eight independent contactors, the microprocessor automatically acquires four separate resistance readings for a given hole. The average value of these measurements is related to the copper thickness of the hole wall by comparison to formulations and calibration data stored in the processor memory. The resultant digital thickness value is displayed on the front panel of the tester, and it can be simultaneously recorded on a printer or teletype.

Complete electrical schematics and software programs for the present system are included in the Appendices of this report. Although an evaluation of the entire plated-through-hole testing program is in progress, functional tests of the microprocessor system and the associated electronics have already proven satisfactory.

DISCUSSION

SCOPE AND PURPOSE

The goal of this project is to improve the microresistance testing of plated-through holes in printed wiring boards so that this method can be used for the direct measurement of the copper thickness of hole walls. The program includes three integrated development efforts: (1) a new probe design; (2) an improved fixture for accurately locating the probes in relation to the test hole; and (3) a microprocessor system to control the electrical measurements and interpret the resultant resistance measurements directly in terms of hole-wall thickness.

This report describes the development activity associated with the microprocessor system. Its intent is to document the design and functional characteristics of the prototype so that the system can be duplicated for use in future testers.

PRIOR WORK

The microresistance technique for evaluating the quality of plated-through holes was established as a method that can be used at Bendix Kansas City as a go-no-go inspection test for plating thickness. Because holes having various geometric features require complex calibration procedures which prevent the direct determination of plating thickness, a microprocessor system was needed.

ACTIVITY

Description

A prototype microprocessor system has been developed which performs the following three major functions in the test method.

- Resistance data from the test hole are acquired and used to obtain an average value which is representative of the wall thickness of the plating around the circumference of the hole. The probe system has eight individual contactors--four on the top side of the hole and four on the bottom (Figure 1). To determine the resistance of the hole, a constant current is injected from one of the top contactors to the corresponding bottom contactor. A voltage measurement then is made between the similar set of contactors that is located 180 degrees from the current contactors. This voltage measurement is

MEASUREMENT NUMBER	CURRENT INPUT	VOLTAGE
1	1 - 5	3 - 7
2	2 - 6	4 - 8
3	3 - 7	1 - 5
4	4 - 8	2 - 6

Figure 1. Plated-Through-Hole Illustration Showing Contactor Designation and Measurement Sequence for Probe System

converted to a resistance value by a digital resistance-measuring system. Through a series of reed-type relays, the microprocessor system switches the current and voltage connections to the set of contactors that is located 90 degrees from the previously employed set, and a second resistance value is obtained. The process then is repeated for the other two combinations of current-and-voltage measurements.

- The four resistance values are stored in the memory of the microprocessor, and they are examined to determine whether any one is unusual as compared to the others. The values then are averaged, and the average value is used for the conversion to plating thickness.

- The plating thickness is determined by using the data input to the system, the resistance value (as previously determined), the hole diameter, and the hole length (board thickness). The latter two parameters are dialed into the system through the use of the thumbwheel switches located on the front panel of the controller (Figure 2). There also are thumbwheel switches which can be used by the operator to identify the part number and the number of the hole being tested.

Three techniques for converting the measured resistance values to plating thickness are now being evaluated: (1) using an equation for a direct calculation of the plating thickness; (2) storing tables of calibration data in the memory of the microprocessor system and having the system search for the thickness value associated with a given resistance value and set of geometry factors; and (3) combining both of the first two methods. Details concerning the equations and the calibration data involved will be published in a later report. The important point is that the specific technique used is dependent only on the software program for the microprocessor system and does not require hardware changes.

Theory of Operation

The microcomputer used in the system contains 8K of random-access memory, and it interfaces to a teletype printer and paper tape through a DLV11 serial interface. Two DRV11 parallel interfaces are employed: DRV11-1 is used to output information to the relays and LED display, and to read the thumbwheel settings; and DRV11-2 interfaces to the ESI resistance meter, and is read in five BCD digits.

By sending the appropriate code word through DRV11-1, the decoder Integrated Circuit 1 (IC1) causes the appropriate thumbwheels to be read into the computer through the NAND gates and DRV11-1. The digits are read in groups of three until all of the thumbwheel settings for ID, S/N, hole number, diameter, and board thickness have been read.

When the READ pushbutton PBI is activated, the computer issues code words through DRV11-1 to set the relays into various configurations to control the location of the voltage and current probes around the hole. The probes are set into four different configurations, and the resistance of the plated-through hole is read from the ESI resistance meter for each configuration.

Figure 2. Block Diagram of Electrical Compensation System

An average resistance, $\bar{R}$, is calculated from the four readings. This average resistance value is used with the test-hole geometric parameters for the conversion to the plating thickness of the hole wall. The plating thickness is displayed on the three-digit LED display in thousandths of an inch, and the display data are output from the DRV11-1 interface. The PRINT pushbutton switch PD2 is activated when desired to print on the teletype the data stored since the last printout. With the present program, data from approximately 50 holes can be stored for printout.

Detailed electrical schematics and other diagrams which explain the microprocessor system are included in Appendix A. Figures A-1 through A-3 show, in detail, the interface electronics to the thumbwheels, relays, and display. Figure A-4 shows the wiring configuration and pin numbers to switch the arrangement of the probes, while Figure A-5 shows the physical location of the ICs and relays on the printed circuit board. Figure A-6 shows the power supplies that are required together with a schematic of the additional circuitry used to control the microprocessor from the front panel of the tester.

Figure A-7, along with Tables A-1 and A-2, outlines the pin connections from the computer interface cards (DLV11, DRV11-1, and DRV11-2) to the outside circuitry. Table A-3 lists the integrated circuits, thumbwheels, relays, and displays that are required in the construction of the tester.

Table A-4 shows the method used to construct the software code word to store in the memory which causes the probes to switch into the desired arrangement. An example of the code is provided in Table A-5.

Figure 2 is a block diagram which shows the interrelationship of the major components of the system. The teletype has been used in the prototype system as a method for communicating with the microprocessor and for obtaining the data printout. It will be replaced with a digital printer in the final design because changes in the software programs will not be necessary.

Figure 3 is a flow diagram illustrating the interaction of the operator with the tester. The first operational step consists of setting the thumbwheel switches on the front panel to identify the part number of the part to be tested. The specific hole to be tested then is selected, and the hole number, hole diameter, and laminate thickness are set by using the appropriate thumbwheel switches. The value used for the hole diameter is the diameter of the drill that has been used, and the laminate thickness is the nominal thickness of the laminate before plating.

Figure 3. Diagram Showing Interaction of Operator With Tester

The operator locates the test hole by centering it in relation to the reticle in the microscope portion of the test fixture. The table then is rotated to the left to position the hole in line with the probe system, and the lever is pulled down to bring the probes in contact with the hole. The READ button on the front panel is pushed to initiate the test. This procedure is repeated until all of the holes on the board have been tested. After the last hole has been tested, the operator again pushes the READ button, and the data, including the identification, for all the holes on the board are printed on the teletype or printer.

The front panel and the back of the microprocessor system are shown in Figures 4 and 5, respectively. The thumbwheel switches, the ESI measuring system, and the thickness readout can be seen in the front view. The back view shows the reed relays, the basic microprocessor, the back of the ESI unit, the power supply, and the connecting electrical circuitry.

Figure B-1 of Appendix B shows the location in the memory where each of the software programs and the data are stored, while Figure B-2 shows the major operations that are performed. Details can be obtained by examining the computer listings that are now being used (Appendices C and D). The Plated-Through-Hole Program (Appendix C) was written by Bendix Kansas City personnel to perform the desired operations. The Conversion Routines (Appendix D) perform standard conversions of data from one mode to another.

ACCOMPLISHMENTS

The development of the electrical compensation system described in this report demonstrates the feasibility of using a microprocessor to control the acquisition of resistance data from a multiple-contact probe, and to convert the measured resistance of plated-through holes to plating thickness. The use of this concept makes possible the determination of the average thickness of a connection through a printed circuit board by determining its average resistance without reorienting the test hole.

FUTURE WORK

The electrical compensation system will be evaluated in conjunction with the entire plated-through-hole testing program to establish its reliability. The evaluation will include a joint study with the Design Laboratory at LLL, and it will be designed and conducted by the Quality Control Engineering group.

Figure 4. Front Panel of Microprocessor System

Figure 5. Rear View of Microprocessor System Showing Internal Components

Appendix A

ELECTRICAL SCHEMATICS AND DIAGRAMS OF
PROTOTYPE TESTER

Figure A-1. Schematic of Thumbwheel Switch Section of Electrical Compensation System

Figure A-2. Schematic of Relay Section of Electrical Compensation System

Figure A-3. Schematic of Digital Display Section of Electrical Compensation System

Figure A-4. Relay Matrix

Figure A-5. Location of Integrated Circuits and Relays on Printed Circuit Board

Figure A-6. Power Supply and Control Circuits for Microprocessor

SERIAL I/O DLV11
ADDRESS: 177560
VECTOR: 060

Figure A-7. DLV11 Pin Connections

Table A-1. DRV11-1 Pin Connections*

Signal	Connector and Pin	Relay Function	Display Function
OUT00	1 - C	A	LSD A
OUT01	1 - K	B	LSD B
OUT02	1 - NN	C	LSD C
OUT03	1 - V	A	LSD D
OUT04	1 - L	B	Center A
OUT05	1 - N	C	Center B
OUT06	1 - R	A	Center C
OUT07	1 - T	B	Center D
OUT08	1 - W	C	MSD A
OUT09	1 - X	A	MSD B
OUT10	1 - Z	B	MSD C
OUT11	1 - AA	C	MSD D
OUT12	1 - BB	A	
OUT13	1 - FF	B	
OUT14	1 - HH	C	
OUT15	1 - JJ	D	
GND	1 - UU		

Thumbwheel Function

IN00	2 - TT	MSD A
IN01	2 - LL	MSD B
IN02	2 - H,E	MSD C
IN03	2 - BB	MSD D
IN04	2 - KK	Center A
IN05	2 - HH	Center B
IN06	2 - EE	Center C
IN07	2 - CC	Center D
IN08	2 - Z	LSD A
IN09	2 - Y	LSD B

Table A-1 Continued. DRV11-1 Pin Connections*

Signal	Connector and Pin	Thumbwheel Function	
IN10	2 - W	LSD	C
IN11	2 - V	LSD	D
INIT	2 - RR		
REQ A	1 - LL	Print	
REQ B	2 - S	Read	
GND	2 - J		
*Parallel I/O DRV11-1. Address: 16777X.			

Table A-2. Pin Connections, DRV11-2 to ESI Meter*

Signal	Connector and Pin	ESI Meter	
		Pin	Function
IN00**	2 - TT	50	Bit 1
IN01	2 - LL	46	Bit 2
IN02	2 - H,E	47	Bit 4
IN03	2 - BB	49	Bit 8
IN04***	2 - KK	23	DEC Pt 3
IN05	2 - HH	24	DEC Pt 2
IN06	2 - EE	25	DEC Pt 1
REQ A†	1 - LL	29	Data Ready
OUT00††	1 - C	48	Read Digit 1 (LSD)
OUT01	1 - K	26	Read Digit 2
OUT02	1 - NN	43	Read Digit 3
OUT03	1 - U	38	Read Digit 4
OUT04	1 - L	31	Read Digit 5 (MSD)
GND	1 - R	5	Gnd
CSRD	2 - K	37	SB
Inverted BCD Output From ESI (0 = True)		4†††	+5V

*Parallel I/O: DRV11-2 to ESI meter. Address: 16776X .
Vector: 130.

**Digits 1 through 4 are BCD. Digit 5 contains the following:
Bit 1 = MSD 0, 1; Bit 2 = Overrange = 0; and Bit 4 = Sign,
1 = Minus, 0 = Plus.

***0 = Decimal Point (DP). DP1 = 1.XXX (2 Ω FS); DP2 = 1X.XXX
(20 m Ω FS); DP3 = 1XX.XX (200 m Ω FS).

†Data Ready, normally 0, goes high for 4 ms at end of
measurement.

††1 = Read Digit.

†††+5V from ESI brought to Pin 4; pull-up resistors in connector
are tied from inputs to Pin 4.

Table A-3. Parts List for Integrated Circuits, Relays, Readouts, and Thumbwheel Switches

Part	Serial Number	Quantity	Description
Integrated Circuit	SN7438N	18	NAND Gates
	SN74154N	1	4-to-16 Decoder
	SN74155N	4	3-to-8 Decoder
	SN7404	3	Inverter
	SN7474	1	Flip-Flop
	2N3906	32	Transistor
Reed Relay	W101MPCX-122	32	
LED Readout	5082-7302	3	
Thumbwheel Switch	29102/B-3	5	
	29102/B-6	1	

Table A-4. Codes for Probe Configurations*

ESI-Input Connector Pin	Word-Bit Position	Probe and Connector Pin							
		1	2	3	4	5	6	7	8
		A	B	C	D	E	F	H	J
Pin 1 (Current 1)	00	0	1	0	1	0	1	0	1
	01	0	0	1	1	0	0	1	1
	02	0	0	0	0	1	1	1	1
Pin 2 (Voltage 1)	03	0	1	0	1	0	1	0	1
	04	0	0	1	1	0	0	1	1
	05	0	0	0	0	1	1	1	1
Pin 4 (Voltage 2)	06	0	1	0	1	0	1	0	1
	07	0	0	1	1	0	0	1	1
	08	0	0	0	0	1	1	1	1
Pin 5 (Current 2)	09	0	1	0	1	0	1	0	1
	10	0	0	1	1	0	0	1	1
	11	0	0	0	0	1	1	1	1
Output Code	12	0	0	0	0	0	0	0	0
	13	1	1	1	1	1	1	1	1
	14	1	1	1	1	1	1	1	1
	15	0	0	0	0	0	0	0	0

*To use the table, add the bit positions to make a 16-bit word.

Example: Refer to Table A-5 for an example to set Current 1 = Probe 1, Voltage 1 = Probe 3, Voltage 2 = Probe 5, Current 2 = Probe 7.

Table A-5. Probe-Configuration-Code Example

		Word-Bit Position															
		15	14 13 12			11 10 09			08 07 06			05 04 03			02 01 00		
Function						I2			V2			V1			I1		
Binary	0	1	1	0	1	1	0	1	0	0	0	1	0	0	0	0	0
Octal	0	6			6			4			2			0			

Appendix B

MEMORY-ORGANIZATION AND SOFTWARE FLOW DIAGRAMS

Figure B-1. Memory-Organization Diagram

Figure B-2. Software Flow Diagram

Appendix C

PLATED-THROUGH-HOLE PROGRAM

```

1          ;PLATED THRU HOLE PROGRAM
2          ;CALLED BY PCHOLE
3          ;DOES NOT PRINT 4 RESISTANCE
4          ;READINGS ONLY THE AVERAGE
5          ;I/O ADDRESSES: TELETYPE =17756X
6          ;THUMB WHEELS & DISPLAY=16777X
7          ;ESI = 16776X
8          ;LINK PROGRAM TO START AT 2000
9
10
11          000000      .ASECT
12          002000      .=2000
13
14
15          000000 R0=%0
16          000001 R1=%1
17          000002 R2=%2
18          000003 R3=%3
19          000004 R4=%4
20          000005 R5=%5
21          000006 SP=%6
22          000007 PC=%7
23          000015 CR=15
24          000012 LF=12
25          000040 SA=40
26
27 02000      V003: ;VERSION ID
28 02000 010706 START: MOV PC, SP
29 02002 005746 TST -(SP) ;SET UP STACK
30 02004 012704 MOV #1000, R4 ;ADD. FOR PRINTOUT STORAGE
31 02010 012737 RDHOLE: MOV #100000, @#167772 ;CLEAR FLIP FLOP
32 02016 005037 CLR @#167772
33 02022 022704 CMP #1000, R4 ;CHECK FOR FIRST READ
34 02026 001406 BEQ RDHOL1 ;SKIP PRINT TEST
35 02030 032737 RDHOX: BIT #200, @#167770 ;TEST FOR REQ A(PRINT)
36 02036 001402 BEQ RDHOL1
37 02040 000167 JMP PRINT1 ;GO TO PRINT RTN
38 02044 032737 RDHOL1: BIT #100000, @#167770
39 02052 001756 BEQ RDHOLE ;NOT REQ B (READ) LOOP
40 02054 012737 MOV #77777, @#167772 ;MOVE BLANK TO DISPLAY
41 02062 004767 JSR PC, SAVHO ;SAV HOLE #
42 02066 004767 JSR PC, THUMB ;SAV DIA.
43 02072 004767 JSR PC, SAVBT ;SAV BOARD THICKNESS

```

```

000276
44 02076 004767 JSR PC, MEAS ;GET 4 ESI READINGS
000470
45 02102 004767 JSR PC, CALC ; DO ARITH.
001246
46 02106 004767 JSR PC, DISPLA ;DISPLAY RESULTS
003464
47 02112 004767 JSR PC, SAVE ;SAVE RESULTS
002562
48 02116 000167 JMP RDHOLE
177666
49 02122 000015 MES.ST: ST2-ST1
50 02124 012 ST1: .BYTE 012,012,015
02125 012
02126 015
51 02127 040 .ASCII / I.D. * = /
02130 111
02131 056
02132 104
02133 056
02134 040
02135 043
02136 040
02137 075
02140 040
52 02141 ST2: .EVEN
53
54 ; BLANK DISPLAY UNTIL READING FINISHED
55
56 ; GET SERIAL * AND SAVE FOR PRINTOUT
57
58 02142 012737 SAVID: MOV #10000, @#167772 ; GET 3 ID #'S
010000
167772
59 02150 012703 MOV #AIDENT+4, R3
002214
60 02154 013700 MOV @#167774, R0
167774
61 02160 004767 JSR PC, ASCII
000352
62 02164 012737 MOV #0, @#167772 ;GET 3 MORE
000000
167772
63 02172 012703 MOV #AIDEN2+4, R3
002220
64 02176 013700 MOV @#167774, R0
167774
65 02202 004767 JSR PC, ASCII
000330
66 02206 000207 RTS PC
67 02210 000000 AIDENT: .WORD 0,0
02212 000000
68 02214 000000 AIDEN2: .WORD 0,0
02216 000000
69
70 02220 012737 SAVSN: MOV #40000, @#167772
040000

```

```

167772
71 02226 012703      MOV      #ASAVSN+4, R3
      002250
72 02232 013700      MOV      @#167774, R0      ;GET 3 S/N
      167774
73 02236 004767      JSR      PC, ASCII
      000274
74 02242 000207      RTS      PC
75
76 02244 000000 ASAVSN: .WORD 0,0
      02246 000000
77
78      ; GET HOLE NUMBER AND SAVE FOR PRINTOUT
79
80 02250 012737 SAVHO: MOV      #50000, @#167772
      050000
      167772
81 02256 012703      MOV      #ASAVHO+4, R3
      002300
82 02262 013700      MOV      @#167774, R0      ;GET 3 HOLE DIGITS
      167774
83 02266 004767      JSR      PC, ASCII
      000244
84 02272 000207      RTS      PC
85
86 02274 000000 ASAVHO: .WORD 0,0
      02276 000000
87
88      ;READ 3 BCD DIGITS FROM THUMB WHEEL
89      ;FOR HOLE DIAMETER
90      ;CALL BY JSR PC ,THUMB
91 02300 012737 THUMB: MOV      #30000, @#167772
      030000
      167772
92 02306 012703      MOV      #ADIAM+6, R3      ;END ASCII BUF
      002364
93 02312 013700      MOV      @#167774, R0      ;READ 4 BCD DIGITS
      167774
94
95 02316 004767      JSR      PC, INVER      ;SHIFT DIGITS IN R0
      000140
96 02322 004767      JSR      PC, ASCII      ;CONVRT TO ASCII
      000210
97 02326 012701      MOV      #6, R1      ;BUF LENGTH
      000006
98 02332 012700      MOV      #ADIAM, R0      ;ASCII BUF @
      002356
99 02336 004767      JSR      PC, INCVRT      ;CARRY SET IF ERROR
      000730
100 2342 010067      MOV      R0, DIAM      ;HO NUM
      000016
101 2346 001410      BEQ      T.ERR      ;DIAM IS ZERO
102 2350 010167      MOV      R1, DIAM+2      ;LO NUM
      000012
103 2354 000207      RTS      PC      ;TO MAIN
104
105 2356      060 ADIAM: .ASCII /0./      ; 0.XXXX

```



```

2357 056
106 002364 .,+.4
107 2364 000000 DIAM: .WORD 0,0
2366 000000
108 2370 000261 T.ERR: SEC
109 2372 000207 RTS PC ;SHOW ERROR
110
111
112 ; GET BOARD THICKNESS AND SAVE FOR CALC
113
114 2374 012737 SAVBT: MOV #20000, @#167772
020000
167772
115 2402 012703 MOV #ASAVBT+6, R3
002456
116 2406 013700 MOV @#167774, R0 ; GET BOARD THICKNESS
167774
117 2412 004767 JSR PC, INVER ;SHIFT DIGITS IN R0
000044
118 2416 004767 JSR PC, ASCII
000114
119 2422 012701 MOV #5, R1
000005
120 2426 012700 MOV #ASAVBT, R0
002450
121 2432 004767 JSR PC, INCVRT
000634
122 2436 010067 MOV R0, BOARD
000014
123 2442 010167 MOV R1, BOARD+2
000012
124 2446 000207 RTS PC
125
126 2450 060 ASAVBT: .ASCII/0./ ; 0.XXX
2451 056
127 002456 .,+.4
128 2456 000000 BOARD: .WORD 0,0
2460 000000
129
130 ; ROTATE BCD DIGITS IN R0
131
132 2462 005002 INVER: CLR R2
133 2464 010346 MOV R3, -(SP) ; SAVE R3
134 2466 012703 MOV #4, R3
000004
135 2472 010001 INVER2: MOV R0, R1
136 2474 042701 BIC #177760, R1
177760
137 2500 060102 ADD R1, R2
138 2502 005303 DEC R3
139 2504 001003 BNE INVER3
140 2506 010200 MOV R2, R0
141 2510 012603 MOV (SP)+, R3
142 2512 000207 RTS PC
143 2514 006302 INVER3: ASL R2
144 2516 006302 ASL R2
145 2520 006302 ASL R2

```

146	2522	006302	ASL	R2	
147	2524	006200	ASR	R0	
148	2526	006200	ASR	R0	
149	2530	006200	ASR	R0	
150	2532	006200	ASR	R0	
151	2534	000756	BR	INVER2	
152					
153					
154					
155					
156					
157					
158					
159					
160	2536	005100	ASCII:	COM	R0
161	2540	016701	MOV	FOUR, R1	; NEG COUNTER
		000016			
162	2544	010002	ANEXT:	MOV	R0, R2
163	2546	042702	BIC	#177760, R2	;LOW BCD DIGIT
		177760			
164	2552	062702	ADD	#60, R2	;ASCII DIGIT
		000060			
165	2556	110243	MOVB	R2, -(R3)	;STORE RT TO LEFT
166	2560	072027	.WORD	072027	;ASH R0, #-4
167	2562	177774	FOUR:	.WORD -4	;NUM BITS RIGHT
168	2564	005201	INC	R1	
169	2566	001366	BNE	ANEXT	;BIF MORE
170	2570	000207	RTS	PC	
171					
172					
173					
174	2572	013737	MEAS:	MOV	@*RELAY, @*167772
		002744			;SET RELAY 1
		167772			
175	2600	004767	JSR	PC, ESI1	;READ ESI
		000150			
176	2604	016767	MOV	RESIS, RESI1	;STORE RESULTS
		000426			
		000112			
177	2612	016767	MOV	RESIS+2, RESI1+2	
		000422			
		000106			
178	2620	013737	MOV	@*RELAY+2, @*167772	;SET RELAY 2
		002746			
		167772			
179	2626	004767	JSR	PC, ESI1	
		000122			
180	2632	016767	MOV	RESIS, RESI2	
		000400			
		000070			
181	2640	016767	MOV	RESIS+2, RESI2+2	
		000374			
		000064			
182	2646	013737	MOV	@*RELAY+4, @*167772	;SET RELAY 3
		002750			
		167772			
183	2654	004767	JSR	PC, ESI1	

```

000074
184 2660 016767      MOV      RESIS, RESI3
000352
000046
185 2666 016767      MOV      RESIS+2, RESI3+2
000346
000042
186 2674 013737      MOV      @*RELAY+6, @*167772      ;SET RELAY 4
002752
167772
187 2702 004767      JSR      PC, ESI1
000046
188 2706 016767      MOV      RESIS, RESI4
000324
000024
189 2714 016767      MOV      RESIS+2, RESI4+2
000320
000020
190 2722 000207      RTS      PC
191
192 2724 000000 RESI1: .WORD 0,0
2726 000000
193 2730 000000 RESI2: .WORD 0,0
2732 000000
194 2734 000000 RESI3: .WORD 0,0
2736 000000
195 2740 000000 RESI4: .WORD 0,0
2742 000000
196 2744 062057 RELAY: .WORD 062057, 061634
2746 061634
197 2750 060275      .WORD 060275, 066143
2752 066143
198
199 2754 012737 ESI1:  MOV      #1, @*167760      ;START ESI
000001
167760
200 2762 012737      MOV      #0, @*167760      ;STARTS ON NEG LOGIC
000000
167760
201 2770 013700 ESI2:  MOV      @*167760, R0
167760
202 2774 032700      BIT      #200, R0      ;TEST REQ A FOR FINISH
000200
203 3000 001773      BEQ      ESI2
204 3002 004767      JSR      PC, ESI
000002
205 3006 000207      RTS      PC
206
207      ;READ RESISTANCE FROM ESI INPUTS ARE NEG LOGIC
208      ;ERROR IF OVFL, NEG NUM, DEC PT WRONG, OR ZERO RESISTANCE
209
210 3010 012701 ESI:    MOV      #1, R1      ;DIGIT SELECT
000001
211 3014 012702      MOV      #EBUF, R2      ;TEMP STORAGE
003212
212 3020 010137 N.ESI: MOV      R1, @*167762      ;SEL DIGIT
167762

```

213	3024	013712	MOV	@#167764, (R2)	;READ DIGIT
		167764			
214	3030	005122	COM	(R2)+	;POS LOGIC
215	3032	006301	ASL	R1	;NEXT DIGIT
216	3034	020127	CMP	R1, #40	;END?
		000040			
217	3040	001367	BNE	N.ESI	;BIF NO
218					
219					;GET ASCII EXPONENT
220					
221	3042	012701	MOV	#ARESI+8,, R1	;@ AFTER 'E'
		003234			
222	3046	012702	MOV	#EBUF, R2	
		003212			
223	3052	011200	MOV	(R2),R0	;LEAVE DP BITS
224	3054	042700	BIC	#177617, R0	
		177617			
225	3060	072027	.WORD	072027	;ASH R0, #-3
226	3062	177775	.WORD	-3	;SHIFT RIGHT 3
227	3064	016011	MOV	EXP(R0), (R1)	;OFFSET FROM TABLE
		003200			
228	3070	105741	TSTB	-(R1)	;POINT TO 'E'
229					
230					;GET DIGITS 1-4
231					
232	3072	012703	MOV	#4, R3	;COUNTER
		000004			
233	3076	012200	NEXT.D: MOV	(R2)+, R0	;GET DIGIT
234	3100	042700	BIC	#177760, R0	;BCD
		177760			
235	3104	062700	ADD	#60, R0	;ASCII
		000060			
236	3110	110041	MOVB	R0, -(R1)	;FROM RT TO LEFT
237	3112	005303	DEC	R3	
238	3114	001370	BNE	NEXT.D	
239					;DIGIT 5
240	3116	032712	BIT	#4, (R2)	;POS NUM
		000004			
241	3122	001424	BEQ	ESI.ER	;BIF NO
242	3124	032712	BIT	#2, (R2)	;OVFL SET
		000002			
243	3130	001044	BNE	OVFLO	;BRANCH IF OVERFLOW
244	3132	042712	BIC	#177776, (R2)	;LEAVE BIT 00
		177776			
245	3136	062712	ADD	#60, (R2)	;ASCII 0 TO 1
		000060			
246	3142	111241	MOVB	(R2),-(R1)	
247					;CONVRT TO BINARY
248	3144	012701	CONAB: MOV	#10,, R1	;LENGTH
		000012			
249	3150	012700	MOV	#ARESI, R0	;ASCII @
		003224			
250	3154	004767	JSR	PC, INCVRT	;CARRY SET IF ERROR
		000112			
251	3160	010067	MOV	R0, RESIS	;HI WORD
		000052			
252	3164	001403	BEQ	ESI.ER	

```

253 3166 010167      MOV      R1, RESIS+2      ;LO WORD
      000046
254 3172 000207      RTS      PC
255
256 3174 000261 ESI.ER: SEC
257 3176 000207      RTS      PC
258
259 3200      130 EXP:      .ASCII/XX/          ;NOT USED ERROR
      3201      130
260 3202      060 .ASCII/00/          ;200M OHMS EXP
      3203      060
261 3204      055 .ASCII/-1/          ;20 M OHMS EXP
      3205      061
262 3206      130 .ASCII/XX/
      3207      130
263 3210      053 .ASCII/+1/          ;2 OHM EXP
      3211      061
264
265 3212 000000 EBUF:      .WORD 0,0,0,0,0
      3214 000000
      3216 000000
      3220 000000
      3222 000000
266 3224      060 ARESIS: .ASCII/0.XXXXXEXX/    ;ASCII NUMBER
      3225      056
      3226      130
      3227      130
      3230      130
      3231      130
      3232      130
      3233      105
      3234      130
      3235      130
267 3236 000000 RESIS:      .WORD 0,0          ;REAL NUMBER
      3240 000000
268      ;OVERFLOW ROUTINE INSERTS ZEROS IN BUFFER
269
270 3242 012703 OVFL0: MOV      #4, R3          ;COUNTER
      000004
271 3246 112721 OVFLX: MOVB      #60, (R1)+      ;SAVE ZERO
      000060
272 3252 005303      DEC      R3
273 3254 001374      BNE      OVFLX          ;NOT DONE
274 3256 062701      ADD      #4, R1          ;INC BUFFER POINTER
      000004
275 3262 112741      MOVB      #60, -(R1)      ;SAVE MSD AS ZERO
      000060
276 3266 000167      JMP      CONAB          ;RETURN TO ESI ROUTINE
      177652
277
278
279      ;.TITLE ASCII INPUT CONVERSION
280      ;.GLOBL INCVRT, $RCI
281      ;CALLED BY JSR PC, INCVRT
282
283      ;RTN TO CONVERT DIAM, RESIS, AND BOARD THICK
284      ;TO 2 WORD REAL NUMBERS

```

```

285          ;R0=@ OF ASCII CHAR STRING AND
286          ;R1=LENGTH UPON ENTRY
287          ;R0= HO WORD AND R1= LO WORD UPON EXIT
288          ;ASCII CHAR STRING MUST HAVE DECIMAL POINT
289          ;AND/OR E FIELD
290
291 3272 010046 INCVRT: MOV     R0,-(SP)      ;CHAR STRING
292 3274 010146      MOV     R1,-(SP)      ;LENGTH
293 3276 005046      CLR     -(SP)        ;DEC PLACES
294 3300 005046      CLR     -(SP)        ;P SCALE
295 3302 004737      JSR     PC, @#11600   ;%RCI CONVERSION
296          011600
296          ;CARRY SET FOR ERROR
297 3306 012600      MOV     (SP)+, R0      ;HO WORD
298 3310 012601      MOV     (SP)+, R1      ;LO WORD
299 3312 000207      RTS     PC
300
301          ;.TITLE CALCULATION MODULE
302          ;.GLOBL CALC, DIAM,BOARD, RESIS, RESULT
303          ;.GLOBL $IOX
304          ;FIS INTERRUPT ERROR HANDLER
305
306 3314 010046 FISERR: MOV     R0,-(SP)      ;SAVE REG FROM IOX
307 3316 010146      MOV     R1,-(SP)
308 3320 010246      MOV     R2,-(SP)
309 3322 012700      MOV     $ERRMES, R0    ;@ OF ERROR MESSAGE
310          004030
310 3326 004767      JSR     PC, $IOX
311          001314
311 3332 012602      MOV     (SP)+, R2      ;RESTORE REG
312 3334 012601      MOV     (SP)+, R1
313 3336 012600      MOV     (SP)+, R0
314 3340 106466      .WORD 106466          ;MTPS 2(SP)
315 3342 000002      .WORD 2              ; GET OLD PSW
316 3344 000000      HALT                    ; TO EXAM CC'S, REG'S
317 3346 000177      JMP     @ERRNT         ;ERROR RTN @
318          000000
318
319 3352 003772 ERRNT: .WORD ERROR
320
321          003354      .ASECT
322          000244      .=244
323 0244 003314      .WORD FISERR          ;FIS ERROR VECTOR
324 0246 000000      .WORD 0              ; PSW
325
326          003354      .=ERRNT+2          ;CONTINUE WITH CODE
327
328          ;CALLED BY JSR PC,CALC
329          ;TO COMPUTE PLATING THICKNESS
330          ; CONST = VOL RESI C COPPER/3.1416]
331          ; ARGSQ = BOARD THICK*CONST/RESIS + DIAM**2/4.0
332          ; RESULT = SQRT(ARGSQ) - DIAM/2.0
333
334          ;FIS OP CODES
335          ;ALL USE R6 AS STACK POINTER
336
337          075006 FADDR6 = 075006

```

```

338      075016 FSUBR6 = 075016
339      075026 FMULR6 = 075026
340      075036 FDIUR6 = 075036
341
342
343 3354 010667 CALC:  MOV      SP, STACK      ;SAVE STACK POINTER
          000414
344
345 3360 005001          CLR      R1              ; FOR ERROR RECOVERY
346 3362 032767          BIT      #177, RESI1    ;TEST IF RESIS IS ZERO
          000177
          177334
347 3370 001002          BNE      CTX1
348 3372 005201          INC      R1
349 3374 000404          BR       CTX2
350 3376 016746 CTX1:    MOV      RESI1+2, -(SP)
          177324
351 3402 016746          MOV      RESI1, -(SP)
          177316
352 3406 032767 CTX2:    BIT      #177, RESI2
          000177
          177314
353 3414 001002          BNE      CTX3
354 3416 005201          INC      R1
355 3420 000404          BR       CTX4
356 3422 016746 CTX3:    MOV      RESI2+2, -(SP)
          177304
357 3426 016746          MOV      RESI2, -(SP)
          177276
358 3432 032767 CTX4:    BIT      #177, RESI3
          000177
          177274
359 3440 001002          BNE      CTX5
360 3442 005201          INC      R1
361 3444 000404          BR       CTX6
362 3446 016746 CTX5:    MOV      RESI3+2, -(SP)
          177264
363 3452 016746          MOV      RESI3, -(SP)
          177256
364 3456 032767 CTX6:    BIT      #177, RESI4
          000177
          177254
365 3464 001002          BNE      CTX7
366 3466 005201          INC      R1
367 3470 000404          BR       CTX8
368 3472 016746 CTX7:    MOV      RESI4+2, -(SP)
          177244
369 3476 016746          MOV      RESI4, -(SP)
          177236
370 3502 022701 CTX8:    CMP      #3, R1
          000003
371 3506 001403          BEQ      CTXE           ;ERROR IF MORE THAN 2
372 3510 022701          CMP      #4, R1
          000004
373 3514 001002          BNE      CTX9
374 3516 000167 CTXE:    JMP      ERRDI         ;JUMP TO ERROR RTN
          000360

```

375	3522	022701	CTX9:	CMP	#0, R1	
		000000				
376	3526	001417		BEQ	CALC2	
377	3530	022701		CMP	#1, R1	
		000001				
378	3534	001404		BEQ	CALC1	
379	3536	075006		.WORD	FADDR6	;(R1+R2)
380	3540	162716		SUB	#200, (SP)	;(R1+R2)/2.0
		000200				
381	3544	000415		BR	CALC3	
382	3546	075006	CALC1:	.WORD	FADDR6	;(R1+R2)
383	3550	075006		.WORD	FADDR6	;(R1+R2+R3)
384	3552	016746		MOV	CON3+2, -(SP)	
		000230				
385	3556	016746		MOV	CON3, -(SP)	
		000222				
386	3562	075036		.WORD	FDIVR6	;(R1+R2+R3)/3.0
387	3564	000405		BR	CALC3	
388	3566	075006	CALC2:	.WORD	FADDR6	;(R1+R2)
389	3570	075006		.WORD	FADDR6	;(R1+R2+R3)
390	3572	075006		.WORD	FADDR6	;(R1+R2+R3+R4)
391	3574	162716		SUB	#400, (SP)	;(R1+R2+R3+R4)/4.0
		000400				
392	3600	016746	CALC3:	MOV	MULT+2, -(SP)	;1.0E+2
		000206				
393	3604	016746		MOV	MULT, -(SP)	
		000200				
394	3610	075026		.WORD	FMULR6	;RES X 1.0E+2
395	3612	012667		MOV	(SP)+, RESISA	;SAVE RESULTS
		000206				
396	3616	012667		MOV	(SP)+, RESISA+2	
		000204				
397	3622	004767		JSR	PC, KFACT	
		000360				
398						
399	3626	016746		MOV	KSTOR+2, -(SP)	
		000576				
400	3632	016746		MOV	KSTOR, -(SP)	
		000570				
401	3636	016746		MOV	BOARD+2, -(SP)	
		176616				
402	3642	016746		MOV	BOARD, -(SP)	
		176610				
403	3646	075026		.WORD	FMULR6	;KT
404	3650	016746		MOV	RESISA+2, -(SP)	
		000152				
405	3654	016746		MOV	RESISA, -(SP)	
		000144				
406	3660	016746		MOV	CON6+2, -(SP)	
		000104				
407	3664	016746		MOV	CON6, -(SP)	
		000076				
408	3670	075026		.WORD	FMULR6	;R TIMES 1.0E+4
409	3672	016746		MOV	DIAM+2, -(SP)	
		176470				
410	3676	016746		MOV	DIAM, -(SP)	
		176462				


```

411 3702 016746      MOV      CON4+2, -(SP)
      000052
412 3706 016746      MOV      CON4, -(SP)
      000044
413 3712 075016      .WORD    FSUBR6      ;D-.001
414 3714 075026      .WORD    FMULR6      ;R(D-.001)
415 3716 016746      MOV      CON5+2, -(SP)
      000042
416 3722 016746      MOV      CON5, -(SP)
      000034
417 3726 016746      MOV      KSTOR+2, -(SP)
      000476
418 3732 016746      MOV      KSTOR, -(SP)
      000470
419 3736 075026      .WORD    FMULR6      ;2.5K
420 3740 075016      .WORD    FSUBR6      ;R(D-.001)-2.5K
421 3742 075036      .WORD    FDIUR6      ;KT/R(D-.001)-2.5K=RESULTS
422 3744 012667      MOV      (SP)+, RESULT  ;HI ANSWER
      000050
423 3750 012667      MOV      (SP)+, RESULT+2 ;LO ANSWER
      000046
424 3754 000207      RTS      PC
425
426 3756 035603 CON4: .FLT2    .001
      3760 011157
427 3762 040440 CON5: .FLT2    2.5
      3764 000000
428 3766 043434 CON6: .FLT2    1.0E+4
      3770 040000
429
430
431 3772 012706 ERROR: MOV      (PC)+, SP      ;WILL EXIT IN MIDDLE
432 3774 000000 STACK: .WORD    0
433 3776 000207      RTS      PC      ;WITH C SET FROM FISERR RT
434
435 4000 032545 CONST: .WORD    032545, 003565 ;2.133E-7
      4002 003565
436 4004 040500 CON3: .FLT2    3.0
      4006 000000
437
438 4010 041710 MULT:  .FLT2    1.0E+2
      4012 000000
439
440      ;VOL RESIS COPPER/PI
441
442 4014 000000 ARGSQ: .WORD    0,0
      4016 000000
443
444 4020 000000 RESULT: .WORD    0,0
      4022 000000
445
446 4024 000000 RESISA: .WORD    0,0
      4026 000000
447
448 4030 000047 ERRMES: .WORD    ER1-ERO
449 4032 052 ERO:      .ASCII    /***CALCULATION ERROR...REPEAT TEST***/
      4033 052

```

4034	052		
4035	103		
4036	101		
4037	114		
4040	103		
4041	125		
4042	114		
4043	101		
4044	124		
4045	111		
4046	117		
4047	116		
4050	040		
4051	105		
4052	122		
4053	122		
4054	117		
4055	122		
4056	056		
4057	056		
4060	056		
4061	122		
4062	105		
4063	120		
4064	105		
4065	101		
4066	124		
4067	040		
4070	124		
4071	105		
4072	123		
4073	124		
4074	052		
4075	052		
4076	052		
450	4077	015	.BYTE 015,012
	4100	012	
451	4101		ER1: .EVEN
452			
453			
454			;ERROR DISPLAY DISPLAYS A FLASHING ---
455			;UNTIL A READ OR PRINT COMMAND
456			
457	4102	012706	ERRDI: MOV #1776, SP ;SET STACK
		001776	
458	4106	012737	MOV #76735, @#167772 ;DISPLAY ---
		076735	
		167772	
459	4114	004767	JSR PC, DELAY ;DELAY .2SEC
		000054	
460	4120	012737	MOV #77777, @#167772 ;BLANK DISPLAY
		077777	
		167772	
461	4126	004767	JSR PC, DELAY ;DELAY AGAIN
		000042	
462	4132	012737	MOV #100000, @#167772 ;CLEAR FLIP FLOP
		100000	

```

167772
463 4140 005037 CLR @#167772
167772
464 4144 032737 BIT #200, @#167770 ;TEST FOR PRINT
000200
167770
465 4152 001402 BEQ ERRX ;NOT PRINT
466 4154 000167 JMP RDHOX
175650
467 4160 032737 ERRX: BIT #100000, @#167770 ;TEST FOR READ
100000
167770
468 4166 001745 BEQ ERRDI ;BLINK DISPLAY AGAIN
469 4170 000167 JMP RDHOX ;RETURN TO MAIN
175634
470
471 ;DELAY GENERATES APPROX .2 SEC DELAY
472
473 4174 012700 DELAY: MOV #62560, R0
062560
474 4200 005300 DELAY: DEC R0
475 4202 001376 BNE DELAY ;NOT DONE
476 4204 000207 RTS PC ;RETURN END OF DELAY
477 ;KFACT CALCULATES K FROM KNOWN VALUES
478 ;DETERMINED FROM BOARD THICKNESS
479
480 4206 012737 KFACT: MOV #20000, @#167772
020000
167772
481 4214 013700 MOV @#167774, R0 ;GET T
167774
482 4220 005100 COM R0
483 4222 006200 ASR R0
484 4224 006200 ASR R0
485 4226 006200 ASR R0
486 4230 006200 ASR R0
487 4232 042700 BIC #177760, R0
177760
488 4236 012702 MOV #5, R2 ;# VALUES IN TABLE
000005
489 4242 012701 MOV #TABLK, R1 ;LOCATION OF TABLE
004344
490 4246 021100 KFACT2: CMP (R1), R0 ;COMPARE TABLE WITH DATA
491 4250 001406 BEQ KFACT3 ;BRANCH IF SAME
492 4252 062701 ADD #10., R1 ;GOTO NEXT VALUE
000012
493 4256 005302 DEC R2
494 4260 001372 BNE KFACT2 ;CONTINUE
495 4262 000261 SEC ;ERROR
496 4264 000207 RTS PC
497 4266 062701 KFACT3: ADD #4, R1
000004
498 4272 011146 MOV (R1), -(SP) ;B CONSTANT
499 4274 162701 SUB #2, R1
000002
500 4300 011146 MOV (R1), -(SP)
501 4302 062701 ADD #6, R1

```

```

000006
502 4306 011146      MOV      (R1), -(SP)      ;A CONSTANT
503 4310 162701      SUB       #2, R1
000002
504 4314 011146      MOV      (R1), -(SP)
505 4316 016746      MOV      DIAM+2, -(SP) ;DIAM
176044
506 4322 016746      MOV      DIAM, -(SP)
176036
507 4326 075026      .WORD    FMULR6          ;A TIMES DIAM
508 4330 075006      .WORD    FADDR6          ;A TIMES DIAM + B = K
509 4332 012667      MOV      (SP)+, KSTOR      ;SAVE K
000070
510 4336 012667      MOV      (SP)+, KSTOR+2 ;
000066
511 4342 000207      RTS       PC
512
513 4344 000001 TABLK: .WORD    1
514 4346 037361      .FLT2    .11809, -1.02797      ; B & A
4350 154453
4352 140203
4354 112205
515 4356 000003      .WORD    3
516 4360 037361      .FLT2    .11809, -1.02797      ; B & A
4362 154453
4364 140203
4366 112205
517 4370 000006      .WORD    6
518 4372 037601      .FLT2    .25234, -2.08183      ; B & A
4374 031265
4376 140405
4400 036264
519 4402 000004      .WORD    4
520 4404 037521      .FLT2    .20461, -1.66244      ; B & A
4406 102511
4410 140324
4412 145325
521 4414 000002      .WORD    2
522 4416 037521      .FLT2    .20461, -1.66244      ; B & A
4420 102511
4422 140324
4424 145325
523
524 4426 000000 KSTOR: .WORD    0,0      ; K FACTOR STORAGE
4430 000000
525
526 4432 000207      RTS       PC
527
528      ;.TITLEPRINT SETUP
529      ;.GLOBLE PRINT, $ECO, $FCO, OUTBUF
530      ;.GLOBLE DIAM,BOARD,RESIS,RESULT
531
532
533
534      ;SET UP ARG ON STACK
535
536 4434 010046 FCVRT: MOV      R0, -(SP)      ;BUF @

```

537	4436	012746	MOV	#7, -(SP)	#BUF LEN
		000007			
538	4442	012746	MOV	#4, -(SP)	#DEC PLACES
		000004			
539	4446	005046	CLR	-(SP)	#P SCALE
540	4450	016146	MOV	2(R1), -(SP)	#LO BIN NUM
		000002			
541	4454	011146	MOV	(R1), -(SP)	#HI BIN NUM
542	4456	004737	JSR	PC, @#13206	#FCO
		013206			
543	4462	000207	RTS	PC	
544					
545	4464	010046	ECVRT: MOV	R0, -(SP)	#BUF @
546	4466	012746	MOV	#12, -(SP)	#LEN
		000014			
547	4472	012746	MOV	#5, -(SP)	#DEC PLACES
		000005			
548	4476	005046	CLR	-(SP)	#P SCALE
549	4500	016146	MOV	2(R1), -(SP)	#LO BIN NUM
		000002			
550	4504	011146	MOV	(R1), -(SP)	#HI BIN NUM
551	4506	004737	JSR	PC, @#13242	#ECO
		013242			
552	4512	000207	RTS	PC	
553					
554				#IO BLOCKES	
555					
556	4514	000007	RESOUT: .WORD	T2-T1	#LENGTH
557	4516	004525	T1:	.=.+7	
558	4525		T2:	.EVEN	
559	4526	000007	ANSOUT: .WORD	Z2-Z1	#LENGTH
560	4530	004537	Z1:	.=.+7	
561	4537		Z2:	.EVEN	
562	4540	000103	OUTBUF: .WORD	M2-M1	#LENGTH
563	4542	015	M1:	.BYTE CR,LF	
	4543	012			
564	4544	122		.ASCII /R1=/	
	4545	061			
	4546	075			
565	4547	004563	RES1: .=.+12.		#+0.0000E-00 4 TIMES
566	4563	040		.ASCII / R2=	
	4564	122			
	4565	062			
	4566	075			
567	4567	004603	RES2: .=.+12.		
568	4603	040		.ASCII / R3=	
	4604	122			
	4605	063			
	4606	075			
569	4607	004623	RES3: .=.+12.		
570	4623	040		.ASCII / R4=	
	4624	122			
	4625	064			
	4626	075			
571	4627	004643	RES4: .=.+12.		
572	4643	015		.BYTE CR,LF	
	4644	012			

```

573 4645      M2:      .EVEN
574
575           ;.TITLE $IOX
576           ;.GLOBL $IOX, $IOWAT
577
578           ;$IOX PRINTS MESSAGE TO TELETYPE
579           ;R0 = @ OF BUF
580           ;SAVES NO RESISTERS
581           ;CALLED BY JSR PC, $IOX
582
583 4646 012001 $IOX:  MOV      (R0)+, R1      ;BYTE COUNT
584 4650 112002 $LOOP: MOVB     (R0)+, R2      ;GET CHAR
585 4652 004767      JSR      PC, $OUT        ;PRINT IT
586           000006
587 4656 005301      DEC      R1              ;DONE
588 4660 001373      BNE     $LOOP            ;BIF NO
589 4662 000207      RTS      PC
590 4664 010237 $OUT:  MOV      R2,@#177566
591 4670 105737 $OUT1: TSTB     @#177564      ;DONE ?
592 4674 100375      BPL     $OUT1            ;BIF NO
593 4676 000207      RTS      PC
594
595           ; SAVE RTN SAVES DATA FOR LATER PRINTOUT
596
597 4700 016724 SAVE:  MOV      ASAVH0, (R4)+   ;SAVE HOLE #
598           175370
599 4704 016724      MOV      ASAVH0+2, (R4)+
600           175366
601 4710 016724      MOV      ADIAM+2, (R4)+   ;SAVE DIAM
602           175444
603 4714 016724      MOV      ADIAM+4, (R4)+
604           175442
605 4720 016724      MOV      ASAVBT+2, (R4)+ ;SAVE BOARD THICKNESS
606           175526
607 4724 016724      MOV      ASAVBT+4, (R4)+
608           175524
609 4730 016724      MOV      RESISA, (R4)+   ;SAVE AVG. RESIS
610           177070
611 4734 016724      MOV      RESISA+2, (R4)+
612           177066
613 4740 016724      MOV      DRESL, (R4)+    ;SAVE PLAT THICKNESS
614           000770
615 4744 016724      MOV      DRESL+2, (R4)+
616           000766
617 4750 016724      MOV      RESI1, (R4)+
618           175750
619 4754 016724      MOV      RESI1+2, (R4)+
620           175746
621 4760 016724      MOV      RESI2, (R4)+
622           175744
623 4764 016724      MOV      RESI2+2, (R4)+
624           175742
625 4770 016724      MOV      RESI3, (R4)+
626           175740

```

612	4774	016724	MOV	RESI3+2, (R4)+	
		175736			
613	5000	016724	MOV	RESI4, (R4)+	
		175734			
614	5004	016724	MOV	RESI4+2, (R4)+	
		175732			
615	5010	000207	RTS	PC	
616					
617					
618	5012	012724	PRINT1: MOV	#377, (R4)+	;END OF FILE
		000377			
619	5016	012700	MOV	#MES.ST, R0	;OUTPUT ID# MESSAGE
		002122			
620	5022	004767	JSR	PC, #IOX	;PRINT IT
		177620			
621	5026	004767	JSR	PC, SAVSN	;SAV S/N
		175166			
622	5032	004767	JSR	PC, SAVID	;SAV ID #
		175104			
623	5036	012704	MOV	#1000, R4	;GET START OF FILE
		001000			
624	5042	012701	MOV	#3, R1	;BYTE COUNT
		000003			
625	5046	012700	MOV	#AIDEN2+4, R0	
		002220			
626	5052	004767	JSR	PC, PRINTX	;OUTPUT
		000426			
627	5056	012701	MOV	#3, R1	
		000003			
628	5062	012700	MOV	#AIDENT+4, R0	
		002214			
629	5066	004767	JSR	PC, PRINTX	;OUTPUT 3 MORE NUMBERS
		000412			
630	5072	012702	MOV	#SA, R2	
		000040			
631	5076	004767	JSR	PC, #OUT	
		177562			
632	5102	012700	PRINT2: MOV	#ASAVSN+4, R0	
		002250			
633	5106	012701	MOV	#3, R1	
		000003			
634	5112	004767	JSR	PC, PRINTX	
		000366			
635					
636	5116	012700	MOV	#MES.HD, R0	;OUT HEADING
		005534			
637	5122	004767	JSR	PC, #IOX	
		177520			
638	5126	012701	PRINT4: MOV	#3, R1	;BYTE COUNT
		000003			
639	5132	062704	PRINT5: ADD	#4, R4	
		000004			
640	5136	010400	MOV	R4, R0	
641	5140	004767	JSR	PC, PRINTX	;OUTPUT HOLE #
		000340			
642	5144	012702	MOV	#SA, R2	
		000040			

643	5150	004767	JSR	PC, \$OUT	
		177510			
644	5154	004767	JSR	PC, \$OUT	
		177504			
645	5160	012701	MOV	\$3, R1	
		000003			
646	5164	004767	JSR	PC, PRINTZ	\$OUTPUT DIAM
		000330			
647	5170	012702	MOV	\$SA, R2	
		000040			
648	5174	004767	JSR	PC, \$OUT	
		177464			
649	5200	004767	JSR	PC, \$OUT	
		177460			
650	5204	012701	MOV	\$3, R1	
		000003			
651	5210	005204	INC	R4	
652	5212	010400	MOV	R4, R0	
653	5214	004767	JSR	PC, PRINTZ	\$OUTPUT T
		000300			
654	5220	012702	MOV	\$SA, R2	
		000040			
655	5224	004767	JSR	PC, \$OUT	
		177434			
656	5230	012700	MOV	\$RESOUT, R0	\$RESIS FIELD
		004514			
657	5234	005204	INC	R4	
658	5236	010401	MOV	R4, R1	
659	5240	004767	JSR	PC, FCVRT	
		177170			
660	5244	012700	MOV	\$RESOUT+4, R0	
		004520			
661	5250	012701	MOV	\$4, R1	
		000004			
662	5254	112002	MOVB	(R0)+, R2	
663	5256	004767	JSR	PC, \$OUT	
		177402			
664	5262	005301	DEC	R1	
665	5264	001373	BNE	RESX	
666	5266	012702	MOV	\$SA, R2	
		000040			
667	5272	004767	JSR	PC, \$OUT	
		177366			
668	5276	004767	JSR	PC, \$OUT	
		177362			
669	5302	062704	ADD	\$4, R4	
		000004			
670	5306	012700	MOV	\$ANSOUT, R0	
		004526			
671	5312	010401	MOV	R4, R1	
672	5314	004767	JSR	PC, FCVRT	
		177114			
673	5320	012700	MOV	\$ANSOUT+3, R0	
		004531			
674	5324	112002	MOVB	(R0)+, R2	
675	5326	004767	JSR	PC, \$OUT	
		177332			

676	5332	012702	MOV	#56, R2
		000056		
677	5336	004767	JSR	PC, \$OUT
		177322		
678	5342	112002	MOVB	(R0)+, R2
679	5344	004767	JSR	PC, \$OUT
		177314		
680	5350	112002	MOVB	(R0)+, R2
681	5352	004767	JSR	PC, \$OUT
		177306		
682	5356	062704	ADD	#4, R4
		000004		
683	5362	012700	MOV	#RES1, R0
		004547		
684	5366	010401	MOV	R4, R1
685	5370	004767	JSR	PC, ECVRT
		177070		
686	5374	062704	ADD	#4, R4
		000004		
687	5400	012700	MOV	#RES2, R0
		004567		
688	5404	010401	MOV	R4, R1
689	5406	004767	JSR	PC, ECVRT
		177052		
690	5412	062704	ADD	#4, R4
		000004		
691	5416	012700	MOV	#RES3, R0
		004607		
692	5422	010401	MOV	R4, R1
693	5424	004767	JSR	PC, ECVRT
		177034		
694	5430	062704	ADD	#4, R4
		000004		
695	5434	012700	MOV	#RES4, R0
		004627		
696	5440	010401	MOV	R4, R1
697	5442	004767	JSR	PC, ECVRT
		177016		
698	5446	062704	ADD	#4, R4
		000004		
699	5452	012702	MOV	#CR, R2
		000015		
700	5456	004767	JSR	PC, \$OUT
		177202		
701	5462	012702	MOV	#LF, R2
		000012		
702	5466	004767	JSR	PC, \$OUT
		177172		
703	5472	122714	CMPB	#377, (R4)
		000377		
704	5476	001213	BNE	PRINT4
705	5500	000167	JMP	START
		174274		
706	5504	114002	PRINTX: MOVB	-(R0), R2
707	5506	004767	JSR	PC, \$OUT
		177152		
708	5512	005301	DEC	R1

```

709 5514 001373      BNE    PRINTX
710 5516 000207      RTS     PC
711 5520 112402 PRINTZ: MOVB   (R4)+, R2
712 5522 004767      JSR     PC, $OUT
      177136
713 5526 005301      DEC     R1
714 5530 001373      BNE     PRINTZ
715 5532 000207      RTS     PC
716
717
718 5534 000040 MES.HD: .WORD HD2-HD1
719 5536 015 HD1:    .BYTE  CR,LF
720 5540 110        .ASCII /HOLE DIAM T RES THICKNESS/
      5541 117
      5542 114
      5543 105
      5544 040
      5545 104
      5546 111
      5547 101
      5550 115
      5551 040
      5552 040
      5553 124
      5554 040
      5555 040
      5556 122
      5557 105
      5560 123
      5561 040
      5562 040
      5563 124
      5564 110
      5565 111
      5566 103
      5567 113
      5570 116
      5571 105
      5572 123
      5573 123
721 5574 015        .BYTE  CR,LF
722 5575 012
723 5576          HD2:    .EVEN
724
725          ;DISPLAY RTN
726          ;TAKES NUM IN RESULT AND DISPLAYS ON LED'S
727
728 5576 016746 DISPLA: MOV     RESULT+2, -(SP)
      176220
729 5602 016746      MOV     RESULT, -(SP)
      176212
730 5606 016746      MOV     MULT+2, -(SP)
      176200
731 5612 016746      MOV     MULT, -(SP)
      176172

```

732	5616	075026	.WORD	FMULR6		
733	5620	012667	MOV	(SP)+, DRESL		;CHANGE ANSWER TO MILLS
		000110				
734	5624	012667	MOV	(SP)+, DRESL+2		
		000106				
735	5630	012700	MOV	#ADIS, R0		
		005740				
736	5634	012701	MOV	#DRESL, R1		
		005734				
737	5640	004767	JSR	PC, FCVRT		
		176570				
738	5644	012700	MOV	#ADIS+2, R0		
		005742				
739	5650	005200	INC	R0		
740	5652	112002	MOVB	(R0)+, R2		
741	5654	042702	BIC	#177760, R2		
		177760				
742	5660	010203	MOV	R2, R3		
743	5662	112002	MOVB	(R0)+, R2		
744	5664	042702	BIC	#177760, R2		
		177760				
745	5670	006303	ASL	R3		
746	5672	006303	ASL	R3		
747	5674	006303	ASL	R3		
748	5676	006303	ASL	R3		
749	5700	060203	ADD	R2, R3		
750	5702	112002	MOVB	(R0)+, R2		
751	5704	042702	BIC	#177760, R2		
		177760				
752	5710	006303	ASL	R3		
753	5712	006303	ASL	R3		
754	5714	006303	ASL	R3		
755	5716	006303	ASL	R3		
756	5720	060203	ADD	R2, R3		
757	5722	062703	ADD	#70000, R3		
		070000				
758	5726	010337	MOV	R3, @#167772		;DISPLAY RESULTS ON LED'S
		167772				
759	5732	000207	RTS	PC		
760	5734	000000	DRESL:	.WORD 0,0		
		5736	000000			
761						
762	5740	000007	ADIS:	.WORD X2-X1		;LENGTH
763	5742	005751	X1:	.=.+7		
764	5751		X2:	.EVEN		
765						
766						
767						
768		002000'	.END	START		

.MAIN. MACRO V06-03 22-JUL-77 11:48 PAGE 1-22
SYMBOL TABLE

ADIAH	002356	ADIS	005740	AIDENT	002210
AIDEN2	002214	ANEXT	002544	ANSOUT	004526
ARESI5	003224	ARGSQ	004014	ASAVBT	002450
ASAVHO	002274	ASAVSN	002244	ASCII	002536
BOARD	002456	CALC	003354	CALC1	003546
CALC2	003566	CALC3	003600	CONAB	003144
CONST	004000	CON3	004004	CON4	003756
CON5	003762	CON6	003766	CR	= 000015
CTXE	003516	CTX1	003376	CTX2	003406
CTX3	003422	CTX4	003432	CTX5	003446
CTX6	003456	CTX7	003472	CTX8	003502
CTX9	003522	DELAX	004200	DELAY	004174
DIAH	002364	DISPLA	005576	DRESL	005734
EBUF	003212	ECVRT	004464	ERO	004032
ERRDI	004102	ERRMES	004030	ERRNT	003352
ERRDR	003772	ERRX	004160	ER1	004101
ESI	003010	ESI.ER	003174	ESI1	002754
ESI2	002770	EXP	003200	FADDR6=	075006
FCVRT	004434	FDIVR6=	075036	FISERR	003314
FMULR6=	075026	FOUR	002562	FSUBR6=	075016
HD1	005536	HD2	005576	INCVRT	003272
INVER	002462	INVER2	002472	INVER3	002514
KFACT	004206	KFACT2	004246	KFACT3	004266
KSTOR	004426	LF	= 000012	MEAS	002572
MES.HD	005534	MES.ST	002122	MULT	004010
M1	004542	M2	004645	NEXT.D	003076
N.ESI	003020	OUTBUF	004540	DVFLO	003242
OVFLX	003246	PRINTX	005504	PRINTZ	005520
PRINT1	005012	PRINT2	005102	PRINT4	005126
PRINT5	005132	RDHOLE	002010	RDHOL1	002044
RDHOX	002030	RELAY	002744	RESIS	003236
RESISA	004024	RESI1	002724	RESI2	002730
RESI3	002734	RESI4	002740	RESOUT	004514
RESULT	004020	RESX	005254	RES1	004547
RES2	004567	RES3	004607	RES4	004627
SA	= 000040	SAVBT	002374	SAVE	004700
SAVHO	002250	SAVID	002142	SAVSN	002220
STACK	003774	START	002000	ST1	002124
ST2	002141	TABLK	004344	THUMB	002300
T.ERR	002370	T1	004516	T2	004525
V003	002000	X1	005742	X2	005751
Z1	004530	Z2	004537	\$IOX	004646
\$LOOP	004650	\$OUT	004664	\$OUT1	004670

. ABS. 005752 000
000000 001

ERRORS DETECTED: 0

FREE CORE: 12231. WORDS

PCHOLE.OBJ,LP:/LI<PCHOLE.MAC

Appendix D
CONVERSION ROUTINES

ASCII TO REAL CONVERSION ROUTINES

SRCI, SDCI ROUTINES

. = 11600 / 11600 TO 15014

R0=X0
R1=X1
R2=X2
R3=X3
R4=X4
R5=X5
SP=X6
PC=X7

```

NUMEND=0
POINTL=2
DIGITS=4
BEXP=6
ESTGN=R.
SIGN=10.
EEYP=12.
P=30.
D=32.
ERF=26.
LENGTH=34.
TEMP=LENGTH
RESULT=P
START=36.
END=START
SRCI:  CLR      -(SP)
      INC      #SP
      BR      CNVS8
SDCI:  CLR      -(SP)
CNVS8:  MOV      R0,-(SP)
      MOV      R1,-(SP)
      MOV      R2,-(SP)
      MOV      R3,-(SP)
      MOV      R4,-(SP)
      MOV      R5,-(SP)
      CLR      -(SP)
      CLR      -(SP)
      CLR      -(SP)
      MOV      #65.,-(SP)
      MOV      #18.,-(SP)
      CLR      -(SP)
      CLR      -(SP)
      MOV      START(SP),R5
      ADD      LENGTH(SP),END(SP)
      CLR      R0
      CLR      R1
      CLR      R2
      CLR      R3
SCN88:  MOV8     (R5)+,R4
      BIC      #177600,R4
      CMPB     R4,#1
      BNE      SG388
      CMP      R5,START(SP)
      RLT      SCN88
      JMP      ZER88
SG388:  CMPB     R4,#1+
```

	BEQ	FLD38
	CMPS	R4, #1
	BNE	NCK38
	INC	SIGN(SP)
	BR	FLD38
NXT38:	MOVB	(R5)+, R4
	9IC	#177600, R4
	CMPS	R4, #1
	BNE	NCK38
	MOV	#10, R4
NCK38:	CMPS	R4, #10
	BLT	PCK38
	BNE	NNZ38
	TST	R0
	BNE	NNZ38
	TST	R1
	BNE	NNZ38
	TST	R2
	BNE	NNZ38
	TST	R3
	BEQ	FLD38
NNZ38:	CMPS	R4, #19
	BGT	EXC38
	DEC	DIGITS(SP)
	BGE	A2I38
	INC	EEXP(SP)
	BR	FLD38
A2I38:	SUB	#60, R4
	JSR	PC, M1538
	JSR	PC, LFT38
	ADD	R4, R3
	ADC	R2
	ADC	R1
	ADC	R0
FLD38:	CMP	R5, END(SP)
	BLT	NXT38
	MOV	R5, #SP
SCL38:	TST	R0
	BNE	SC138
	TST	R1
	BNE	SC138
	TST	R2
	BNE	SC138
	TST	R3
	BEQ	ZER38
SC138:	CMP	#SP, R5
	BNE	NOP38
	SUB	P(SP), EEXP(SP)
NOP38:	TST	POINTL(SP)
	BNE	PNT38
	MOV	O(SP), #SP
PNT38:	SUB	POINTL(SP), #SP
	SUB	#SP, EEXP(SP)
	SGT	MUL38
	BLT	DIV38
	JMP	FLT38
MUL38:	CMP	R0, #31462
	BHI	MOV38
	JSR	PC, M1538
	INC	BEXP(SP)
D1038:	DEC	EEXP(SP)
	BGT	MUL38
	JMP	FLT38
MOV38:	JSR	PC, M5438
	ADD	#3, BEXP(SP)
	BR	D1038

```

PCK$8:  CMPB    R4,#1,
        BNE     ERR$8
PTF$8:  TST     POINTL(SP)
        BNE     ERR$8
        MOV     R5,POINTL(SP)
        RR      FLD$8
ERR$8:  COMB    ERF+1(SP)
ZER$8:  CLR     R0
        CLR     R1
        CLR     R2
        CLR     R3
        JMP     STR$8
EXC$8:  CMPB    R4,#1E
        BEQ     EXT$8
        CMPB    R4,#1D
        BNE     ERR$8
EXT$8:  MOV     R5,#SP
        DEC     #SP
        MOV     R3,TEMP(SP)
        CLR     R3
        CMP     R5,END(SP)
        AGE     ERR$8
        MOVB    (R5)+,R4
        BIC     #177600,R4
        CMPB    R4,#1+
        BEQ     EF1$8
        CMPB    R4,#1-
        BNE     ENM$8
        INC     ESIGN(SP)
EF1$8:  CMP     R5,END(SP)
        BGE     ERR$8
EF2$8:  MOVB    (R5)+,R4
        BIC     #177600,R4
ENM$8:  CMPB    R4,#1
        BNE     EN1$8
        MOV     #10,R4
EN1$8:  CMPB    R4,#10
        BLT     ERR$8
        CMPB    R4,#19
        BGT     ERR$8
        SUB     #60,R4
        ASL     R3
        ADD     R3,R4
        ASL     R3
        ASL     R3
        ADD     R4,R3
        CMP     R5,END(SP)
        BLT     EF2$8
        TST     ESIGN(SP)
        BEQ     EN2$8
        NEG     R3
EN2$8:  ADD     R3,EEXP(SP)
        MOV     TEMP(SP),R3
        JMP     SCL$8
DIV$8:  TST     R0
        BLT     DV1$8
DV2$8:  DEC     BEXP(SP)
        JSR     PC,LFT$8
        BPL     DV2$8
DV1$8:  MOV     #16,,R4
        JSR     PC,RIT$8
        MOV     R3,-(SP)
        MOV     R2,-(SP)
        MOV     R1,-(SP)
        MOV     R0,-(SP)
DV3$8:  JSR     PC,RIT$8

```



```

      CLC
      JSR      PC,RITS8
      MOV      #2,R5
      CLC
DV458: JSR      PC,RITS8
      ADD      6(SP),R3
      ADC      R2
      ADC      R1
      ADC      R0
      ADD      4(SP),R2
      ADC      R1
      ADC      R0
      ADD      2(SP),R1
      ADC      R0
      ADD      #SP,R0
      DEC      R5
      BGT      OV458
      DEC      R4
      BGT      OV358
      ADD      #8,,SP
      SUB      #3,BEXP(SP)
      INC      EEXP(SP)
      BLT      DIV58
      FLT58: DEC      BEXP(SP)
      JSR      PC,FLT58
      BCC      FLT38
      ADD      #200,BEXP(SP)
      BLE      UND58
      CMP      BEXP(SP),#377
      BGT      OVR58
      CLRB      R3
      BISB      R2,R3
      SWAB      R3
      CLRB      R2
      BISB      R1,R2
      SWAB      R2
      CLRB      R1
      BISB      R0,R1
      SWAB      R1
      CLRB      R0
      BISB      REXP(SP),R0
      SWAB      R0
      ROR      SIGN(SP)
      JSR      PC,RITS8
      ADC      R3
      ADC      R2
      ADC      R1
      ADC      R0
      BVS      OVR58
      BCS      OVR58
STR38: TSTB      ERF(SP)
      BEQ      DPR58
      ROL      R2
      ADC      R1
      ADC      R0
      BVS      OVR58
      BCS      OVR58
      MOV      R0,R2
      MOV      R1,R3
DPR58: MOV      R0,RESULT(SP)
      MOV      R1,RESULT+2(SP)
      MOV      R2,RESULT+4(SP)
      MOV      R3,RESULT+6(SP)
      ADD      #14,,SP
      MOV      (SP)+,R5
      MOV      (SP)+,R4

```

```

MOV      (SP)+,R3
MOV      (SP)+,R2
MOV      (SP)+,R1
MOV      (SP)+,R0
TSTB     #SP
BEQ      RRNSB
MOV      (SP)+,2(SP)
MOV      (SP)+,2(SP)
RRNSB:   ROL      (SP)+
RTS      PC

OVRB:
UNDB:   JMP      ERRB

M54B:   CMP      R0,#146314
BLO      M55B
CLC
JSR      PC,RITB
INC      REXP+2(SP)
M55B:   MOV      R0,-(SP)
MOV      R1,-(SP)
MOV      R2,-(SP)
MOV      R3,-(SP)
CLC
JSR      PC,RITB
CLC
JSP      PC,RITB
BR      M5AB
ML5B:   MOV      R0,-(SP)
MOV      R1,-(SP)
MOV      R2,-(SP)
MOV      R3,-(SP)
JSR      PC,LFTB
JSR      PC,LFTB
M5AB:   ADD      (SP)+,R3
ADC      R2
ADC      R1
ADC      R0
ADD      (SP)+,R2
ADC      R1
ADC      R0
ADD      (SP)+,R1
ADC      R0
ADD      (SP)+,R0
RTS      PC
LFTB:   ASL      R3
ROL      R2
ROL      R1
ROL      R0
RTS      PC
RITB:   ROR      R0
ROR      R1
ROR      R2
ROR      R3
RTS      PC

```

ISECO,IFCO,SGCO,SDCO ROUTINES

```

POINT=2
REXP=4
EEXP=6
TYPE=12.
P=16.
D=18.
L=20.
S=22.
SGCO:  MOV    #42403,R0
        BR     XC039
IFCO:  CLR     R0
        BR     XC039
SDCO:  MOV     (SP)+,R0
        MOV     (SP)+,R1
        MOV     (SP)+,R2
        MOV     #SP,R3
        MOV     #42002,#SP
        MOV     R4,-(SP)
        MOV     4(SP),R4
        MOV     R0,4(SP)
        BR     XC139
ISECO: MOV     #42402,R0
XC039: MOV     (SP)+,R3
        MOV     (SP)+,R1
        MOV     (SP)+,R2
        MOV     R3,-(SP)
        MOV     R0,-(SP)
        CLR     R3
        MOV     R4,-(SP)
        CLR     R4
XC139: MOV     R5,-(SP)
        CLR     -(SP)
        CLR     -(SP)
        CMP     -(SP),-(SP)
        ADD     S(SP),L(SP)
        MOV     S(SP),R0
CLES9: MOV8    #1,(R0)+
        CMP     R0,L(SP)
        BLO     CLES9
        ROL     R1
        ROL     #SP
        SWAB    R1
        MOV8    R1,BEXP(SP)
        BNE     NNZS9
        CLR     R0
        BR      NODS9
NNZS9: SEC
        ROR     R1
        CLR9    R1
        SWAB    R2,R1
        BIS8    R2,R1
        CLR9    R2
        SWAB    R3
        BIS8    R3,R2
        CLR9    R3
        SWAB    R4

```

	R155	R4,R3
	CLR3	R4
	SUR	#200,BEXP(SP)
	RLT	DIV59
	BEQ	NOM59
MUL59:	TST	R1
	RLT	ML159
	ASL	R4
	ROL	R3
	ROL	R2
	ROL	R1
	DEC	REXP(SP)
	BGT	MUL59
	BR	NOM59
ML159:	JSR	PC,M4559
	INC	EEXP(SP)
	SUB	#3,BEXP(SP)
	BGT	MUL59
	BEQ	NOM59
DIV59:	CMP	R1,#146314
	BHIS	DV159
	CMP	BEXP(SP),#-3
	BGT	DV159
	JSR	PC,M5459
	DEC	EEXP(SP)
	ADD	#2,BEXP(SP)
	BR	DV259
DV159:	JSR	PC,RIT59
DV259:	INC	REXP(SP)
	BNE	DIV59
NOM59:	CLR	R0
NO159:	JSR	PC,M5459
	JSR	PC,ML859
	TST	R0
	BNE	NOD59
	DEC	EEXP(SP)
	BR	NO159
NOD59:	TSTB	TYPE(SP)
	REQ	FFT59
	RORB	TYPE(SP)
	ACC	FFT59
	TST	EEXP(SP)
	RLT	FFT59
	CMP	EEXP(SP),D(SP)
	BGT	FFT59
	CLRB	TYPE(SP)
	SUR	#4,L(SP)
	SUB	EEXP(SP),D(SP)
	CLR	P(SP)
FFT59:	MOV	EEXP(SP),R5
FF59:	ADD	D(SP),R5
	ADD	P(SP),R5
	JSR	PC,RUD59
	MOV	L(SP),R5
	SUB	O(SP),R5
	TSTB	TYPE(SP)
	BNE	FF59
	ADD	EEXP(SP),P(SP)
	BLE	FF59
	SUR	P(SP),R5
	SUB	#2,R5
	JSR	PC,ISN59
	BR	FF59

```

FF559:  SUB    #3,R5
        JSP    PC,ISN59
        MOV8   #10,(R5)+
        MOV9   #1,(R5)+
FF459:  CMP    R5,L(SP)
        BHTS   FF359
        MOV8   #10,(R5)+
        BR     FF459
FF359:  MOV    L(SP),R5
        SUB    D(SP),R5
        DEC    R5
        MOV    R5,POINT(SP)
        TST    P(SP)
        BGT    FF659
        INC    R5
FF659:  SUB    P(SP),R5
        JSR    PC,DGS59
        TSTB   TYPE(SP)
        BEQ    DNE59
        BR     EFT59
EFT59:  SUB    #4,L(SP)
        CLR    R5
        TST    P(SP)
        BLE    FFE59
        MOV    D(SP),R5
        ADD    P(SP),R5
        JSR    PC,RUD59
        MOV    L(SP),R5
        SUB    D(SP),R5
        DEC    R5
        MOV    R5,POINT(SP)
        SUB    P(SP),R5
        DEC    R5
        JSR    PC,ISN59
        JSR    PC,DGS59
EFE59:  SUB    P(SP),EEXP(SP)
        MOV    L(SP),R3
        MOV8   TYPE+1(SP),(R3)+
        MOV    EEXP(SP),R4
        BGE    EXP59
        NEG    R4
        MOV9   #1-,(R3)+
        BR     EX159
EXP59:  MOV8   #1,(R3)+
EX159:  MOV8   #10,R3
EX359:  SUB    #10,R4
        BLT    EX259
        INCB   #R3
        BR     EX359
EX259:  ADD    #72,R4
        MOV8   R4,1(R3)
DNE59:  ADD    #8,SP
        MOV    (SP)+,R5
        MOV    (SP)+,R4
        MOV    (SP)+,6(SP)
        MOV    (SP)+,6(SP)
        CMP    (SP)+,(SP)+
        ROL    (SP)+
        RTS    PC

M5459:  MOV    R1,-(SP)
        MOV    R2,-(SP)
        MOV    R3,-(SP)
        MOV    R4,-(SP)

```

```

        JSR      PC,RIT$9
        JSR      PC,RIT$9
        ADC      R4
        ADC      R3
        ADC      R2
        ADC      R1
        ADD      (SP)+,R4
        ADC      R3
        ADC      R2
        ADC      R1
        ADC      R0
        ADD      (SP)+,R3
        ADC      R2
        ADC      R1
        ADC      R0
        ADD      (SP)+,R2
        ADC      R1
        ADC      R0
        ADD      (SP)+,R1
        ADC      R0
        RTS      PC

M45$9:  MOV      #16.,R5
        JSR      PC,RIT$9
        MOV      R4,-(SP)
        MOV      R3,-(SP)
        MOV      R2,-(SP)
        MOV      R1,-(SP)
M51$9:  JSR      PC,RIT$9
        JSR      PC,RIT$9
        MOV      #2,R0
M52$9:  JSR      PC,RIT$9
        ADD      6(SP),R4
        ADC      R3
        ADC      R2
        ADC      R1
        ADD      4(SP),R3
        ADC      R2
        ADC      R1
        ADD      2(SP),R2
        ADC      R1
        ADD      #SP,R1
        DEC      R0
        BGT      M52$9
        DEC      R5
        BGT      M51$9
        ADD      #8.,SP
        RTS      PC

ML8$9:  MOV      R5,-(SP)
        MOV      #3,R5
M81$9:  ASL      R4
        ROL      R3
        ROL      R2
        ROL      R1
        ROL      R0
        DEC      R5
        BGT      M81$9
        MOV      (SP)+,R5
        RTS      PC
ERR$9:  TST      (SP)+
        MOV      3(SP),R3
        MOV      L(SP),R4

```

```

        TSTB      TYPE(SP)
        BEQ       STS39
        ADD       #4,R4
STS39:  MOV8      #1*,(R3)+
        CMP       R3,R4
        BLO      STS39
        COM       TYPE(SP)
        BR        DNE39

RU039:  CMP       R5,#20.
        BGT       RU139
        MOV       R5,BEXP+0+2(SP)
        BEQ       RU339
        BLT       RU139
        MOV       R0,-(SP)
        MOV       R1,-(SP)
        MOV       R2,-(SP)
        MOV       R3,-(SP)
        MOV       R4,-(SP)
        MOV       #100000,R1
        CLR       R2
        CLR       R3
        CLR       R4
RDF39:  DEC       BEXP+2+2+10.(SP)
        BEQ       RDD39
        JSR       PC,M4539
        JSR       PC,RIT39
        JSR       PC,RIT39
        JSR       PC,RIT39
        BR        RDF39
RDD39:  CLR       R0
        ADD       (SP)+,R4
        ADC       R3
        ADC       R2
        ADC       R1
        ADD       (SP)+,R3
        ADC       R2
        ADC       R1
        ADD       (SP)+,R2
        ADC       R1
        ADD       (SP)+,R1
        ADC       R0
        ADD       (SP)+,R0
RU239:  CMP       #10.,R0
        BGT       RU139
        INC       BEXP+2(SP)
RU139:  RTS       PC
RU339:  ADD       #5,R0
        BR        RU239

ISN39:  CMP       R5,3-0+2(SP)
        BLO      SPC39
        ROR       0+2(SP)
        RCC       ISR39
        MOV8      #1-,R5
ISR39:  INC       R5
        RTS       PC
SPC39:  ROR       0+2(SP)
        BCS       ERR39
        INC       R5

```

```

        CMP      R5,S+2(SP)
        BLO      ERR$9
        RTS      PC

DG$9:   CMP      #10.,R0
        BGT      DG1$9
        MOV8     #1,(R5)+
        SUB      #10.,R0
DG1$9:  CMP      POINT+2(SP),R5
        BNE      DG2$9
        MOV8     #1.,(R5)+
DG2$9:  CMP      L+2(SP),R5
        BLOS     DIG$9
DG3$9:  ADD      #00,R0
        MOV8     R0,(R5)+
        CLR      R0
        JSR      PC,M54$9
        JSR      PC,ML8$9
        BR       DG1$9
DIG$9:  RTS      PC

RIT$9:  CLC
        ROR      R1
        ROR      R2
        ROR      R3
        ROR      R4
        RTS      PC
        .END

```


DISTRIBUTION

	Copy
R. Bulcock, DOE-KCAO, 1D49	1
A. Andrade, LLL	2
C. Jennings, SLA	3
J. F. Hadley, D/420, BN30	4
J. A. Ford, D/421, BN30	5
W. B. Abel, D/424, BQ30	6
J. S. Carson, D/424, BQ30	7
R. E. Hines, D/424, BQ30	8
J. E. Long, D/531, 1A46	9
H. T. Barnes, D/554, BD50	10-11
L. Stratton, D/554, 2C44	12-14
R. P. Frohmberg, D/800, 2A39	15
R. Morgan, D/821, 2A36	16
R. E. Kessler, D/865, 2C40	17

BDX-613-1943 (Rev.)

ELECTRICAL COMPENSATION SYSTEM FOR PLATED-
THROUGH-HOLE TESTER, W. B. Abel, UNCLAS
Milestone, October 1978

A microprocessor system has been developed to control the acquisition and interpretation of resistance-measurement data from plated-through holes in printed wiring boards. The system provides a direct digital display of the thickness of the copper plating on the hole walls. The complex calibration for holes having different geometries is stored in the memory of the system, and the geometry factors are input through thumbwheel switches. A special probe system makes possible the determination of an

SPECIAL PROJECTS: Nondestructive Testing

ELECTRICAL COMPENSATION SYSTEM FOR PLATED-
THROUGH-HOLE TESTER, W. B. Abel, UNCLAS
Milestone, BDX-613-1943 (Rev.), October 1978

A microprocessor system has been developed to control the acquisition and interpretation of resistance-measurement data from plated-through holes in printed wiring boards. The system provides a direct digital display of the thickness of the copper plating on the hole walls. The complex calibration for holes having different geometries is stored in the memory of the system, and the geometry factors are input through thumbwheel switches. A special probe system makes possible the determination of an

ELECTRICAL COMPENSATION SYSTEM FOR PLATED-
THROUGH-HOLE TESTER, W. B. Abel, UNCLAS
Milestone, BDX-613-1943 (Rev.), October 1978

A microprocessor system has been developed to control the acquisition and interpretation of resistance-measurement data from plated-through holes in printed wiring boards. The system provides a direct digital display of the thickness of the copper plating on the hole walls. The complex calibration for holes having different geometries is stored in the memory of the system, and the geometry factors are input through thumbwheel switches. A special probe system makes possible the determination of an

average resistance value for each test hole without moving the probe contacts in relation to the hole.

average resistance value for each test hole without moving the probe contacts in relation to the hole.

average resistance value for each test hole without moving the probe contacts in relation to the hole.