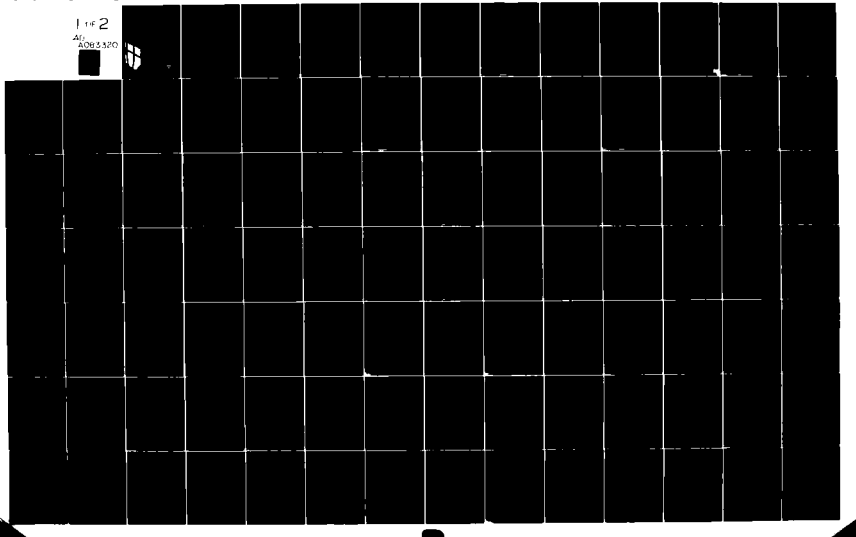
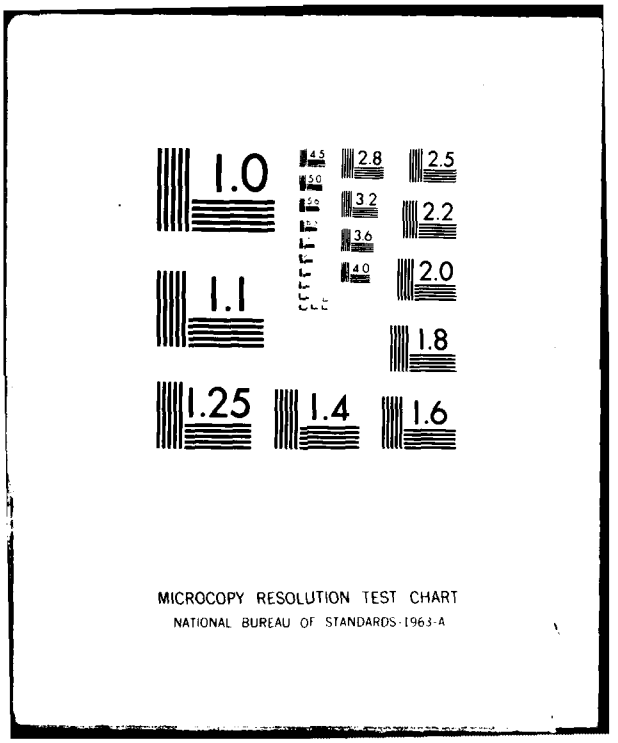
AD-A083 320 CONSTRUCTION ENGINEERING RESEARCH LAB (ARMY) CHAMPAIGN IL F/G 20/1
TRUE-INTEGRATING ENVIRONMENTAL NOISE MONITOR AND SOUND-EXPOSURE--ETC(U)
MAR 80 A J AVERBUCH, L M LITTLE
UNCLASSIFIED CERL-TR-M-41-VOL-3 NL



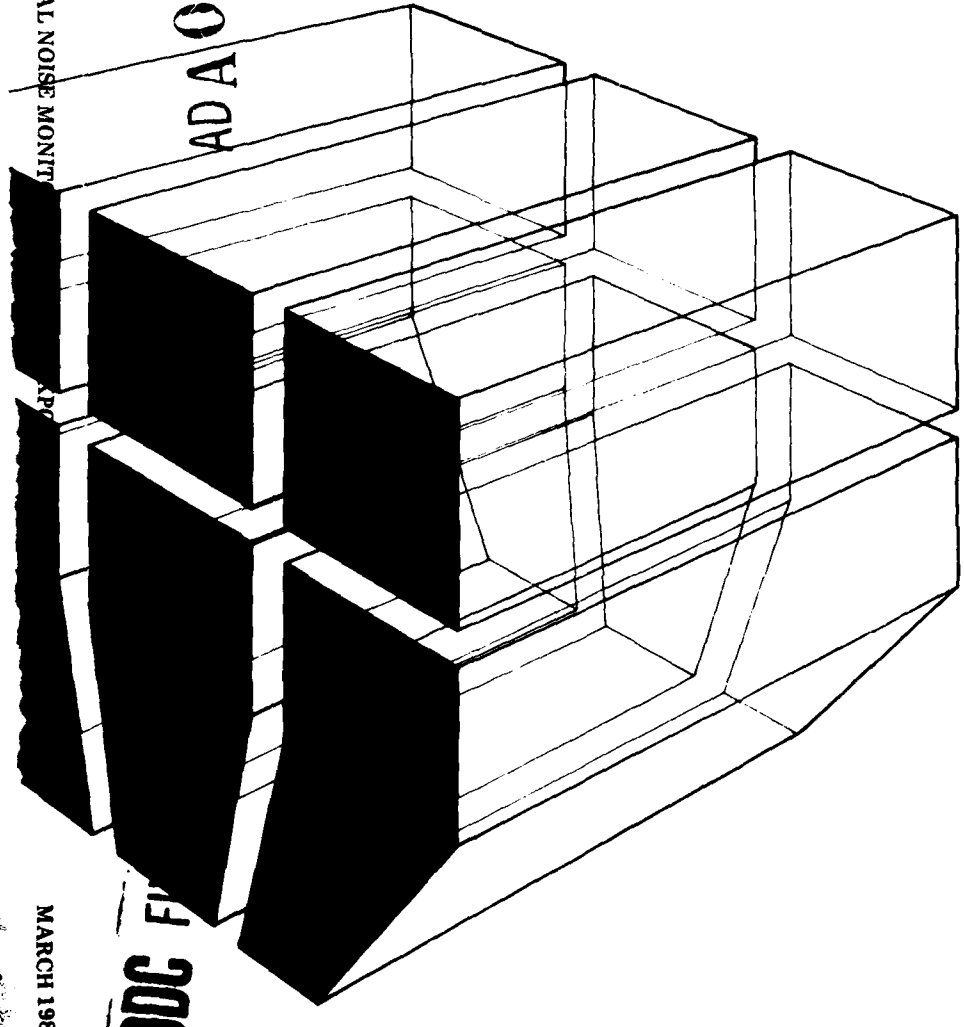

1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

construction engineering research laboratory

United States Army Corps of Engineers
...Serving the Army
...Serving the Nation

TECHNICAL REPORT N-41
March 1980
Prediction of Noise Impact Within and Adjacent to Army Facilities

TRUE-INTEGRATING ENVIRONMENTAL NOISE MONITOR AND SOUND-EXPOSURE LEVEL METER VOLUME III: MICROPROCESSOR PROGRAM AND DATA INTERFACE DESCRIPTION



by
A. J. Averbuch
L. M. Little

CERL

Approved for public release; distribution unlimited.

T. R. N-41
TRUE-INTEGRATING ENVIRONMENTAL NOISE MONIT
MARCH 1980

USER EVALUATION OF REPORT

REFERENCE: Technical Report N-41, *True-Integrating Environmental Noise Monitor and Sound-Exposure Level Meter, Volume III: Microprocessor Program and Data Interface Description*

Please take a few minutes to answer the questions below, tear out this sheet, and return it to CERL. As a user of this report, your customer comments will provide CERL with information essential for improving future reports.

1. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which report will be used.)

_____

_____

_____

2. How, specifically, is the report being used? (Information source, design data or procedure, management procedure, source of ideas, etc.)_____

_____

_____

3. Has the information in this report led to any quantitative savings as far as man-hours/contract dollars saved, operating costs avoided, efficiencies achieved, etc.? If so, please elaborate. _____

_____

_____

4. What is your evaluation of this report in the following areas?

   a. Presentation: _____

   b. Completeness: _____

   c. Easy to Understand: _____

   d. Easy to Implement: _____

i. General Comments (Indicate what you think should be changed to make this report and future reports of this type more responsive to your needs, more usable, improve readability, etc.) _____

_____

_____

_____

_____

4. If you would like to be contacted by the personnel who prepared this report to raise specific questions or discuss the topic, please fill in the following information.

Name: _____

Telephone Number: _____

Organization Address: _____

_____

_____

5. Please mail the completed form to:

Department of the Army
CONSTRUCTION ENGINEERING RESEARCH LABORATORY
ATTN: CERL-SOI
P.O. Box 4005
Champaign, IL 61820

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: CERL-TR-N-41-Vol-3	2. GOVT ACCESSION NO.: AD-A083320
3. RECIPIENT'S CATALOG NUMBER: (9)	
4. TITLE (and Subtitle): TRUE-INTEGRATING ENVIRONMENTAL NOISE MONITOR AND SOUND-EXPOSURE LEVEL METER, VOLUME III, MICRO-PROCESSOR PROGRAM AND DATA INTERFACE DESCRIPTION	5. TYPE OF REPORT & PERIOD COVERED: FINAL
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): A. J. Averbuch, L. M. Little	8. CONTRACT OR GRANT NUMBER(s): (17)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: U.S. ARMY CONSTRUCTION ENGINEERING RESEARCH LABORATORY P.O. Box 4005, Champaign, IL 61820	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: (16) 4A762720A896-03-001
11. CONTROLLING OFFICE NAME AND ADDRESS: (11)	12. REPORT DATE: March 1980
	13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): (12) 112	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Copies are obtainable from the National Technical Information Service
Springfield, VA 22151

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

noise (sound)
monitors

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

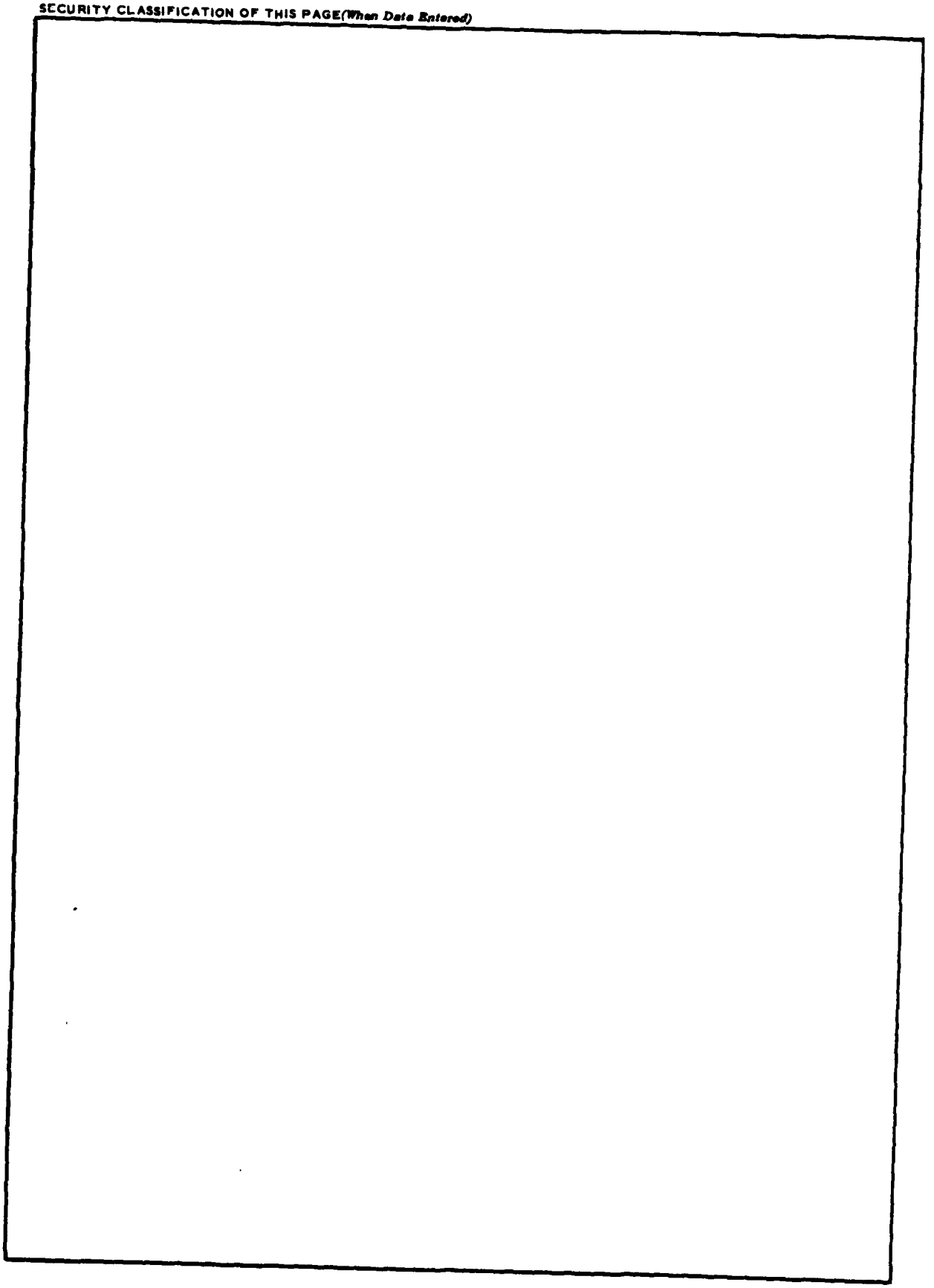
This report describes the internal microprocessor program used to operate the U.S. Army Construction Engineering Research Laboratory (CERL) True-Integrating Environmental Noise Monitor and Sound-Exposure Level Meter. Several data interface accessories are also described; complete program listings are included.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

405279

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

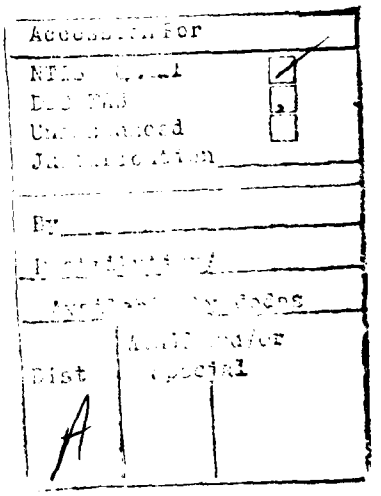


FOREWORD

This research was conducted for the Directorate of Military Programs, Office of the Chief of Engineers (OCE), under Project 4A762720A896, "Environmental Quality for Construction and Operation of Military Facilities"; Task 03, "Pollution Control Technology"; Work Unit 001, "Prediction of the Noise Impact Within and Adjacent to Army Facilities." The QCR number is 1.03.011. Mr. F. P. Beck, DAEN-MPE-I, is the OCE Technical Monitor.

The work was performed by the Environmental Division (EN), U.S. Army Construction Engineering Research Laboratory (CERL). Dr. R. K. Jain is Chief of EN.

COL Louis J. Circeo is Commander and Director of CERL, and Dr. L. R. Shaffer is Technical Director.



CONTENTS

TABLES

FIGURES

TRUE-INTEGRATING ENVIRONMENTAL
NOISE MONITOR AND SOUND-
EXPOSURE LEVEL METER
VOLUME III: MICROPROCESSOR PROGRAM
AND DATA INTERFACE DESCRIPTION

1 INTRODUCTION

Background

This is the third volume of a four-volume set of reports which describe the experimental background, use, specifications, and construction of the True-Integrating Environmental Noise Monitor and Sound-Exposure Level Meter developed by the U.S. Army Construction Engineering Research Laboratory (CERL).

This volume, in conjunction with Volume I, User's Guide; Volume II, Wiring and Parts Lists, Parts Layouts, and Schematics; Volume IV, Mechanical Construction and Electrical Checkout; and a set of contractual general provisions constitutes a purchase specification for the CERL noise monitor system.* Although this system is relatively complex and performs a variety of sound recording and analyzing functions, its construction is straightforward; the information in Volumes I through IV should enable an electronics system manufacturer or electronics laboratory to build the CERL noise monitor system.

Purpose

The purpose of this volume is to give a detailed description and explanation of the microprocessor program and interfaces for the True-Integrating Environmental Noise Monitor and Sound-Exposure Level Meter.

Outline of Report

Chapter 2 describes the program used by the monitor's internal microprocessor, including a list of the various hardware commands and a step-by-step description of the program operation. Chapter 3 describes external devices that can be attached to the monitor, the minisample tape recorder error correction technique, and formats of the magnetic tape records.

* Volume I published May 1978, CERL Technical Report (TR) N-41/ADA060958; Volume II published June 1979, CERL TR N-41/ADA072002; Volume IV, CERL TR N-41, published March 1980.

Mode of Technology Transfer

This four-volume set of reports contains the necessary technical information for the purchase specification and construction of the True-Integrating Environmental Noise Monitor and Sound-Exposure Level Meter.

2 MICROPROGRAM DESCRIPTION

General

The unit begins to execute the microprogram* at Loc 7777 when the power is first turned on. The program performs an initialization to set up the operating conditions of the monitor. Then, the program waits in the switch testing loop. Whenever a flag or a switch is set, the routine is exited and the requested operation performed. Appendix A lists program definable names which are tied to hardware function. Appendix B lists the microprogram used by the monitor. Appendices C through E contain a number of short test programs used in troubleshooting the monitor.

The initialization routine sets up the unit by clearing all flags and interfaces, setting up the read/write memory, and setting the unit to standby mode.

The initialization routine:

1. Clears printer interface**
2. Clears lights and memory control
3. Sets up interrupt+ jump
4. Transfers part of program from Field 2 to Field 0
5. Sets up page 0
6. Clears process control buffer
7. Determines size of Field 1 data memory
8. Sets mode to 10 (standby)
9. Zeroes energy buffers
10. Sets timing to 0.1 seconds (s)
11. Sets threshold to 9 decibels (dB)
12. Sets memory format to 6414 octal and resets all buffer pointers
13. Turns on interrupt
14. Begins switch testing routine.

---

* Microprograms are a series of steps executed by a computer which is composed of only a few large-scale integrated circuits. As integrated circuit technology progresses, the distinction between microprograms and standard computer programs will probably disappear.

**Printer, as used in this report, is a device which prints signed numbers up to six digits long, one number per line, with no alphabetic characters. Many desk top calculators have printers like this.

+ Interrupt is a capability of a computer to stop whatever it is doing to go and do something else. An interrupt request is only honored at the completion of a computer instruction so that the computer never loses any data as a result of an interrupt. This capability makes the computer seem to be able to do more than one thing at a time.

Switch Testing Routine

The switch testing routine determines what the monitor will do next by serially checking all flags. When "flag" set or a switch is depressed, the switch testing routine jumps to the appropriate routine. To do this, the monitor:

1. Reads front panel switches.

2. Checks for data ready flag from the data acquisition hardware:

   a. If flag is set, action is taken only during autocalibration

   b. If flag is not set, the monitor skips to the next test.

3. Skips this test if the printer is idle; otherwise, tests printer flag:

   a. If flag is set, goes to the printer service routine.

   b. If not set, skips to next test.

4. Tests for START switch depressed:

   a. If switch is not depressed, clears edge trigger flag and skips to next test.

   b. If switch is depressed, goes to START switch service routine only if edge trigger flag is clear.

5. Tests for SAMPLE switch depressed:

   a. If switch is not depressed, clears edge trigger flag and skips to next test.

   b. If switch is depressed, goes to SAMPLE switch service routine only if edge trigger is clear.

6. Debounces START/STOP print switch:

   a. If set for 80 milliseconds (ms), goes to START/STOP print service routine.

   b. If not set for 80 ms, skips to next test.

7. Tests for real time clock flag:

a. If set, goes to real time clock service routine.

b. If not set, skips to next test.

8. Tests for threshold exceeded flag:

a. If set, goes to threshold service routine on both positive and negative edges.

b. If not set, skips to next test.

9. Skips this test if display request not active; otherwise checks display flag:

a. If set, updates H-P* display.

b. If not set, skips to next test.

10. Skips this test if dump routine is idle; otherwise does indirect jump via dump action table.

11. Skips this test if tape output is idle; otherwise tests UART** output flag:

a. If set, goes to tape output service routine (UART's transmitter buffer is empty).

b. If not set, skips to next test.

12. Skips this test if tape input is idle; otherwise, tests UART input flag:

a. If set, goes to tape input service routine.

b. If not set, skips to next test.

---

* An H-P display is a four-digit display comprised of light-emitting diode (LED) arrays with integral decoder and drivers made by the Hewlett-Packard Corporation.

**The Universal Asynchronous Receiver Transmitter (UART) is a large-scale integrated circuit used to convert a parallel stream of data (eight bits at a time) to a serial stream with start and stop bits and a parity bit (if requested). The circuit can also simultaneously perform the reverse operation while checking for overrun (failure to retrieve data before reception of next data), and correct parity and framing (failure to receive stop bit when expected).

13. Reads FUNCTION switch code and saves it in memory.

14. Debounces EXECUTE switch:

a. If set for 80 ms, goes to function switch service routine only if DISPLAY switch is set.

b. If not set, skips to next test.

15. Debounces DISPLAY switch; if set for 80 ms, goes to section of function switch service routine entry point corresponding to the state of the EXECUTE switch.

16. Goes to Step 1.

Interrupt Handler

This routine retrieves data from the data acquisition hardware, converts the data to internal format, and handles the conversion to engineering units. It also computes the total energy during modes 0, 1, 2, 6, and 8. After completing computation, this routine stores the data in circular buffers. The variable MEMFLG is set by the main program to tell the interrupt routine which data should be stored in memory. The steps taken by the interrupt handler routine are:

1. Saves the current computer state in a separate area. This is needed so that the processor can be restored to the condition that existed just before the interrupt (after the data computation is performed).

2. Checks MEMFLG:

a. If MEMFLG is clear, then no data are being requested and the program jumps to the routine exit point.

b. If MEMFLG is not clear, the interrupt routine continues.

3. Reads the hardware time register, converts number to Log_2 seconds, and updates accumulated time for energy calculations.

4. Reads hardware peak* register for the program-selected channel, converts number to Log_2, and adds in the appropriate gain constant.

* Peak is the maximum instantaneous deviation of sound pressure. This can be either positive or negative; however, the monitor only displays the absolute value.

5. Compares the current peak reading with the current overall peak:

a. If the new peak is greater, it replaces the overall peak.

b. If new peak is not greater, the routine continues.

6. Reads Channel 1 (CH1) energy, converts to Log_2, adds in the CH1 gain constant, updates the accumulated CH1 energy, and finally computes the equivalent continuous sound level (L_{eq}) (the average level of this sample).*

7. Same as Step 6, but for Channel 2 (CH2).

8. Saves the values in Steps 3 through 7 in a display buffer (most recent values); the overall values are saved instead if bit 9 is set in MEMFLG.

9. Adjusts the calibration constant. Both channels can be done simultaneously during autocalibration or individually during manual calibration:

a. If bit 10 of MEMFLG is set, CH1 is adjusted.

b. If bit 11 of MEMFLG is set, CH2 is adjusted.

c. If neither bit 10 or 11 is set, no adjustment is performed.

10. Tests the data ignore flag. If the program requests a sample during modes 5 through 9, this flag is set:

a. If set, only the display buffer is updated; program continues with Step 13.

b. If not set, the routine continues.

11. Stores data in circular buffers:

a. Only those data requested by MEMORY FORMAT are saved.

b. If this is the first interrupt after 2400 hr, L_{dn} is saved in its own circular bufffer.** The energy accumulators are then reset to zero.

---

* The L_{eq} is the steady level, in A-weighted decibels (dBA), that would produce the same A-weighted sound energy over a stated period of time as a time-varying sound.

**L_{dn} is a 24-hr L_{eq}, except a 10 dB penalty is added to all levels measured between 2200 and 0700 hr.

12. Tests for autocalibration request:

a. If no request is present, jumps to Step 13.

b. If a request is present, enables the jump to the autocalibration routine. (Since this test is performed in the interrupt routine, autocalibration always occurs at the end of a block.)

13. Restores the current computer state from the separate buffer area. (This enables the processor to continue the main program from the exact point at which it was interrupted.)

Autocalibration Routine

This routine turns on the electrostatic actuator* attached to the outdoor microphone system, reads the resultant microphone output, and, if necessary, adjusts the gain constants. This routine is called once every 6 hr in modes 6 and 8 by the real time clock routine.

When the user enters mode 6 or 8, an entry is made in the real time clock task table to activate autocalibration in 6 hr. At the end of this 6-hr period, the real time clock sets a flag for the interrupt routine. When the current data block is completed, an interrupt occurs. During the interrupt service routine, the actual autocalibration procedure is enabled by replacing a no operation (NOP) instruction in the switch testing loop with a jump instruction pointing to the first instruction of the autocalibration routine.

After autocalibration (duration can be from a minimum of 5 s to a maximum of 40 s), the NOP instruction is reinstated and a short block is set up to keep the monitor synchronized. The real time clock is used to time the short block. After data are collected from the short block, the real time clock routine sets up the hardware to resume normal data collection.

The autocalibration procedure:

1. Sets the real time clock task table for the time that the next block should start. If the user has selected a block length less than 40 s, an error message is displayed (---4), and the autocalibration procedure is aborted.

---

* An electrostatic actuator is a device which is attached to a microphone and which uses an electric field to move the microphone diaphragm. Because this movement duplicates the action of acoustic pressure, and is repeatable, the device can be used to calibrate a microphone. Holes in the actuator allow acoustic pressure to readily pass through; therefore, the actuator does not have to be removed when the microphone is being used.

2. Sets the real time clock task table for the time of the next calibration (5 hr, 59 min, and 21 s away).

3. Sets the real time clock task table to inhibit the 2200 and 0700 hr threshold shifts.

4. Saves the threshold level and the sample length in a buffer. The threshold is set to zero and the sample length is set to 0.5 s.

5. Waits 1.5 s, then checks the microphone output level:

a. If the background level is more than 15 dB below the reference level, the autocalibration continues with Step 6.

b. If not, six retries are performed before discontinuing.

6. Turns on the electrostatic actuator and waits 1.5 s for it to stabilize.

7. Reads the microphone output and tests level and peak:

a. If the average level has not changed by more than 0.7 dB, and if the peak level is within 6 dB of the average level, then the gain constant is adjusted so that the microphone output matches the reference level.

b. If the level has changed by more than 0.7 dB, turns off the electrostatic actuator and returns to Step 5.

8. Turns off the electrostatic actuator in preparation for background noise check. The program waits 1.5 s to allow system to stabilize. Background level must be more than 15 dB below reference level; otherwise, the program returns to Step 5.

9. Restores the monitor threshold level and sample length.

10. Resets the real time clock task table to enable the 2200 and 0700 hr threshold shifts.

11. Begins executing short block.

Printer Flag Service Routine

This routine sets up pointers to the printer buffer. This buffer contains intermediate data used by the output and error correction code subroutines. The intermediate data have to be stored while the printer is printing to allow the subroutines to be shared with the minisample

tape recorder routines* Once the pointers are set, the routine jumps to wherever the program left off in the printer service routine.

The printer flag service routine:

1. Sets pointers

2. Retrieves return address

3. Jumps to return address.

Start Switch Service Routine

This routine sets up the conditions for taking data in the selected mode. To conserve memory space, the routine consists of a series of tables which determine the steps for each mode. These tables are interspersed by a short series of instructions.

The start switch service routine:

1. Checks the edge trigger flag:

   a. If set, returns to switch testing routine.

   b. If not set, executes rest of routine.

2. Sets edge trigger flag.

3. Clears threshold flag.

4. Terminates most recent data display.

5. Determines which mode table to use.

6. Turns off interrupt system.

7. Picks up memory flag from selected mode table (see Loc 0124 in Appendix B for a description of memory flag bits).

8. Resets appropriate bits in the mode word (see the symbol definitions in Appendix D for a description of mode bits).

9. Sets appropriate bits in the mode word.

---

* The minisample tape recorder is used to take samples of sound during measurements; this allows the operator to identify noise emission sources.

10. Executes a series of tasks whose addresses are given in the particular mode table.

11. Returns to switch testing routine.

Sample Switch Service Routine

This routine sets up the condition for sampling data in the selected mode. To conserve memory space, the routine consists of a series of tables which determine the steps for each mode. These tables are interpreted by a short series of instructions.

The sample switch service routine:

1. Checks the edge trigger flag:

    a. If set, returns to switch testing routine.

    b. If not set, executes rest of routine.

2. Sets edge trigger flag.

3. Checks MEMFLAG:

    a. If no bits are set, returns to switch testing routine.

    b. If bits are set, continues routine.

4. Goes to Step 4 of start switch service routine.

Print Switch Service Routine (With Printer Subroutine)

This routine transmits the information stored in the printer to an external device in a format tailored to that device. Depressing, then redepressing, alternately begins and halts the transmission of data. The routine sends a leader and header, the data stored in the main circular buffer, and the data in the L_{dn} circular buffer.

All the data pass through the printer subroutine. This routine has three major entry points which take the three major data types and convert them to binary coded decimal (BCD) format. The three major data types are positive logarithmic numbers, signed logarithmic numbers, and positive numerics.

The print switch service routine:

1. Sets up pointers to the magnetic tape.

2. Checks the printer flag test of the switch testing routine:

a. If a K2 instruction* is present, outputs a line feed, disables printer test by overwriting the K2 instruction with a JMP instruction,** and returns to the switch testing routine.

b. If a JMP instruction is present, enables a printer test by overwriting the JMP instruction with a K2 instruction and continues routine.

3. Turns on the printer and waits for 100 ms.

4. Determines which external device is connected to the monitor (see Loc 102 in Appendix B's device code listing):

a. If no device is connected, disables printer test by overwriting the K2 instruction with a JMP instruction and returns to the switch testing routine.

b. If a device is connected, sets a flag representing the particular device and continues the routine.

5. Checks to see if a tape recorder is connected to the monitor:

a. If a tape recorder is connected, sends a leader consisting of 16 bytes of 377_8 followed by synch word consisting of a byte of 13_8 and a byte of 320_8.

b. If other device is connected, does not output leader.

6. Outputs a header which consists of a line feed and a series of numbers which indicates the state of the monitor. These numbers are comprised of three data types which are converted to printable information by the print subroutine:

a. The print subroutine has three entry points:

(1) BPRINT -- converts incoming 12-bit value to a positive, four-digit BCD number.

(2) APRINT -- converts incoming 12-bit value (assumed to be a positive, logarithmic number) to a numeric, and then to a positive, four-digit BCD number.

---

* A K2 instruction loads an octal 2 in the computer's acumulator.
**A JMP instruction causes the computer to begin executing at the operand address.

(3) PRINT -- converts incoming 12-bit value (assumed to be a signed, logarithmic number) to a numeric, and then to a signed four-digit BCD.

b. Saves the numeric in a temporary location in case the output device is the magnetic tape.

c. Sends data in three formats:

(1) If printer, four-digit BCD with separate bits for decimal points and sign.

(2) If calculator, seven-digit BCD with special codes for negative (1,000,000) and line feed (2,000,000). No decimal points are transmitted.

(3) If magnetic tape, a continuous stream of 8-bit characters consisting of groups of 51 data bits to which a 12-bit error correcting code and a 1-bit parity code are appended.

d. Sends print pulse:

(1) If printer, continues with switch testing routine.

(2) If tape, continues with switch testing routine.

(3) If programmable calculator, waits 100 ms and tests for data accepted. If accepted, continues with switch testing routine; if not accepted, causes error to appear in the display. The programmable calculator must be programmed for a 100 ms delay (minimum) of its own before the next data request to insure proper interaction between the calculator and the monitor.

7. Outputs line feed.

8. Checks to see if main circular buffer is empty:

a. If not empty, outputs block number, outputs block information as specified by the format word and the printer subroutine (see table at Loc 4254 in Appendix B), adjusts pointers, and goes back to Step 7.

b. If empty and programmable calculator, sets up printer flag return to Step 7. (This effectively puts the empty buffer test into the switch testing routine.)

c. If empty and not programmable calculator, outputs a zero block number and continues routine.

9. Outputs a line feed.

10. Checks to see if L_{dn} circular buffer is empty:

a. If not empty, outputs block number, outputs L_{dn} information as specified by the format word (see table at Loc 4254 in Appendix B), adjusts pointers, and goes back to Step 9.

b. If empty, outputs a zero block number and disables printer by changing the instruction in the printer flag test to a JMP instruction.

Real Time Clock Service Routine

This routine clears the real time clock flag; updates the time of day clock; maintains separate words for seconds, minutes, hours, and days; and seeks process buffer to see if any processes are to be activated at this time. If so, it activates them and then continues the scan. At the completion of the scan, it returns to the switch testing routine.

The real time clock service routine:

1. Clears real time clock flag.

2. Turns off the interrupt system.

3. Updates the time of day clock by one count (1 s).

4. Turns the interrupt system back on.

5. Scans through the following process table:

a. Performs autostart.

b. Performs autocalibration.

c. Puts L_{dn} in L_{dn} circular buffer.

d. Turns off minisample tape recorder (recorder in clock-controlled mode).

e. Turns on minisample tape recorder (recorder in clock-controlled mode).

f. Adjusts threshold for nighttime operation (reduces threshold by 10 dB).

g. Adjusts threshold for daytime operation (increases threshold by 10 dB).

h. Takes short block of data (part of autocalibration).

6. Returns to switch testing routine.

Threshold Flag Service Routine

This routine performs operations on both the positive and negative edges of the threshold flag. On the positive edge and in modes 6 and 7, it turns on the minisample tape recorder, sends out a leader, and outputs identification data. On the positive edge and in mode 3, it turns on the start light and follows the same procedure as in modes 6 and 7. On the negative edge and in modes 6 and 7, it turns off the minisample tape recorder. On the negative edge and in mode 3, it turns off the start light and follows the same procedure as in modes 6 and 7.

Specifically, when the positive edge is detected, the threshold flag service routine executes Steps 1 through 13. When the negative edge is detected, the routine executes Steps 15 through 18.

The threshold flag service routine:

1. Checks threshold edge trigger flag:

a. If set, returns to switch testing routine.

b. If not set, executes rest of routine.

2. Sets edge trigger flag.

3. Tests MEMFLG:

a. If bit 3 is not set, returns to switch testing routine.

b. If bit 3 is set, turns on minisample tape recorder. In addition, if the monitor is in mode 3, turns on the start light.

4. Enables the minisample tape output flag test.

5. Sets up pointers to the minisample tape buffer.

6. Outputs a leader consisting of 16 bytes of 377_8 and a synch word consisting of a byte of 13_8 and a byte of 320_8.

7. Outputs the serial number.

8. Outputs block number.

9. Outputs day.

10. Outputs hours and minutes as "HH.MM."

11. Sends out blanks to insure that the error correction code routine receives at least 51 data bits.

12. Disables the minisample tape output flag test.

13. Disables the minisample tape input flag test.

14. Returns to switch testing routine.

15. Checks threshold edge trigger flag:

    a. If not set, returns to switch testing routine.

    b. If set, executes rest of routine.

16. Clears edge trigger flag.

17. Tests MEMFLG:

    a. If bit 3 is not set, returns to switch testing routine.

    b. If bit 3 is set, turns off minisample tape recorder. In addition, if the monitor is in mode 3, turns off the start light.

18. Returns to switch testing routine.

Display Update Routine

Once the interrupt handler sets a flag (DISFL) to -1 (telling the switch testing routine that new data are ready), the display update routine is executed. The display update routine automatically updates the display if the user requests one of the following:

1. CH1

2. CH2

3. Sample length

4. Peak.

The display update routine:

1. Sets DISFL to +1.

2. Checks display channel:

   a. If zero, returns to switch testing routine.

   b. If a positive number is requested, continues routine.

3. Uses channel number as an index into a jump routine.

4. Retrieves requested information:

   a. If peak, jumps to the function switch routine step which displays large, positive numbers.

   b. If sample length, jumps to the function switch routine step which displays signed numbers.

   c. If CH1-CH2, tests display switch for L_{eq} or sound-exposure level (SEL):*

      (1) If the display is SEL, adds time to L_{eq} value using a 13-bit addition:

         (a) If the result of the addition is negative, jumps to the function routine step which displays signed numbers.

         (b) If the result is positive, jumps to the function switch routine step which displays large, positive numbers.

      (2) If the display is L_{eq}, jumps to the function switch routine step which displays large, positive numbers.

Memory Data Display Flag Service Routine

When the user requests the display of data from the main or L_{dn} circular buffers, pointers are set up and the memory data display flag test is enabled. The memory data display flag service routine then displays a block number followed by the requested data. It continues to increment the block number and data pointers as long as the user depresses the display switch. When the routine reaches the end of the buffer, it returns to block number 1 and continues the scan.

* SEL is the integral over time of the acoustic pressure squared.

The memory data display flag service routine:

1. Increments the counter (DUMPFL):

   a. If the result is not zero, continues switch testing routine.

   b. If the result is zero, continues routine.

2. Resets the counter (DUMPFL).

3. Checks the next item in the data table:

   a. If zero (end of table), updates pointers and recalls the memory data display function routine.

   b. If not zero, sets the address of instructions in the function routine which will correctly display current data.

4. Retrieves data from memory and jumps to address computed in Step 3b.

Minisample Tape Output Flag Service Routine

This routine sets up pointers to the minisample tape output buffer. This buffer contains intermediate data used by the output and error correction code subroutines. The data have to be stored while the (minisample) tape is running to allow the subroutines to be re-entered. Then the subroutines can be shared with the printer routine. Once the pointers are set, the routine then jumps to wherever the program left off in the threshold flag service routine (see p 21).

The minisample tape output flag service routine:

1. Sets pointers

2. Retrieves return address

3. Jumps to return address.

Tape Input Flag Service Routine

This routine sets up pointers to the tape input buffers. The buffer contains intermediate data used by the tape input routine (see p 27). Once the pointers are set, the monitor jumps to wherever the program left off in the tape input routine.

The tape input flag service routine:

1. Sets up pointers
2. Picks up error flags
3. Inputs a byte
4. Masks out upper four bits
5. Updates checksum
6. Retrieves return address
7. Jumps to return address.

Function Switch Service Routine

This routine has two entry points, one for the execute switch (HSW4) and on for the display switch (HSW5). If the user enters at the execute switch (HSW4) and the display switch is not set, no action occurs. If the user enters at the execute switch and the display switch is set, a write is performed. If the user enters at the display switch (HSW5) and the execute switch is not set, a read is performed. If the user enters at the display switch and the execute switch is set, a write is performed. All writes are followed automatically by a read so that the new information can be displayed immediately.

The routine reads the position of the function switch, combines it with the position of the shift switch, and determines which entry in the function table will be selected. The entry is the address of the routine to be executed.

If the user enters at the execute switch (HSW4), the routine executes Steps 1 through 8 and then continues at Step 5; if the user enters at the display switch (HSW5), the routine executes Steps 4 and 5 and then continues at Step 6.

The function switch service routine:

1. Tests for display switch depressed:
   a. If not depressed, returns to switch testing routine.
   b. If depressed, continues routine.
2. Sets pointer to function write table.
3. Jumps to Step 5.

4. Checks for execute switch depressed:

   a. If depressed, jumps to Step 1 to do a write.

   b. If not depressed, continues routine.

5. Sets pointer to function read table.

6. Checks shift switch position:

   a. If in the black position, no action.

   b. If in the white position, increases the function pointer by 16 locations.

7. Retrieves address of function to be executed for the table.

8. Jumps to that address.

Subroutine Call and Return Routines

Normal subroutine calls for this computer store the return address in the first location of the subroutine. Since this microprogram is stored in read-only memory, the normal subroutine call will not work. Therefore, a routine is called in read/write memory which replaces the original computer subroutine call instruction. Unlike the computer's subroutine call instruction (in which the destination address was part of the instruction), the new subroutine call instruction is followed by the destination address.

When the program requires a subroutine call, it executes a computer subroutine call to read/write memory where the new call instruction is located. A return address is saved in the first location. This return address is placed on a stack to allow a nesting of the subroutines. The routine then picks up the destination address following the call and jumps to that address.

To exit a subroutine, the original computer instruction is simply an indirect jump which uses the subroutine's first location as the operand address. This microprogram implements a return instruction which retrieves the return address off the stack and jumps to that address.

The subroutine call routine:

1. Jumps to the rest of the routine stored in read-only memory

2. Saves accumulator in a temporary location

3. Decrements stack pointer

4. Picks up return address

5. Increments the return address

6. Pushes the return address on the stack

7. Picks up destination address

8. Restores accumulator

9. Jumps to destination address.

The subroutine return routine:

1. Saves the accumulator in a temporary location

2. Pops return address from stack

3. Increments the stack pointer

4. Restores accumulator

5. Jumps to return address.

Tape Input Service Routine

This routine is an absolute binary loader subroutine used to load programs or data into the read/write memory. Data may be loaded into any field of memory (but not across field boundaries). Programs can be loaded anywhere, but can only be executed in field 0. The monitor is currently set up with 1024 words of read/write memory starting at field 0, Loc 0000. Because the switch testing routine and important program pointers are located in low memory, nothing should be loaded below Loc LDNBOT (address 0572_8). The highest address available is limited by the presence of the tape input routine to Loc VLOPG (address 1642_8). It is acceptable to write data over the autocalibration routine instructions located at AUTOCM and up (address 1300_8), since autocalibration and test programs are never used simultaneously.

The routine expects a leader, a start address with field, the number of words to transfer, the data, and finally, an eight-bit truncated checksum. As characters are read in, the last error detected is saved. The possible errors are parity, framing, and overrun. Except for saving the error status, errors are ignored during reading. Characters are always processed.

When the leader is detected, the routine turns off the display. The display remains dark during data transfer. At the completion of the data transfer, the computed eight-bit checksum is compared to the eight-bit checksum read, the display is turned on, and the result of the checksum comparison is displayed. If the display reads zero, then no errors were detected. If the display does not read zero, there was an error and the tape must be reread.

The tape input service routine:

1. Performs initialization:
    a. Energizes the tape motor control relay.
    b. Enables tape input flag test.
    c. Clears UART data ready flag.
    d. Sets leader character counter to -4.
2. Detects leader:
    a. Reads a character from UART.
    b. Compares character with 377_8:
        (1) If not 377_8, goes to Step 1d.
        (2) If 377_8, advances counter and continues.
    c. Compares counter with 4:
        (1) If not 4, goes to Step 2a.
        (2) If 4, continues.
3. Detects synch byte:
    a. Reads a character from UART.
    b. Counts the number of binary 1 s:
        (1) If greater than 3, goes to step 3a.
        (2) If less than or equal to 3, continues.
4. Initializes data transfer:
    a. Turns off display.
    b. Sets checksum to zero.

c. Sets error status buffer to zero.

d. Sets address of error to zero.

5. Reads memory field.

6. Reads starting address.

7. Reads finish address.

8. Reads number of words to transfer.

9. Reads data while computing running checksum.

10. Reads checksum.

11. Completes data transfer:

a. Displays low order eight-bit checksum.

b. Turns off tape recorder.

c. Turns on display.

d. Disables tape input flag test.

e. Returns to switch testing routine.

3 EXTERNAL DATA DEVICES

Thermal Printer

The Datel DPP-7 thermal printer is a small, battery-operated digital printer which is used to obtain a hard copy listing of the data stored in the monitor. The connections required to interface the printer to the noise monitor are shown in Table 1.

To use the printer, the user depresses the momentary contact switch on the front of the printer to start the printing process. The process will stop automatically when all data in the monitor have been printed. (The format of the printout is shown in Volume I, pp 12 and 13.) The process can be stopped at any time by depressing the switch again. The microprogram operation is described on page 9 of this volume.

The information on program pulses used to operate the printer and the codes accepted by the printer are given in Appendix A. Although the digital logic contained in the printer is powered from the monitor's 5-volt (V) supply, the current required by the printer's paper advance mechanism is too great. A separate battery is used to supply this current; the battery's charger schematic is shown in Figure 1.

Programmable Calculator

The Wang Laboratories Model 600-A programmable calculator is a desktop, 120-V alternating current-(ac-)powered calculator used to obtain data from the monitor (usually as it is collected) for further processing and printing. The connections required to interface the calculator to the noise monitor are shown in Table 2.

To use the calculator, the user depresses the momentary contact switch which has been mounted on the connector to the Wang. This grounds the signal labeled key print and starts the data transfer. The format of the transfer is described in Chapter 2 of this volume and is the same as that shown for the Datel printer (Volume I, pp 12 and 13), with the exception that no decimal points are transmitted and a line feed is represented as the number 2,000,000. Negative numbers are sent by adding 1,000,000 to the absolute value of the number to be transmitted.

The sign of the number read by the calculator is connected to the wind meter. Whenever the wind is above the threshold set by the user on the monitor, the numbers read will be negative. Since the transmitted data are for the previous time period, true time alignment is not possible.

The data transmission process continues until all stored data are transmitted. The monitor microprogram then puts the calculator on standby. When more data are received, transmission resumes. The data are fully buffered by the monitor in the event data transmission to the calculator is slower than data collection. Data transmission can be stopped at any time by momentarily depressing the print switch.

Because of timing restrictions in the monitor, a 100 ms delay must be programmed after each calculator-read operation to guarantee recognition by the monitor. The calculator program to read the monitor should be:

Group I

Recall, [00]	Reads the monitor
Store, [06]	Saves value
Alpha, f(x), [00]	Generates π
Ln	Takes π's natural log
Clr display	Throws values away; 100 ms delay complete

Information on program pulses used to operate the programmable calculator is given in Appendix A.

Data Output Tape Recorder

The data output tape recorder can be any reel-to-reel or cassette recorder having a bandwidth of at least 6 kilohertz (kHz). A special interface is required to convert the parallel digital data from the monitor to serial data to a frequency-shift modulated tone signal suitable for recording. The connections required are shown in Table 3. The schematic is the same as shown in Volume II, p 138 (Board 18), except that the common bus for transmit and receive has been separated. The transmit section takes parallel data from the monitor and converts it to serial data. The receive section converts the serial data back to parallel data and is used with a computer. The transmission rate is 300 baud.

The interface from the computer to the cassette will have error detection lines, control lines, and data lines (Figure 2, Table 4). Parity error, overrun error, and framing error are detectable during the read. No errors are detectable during a write. The error lines go high when an error is detected.

Overrun error occurs when the user is late in reading the data and the interface has completed reading the next word. The old data are lost when the line goes high.

Framing error occurs when the interface is unable to find a stop bit. This can be caused by an extra start bit or a dropped data bit.

Parity error occurs when the parity of the data does not match the parity bit.

The control lines are cassette on/off, data ready and data accepted, print, and transmitter buffer empty (TBRE). Cassette on/off enables and disables a relay, thus turning the cassette recorder on and off. When it is high, the cassette is on; when it is low, the cassette is off. Data ready is set high by the interface when it has read a character. Data accepted is pulsed low by the user when he/she has read the data line. This then clears the data-ready line. Print is pulsed low by the user when the data are ready to be written. TBRE goes high when the data have been transmitted to the cassette. The leading edge of print clears TBRE.

Data are stored on the cassette as a bit stream using Kansas City standards.[1] The format of the tape is a leader, synch bytes, and records (Figure 3). The leader consists of 16 bytes of 377_8, but only four need to be detected. Next comes two synch bytes, 13_8 and 320_8 (Figure 4). The records are bit streams 64 bits long and consist of 51 bits of data, 12 bits of error correcting code, and 1 bit odd parity (Figure 5). The error correcting code is a cyclic code with the following polynomial:

$$X^{12} + X^{10} + X^8 + X^5 + X^4 + X^3 + 1.$$

The type of data used determines the data format on the tape. Figure 6 shows the format for stored data as written to tape. Table 5 shows the detailed information for the header of these data including the permissible range of numbers. Table 6 shows the details of a block of memory data and Table 7 shows the details of an L_{dn} block of memory data. Table 8 shows details of a zero block which terminates a section of data. Table 9 describes the details of a line feed.

The cassette tapes have a slightly different format. The arrangement of the tape is shown in Figure 7; details are given in Table 10

Minisample Tape Recorder Interface

The minisample tape recorder interface box allows the audio

[1] Manfred Peschke and Virginia Peschke, "BYTE's Audio Cassette Standards Symposium," BYTE, Volume 1 Number 6 (February 1976), p 72.

cassette recorder to be used to record threshold-exceeded data. The recorder must be stereo, with one channel storing the sound present and the other storing a digital time code used to correlate data. The schematic diagram for this interface is shown in Figure 8. Note that power for this interface and for the tape recorder is supplied by the monitor.

Table 1

Output Side Connector to Datel Printer

Monitor Signal Name	J17 48 Pin Side Connector	Cable Color Code	Datel Printer Connector	Datel Signal Name
PIO 36	A	WH/RED & ORG	C2 B 15	BCD 1
PIO 35	B	ORG/WH & BLUE	C1 A 10	BCD 2
PIO 34	C	BLUE/WH & ORG	C1 B 15	BCD 4
PIO 33	D	GRN/WH & BLK	C1 B 9	BCD 8
PIO 32	E	ORG/BLK & WH	C2 B 14	BCD 10
PIO 31	F	WH/RED & BLUE	C1 A 9	BCD 20
PIO 30	G	BLK/WH & GRN	C1 A 13	BCD 40
PIO 29	H	BLUE/BLK	C1 B 8	BCD 80
PIO 28	J	BLK/RED	C2 A 15	BCD 100
PIO 27	K	GRN	C2 B 11	BCD 200
PIO 26	L	BLUE/WH & BLK	C1 A 14	BCD 400
PIO 25	M	WH/BLK & GRN	C2 B 7	BCD 800
PIO 24	N	RED/WH & BLUE	C2 A 13	BCD 1K
PIO 23	P	RED/BLK & GRN	C2 B 10	BCD 2K
PIO 22	Q	GRN/BLK & ORG	C1 A 15	BCD 4K
PIO 21	R	ORG/BLK	C2 A 1	BCD 8K
PIO 20	S	BLK	C1 A 12	BCD 10K
PIO 19	T	BLK/GRN & RED	C2 B 8	BCD 20K
PIO 18	U	RED/WH	C1 B 10	BCD 40K
PIO 17	V	GRN/WH & BLUE	C2 B 2	BCD 80K
PIO 16	W	WH/GRN & RED	C1 A 11	BCD 100K
PIO 15	X	WH/BLK & RED	C2 A 8	BCD 200K
PIO 14	Y	RED/WH & BLK	C1 A 8	BCD 400K
PIO 13	Z	GRN/BLK	C2 B 1	BCD 800K
PIO 12	a	RED	C1 A 6	DEC. PT. 1
PIO 11	b	WH	C1 A 7	DEC. PT. 2
PIO 10	c	ORG	C1 B 6	DEC. PT. 3
PIO 9	d	GRN/WH	C1 A 1	DEC. PT. 4
PIO 8	e	BLUE/RED & ORG	C1 B 2	DEC. PT. 5
PIO 7	f	BLK/ORG & RED	C1 B 3	DEC. PT. 6
PIO 6	g	ORG/GRN	C1 B 1	minus sign
PIO 5	h	BLK/WH & RED	C1 A 5	plus sign
Print	z	SHIELDED WIRE	C1 B 14	print-advance
Data request	t	ORG/BLUE & RED	C2 B 12	busy
Logic ground	u	GRN/RED & ORG	C1 A 4	logic ground
Logic ground	v	ORG/RED	C2 B 5	logic ground
+5 V	x	WH/BLK & BLUE	C1 B 13 C2 B 13	+5 V

Table 1 (Cont'd)

Monitor Signal Name	J17 48 Pin Side Connector	Cable Color Code		Datel Printer Connector	Datel Signal Name
$\overline{\text{KEY PRINT}}$	n	BLUE/RED		C1 B 5	key print
PSEL 2	r	WH/BLK		GND	extra
PSEL 1	s	ORG/GRN & BLK		N.C.	extra
		GND	←	C1 B 11	chg. data pol.
		+5v	←	C1 B 7	chg. print pol.
		+5v	←	C1 B 4	lead 0 suppress
		+5v	←	C1 A 3	no print advance
		+5v	←	C2 B 6	test
		+5v	←	C1 A 2	chg. busy pol.
				C2 B 4	out of paper

Table 2

Output Side Connector to Wang

Monitor Signal Name	J17 48 Pin Side Connector	Cable Color Code	36 Pin Wang Connector	Wang Signal Name
PIO 36	A	WH/RED & ORG	13	BCD 1
PIO 35	B	ORG/WH & BLUE	14	BCD 2
PIO 34	C	BLUE/WH & ORG	15	BCD 4
PIO 33	D	GRN/WH & BLK	16	BCD 8
PIO 32	E	ORG/BLK & WH	27	BCD 10
PIO 31	F	WH/RED & BLUE	28	BCD 20
PIO 30	G	BLK/WH 7 & GRN	29	BCD 40
PIO 29	H	BLUE/BLK	30	BCD 80
PIO 28	J	BLK/RED	9	BCD 100
PIO 27	K	GRN	10	BCD 200
PIO 26	L	BLUE/WH & BLK	11	BCD 400
PIO 25	M	WH/BLK & GRN	12	BCD 800
PIO 24	N	RED/WH & BLUE	23	BCD 1K
PIO 23	P	RED/BLK & GRN	24	BCD 2K
PIO 22	Q	GRN/BLK & ORG	25	BCD 4K
PIO 21	R	ORG/BLK	26	BCD 8K
PIO 20	S	BLK	5	BCD 10K
PIO 19	T	BLK/GRN & RED	6	BCD 20K
PIO 18	U	RED/WH	7	BCD 40K
PIO 17	V	GRN/WH & BLUE	8	BCD 80K
PIO 16	W	WH/GRN & RED	19	BCD 100K
PIO 15	X	WH/BLK & RED	20	BCD 200K
PIO 14	Y	RED/WH & BLK	21	BCD 400K
PIO 13	Z	GRN/BLK	22	BCD 800K
PIO 12	a	RED	1	BCD 1M
PIO 11	b	WH	2	BCD 2M
PIO 10	c	ORG	3	BCD 4M
PIO 9	d	GRN/WH	4	BCD 8M
PRINT	x	SHIELDED WIRE	18	$\overline{\text{PRINT}}$
DATA REQUEST	t	ORG/BLUE & RED	31	$\overline{\text{EXECUTE}}$
LOGIC GROUND	u	GRN/RED & ORG	32	LOGIC GROUND
LOGIC GROUND	v	ORG/RED	33	LOGIC GROUND
$\underline{\text{LOGIC GROUND}}$	w	BLK/WH & BLUE	34	$\underline{\text{LOGIC GROUND}}$
$\overline{\text{KEY PRINT}}$	n	BLUE/RED	PRINT SWITCH	$\overline{\text{KEY PRINT}}$
WINDW	p	BLUE/RED & GRN	17	SIGN
SHIELD	y	SHIELD	32	LOGIC GROUND
CHASSIS DIGITAL GROUND	q	WH/RED	36	LOGIC GROUND
PSEL 1	s	ORG/BLK & GRN	DIGITAL GND	PSEL 1

Table 3

Output Side Connector to Cassette Interface

Monitor Signal Name	J17 48 Pin Side Connector	Cable Color Code	50 Pin Interface Input Connector	44 Pin UART Board Connector	Cassette Interface Signal Name
PIO 1	m	RED/GRN	34	B	RELAY
PIO 12	a	RED	43	N	TBR 1
PIO 11	b	WH	23	M	TBR 2
PIO 10	c	ORG	12	L	TBR 3
PIO 9	d	GRN/WH	14	K	TBR 4
PIO 8	e	BLUE/RED & ORG	3	J	TBR 5
PIO 7	f	BLK/ORG & RED	38	H	TBR 6
PIO 6	g	ORG/GRN	11	F	TBR 7
PIO 5	h	BLK/WH & RED	39	E	TBR 8
PRINT	z	SHIELDED WIRE	36	V	TOUT
DATA REQUEST	t	ORG/BLUE & RED	15	R	TBRE
+5 V	x	WH/BLK & BLUE	26		+5 V
GROUND	u	GRN/RED & ORG	41		GROUND
GROUND	v	ORG/RED	37		GROUND
GROUND	w	BLK/WH & BLUE	17		GROUND
SHIELD GROUND	y	SHIELD	28		GROUND
CHASSIS GROUND	q	WH/RED	32		GROUND
PSEL 2	r	WH/BLK	47		GROUND
PSEL 1	s	ORG/BLK & GRN	8		GROUND
KEY PRINT	n	BLUE/RED	19	PRINT SWITCH C	
				D	TAPE RECORDER MOTOR

Table 4

Cassette Computer Interface Signals

		READ	
0 L	DATA READ	1	1 = LSB
0 L	DATA READ	2	
0 L	DATA READ	3	
0 L	DATA READ	4	
0 L	DATA READ	5	
0 L	DATA READ	6	
0 L	DATA READ	7	
0 L	DATA READ	8	8 = MSB
I L	CASSETTE ON		
0 L	DATA READY		
I L	$\overline{\text{DATAACCEPTED}}$	($\overline{\text{TIN}}$)	
0 L	PARITY ERROR		
0 L	OVERFLOW ERROR		
0 L	FRAMING ERROR		
		WRITE	
I L	DATA WRITE	1	1 = LSB
I L	DATA WRITE	2	
I L	DATA WRITE	3	
I L	DATA WRITE	4	
I L	DATA WRITE	5	
I L	DATA WRITE	6	
I L	DATA WRITE	7	
I L	DATA WRITE	8	
I P	$\overline{\text{PRINT}}$		
0 L	TBRE (TRANSMITTER BUFFER EMPTY)		

TO AND FROM COMPUTER
I = In to Interface
0 = Out of Interface
L = Level Data
P = Pulse Data

Table 5

Header

Word*	Name	Description
1	Line Feed	0
2	(Five words of zero)	0
3		0
4		0
5		0
6	Day of year	0 to 364
7	Hour	0 to 23
8	Minutes	0 to 59
9	Serial number	GGUU (GG = group number; UU = unit number)
10	Mode number accumulation time	2 to 9
11	Minutes	1 to 999 (minutes or seconds must be zero)
12	Tenths of a second	1 to 999
13	Number of channels	1 to 2
	Calibrator levels	
14	Channel 1	605 to 1604 dB
15	Channel 2	605 to 1604 dB
	Gain constant	
16	Channel 1	-799 to 799 dB
17	Channel 2	-799 to 799 dB
18	Peak detector Channel #	1 or 2
19	Threshold level	0 to 182 or 800 to 982 dB (If 800 to 982 night factor suppressed.)
	Mini Sample	
20	On time	1 to 999 s
21	Total time	1 to 999 min
22	Format of data stored	0 to 7777 octal
23	Line feed	0
24	(five words of zero)	0
25		0
26		0
27		0

*All words are 12 bits long; all decibels are to tenths (e.g., 60.5 dB is 605 dB).

Table 6

Block Data Format

Word*	Name	Description
1	Block number	
2	Channel 1 level	0 to 1920 dB
3	Channel 2 level	0 to 1920 dB
4	Peak	0 to 1920 dB
5	Sample length	-960 to 960 s (in decibels)
6	Channel 1 gain	-799 to 799 dB
7	Channel 2 gain	-799 to 799 dB
8	Time	HHMM (HH = hour; MM = minutes)
9	Line feed	0
10	(Five words of zero)	0
11		0
12		0
13		0

* All words are 12 bits long; all decibels are to tenths (e.g., 192.0 dB is 1920 dB).

Table 7

L_{dn} Data Format

Word*	Name	Description
1	Block number	
2	Channel 1 L_{dn}	0 to 1920 dB
3	Channel 2 L_{dn}	0 to 1920 dB
4	Duration day	0 to 1920 dB
5	Duration night	0 to 1920 dB
6	Channel 1 L_d	0 to 1920 dB
7	Channel 1 L_n	0 to 1920 dB
8	Channel 2 L_d	0 to 1920 dB
9	Channel 2 L_n	0 to 1920 dB
10	Line feed (five words of zero)	0
11		0
12		0
13		0
14		0

*All words are 12 bits long; all decibels are to tenths (e.g., 192.0 dB is 1920 dB); negative numbers are 2's complement.

Table 8

Zero Block Format

Word*	Name	Value
1	Block Number	0

Table 9

Line Feed Format

Word*	Name	Value
1	(Line	0
2	feed	0
3	all	0
4	zeros)	0
5		0

*All words are 12 bits long.

Table 10

Time Data

Word	Name	Description
1	Serial number	GGUU (GG = group number; UU = unit number)
2	Block number High	
	Low	
	Day of the year	0 to 364
	Hours + minutes	HHMM

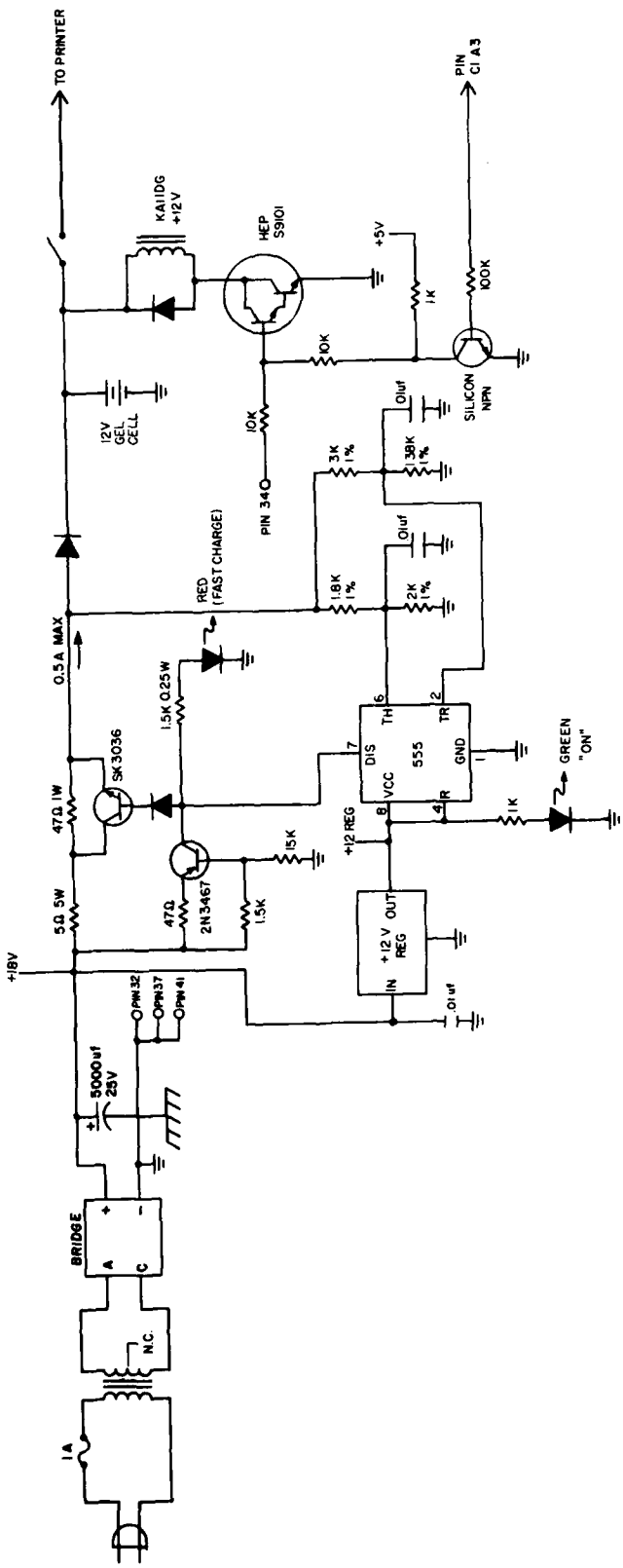


Figure 1. Schematic of battery charger.

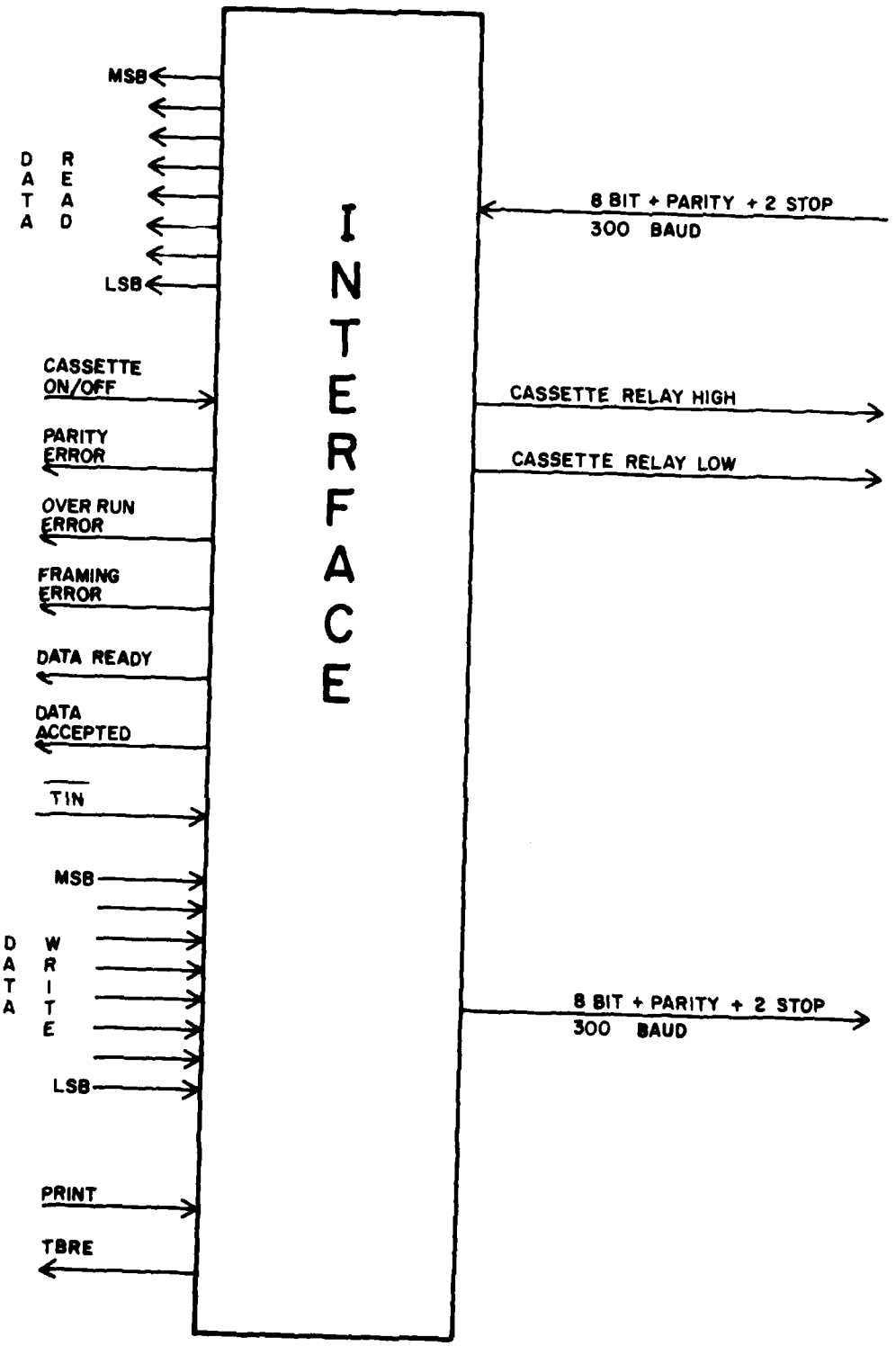


Figure 2. Cassette interface.

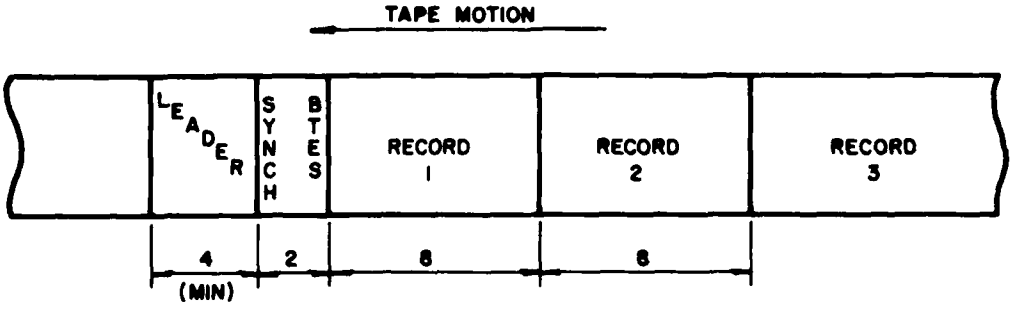


Figure 3. Tape format.

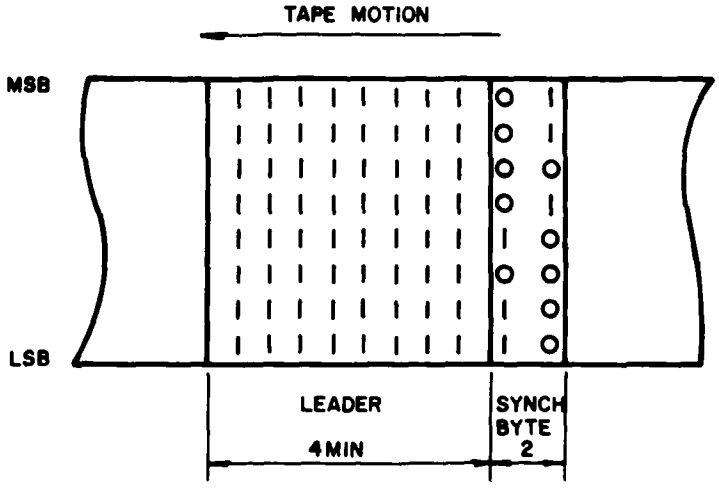


Figure 4. Leader and synch format.

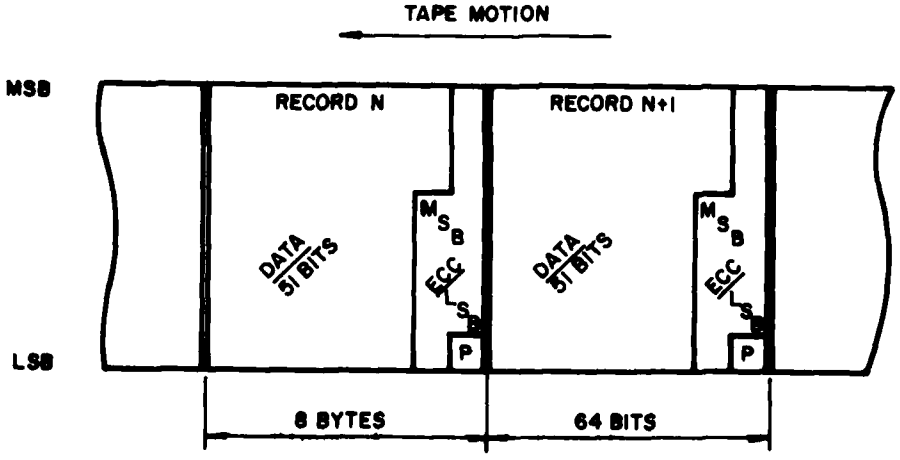


Figure 5. Record format.

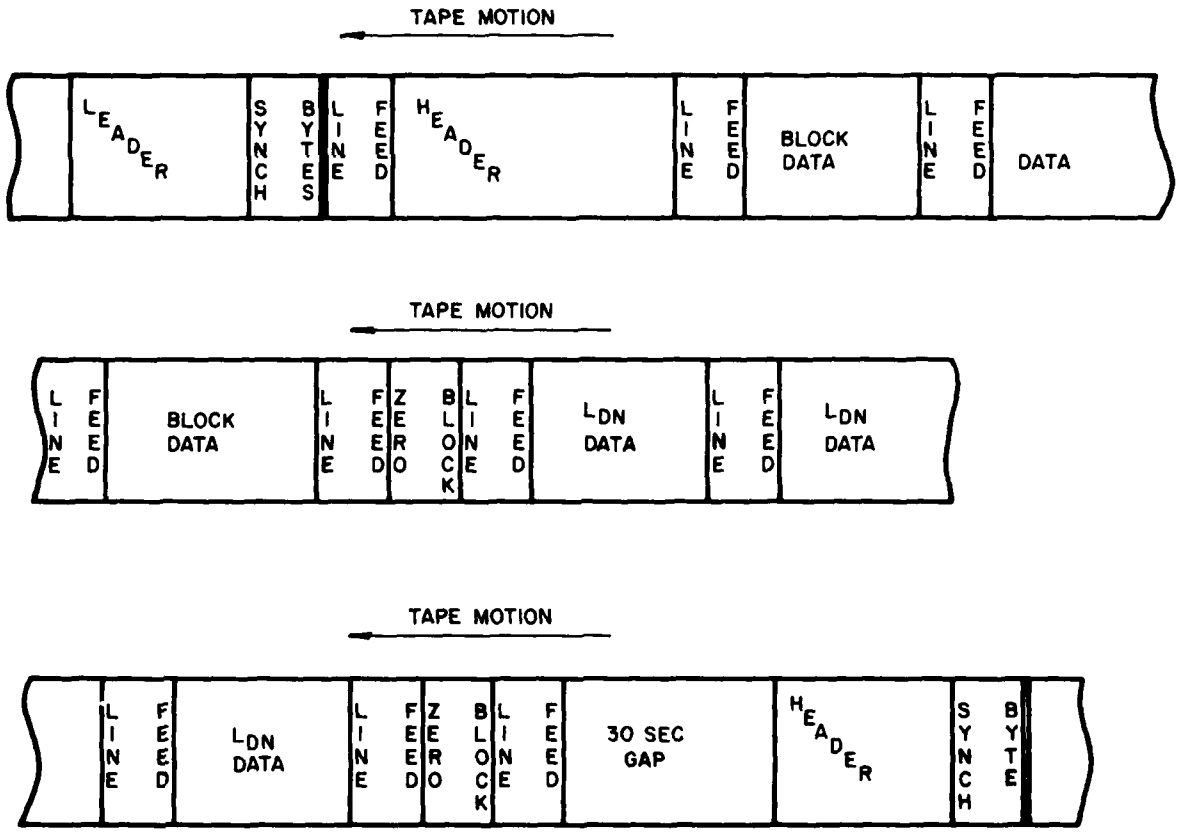


Figure 6. Stored data format.

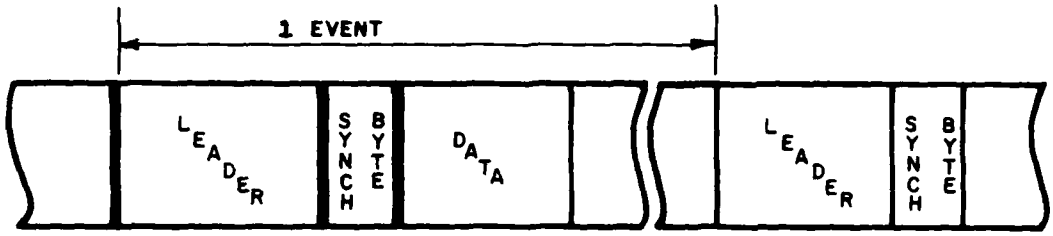


Figure 7. Threshold data.

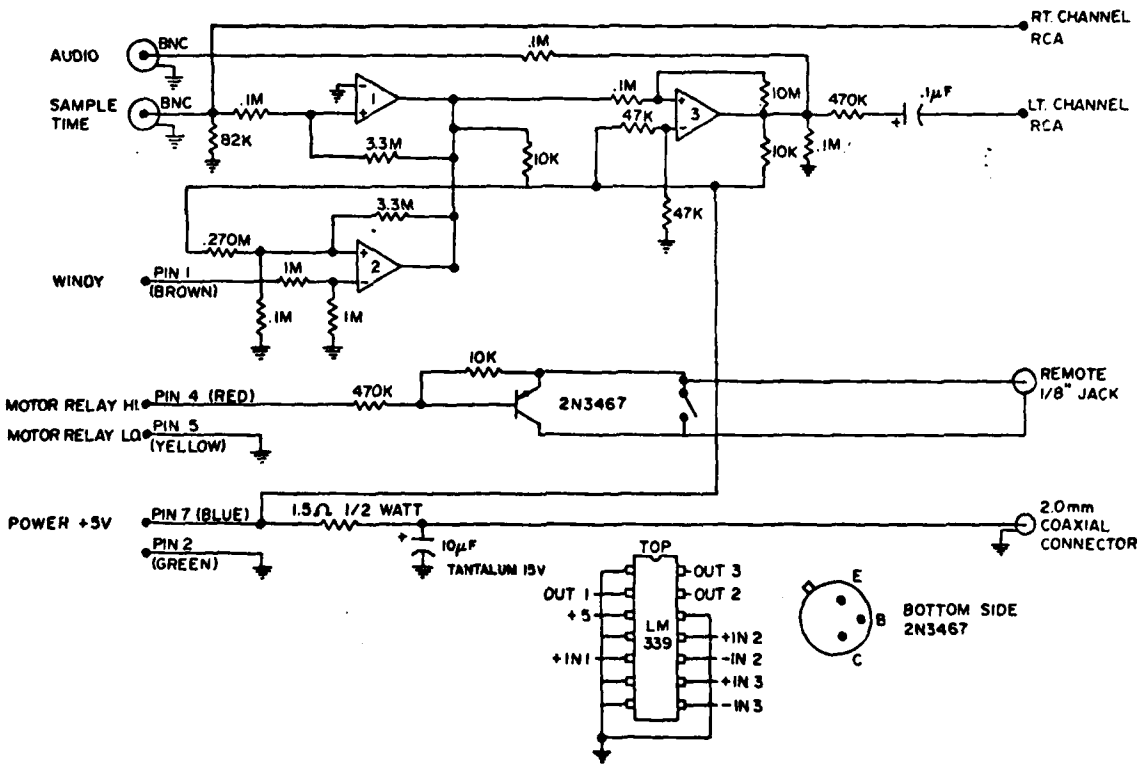


Figure 8. Schematic for minisample tape recorder interface.

APPENDIX A:

PROGRAM PULSE AND DATA FORMATS

Display Codes

For Wang:		For H-P:	
	10 = .		10 = Test
	11 = =		11 = blank
	12 = 4		12 = blank
	13 = 5		13 = minus sign
	14 = t		14 = blank
	15 = blank		15 = blank

Program Pulses

Write	6600	SEC	
Write	6601	MIN	
Write	6602	THSET	
Write	6603	SELCHAN	
Write	6604	LODIS	
Write	6605	HIDIS	
Write	6606	DATA ACCEPTED	
Write	6607	MODE	
Write	6610	OUTLO	Positive 12-V pulse
Write	6611	OUTMI	Positive 12-V pulse
Write	6612	OUTHI	Positive 12-V pulse
Write	6613	BEGIN	Positive 12-V pulse
Write	6614	SAMPLE	Positive 12-V pulse
Write	6615	OPRINT	Positive 12-V pulse
Write	6616	RTC clear	Positive 12-V pulse
Write	6617	TOUT	Positive 5-V pulse to UART
Read	6620	EXTRA	
Read	6621	SWREG	Negative 5-V pulse
Read	6622	FUNLO	Negative 5-V pulse
Read	6623	FUNHI	Negative 5-V pulse
Read	6624	OUTPUTD	Negative 5-V pulse
Read	6625	PEAKD	Negative 5-V pulse
Read	6626	STATIN	to UART
Read	6627	TIN	to UART

Program Accessible Signal Definitions

THSET Bd 9	HDX11	A2	A2, A1, A0 are range bits
	HDX10	A1	
	HDX9	A0	
	HDX9	MSB	MSB to LSB are amplitude bits

```
                    ...     ...
                    HDX1    LSB+1
                    HDX0    LSB

SEC Bd 11 sets time divider to 1
MIN Bd 11 sets time divider to 600
MIN and SEL cause a pulse labeled TIME SET
TIME SET  Bd 11  HDX11 = MSB  sets count for BCD divider for block length
                    ...     ...
                    HDX0 = LSB
BEGIN     Bd 11  Causes synch pulse, read sam; zeroes 72-bit
                    accumulators and loads time down counters;
                    selects which data are to be read.
SELCHAN   Bd 11  HDX0  HDX1
                   0     0     Channel 1
                   0     1     Channel 2
                   1     0     Time
                   1     1       0

Data Accept Bd 11          Clears data ready
OPRINT      Bd 11          1 ms PRINT to printer connector

OUTLO, OUTMID, OUTHI  LOUTLO, LOUITMID, LOUTHI
            Bd 11 and Bd 12  active low pulse;  1 = latch  0 = transmit data
            (see wiring list for printer)

RTC       Bd 16                  Real time clock (1 s) flag
RTC Clear Bd 16                  Clears real time clock flag
MODE      Bd 18     HDX0-HDX3    MODE SELECT
                    HDX4- HDX5   FIELD SELECT
          Bd 16     HDX8         STANDBY LED
                    HDX9         CALIBRATE RELAY
                    HDX10        MINITAPE MOTOR
                    HDX11        VDISPLAY

LODIS  Bd 16    HDX11                              HDX7        HDX3
                HDX10  No connection               HDX6 LSD+1  HDX2
                HDX9                               HDX5        HDX1 LSD
                HDX8                               HDX4        HDX0

HIDIS  Bd 16    HDX11  Decimal point (DP) MSD      HDX7        HDX3
                HDX10         DP  MSD-1            HDX6 MSD    HDX2 MSD-1
                HDX9          DP  LSD+1            HDX5        HDX1
                HDX8          DP  LSD              HDX4        HDX0

OUTPUT D   Bd 15       DX11 MSB                        Data 2 from OSR mux
                       ...  ...
                       DX0 LSB
```

SWREG	Bd 16	DX11 MSB8 ... ... DX0 LSB1	12-bit switch reg
FUNLO	Bd 15	DX11 SEL/LEQ for display DX10 PSEL 1 DX9 PSEL 2 DX8 threshold exceeded flag DX7 RTC flag DX6 SINGLE DX5 KEYPRINT DX4 KEYSAMPLE DX3 KEYSTART DX2 SHIFT F DX1 DATA REQUEST (printer) DX0 DATA READY (OSR)	H LEQ, L=SEL H = 1 H = 1 H = 1 H = 1 L = SINGLE, H = DUAL H = 1 H = 1 H = 1 H = red, L = Black H = 1 H = 1
FUNHI	Bd 18	DX11 Serial number MSB DX10 . DX9 . DX8 . DX7 . DX6 Serial number LSB	
	Bd 16	DX5 DISPLAY SWITCH DX4 EXECUTE SWITCH (F4) DX3 F3 DX2 F2 DX1 F1 DX0 F0	
TOUT	Bd 17	DX7 TBR8 ... ... DX0 TBR1	Transmit data
TIN	Bd 17	DX7 RBR8 ... ... DX0 RBR1	Receive data
STATIN	Bd 17	DX7 DR DX6 TBRE DX5 PE DX4 FE DX3 OE	Received data ready Transmit buffer empty Parity error Framing error Overrun error
PEAKD	Bd 17	DX11 A2 DX10 A1 DX9 A0 DX8 MSB ... .. DX0 LSB	A2, A1, A0 are range bits MSB through LSB are magnitude bits

Output to 50 pin connector			Datel	Wang	Cassette
OUTLO	LSB	PIO 36	BCD 1	BCD 1	
		35	2	2	
		34	4	4	
		33	8	8	
		PIO 32	BCD 10		
		31	20	BCD 10	
		30	40	20	
		29	80	40	
		PIO 28		80	
		27	BCD 100		
	MSB	26	200	BCD 100	
		25	400	200	
OUTMI	LSB		800	400	
		PIO 24		800	
		23	BCD 1000		
		22	2000	BCD 1000	
		21	4000	2000	
			8000	4000	
		PIO 20		8000	
		19	BCD 10k		
		18	20k	BCD 10k	
		17	40k	20k	
			80k	40k	
		PIO 16		80k	
		15	BCD 100k		
	MSB	14	200k	BCD 100k	
		13	400k	200k	
OUTHI	LSB		800k	400k	
		PIO 12		800k	
		11	Dec pt 1		
		10	pt 2	BCD 1M	
		7	pt 3	2M	LSB
			pt 4	4M	LSB
		PIO 8		8M	LSB
		7	pt 5		LSB
		6	pt 6	LSB	
		5	minus sign	LSB	LSB
			plus sign	LSB	LSB
		PIO 4		MSB	LSB
		3			MSB
	MSB	2			
		1			
			Power on Relay		
OPRINT					Motor on Relay
FUNLO		PRINT			
FUNLO		DATA REQUEST	PRINT & ADVANCE		
		KEYPRINT	BUSY	PRINT	
		WINDY	KEYPRINT	EXECUTE	TBRL
					TBRE
				Sign	

APPENDIX B

LISTING OF PROGRAM WHICH CONTROLS
THE MONITOR CHARACTERISTICS

```
/
EXPUNGE                          /CLEAR SYMBOL TABLE
/
/
/MAJOR OP CODES
/
AND=0000
TAD=1000
ISZ=2000
DCA=3000
JMS=4000
JMP=5000
/
/
/
/
/GROUP 1 MICROINSTRUCTIONS
/
NOP=7000
CLA=7200
CLL=7100
CMA=7040
CML=7020
RAR=7010
RTR=7012
RAL=7004
RTL=7006
IAC=7001
BSW=7002
/
/
/COMBINED GROUP 1 MICROINSTRUCTIONS
/
CIA=7041                         /CMA IAC
STA=7240                         /CLA CMA
STL=7120                         /CLL CML
GLK=7204                         /CLA RAL
/
/
/
/CONSTANTS
K1=7301                          /CLA CLL IAC
K2=7305                          /CLA CLL IAC RAL
K3=7325                          /CLA CLL CML IAC RAL
K4=7307                          /CLA CLL IAC RTL
K6=7327                          /CLA CLL CML IAC RTL
K4000=7330                       /CLA CLL CML RAR
K6000=7333                       /CLA CLL CML IAC RTR
KM3=7346                         /CLA CLL CMA RTL
KM2=7344                         /CLA CLL CMA RAL
/
/
/
/GROUP 2 MICROINSTRUCTIONS
/
SMA=7500
SZA=7440
SPA=7510
SNA=7450
```

```
SNL=7420
SZL=7430
SKP=7410
HLT=7402
/
/
/
/GROUP 3 MICROINSTRUCTIONS
MQL=7421
MQA=7501
SWP=7521
CAM=7621
ACL=7701
/
/
/PROCESSOR IOT'S
SKON=6000
ION=6001
IOF=6002
SRQ=6003
GTF=6004
RTF=6005
CAF=6007
/
/
/
/ I/O DEFINITIONS
/
/
RSF=6011
RRB=6012
RFC=6014
PSF=6021
PLS=6026
/
/
SEC=6600
MIN=6601
THSET=6602
SELCHN=6603
LODIS=6604
HIDIS=6605
GOTDAT=6606
SELMD=6607
/
OUTLO=6610
OUTMI=6611
OUTHI=6612
BEGIN=6613
SAMPLE=6614
OPRINT=6615
CLCF=6616
TOUT=6617
/
SWREG=6621
FUNLO=6622
FUNHI=6623
OUTREG=6624
PEAK=6625
STATIN=6626
```

```
TIN=6627
/
/
/
/DEFINITION TABLE
TTI=6100                /READ1  READ TTI BUFFER, RESET FLAG
TTO=6101                /WRITE1 LOAD TTO BUFFER, RESET FLAG
TTIS=6102               /SKIP1  SKIP ON TTI FLAG, DATA RECEIVED
TTOS=6103               /SKIP2  SKIP ON TTO FLAG, TRANSMIT BUFFER EMPTY
TTRA=6104               /RCRA   READ CONTROL REG A
TTWA=6105               /WCRA   WRITE CONTROL REG A
TTIRS=6106              /SFLG1  SET READER RUN RELAY
TTIRC=6107              /CFLG1  CLEAR READER RUN RELAY
/6110 NOT USED
TTOC=6111               /WRITE2 LOAD UART CONTROL BITS
TTISB=6112              /SKIP3  SKIP ON UART START BIT DETECTED
/6113 NOT USED
TTWV=6114               /WVR    WRITE VECTOR REGISTER
TTWB=6115               /WCRB   WRITE CONTROL REG B
/6116 NOT USED
/6117 NOT USED
/
/
/
/
/MODE DEFINITIONS
VDIS=4000
MINITA=2000
CALREL=1000
STNBLD=400
STRTLD=200
PKCHN=100
FLDSEL=60
MODNUM=17
/
FIXTAB                  /KEEP IT FOREVER
/
/
```

```
                FIELD 1
                /
                /
                /PAGE 0 DEFINITIONS.
                *10
0010  0000  INT1,   0              /POINTERS DURING INTERRUPT SERVICE ROUTINE
0011  0000  INT2,   0              /USED MAINLY BY ADDE FOR MULTIPLE PRECISION ARITHMETIC
0012  0000  INT3,   0
0013  0000  XRT1,   0              /TEMPORARY REG WITH AUTO INDEX
0014  0000  XRT2,   0              /TEMPORARY REG WITH AUTO INDEX
0015  0000  XRT3,   0              /TEMPORARY REG WITH AUTO INDEX
0016  0571  LDN17,  LDNBOT-1       /LDN INPUT POINTER
0017  7777  XR17,   -1             /INPUT POINTER
                /
                /
                PRLEN=12
                *22+PRLEN          /NEED MORE ROOM FOR PRINTER POINTERS
                /BUFL=(MEMTOP-BUFBEG)/LEN
                /BUFL IS SET DURING INIT
                BUFL=1252          /ASSUMES BUFFER AT 3777
0034  6526  STORE,  -BUFL          /LENGTH OF MEMORY STORAGE IN BLOCKS
0035  0003  LEN,    3             /NUMBER OF MEMORY WORDS PER BLOCK
0036  6526  INFLO,  -BUFL          /INPUT BUFFER COUNTER
0037  0000  XR16,   0              /OUTPUT POINTER
0040  6526  OUTFLO, -BUFL          /OUTPUT BUFFER COUNTER
0041  7777  EMPTY,  -1             /BUFFER STATUS FLAG. -1=EMPTY,
                                   /0=HAS SOMETHING, 1=FULL
                /LDNSTR IS SET DURING INIT
                LDNSTR=33
0042  7745  LDNST,  -LDNSTR        /LENGTH OF LDN BUFFER IN BLOCKS
0043  0002  LDNLEN, 2              /NUMBER OF MEMORY WORDS PER LDN BLOCK
0044  7745  LDNIN,  -LDNSTR        /LDN INPUT BUFFER COUNTER
0045  0572  LDN16,  LDNBOT         /LDN OUTPUT POINTER
0046  7745  LDNOT,  -LDNSTR        /LDN OUTPUT BUFFER COUNTER
0047  7777  LDNMT,  -1             /LDN BUFFER STATUS FLAG
                /
0050  0000  SAVAC,  0              /SAVES AC DURING INTERRUPT
0051  0000  SAVLK,  0              /SAVES LINK DURING INTERRUPT
0052  0000  SAVMQ,  0              /SAVES MQ DURING INTERRUPT
0053  0000  SVSUB,  0              /SAVES STACK CALL DURING INTERRUPT
0054  0000  SVSUB1, 0              /SAVES STACK TEMP DURING INTERRUPT
0055  0000  TEMP3,  0              /TEMPORARY LOCATION USED BY INTERRUPT
0056  0000  MEMTOP, 0              /HOLDS TOP LOCATION OF DATA MEMORY (FIELD ONE ONLY)
                /DISPLAY BUFFERS
0057  0000  CH1,    0              /CH1 LEVEL
0060  0000  CH2,    0              /CH2 LEVEL
0061  0000  PK,     0              /PEAK LEVEL
0062  0000  TIME,   0              /TIME
0063  0000  SMPIG,  0              /IGNORE DATA DURING SAMPLE IF
                                   /=0 TAKE DATA, =1 IGNORE
                                   /SIMILAR TO STANDBY FOR ONE READING
                /
0064  6526  DFLO,   -BUFL          /DUMP BUFFER COUNTER
0065  0000  DX16,   0              /DUMP ADDRESS POINTER
0066  7745  DLFLO,  -LDNSTR        /LDN DUMP BUFFER COUNTER
```

```
0067  0572  DLX16,  LDNBOT          /LDN DUMP ADDRESS POINTER
0070  0000  DTIME,  0               /HOLDS TIME FROM BUFFER DURING DUMP      /LOAD FIELD
0071  0000  DUMPFL, 0               /DUMP TIMER                              /START LOAD ADDRESS
0072  0000  DUMPT,  0               /TEMPORARY REGISTER USED BY DUMP         /END LOAD ADDRESS
0073  0000  DUMPTI, 0               /TEMPORARY REGISTER USED BY DUMP         /NUMBER OF WORDS TO TRANSFER
              /
0074  0000  AUTOCT, 0               /COUNT FOR AUTOCAL TRIES
0075  0000  AUTOFL, 0               /DATA READY FLAG DURING AUTOCALIBRATION
0076  0000  TSW3F,  0               /START SWITCH ON FLAG.  FLAGS NEEDED TO MAKE SWITCH
0077  0000  TSW4F,  0               /POSITIVE EDGE TRIGGERED.  SAMPLE SWITCH ON FLAG.
0100  0000  PRPNT,  0               /POINTER TO NEXT PROCESS TO BE CHECKED
0101  0000  TSW8F,  0               /THRESHOLD FLAG IS BIDIRECTIONAL EDGE TRIGGERED
              /
              /                     /WIRED  PIN ALPHA 4  ALPHA T
                                    /IOT READ  BIT 10  BIT 9
0102  0000  PRNTER, 0               /DEV     WIRED    IOT READ
                                    /TR      00       11
                                    /DAT     01       01
                                    /WANG    10       10
                                    /SPARE   11       00
              /
0103  0000  DATPNT, 0               /HOLDS DATA POINTER DURING PRINT
0104  0000  PRPH,   0               /HOLDS ADDRESS IN MASK TABLE DURING PRINT
              BITOUT=22             /HOLDS ADDRESS OF OUT BIT ROUTINES
              TPSV=23               /HOLDS DATA WORD DURING PRINT
              PRSTCT=24             /COUNTS 51 BITS OF DATA
              PRCNT=25              /COUNTS 12 BITS PER WORD
              PRREM=26              /HOLDS REMAINDER POLYNOMIAL
              PRBYT=27              /BUILDS OUTPUT BYTE
              PRBYTC=30             /COUNTS 8 BITS PER BYTE
              PRPAR=31              /HOLDS PARITY FLAG
              DATOUT=32             /TEMPORARY STORAGE AND ALSO USED FOR DECIMAL POINT FOR PRINTER
              PRRET=33              /USED AS POINTER TO RETURN ADDRESS BUFFER
0105  0000  TRBLK,  0               /HOLDS BLOCK NUMBER FOR MINISAMPLE RECORDER
0106  0000  TFUN0,  0               /HAS LO FUNCTION REGISTER
0107  0000  TFUN1,  0               /HAS HI FUNCTION REGISTER
              /
              /
```

```
                /
                *114
                /
                /
                /SUBROUTINE CALL FOR ROM
                /
0114  0000      SUB1,   0               /HAS AC
0115  0000      SUB,    0
                /
                /
                /  ******   FROM THIS POINT TO TSWF WILL BE STORED IN PROM
                /
                /
0116  5517      PAGE0,  JMP I .+1
0117  6652              SUBX
0120  0467      STACK,  STK             /STACK POINTER
0121  6000      MEMAD,  6000            /HOLDS DEBUG ADDRESS
0122  0003      MEMFLD, 3               /HOLDS DEBUG FIELD
0123  6665      IRETN,  RETN            /POINTER TO POP ROUTINE
                /
                /
                GOSUB=JMS SUB
                RETURN=JMP I IRETN
                /
                DEFINE CALL XX
                <       GOSUB
                        XX
                >
                /
                /
0124  0000      MEMFLG, 0               /BIT MAP FOR MEMORY AND TAPE RECORDER CONTROL
                                        /BIT 11 = STORE DISPLAY BUF IN MEMORY
                                        /BIT 10 = USE DISPLAY BUFFER TO UPDATE KCAL
                                        /BIT  9 = PUT ENERGY BUFFER IN DISPLAY BUFFER
                                        /BIT  8 = PUT DISPLAY BUFFER IN ENERGY BUFFER
                                        /BIT  7 = PUT DISPLAY INTO LD OR LN BUFFER
                                        /BIT  6 = PUT DATA INTO DISPLAY BUFFER
                                        /BIT  5 = PUT LD AND LN IN LDN BUFFER AND MEMORY
                                        /BIT  3 = TURN ON TAPE RECORDER WHEN THRESHOLD IS EXCEEDED
                                        /BIT  2 = DO AUTOCAL
                                        /BIT  1 = DO CAL FOR KCAL1
                                        /BIT  0 = DO CAL FOR KCAL2
                /
                MODST=VDIS!STNBLD!PKCHN!12  /INITIAL MODE STATUS WORD
0125  4512      MODE,   MODST           /HAS MODE STATUS WORD
0126  4512      SVMD,   MODST           /HAS SAVED MODE.  REMEMBERS WHERE WE WERE
0127  6414      FORMAT, 6414            /FLAGS INDICATING INFORMATION STORED IN MEMORY BLOCK
0130  0005      RECON,  5               /MINISAMPLE ON TIME IN SECONDS
0131  0017      RECOFF, 17              /MINISAMPLE PERIOD IN MINUTES
0132  0000      DISPCH, 0               /CURRENT CHANNEL BEING DISPLAYED
                                        /0=NONE, 1=CH1, 2=CH2, 3=TIME, 4=PEAK
0133  0000      DISFL,  0               /0=NO DATA                      SET BY START AND SAMPLE
                                        /1=DATA DISPLAYED               SET BY DISUP
                                        /-1=DATA READY FOR DISPLAY      SET BY INTSERV
0134  0000      KCAL1,  0               /CALIBRATION CONSTANT.  ADD TO
0135  0000      KCAL2,  0               /NUMBER TO GET TRUE VALUE.
0136  3572      REFLV1, 3572            /HAS CH1 CALIBRATION LEVEL IN BINARY
0137  3572      REFLV2, 3572            /CH2 CALIBRATION LEVEL
                                        /35.72 BINARY IS 90.0 DB
```

```
                    /
0140  0001  TSEC,    1              /HAS TENTHS OF SECONDS IN BCD FORM
0141  0000  TMIN,    0              /HAS MINUTES IN BCD FORM
                                     /IF MIN=0, MUST BE SEC
0142  0000  CLSEC,   0              /SECONDS
0143  0000  CLMIN,   0              /MINUTES
0144  0000  CLHR,    0              /HOUR OF THE DAY (24 HOUR CLOCK)
0145  0000  CLDAY,   0              /DAY OF THE YEAR
0146  0000  THOLD,   0              /THRESHOLD FOR CH1 IN BINARY
0147  0000  THFLG,   0              /NEGATIVE NUMBER MEANS SUPPRESS 10 DB THRESHOLD
                                     /CHANGE DURING BLOCK MODE AT NIGHT
                    /
                    /
0150  0247  IDMP,    TDMPX          /POINTER TO DUMP ROUTINE
0151  0247  DMPX,    TDMPX          /POINTER TO RESET DUMP ROUTINE TO OFF
0152  2003  SW0A,    SW0            /POINTER TO INTERRUPT ROUTINE
                    /
                    /
0153  0354  TEMPEX, TEMPE-1         /SAVE SPACE
0154  7500  LOGTBL, LOGTAB
                    /
                    /
                    /CONSTANT TABLE
                    /
0155  7764  KM12,    -14
0156  0010  KD10,    10
0157  0017  KD17,    17
0160  0020  KD20,    20
0161  0077  KD77,    77
0162  0144  KD144,   144
0163  0320  KD320,   320
0164  7400  KD7400,  7400
0165  7700  KD7700,  7700
0166  7760  KD7760,  7760
                    /
                    DMASK,           /DOUBLE DUTY TABLE AND CONSTANTS
0167  4000  KD4000,  4000
0170  2000  KD2000,  2000
0171  1000  KD1000,  1000
0172  0400  KD400,   400
0173  0200  KD200,   200
0174  0100  KD100,   100
0175  0040  KD40,    40
0176  0000          0              /END OF TABLE
                    /
                    /
                    GRPNO=144        /GROUP NUMBER FOR CERL  ---------------
                    /
                    /
```

```
/
                /
                *177
                /
                /TIME ELAPSED FOR TRIP THROUGH SWSET IS ABOUT 1.6 MS
                /
0177  6622  SWSET,  FUNLO           /READ FRONT PANEL SWITCHES
0200  3106         DCA TFUN0       /SAVE
                /
0201  1075  TSW0,   TAD AUTOFL      /IS DATA READY FOR USE BY AUTOCAL?
0202  7710         SPA CLA
0203  7000         NOP             /WHILE IN THE INACTIVE STATE
                /   JMP I TSW0A     /GO DO SOMETHING WITH AUTO CALIBRATION
                TSW0X,
                /
0204  5210  TSW1,   JMP TSW1X       /IGNORE DURING PRINTER IDLE
                /   K2              /TEST FLAG DURING ACTIVE
0205  0106         AND TFUN0
0206  7640         SZA CLA
0207  5740         JMP I TSW1A     /FOUND FLAG
                TSW1X,
                /
                /CHECK START SWITCH
0210  1106  TSW3,   TAD TFUN0       /TEST FOR START SWITCH
0211  0156         AND KD10
0212  7640         SZA CLA
0213  5737         JMP I TSW3B
0214  3076         DCA TSW3F       /CLEAR EDGE TRIGGER FLAG
                TSW3X,
                /
0215  1106  TSW4,   TAD TFUN0       /TEST FOR SAMPLE SWITCH
0216  0160         AND KD20
0217  7640         SZA CLA
0220  5736         JMP I TSW4B
0221  3077         DCA TSW4F       /CLEAR EDGE TRIGGER FLAG
                TSW4X,
                /
0222  4274  TSW5,   JMS DBNCE       /START/STOP PRINT
0223  0040         40
0224  0000         0
0225  3704         SW5
                /
0226  1106  TSW7,   TAD TFUN0       /REAL TIME CLOCK
0227  0173         AND KD200       /CHECK FLAG BIT
0230  7640         SZA CLA
0231  5735         JMP I TSW7B     /GO PROCESS CLOCK
                /
0232  1106  TSW8,   TAD TFUN0       /CHECK THRESHOLD FLAG
0233  0172         AND KD400
0234  7640         SZA CLA
0235  5733         JMP I TSW8B
0236  1101         TAD TSW8F       /IS IT JUST TURNED OFF?
0237  7640         SZA CLA         /THIS ALLOWS NEGATIVE EDGE TRIGGER TOO
0240  5734         JMP I TSW8C
                TSW8X,
                /
                /
0241  1132  TDIS,   TAD DISPCH      /DISPLAY REQUEST?
0242  7640         SZA CLA         /NO. SKIP
0243  1133         TAD DISFL       /NEW DATA READY?
```

```
0244  7710          SPA CLA
0245  5730          JMP I DISSET    /YES.  GO UPDATE
                TDISX,
                /
                /
0246  5550  TDMP,   JMP I IDMP      /POINTS TO DUMP HANDLER WHEN ACTIVE
                TDMPX,
                /
0247  5253  TPOUT,  JMP TPOUTX      /BYPASS UNTIL NEEDED
                /   STATIN          /READ TAPE STATUS
0250  0174          AND KD100       /TRANSMIT STATUS
0251  7640          SZA CLA
0252  5732          JMP I TOUTB
                TPOUTX,
                /
0253  5257  TPIN,   JMP TPINX
                /   STATIN          /READ TAPE STATUS
0254  0173          AND KD200
0255  7640          SZA CLA
0256  5731          JMP I TINB      /BRING IN A CHARACTER
                TPINX,
                /
                /
                /GET FUNCTION SWITCH
                /
0257  1106  TFUN,   TAD TFUN0
0260  3107          DCA TFUN1
0261  6623          FUNHI
0262  3106          DCA TFUN0
                /
0263  4274  HTSW4,  JMS DBNCE       /IS EXECUTE SET?
0264  0020          20
0265  0000          0
0266  6561          HSW4
                /
0267  4274  HTSW5,  JMS DBNCE       /IS DISPLAY SET?
0270  0040          40
0271  0000          0
0272  6600          HSW5
                /
0273  5177          JMP SWSET
                /
                /
```

```
/
                /THIS ROUTINE DEBOUNCES THE SWITCHES
                /DEBOUNCE FLAG IS - FOR COUNTING
0274    0000    DBNCE,  0
0275    1674            TAD I DBNCE     /GET BIT FOR THIS CALL
0276    2274            ISZ DBNCE       /GET SET FOR FLAG
0277    0106            AND TFUN0       /LOOK AT SWITCHES
0300    7640            SZA CLA
0301    5306            JMP DBNCB
0302    3674            DCA I DBNCE     /CLEAR SWITCH FLAG
0303    2274    DBEX,   ISZ DBNCE
0304    2274            ISZ DBNCE
0305    5674            JMP I DBNCE
0306    1674    DBNCB,  TAD I DBNCE     /SWITCH BIT SET.  TEST FLAG
0307    7440            SZA
0310    5313            JMP DBNCC
0311    1327            TAD DBNCK       /SET UP DEBOUNCE
0312    5302            JMP DBEX-1
                /
0313    7700    DBNCC,  SMA CLA         /ARE WE COUNTING?
0314    5303            JMP DBEX        /IGNORE CONTINOUS ON
0315    2674            ISZ I DBNCE     /KEEP COUNTING
0316    5303            JMP DBEX
                /
0317    2674            ISZ I DBNCE     /DONE.  SET FLAG
0320    2274            ISZ DBNCE
0321    1674            TAD I DBNCE
0322    3274            DCA DBNCE
0323    3132            DCA DISPCH      /STOP READOUT
0324    1151            TAD DMPX        /STOP DUMP
0325    3150            DCA IDMP
0326    5674            JMP I DBNCE     /GO PROCESS
0327    7754    DBNCK,  -24             /ABOUT .08 SEC
                /
                /
0330    7324    DISSET, DISUP           /UPDATE DISPLAY ROUTINE
0331    1702    TINB,   TPINH           /ADDRESS OF TAPE INPUT HANDLER
0332    5242    TOUTB,  TAPOUT          /ADDRESS OF TAPE OUTPUT HANDLER
0333    5054    TSW8B,  SW8             /POINTER TO THRESHOLD ROUTINE
0334    5200    TSW8C,  SW8L            /POINTER TO THRESHOLD OFF ROUTINE
0335    4656    TSW7B,  SW7             /POINTER TO CLOCK SERVICE ROUTINE
0336    3600    TSW4B,  SW4             /POINTER FOR SAMPLE SWITCH ROUTINE
0337    3337    TSW3B,  SW3             /START SWITCH HANDLER
0340    3250    TSW1A,  SW1             /POINTER TO PRINTER HANDLER
0341    1310    TSW0A,  AUTOCAL         /POINTER TO AUTOCALIBRATION ROUTINE
0342    1426            AUTO2           /WAIT FOR THINGS TO SETTLE DOWN
0343    1426            AUTO2
0344    1431            AUTO1           /TABLE FOR AUTOCAL PROCEDURE
0345    1426            AUTO2           /3 SAMPLES FOR CALIBRATOR TO STABILIZE
0346    1426            AUTO2
0347    1426            AUTO2
0350    1461            AUTO3           /TAKE DATA AND ADJUST KCAL
0351    1426            AUTO2           /TAKE 2 SAMPLES TO KILL TIME AFTER CALIBRATION
0352    1426            AUTO2
0353    1612            AUTO4           /IF BACKGROUND IS OK, SET UP SHORT BLOCK
0354    1637            AUTO5           /ERROR.  SHORT BLOCK TOO SHORT
                /
                /
```

```
TSWF,
TEMPE,                  /HOLDING REGISTER FOR 7 WORD SHIFTING
/
/
*.+7
CH1E,                   /ACCUMULATOR FOR CH1 ENERGY
/
*.+6
CH2E,                   /ACCUMULATOR FOR CH2 ENERGY
/
*.+6
ACTIME,                 /ACCUMULATED TIME
/
*.+6
OPK,                    /OVERALL PEAK
/
*.+1
CH1LD,                  /DAY LEQ
/
*.+6
CH2LD,                  /DAY LEQ
/
*.+6
TMD,                    /LENGTH DAY
/
*.+6
CH1LN,                  /NIGHT LEQ
/
*.+6
CH2LN,                  /NIGHT LEQ
/
*.+6
TMN,                    /LENGTH NIGHT
/
*.+6
STAK,                   /SPACE FOR SOFTWARE SUBROUTINE STACK
/
*.+14
STK,
/
*.+1
PRBUF,                  /SPACE FOR 8 CLOCK CONTROLLED PROCESSES
PROC1=PRBUF             /TO INDICATE TIME OF PERFORMANCE
PROC2=PROC1+4
PROC3=PROC2+4
PROC4=PROC3+4
PROC5=PROC4+4
PROC6=PROC5+4
PROC7=PROC6+4
PROC8=PROC7+4
/
*PROC8+4
PRDAT,                  /BUFFER FOR PRINTER DATA
/
*.+PRLEN+1
PRSVR,                  /3RD LEVEL RETURN FOR PRINT
/
*.+1
TRDAT,                  /BUFFER FOR TAPE RECORDER DATA
```

```
/
*.+PRLEN+1
ITPDAT,
/
*.+PRLEN+1
LDNBOT,                 /SPACE FOR LDN BUFFER
/
*.+100
LDNTOP,
/
*.+1
BUFBG,                  /DATA BUFFER IS IN FIELD 1 STARTING AT ZERO AND ENDING AT MEMTOP
/
/DATA IS STORED IN BUFFER AS 6 BIT BINARY EXPONENT
/AND 6 BIT BINARY FRACTION.  THIS IS IN COMPRESSED
/FLOATING POINT FORMAT.
/
/
        PAGE
```

```
/
                /
                *1000
                PAGEZ,                  /THIS IS WHERE PAGE ZERO WILL GO
                /
                *PAGEZ+TSWF-PAGE0
1237  0000              0
                /
                /
                *1300
1300  4115  AUTOCM, CALL BCDBIN        /HANDLE MINUTES
1301  6625
1302  3021          DCA 21
1303  3020          DCA 20
1304  4115          CALL SW3FT+2
1305  3560
1306  3777          DCA PROC8           /ALLOW SYNCH ON MINUTE BOUNDARY
1307  5340          JMP AUTOC2          /CONTINUE
                /
                /
1310  1376  AUTOCAL, TAD (PROC8-1      /SET UP TIME FOR NEXT LONG BLOCK
1311  3015          DCA XRT3
1312  1141          TAD TMIN            /SEE HOW LONG
1313  7440          SZA
1314  5300          JMP AUTOCM
1315  1140          TAD TSEC            /MUST BE SECONDS.  CONVERT TO SECONDS
1316  0166          AND KD7760          /DIVIDE BCD BY TEN IS SHIFT RIGHT
1317  7112          CLL RTR
1320  7012          RTR
1321  4115          CALL BCDBIN
1322  6625
1323  3006          DCA 6
1324  1006          TAD 6
1325  1375          TAD (-50            /IF <40 SECONDS, WE GOT BIG TROUBLE
1326  7700          SMA CLA
1327  5335          JMP AUTOC1
                /
1330  4115          CALL ERROR4         /COMPLAIN
1331  3322
                /
1332  1374  AUTOF,  TAD (NOP            /STOP AUTOCAL
1333  3773          DCA TSW0+2
1334  5773          JMP TSW0+2          /CONTINUE OLD WAY
                /
1335  1006  AUTOC1, TAD 6               /GET SECONDS
1336  4115          CALL SW3FT          /SET UP PROCESS 8
1337  3556
                /
                /
                      DECIMAL
                /
1340  4115  AUTOC2, CALL ADDCLK         /SET UP PROCESS 2 FOR NEXT AUTOCAL
1341  5010
1342  0473          PROC2-1
1343  0025          21                  /5 HOURS 59 MINUTES 21 SECONDS
1344  0073          59
1345  0005          5
1346  0000          0
                /
                      OCTAL
```

```
                /
1347  1372              TAD (-6           /SET UP RETRY COUNTER
1350  3074              DCA AUTOCT
1351  1371              TAD (BUFBG-1      /SAVE DATA OUT OF THE WAY
1352  3013              DCA XRT1
1353  1124              TAD MEMFLG
1354  3413              DCA I XRT1
1355  1174              TAD KD100
1356  3124              DCA MEMFLG
1357  7240              STA               /FIX UP THRESHOLD
1360  6602              THSET
1361  1146              TAD THOLD
1362  3413              DCA I XRT1
1363  3146              DCA THOLD
                /
                /
1364  5770              JMP AUTOC3&7700
1370  1400              PAGE
1371  0672
1372  7772
1373  0203
1374  7000
1375  7730
1376  0523
1377  0524
                /
                /
1400  1777              TAD PROC6
1401  3413              DCA I XRT1        /SAVE PROCESS WORDS
1402  7330              K4000
1403  3777              DCA PROC6
1404  1776              TAD PROC7
1405  3413              DCA I XRT1
1406  7330              K4000
1407  3776              DCA PROC7
                /
1410  1141              TAD TMIN          /SAVE BLOCK LENGTH
1411  3413              DCA I XRT1
1412  3141              DCA TMIN          /SET NEW TIME OF 0.5 SEC
1413  1140              TAD TSEC
1414  3413              DCA I XRT1
1415  1375              TAD (5
1416  3140              DCA TSEC
1417  1140              TAD TSEC          /AND SET HARDWARE
1420  6600              SEC
                /
1421  6606  AUTOC3,  GOTDAT             /CLEAR FLAG FOR DATA
1422  6002              IOF
1423  6613              BEGIN             /LOAD TIME IN HARDWARE
1424  4115              CALL XSW3B        /SEE WHEN WE'RE DONE
1425  3305
1426  3075  AUTO2,    DCA AUTOFL        /CLEAR DATA FLAG
1427  2774              ISZ TSW0+2        /DON'T RETURN HERE
1430  5773              JMP TSW0X
                /
                /
1431  1372  AUTO1,    TAD (AUTO1A       /SET UP RETURN
1432  3007              DCA 7
1433  1136  AUTO1B,   TAD REFLV1        /CHECK THAT THE CURRENT LEVELS ARE LESS
1434  7161              CIA STL           /THAN 15 DB BELOW THE REFERENCE LEVEL
```

```
1435  1057          TAD CH1
1436  1371          TAD (477          /15 DB IN INTERNAL NOTATION
1437  7620          SNL CLA
1440  5770          JMP AUTODN
                    /
1441  6622          FUNLO             /ARE THERE TWO CHANNELS?
1442  0174          AND KD100
1443  7650          SNA CLA
1444  5407          JMP I 7
1445  1137          TAD REFLV2        /OK.  NOW CHECK OTHER CHANNEL
1446  7161          CIA STL
1447  1060          TAD CH2
1450  1371          TAD (477
1451  7620          SNL CLA
1452  5770          JMP AUTODN
1453  5407          JMP I 7
                    /
1454  4115  AUTO1A, CALL SETMD         /LEVELS ARE OK.  TURN ON CALIBRATOR
1455  5506
1456  6777          -CALREL-1
1457  1000          CALREL
1460  5226          JMP AUTO2
                    /
1461  1136  AUTO3,  TAD REFLV1         /ARE NEW VALUES WITHIN 0.7 DB
1462  7161          CIA STL
1463  1057          TAD CH1
1464  7430          SZL               /TAKE ABSOLUTE VALUE
1465  7141          CIA CLL
1466  1367          TAD (-17          /0.7 DB IN INTERNAL NOTATION
1467  7630          SZL CLA
1470  5770          JMP AUTODN
                    /
1471  6622          FUNLO             /CHECK FOR CHANNEL 2
1472  0174          AND KD100
1473  7650          SNA CLA
1474  5305          JMP AUTO3A
1475  1137          TAD REFLV2
1476  7161          CIA STL
1477  1060          TAD CH2
1500  7430          SZL
1501  7141          CIA CLL
1502  1367          TAD (-17
1503  7630          SZL CLA
1504  5770          JMP AUTODN
                    /
1505  1125  AUTO3A, TAD MODE           /OK ON LEVEL.  NOW CHECK PEAK
1506  0174          AND KD100
1507  7640          SZA CLA
1510  5320          JMP AUTO3B
1511  1061          TAD PK            /CHECK PEAK ON CHANNEL ONE
1512  7161          CIA STL
1513  1057          TAD CH1
1514  1173          TAD KD200         /6 DB IN INTERNAL NOTATION
1515  7630          SZL CLA
1516  5770          JMP AUTODN
1517  5332          JMP AUTO3C
                    /
1520  6622  AUTO3B, FUNLO             /MUST BE CHANNEL 2.  IS IT ACTIVE?
1521  0174          AND KD100
1522  7650          SNA CLA
```

```
1523  5332          JMP AUTO3C
1524  1061          TAD PK
1525  7161          CIA STL
1526  1060          TAD CH2
1527  1173          TAD KD200
1530  7630          SZL CLA
1531  5770          JMP AUTODN
                    /
1532  1057  AUTO3C, TAD CH1          /MEETS ALL CONDITIONS.  ADJUST KCALS
1533  7161          CIA STL
1534  1136          TAD REFLV1
1535  1134          TAD KCAL1
1536  3134          DCA KCAL1
                    /
1537  6622          FUNLO           /SEE ABOUT CHANNEL 2
1540  0174          AND KD100
1541  7650          SNA CLA
1542  5350          JMP AUTO3D
1543  1060          TAD CH2
1544  7161          CIA STL
1545  1137          TAD REFLV2
1546  1135          TAD KCAL2
1547  3135          DCA KCAL2
1550  4115  AUTO3D, CALL SETMD       /TURN OFF CAL TONE
1551  5506
1552  6777          -CALREL-1
1553  0000          0
1554  5226          JMP AUTO2       /WAIT A WHILE.  THEN FINISH UP
                    /
1567  7761          PAGE
1570  1600
1571  0477
1572  1454
1573  0204
1574  0203
1575  0005
1576  0520
1577  0514
                    /
1600  4115  AUTODN, CALL SETMD       /KILL CAL RELAY
1601  5506
1602  6777          -CALREL-1
1603  0000          0
1604  2074          ISZ AUTOCT      /HAVE WE TRIED ENOUGH?
1605  7410          SKP
1606  5215          JMP AUTO4A
1607  1377          TAD (5400+TSW0A /RESET JUMP TABLE POINTER
1610  3776          DCA TSW0+2
1611  5775          JMP AUTO2       /TAKE SAMPLES TO LET CAL TONE DIE
                    /
                    /
                    /
                    /
                    /
1612  1374  AUTO4,  TAD (AUTO4A     /SET UP RETURN FROM BACKGROUND CHECK
1613  3007          DCA 7
1614  5773          JMP AUTO1B
                    /
                    /
1615  1372  AUTO4A, TAD (BUFBG-1    /RESTORE DATA
```

```
1616  3013          DCA XRT1
1617  1413          TAD I XRT1
1620  3124          DCA MEMFLG
1621  1413          TAD I XRT1
1622  3146          DCA THOLD
1623  1413          TAD I XRT1
1624  3771          DCA PROC6
1625  1413          TAD I XRT1
1626  3770          DCA PROC7
1627  1413          TAD I XRT1
1630  3141          DCA TMIN
1631  1413          TAD I XRT1
1632  3140          DCA TSEC
1633  4115          CALL TTSET      /RESTORE HARDWARE
1634  5474
1635  5767          JMP AUTOC3      /SET UP DATA TAKING
                /
                /
                /
1636  6613  NAUTOC, BEGIN           /RESTART BLOCK ON TIME
1637  7330  AUTO5,  K4000
1640  3766          DCA PROC8       /NO MORE SHORT BLOCKS
1641  5765          JMP AUTOF       /CLEAN UP
                /
                /
                VLOPG,
                /
1765  1332          PAGE
1766  0524
1767  1421
1770  0520
1771  0514
1772  0672
1773  1433
1774  1615
1775  1426
1776  0203
1777  5741
                /
```

```
7653  1005          TAD 5           /NOW SEND BACK COMPLEMENT
7654  7040          CMA
7655  3407          DCA I 7         /STUFF BACK IN TEST FIELD
7656  1407          TAD I 7
7657  7001          IAC
7660  1005          TAD 5           /DO THEY COMPARE?
7661  7640          SZA CLA
7662  5271          JMP STRT3       /NO.  DONE.
7663  7100          CLL             /YES.  DO NEXT BLOCK
7664  1171          TAD KD1000
7665  1007          TAD 7
7666  3007          DCA 7
7667  7420          SNL
7670  5247          JMP STRT2       /NO.  KEEP LOOKING
                /
7671  1125  STRT3,  TAD MODE        /SET UP HARDWARE
7672  6607          SELMD
7673  1007          TAD 7           /GO BACK ONE STEP
7674  1374          TAD (-1000
7675  3056          DCA MEMTOP
                /
                /
7676  4115          CALL ACCLR      /CLEAR OUT ENERGY BUFFERS
7677  7703
7700  4115          CALL TTSET      /SET MINIMUM TIME AND THRESHOLD
7701  5474
7702  5767          JMP WFUN5+2     /SET ALL POINTERS FOR DATA BUFFERS
                /
7703  1366  ACCLR,  TAD (TEMPE-STAK /THIS CLEARS THE ENERGY BUFFER
7704  3011          DCA INT2
7705  1153          TAD TEMPEX
7706  3010          DCA INT1
7707  3410          DCA I INT1
7710  2011          ISZ INT2
7711  5307          JMP .-2
7712  5523          RETURN
                /
                HIPG,
                /
7766  7702          PAGE
7767  6044
7770  7740
7771  0467
7772  7541
7773  0115
7774  7000
7775  0777
7776  5552
7777  5200
```

```
/
                /
                /
                /SET LINK TABLE BELOW START LOCATION
7600    0377    *7600;  AND (JMP START
                /
                /THIS CODE DOES THE INITIALIZATION
                *7600
                START,
7600    6612            OUTHI           /TURN OFF POWER BIT OF PRINTER OR CASSETTE
7601    6607            SELMD           /CLEAR STATUS LIGHTS AND MEMORY CONTROL
7602    1376            TAD (JMP I SW0A /SET UP INTERRUPT JMP
7603    3001            DCA 1
                /
7604    1375            TAD (777        /TRANSFER DATA FROM FIELD 2 TO FIELD 0
7605    3010            DCA 10
7606    1010            TAD 10
7607    3011            DCA 11
7610    1374            TAD (-1000
7611    3007            DCA 7
                /
7612    1357    STRT1,  TAD X40
7613    6607            SELMD
7614    1410            TAD I 10
7615    3006            DCA 6
7616    6607            SELMD
7617    1006            TAD 6
7620    3411            DCA I 11
7621    2007            ISZ 7
7622    5212            JMP STRT1
                /
                /
                /PAGE 0 BLOCK STARTS AT PAGEZ
7623    1375            TAD (PAGEZ-1    /TRANSFER BLOCK TO PAGE 0
7624    3010            DCA 10
7625    1373            TAD (PAGE0-1
7626    3011            DCA 11
7627    1372            TAD (PAGE0-TSWF /# OF LOCS TO GO
7630    3007            DCA 7           /TO PAGE 0
7631    1410            TAD I 10
7632    3411            DCA I 11
7633    2007            ISZ 7
7634    5231            JMP .-3
                /
7635    1371            TAD (PRBUF-1    /CLEAR PROCESS CONTROL BUFFER
7636    3010            DCA 10
7637    1370            TAD (PRBUF-PRDAT
7640    3007            DCA 7
7641    7330            K4000
7642    3410            DCA I 10
7643    2007            ISZ 7
7644    5241            JMP .-3
                /
                /
7645    1375            TAD (777        /LOOK AT TOP WORD OF EACH 512 WORD BLOCK
7646    3007            DCA 7
7647    1160    STRT2,  TAD KD20        /SET FIELD TO 1
7650    6607            SELMD
7651    1407            TAD I 7         /GET RANDOM DATA
7652    3005            DCA 5
```

```
/
                /
                *2000
2000  4115              CALL ERRORS       /IN CASE OF RUNAWAY PROGRAM
2001  3321
2002  5177              JMP SWSET         /CONTINUE ANYWAY
                /
                /
                /THIS VERSION TAKES ABOUT .5 MS IN STANDBY
                /ABOUT 20 MS FOR DATA READING
                /ABOUT 35 MS FOR DATA READING AND ENERGY CALCULATION
                /
2003  3050      SW0,      DCA SAVAC         /SAVE AC
2004  7010                RAR               /AND LINK
2005  3051                DCA SAVLK
2006  7521                SWP               /GET MQ
2007  3052                DCA SAVMQ         /AND SAVE IT TOO!
2010  1115                TAD SUB           /SAVE STACK CALL
2011  3053                DCA SVSUB
2012  1114                TAD SUB1          /SAVE STACK TEMP LOCATION
2013  3054                DCA SVSUB1
                /
2014  1124                TAD MEMFLG        /IS THERE ANYTHING TO DO?
2015  7650                SNA CLA
2016  5777                JMP SW0F          /NO.  SKIP EVERYTHING
                /
2017  7305                K2                /GET TIME
2020  6603                SELCHN
2021  3062                DCA TIME          /CLEAR OUT TIME
2022  3010                DCA INT1          /CLEAR OUT KCAL
2023  1376                TAD (ACTIME-1
2024  4115                CALL ADDE         /ACCUMULATE TIME
2025  2674
2026  3062                DCA TIME          /LOG2 FORM - RANGE 31.98
                /
2027  6625                PEAK              /GET PEAK VALUE
2030  3061                DCA PK
2031  1155                TAD KM12          /SET UP MAXIMUM NUMBER OF SHIFTS
2032  3060                DCA TEMP1         /WHILE GETTING THE PEAK VALUE
2033  1375                TAD (777          /GET ONLY THE A/D BITS
2034  0061                AND PK
2035  7104                CLL RAL           /THIS NORMALIZES THE A/D BITS
2036  7430                SZL
2037  5242                JMP .+3           /FOUND HIGH ORDER BIT
2040  2060                ISZ TEMP1         /KEEP COUNTING
2041  5235                JMP .-4
                /
2042  7002                BSW               /OK.  GET 6 HIGH ORDER BITS
2043  0375                AND (777
2044  3133                DCA TEMP2
2045  1061                TAD PK            /GET RANGE BITS AND CONVERT TO EXPONENT
2046  7012                RTR
2047  7002                BSW
2050  0374                AND (16
2051  3061                DCA PK
2052  1060                TAD TEMP1         /GET SHIFT COUNT AND CONVERT TO EXPONENT
2053  7440                SZA               /WATCH OUT FOR A 0 PEAK
2054  7040                CMA
2055  1061                TAD PK            /COMBINED EXPONENT
2056  7002                BSW               /STUFF IN HIGH ORDER
```

```
2057  1133          TAD TEMP2       /NOW IN SHIFT REGISTER FORMAT
2060  4115          CALL LOGCON+1   /NOW CONVERT TO LOG2
2061  2660
2062  7104          CLL RAL         /SQUARE TO GET DB TO READ RIGHT
2063  3061          DCA PK
                    /
2064  1125          TAD MODE        /ADJUST FOR CALIBRATION
2065  0174          AND KD100       /SEE WHICH CHANNEL
2066  7640          SZA CLA
2067  5272          JMP .+3         /MUST BE CH2
2070  1134          TAD KCAL1       /THIS TIME IT'S CHANNEL 1
2071  7410          SKP
2072  1135          TAD KCAL2
2073  7100          CLL
2074  7510          SPA
2075  7020          CML             /DO 13 BIT ADDITION
2076  1061          TAD PK
                    /
                    /
2077  5773          JMP SW0P
                LOPG,
2173  2200          PAGE
2174  0016
2175  0777
2176  0377
2177  2441
                SW0P,
                    /
2200  7430          SZL
2201  7200          CLA             /DON'T ALLOW NEGATIVE PEAK LEVEL
2202  3061          DCA PK          /SAVE
2203  1061          TAD PK          /UPDATE OVERALL PEAK
2204  7161          STL CIA
2205  1777          TAD OPK
2206  7620          SNL CLA
2207  5212          JMP .+3
2210  1061          TAD PK          /NEW IS GREATER THAN OLD.
2211  3777          DCA OPK         /DISCARD OLD
2212  6603          SELCHN          /GET CHAN 1
2213  1134          TAD KCAL1
2214  3010          DCA INT1        /SEND CAL ALONG
2215  1376          TAD (CH1E-1
2216  4115          CALL ADDE       /ACCUMULATE CH1 ENERGY
2217  2674
2220  3057          DCA CH1         /LOG2 FORM - RANGE 63.98
                    /
2221  7001          IAC
2222  6603          SELCHN          /GET CHAN 2
2223  1135          TAD KCAL2
2224  3010          DCA INT1        /SEND CAL ALONG
2225  1375          TAD (CH2E-1
2226  4115          CALL ADDE       /ACCUMULATE CH2 ENERGY
2227  2674
2230  3060          DCA CH2         /LOG2 FORM - RANGE 63.98
                    /
2231  6622          FUNLO           /DUAL SINGLE CORR.
2232  0174          AND KD100       /ADDING 100 IS EQUAL TO TWICE THE TIME
2233  1062          TAD TIME
2234  1374          TAD (-1747      /MAKE REFERENCE=1 SECOND
2235  3062          DCA TIME        /LOG2(50000) = 15.6096
```

```
                /
                /
2236  1171          TAD KD1000      /SHOULD WE PUT ENERGY IN DISPLAY?
2237  0124          AND MEMFLG
2240  7650          SNA CLA
2241  5277          JMP SW0B
                /
2242  1777          TAD OPK         /SAVE OVERALL PEAK IN DISPLAY
2243  3061          DCA PK
                /
2244  1373          TAD (RADTB1-1   /GET DATA IN INTERNAL LOG FORM
2245  4115          CALL RADDE
2246  3135
                /
2247  1124  SW0A0,   TAD MEMFLG      /ARE WE TO ADJUST CAL?
2250  7012          RTR             /ALIGN TO SAVE CODE
2251  0172          AND KD400
2252  7650          SNA CLA
2253  5272          JMP SW0A2
2254  7420          SNL             /IS IT CHANNEL 1?
2255  5262          JMP SW0A1
2256  1057          TAD CH1         /YES.  GET MEASURED VALUE
2257  7041          CIA
2260  1136          TAD REFLV1      /COMPARE TO EXPECTED VALUE
2261  3134          DCA KCAL1       /AND UPDATE GAIN CONSTANT
                /
2262  7001  SW0A1,   IAC
2263  0124          AND MEMFLG      /IS IT CHANNEL 2?
2264  7650          SNA CLA
2265  5272          JMP SW0A2
2266  1060          TAD CH2
2267  7041          CIA
2270  1137          TAD REFLV2
2271  3135          DCA KCAL2
                /
2272  1124  SW0A2,   TAD MEMFLG      /SAVE FLAG BITS FOR THE MOMENT
2273  3010          DCA INT1
2274  3124          DCA MEMFLG      /GO TO STANDBY
2275  1010          TAD INT1        /GET BITS BACK
2276  5303          JMP .+5         /SKIP SAMPLE TEST
                /
                /
2277  1063  SW0B,    TAD SMPIG       /SAMPLE IN BLOCK MODE?
2300  7640          SZA CLA
2301  5772          JMP SW0E        /YES.  DON'T PUT IN BUFFER
2302  1124          TAD MEMFLG      /PUT IN MEMORY?
2303  7700          SMA CLA
2304  5345          JMP SW0D1       /DON'T PUT DATA IN MEM
                /
2305  1041          TAD EMPTY       /IS BUFFER FULL?
2306  7740          SMA SZA CLA
2307  5345          JMP SW0D1       /YES.  IGNORE CURRENT DATA
                /
2310  1144          TAD CLHR        /GET HOURS
2311  7002          BSW             /STUFF IN HIGH
2312  3012          DCA INT3        /=64
2313  1012          TAD INT3
2314  7110          CLL RAR         /=32
2315  1012          TAD INT3        /=96
2316  3012          DCA IN13
```

```
2317  1144          TAD CLHR
2320  7106          CLL RTL           /=4
2321  1012          TAD INT3          /=100
2322  1143          TAD CLMIN         /COMBINE TIMES IN ONE WORD
2323  3012          DCA INT3          /STUFF IN TEMP
2324  4115          CALL SW0B2        /PUT DATA IN MEMORY
2325  2516
2326  0167          DMASK
2327  2450          IMT-DMASK-1
2330  0020          20                /IN FIELD ONE
2331  2036          ISZ INFLO         /END OF BUFFER?
2332  5337          JMP SW0D          /GO CONTINUE TESTING
                    /
2333  1034          TAD STORE         /RESET COUNTER
2334  3036          DCA INFLO
2335  7240          STA               /RESET XR
2336  3017          DCA XR17
                    /
2337  1036  SW0D,   TAD INFLO        /BUFFER FULL?
2340  7041          CIA
2341  1040          TAD OUTFLO
2342  7650          SNA CLA
2343  7001          IAC               /YES. SET FLAG
2344  3041          DCA EMPTY
                    /
                    /
2345  1124  SW0D1,  TAD MEMFLG       /SHOULD WE DO LDN?
2346  0175          AND KD40
2347  7450          SNA
2350  5772          JMP SW0E
2351  7040          CMA
2352  0124          AND MEMFLG        /OK.  BUT KILL FLAG BIT.  SET ONLY ONCE A DAY
2353  3124          DCA MEMFLG
                    /
2354  1047          TAD LDNMT         /BUFFER MAY BE FULL
2355  7740          SMA SZA CLA
2356  5772          JMP SW0E
                    /
                    /
2357  1376          TAD (CH1E-1       /COMPUTE LDN FOR CH1
2360  4115          CALL SW0LDN
2361  2465
2362  1375          TAD (CH2E-1       /COMPUTE LDN FOR CH2
2363  4115          CALL SW0LDN
2364  2465
                    /
2365  1062          TAD TIME          /SAVE TIME FOR DISPLAY
2366  3776          DCA CH1E-1        /OUT OF THE WAY
                    /
                    /
2367  5771          JMP SW0PP
2371  2400          PAGE
2372  2434
2373  2577
2374  6031
2375  0371
2376  0363
2377  0406
              SW0PP,
                    /
```

```
2400  1377
2401  4115          TAD (RADTB2-1
2402  3135          CALL RADDE      /FIXUP LDN DATA INTO INTERNAL LOG FORM
2403  4115
2404  3136          CALL RADDE+1
2405  4115
2406  3136          CALL RADDE+1
2407  1776
2410  3062          TAD CH1E-1
                    DCA TIME        /RETRIEVE TIME FOR DISPLAY
                /
2411  4115  SW0D2,  CALL SW0B2       /PUT DATA IN LDN MEMORY
2412  2516
2413  7157
2414  3467          LMASK
2415  0000          IMTL-LMASK-1
                    0               /IN FIELD ZERO
                /
2416  4115
2417  7703          CALL ACCLR
                /
2420  2044
2421  5226          ISZ LDNIN       /END OF BUFFER?
2422  1042          JMP SW0D3       /GO CONTINUE TESTING
2423  3044          TAD LDNST       /RESET COUNTER
                    DCA LDNIN
                /
2424  1375
2425  3016          TAD (LDNBOT     /RESET POINTER
                    DCA LDN17
                /
2426  1044  SW0D3,  TAD LDNIN       /IS BUFFER FULL?
2427  7041          CIA
2430  1046          TAD LDNOT
2431  7650          SNA CLA
2432  7001          IAC
2433  3047          DCA LDNMT
                /
2434  3063  SW0E,   DCA SMPIG       /CLEAR IGNORE FLAG
2435  7240          STA
2436  3133          DCA DISFL       /TELL PROGRAM THAT DATA IS AVAILABLE
2437  7240          STA
2440  3075          DCA AUTOFL      /SET FLAG FOR AUTOCAL
                /
2441  7307  SW0F,   K4
2442  0124          AND MEMFLG      /ARE WE TO DO AUTO CAL?
2443  7450          SNA
2444  5252          JMP SW0F1
2445  7040          CMA
2446  0124          AND MEMFLG      /YES.  BUT DON'T OVER DO IT
2447  3124          DCA MEMFLG
2450  1374          TAD (5400+TSW0A /SET UP JUMP FOR AUTO CAL
2451  3773          DCA TSW0+2
                /
2452  1054  SW0F1,  TAD SVSUB1      /RESTORE SUBROUTINE CALL REGISTERS
2453  3114          DCA SUB1
2454  1053          TAD SVSUB
2455  3115          DCA SUB
2456  1052          TAD SAVMQ       /RESTORE MQ
2457  7421          MQL
2460  6606          GOTDAT          /CLEAR SYNCH F-F
2461  1051          TAD SAVLK       /RESTORE REGISTERS
2462  6005          RTF             /RESTORE LINK AND CLEAR AC
2463  1050          TAD SAVAC
```

```
2464  5400          JMP I 0          /RETURN TO WHERE WE CAME FROM
                    /
                    /
2465  3012  SW0LDN, DCA INT3         /SAVE DESTINATION ADDRESS
2466  1372          TAD (-6          /TO COMPUTE LDN, FIRST CLEAR RESULT REGISTER
2467  3133          DCA TEMP2
2470  3412          DCA I INT3
2471  2133          ISZ TEMP2
2472  5270          JMP .-2
                    /
2473  1371          TAD (-12         /MAKE LDN 10DB LOUDER
2474  3055          DCA TEMP3
2475  1372  SW0LNA, TAD (-6
2476  1012          TAD INT3         /RESTORE ADDRESS
2477  3012          DCA INT3
2500  1370          TAD (CH1LN-CH1E /GET ADDEND ADDRESS
2501  1012          TAD INT3
2502  4115          CALL ADDE7+1     /ADD IT ALL UP
2503  3110
2504  2055          ISZ TEMP3
2505  5275          JMP SW0LNA
                    /
2506  1372          TAD (-6
2507  1012          TAD INT3         /RESTORE ADDRESS
2510  3012          DCA INT3
2511  1367          TAD (CH1LD-CH1E /ADD IN DAY
2512  1012          TAD INT3
2513  4115          CALL ADDE7+1
2514  3110
2515  5523          RETURN
                    /
                    /
                    /
2516  7240  SW0B2,  STA
2517  1520          TAD I STACK
2520  3010          DCA INT1         /GET POINTER TO ARGUMENT LIST
2521  2120          ISZ STACK        /POP ADDRESS OFF STACK
2522  1410          TAD I INT1       /GET DATA TABLE ADDRESS
2523  3055          DCA TEMP3
2524  1410          TAD I INT1       /GET OFFSET
2525  3011          DCA INT2
2526  1410          TAD I INT1       /GET FIELD OF DATA
2527  3115          DCA SUB          /SAVE IN SUB LEGAL HERE
2530  1455  SW0B2A, TAD I TEMP3      /GET MASK FROM TABLE
2531  7450          SNA
2532  5410          JMP I INT1       /0.  DONE
2533  0127          AND FORMAT       /GET APPROPRIATE BITS
2534  7041          CIA
2535  1455          TAD I TEMP3      /MULTIPLE BITS ARE ALLOWED
2536  2055          ISZ TEMP3        /ADVANCE POINTER
2537  7640          SZA CLA
2540  5330          JMP SW0B2A
2541  1055          TAD TEMP3        /FOUND A MATCH.  GET ADDRESS OF DATA
2542  1011          TAD INT2         /GET OFFSET
2543  3133          DCA TEMP2
2544  1533          TAD I TEMP2      /GET ADDRESS OF DATA
2545  3133          DCA TEMP2
2546  1533          TAD I TEMP2      /GET DATA
2547  3133          DCA TEMP2        /SAVE IT
2550  1115          TAD SUB          /GET FIELD
```

```
2551  7450          SNA
2552  5362          JMP SW0B2B
2553  1125          TAD MODE
2554  6607          SELMD                 /SET HARDWARE
2555  1133          TAD TEMP2             /RETREIVE DATA
2556  3417          DCA I XR17
2557  1125          TAD MODE              /RESTORE FIELD
2560  6607          SELMD
2561  5330          JMP SW0B2A
                    /
                    /
2562  1133  SW0B2B,  TAD TEMP2            /STUFF IN LDN BUFFER
2563  3416          DCA I LDN17
2564  5330          JMP SW0B2A
                    /
                    /
2567  0023          PAGE
2570  0045
2571  7766
2572  7772
2573  0203
2574  5741
2575  0572
2576  0363
2577  2607
                    /
                    /
2600  0406  RADTB1,  ACTIME+6
2601  0062          TIME
2602  0372          CH1E+6
2603  0057          CH1
2604  0400          CH2E+6
2605  0060          CH2
2606  0000          0
2607  0062          TIME
                    /
2610  0406  RADTB2,  ACTIME+6
2611  0062          TIME
2612  0372          CH1E+6
2613  0364          CH1E
2614  0400          CH2E+6
2615  0372          CH2E
2616  0000          0
2617  0400          ACTIME
                    /
2620  0431          TMD+6
2621  0062          TIME
2622  0415          CH1LD+6
2623  0407          CH1LD
2624  0423          CH2LD+6
2625  0415          CH2LD
2626  0000          0
2627  0423          TMD
                    /
2630  0453          TMN+6
2631  0062          TIME
2632  0437          CH1LN+6
2633  0431          CH1LN
2634  0445          CH2LN+6
2635  0437          CH2LN
```

```
2636  0000          0
2637  0445          TMN
                    /
                    /
2640  0057  IMT,    CH1
2641  0060          CH2
2642  0061          PK
2643  0062          TIME
2644  0134          KCAL1
2645  0135          KCAL2
2646  0012          INT3
                    /
                    /
2647  0364  IMTL,   CH1E              /CH1 LDN
2650  0372          CH2E              /CH2 LDN
2651  0423          TMD
2652  0445          TMN
2653  0407          CH1LD
2654  0431          CH1LN
2655  0415          CH2LD
2656  0437          CH2LN
                    /
                    /
                    /
                    /
                    TEMP1=CH2
                    TEMP2=DISFL
                    /
2657  6624  LOGCON, OUTREG            /GET DATA
2660  3055          DCA TEMP3         /SAVE NUMBER
2661  1055          TAD TEMP3         /GET FRACTIONAL PART
2662  0161          AND KD77
2663  1154          TAD LOGTBL        /FORM TABLE ADDRESS
2664  3133          DCA TEMP2
2665  1055          TAD TEMP3         /GET HIGH BITS
2666  0165          AND KD7700
2667  3055          DCA TEMP3
2670  1533          TAD I TEMP2       /ADD LOW BITS
2671  0161          AND KD77
2672  1055          TAD TEMP3
2673  5523          RETURN
                    /
                    /
```

```
/
                /
2674  3012  ADDE,   DCA INT3          /SAVE POINTER TO RESULT
2675  1010          TAD INT1          /INITIALIZE DOUBLE PRECISION ADD
2676  7710          SPA CLA
2677  7240          STA
2700  3011          DCA INT2          /EXTENDED SIGN FOR KCAL
2701  4115          CALL LOGCON       /GET VALUE AND CONVERT TO LOG2
2702  2657
2703  7100          CLL               /DO 14 BIT ADD
2704  1010          TAD INT1          /KCAL
2705  7430          SZL
2706  2011          ISZ INT2
2707  7000          NOP
2710  3055          DCA TEMP3         /SAVE CALIBRATED ENERGY
2711  1011          TAD INT2          /HOW IS SIGN OF RESULT?
2712  7710          SPA CLA
2713  5523          RETURN            /NEG.  SKIP EVERYTHING!!
                /
2714  1011          TAD INT2          /CONVERT LOG2 BACK TO NUMERIC
2715  7110          CLL RAR           /SAVE OVERFLOW BIT IN LINK
2716  1055          TAD TEMP3
2717  7010          RAR               /CONVERT TO LOG4
2720  3133          DCA TEMP2
                /
2721  1011          TAD INT2          /SET UP 13 BIT SUBTRACT
2722  7110          CLL RAR
2723  1062          TAD TIME
2724  7061          CML CIA
2725  1055          TAD TEMP3         /ADD IN ENERGY
2726  7430          SZL
2727  7200          CLA               /KILL FOR NEGATIVE RESULT
2730  3055          DCA TEMP3         /NOW HAS LEQ
                /
2731  1172          TAD KD400         /TEST FOR SOFTWARE ACCUMULATOR
2732  0124          AND MEMFLG
2733  7650          SNA CLA
2734  5777          JMP ADDE6
                /
2735  1063          TAD SMPIG         /DON'T ACCUMULATE FOR THIS CASE
2736  7640          SZA CLA
2737  5777          JMP ADDE6
                /
2740  1133          TAD TEMP2         /CONVERT BACK TO NUMERIC USING
2741  0161          AND KD77          /ANTILOG4 TABLE
2742  1154          TAD LOGTBL
2743  3060          DCA TEMP1         /THIS NOW SAVES ADDRESS OF DATA
                /
2744  1133          TAD TEMP2         /GET EXPONENT
2745  7002          BSW
2746  0161          AND KD77
2747  7104          CLL RAL           /COMPUTE NUMBER OF SHIFTS TO NORMALIZE
2750  1376          TAD (2            /ADD HARDWARE CONSTANT
2751  3133          DCA TEMP2
                /
                /
2752  1153          TAD TEMPEX        /SET UP POINTERS
2753  3010          DCA INT1
2754  1375          TAD (-7           /FOR NORMALIZATION USE EXTRA REGISTER
2755  3011          DCA INT2
```

```
                /
2756  3410      DCA I INT1        /CLEAR OUT THE REGISTER
2757  2011      ISZ INT2
2760  5356      JMP .-2
                /
2761  1153      TAD TEMPEX        /NEXT DO A WORD BY WORD SHIFT FOR SPEED
2762  3010      DCA INT1
2763  1133      TAD TEMP2         /RETRIEVE COUNT
2764  1155      TAD KM12          /DIVIDE BY 12 BITS PER WORD
2765  7510      SPA
2766  5371      JMP .+3           /DONE
2767  2010      ISZ INT1          /ADVANCE POINTER
2770  5364      JMP .-4           /AND TRY AGAIN
                /
2771  5774      JMP ADDE1&7600
2774  3000      PAGE
2775  7771
2776  0002
2777  3077
                /
3000  1377      TAD C14           /RESTORE
3001  7040      CMA
3002  3133      DCA TEMP2         /NOW WE HAVE REMAINDER.
                /
3003  7621      CAM               /2 WORD LONG BIT SHIFT
3004  1460      TAD I TEMP1       /GET VALUE FROM TABLE
3005  0165      AND KD7700
                /
3006  2133  ADDE1,  ISZ TEMP2     /DONE?  THIS ROUTINE TAKES 2 MS WORST CASE
3007  7410      SKP
3010  5216      JMP ADDE2
3011  7104      CLL RAL           /DOUBLE PRECISION SHIFT
3012  7521      SWP
3013  7004      RAL
3014  7521      SWP
3015  5206      JMP ADDE1
                /
                /
3016  3410  ADDE2,  DCA I INT1     /PUT AWAY LOW ORDER
3017  7521      SWP
3020  3410      DCA I INT1        /AND HIGH ORDER
                /
3021  4115      CALL ADDE7        /UPDATE SOFTWARE ACCUMULATOR
3022  3107
3023  1173      TAD KD200         /DO WE DO LDN?
3024  0124      AND MEMFLG
3025  7650      SNA CLA
3026  5277      JMP ADDE6
                /
3027  1006      TAD 6             /SAVE 6 AND 7 TO ALLOW USE OF BCDBIN
3030  3060      DCA TEMP1
3031  1007      TAD 7
3032  3133      DCA TEMP2
                /
3033  1140      TAD TSEC          /YES.  WHICH ONE DAY OR NIGHT?
3034  7450      SNA
3035  5301      JMP ADDEM
3036  0166      AND KD7760        /DIVIDE BCD BY 10
3037  7112      CLL RTR
3040  7012      RTR
```

```
3041  4115            CALL BCDBIN
3042  6625
3043  7110
3044  7041            CLL RAR             /DIVIDE TIME INTERVAL BY 2
3045  1142            CIA
3046  7700            TAD CLSEC           /DOES MID POINT OF INTERVAL MEET THE REQUIREMENT?
3047  5254            SMA CLA
3050  7240            JMP ADDES2
                      STA
3051  1143  ADDES1,   TAD CLMIN          /HANDLE CARRY IF NEEDED
3052  7710            SPA CLA
3053  7240            STA
3054  1144  ADDES2,   TAD CLHR
3055  1376            TAD (-7             /7 AM OR AFTER
3056  7510            SPA
3057  5265            JMP ADDES3
3060  1375            TAD (-17            /LESS THAN 10 PM
3061  7700            SMA CLA
3062  5265            JMP ADDES3
                /
3063  1374            TAD (CH1LD-CH1E-6       /ADD TO DAY
3064  5267            JMP .+3
3065  7200  ADDES3,   CLA
3066  1373            TAD (CH1LN-CH1E-6       /ADD TO NIGHT
3067  1012            TAD INT3            /CURRENT ADDRESS OF DATA JUST ENTERED
3070  3012            DCA INT3            /NOW HAVE LDN BUFFER ADDRESS
3071  1133            TAD TEMP2
3072  3007            DCA 7               /RESTORE 6 AND 7 FOR BCDBIN
3073  1060            TAD TEMP1
3074  3006            DCA 6
3075  4115            CALL ADDE7
3076  3107
                /
3077  1055  ADDE6,    TAD TEMP3           /GET DATA
3100  5523            RETURN              /AND EXIT
                /
3101  1141  ADDEM,    TAD TMIN            /NOT SECONDS SO MUST BE MINUTES
3102  4115            CALL BCDBIN
3103  6625
3104  7110            CLL RAR
3105  7041            CIA
3106  5251            JMP ADDES1          /CHECK FOR MIDPOINT
                /
                /
3107  1372  ADDE7,    TAD (TEMPE          /SET UP MULTIPLE PRECISION ADD FROM TEMP TO RESULT
3110  3010            DCA INT1
3111  1012            TAD INT3
3112  3011            DCA INT2
3113  1371            TAD (-6
3114  3133            DCA TEMP2
3115  7100            CLL
                /
                /
[illegible]   ADDE8,    RAL                 /GET CARRY
[illegible]             TAD I INT1          /GET AUGEND
[illegible]             TAD I INT2          /GET ADDEND
[illegible]             DCA I INT3          /STUFF AWAY SUM
[illegible]             ISZ TEMP2           /DO ALL WORDS
[illegible]             JMP ADDE8
                        RETURN
```

```
/
                /
3125  1412  RADDE0, TAD I INT3      /WHERE DO WE PUT TIME?
3126  3055          DCA TEMP3
3127  6622          FUNLO           /GET DUAL SINGLE CORRECTION
3130  0174          AND KD100
3131  1370          TAD (-1747      /CONVERT TO DB SECONDS
3132  1062          TAD TIME
3133  3455          DCA I TEMP3
3134  5523          RETURN
                /
                /
3135  3012  RADDE,  DCA INT3        /SAVE TABLE ADDRESS
3136  3062          DCA TIME        /CLEAR OUT TIME TO EVALUATE TIME
3137  1412          TAD I INT3      /IS TABLE VALUE ZERO?
3140  7450          SNA
3141  5325          JMP RADDE0      /ZERO MEANS END OF TABLE
3142  3055          DCA TEMP3       /SAVE ADDRESS OF BUFFER
3143  1412          TAD I INT3      /GET ADDRESS OF RESULT
3144  3133          DCA TEMP2       /SAVE OUT OF THE WAY
                /
3145  1367          TAD (-107       /MAX NUMBER OF SHIFTS IN BUFFER
3146  3010          DCA INT1
                /
3147  7240  RADDE1, STA             /DECREMENT ADDRESS POINTER
3150  1055          TAD TEMP3
3151  3055          DCA TEMP3
3152  1455          TAD I TEMP3     /IS HIGH ORDER 0?
3153  7440          SZA
3154  5766          JMP RADDE2
3155  1377          TAD (14         /12 BITS PER WORD
3156  1010          TAD INT1
3157  3010          DCA INT1
3160  1010          TAD INT1        /ARE WE DONE?
3161  7710          SPA CLA
3162  5347          JMP RADDE1
3163  5765          JMP RADDE5      /YEP. BUFFER IS EMPTY
                /
3165  3246          PAGE
3166  3200
3167  7671
3170  6031
3171  7772
3172  0355
3173  0037
3174  0015
3175  7761
3176  7771
3177  0014
                /
3200  7421  RADDE2, MQL             /SET UP TWO WORD SHIFT
3201  7240          STA
3202  1055          TAD TEMP3
3203  3055          DCA TEMP3
3204  1455          TAD I TEMP3
                /
3205  7104  RADDE3, CLL RAL         /SHIFT LOW ORDER
3206  7521          SWP             /GET HIGH ORDER
3207  7004          RAL             /PUT BIT FROM LOW INTO HIGH
3210  7420          SNL             /HAVE WE FOUND THE SIGNIFICANT 1?
```

```
3211  5243          JMP RADDE4
                    /
3212  7002          BSW               /YES.  GET 6 BIT DATA
3213  0161          AND KD77
3214  1154          TAD LOGTBL        /MAKE LOG FORMAT
3215  3055          DCA TEMP3
3216  1455          TAD I TEMP3
3217  0161          AND KD77
3220  3011          DCA INT2
3221  1010          TAD INT1          /GET EXPONENT
3222  7041          CIA
3223  0161          AND KD77
3224  7002          BSW
3225  1011          TAD INT2          /COMBINE
3226  3011          DCA INT2
                    /
3227  1010          TAD INT1          /SET UP 13 BIT SUBTRACT
3230  7041          CIA
3231  0174          AND KD100
3232  7100          CLL
3233  7640          SZA CLA           /GET HIGH ORDER BIT IN LINK
3234  7020          CML
3235  1062          TAD TIME
3236  7061          CIA CML
3237  1011          TAD INT2
3240  7430          SZL
3241  7200          CLA               /DON'T LET NEGATIVE RESULT IN.
3242  5246          JMP RADDE5
                    /
                    /
3243  7521  RADDE4, SWP
3244  2010          ISZ INT1
3245  5205          JMP RADDE3
                                      /HIT UNDERFLOW
                    /
                    /
3246  3533  RADDE5, DCA I TEMP2       /PUT RESULT AWAY
3247  5777          JMP RADDE+2       /GET NEXT TABLE ENTRY
                    /
                    /
```

```
/
                /
3250  4115  SW1,    CALL PRSET      /SET UP BUFFER POINTERS
3251  3255
3252  1433          TAD I PRRET     /GO TO WHEREVER WE LEFT OFF.
3253  3007          DCA 7
3254  5407          JMP I 7
                /
                /
3255  1375  PRSET,  TAD ((OUTBIT    /SET UP PRINTER POINTER
3256  3022          DCA BITOUT
3257  1374          TAD (PRDAT      /SET UP FIRST ADDRESS IN BUFFER
3260  3007  PRSET1, DCA 7
3261  1373          TAD (BITOUT     /ADDRESS OF POINTER ARRAY
3262  3013          DCA XRT1
3263  1372          TAD (1-PRLEN
3264  3006          DCA 6
3265  1007  PRSET2, TAD 7
3266  3413          DCA I XRT1
3267  2007          ISZ 7
3270  2006          ISZ 6
3271  5265          JMP PRSET2
3272  5523          RETURN
                /
                /
                /
```

```
/
                /
                /
3273  6002  XSW3A,  IOF
3274  6606          GOTDAT
3275  6613          BEGIN             /RESET COUNTERS
3276  1371          TAD (TEMPE-CH1LD  /CLEAR OUT 72 BIT ACCUMULATORS
3277  3007          DCA 7
3300  1153          TAD TEMPEX
3301  3013          DCA XRT1
3302  3413          DCA I XRT1
3303  2007          ISZ 7
3304  5302          JMP .-2
3305  4115  XSW3B,  CALL DISTST      /CLEAR DISPLAY
3306  3676
3307  3063          DCA SMPIG
3310  3007          DCA 7
3311  2007          ISZ 7             /HANG UP ERROR TEST
3312  7410          SKP
3313  5325          JMP ERROR1
3314  6003          SRQ               /WAIT FOR FLAG
3315  5311          JMP .-4
3316  6606          GOTDAT            /CLEAR USELESS DATA
3317  6001          ION               /IT'S OK NOW.
3320  5523          RETURN
                /
                /
3321  7201  ERROR5, CLA IAC           /RUNAWAY PROGRAM - ERROR
3322  7001  ERROR4, IAC               /PROBLEM WITH AUTO CAL
3323  7001  ERROR3, IAC               /BAD SPECIFICATION FOR MINISAMPLE TIME
3324  7001  ERROR2, IAC               /ERROR 2 - HUNG UP PRINTER - ERROR
3325  7001  ERROR1, IAC               /ERROR 1 - HUNG UP A/D - ERROR
3326  1163          TAD KD320         /PUT MINUS SIGN IN DISPLAY
3327  6604          LODIS
3330  1370          TAD (335
3331  6605          HIDIS
3332  4115          CALL SETMD        /TURN ON DISPLAY
3333  5506
3334  3777          -VDIS-1
3335  4000          VDIS
3336  5523          RETURN
                /
3337  1076  SW3,     TAD TSW3F         /IS THIS FIRST TIME THROUGH?
3340  7640          SZA CLA
3341  5767          JMP TSW3X
3342  7240          STA               /YES.  SET FLAG
3343  3076          DCA TSW3F
3344  3101          DCA TSW8F         /CLEAR THRESHOLD FLAG
3345  1366          TAD (S3TBMJ
3346  3006  SW3CM,  DCA 6             /SAVE TABLE ADDRESS
3347  1151          TAD DMPX          /KILL DUMP IF IN PROGRESS
3350  3150          DCA IDMP
3351  1157          TAD KD17          /GET MODE
3352  0125          AND MODE
3353  1006          TAD 6             /MEMORY FLAG TABLE
3354  3007          DCA 7
3355  1407          TAD I 7
3356  3014          DCA XRT2          /SET AUTO POINTER FOR EASY TABLE LOOKUP
3357  6002          IOF               /DO THINGS IN THE DARK
3360  1414          TAD I XRT2        /GET THE FLAG BITS
```

```
3361  3124            DCA MEMFLG
                      /
3362  5765            JMP SW3CMP
3365  3400            PAGE
3366  3407
3367  0215
3370  0335
3371  7746
3372  7767
3373  0022
3374  0530
3375  3376
3376  4523
3377  3137
                SW3CMP,
                      /
3400  1414            TAD I XRT2
3401  0125            AND MODE          /MASK OUT SOME BITS
3402  1414            TAD I XRT2        /AND PUT SOME IN
3403  3125            DCA MODE
3404  1125            TAD MODE          /UPDATE HARD WARE TOO.
3405  6607            SELMD
3406  5777            JMP MODJMP        /UNIVERSAL JUMP USES TABLE FOR POINTER
                      /
                      /
                /TABLE LISTS
                      /
3407  3432  S3TBMJ,  ST3M0-1
3410  3432            ST3M1-1
3411  3441            ST3M2-1
3412  3447            ST3M3-1
3413  3453            ST3M4-1
3414  3457            ST3M5-1
3415  3457            ST3M6-1
3416  3457            ST3M7-1
3417  3467            ST3M8-1
3420  3467            ST3M9-1
3421  3426            ST3MER-1
3422  3426            ST3MER-1
3423  3426            ST3MER-1
3424  3426            ST3MER-1
3425  3426            ST3MER-1
3426  3467            ST3M15-1
                      /
                      /
3427  0000  ST3MER,  0                 /GO TO STANDBY FOR ERROR
3430  0117            PKCHN!MODNUM
3431  4400            VDIS!STNBLD
3432  3500            SW3AA
                      /
                ST3M0,
3433  0500  ST3M1,   500
3434  5177            -STRTLD-STNBLD-MINITA-1
3435  0200            STRTLD
3436  3505            SW3B
3437  3510            SW3C
3440  3502            SW3A
3441  0215            TSW3X
                      /
3442  0500  ST3M2,   500
```

```
3443  5177          -STRTLD-STNBLD-MINITA-1
3444  2200          STRTLD!MINITA
3445  3505          SW3B
3446  3502          SW3A
3447  5071          SW8T              /THIS EXIT PUTS OUT DIGITAL DATA ON MINITAPE
                /
3450  4110  ST3M3,  4110              /THIS IS NOP
3451  7777          -1
3452  0000          0
3453  3670          SW4B
                /
3454  4100  ST3M4,  4100
3455  1177          -VDIS-STNBLD-STRTLD-MINITA-1
3456  2200          STRTLD!MINITA
3457  3522          SW3D
                /
              ST3M5,
              ST3M6,
3460  4710  ST3M7,  4710
3461  3177          -VDIS-STNBLD-STRTLD-1
3462  0200          STRTLD
3463  3505          SW3B
3464  5544          WFUN1H
3465  3502          SW3A
3466  3524          SW3E
3467  0215          TSW3X
                /
              ST3M8,
              ST3M9,
3470  4700  ST3M15, 4700
3471  3177          -VDIS-STNBLD-STRTLD-1
3472  0200          STRTLD
3473  3505          SW3B
3474  5544          WFUN1H
3475  3502          SW3A
3476  3524          SW3E
3477  3544          SW3F1
                /
                /
                /
                /
3500  6001  SW3AA,  ION               /SIMPLE ESCAPE
3501  5776          JMP ERR0          /TO NON-EXISTANT FUNCTION
                /
3502  4115  SW3A,   CALL XSW3A        /SET HARDWARE
3503  3273
3504  5777          JMP MODJMP
                /
3505  4115  SW3B,   CALL TTSET        /SET THRESHOLD AND TIME
3506  5474
3507  5777          JMP MODJMP
                /
3510  7240  SW3C,   STA               /SET THRESHOLD TO ZERO FOR CAL
3511  6602          THSET
3512  1125          TAD MODE          /MODE 0?
3513  0157          AND KD17
3514  7640          SZA CLA
3515  5320          JMP .+3
3516  3134          DCA KCAL1         /KILL GAIN CONSTANT CH1
3517  5777          JMP MODJMP
```

```
3520  3135          DCA KCAL2       /KILL GAIN CONSTANT CH2
3521  5777          JMP MODJMP
                    /
3522  6001  SW3D,   ION             /DON'T SET HARDWARE AGAIN
3523  5775          JMP SW8T        /GO WRITE TAPE
                    /
3524  1125  SW3E,   TAD MODE        /SKIP FOR MODES 5 AND 15
3525  0157          AND KD17
3526  1374          TAD (-5
3527  7450          SNA
3530  5777          JMP MODJMP
3531  1373          TAD (-12
3532  7650          SNA CLA
3533  5777          JMP MODJMP
3534  1372          TAD (PROC3-1    /ENABLE LDN CALCULATION
3535  3013          DCA XRT1        /SET POINTER
3536  3413          DCA I XRT1      /CLEAR SECONDS
3537  3413          DCA I XRT1      /MINUTES
3540  3413          DCA I XRT1      /HOURS
3541  1170          TAD KD2000
3542  3413          DCA I XRT1      /ANY DAY IS OKAY
3543  5777          JMP MODJMP
                    /
                    /
3544  1371  SW3F1,  TAD (PROC4-1    /SET UP REAL TIME CLOCK FOR TAPE RECORDER
3545  3015          DCA XRT3
3546  1130          TAD RECON
3547  4115          CALL SW3FT
3550  3556
3551  3020          DCA 20
3552  1131          TAD RECOFF
3553  4115          CALL SW3FT+1
3554  3557
3555  5775          JMP SW8T        /GO START TAPE RECORDER
                    /
                    /
3556  3020  SW3FT,  DCA 20          /CLEAR SECONDS
3557  3021          DCA 21
3560  3022          DCA 22          /CLEAR HOURS
3561  3023          DCA 23          /CLEAR DAYS
3562  1370          TAD SW3FR       /SET UP RETURN FROM ADDCLK
3563  3024          DCA 24
3564  1160          TAD KD20        /SET UP POINTER FOR ADDCLK
3565  3004          DCA 4
3566  4115          CALL ADDTM+1    /ADD IT ALL UP
3567  5016
3570  5523  SW3FR,  RETURN
3571  0503          PAGE
3572  0477
3573  7766
3574  7773
3575  5071
3576  6622
3577  5412
                    /
                    /
```

```
/
                /
3600  1077  SW4,    TAD TSW4F       /IS THIS THE FIRST TIME THRU?
3601  7640          SZA CLA
3602  5777          JMP TSW4X
3603  7240          STA
3604  3077          DCA TSW4F       /YES.  SET FLAG
                /
3605  1124          TAD MEMFLG      /SHOULD WE IGNORE?
3606  7650          SNA CLA
3607  5777          JMP TSW4X       /YES.
                /
3610  1376          TAD (S4TMJ      /GET TABLE ADDRESS
3611  5775          JMP SW3CM       /AND USE COMMON CODING
                /
3612  3631  S4TMJ,  ST4M0-1
3613  3635          ST4M1-1
3614  3641          ST4M2-1
3615  3645          ST4M3-1
3616  3651          ST4M4-1
3617  3655          ST4M5-1
3620  3655          ST4M6-1
3621  3655          ST4M7-1
3622  3661          ST4M8-1
3623  3661          ST4M9-1
3624  3426          ST4ERR-1
3625  3426          ST4ERR-1
3626  3426          ST4ERR-1
3627  3426          ST4ERR-1
3630  3426          ST4ERR-1
3631  3661          ST4M15-1
                /
                ST4ERR=ST3MER
                /
                /
3632  3502  ST4M0,  3502
3633  5177          -STNBLD-STRTLD-MINITA-1
3634  0000          0
3635  3666          SW4A
                /
                /
3636  3501  ST4M1,  3501
3637  5177          -STNBLD-STRTLD-MINITA-1
3640  0000          0
3641  3666          SW4A
                /
3642  5500  ST4M2,  5500
3643  5177          -STNBLD-STRTLD-MINITA-1
3644  0000          0
3645  3666          SW4A
                /
                /
                /
3646  4110  ST4M3,  4110
3647  7777          -1
3650  0000          0
3651  3670          SW4B
                /
3652  4100  ST4M4,  4100
3653  5177          -STNBLD-STRTLD-MINITA-1
```

```
3654  0000          0
3655  3670          SW4B
                /
                ST4M5,
                ST4M6,
3656  4710  ST4M7,  4710
3657  7777          -1
3660  0000          0
3661  3672          SW4C
                /
                ST4M8,
                ST4M9,
3662  4700  ST4M15, 4700
3663  7777          -1
3664  0000          0
3665  3672          SW4C
                /
3666  4115  SW4A,   CALL DISTST
3667  3676
3670  6001  SW4B,   ION
3671  5777          JMP TSW4X
                /
3672  7240  SW4C,   STA
3673  3063          DCA SMPIG
3674  6614          SAMPLE
3675  5270          JMP SW4B
                /
3676  1374  DISTST, TAD (252       /PUT TEST PATTERN IN DISPLAY
3677  6604          LODIS
3700  1373          TAD (7652
3701  6605          HIDIS
3702  3133          DCA DISFL      /MAKE DATA NOT READY
3703  5523          RETURN
                /
```

```
/
3704  4115  SW5,    CALL PRSET      /SET UP POINTERS TO DATA
3705  3255
3706  1772          TAD TSW1        /SEE WHAT WE ARE DOING NOW.
3707  7006          RTL
3710  7630          SZL CLA
3711  5771          JMP SW5K        /GO KILL
                    /
3712  7330          K4000           /TURN ON PRINTER OR CASSETTE
3713  6612          OUTHI
3714  3007          DCA 7           /AND WAIT 100 MS
3715  2007          ISZ 7
3716  5315          JMP .-1
                    /
3717  1370          TAD (K2         /WAS JMP.  MAKE K2 TO ENABLE
3720  3772          DCA TSW1
3721  1367          TAD (3000       /GET PRINTER NUMBER
3722  0106          AND TFUN0
3723  7106          CLL RTL
3724  7006          RTL
3725  7450          SNA
3726  5766          JMP SW5K+2      /DON'T LET SPARE IN
3727  1365          TAD (-2         /SET FLAG FOR EASY DECODING
3730  3102          DCA PRNTER
                    /
                    /
                    /OPEN CASSETTE IF NEEDED
                    /
3731  1102          TAD PRNTER
3732  7750          SPA SNA CLA
3733  5340          JMP SW5H
3734  1364          TAD (OUTBT1     /SET UP SHORT CALL TO CASSETTE
3735  3431          DCA I PRPAR
3736  4115          CALL LEADER     /AND OUTPUT LEADER
3737  4545
3740  4115  SW5H,   CALL LF         /OUTPUT HEADER.  START WITH LF
3741  4621
3742  1143          TAD CLMIN       /SAVE FULL CLOCK TO PREVENT CARRIES
3743  7002          BSW
3744  1144          TAD CLHR
3745  3103          DCA DATPNT
3746  1145          TAD CLDAY       /START HEADER WITH DAY
3747  4115          CALL BPRINT
3750  4343
3751  1103          TAD DATPNT
3752  0161          AND KD77        /OUTPUT HOUR
3753  4115          CALL BPRINT
3754  4343
3755  1103          TAD DATPNT
3756  7002          BSW
3757  0161          AND KD77        /OUTPUT MIN
3760  4115          CALL BPRINT
3761  4343
                    /
3762  5763          JMP SW5HP
3763  4000          PAGE
3764  4532
3765  7776
3766  4337
3767  3000
```

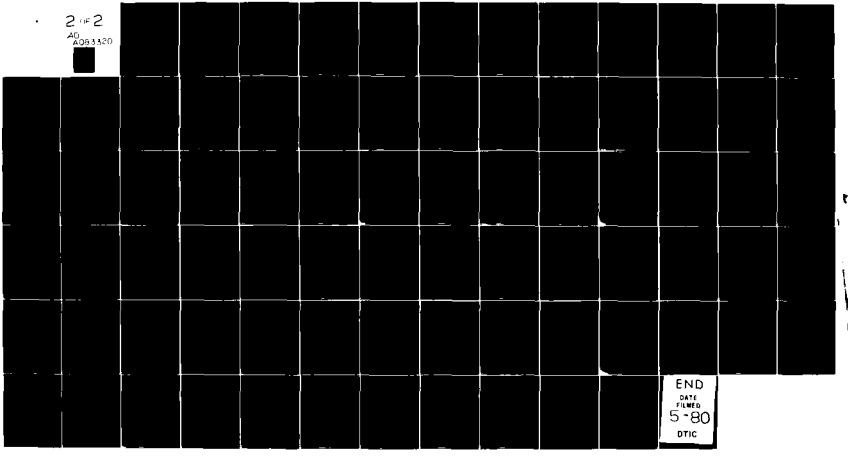
```
3770  7305
3771  4335
3772  0204
3773  7652
3774  0252
3775  3346
3776  3612
3777  0222
                SWSHP,
                      /
                      /
                      /
4000  6623              FUNHI           /GET SERIAL NUMBER
4001  7002              BSW
4002  0161              AND KD77
4003  1377              TAD (GRPNO      /GROUP NUMBER
4004  4115              CALL BPRINT
4005  4343
                      /
4006  1126              TAD SVMD        /GET MODE NUMBER
4007  0157              AND KD17
4010  4115              CALL BPRINT
4011  4343
                      /
4012  1141              TAD TMIN        /GET ACCUMULATION TIME
4013  4115              CALL BCDBIN
4014  6625
4015  4115              CALL BPRINT
4016  4343
4017  2432              ISZ I DATOUT    /CHANGE DECIMAL POINT FOR SECONDS
4020  1140              TAD TSEC
4021  4115              CALL BCDBIN     /CONVERT TO BINARY SO BPRINT WILL WORK
4022  6625
4023  4115              CALL BPRINT
4024  4343
                      /
4025  3432              DCA I DATOUT    /RESET DECIMAL POINT
4026  1174              TAD KD100       /CHANNEL NUMBER OF ANALOG INPUT
4027  0106              AND TFUN0
4030  7640              SZA CLA
4031  7001              IAC             /DUAL
4032  7001              IAC             /SINGLE
4033  4115              CALL BPRINT
4034  4343
                      /
4035  7001              IAC             /SET UP DEC. PT.
4036  3432              DCA I DATOUT
4037  1136              TAD REFLV1      /GET CALIBRATOR LEVELS
4040  4115              CALL APRINT
4041  4346
4042  1137              TAD REFLV2
4043  4115              CALL APRINT
4044  4346
                      /
4045  1134              TAD KCAL1       /GET GAIN CONSTANTS
4046  4115              CALL PRINT
4047  4351
4050  1135              TAD KCAL2
4051  4115              CALL PRINT
4052  4351
```

```
4053  3432          DCA I DATOUT     /RESET DEC. PT.
4054  1174          TAD KD100        /GET PEAK DETECTOR CHANNEL
4055  0126          AND SVMD
4056  7640          SZA CLA
4057  7001          IAC              /CH2
4060  7001          IAC              /CH1
4061  4115          CALL BPRINT
4062  4343
                /
4063  1147          TAD THFLG        /GET CH1 THRESHOLD
4064  4115          CALL BCDBIN
4065  6625
4066  4115          CALL BPRINT
4067  4343
                /
4070  1130          TAD RECON        /GET MINISAMPLE ON TIME
4071  4115          CALL BPRINT
4072  4343
                /
4073  1131          TAD RECOFF       /GET MINISAMPLE PERIOD
4074  4115          CALL BPRINT
4075  4343
                /
4076  1127          TAD FORMAT       /GET MEMORY FORMAT
4077  3423          DCA I TPSV       /IN CASE OF CASSETTE
4100  1376          TAD (SW5C        /FAKE A SUBROUTINE CALL
4101  3433          DCA I PRRET
4102  1423          TAD I TPSV
4103  4115          CALL OCTBCD      /CONVERT BINARY TO OCTAL BCD
4104  6363
4105  5775          JMP PRINT2       /AND SEND IT
                /
4106  4115  SW5C,   CALL LF          /OK,  NOW PUT OUT DATA BLOCKS
4107  4621
4110  1041  SW5C1,  TAD EMPTY       /ALL DATA DONE?
4111  7700          SMA CLA
4112  5333          JMP SW5D
                /
4113  1102          TAD PRNTER       /ARE WE USING THE WANG?
4114  7640          SZA CLA
4115  5321          JMP SW5C2
4116  1374          TAD (SW5C1       /YES.  SET UP RETURN
4117  3433          DCA I PRRET
4120  5773          JMP TSW1X        /THIS ALLOWS US TO GO FOREVER
                /
4121  4115  SW5C2,  CALL BPRINT      /DONE.  BLOCK NUMBER IS ZERO
4122  4343
                /
4123  4115  SW5L,   CALL LF          /OK,  NOW PUT OUT LDN BLOCKS
4124  4621
4125  1047          TAD LDNMT        /ALL LDN BLOCKS DONE?
4126  7700          SMA CLA
4127  5772          JMP SW5E
                /
4130  4115          CALL BPRINT      /DONE.  BLOCK NUMBER IS ZERO
4131  4343
                /
4132  5771          JMP SW5K         /GO TURN OFF PRINTER
                /
4133  1034  SW5D,   TAD STORE
```

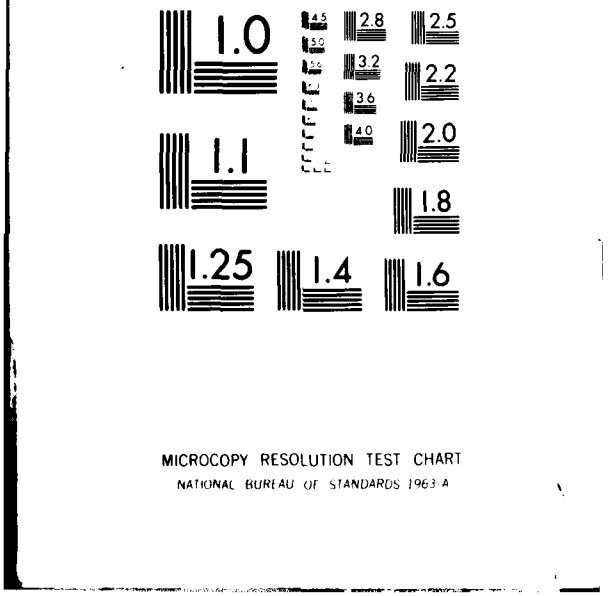
```
4134  7041          CIA
4135  1040          TAD OUTFLO      /OUTPUT BLOCK NUMBER
4136  7001          IAC             /STARTING AT ONE
4137  4115          CALL BPRINT
4140  4343
                /
4141  1037          TAD XR16        /OUTPUT A BLOCK
4142  3103          DCA DATPNT
4143  1370          TAD (DMASK
4144  4115          CALL PRNIT      /PRINT MEMORY AS SPECIFIED IN FORMAT
4145  4206
4146  5351          JMP .+3
4147  4064          PRHTB-DMASK-1
4150  0020          20
                /
4151  6000          SKON            /SO WE DON'T CONFUSE INTERRUPT
4152  7040          CMA
4153  3002          DCA 2           /SAVE FLAG
4154  1103          TAD DATPNT
4155  3037          DCA XR16
4156  2040          ISZ OUTFLO
4157  5363          JMP SW5D2
4160  1034          TAD STORE       /HIT THE END.  RESET POINTERS
4161  3040          DCA OUTFLO
4162  3037          DCA XR16        /FIX UP BLOCK POINTER
                /
4163  1036  SW5D2,  TAD INFLO
4164  7041          CIA
4165  1040          TAD OUTFLO      /BUFFER EMPTY?
                /
4166  5767          JMP PRNIT&7700
4167  4200          PAGE
4170  0167
4171  4335
4172  4273
4173  0210
4174  4110
4175  4362
4176  4106
4177  0144
                /
4200  7650          SNA CLA
4201  7040          CMA
4202  3041          DCA EMPTY       /SET FLAG
4203  2002          ISZ 2           /WHAT SHALL WE DO WITH INTERRUPT
4204  6001          ION             /IT'S SAFE NOW
4205  5777          JMP SW5C
                /
4206  3104  PRNIT,  DCA PRFH
4207  1520          TAD I STACK     /GET RETURN ADDRESS OFF STACK
4210  3776          DCA PRSVR       /SAVE OUT OF THE WAY
4211  2120          ISZ STACK
4212  7001  PRNIT1, IAC
4213  3432          DCA I DATOUT    /SET DECIMAL POINT
4214  1504          TAD I PRFH      /GET MASK
4215  7450          SNA
4216  5251          JMP PRNIT2      /END OF TABLE
4217  0127          AND FORMAT      /COMPARE WITH FORMAT
4220  7041          CIA
4221  1504          TAD I PRFH      /MAY BE MORE THAN ONE BIT SET
```

AD-A083 320 CONSTRUCTION ENGINEERING RESEARCH LAB (ARMY) CHAMPAIGN IL F/G 20/1
TRUE-INTEGRATING ENVIRONMENTAL NOISE MONITOR AND SOUND-EXPOSURE--ETC(U)
MAR 80 A J AVERBUCH, L M LITTLE
UNCLASSIFIED CERL-TR-N-41-VOL-3 NL

2 OF 2
AD
A083320



END
DATE FILMED
5-80
DTIC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

```
4222  2104            ISZ PRFH          /DON'T USE THIS BIT AGAIN
4223  7640            SZA CLA
4224  5212            JMP PRNIT1        /THIS ONE NOT THERE
                      /
4225  1776            TAD PRSVR         /GET POINTER FROM HIDING PLACE
4226  3013            DCA XRT1          /NOW GET TABLE OFFSET
4227  1413            TAD I XRT1
4230  1104            TAD PRFH
4231  3007            DCA 7             /POINTER TO POINTER FOR SUB INDIR
4232  1375            TAD (-PRHTB-6     /CHANGE THE DECIMAL POINT ONLY FOR ONE CASE
4233  1104            TAD PRFH
4234  7650            SNA CLA
4235  2432            ISZ I DATOUT      /HERE IT IS!
                      /
4236  1103            TAD DATPNT        /GET DATA ADDRESS
4237  3006            DCA 6             /SET UP FOR FIELD ONE CALL
4240  1413            TAD I XRT1        /GET DATA FIELD
4241  4115            CALL GETDAT       /AND GET DATA
4242  7124
4243  2103            ISZ DATPNT
4244  7000            NOP
4245  4115            CALL INDIR        /AND PRINT IT
4246  6671
4247  0007            7
4250  5212            JMP PRNIT1        /RETURNS HERE AFTER PRINT.  KEEP LOOKING
                      /
4251  1776  PRNIT2,   TAD PRSVR         /GET RETURN ADDRESS AND EXIT
4252  3007            DCA 7
4253  5407            JMP I 7
                      /
                      /
4254  4346  PRHTB,    APRINT
4255  4346            APRINT
4256  4346            APRINT
4257  4351            PRINT
4260  4351            PRINT
4261  4351            PRINT
4262  4343            BPRINT
                      /
4263  4346  PRLTB,    APRINT
4264  4346            APRINT
4265  4351            PRINT
4266  4351            PRINT
4267  4346            APRINT
4270  4346            APRINT
4271  4346            APRINT
4272  4346            APRINT
                      /
                      /
                      /
4273  1042  SWSE,     TAD LDNST         /GET BLOCK NUMBER
4274  7041            CIA
4275  1046            TAD LDNOT
4276  7001            IAC               /STARTING AT ONE
4277  4115            CALL BPRINT
4300  4343
                      /
4301  1045            TAD LDN16         /OUTPUT A BLOCK
4302  3103            DCA DATPNT
4303  1374            TAD (LMASK
```

```
4304  4115          CALL PRNIT      /GET THE SPECIFIED DATA AND PRINT
4305  4206
4306  5311          JMP .+3
4307  5103          PRLTB-LMASK-1
4310  0000          0
                    /
                    /
4311  6000  SW5E1,   SKON            /SO WE DON'T CONFUSE INTERRUPT
4312  7040          CMA
4313  3002          DCA 2             /SAVE FLAG
4314  1103          TAD DATPNT
4315  3045          DCA LDN16
4316  2046          ISZ LDNOT
4317  5324          JMP SW5E4
4320  1042          TAD LDNST         /HIT THE END.  RESET POINTER
4321  3046          DCA LDNOT
4322  1373          TAD (LDNBOT
4323  3045          DCA LDN16         /FIX UP BLOCK POINTER
4324  1044  SW5E4,   TAD LDNIN
4325  7041          CIA
4326  1046          TAD LDNOT         /BUFFER EMPTY?
4327  7650          SNA CLA
4330  7040          CMA
4331  3047          DCA LDNMT         /SET FLAG
4332  2002          ISZ 2             /WHAT SHALL WE DO WITH INTERRUPT?
4333  6001          ION               /IT'S SAFE NOW
4334  5772          JMP SW5L
                    /
                    /
4335  4115  SW5K,   CALL LF
4336  4621
4337  6612          OUTHI             /TURN OFF PRINTER
4340  1371          TAD (5000+TSW1X /DELETE PRINTER FROM CHAIN
4341  3770          DCA TSW1
4342  5767          JMP TSW5+4
                    /
                    /
```

```
                /
                /       DCA I DATOUT    /SET DECIMAL POINT.  BUT NOT EVERY TIME
                /       TAD VALUE       /THIS IS HOW TO CALL
                /       CALL PRINT      /CONVERT FROM LOG TO INTEGER TO BCD
                /                       /OR APRINT FOR LARGE DB'S
                /
                /
                /       OR.........
                /
                /
                /       DCA I DATOUT    /SET DECIMAL POINT.   BUT NOT EVERY TIME
                /       TAD VALUE
                /       CALL BPRINT     /BCD CONVERSION ONLY
                /
                /
                /
4343  3423      BPRINT, DCA I TPSV      /SAVE VALUE
4344  3002              DCA 2           /CLEAR SIGN.  ONLY PLUS NUMBERS
4345  5354              JMP PRINT1
                /
4346  4115      APRINT, CALL PDBCON     /CONVERT FROM LOG TO INTEGER:  ALLOW LARGE PLUS NUMBERS
4347  6761
4350  5353              JMP .+3
                /
4351  4115      PRINT,  CALL DBCON      /CONVERT FROM LOG TO INTEGER:  ALLOW + OR - SMALL NUMBERS
4352  6763
                /
4353  3423              DCA I TPSV      /SAVE CONVERTED VALUE
4354  1520      PRINT1, TAD I STACK     /POP RETURN ADDRESS
4355  3433              DCA I PRRET
4356  2120              ISZ STACK
4357  1423              TAD I TPSV
4360  4115              CALL BCDCON     /CONVERT VALUE TO BCD
4361  6702
                /
4362  6610      PRINT2, OUTLO           /AC & 7 HAVE LO.  3 HAS HI
4363  1102              TAD PRNTER      /SELECT DEVICE
                /
4364  5766              JMP PRN2A&7700
4366  4400              PAGE
4367  0226
4370  0204
4371  5210
4372  4123
4373  0572
4374  7157
4375  3516
4376  0543
4377  4106
                /
4400  7440              SZA
4401  5206              JMP PRNTST
                /
4402  1003              TAD 3           /HIGH DIGIT
4403  6611      PRN2A,  OUTMI
4404  1002              TAD 2           /SIGN
4405  5216              JMP PRN1A+1     /AND PRINT
                /
                /
```

```
4406  7700  PRNTST, SMA CLA
4407  5233          JMP PRN3
4410  1003          TAD 3               /MUST BE THERMAL PRINTER
4411  6611          OUTMI
4412  1002          TAD 2               /GET SIGN
4413  7002          BSW                 /IN PROPER POSITION
4414  1432          TAD I DATOUT        /NOW PUT IN DECIMAL POINT TOO
                    /
                    /
4415  1167  PRN1A,  TAD KD4000          /KEEP MOTOR RUNNING
4416  6612          OUTHI               /GET HIGH BITS SET
4417  6615          OPRINT              /SEND THEM ALL
4420  1216          TAD PRN1A+1         /ABOUT 100 MS
4421  3007          DCA 7
4422  6622  PRN1B,  FUNLO              /WAIT FOR DATA ACCEPTED
4423  0377          AND (2
4424  7650          SNA CLA
4425  5776          JMP TSW1X           /DO OTHER THINGS WHILE WAITING FOR PRINTER
4426  2007          ISZ 7
4427  5222          JMP PRN1B
4430  4115          CALL ERROR2         /ERROR.  TELL PEOPLE ABOUT IT
4431  3324
4432  5775          JMP SW5K+2          /AND KILL PRINT REQUEST
                    /
4433  1155  PRN3,   TAD KM12           /SET UP FOR 12 BIT WORD
4434  3425          DCA I PRCNT
4435  1033          TAD PRRET
4436  3013          DCA XRT1            /SET UP SECOND LEVEL RETURN
4437  1433          TAD I PRRET
4440  3413          DCA I XRT1
                    /
4441  7330  PRN3A,  K4000              /GET HIGH ORDER BIT OF WORD
4442  0423          AND I TPSV          /AND UPDATE CRC AS NEEDED
4443  1426          TAD I PRREM         /EASY EXCLUSIVE OR
4444  7700          SMA CLA
4445  5260          JMP OUT3B           /SKIP PROCEDURE FOR ZERO RESULT
                    /
4446  1374          TAD (5234           /X IS GENERATOR POLYNOMIAL
4447  7040          CMA                 /XBAR
4450  0426          AND I PRREM         /Y AND XBAR.  Y IS REMAINDER
4451  7421          MQL                 /SAVE
4452  1426          TAD I PRREM         /Y
4453  7040          CMA                 /YBAR
4454  0374          AND (5234           /X AND YBAR
4455  7501          MQA                 /INCLUSIVE OR  (Y&XB + X&YB)
4456  7120          STL                 /SET LOW ORDER BIT TO ONE
4457  5262          JMP .+3
                    /
4460  7100  OUT3B,  CLL                 /SET LOW ORDER BIT TO ZERO
4461  1426          TAD I PRREM         /USE OLD REMAINDER
4462  7004          RAL                 /SHIFT LEFT
4463  3426          DCA I PRREM         /AND SAVE NEW REMAINDER
4464  1423          TAD I TPSV          /SHIFT DATA WORD
4465  7104          CLL RAL
4466  3423          DCA I TPSV
                    /
4467  4115          CALL INDIR
4470  6671
4471  0022          BITOUT              /AND OUTPUT IT
                    /
```

```
4472  2424            ISZ I PRSTCT    /DONE WITH 51 BITS?
4473  5320            JMP PRN3D       /NO
4474  1155            TAD KM12        /YES.  OUTPUT REMAINDER
4475  3424            DCA I PRSTCT
4476  1426  PRN3C,    TAD I PRREM     /ONE BIT AT A TIME
4477  7104            CLL RAL
4500  3426            DCA I PRREM
4501  4115            CALL INDIR
4502  6671
4503  0022            BITOUT
4504  2424            ISZ I PRSTCT
4505  5276            JMP PRN3C
                /
4506  1431            TAD I PRPAR     /OK.  NOW OUTPUT PARITY
4507  7010            RAR
4510  7220            CLA CML         /SEND ODD PARITY
4511  4115            CALL INDIR
4512  6671
4513  0022            BITOUT
4514  1373            TAD (-63        /RESET DATA BIT COUNTER
4515  3424            DCA I PRSTCT
4516  3426            DCA I PRREM     /RESET REMAINDER
4517  3431            DCA I PRPAR     /AND PARITY
4520  2425  PRN3D,    ISZ I PRCNT     /12 BITS?
4521  5241            JMP PRN3A
4522  5772            JMP RET2        /EXIT FROM SECOND LEVEL
              /
              /
4523  7430  OUTBIT,   SZL             /OUTPUT DATA HELD IN LINK
4524  2431            ISZ I PRPAR     /UPDATE PARITY
4525  1427            TAD I PRBYT     /BUILD 8 BIT BYTE
4526  7004            RAL
4527  3427            DCA I PRBYT
4530  2430            ISZ I PRBYTC    /DONE 8 BITS?
4531  5523            RETURN          /NOT YET!
              /
4532  1520  OUTBT1,   TAD I STACK     /SAVE CALL ON LEVEL ONE
4533  3433            DCA I PRRET
4534  2120            ISZ STACK
4535  1371            TAD (-10        /RESET BYTE COUNTER
4536  3430            DCA I PRBYTC
4537  1427            TAD I PRBYT
4540  7104            CLL RAL         /KEEP MOTOR RUNNING BY SETTING HIGH BIT
4541  7130            STL RAR
4542  6612            OUTHI           /SEND HI ONLY
4543  6615            OPRINT          /LOAD TRANSMITTER
4544  5776            JMP TSW1X       /DO OTHER THINGS WHILE WAITING
              /
              /
```

```
                /
                /
4545  1033  LEADER, TAD PRRET
4546  3013         DCA XRT1
4547  1520         TAD I STACK      /PUT RETURN ON SECOND LEVEL
4550  3413         DCA I XRT1
4551  2120         ISZ STACK
4552  1166         TAD KD7760       /OUTPUT 16 LEADER BYTES
4553  3426         DCA I PRREM      /THIS CAN HOLD IT FOR A WHILE
4554  7350         CLL STA RAR      /SEND RUBS
4555  3427         DCA I PRBYT
4556  4115         CALL INDIR       /SEND 8 BITS AT A TIME
4557  6671
4560  0031         PRPAR            /GOOD PLACE TO KEEP ADDRESS
4561  2426         ISZ I PRREM
4562  5356         JMP .-4
                /
4563  1370         TAD (13          /NOW PUT OUT SYNCH BYTES
4564  3427         DCA I PRBYT
                /
4565  5767         JMP RET2&7700
4567  4600         PAGE
4570  0013
4571  7770
4572  4614
4573  7715
4574  5234
4575  4337
4576  0210
4577  0002
                /
                /
4600  4115         CALL INDIR
4601  6671
4602  0031         PRPAR
4603  1163         TAD KD320        /THIS IS BIT INVERTED FROM FIRST ONE
4604  3427         DCA I PRBYT
4605  4115         CALL INDIR
4606  6671
4607  0031         PRPAR
4610  3426         DCA I PRREM      /RESET REMAINDER
4611  3431         DCA I PRPAR      /AND PARITY
4612  1377         TAD (-63         /SET UP DATA BIT COUNT
4613  3424         DCA I PRSTCT
                /
4614  1033  RET2,  TAD PRRET       /RETURN FROM SECOND LEVEL
4615  3013         DCA XRT1
4616  1413         TAD I XRT1
4617  3007         DCA 7
4620  5407         JMP I 7
                /
                /
```

```
/
                /
4621  3432  LF,       DCA I DATOUT      /CLEAR DECIMAL POINT
4622  1520            TAD I STACK       /POP RETURN OFF STACK
4623  3433            DCA I PRRET       /SAVE ON LEVEL ONE
4624  2120            ISZ STACK
4625  1102            TAD PRNTER
4626  7440            SZA
4627  5234            JMP LF1
                /
4630  6610            OUTLO             /PROGRAMMABLE CALCULATOR.  PUT OUT SYNCH MARK
4631  6611            OUTMI
4632  7305            K2
4633  5776            JMP PRN1A+1       /SEND HIGH
                /
4634  7700  LF1,      SMA CLA
4635  5243            JMP LF2
4636  7240            STA               /FOR THERMAL PRINTER, SEND ALL ONES
4637  6610            OUTLO
4640  7240            STA
4641  6611            OUTMI
4642  5775            JMP PRN1A         /BUT NO DECIMAL POINTS OR SIGN
                /
4643  1374  LF2,      TAD (-5           /PUT OUT BLANKS TO CLEAR OUT ECC
4644  3103            DCA DATPNT
4645  1433            TAD I PRRET       /SAVE RETURN ON LEVEL 3
4646  3432            DCA I DATOUT
4647  4115  LF2A,     CALL BPRINT       /PRINT A BLANK
4650  4343
                /
4651  2103  LF2B,     ISZ DATPNT
4652  5247            JMP LF2A
4653  1432            TAD I DATOUT      /DONE.  GET RETURN
4654  3007            DCA 7
4655  5407            JMP I 7           /AND EXIT
                /
```

```
/
                /       THIS IS CLOCK SERVICE ROUTINE
                /
                /
4656  6616  SW7,    CLCF            /CLEAR FLAG AND ADVANCE CLOCK WITH INTERRUPT OFF
4657  6000          SKON            /TEST FOR MACHINE INTERRUPT ON
4660  7240          STA
4661  3003          DCA 3           /SAVE INFO HERE
                /
4662  4115          CALL ADDCLK
4663  5010
4664  0141  SW7A,    CLSEC-1
4665  0001          1
4666  0000          0
4667  0000          0
4670  0000          0
                /
4671  2003          ISZ 3           /RESTORE INTERRUPT IF NECESSARY
4672  6001          ION
4673  1373          TAD (PRBUF      /SET UP POINTER TO PROCESS BUFFER
4674  3100          DCA PRPNT
4675  1100  SW7B,    TAD PRPNT       /SET UP SECOND POINTER TO PROCESS BUFFER
4676  3021          DCA 21
4677  1264          TAD SW7A        /SET UP POINTER TO CLOCK
4700  3013          DCA XRT1
4701  1372          TAD (-4
4702  3007          DCA 7           /4 WORDS PER BLOCK
                /
4703  1421  SW7C,    TAD I 21        /GET PROCESS WORD
4704  7510          SPA
4705  5331          JMP SW7D        /NO PROCESS HERE
4706  7106          CLL RTL         /CHECK FOR ALL PASS FLAG
4707  7630          SZL CLA
4710  5321          JMP SWC1
4711  1421          TAD I 21
4712  7041          CIA
4713  1413          TAD I XRT1      /A MATCH?
4714  7640          SZA CLA
4715  5331          JMP SW7D        /NO.  EXIT
4716  2021          ISZ 21          /ADVANCE POINTER FOR NEXT COMPARE
4717  2007          ISZ 7           /DONE ALL?
4720  5303          JMP SW7C        /NOPE
                /
4721  1371  SWC1,    TAD (-PRBUF     /FOUND A MATCH.  GO PROCESS
4722  1100          TAD PRPNT       /CALCULATE SUBSCRIPT
4723  7112          CLL RTR         /DIVIDE BY 4
4724  1370          TAD (PRTAB
4725  3007          DCA 7
4726  1407          TAD I 7         /GET ADDRESS OF PROCESS
4727  3007          DCA 7
4730  5407          JMP I 7         /GO THERE
                /
4731  7307  SW7D,    K4              /ADVANCE TO NEXT PROCESS
4732  1100          TAD PRPNT
4733  3100          DCA PRPNT
4734  1367          TAD (-PRDAT
4735  1100          TAD PRPNT       /DONE WITH ALL?
4736  7710          SPA CLA
4737  5275          JMP SW7B        /NO.  TRY AGAIN
4740  5766          JMP TSW7+4      /YES.  TRY SOMETHING ELSE
```

```
                /
                /
4741  3344  PRTAB,  SW3+5           /AUTO START
4742  4756           SAUTOC          /AUTO CALIBRATION
4743  4751           LDNPUT          /PUT LDN IN MEMORY
4744  5211           SW8TO           /DONE WITH ON PERIOD.  TURN OFF TAPE RECORDER
4745  3544           SW3F1           /SET UP TAPE RECORDER AGAIN AND TURN IT ON
4746  5002           SUB10           /THRESHOLD ADJUST TIME
4747  5000           AD10            /THRESHOLD ADJUST TIME
4750  1636           NAUTOC          /TAKE SHORT BLOCK AND DO AUTOSTART
                /
                /
4751  1365  LDNPUT, TAD (-40-1       /PUT OUT LDN AT NEXT BLOCK
4752  0124           AND MEMFLG
4753  1175           TAD KD40
4754  3124           DCA MEMFLG
4755  5331           JMP SW7D        /LOOK FOR OTHER MATCHES
                /
                /
4756  7307  SAUTOC, K4               /SET UP AUTOCAL AT THE BEGINNING OF NEXT BLOCK
4757  7421           MQL
4760  1124           TAD MEMFLG
4761  7501           MQA
4762  3124           DCA MEMFLG
4763  5331           JMP SW7D
                /
                /
4765  7737           PAGE
4766  0232
4767  7250
4770  4741
4771  7310
4772  7774
4773  0470
4774  7773
4775  4415
4776  4416
4777  7715
                /
                /
5000  1377  AD10,   TAD (0324       /10 DB IS 3.2446 OCTAL IN INTERNAL REPRESENTATION
5001  7410           SKP
5002  1376  SUB10,  TAD (-0324
5003  1146           TAD THOLD
5004  3146           DCA THOLD
5005  4115           CALL XFUN3
5006  5740
5007  5775           JMP SW7D
                /
```

```
/
                /
5010  1520  ADDCLK, TAD I STACK      /GET RETURN ADDRESS OFF STACK
5011  3004          DCA 4
5012  2120          ISZ STACK
                /
5013  1404          TAD I 4          /GET DESTINATION ADDRESS
5014  2004          ISZ 4
                /
5015  3015  ADDTM,  DCA XRT3         /SAVE DESTINATION ADDRESS
5016  1374          TAD (CLSEC-1     /GET CLOCK POINTER SET UP
5017  3013          DCA XRT1
                /
                    DECIMAL
                /
5020  3006          DCA 6            /CLEAR CARRIES
5021  1373          TAD (-60
5022  4115          CALL ADDT        /SECONDS
5023  5036
5024  1373          TAD (-60
5025  4115          CALL ADDT        /MINUTES
5026  5036
5027  1372          TAD (-24
5030  4115          CALL ADDT        /HOURS
5031  5036
5032  1371          TAD (-365
5033  4115          CALL ADDT        /DAYS
5034  5036
                /
                    OCTAL
                /
5035  5404          JMP I 4          /DONE.  EXIT
                /
                /
5036  3007  ADDT,   DCA 7            /SAVE DIVISOR
5037  1006          TAD 6            /GET CARRIES FROM PREVIOUS RUN
5040  1413          TAD I XRT1       /CLOCK
5041  1404          TAD I 4          /NEW NUMBER
5042  2004          ISZ 4
5043  3005          DCA 5            /SAVE RESULT TEMPORARILY
5044  3006          DCA 6            /CLEAR QUOTIENT
5045  1005          TAD 5
5046  4115          CALL DIVD
5047  6200
5050  7200          CLA
5051  1005          TAD 5            /GET REMAINDER
5052  3415          DCA I XRT3       /RESULT
5053  5523          RETURN
                /
                /
```

```
/
5054  1101  SW8,    TAD TSW8F       /IS THIS FIRST TIME THROUGH?
5055  7640          SZA CLA
5056  5770          JMP TSW8X
5057  7240          STA
5060  3101          DCA TSW8F       /YES.  SET FLAG TO PREVENT REPEATS
5061  4115  SW8A,   CALL SW8C       /TEST FLAGS
5062  5145
5063  5770          JMP TSW8X       /NO ACTION RETURN
5064  5271          JMP SW8T        /GO TURN ON TAPE RECORDER
5065  4115          CALL SETMD      /YES.  FLASH ON START LIGHT
5066  5506
5067  3177          -VDIS-STRTLD-STNBLD-1
5070  0200          STRTLD
                /
5071  1367  SW8T,   TAD (STATIN     /YES.  ENABLE FLAG TESTING
5072  3766          DCA TPOUT
5073  4115          CALL TPSET      /SET UP POINTERS
5074  5247
5075  4115          CALL SETMD
5076  5506
5077  5777          -MINITA-1       /START MOTOR
5100  2000          MINITA
                /
5101  1365          TAD (TRBIT1     /SET UP SHORT PRINT
5102  3431          DCA I PRPAR
5103  4115          CALL LEADER     /OUTPUT LEADER
5104  4545
                /
5105  6623          FUNHI           /OUTPUT SERIAL NUMBER
5106  7002          BSW             /GET UNIT NUMBER
5107  0161          AND KD77
5110  1364          TAD (GRPNO      /AND GROUP NUMBER
5111  4115          CALL TR3
5112  5235
                /
5113  1105          TAD TRBLK       /PUT OUT BLOCK NUMBER
5114  2105          ISZ TRBLK
5115  7000          NOP
5116  4115          CALL TR3
5117  5235
                /
5120  1144          TAD CLHR        /SEND DAY BUT SAVE HOUR AND MIN TO PREVENT CARRIES
5121  3006          DCA 6
5122  1162          TAD KD144       /PACK HR*100+MIN
5123  3005          DCA 5
5124  4115          CALL MPLY
5125  6736
5126  1143          TAD CLMIN
5127  3432          DCA I DATOUT
5130  1145          TAD CLDAY
5131  4115          CALL TR3        /SEND DAY
5132  5235
5133  1432          TAD I DATOUT
5134  4115          CALL TR3        /SEND HOUR AND MINUTES COMBINED
5135  5235
                /
5136  4115          CALL TR3        /SEND BLANKS TO ACT AS FILLER
5137  5235
5140  4115          CALL TR3
```

```
5141  5235
5142  1363  SW8T1,  TAD (5000+TPOUTX /TURN OFF CASSETTE FLAG CHECK
5143  3766          DCA TPOUT
5144  5770          JMP TSW8X       /DO OTHER THINGS WHILE WAITING
                    /
                    /
                    /
5145  1156  SW8C,   TAD KD10        /THRESHOLD CONTROL OF TAPE?
5146  0124          AND MEMFLG
5147  7650          SNA CLA
5150  5523          RETURN          /NO.  NO ACTION
5151  2520          ISZ I STACK
5152  1157          TAD KD17        /IS IT MODE 3?
5153  0125          AND MODE
5154  1362          TAD (-3
5155  7640          SZA CLA
5156  5523          RETURN          /NO.
5157  2520          ISZ I STACK
5160  5523          RETURN          /FIDDLE WITH LIGHTS
                    /
                    /
5162  7775          PAGE
5163  5253
5164  0144
5165  5225
5166  0247
5167  6626
5170  0241
5171  7223
5172  7750
5173  7704
5174  0141
5175  4731
5176  7454
5177  0324
                    /
                    /
5200  3101  SW8L,   DCA TSW8F        /CLEAR FLAG TO PREVENT REPEATS
5201  4115          CALL SW8C       /TEST FLAGS
5202  5145
5203  5777          JMP TSW8X       /NO ACTION RETURN
5204  5211          JMP SW8T0       /JUST TURN OFF TAPE
5205  4115          CALL SETMD      /YES.  KILL START LIGHT
5206  5506
5207  7177          -STRTLD-STNBLD-1
5210  0000          0
                    /
5211  4115  SW8T0,  CALL SETMD      /TURN OFF MINITAPE MOTOR
5212  5506
5213  5777          -MINITA-1
5214  0000          0
5215  5776          JMP SW8T1       /ALL DONE
                    /
                    /
                    /
5216  7430  TRBIT,  SZL             /OUTPUT DATA HELD IN LINK
5217  2431          ISZ I PRPAR     /UPDATE PARITY
5220  1427          TAD I PRBYT     /BUILD 8 BIT BYTE
5221  7004          RAL
5222  3427          DCA I PRBYT
```

```
5223  2430          ISZ I PRBYTC     /DONE?
5224  5523          RETURN
                    /
5225  1520  TRBIT1, TAD I STACK      /SAVE RETURN ON LEVEL ONE
5226  3433          DCA I PRRET
5227  2120          ISZ STACK
5230  1375          TAD (-10
5231  3430          DCA I PRBYTC     /RESTORE COUNTER
5232  1427          TAD I PRBYT      /SEND DATA
5233  6617          TOUT
5234  5774          JMP TPOUTX       /WAIT FOR FLAG
                    /
5235  3423  TR3,    DCA I TPSV       /SAVE VALUE
5236  1520          TAD I STACK      /POP RETURN
5237  3433          DCA I PRRET
5240  2120          ISZ STACK
5241  5773          JMP PRN3         /GO CALCULATE CRC ETC.
                    /
                    /
5242  4115  TAPOUT, CALL TPSET       /FOUND FLAG.  SET UP POINTERS
5243  5247
5244  1433          TAD I PRRET      /AND GET RETURN ADDRESS
5245  3007          DCA 7
5246  5407          JMP I 7
                    /
                    /
5247  1371  TPSET,  TAD ((TRBIT      /SET OUTBIT ROUTINE
5250  3022          DCA BITOUT
5251  1370          TAD (TRDAT       /AND DATA BUFFER
5252  5767          JMP PRSET1       /COMMON CODING
                    /
                    /
                    /
```

```
/
                /
5253  4115  WFUN0,  CALL FUN0A       /GET CALIBRATOR LEVEL FROM SWITCHES
5254  5271
5255  3136          DCA REFLV1       /STORE AWAY
5256  1136  FUN0,   TAD REFLV1       /GET CAL LEVEL AND DISPLAY
5257  4115          CALL PDBCON
5260  6761
5261  3007  FUN0B,  DCA 7
5262  1171          TAD KD1000       /SET D.P.
5263  5766          JMP FUN1B
                /
5264  4115  WSFN0,  CALL FUN0A       /GET DATA
5265  5271
5266  3137          DCA REFLV2       /STORE AWAY
5267  1137  SFUN0,  TAD REFLV2       /GET CH2 CAL LEVEL AND DISPLAY
5270  5257          JMP FUN0+1
                /
                /
5271  6621  FUN0A,  SWREG            /GET NUMBER FROM SWITCHES
5272  4115          CALL BCDBIN      /CONVERT TO BIN
5273  6625
5274  1365          TAD (-1135       /60.5
5275  7510          SPA              /12 BIT ADD OK HERE
5276  1364          TAD (1750        /LESS THAN 60.5 -- ADD 100
5277  1363          TAD (1135        /RESTORE
5300  4115          CALL DPMP        /MULTIPLY BY .03321928
5301  7011
5302  0210          0210             /OR .0210041703 OCTAL
5303  0417          0417             /= 1/(10*10*LOG10(2))
5304  5523          RETURN           /GOT IT DONE
                /
                /
                /
5305  1362  WFUN1,  TAD (PROC2-1
5306  3013          DCA XRT1         /CLEAR PROCESS BUFFER EXCEPT FOR AUTO START
5307  1361          TAD (PROC2-PRDAT
5310  3007          DCA 7
5311  7330          K4000
5312  3413          DCA I XRT1
5313  2007          ISZ 7
5314  5311          JMP .-3
                /
5315  6621          SWREG            /GET MODE NUMBER FROM SWITCHES
5316  0157          AND KD17         /DELETE EXTRANEOUS
5317  3007          DCA 7
5320  1174          TAD KD100        /SAVE ONLY PEAK
5321  0125          AND MODE
5322  1360          TAD (VDIS!STNBLD  /GET STANDBY AND DISPLAY ON
5323  1007          TAD 7            /AND MODE #
5324  3006          DCA 6
                /
5325  1125          TAD MODE         /IS THERE ANYTHING TO SAVE?
5326  0157          AND KD17
5327  1357          TAD (-5          /MUST BE MODE 5 TO 9
5330  7510          SPA
5331  5347          JMP WFUN1A
5332  1357          TAD (-5
5333  7700          SMA CLA
5334  5347          JMP WFUN1A
```

```
5335  1124          TAD MEMFLG      /ARE WE TAKING DATA?
5336  7700          SMA CLA
5337  5347          JMP WFUN1A
5340  3133          DCA DISFL       /YES.  KEEP SHORT BLOCK
5341  6614          SAMPLE
5342  1133          TAD DISFL
5343  7650          SNA CLA
5344  5342          JMP .-2
5345  1125          TAD MODE        /SAVE OLD MODE
5346  5351          JMP .+3
                    /
5347  7200  WFUN1A, CLA
5350  1006          TAD 6
5351  3126          DCA SVMD
5352  1006          TAD 6
5353  3125          DCA MODE
5354  1125          TAD MODE
                    /
                    /
5355  5756          JMP MODJMP&7700
5356  5400          PAGE
5357  7773
5360  4400
5361  7744
5362  0473
5363  1135
5364  1750
5365  6643
5366  5606
5367  3260
5370  0544
5371  5372
5372  5216
5373  4433
5374  0253
5375  7770
5376  5142
5377  0241
                    /
                    /
5400  6607          SELMD           /FIX UP LIGHTS
5401  3101          DCA TSW8F       /CLEAR THRESHOLD FLAG
                    /
5402  1377          TAD (TABMJ      /SET UP JUMP TO MODE PROCESS
5403  1007          TAD 7
5404  3007          DCA 7
5405  1407          TAD I 7
5406  3014          DCA XRT2
5407  6002          IOF
5410  1414          TAD I XRT2
5411  3124          DCA MEMFLG
5412  1414  MODJMP, TAD I XRT2
5413  3007          DCA 7
5414  5407          JMP I 7
                    /
5415  5434  TABMJ,  TABM0-1
5416  5434          TABM1-1
5417  5441          TABM2-1
5420  5444          TABM3-1
5421  5447          TABM4-1
```

```
5422  5441          TABM5-1
5423  5452          TABM6-1
5424  5456          TABM7-1
5425  5452          TABM8-1
5426  5456          TABM9-1
5427  5461          TABMER-1
5430  5461          TABMER-1
5431  5461          TABMER-1
5432  5461          TABMER-1
5433  5461          TABMER-1
5434  5441          TABM15-1
                /
                /
                /
                /
                TABM0,
5435  0100      TABM1,  100
5436  5537              WFUN1G
5437  3510              SW3C
5440  5464              WFUN1C
5441  0005              5
                /
                /
                TABM2,
                TABM5,
5442  0100      TABM15, 100
5443  5464              WFUN1C
5444  0001              1
                /
5445  4110      TABM3,  4110
5446  5521              WFUN1E
5447  5000              5000
                /
5450  0100      TABM4,  100
5451  5521              WFUN1E
5452  5000              5000
                /
                TABM6,
5453  0100      TABM8,  100
5454  5527              WFUN1F
5455  5521              WFUN1E
5456  0140              140
                /
                TABM7,
5457  0100      TABM9,  100
5460  5521              WFUN1E
5461  0140              140
                /
5462  0000      TABMER, 0
5463  6622              ERR0
                /
5464  1414      WFUN1C, TAD I XRT2
5465  3140              DCA TSEC
5466  3141              DCA TMIN
5467  4115      WFUN1D, CALL TTSET
5470  5474
5471  4115              CALL XSW3A
5472  3273
5473  5776              JMP FUN1
                /
```

```
5474  4115  TTSET,  CALL XFUN3
5475  5740
5476  1141          TAD TMIN
5477  7450          SNA
5500  5303          JMP .+3
5501  6601          MIN
5502  5523          RETURN
5503  1140          TAD TSEC
5504  6600          SEC
5505  5523          RETURN
                /
5506  7240  SETMD,  STA
5507  1520          TAD I STACK
5510  3013          DCA XRT1
5511  2120          ISZ STACK
5512  1413          TAD I XRT1
5513  0125          AND MODE
5514  1413          TAD I XRT1
5515  3125          DCA MODE
5516  1125          TAD MODE
5517  6607          SELMD
5520  5413          JMP I XRT1
                /
5521  1414  WFUN1E, TAD I XRT2
5522  3141          DCA TMIN
5523  3140          DCA TSEC
5524  4115          CALL ACCLR      /CLEAR LDN BUFFERS
5525  7703
5526  5267          JMP WFUN1D
                /
5527  4115  WFUN1F, CALL ADDCLK     /SET UP AUTOCAL EVERY SIX HOURS
5530  5010
5531  0473          PROC2-1
5532  0000          0
5533  0000          0
5534  0006          6
5535  0000          0
5536  5212          JMP MODJMP
                /
                /
                /
                /
5537  4115  WFUN1G, CALL SETMD
5540  5506
5541  6777          -CALREL-1
5542  1000          CALREL
5543  5212          JMP MODJMP
                /
5544  1147  WFUN1H, TAD THFLG       /SET UP THRESHOLD SHIFT IF REQUESTED
5545  7710          SPA CLA
5546  5212          JMP MODJMP
                /
5547  1146          TAD THOLD       /IS THRESHOLD AT LEAST 10 DB?
5550  1375          TAD (-0324
5551  7710          SPA CLA
5552  5212          JMP MODJMP
                /
5553  1374          TAD (PROC6-1    /SET CLOCK FOR 10PM
5554  3013          DCA XRT1
5555  3413          DCA I XRT1
```

```
5556  3413          DCA I XRT1
5557  1373          TAD (26
5560  3413          DCA I XRT1
5561  1170          TAD KD2000      /ANY DAY IS OKAY
5562  3413          DCA I XRT1
5563  3413          DCA I XRT1      /SET CLOCK FOR 7AM
5564  3413          DCA I XRT1
5565  1372          TAD (7
5566  3413          DCA I XRT1
5567  1170          TAD KD2000      /ANY DAY IS OKAY
5570  3413          DCA I XRT1
5571  5212          JMP MODJMP
                /
                /
5572  0007          PAGE
5573  0026
5574  0513
5575  7454
5576  5600
5577  5415
                /
5600  1125  FUN1,   TAD MODE        /GET MODE NUMBER
5601  0157          AND KD17        /FROM MODE WORD
5602  3007          DCA 7
5603  5206          JMP FUN1B
5604  3007  FUN1A,  DCA 7           /SAVE
5605  1172          TAD KD400       /SET R. H. DEC. PT.
5606  3020  FUN1B,  DCA 20
5607  3002          DCA 2           /CLEAR SIGN
5610  1007          TAD 7
5611  5231          JMP FUN2A       /AND DISPLAY
                /
                /
                /
                /
5612  4115  WFUN2,  CALL WFUN2A     /GET GAIN CONSTANT
5613  5655
5614  3134          DCA KCAL1
                /
5615  1134  FUN2,   TAD KCAL1       /DISPLAY GAIN CONSTANT
5616  5223          JMP SFUN2+1
                /
5617  4115  WSFN2,  CALL WFUN2A     /GET GAIN CONSTANT CH2
5620  5655
5621  3135          DCA KCAL2
                /
5622  1135  SFUN2,  TAD KCAL2       /GET GAIN CONSTANT AND DISPLAY
5623  4115          CALL DBCON      /CONVERT TO DB
5624  6763
5625  3007          DCA 7
5626  1171          TAD KD1000      /SET DECIMAL POINT
5627  3020          DCA 20
5630  1007          TAD 7           /GET INFO BACK
                /
5631  4115  FUN2A,  CALL FUN2AA
5632  5634
5633  5777          JMP HTSW5+4
                /
5634  4115  FUN2AA, CALL BCDCON     /FORMAT CONVERSION
5635  6702
```

```
5636  6604  FUN2AB, LODIS           /LOAD LOW DIGITS
5637  1007          TAD 7           /GET 3RD DIGIT
5640  0164          AND KD7400
5641  7104          CLL RAL         /START SHIFT
5642  1003          TAD 3
5643  7006          RTL             /MOVE OVER ONE DIGIT
5644  7006          RTL
5645  3007          DCA 7
5646  1002          TAD 2           /GET SIGN
5647  7640          SZA CLA
5650  1163          TAD KD320       /MAKE A MINUS IN HIGH
5651  1007          TAD 7
5652  1020          TAD 20          /GET D.P.
5653  6605          HIDIS
5654  5523          RETURN
                    /
                    /
5655  6621  WFUN2A, SWREG           /GET GAIN CONSTANT
5656  0376          AND (3777
5657  4115          CALL BCDBIN     /CHANGE FORMAT.  RANGE IS ZERO TO 79.9 DB
5660  6625
5661  4115          CALL DPMP       /TO CONVERT TO LOG2, MPY BY .03321928
5662  7011
5663  0210          0210            /OR .0210041703 OCTAL
5664  0417          0417            /= 1/(10**10*LOG10(2))
5665  3007          DCA 7           /NOW IN STANDARD FORM
5666  6621          SWREG           /GET SIGN
5667  7004          RAL
5670  7200          CLA
5671  1007          TAD 7
5672  7430          SZL             /PUT SIGN ON GAIN CONSTANT
5673  7041          CIA
5674  5523          RETURN
                    /
                    /
5675  1125  WFUN3,  TAD MODE        /DELETE CHANNEL FROM MODE
5676  0375          AND (-PKCHN-1   /MASK OUT CURRENT CHANNEL
5677  3007          DCA 7
5700  6622          FUNLO           /GET FLAG FOR SINGLE OR DUAL
5701  0174          AND KD100       /THIS IS BIT
5702  7650          SNA CLA
5703  5310          JMP WFUN3A      /ALWAYS CHANNEL ONE IF SINGLE CHANNEL
5704  6621          SWREG           /GET CHANNEL FROM LOW BIT OF SWREG
5705  7010          RAR
5706  7620          SNL CLA
5707  1174          TAD KD100       /MUST BE CHANNEL 2
5710  1007  WFUN3A, TAD 7
5711  3125          DCA MODE        /NOW HAVE UPDATED MODE WORD
5712  1125          TAD MODE
5713  6607          SELMD           /UPDATE HARDWARE
                    /
5714  1125  FUN3,   TAD MODE        /GET PEAK CHANNEL
5715  0174          AND KD100
5716  7640          SZA CLA
5717  7001          IAC
5720  7001          IAC
5721  5202          JMP FUN1+2      /DISPLAY CHANNEL NO.
```

```
/
5722  6621  WSFN3,  SWREG           /GET VALUE FROM SWITCHES
5723  0374          AND (1777       /MASK OUT SIGN BIT
5724  4115          CALL BCDBIN     /CONVERT
5725  6625
5726  4115          CALL DFMP       /MPY BY .3321928096 OR .252052264 OCTAL
5727  7011
5730  2520          2520
5731  5227          5227
5732  3146          DCA THOLD       /SAVE RESULT
5733  6621          SWREG           /GET FLAG
5734  3147          DCA THFLG       /SAVE USERS DESIRE FOR 10 DB SHIFT SUPPRESSION
              /
5735  3002  SFUN3,  DCA 2           /CLEAR SIGN
5736  1147          TAD THFLG       /OUTPUT THRESHOLD
5737  5773          JMP FUN4A
              /
5740  1134  XFUN3,  TAD KCAL1       /SUBTRACT CAL FROM THRESHOLD
5741  7161          STL CIA         /DO IT 13 BIT
5742  7500          SMA
5743  7100          CLL
5744  1146          TAD THOLD
5745  7010          RAR             /TAKE SQUARE ROOT
5746  1372          TAD (7100       /SUBTRACT HARDWARE CONSTANT
5747  3007          DCA 7           /SAVE VALUE
5750  1007          TAD 7           /GET FRACTION ANTILOG4(X)
5751  7010          RAR
5752  0161          AND KD77
5753  1154          TAD LOGTBL
5754  3006          DCA 6
5755  1406          TAD I 6
5756  7002          BSW
5757  0161          AND KD77
5760  3006          DCA 6
5761  1007          TAD 7           /NOW HANDLE EXPONENT
5762  7510          SPA             /WHAT SIGN?
5763  5771          JMP XFUN3B      /-
5764  7110          CLL RAR         /+  COMBINE
5765  0165          AND KD7700
5766  1006          TAD 6
              /
              /
5767  5770          JMP XFUN3A&7700
5770  6000          PAGE
5771  6007
5772  7100
5773  6037
5774  1777
5775  7677
5776  3777
5777  0273
              /
              /
6000  7106          CLL RTL         /ALIGN FOR HARDWARE
6001  7004          RAL
              /
6002  7450  XFUN3A, SNA             /DON'T LET A ZERO IN
6003  7001          IAC
6004  7041          CIA             /ADDERS NEED COMPLEMENT
6005  6602          THSET
```

```
6006  5523          RETURN
                    /
6007  7002  XFUN3B,  BSW                 /GET NUMBER OF SHIFTS
6010  0161          AND KD77
6011  1165          TAD KD7700          /EXTEND MINUS SIGN
6012  3007          DCA 7
6013  1006          TAD 6               /ALIGN FRACTION
6014  7106          CLL RTL
6015  7410          SKP
6016  7110          CLL RAR
6017  2007          ISZ 7
6020  5216          JMP .-2
6021  5202          JMP XFUN3A
                    /
                    /
                    /
                    /
                    /
/
                    /
                    /
                    /
6022  6621  WSFN4,   SWREG               /GET SECONDS
6023  3140          DCA TSEC
6024  3141          DCA TMIN            /CLEAR MINUTES
                    /
6025  3002  SFUN4,   DCA 2               /CLEAR SIGN
6026  1140          TAD TSEC            /GET SECONDS IN BCD
6027  4115          CALL BCDBIN
6030  6625
6031  5777          JMP FUN0B           /GO OUTPUT
                    /
                    /
                    /
                    /
6032  6621  WFUN4,   SWREG               /READ MINUTES
6033  3141          DCA TMIN
6034  3140          DCA TSEC
                    /
6035  3002  FUN4,    DCA 2               /CLEAR SIGN
6036  1141          TAD TMIN            /GET MINUTES IN BCD
6037  4115  FUN4A,   CALL BCDBIN
6040  6625
6041  5776          JMP FUN1A
                    /
```

```
/
                /
6042  6621  WFUNS,  SWREG            /GET NEW FORMAT WORD
6043  3127          DCA FORMAT
6044  3007          DCA 7            /CLEAR COUNTER
6045  1127          TAD FORMAT
6046  0375          AND (7740        /LOOK ONLY AT DATA BUFFER
6047  7104          CLL RAL          /GET THE NUMBER OF ONES SET
6050  7430          SZL
6051  2007          ISZ 7            /FOUND A ONE
6052  7440          SZA
6053  5247          JMP .-4          /STILL HAS A ONE IN AC
6054  1007          TAD 7
6055  7450          SNA              /DON'T ALLOW ZERO RESULT
6056  7001          IAC
6057  3035          DCA LEN          /SAVE LENGTH PER BLOCK
6060  1035          TAD LEN          /SET UP DIVIDE
6061  7041          CIA
6062  3007          DCA 7
6063  3006          DCA 6            /CLEAR QUOTIENT
6064  1056          TAD MEMTOP       /GET AVAILABLE SPACE
6065  7001          IAC
6066  4115          CALL DIVD        /FIND # OF BLOCKS
6067  6200
6070  7041          CIA
6071  3034          DCA STORE        /UPDATE
6072  7305          K2
6073  0127          AND FORMAT       /GET LENGTH
6074  3007          DCA 7            /THIS IS NOW INITIAL VALUE
6075  1127          TAD FORMAT       /GET LDN'S
6076  7012          RTR
6077  7010          RAR
6100  0374          AND (1           /GET CHAN 1
6101  7430          SZL
6102  7001          IAC              /ADD CHAN 2
6103  3006          DCA 6
6104  7001          IAC              /GET LEVELS BIT
6105  0127          AND FORMAT
6106  7650          SNA CLA
6107  5312          JMP .+3          /SKIP LEVELS
6110  1006          TAD 6            /ADD EXTRA LOCATIONS
6111  1006          TAD 6            /NO. OF CHAN TIMES 2
6112  1006          TAD 6            /THIS NOW HAS EVERYTHING
6113  1007          TAD 7            /GET LENGTHS TOO
6114  3043          DCA LDNLEN       /NOW HAVE # OF LOCS FOR LDN BLOCK
6115  1043          TAD LDNLEN       /SET UP DIVIDE
6116  7041          CIA
6117  3007          DCA 7
6120  3006          DCA 6            /CLEAR QUOTIENT
6121  1373          TAD (LDNTOP-LDNBOT+1  /LENGTH OF LDN BUFFER
6122  4115          CALL DIVD
6123  6200
6124  7041          CIA
6125  3042          DCA LDNST        /UPDATE # OF LDN BLOCKS
6126  6002          IOF              /SET UP MEMORY POINTERS WITH INTERRUPT OFF
6127  7240          STA
6130  3017          DCA XR17
6131  1017          TAD XR17
6132  7001          IAC
6133  3037          DCA XR16
```

```
6134  1037          TAD XR16
6135  3065          DCA DX16
6136  1034          TAD STORE
6137  3036          DCA INFLO
6140  1036          TAD INFLO
6141  3040          DCA OUTFLO
6142  1040          TAD OUTFLO
6143  3064          DCA DFLO
6144  7240          STA
6145  3041          DCA EMPTY
6146  1372          TAD (LDNBOT-1
6147  3016          DCA LDN17
6150  1016          TAD LDN17
6151  7001          IAC
6152  3045          DCA LDN16
6153  1045          TAD LDN16
6154  3067          DCA DLX16
6155  1042          TAD LDNST
6156  3044          DCA LDNIN
6157  1044          TAD LDNIN
6160  3046          DCA LDNOT
6161  1046          TAD LDNOT
6162  3066          DCA DLFLO
6163  7240          STA
6164  3047          DCA LDNMT
6165  3105          DCA TRBLK
6166  6001          ION
                    /
                    /
6167  1127  FUN5,   TAD FORMAT
6170  5771          JMP SFN14A      /OUTPUT FORMAT IN OCTAL
                    /
                    /
6171  6350          PAGE
6172  0571
6173  0101
6174  0001
6175  7740
6176  5604
6177  5261
                    /
                    /
```

```
/
                    /
                    /
                    /
                    /THIS IS DIVIDE FOR + NUMBERS LESS THAN 4095
                    /THIS ROUTINE HAS DIVISOR IN LOC 7, DIVIDEND IN A-C ON ENTRY.
                    /ALSO, LOC 6 MUST HAVE BEEN CLEARED
                    /QUOTIENT IN A-C IN 6, REM IN 5
                    /USES LOCS 5,6 AND 7
6200  7120  DIVD,   STL
6201  1007          TAD 7             /SUBTRACTION METHOD
6202  7430          SZL
6203  5206          JMP .+3           /NEG.  MUST BE DONE
6204  2006          ISZ 6             /POS.  UPDATE QUOTIENT
6205  5200          JMP .-5
6206  7041          CIA
6207  1007          TAD 7             /COMPENSATE REMAINDER FOR GOING TO FAR
6210  7041          CIA
6211  3005          DCA 5             /REMAINDER
6212  1006          TAD 6             /EXIT WITH QUOTIENT IN A-C
6213  5523          RETURN
                    /
                    /
```

```
/
                /
6214  4115  WFUN6,  CALL NUMONE      /GET MINISAMPLE RECORDER ON TIME
6215  6226
6216  3130          DCA RECON
                /
6217  1130  FUN6,  TAD RECON        /RECALL TIME
6220  5777          JMP FUN1A       /AND DISPLAY
                /
                /
6221  4115  WSFN6, CALL NUMONE      /GET MINISAMPLE RECORDER INTERVAL TIME
6222  6226
6223  3131          DCA RECOFF      /THE RESULTS ARE IN MINUTES
                /
6224  1131  SFUN6, TAD RECOFF       /RECALL TIME
6225  5777          JMP FUN1A       /AND DISPLAY
                /
6226  6621  NUMONE, SWREG           /GERT VALUE FROM SWITCHES
6227  4115          CALL BCDBIN     /CONVERT TO BINARY
6230  6625
6231  7450          SNA
6232  7001          IAC             /DON'T LET A ZERO IN
6233  5523          RETURN
                /
                /
                /
                /
6234  4115  WFUN7,  CALL WFUN7A     /GET PRESENT TIME - DAY
6235  6245
6236  3145          DCA CLDAY       /AND SAVE
6237  3144          DCA CLHR        /CLEAR REST OF CLOCK TO PREVENT OVERFLOWS
6240  3143          DCA CLMIN
                /
                /
6241  1145  FUN7,  TAD CLDAY       /GET DAY
6242  5777          JMP FUN1A       /AND DISPLAY
                /
                /
                /
6243  3776  WFUN7B, DCA PRBUF       /ALWAYS CLEAR SECONDS
6244  7410          SKP
6245  3142  WFUN7A, DCA CLSEC       /ALWAYS CLEAR SECONDS
6246  6621          SWREG           /GET SWITCHES
6247  4115          CALL BCDBIN     /AND CONVERT TO BINARY
6250  6625
6251  5523          RETURN          /DONE
                /
                /
                /
6252  4115  WFUN8,  CALL WFUN7A     /GET PRESENT TIME - HOUR
6253  6245
6254  3144          DCA CLHR
6255  3143          DCA CLMIN       /CLEAR REST OF CLOCK TO PREVENT OVERFLOWS
                /
                /
6256  1144  FUN8,  TAD CLHR        /GET HOUR
6257  5777          JMP FUN1A       /AND DISPLAY
                /
                /
6260  4115  WFUN9,  CALL WFUN7A     /GET PRESENT TIME - MINUTE
```

```
6261  6245
6262  3143          DCA CLMIN
                    /
                    /
6263  1143  FUN9,   TAD CLMIN       /GET MINUTE
6264  5777          JMP FUN1A
                    /
                    /
```

```
/
                    /
6265  4115  WSFN7,  CALL WFUN7B     /GET START TIME - DAY
6266  6243
6267  3775          DCA PRBUF+3     /SAVE IN PROCESS BUFFER
6270  3774          DCA PRBUF+2     /CLEAR HOURS
6271  3773          DCA PRBUF+1     /CLEAR MINUTES
                    /
                    /
6272  1775  SFUN7,  TAD PRBUF+3     /GET START DAY
6273  5313          JMP SFUN9+1     /AND DISPLAY
                    /
                    /
6274  1775  WSFN8,  TAD PRBUF+3     /HAS DAY BEEN ENTERED?
6275  7700          SMA CLA
6276  5301          JMP .+3         /YES.
6277  1145          TAD CLDAY       /NO.  USE CURRENT DAY
6300  3775          DCA PRBUF+3
                    /
6301  4115          CALL WFUN7B     /GET HOUR
6302  6243
6303  3774          DCA PRBUF+2     /SAVE
6304  3773          DCA PRBUF+1     /CLEAR MINUTES
                    /
                    /
6305  1774  SFUN8,  TAD PRBUF+2     /GET START HOUR
6306  5313          JMP SFUN9+1     /AND DISPLAY
                    /
                    /
6307  4115  WSFN9,  CALL WFUN7B     /GET START MINUTE
6310  6243
6311  3773          DCA PRBUF+1
                    /
6312  1773  SFUN9,  TAD PRBUF+1
6313  0372          AND (3777       /DON'T DISPLAY OFF FLAG
6314  5777          JMP FUN1A
                    /
```

```
/
                /
6315  6621  WFUN14, SWREG           /GET MEMORY ADDRESS
6316  3121          DCA MEMAD
                /
6317  1121  FUN14,  TAD MEMAD      /READ CURRENT MEM ADDRESS
6320  5350          JMP SFN14A
                /
6321  6000  WSFN14, SKON
6322  7040          CMA
6323  3055          DCA TEMP3
6324  1122          TAD MEMFLD
6325  7106          CLL RTL
6326  7006          RTL
6327  1125          TAD MODE
6330  6607          SELMD
6331  6621          SWREG           /GET CONTENTS OF ADDRESS
6332  3521          DCA I MEMAD     /PUT IT AWAY
6333  1125          TAD MODE
6334  6607          SELMD
6335  2055          ISZ TEMP3
6336  6001          ION
                /
6337  1121  SFUN14, TAD MEMAD      /READ CURRENT CONTENTS
6340  3006          DCA 6           /SET UP DATA GET
6341  1122          TAD MEMFLD
6342  7106          CLL RTL
6343  7006          RTL
6344  4115          CALL GETDAT
6345  7124
6346  2121          ISZ MEMAD
6347  7000          NOP             /SAFETY FIRST
6350  4115  SFN14A, CALL OCTBCD     /CONVERT OCTAL TO BCD
6351  6363
6352  4115          CALL FUN2AB     /PUT NO. IN DISPLAY
6353  5636
6354  5177          JMP SWSET
                /
                /
                /BEGIN EXECUTION AT CURRENT MEMORY ADDRESS IN FIELD 0
6355  5521  WFUN15, JMP I MEMAD
                /
6356  6621  WSFN15, SWREG           /GET FIELD IN LOW BITS
6357  0371          AND (3
6360  3122          DCA MEMFLD
                /
6361  1122  SFUN15, TAD MEMFLD     /TELL CURRENT DATA FIELD
6362  5350          JMP SFN14A
                /
                /
                /THIS ROUTINE CONVERTS 12 BIT BINARY TO 4 DIGIT OCTAL
                /THIS MIMICS THE BINBCD ROUTINE
                /
6363  7421  OCTBCD, MQL             /SAVE INPUT AND CLEAR AC
6364  3007          DCA 7           /CLEAR RESULT
6365  7346          KM3             /SET UP 9 BIT DOUBLE PRECISION SHIFT
                /
                /
6366  5770          JMP OCTB1&7700
6370  6400          PAGE
```

```
6371  0003
6372  3777
6373  0471
6374  0472
6375  0473
6376  0470
6377  5604
                /
                /
6400  3006              DCA 6
6401  7346  OCTB1,      KM3             /INSERT CLEAR BIT AFTER EVERY 3
6402  3005              DCA 5
6403  1007              TAD 7
6404  7521  OCTB2,      SWP
6405  7110              CLL RAR
6406  7521              SWP
6407  7010              RAR
6410  2005              ISZ 5
6411  5204              JMP OCTB2
6412  7110              CLL RAR
6413  3007              DCA 7
6414  2006              ISZ 6
6415  5201              JMP OCTB1
6416  7521              SWP
6417  3003              DCA 3
6420  3002              DCA 2           /SIGN
6421  3020              DCA 20          /D.P.
6422  1007              TAD 7
6423  5523              RETURN
                /
                /
                /
6424  7001  SFUN11,     IAC             /PEAK
6425  7001  FUN11,      IAC             /TIME
6426  7001  SFUN10,     IAC             /CH2
6427  7001  FUN10,      IAC             /CH1
6430  3132              DCA DISPCH
6431  1133              TAD DISFL       /SET DATA READY IF POSSIBLE
6432  7650              SNA CLA
6433  5777              JMP HTSW5+4
6434  7240              STA
6435  3133              DCA DISFL
6436  5777              JMP HTSW5+4
                /
                /
                /
6437  4115  WFUN12,     CALL WSFN5A     /GET BIN NUMBER AND LOAD BUFFER REGISTERS
6440  6530
6441  0034              STORE
6442  0035              LEN
6443  0064              DFLO
6444  0000              0
6445  3065              DCA DX16
                /
6446  1376  FUN12,      TAD (DMP1       /RESET PHASE
6447  3150              DCA IDMP
6450  5777              JMP HTSW5+4
                /
                /
                /

6451  4115  WSFN12,     CALL WSFN5A     /GET LDN BIN NUMBER AND LOAD BUFFER REGISTERS
6452  6530
6453  0042              LDNST
6454  0043              LDNLEN
6455  0066              DLFLO
6456  0572              LDNBOT
6457  3067              DCA DLX16
```

```
                        /
6460  1375  SFUN12,  TAD (LDMP1        /USE OTHER DUMP TABLE
6461  5247           JMP FUN12+1
                        /
/
                        /
                        /
                        /
6462  6002  WSFN5,   IOF               /DON'T CONFUSE INTERRUPT HANDLER
6463  4115           CALL WSFN5A       /GET BIN NUMBER FOR DATA IN
6464  6530
6465  0034           STORE
6466  0035           LEN
6467  0036           INFLO
6470  7777           -1
6471  3017           DCA XR17
6472  6001           ION               /OK NOW!
                        /
6473  1034  SFUN5,   TAD STORE
6474  7041           CIA
6475  7001           IAC
6476  1036           TAD INFLO
6477  5774           JMP FUN1+2
                        /
6500  4115  WFUN13,  CALL WSFN5A      /GET PRINTER OUTPUT BIN NUMBER
6501  6530
6502  0034           STORE
6503  0035           LEN
6504  0040           OUTFLO
6505  0000           0
6506  3037           DCA XR16
                        /
6507  1034  FUN13,   TAD STORE
6510  7041           CIA
6511  7001           IAC
6512  1040           TAD OUTFLO
6513  5774           JMP FUN1+2
                        /
6514  4115  WSFN13,  CALL WSFN5A      /LDN PRINT LOCATION
6515  6530
6516  0042           LDNST
6517  0043           LDNLEN
6520  0046           LDNOT
6521  0572           LDNBOT
6522  3045           DCA LDN16
                        /
                        /
6523  1042  SFUN13,  TAD LDNST
6524  7041           CIA
6525  7001           IAC
6526  1046           TAD LDNOT
6527  5774           JMP FUN1+2
                        /
                        /
                        /
6530  7240  WSFN5A,  STA               /GET ARGUMENT TABLE ADDRESS
6531  1520           TAD I STACK
6532  3013           DCA XRT1
6533  2120           ISZ STACK
6534  1413           TAD I XRT1
6535  3020           DCA 20            /NUMBER OF BLOCKS ALLOWED IN BUFFER
6536  1413           TAD I XRT1
6537  3021           DCA 21            /NUMBER OF MEMORY WORDS PER BLOCK
6540  1413           TAD I XRT1
```

```
6541  3022            DCA 22          /FLO COUNTER
                      /
6542  4115            CALL NUMONE     /GET USER'S VALUE
6543  6226
6544  1373            TAD (-1         /ADJUST TO INTERNAL COUNT
6545  1420  WSFN5B,   TAD I 20        /STORE
6546  3422            DCA I 22        /SET FLO
6547  1421            TAD I 21        /GET NUMBER OF WORDS PER BLOCK
6550  3006            DCA 6
6551  1420            TAD I 20
6552  7041            CIA             /NUMBER OF BLOCKS IN BUFFER
6553  1422            TAD I 22
6554  3005            DCA 5           /NOW HAVE LOCAL BIN NUMBER
6555  4115            CALL MPLY       /LEN * BIN #.  /EQUIVALENT MEMORY ADDRESS
6556  6736
6557  1413            TAD I XRT1      /ADD IN BASE OF BUFFER
6560  5413            JMP I XRT1      /EXIT
                    /
                    /
                    /
                    /
/
                    /
6561  1772  HSW4,    TAD HTSW5+2     /EXECUTE FUNCTION.  TEST FOR DISPLAY
6562  7750            SPA SNA CLA
6563  5771            JMP HTSW4+4     /NO.  IGNORE
6564  1370            TAD (HIFUN
6565  5767            JMP HSW5X       /GO THERE.
                    /
                    /
6567  6604            PAGE
6570  7440
6571  0267
6572  0271
6573  7777
6574  5602
6575  7230
6576  7045
6577  0273
                    /
                    /
6600  1777  HSW5,    TAD HTSW4+2     /DISPLAY FUNCTION.  IS EXECUTE SET?
6601  7740            SMA SZA CLA
6602  5776            JMP HSW4        /DO A WRITE
6603  1375            TAD (FUNLOC     /DO A READ.  GET FUNCTION
6604  3007  HSW5X,    DCA 7
6605  7307            K4              /IS SHIFT SET?
6606  0107            AND TFUN1
6607  7640            SZA CLA
6610  1160            TAD KD20        /YES.  DIFFERENT TABLE
6611  1007            TAD 7
6612  3007            DCA 7
6613  1157            TAD KD17
6614  0106            AND TFUN0       /GET TABLE ENTRY
6615  1007            TAD 7
6616  3007            DCA 7
6617  1407            TAD I 7         /GET ADDRESS
6620  3007            DCA 7
6621  5407            JMP I 7         /GO DO IT.
                    /
6622  4115  ERR0,    CALL ERROR1+1   /NON-EXISTANT FUNCTION
6623  3326
6624  5774            JMP HTSW4+4     /EXIT
                    /
```

```
/
6625  3007  BCDBIN, DCA 7           /SAVE AC.
6626  1007          TAD 7           /256H + 16M + L
6627  0164          AND KD7400      /GET HIGH DIGIT
6630  7112          CLL RTR         /64H
6631  3006          DCA 6
6632  1006          TAD 6
6633  7010          RAR             /32H
6634  1006          TAD 6           /96H
6635  7041          CIA
6636  1007          TAD 7
6637  3007          DCA 7           /160H + 16M + L
6640  1007          TAD 7
6641  0166          AND KD7760      /GET HIGH AND MID
6642  7112          CLL RTR         /40H + 4M
6643  3006          DCA 6
6644  1006          TAD 6
6645  7010          RAR             /20H + 2M
6646  1006          TAD 6           /60H + 6M
6647  7041          CIA
6650  1007          TAD 7           /100H + 10M + L
6651  5523          RETURN
              /
6652  3114  SUBX,   DCA SUB1        /SAVE AC
6653  7060          CMA CML         /ADJUST STACK POINTER
6654  1120          TAD STACK
6655  3120          DCA STACK
6656  1115          TAD SUB         /PUT RETURN ADDRESS ON STACK
6657  7001          IAC
6660  3520          DCA I STACK
6661  1515          TAD I SUB       /GET DESTINATION ADDRESS
6662  3115  SUBX1,  DCA SUB
6663  1114          TAD SUB1        /RESTORE AC
6664  5515          JMP I SUB       /GO TO DESTINATION ADDRESS
              /
6665  3114  RETN,   DCA SUB1        /SAVE AC
6666  1520          TAD I STACK     /GET RETURN ADDRESS
6667  2120          ISZ STACK       /DELETE ENTRY
6670  5262          JMP SUBX1       /RETURN COMPLETE
              /
6671  3114  INDIR,  DCA SUB1        /INDIRECT ADDRESSING FOR SUBROUTINE CALL
6672  1520          TAD I STACK     /GET ARGUMENT ADDRESS
6673  3115          DCA SUB
6674  2520          ISZ I STACK     /ADJUST RETURN ADDRESS
6675  1515          TAD I SUB       /GET POINTER TO POINTER
6676  3115          DCA SUB
6677  1515          TAD I SUB
6700  3115          DCA SUB
6701  5261          JMP SUBX1-1
```

```
/
                /THIS ROUTINE DOES A 4 BCD DIGIT CONVERSION
                /USES REG 3 THRU 7.  ENTER WITH NO. IN AC
                /THE LARGEST NUMBER THAT CAN BE CONVERTED IS 7777 OCTAL OR 4095 DECIMAL
6702  3004  BCDCON, DCA 4            /SAVE NUMBER
6703  3006         DCA 6
6704  1373         TAD (-1750       /FORM THOUSANDS DIGIT
6705  3007         DCA 7
6706  1004         TAD 4            /GET NUMBER
6707  4115         CALL DIVD
6710  6200
6711  3003         DCA 3            /SAVE THOUSANDS
                /
6712  3006         DCA 6            /CLEAR QUOTIENT
6713  1372         TAD (-144        /FORM HUNDREDS
6714  3007         DCA 7
6715  1005         TAD 5
6716  4115         CALL DIVD
6717  6200
6720  7106         CLL RTL          /ALIGN RESULT
6721  7006         RTL
6722  3006         DCA 6
6723  1371         TAD (-12         /FORM TENS DIGIT
6724  3007         DCA 7
6725  1005         TAD 5
6726  4115         CALL DIVD
6727  6200
6730  7106         CLL RTL          /ALIGN RESULT
6731  7006         RTL
6732  1005         TAD 5            /GET ONES
6733  3007         DCA 7
6734  1007         TAD 7
6735  5523         RETURN           /AC AND 7 HAVE LO.  3 HAS HIGH
                /


/
                /
                /
                /
                /THIS ROUTINE TAKES (46+130N) CYCLES
                /FOR N=6, IT TAKES 826 CYCLES MAX)
                /USES REG 5 THRU 7.  LOAD 5 WITH MULTIPLICAND.
                /LOAD 6 WITH MULTIPLIER.
                /
6736  1155  MPLY,   TAD KM12         /DO 12 BITS
6737  3007         DCA 7            /SAVE # OF PLACES TO MPY
6740  1006         TAD 6            /PICK UP MULTIPLIER
6741  7421         MQL              /LOAD IT
                /
6742  7104  MPLS,   CLL RAL          /D. P. SHIFT OF RESULT
6743  7521         SWP
6744  7004         RAL              /ALSO PICKS UP HIGH ORDER
6745  7521         SWP              /BIT OF MULTIPLIER.
6746  7420         SNL              /DO WE ADD INTO LOW ORDER?
6747  5356         JMP MPLT
6750  7100         CLL
6751  1005         TAD 5            /YES. GET MULTIPLICAND
6752  7521         SWP              /CARRY IS IN LINK.  D.P. ADD
6753  7430         SZL              /UPDATE HIGH ORDER
6754  7001         IAC
6755  7521         SWP              /RESTORE POSITIONS
6756  2007  MPLT,   ISZ 7           /ARE WE DONE?
6757  5342         JMP MPLS
                /
6760  5523         RETURN           /DONE.  LOW ORDER IN AC
                                    /HIGH ORDER IN MQ
                /
                /
                /
                /
                /
```

```
/
                /THIS ROUTINE CONVERTS THE LOG FORMAT STORED
                /IN THE BUFFER TO 10*DB STORED AS A ONE WORD INTEGER.
                /THIS ROUTINE USES REGS 3 THRU 7.
                /RESULT IN AC AND SIGN IN REG 2
                /ENTER WITH NO. IN AC.
                /
6761  7100      PDBCON, CLL             /THIS ENTRY FOR LARGE POSITIVE NUMBERS
6762  5770              JMP DBCONA
                /
6763  7100      DBCON,  CLL             /GET SIGN
6764  7510              SPA
6765  7061              CML CIA         /GET ABSOLUTE VALUE
                /
6766  5770              JMP DBCONA
6770  7000              PAGE
6771  7766
6772  7634
6773  6030
6774  0267
6775  7400
6776  6561
6777  0265
                /
7000  3007      DBCONA, DCA 7
7001  7004              RAL             /SAVE SIGN
7002  3002              DCA 2
7003  1007              TAD 7
7004  4115              CALL DPMP       /MULTIPLY BY 30.103 OR 36.0646
7005  7011
7006  0036              0036
7007  0646              0646            /ONE WORD INTEGER NNN.M
7010  5523              RETURN
                /
                /
                /
                /
/
                /THIS ROUTINE USES REGS 3 THRU 7 AND XRT1
7011  3005      DPMP,   DCA 5           /MULTIPLICAND IN AC
7012  7240              STA
7013  1520              TAD I STACK     /GET ADDRESS OF CALL
7014  3013              DCA XRT1
7015  2120              ISZ STACK
7016  1413              TAD I XRT1      /GET HIGH BITS OF MULTIPLIER
7017  3004              DCA 4           /KEEP FOR A WHILE
7020  1413              TAD I XRT1      /GET LOW BITS OF MULTIPLIER
7021  3006              DCA 6           /SET UP MULTIPLY
                /
7022  4115              CALL MPLY       /FIRST MULTIPLY
7023  6736
7024  7521              SWP             /IGNORE LOW WORD
7025  3003              DCA 3           /SAVE HIGH WORD
7026  1004              TAD 4           /GET HIGH MULTIPLIER
7027  3006              DCA 6
7030  4115              CALL MPLY
7031  6736
                /
7032  7100              CLL             /DO A D.P. ADD
7033  1003              TAD 3           /MIDDLE RESULT
7034  1175              TAD KD40        /EQUIVALENT TO .5 DECIMAL
7035  0165              AND KD7700
7036  3004              DCA 4
7037  7521              SWP
7040  7430              SZL             /UPDATE HIGH RESULT
7041  7001              IAC
                /
7042  1004              TAD 4           /EXIT WITH LOW RESULT IN AC
7043  7002              BSW             /NOW IT'S IN MACHINE FORMAT
7044  5413              JMP I XRT1      /XX.XX
                /
                /
                /
                /
```

```
/
                /
7045  1777  DMP1,   TAD HTSW5+2          /IS DISPLAY ON?
7046  7750          SPA SNA CLA
7047  5343          JMP DMP1A            /NO. ZAP DUMP
                /
7050  1172          TAD KD400            /WAS TIME SPECIFIED?
7051  0127          AND FORMAT
7052  7650          SNA CLA
7053  5274          JMP DMP1B
7054  3007          DCA 7                /YES.  BUT WHERE IS IT?
7055  1127          TAD FORMAT
7056  0164          AND KD7400           /LOOK AT PRIOR BITS ONLY
7057  7104          CLL RAL
7060  7430          SZL                  /FOUND A ONE
7061  2007          ISZ 7                /DONE?
7062  7440          SZA
7063  5257          JMP .-4
                /
7064  7240          STA
7065  1007          TAD 7                /MUST BE AT LEAST ONE
7066  1065          TAD DX16             /POINTS TO VALUE IN BUFFER
7067  3006          DCA 6
7070  1160          TAD KD20
7071  4115          CALL GETDAT
7072  7124
7073  7410          SKP                  /GET VALUE
7074  7333  DMP1B,  K6000               /THIS IS FOR UNSPECIFIED TIME
7075  3070          DCA DTIME
                /
7076  1065          TAD DX16             /GET ADDRESS OF DATA
7077  3073          DCA DUMPT1
7100  1376          TAD (DMASK
7101  3072          DCA DUMPT
7102  1375          TAD (DMP2
7103  3150          DCA IDMP
7104  1336          TAD DELAY2           /SET DELAY
7105  3071          DCA DUMPFL           /GET BIN NUMBER
7106  1034          TAD STORE            /STARTING AT ONE
7107  7041          CIA
7110  7001          IAC                  /ADD REMAINDER
7111  1064          TAD DFLO             /GO DISPLAY
7112  5774          JMP FUN1+2
                /
7113  2071  DMP2,   ISZ DUMPFL
7114  5773          JMP TDMPX
                /
7115  1336          TAD DELAY2
7116  3071          DCA DUMPFL
                /
7117  4115          CALL DUMPLK
7120  7261
7121  5346          JMP DMP3
7122  7010          DHTAB-DMASK-1
7123  0020          20
                /
                /
7124  7421  GETDAT, MQL                  /SAVE FIELD CONTAINED IN AC
7125  6000          SKON                 /SAVE INTERRUPT STATUS
7126  7040          CMA
```

```
7127  3055          DCA TEMP3
7130  7501          MQA             /RETRIEVE FIELD
7131  1125          TAD MODE
7132  6607          SELMD           /SET FIELD REGISTER
7133  1406          TAD I 6         /GET DATA FROM DATA FIELD
7134  7421          MQL
7135  1125          TAD MODE        /RESTORE HARDWARE
7136  6607  DELAY2, SELMD
7137  2055          ISZ TEMP3       /RESTORE INTERRUPT TO PREVIOUS STATUS
7140  6001          ION
7141  7501          MQA             /RETRIEVE DATA
7142  5523          RETURN
                    /
                    /
7143  1151  DMP1A,  TAD DMPX        /NO DATA.  ZAP DUMP
7144  3150          DCA IDMP
7145  5773          JMP TDMPX
                    /
                    /
7146  7200  DMP3,   CLA
7147  1073          TAD DUMPT1      /UPDATE BUFFER POINTER
7150  3065          DCA DX16
7151  2064          ISZ DFLO        /DOUBLE PRECISION
7152  5772          JMP FUN12       /DO A RESET
7153  1034          TAD STORE       /WRAPPED AROUND
7154  3064          DCA DFLO
7155  3065          DCA DX16
7156  5772          JMP FUN12
                    /
                    /
                    /
                    /
7157  0010  LMASK,  10              /CH1 LDN
7160  0004          4               /CH2 LDN
7161  0002          2               /TIME DURATION DAY
7162  0002          2               /TIME DURATION NIGHT
7163  0011          11              /LEQ DAY CH1
7164  0011          11              /LEQ NIGHT CH1
7165  0005          5               /LEQ DAY CH2
7166  0005          5               /LEQ NIGHT CH2
7167  0000          0               /END OF TABLE
                    /
                    /
7172  6446          PAGE
7173  0247
7174  5602
7175  7113
7176  0167
7177  0271
                    /
                    /
                    /
7200  7215  DHTAB,  DCH
7201  7215          DCH
7202  5257          FUN0+1          /PK
7203  5623          SFUN2+1         /TIME
7204  5623          SFUN2+1         /GAIN CH1
7205  5623          SFUN2+1         /GAIN CH2
7206  7207          DCLK
                    /
```

```
7207  3007  DCLK,   DCA 7
7210  3002          DCA 2
7211  1170          TAD KD2000
7212  3020          DCA 20
7213  1007          TAD 7
7214  5777          JMP FUN2A
                /
                /
                /
7215  3007  DCH,    DCA 7
7216  1070          TAD DTIME
7217  5346          JMP DISUB       /GO DISPLAY
                /
                /
                /
7220  5257  LTAB,   FUN0+1
7221  5257          FUN0+1
7222  5623  LTAB1,  SFUN2+1
7223  5623          SFUN2+1
7224  5257          FUN0+1
7225  5257          FUN0+1
7226  5257          FUN0+1
7227  5257          FUN0+1
                /
7230  1776  LDMP1,  TAD HTSW5+2
7231  7750          SPA SNA CLA     /IS DISPLAY ON?
7232  5775          JMP DMP1A       /NO.  ZAP DUMP
                /
7233  1067          TAD DLX16       /SET UP DATA POINTER
7234  3073          DCA DUMPT1
7235  1374          TAD (LMASK      /SET UP MASK TABLE POINTER
7236  3072          DCA DUMPT
7237  1265          TAD DELAY1      /SET USER READ TIME
7240  3071          DCA DUMPFL
7241  1373          TAD (LDMP2      /DON'T RETURN HERE
7242  3150          DCA IDMP
7243  1042          TAD LDNST       /GET BIN NUMBER
7244  7041          CIA             /STARTING AT ONE
7245  7001          IAC
7246  1066          TAD DLFLO
7247  5772          JMP FUN1+2
                /
7250  2071  LDMP2,  ISZ DUMPFL      /WAIT A WHILE
7251  5771          JMP TDMPX
                /
7252  1265          TAD DELAY1      /SET USER READ TIME
7253  3071          DCA DUMPFL
                /
                /
7254  4115          CALL DUMPLK
7255  7261
7256  5312          JMP LDMP3
7257  0040          LTAB-LMASK-1
7260  0000          0
                /
                /
7261  1472  DUMPLK, TAD I DUMPT     /ROUTINE TO READ THE FORMAT WORDS
7262  7450          SNA
7263  5523          RETURN          /FOUND END OF TABLE
7264  0127          AND FORMAT
```

```
7265  7041  DELAY1, CIA              /CAN MATCH MORE THAN ONE BIT THIS WAY
7266  1472          TAD I DUMPT
7267  2072          ISZ DUMPT
7270  7640          SZA CLA
7271  5261          JMP DUMPLK       /KEEP LOOKING
7272  1520          TAD I STACK      /OK.  POP RETURN
7273  3013          DCA XRT1
7274  2120          ISZ STACK
7275  1413          TAD I XRT1       /GET TABLE OFFSET
7276  1072          TAD DUMPT
7277  3007          DCA 7
7300  1407          TAD I 7          /GET ADDRESS WHERE TO GO TO
7301  3007          DCA 7
7302  1073          TAD DUMPT1
7303  3006          DCA 6
7304  1413          TAD I XRT1       /GET FIELD
7305  4115          CALL GETDAT      /GET DATA
7306  7124
7307  2073          ISZ DUMPT1
7310  5407          JMP I 7
7311  5407          JMP I 7          /GO DISPLAY
                /
                /
7312  7200  LDMP3,  CLA
7313  1073          TAD DUMPT1       /UPDATE BUFFER POINTER
7314  3067          DCA DLX16
7315  2066          ISZ DLFLO
7316  5770          JMP SFUN12
7317  1042          TAD LDNST
7320  3066          DCA DLFLO
7321  1367          TAD (LDNBOT
7322  3067          DCA DLX16
7323  5770          JMP SFUN12
                /
7324  7201  DISUP,  CLA IAC          /SET FLAG TO PREVENT REPEAT
7325  3133          DCA DISFL
7326  1132          TAD DISPCH       /WHICH CHANNEL?
7327  7450          SNA
7330  5766          JMP TDISX        /NONE
7331  1334          TAD DISTAB
7332  3007          DCA 7            /AND INDEX INTO TABLE
7333  5407          JMP I 7          /GO THERE
                /
                /
7334  7334  DISTAB, DISTAB
7335  5341          JMP DISCH1
7336  5343          JMP DISCH2
7337  5362          JMP DISTIM
7340  5364          JMP DISPK
                /
7341  1057  DISCH1, TAD CH1          /GET DATA
7342  7410          SKP
7343  1060  DISCH2, TAD CH2
7344  3007          DCA 7
7345  1062          TAD TIME
7346  3006  DISUB,  DCA 6
7347  7330          K4000            /IS DISPLAY SEL?
7350  0106          AND TFUN0
7351  7650          SNA CLA
7352  1006          TAD 6            /YES.  ADD TIME
```

```
7353  7100          CLL               /NOW DO 13 BIT ADD
7354  7510          SPA               /SET LINK ACCORDING TO SIGN OF TIME
7355  7020          CML
7356  1007          TAD 7             /RETRIEVE DATA
7357  7430          SZL
7360  5622  DISUM,  JMP I LTAB1
7361  5620          JMP I LTAB
                    /
7362  1062  DISTIM, TAD TIME
7363  5622          JMP I LTAB1
                    /
7364  1061  DISPK,  TAD PK
7365  5620          JMP I LTAB
                    /
                    /
7366  0246          PAGE
7367  0572
7370  6460
7371  0247
7372  5602
7373  7250
7374  7157
7375  7143
7376  0271
7377  5631
                  /
                  /
                  /

/
                  /
                  /
                  /
                  /FUNCTION JUMP TABLE.
7400  5256  FUNLOC, FUN0            /CALIBRATOR LEVEL CH1
7401  5600          FUN1            /MODE OF OPERATION
7402  5615          FUN2            /GAIN CONSTANT CH1
7403  5714          FUN3            /PEAK DETECTOR CHANNEL
7404  6035          FUN4            /BLOCK TIME MINUTES
7405  6167          FUN5            /STORAGE FORMAT
7406  6217          FUN6            /MINISAMPLE RECORD TIME IN SECONDS
7407  6241          FUN7            /PRESENT TIME DAYS
7410  6256          FUN8            /PRESENT TIME HOURS
7411  6263          FUN9            /PRESENT TIME MINUTES
7412  6427          FUN10           /LEVEL CH1
7413  6425          FUN11           /SAMPLE LENGTH
7414  6446          FUN12           /BLOCK DATA DUMP STARTING LOCATION
7415  6507          FUN13           /BLOCK DATA PRINT STARTING LOCATION
7416  6317          FUN14           /READ MEMORY ADDRESS
7417  2100          LBEGIN          /READ PROGRAM FROM CASSETTE
                    /
7420  5267          SFUN0           /CALIBRATOR LEVEL CH2
7421  0273          HTSW5+4         /MAKE THIS POSITION A NOP
7422  5622          SFUN2           /GAIN CONSTANT CH2
7423  5735          SFUN3           /READ CH1 THRESHOLD
7424  6025          SFUN4           /BLOCK TIME IN SECONDS
7425  6473          SFUN5           /NEXT DATA BLOCK STORAGE LOCATION
7426  6224          SFUN6           /MINISAMPLE RECORD SAMPLE INTERVAL IN MINUTES
7427  6272          SFUN7           /START TIME DAYS
7430  6305          SFUN8           /START TIME HOURS
7431  6312          SFUN9           /START TIME MINUTES
7432  6426          SFUN10          /LEVEL CH2
7433  6424          SFUN11          /PEAK
7434  6460          SFUN12          /LDN DUMP STARTING LOCATION
7435  6523          SFUN13          /LDN PRINT STARTING LOCATION
7436  6337          SFUN14          /READ MEMORY CONTENTS IN DATA FIELD
7437  6361          SFUN15          /READ MEMORY FIELD
                    /
```

```
7440  5253  HIFUN,  WFUN0          /SET CALIBRATOR LEVEL CH1
7441  5305          WFUN1          /SET MODE OF OPERATION
7442  5612          WFUN2          /SET GAIN CONSTANT CH1
7443  5675          WFUN3          /SET PEAK DETECTOR CHANNEL
7444  6032          WFUN4          /SET BLOCK TIME IN MINUTES
7445  6042          WFUN5          /SET STORAGE FORMAT
7446  6214          WFUN6          /SET MINISAMPLE RECORD TIME IN SECONDS
7447  6234          WFUN7          /SET PRESENT TIME DAYS
7450  6252          WFUN8          /SET PRESENT TIME HOURS
7451  6260          WFUN9          /SET PRESENT TIME MINUTES
7452  6622          ERR0
7453  6622          ERR0
7454  6437          WFUN12         /SET BLOCK DATA DUMP STARTING LOCATION
7455  6500          WFUN13         /SET BLOCK DATA PRINT STARTING LOCATION
7456  6315          WFUN14         /WRITE MEMORY ADDRESS
7457  6355          WFUN15         /BEGIN EXECUTION AT CURRENT MEMORY ADDRESS
                /
7460  5264          WSFN0          /SET CALIBRATOR LEVEL CH2
7461  0273          HTSW5+4        /MAKE THIS POSTION A NOP
7462  5617          WSFN2          /SET GAIN CONSTANT CH2
7463  5722          WSFN3          /SET CH1 THRESHOLD
7464  6022          WSFN4          /SET BLOCK TIME IN SECONDS
7465  6462          WSFN5          /SET NEXT DATA BLOCK STORAGE LOCATION
7466  6221          WSFN6          /SET MINISAMPLE RECORD SAMPLE INTERVAL IN MINUTES
7467  6265          WSFN7          /SET START TIME DAYS
7470  6274          WSFN8          /SET START TIME HOURS
7471  6307          WSFN9          /SET START TIME MINUTES
7472  6622          ERR0
7473  6622          ERR0
7474  6451          WSFN12         /SET LDN DUMP STARTING LOCATION
7475  6514          WSFN13         /SET LDN PRINT STARTING LOCATION
7476  6321          WSFN14         /WRITE MEMORY CONTENTS IN DATA FIELD
7477  6356          WSFN15         /WRITE MEMORY FIELD
                /
                /
                /
                /
```

```
/
                    /
                    /THIS IS THE LOG AND ANTI-LOG TABLES
                    /HIGH 6 BITS IS ANTI LOG BASE 4
                    /LOW 6 BITS IS LOG BASE 2
                    LOGTAB,
7500    2000        2000;
7501    2001        2001;
7502    2103        2103;
7503    2104        2104;
7504    2106        2106;
7505    2207        2207;
7506    2210        2210;
7507    2312        2312
7510    2313        2313;
7511    2314        2314;
7512    2415        2415;
7513    2417        2417;
7514    2520        2520;
7515    2521        2521;
7516    2622        2622;
7517    2623        2623
7520    2725        2725;
7521    2726        2726;
7522    3027        3027;
7523    3030        3030;
7524    3131        3131;
7525    3132        3132;
7526    3233        3233;
7527    3234        3234
7530    3335        3335;
7531    3336        3336;
7532    3437        3437;
7533    3540        3540;
7534    3541        3541;
7535    3642        3642;
7536    3743        3743;
7537    3744        3744
7540    4045        4045;
7541    4146        4146;
7542    4147        4147;
7543    4250        4250;
7544    4351        4351;
7545    4452        4452;
7546    4453        4453;
7547    4554        4554
7550    4655        4655;
7551    4756        4756;
7552    5057        5057;
7553    5157        5157;
7554    5160        5160;
7555    5261        5261;
7556    5362        5362;
7557    5463        5463
7560    5564        5564;
7561    5664        5664;
7562    5765        5765;
7563    6066        6066;
7564    6167        6167;
7565    6270        6270;
7566    6470        6470;
7567    6571        6571
7570    6672        6672;
7571    6773        6773;
7572    7074        7074;
7573    7174        7174;
7574    7375        7375;
7575    7476        7476;
7576    7577        7577;
7577    7777        7777
                    /
                    /
                    /
```

```
/
                /
                *LOPG
                /
                /
                /   INPUT LEADER CHECK (AT LEAST 4 OF 0377 CODE AND A 0000)
                /
2100  4115      LBEGIN, CALL SETMD      /START TAPE
2101  5506
2102  5777              -MINITA-1
2103  2000              MINITA
2104  1372              TAD (STATIN     /ENABLE FLAG TESTING
2105  3771              DCA TPIN
2106  4115              CALL TINSET     /SET UP POINTERS
2107  7753
2110  6627              TIN             /CLEAR TIN FLAG
2111  7200              CLA
2112  1370              TAD (-4         /LOOK FOR LEADER
2113  3424              DCA I PRSTCT    /FIRST GET 4 OR MORE 200'S
2114  4115      IX,     CALL UIN        /GET INPUT CHAR
2115  1676
2116  1367              TAD (-377
2117  7640              SZA CLA         /IS IT LEADER?
2120  5300              JMP LBEGIN      /NO.  TRY AGAIN
2121  2424              ISZ I PRSTCT    /KEEP COUNT
2122  5314              JMP IX
2123  3424      IX1,    DCA I PRSTCT
2124  4115              CALL UIN        /GO GET A CHAR
2125  1676
2126  7110              CLL RAR
2127  7430              SZL             /COUNT THE ONE'S
2130  2424              ISZ I PRSTCT
2131  7440              SZA             /DONE WITH WORD?
2132  5326              JMP .-4
                /
2133  7346              KM3
2134  1424              TAD I PRSTCT    /NO MORE THAN 3 ONE'S ALLOWED
2135  7740              SZA SMA CLA
2136  5323              JMP IX1         /NO.  KEEP LOOKING
                /
                /
                /MAIN INPUT ROUTINE
                /
2137  4115      UARTI,  CALL SETMD      /TURN OFF DISPLAY
2140  5506
2141  3777              -VDIS-1
2142  0000              0
2143  3426              DCA I PRREM     /ZERO CHECKSUM
2144  3431              DCA I PRPAR     /CLEAR ERROR STATUS BUFFER
2145  3427              DCA I PRBYT     /CLEAR ERROR ADDRESS (WORD PAIR)
2146  4115              CALL DLOOP      /GET FIELD
2147  7713
2150  3070              DCA DTIME
2151  1425              TAD I PRCNT     /GET FIRST ADDRESS
2152  3071              DCA DUMPFL
2153  4115              CALL DLOOP      /GET LAST ADDRESS
2154  7713
2155  3072              DCA DUMPT
2156  1425              TAD I PRCNT     /GET NUMBER OF WORDS TO TRANSFER
2157  3073              DCA DUMPT1
```

```
                    /
                    /
2160  5766          JMP UARTI1      /NEED MORE ROOM
2166  1642          PAGE
2167  7401
2170  7774
2171  0253
2172  6626
                    /
                    *VLOPG
                    /
1642  4115  UARTI1, CALL DLOOP      /GET DATA
1643  7713
1644  4115          CALL PUTDAT
1645  1725
1646  2073          ISZ DUMPT1      /TEST FOR END OF BLOCK
1647  7410          SKP
1650  5256          JMP ICHK
1651  1425          TAD I PRCNT
1652  4115          CALL PUTDAT
1653  1725
1654  2073          ISZ DUMPT1      /TEST FOR END OF BLOCK
1655  5242          JMP UARTI1
                    /
                    /
1656  4115  ICHK,   CALL UIN        /GET CHECKSUM
1657  1676
1660  7200          CLA
1661  1426  ITPKILL,  TAD I PRREM   /CHECK SUM SHOULD BE ZERO
1662  0364          AND (377        /TO 8 BITS
1663  4115          CALL OCTBCD     /SHOW IN DISPLAY
1664  6363
1665  4115          CALL FUN2AB     /IN BCD
1666  5636
1667  4115          CALL SETMD      /TURN OFF TAPE AND TURN ON DISPLAY
1670  5506
1671  1777          -VDIS-MINITA-1
1672  4000          VDIS
1673  1363          TAD (5000+TPINX /STOP TAPE CHECKING
1674  3762          DCA TPIN
                    /
1675  5761          JMP TPINX       /DONE.....
                    /
                    /
                    /
                    /
                    /         UIN(UART INPUT SUBROUTINE)
                    /
1676  1520  UIN,    TAD I STACK     /SAVE RETURN IN BUFFER
1677  3433          DCA I PRRET
1700  2120          ISZ STACK
1701  5761          JMP TPINX       /DO OTHER THINGS UNTIL TAPE IS READY
                    /
1702  4115  TPINH,  CALL TINSET     /SET UP POINTERS
1703  7753
1704  6626          STATIN          /GET ERROR FLAGS
1705  0360          AND (70         /SAVE ERROR, IF ANY
1706  7450          SNA
1707  5313          JMP .+4
1710  3431          DCA I PRPAR
```

```
1711  1071          TAD DUMPFL
1712  3427          DCA I PRBYT
1713  6627          TIN
1714  0364          AND (377          /INPUT FROM UART
1715  3006          DCA 6             /MASK OUT UPPER 4 BITS
1716  1006          TAD 6             /SAVE TEMPORARILY
1717  1426          TAD I PRREM       /UPDATE CHECKSUM
1720  3426          DCA I PRREM
1721  1433          TAD I PRRET       /RETRIEVE RETURN ADDRESS
1722  3007          DCA 7
1723  1006          TAD 6
1724  5407          JMP I 7
                /
                /
1725  3007  PUTDAT, DCA 7             /SAVE DATA TEMPORARILY
1726  6000          SKON
1727  7040          CMA
1730  3055          DCA TEMP3
1731  1070          TAD DTIME         /GET FIELD
1732  7106          CLL RTL           /AND ALIGN FOR HARDWARE
1733  7006          RTL
1734  1125          TAD MODE
1735  6607          SELMD             /SET HARDWARE
                /
1736  1007          TAD 7
1737  3471          DCA I DUMPFL      /PUT DATA AWAY
                /
1740  1125          TAD MODE          /RESTORE HARDWARE
1741  6607          SELMD
1742  2055          ISZ TEMP3
1743  6001          ION
1744  2071          ISZ DUMPFL        /ADVANCE ADDRESS POINTER
1745  7000          NOP
1746  5523          RETURN
                /
1760  0070          PAGE
1761  0257
1762  0253
1763  5257
1764  0377
                /
```

```
/
                /
                /
                /
                *HIPG
                /
                /
7713  1033  DLOOP,  TAD PRRET        /SAVE RETURN ON SECOND LEVEL
7714  3013         DCA XRT1
7715  1520         TAD I STACK
7716  3413         DCA I XRT1
7717  2120         ISZ STACK
                /
7720  4115         CALL UIN
7721  1676
7722  3424         DCA I PRSTCT     /HAVE FIRST CHAR
7723  4115         CALL UIN
7724  1676
7725  7106         CLL RTL
7726  7006         RTL              /SHIFT LEFT 4 BITS
7727  3425         DCA I PRCNT      /STORE IT
7730  1365         TAD (7400        /MASK OUT-
7731  0425         AND I PRCNT      /          LO 8 BITS
7732  1424         TAD I PRSTCT     /COMBINE WITH FIRST WORD
7733  3424         DCA I PRSTCT     /PUT IN PROPER MEMORY LOCATION
7734  1425         TAD I PRCNT      /SET UP 2ND WORD HIGH BITS
7735  7106         CLL RTL
7736  7006         RTL
7737  0365         AND (7400
7740  3425         DCA I PRCNT
7741  4115         CALL UIN         /GET LAST CHAR
7742  1676
7743  1425         TAD I PRCNT
7744  3425         DCA I PRCNT
7745  1033         TAD PRRET        /GET RETURN FROM BUFFER
7746  3013         DCA XRT1
7747  1413         TAD I XRT1
7750  3007         DCA 7
7751  1424         TAD I PRSTCT     /EXIT WITH FIRST WORD IN AC
7752  5407         JMP I 7
                /
                /
                /
                /
                /
                /
                /
                /
                /
7753  1363  TINSET, TAD ((UIN       /SET UP INPUT CALL IF NEEDED
7754  3022         DCA BITOUT
7755  1362         TAD (ITPDAT      /ADRESS OF BUFFER
7756  5761         JMP PRSET1       /SET UP POINTERS
                /
                /
7757  0040  X40,    40
                /
                /
7761  3260         PAGE
7762  0557
```

```
7763  7764
7764  1676
7765  7400
```

```
0000  0200  0177
0200  0355  0377
1200  1365  1367
1400  1555  1566
1600  1747  1757
2000  2161  2165
2200  2370  2370
2400  2565  2566
2600  2772  2773
3000  3164  3164
3200  3363  3364
3400  3571  3570
3600  3763  3762
4000  4167  4166
4200  4365  4365
4400  4566  4566
4600  4764  4764
5000  5161  5161
5200  5356  5355
5400  5572  5571
5600  5770  5767
6000  6171  6170
6200  6367  6367
6400  6566  6566
6600  6767  6767
7000  7170  7171
7200  7366  7365
7400  7600  7577
7600  7760  7760
```

FIELD 0

```
                    /
                    CDF=6201
                    /
                    /PROM BUILD PROGRAM.
                    *600
0600  0000  TPUN,    0
0601  6021          PSF
0602  5201          JMP .-1
0603  6026          PLS
0604  7200          CLA
0605  5600          JMP I TPUN
                    /
0606  0000  CKPUN,   0
0607  3005          DCA 5
0610  1005          TAD 5
0611  1006          TAD 6
0612  3006          DCA 6
0613  1005          TAD 5
0614  4200          JMS TPUN
0615  5606          JMP I CKPUN
                    /
0616  0000  GETLO,   0
0617  4323          JMS TI10
0620  0377          AND (377
0621  4206          JMS CKPUN
0622  5616          JMP I GETLO
                    /
0623  0000  GETHI,   0
0624  4323          JMS TI10
0625  7012          RTR
0626  7012          RTR
0627  0377          AND (377
0630  4206          JMS CKPUN
0631  5623          JMP I GETHI
                    /
0632  0000  GETBT,   0
0633  4323          JMS TI10          /GET LO BITS OF MEMORY INTO
0634  7006          RTL               /HIGH BITS OF ROM
0635  7006          RTL
0636  0376          AND (360
0637  3004          DCA 4
0640  6211          CDF 10
0641  1411          TAD I 11          /GET HIGH BITS OF MEMORY INTO
0642  6201          CDF 00            /LOW BITS OF ROM
0643  7004          RAL
0644  7006          RTL
0645  7006          RTL
0646  0375          AND (17
0647  1004          TAD 4
0650  4206          JMS CKPUN
0651  5632          JMP I GETBT
                    /
                    /
0652  0000  LEADR,   0
0653  1374          TAD (-240
0654  3007          DCA 7
0655  4200          JMS TPUN
0656  2007          ISZ 7
0657  5255          JMP .-2
```

```
0660  7001            IAC
0661  4200            JMS TPUN
0662  4200            JMS TPUN
0663  3006            DCA 6
0664  5652            JMP I LEADR
                /
0665  0000  SETUP,    0
0666  3010            DCA 10
0667  1373            TAD (-400
0670  3007            DCA 7
0671  5665            JMP I SETUP
                /
0672  0000  PUNCK,    0
0673  1006            TAD 6
0674  4200            JMS TPUN
0675  4252            JMS LEADR
0676  5672            JMP I PUNCK
                /
0677  0000  PNLO,     0                   /PUNCH 4 LO BITS
0700  4323            JMS TI10
0701  0375            AND (17
0702  4206            JMS CKPUN
0703  5677            JMP I PNLO
                /
0704  0000  PNMI,     0                   /PUNCH 4 MIDDLE BITS
0705  4323            JMS TI10
0706  7012            RTR
0707  7012            RTR
0710  0375            AND (17
0711  4206            JMS CKPUN
0712  5704            JMP I PNMI
                /
0713  0000  PNHI,     0                   /PUNCH 4 HI BITS
0714  4323            JMS TI10
0715  7006            RTL
0716  7006            RTL
0717  7004            RAL
0720  0375            AND (17
0721  4206            JMS CKPUN
0722  5713            JMP I PNHI
                /
0723  0000  TI10,     0
0724  6211            CDF 10
0725  1410            TAD I 10
0726  6201            CDF 00
0727  5723            JMP I TI10
                /
0730  0000  DI11,     0
0731  6211            CDF 10
0732  3411            DCA I 11
0733  6201            CDF 00
0734  5730            JMP I DI11
                /
0773  7400            PAGE
0774  7540
0775  0017
0776  0360
0777  0377
```

```
/
          *1000
          /THIS ROUTINE PUNCHES ONLY THOSE LOCATIONS USED
          /BY THE ERASABLE ROM.
1000  6026  STRTT,  PLS
1001  1377        TAD (PAGE0-1     /MOVE PG 0
1002  3010        DCA 10
1003  1376        TAD (PAGEZ-1
1004  3011        DCA 11
1005  1375        TAD (PAGE0-TSWF
1006  3007        DCA 7
                  /
1007  4774        JMS TI10
1010  4773        JMS DI11
1011  2007        ISZ 7
1012  5207        JMP .-3
                  /
1013  3023        DCA 23           /DO BIT INVERSION FOR PROG
1014  7240        STA              /STARTING ADDRESS
1015  3010        DCA 10
1016  1010        TAD 10
1017  3011        DCA 11
                  /
1020  4774  BK,   JMS TI10         /GET A WORD
1021  3020        DCA 20
1022  3021        DCA 21           /CLEAR RESULT
1023  1372        TAD (-14         /DO A FULL WORD
1024  3022        DCA 22
                  /
1025  1020  BKLP, TAD 20           /GET WORD
1026  7104        CLL RAL          /GET ANOTHER BIT
1027  3020        DCA 20           /SAVE WHAT'S LEFT
1030  1021        TAD 21           /PUT BIT INTO RESULT
1031  7010        RAR              /BACKWARDS
1032  3021        DCA 21
                  /
1033  2022        ISZ 22
1034  5225        JMP BKLP
                  /
1035  1021        TAD 21           /PUT WORD BACK
1036  7040        CMA              /IN COMPLIMENT FORM
1037  4773        JMS DI11         /FOR 40098'S
                  /
1040  2023        ISZ 23
1041  5220        JMP BK
                  /
                  /•••••••   RECORD NUMBER  1
                  /
                  /
1042  4771        JMS LEADR
1043  1370        TAD (6777        /CHIP 1 LOW 8 7000-7377
1044  4767        JMS LOW
                  /
                  /•••••••   RECORD NUMBER  2
                  /
                  /
1045  1366        TAD (7377        /CHIP 5 HIGH 8 7400-7777
1046  4765        JMS HIGH
                  /
                  /•••••••   RECORD NUMBER  3
```

```
                    /
                    /
1047  1370          TAD (6777       /CHIP 3 HI 4 BITS 1-4 7000-7377
1050  3011          DCA 11          /         LO 4 BITS 5-8 7400-7777
1051  1366          TAD (7377
1052  4764          JMS MID
                    /
                    /******* RECORD NUMBER  4
                    /
                    /
1053  1363          TAD (5777       /CHIP 2 LOW 8 6000-6377
1054  4767          JMS LOW
                    /
                    /******* RECORD NUMBER  5
                    /
                    /
1055  1362          TAD (6377       /CHIP 6 HIGH 8 6400-6777
1056  4765          JMS HIGH
                    /
                    /******* RECORD NUMBER  6
                    /
                    /
1057  1363          TAD (5777       /CHIP 4 HI 4 BITS 1-4 6000-6377
1060  3011          DCA 11          /         LO 4 BITS 5-8 6400-6777
1061  1362          TAD (6377
1062  4764          JMS MID
                    /
                    /******* RECORD NUMBER  7
                    /
                    /
1063  1361          TAD (4777       /CHIP 17 LOW 8 5000-5377
1064  4767          JMS LOW
                    /
                    /******* RECORD NUMBER  8
                    /
                    /
1065  1360          TAD (5377       /CHIP 11 HIGH 8 5400-5777
1066  4765          JMS HIGH
                    /
                    /******* RECORD NUMBER  9
                    /
                    /
1067  1361          TAD (4777       /CHIP 14 HI 4 BITS 1-4 5000-5377
1070  3011          DCA 11          /         LO 4 BITS 5-8 5400-5777
1071  1360          TAD (5377
1072  4764          JMS MID
                    /
                    /******* RECORD NUMBER  10
                    /
                    /
1073  1357          TAD (3777       /CHIP 18 LOW 8 4000-4377
1074  4767          JMS LOW
                    /
                    /******* RECORD NUMBER  11
                    /
                    /
1075  1356          TAD (4377       /CHIP 12 HIGH 8 4400-4777
1076  4765          JMS HIGH
                    /
                    /******* RECORD NUMBER  12
```

```
                /
                /
1077  1357          TAD (3777       /CHIP 15 HI 4 BITS 1-4 4000-4377
1100  3011          DCA 11          /         LO 4 BITS 5-8 4400-4777
1101  1356          TAD (4377
1102  4764          JMS MID
                /
                /******   RECORD NUMBER  13
                /
                /
1103  1355          TAD (2777       /CHIP 25 LOW 8 3000-3377
1104  4767          JMS LOW
                /
                /******   RECORD NUMBER  14
                /
                /
1105  1354          TAD (3377       /CHIP 19 HIGH 8 3400-3777
1106  4765          JMS HIGH
                /
                /******   RECORD NUMBER  15
                /
                /
1107  1355          TAD (2777       /CHIP 22 HI 4 BITS 1-4 3000-3377
1110  3011          DCA 11          /         LO 4 BITS 5-8 3400-3777
1111  1354          TAD (3377
1112  4764          JMS MID
                /
                /******   RECORD NUMBER  16
                /
                /
1113  1353          TAD (1777       /CHIP 26 LOW 8 2000-2377
1114  4767          JMS LOW
                /
                /******   RECORD NUMBER  17
                /
                /
1115  1352          TAD (2377       /CHIP 20 HIGH 8 2400-2777
1116  4765          JMS HIGH
                /
                /******   RECORD NUMBER  18
                /
                /
1117  1353          TAD (1777       /CHIP 23 HI 4 BITS 1-4 2000-2377
1120  3011          DCA 11          /         LO 4 BITS 5-8 2400-2777
1121  1352          TAD (2377
1122  4764          JMS MID
                /
                /******   RECORD NUMBER  19
                /
                /
1123  1376          TAD (777        /CHIP 27 LOW 8 1000-1377
1124  4767          JMS LOW
                /
                /******   RECORD NUMBER  20
                /
                /
1125  1351          TAD (1377       /CHIP 21 HIGH 8 1400-1777
1126  4765          JMS HIGH
                /
                /******   RECORD NUMBER  21
```

```
                /
                /
1127  1376      /        TAD (777        /CHIP 24 HI 4 BITS 1-4 1000-1377
1130  3011      /        DCA I1          /         LO 4 BITS 5-8 1400-1777
1131  1351      /        TAD (1377
1132  4764      /        JMS MID
                /
                /
                /
1133  7402      /        HLT
1134  7402      /        HLT
                /
                /
1151  1377      /        PAGE
1152  2377
1153  1777
1154  3377
1155  2777
1156  4377
1157  3777
1160  5377
1161  4777
1162  6377
1163  5777
1164  1207
1165  1216
1166  7377
1167  1200
1170  6777
1171  0652
1172  7764
1173  0730
1174  0723
1175  7541
1176  0777
1177  0115
                /
1200  0000  LOW,    0
1201  4777      /        JMS SETUP
1202  4776      /        JMS GETLO
1203  2007      /        ISZ 7
1204  5202      /        JMP .-2
1205  4775      /        JMS PUNCK
1206  5600      /        JMP I LOW
                /
1207  0000  MID,    0
1210  4777      /        JMS SETUP
1211  4774      /        JMS GETBT
1212  2007      /        ISZ 7
1213  5211      /        JMP .-2
1214  4775      /        JMS PUNCK
1215  5607      /        JMP I MID
                /
1216  0000  HIGH,   0
1217  4777      /        JMS SETUP
1220  4773      /        JMS GETHI
1221  2007      /        ISZ 7
1222  5220      /        JMP .-2
1223  4775      /        JMS PUNCK
1224  5616      /        JMP I HIGH
                /
1373  0623      /        PAGE
1374  0632
1375  0672
1376  0616
1377  0665
                /
```

```
/
1400  6026  STRT,   PLS
1401  1377          TAD (PAGE0-1    /MOVE PG 0
1402  3010          DCA 10
1403  1376          TAD (PAGEZ-1
1404  3011          DCA 11
1405  1375          TAD (PAGE0-TSWF
1406  3007          DCA 7
                /
1407  4774          JMS TI10
1410  4773          JMS DI11
1411  2007          ISZ 7
1412  5207          JMP .-3
                /
                /
1413  4322          JMS LEDR
                /
1414  1372          TAD (6777       /RECORD #  1  CHIP 3 LO 4 7000-7777
1415  4270          JMS LOWX
                /
1416  1372          TAD (6777       /RECORD #  2  CHIP 10 MIDDLE 4 7000-7777
1417  4277          JMS MIDX
                /
1420  1372          TAD (6777       /RECORD #  3  CHIP 17 HI 4 7000-7777
1421  4306          JMS HIGHX
                /
1422  1371          TAD (5777       /RECORD #  4  CHIP 2 LOW 4 6000-6777
1423  4270          JMS LOWX
                /
1424  1371          TAD (5777       /RECORD #  5  CHIP 9 MIDDLE 4 6000-6777
1425  4277          JMS MIDX
                /
1426  1371          TAD (5777       /RECORD #  6  CHIP 16 HI 4 BITS 6000-6777
1427  4306          JMS HIGHX
                /
1430  1370          TAD (4777       /RECORD #  7  CHIP 4 LO 4 5000-5777
1431  4270          JMS LOWX
                /
1432  1370          TAD (4777       /RECORD #  8  CHIP 11 MIDDLE 4 5000-5777
1433  4277          JMS MIDX
                /
1434  1370          TAD (4777       /RECORD #  9  CHIP 18 HI 4 BITS 5000-5777
1435  4306          JMS HIGHX
                /
1436  1367          TAD (3777       /RECORD # 10  CHIP 32 LO 4 4000-4777
1437  4270          JMS LOWX
                /
1440  1367          TAD (3777       /RECORD # 11  CHIP 28 MIDDLE 4 4000-4777
1441  4277          JMS MIDX
                /
1442  1367          TAD (3777       /RECORD # 12  CHIP 23 HI 4 BITS 4000-4777
1443  4306          JMS HIGHX
                /
1444  1366          TAD (2777       /RECORD # 13  CHIP 33 LO 4 3000-3777
1445  4270          JMS LOWX
                /
1446  1366          TAD (2777       /RECORD # 14  CHIP 29 MIDDLE 4 3000-3777
1447  4277          JMS MIDX
                /
1450  1366          TAD (2777       /RECORD # 15  CHIP 24 HI 4 BITS 3000-3777
```

```
1451  4306          JMS HIGHX
                    /
1452  1365          TAD (1777       /RECORD # 16  CHIP 34 LO 4 2000-2777
1453  4270          JMS LOWX
                    /
1454  1365          TAD (1777       /RECORD # 17  CHIP 30 MIDDLE 4 2000-2777
1455  4277          JMS MIDX
                    /
1456  1365          TAD (1777       /RECORD # 18  CHIP 25 HI 4 BITS 2000-2777
1457  4306          JMS HIGHX
                    /
1460  1376          TAD (777        /RECORD # 19  CHIP  5 LO 4 1000-1777
1461  4270          JMS LOWX
                    /
1462  1376          TAD (777        /RECORD # 20  CHIP  6 MIDDLE 4 1000-1777
1463  4277          JMS MIDX
                    /
1464  1376          TAD (777        /RECORD # 21  CHIP  7 HI 4 BITS 1000-1777
1465  4306          JMS HIGHX
                    /
                    /
                    /
1466  7402          HLT
1467  7402          HLT
                    /
1470  0000  LOWX,   0
1471  4315          JMS SETSUP
1472  4764          JMS PNLO
1473  2007          ISZ 7
1474  5272          JMP .-2
1475  4322          JMS LEDR
1476  5670          JMP I LOWX
                    /
1477  0000  MIDX,   0
1500  4315          JMS SETSUP
1501  4763          JMS PNMI
1502  2007          ISZ 7
1503  5301          JMP .-2
1504  4322          JMS LEDR
1505  5677          JMP I MIDX
                    /
1506  0000  HIGHX,  0
1507  4315          JMS SETSUP
1510  4762          JMS PNHI
1511  2007          ISZ 7
1512  5310          JMP .-2
1513  4322          JMS LEDR
1514  5706          JMP I HIGHX
                    /
1515  0000  SETSUP, 0
1516  3010          DCA 10
1517  1361          TAD (-1000
1520  3007          DCA 7
1521  5715          JMP I SETSUP
                    /
1522  0000  LEDR,   0
1523  1360          TAD (-160
1524  3007          DCA 7
1525  4757          JMS TPUN
1526  2007          ISZ 7
```

```
1527  5325          JMP .-2
1530  7240          STA
1531  4757          JMS TPUN
1532  3006          DCA 6
1533  5722          JMP I LEDR

1557  0600          PAGE
1560  7620
1561  7000
1562  0713
1563  0704
1564  0677
1565  1777
1566  2777
1567  3777
1570  4777
1571  5777
1572  6777
1573  0730
1574  0723
1575  7541
1576  0777
1577  0115

0600  0735  0772
1000  1135  1150
1200  1225  1372
1400  1534  1556
```

Symbol	Value
ACCLR	7703
ACTIME	0400
ADDCLK	5010
ADDE	2674
ADDEM	3101
ADDES1	3051
ADDES2	3054
ADDES3	3065
ADDE1	3006
ADDE2	3016
ADDE6	3077
ADDE7	3107
ADDE8	3116
ADDT	5036
ADDTM	5015
ADIO	5000
APRINT	4346
AUTOCA	1310
AUTOCM	1300
AUTOCT	0074
AUTOC1	1335
AUTOC2	1340
AUTOC3	1421
AUTODN	1600
AUTOF	1332
AUTOFL	0075
AUTO1	1431
AUTO1A	1454
AUTO1B	1433
AUTO2	1426
AUTO3	1461
AUTO3A	1505
AUTO3B	1520
AUTO3C	1532
AUTO3D	1550
AUTO4	1612
AUTO4A	1615
AUTO5	1637
BCDBIN	6625
BCDCON	6702
BITOUT	0022
BK	1029
BKLP	1025
BPRINT	4343
BUFBG	0673
BUFL	1252
CDF	6201
CH1	0057
CH1E	0364
CH1LD	0407
CH1LN	0431
CH2	0060
CH2E	0372
CH2LD	0415
CH2LN	0437
CKFUN	0606
CLDAY	0145
CLHR	0144
CLMIN	0143
CLSEC	0142
DATOUT	0032
DATPNT	0103
DBCON	6763
DBCONA	7000
DBEN	0303
DBNCB	0306
DBNCC	0313
DBNCE	0274
DBNCK	0327
DCH	7215
DCLK	7207
DELAY1	7265
DELAY2	7136
DFLO	0064
DHTAB	7200
DISCH1	7341
DISCH2	7343
DISEL	0133
DISPCH	0132
DISPK	7364
DISSET	0330
DISTAB	7334
DISTIM	7362
DISTST	3676
DISUB	7346
DISUM	7360
DISUP	7324
DIVD	6200
DI11	0730
DLFLO	0066
DLOOP	7713
DLX16	0067
DMASK	0167
DMPX	0151
DMP1	7045
DMP1A	7143
DMP1B	7074
DMP2	7113
DMP3	7146
DPMP	7011
DTIME	0070
DUMPFL	0071
DUMPLK	7261
DUMPT	0072
DUMPT1	0073
DX16	0065
EMPTY	0041
ERROR1	3325
ERROR2	3324
ERROR3	3323
ERROR4	3322
ERROR5	3321
ERRO	6622
FORMAT	0127
FUNLOC	7400
FUN0	5256
FUN0A	5271
FUN0B	5261
FUN1	5600
FUN1A	5604
FUN1B	5606
FUN10	6427
FUN11	6425
FUN12	6446
FUN13	6507
FUN14	6317
FUN2	5615
FUN2A	5631
FUN2AA	5634
FUN2AB	5636
FUN3	5714
FUN4	6035
FUN4A	6037
FUN5	6167
FUN6	6217
FUN7	6241
FUN8	6256
FUN9	6263
GETBT	0632
GETDAT	7124
GETHI	0623
GETLO	0616
GOSUB	4115
GRPNO	0144
HIFUN	7440
HIGH	1216
HIGHX	1506
HIPG	7713
HSW4	6561
HSW5	6600
HSW5X	6604
HTSW4	0263
HTSW5	0267
ICHK	1656
IDMP	0150
IMT	2640
IMTL	2647
INDIR	6671
INFLO	0036
INT1	0010
INT2	0011
INT3	0012
IRCTN	0123
ITPDAT	0557
ITPKIL	1661
IX	2114
IX1	2123
KCAL1	0134
KCAL2	0135
KD10	0156
KD100	0174
KD1000	0171
KD144	0162
KD17	0157
KD20	0160
KD200	0173
KD2000	0170
KD320	0163
KD40	0175
KD400	0172
KD4000	0167
KD7000	0164
KD77	0161
KD7700	0165
KD7760	0166
KM12	0155
LBEGIN	2100
LDMP1	7230
LDMP2	7250
LDMP3	7312
LDNBOT	0572
LDNIN	0044
LDNLEN	0043
LDNMT	0047
LDNOT	0046
LDNPUT	4751
LDNST	0042
LDNSTR	0033
LDNTOP	0672
LDN16	0045
LDN17	0016
LLADER	4545
LEADR	0652
LFDR	1522
LEN	0035
LF	4621
LF1	4634
LF2	4643
LF2A	4647
LF2B	4651
LMASK	7157
LOGCON	2657
LOGTAB	7500
LOGTBL	0154
LOPG	2100
LOW	1200
LOWX	1470
LTAB	7220
LTAB1	7222
MEMAD	0121
MEMLD	0122
MEMFLG	0124
MESTOP	0056
MID	1207
MIDX	1477
MODE	0125
MODJMP	5412
MODST	4512
MPLS	6742
MPLT	6756
MPLY	6736
NAUTOC	1636
NUMONE	6226
OCTBCD	6363
OCTB1	6401
OCTB2	6404
OPK	0406
OUTBIT	4523
OUTBT1	4532
OUTFLO	0040
OUT3B	4460
PAGEZ	[illegible]
PAGE0	0116
PERCON	6761
PK	0061
PNHI	0713
PNLO	0677
PNMI	0704
PRBUF	0470
PRBYT	0027
PRBYTC	0030
PRCNT	0025
PRDAT	0530
PRFB	0104
PRHTB	4254
PRINT	4351
PRINT1	4354
PRINT2	4362
PRLEN	0012
PRLTB	4263
PRNIT	4206
PRNIT1	4212
PRNIT2	4251
PRNTER	0102
PRNTST	4406
PRN1A	4415
PRN1B	4422
PRN2A	4403
PRN3	4433
PRN3A	4441
PRN3C	4476
PRN3D	4520
PROC1	0470
PROC2	0474
PROC3	0500
PROC4	0504
PROC5	0510
PROC6	0514
PROC7	0520
PROCR	0524
PRPAR	0031
PRPNT	0100
PRREM	0026
PRSET	0033
PRSIT	3255
PRSET1	3260
PRSET2	3265
PRSHFT	0024
PRSVR	0543
PRTAB	4741
PUNCK	0672
PUTDAT	1725
RADDE	3135
RADDE0	3125
RADDE1	3147
RADDE2	3200
RADDE3	3205
RADDE4	3243
RADDE5	3246
RADTB1	2600
RADTB2	2610
RDYBUF	0131
RECON	0130
RELEV1	0136
RELEV2	0137
RETN	6665
RETURN	5573
RET2	4614
SAUTOC	4756
SAVAC	0050
SAVLK	0051
SAVMQ	0052
SETMD	5506
SETNUP	1515
SETUP	0665
SFN14A	6350
SFUN0	5267
SFUN10	6426
SFUN11	6424
SFUN12	6460
SFUN13	6523
SFUN14	6337
SFUN15	6361
SFUN2	5622
SFUN3	5725
SFUN4	6025
SFUN5	6473
SFUN6	6224
SFUN7	6272
SFUN8	6305
SFUN9	6312
SMPTG	0063
STACK	0120
STAK	0453
START	7600
STK	0467
STORE	0034
STRT	1400
STRIT	1000
STRT1	7612
STRT2	7647
STRT3	7671
ST3MER	3427
ST3M0	3433
ST3M1	3433
ST3M15	3470
ST3M2	3442
ST3M3	3450
ST3M4	3454
ST3M5	3460
ST3M6	3460
ST3M7	3460
ST3M8	3470
ST3M9	3470
ST4ERR	3427
ST4M0	3632
ST4M1	3636
ST4M15	3662
ST4M2	3642
ST4M3	3646
ST4M4	3652
ST4M5	3656
ST4M6	3656
ST4M7	3656
ST4M8	3662
ST4M9	3662
SUB	0115
SUBA	6652
SUBN1	6662
SUB1	0114
SUB10	5002
SVMD	0126
SVSUB	0053
SVSUB1	0054
SW[illegible]	4721
SWSET	0177
SW0	2003
SW0A	0152
SW0A0	2247
SW0A1	2262
SW0A2	2272
SW0B	2277
SW0B2	2516
SW0B2A	2530
SW0B2B	2562
SW0D	2337
SW0D1	2345
SW0D2	2411
SW0D3	2426
SW0E	2434
SW0F	2441
SW0F1	2452
SW0FDN	2465
SW0FNA	2475
SW0P	2200
SW0PP	2400
SW1	3250
SW3	3337
SW3A	3502
SW3AA	3500
SW3B	3505
SW3C	3510
SW3CM	3346
SW3CMP	3400
SW3D	3522
SW3E	3524
SW3ER	3570
SW3FT	3556
SW3F1	3544
SW4	3600
SW4A	3666
SW4B	3670
SW4C	3672
SWS	3704
SWSC	4106
SWSC1	4110
SWSC2	4121
SWSD	4133
SWSD2	4163
SWSE	4273
SWSE1	[illegible]
SWSE4	4274
SWSH	[illegible]
SWSHP	[illegible]
SWSK	[illegible]
SWSL	4123
SW7	[illegible]
SW7A	4664
SW7B	4675
SW7C	4703
SW7D	[illegible]
SW8	5054
SW8A	[illegible]
SW8C	5145
SW8L	[illegible]
SWST	[illegible]
SWS10	[illegible]
SWS11	[illegible]
SYNCMJ	[illegible]
SYNMJ	[illegible]
TABFER	[illegible]
TABMJ	[illegible]
TABM0	5435
TABM1	5435
TABM15	[illegible]
TABM2	[illegible]
TABM3	[illegible]
TABM4	[illegible]
TABM5	[illegible]
TABM6	5453
TABM7	[illegible]
TABM8	[illegible]
TABM9	[illegible]
TABOUT	[illegible]
TBIN	[illegible]
TBINA	[illegible]
[illegible]	[illegible]
TIMEN	[illegible]
TIME1	[illegible]
TIMEIN	[illegible]
[illegible]	[illegible]
TIME2	[illegible]
TIME3	[illegible]
TIN	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
TIMI	[illegible]
TINB	[illegible]
TINSET	[illegible]
TIO	[illegible]
TMD	0423
TMIN	[illegible]
TMN	0445
TOUTB	[illegible]
TPIN	0253
TPINH	1702
TPINN	0257
TPOUT	6247

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```
TPOUTX 0253
TPSET  5247
TPSV   0023
TPUN   0600
TRBIT  5216
TRBIT1 5225
TRBLK  0105
TRDAT  0544
TR3    5235
TSEC   0140
TSWF   0355
TSW0   0201
TSW0A  0341
TSW0X  0204
TSW1   0204
TSW1A  0340
TSW1X  0210
TSW3   0210
TSW3B  0337
TSW3F  0076
TSW3X  0215
TSW4   0215
TSW4B  0336
TSW4F  0077
TSW4X  0222
TSW5   0222
TSW7   0226
TSW7B  0335
TSW8   0232
TSW8B  0333
TSW8C  0334
TSW8F  0101
TSW8X  0241
TTSET  5474
UARTI  2137
UARTI1 1642
UIN    1676
VLOPG  1642
WFUN0  5253
WFUN1  5305
WFUN1A 5347
WFUN1C 5464
WFUN1D 5467
WFUN1E 5521
WFUN1F 5527
WFUN1G 5537
WFUN1H 5544
WFUN12 6437
WFUN13 6500
WFUN14 6315
WFUN15 6355
WFUN2  5612
WFUN2A 5655
WFUN3  5675
WFUN3A 5710
WFUN4  6032
WFUN5  6042
WFUN6  6214
WFUN7  6234
WFUN7A 6245
WFUN7B 6243
WFUN8  6252
WFUN9  6260
WSFN0  5264
WSFN12 6451
WSFN13 6514
WSFN14 6321
WSFN15 6356
WSFN2  5617
WSFN3  5722
WSFN4  6022
WSFN5  6462
WSFN5A 6530
WSFN5B 6545
WSFN6  6221
WSFN7  6265
WSFN8  6274
WSFN9  6307
XFUN3  5740
XFUN3A 6002
XFUN3B 6007
XRT1   0013
XRT2   0014
XRT3   0015
XR16   0037
XR17   0017
XSW3A  3273
XSW3B  3305
X40    7757
```

APPENDIX C:

LISTING OF PROGRAM USED TO WRITE UV
ERASABLE, READ ONLY MEMORIES

```
      INTEGER TNAME(12),FNAME,TEMP(12)
      INTEGER SHFTL,SHFTR

      INTEGER RBUF(256),EBUF(256),DBUF(256),WBUF(256),NAME(12)

      FNAME=.TRUE.
      DO 11 I=1,256
      DBUF(I)=0
      EBUF(I)=0
      WBUF(I)=0
      RBUF(I)=0
11    CONTINUE

      TYPE 'TO BACK UP OR STOP GIVE A ZERO OR A NULL ANSWER'

10    WRITE(10,997)
997   FORMAT('0 ACTION:',/,
     1       4X,'0  STOP',/,
     1       4X,'1  READ FROM DATA FILE',/,
     1       4X,'2  LIST DATA BUFFERS',/,
     1       4X,'3  WRITE TO PROM FROM DATA BUFFER',/,
     1       4X,'4  READ FROM PROM INTO DATA BUFFER',/,
     1       4X,'5  WRITE A SINGLE BYTE',/,
     1       4X,'6  READ A SINGLE BYTE',/,
     1       4X,'7  READ BEFORE WRITE',/,
     1       4X,'.....=',Z)
      READ(11) I
      IF (I.EQ.0.OR.I.EQ.-1) STOP
      IF (I.LT.0.OR.I.GT.7) GOTO 10

      GOTO (1000,2000,3000,4000,6000,5000,7000) I

1000  WRITE(10,999)
999   FORMAT('0 DATA FILE'S NAME=',Z)
      READ(11,998) NAME(1)
998   FORMAT(S20)
      IF (FNAME.AND.LENGTH(NAME).EQ.0) GOTO 10
      IF (LENGTH(NAME).GT.0) GOTO 1010

      DO 1011 I=1,12
1011  NAME(I)=TNAME(I)

1010  DO 1012 I=1,12
1012  TNAME(I)=NAME(I)
      FNAME=.FALSE.

      WRITE(10,979) NAME(1)
979   FORMAT(' FILE NAME READ =',S20)

      ACCEPT 'RECORD #=',IR
      IF (IR.LE.0) GOTO 10

      CALL FOPEN(1,NAME)

      IF (IR.EQ.1) GOTO 1300
      IRM=IR-1
      DO 1250 I=1,IRM
```

```
      CALL INDATA(WBUF)
1250  CONTINUE

1300  CALL INDATA(WBUF)
      CALL FCLOS(1)
      GOTO 10


C     LIST DATA BUFFERS
2000  WRITE(10,996)
996   FORMAT('0 WHICH DATA BUFFER:',/,
     1        4X,'1  WRITE',/,
     1        4X,'2  READ',/,
     1        4X,'3  DIFFERENCE',/,
     1        4X,'4  ERROR',/,
     1        4X,'     .............=',Z)
      READ(11) I

      IF (I.EQ.0.OR.I.EQ.-1) GOTO 10
      IF (I.LT.0.OR.I.GT.4) GOTO 2000
      GOTO (2100,2200,2300,2400) I

C     WRITE BUFFER
2100  WRITE(10,995)
995   FORMAT('0WRITE BUFFER')

      WRITE(10,994) WBUF
994   FORMAT( (2(2X,4O14,1X,4O14,1X)) )
      GOTO 2000

C     READ BUFFER
2200  WRITE(10,993)
993   FORMAT('0READ BUFFER')
      WRITE(10,994) RBUF
      GOTO 2000

C     DIFFERENCE BETWEEN READ BUFFER AND WRITE BUFFER
2300  DO 2305 I=1,256
      J=RBUF(I)
      K=WBUF(I)
      DBUF(I)=((J.AND. .NOT.K) .OR. (.NOT.J .AND.K)).AND.377K
2305  CONTINUE

      WRITE(10,992)
992   FORMAT('0DIFFERENCE BUFFER')
      WRITE(10,994) DBUF
      GOTO 2000

C     ERROR BUFFER
2400  WRITE(10,991)
991   FORMAT('0ERROR BUFFER')
      WRITE(10,994) EBUF
      GOTO 2000


C     WRITE TO PROM
3000  TYPE
```

```
        TYPE 'WRITING PROM'
        TYPE 'CHECK PROGRAMMER... P1 ON    P2 ON     ... STRIKE NEWLINE'
        READ(11,998) TEMP(1)
        IF (LENGTH(TEMP).NE.0) GOTO 10
        DO 3005 I=1,256
        EBUF(I)=0
3005    CONTINUE

        DO 3110 I=1,256
3200    CALL PIO(I-1,WBUF(I),RBUF(I))

        IF (RBUF(I).EQ.WBUF(I)) GOTO 3100
        EBUF(I)=EBUF(I)+1
        IM1=I-1
        WRITE(10,976) IM1,EBUF(I)
976     FORMAT(' <32> LOCATION',O14,'  IN ERROR',5X,O13,' RETRIES')
        IF (EBUF(I).GT.10) GOTO 3100
        GOTO 3200
3100    IF (I.EQ.1) GOTO 3110
        IF (EBUF(I-1).GT.0) TYPE
3110    CONTINUE

        GOTO 10


C       READ PROM
4000    TYPE
        TYPE 'READING PROM'
        TYPE 'CHECK PROGRAMMER... P1 ON   P2 OFF   ... STRIKE NEWLINE'
        READ(11,998) TEMP(1)
        IF (LENGTH(TEMP).NE.0) GOTO 10

        DO 4100 I=1,256
        CALL PIO(I-1,0,RBUF(I))

4100    CONTINUE
        GOTO 10


C       READ A SINGLE BYTE
5000    ACCEPT 'WHICH BYTE TO BE READ(OCTAL)=',I
        IF (I.EQ.-1) GOTO 10
        IF (I.LT.0.OR.I.GT.377) GOTO 5000
        I1=I/100
        I2=(I-100*I1)/10
        I3=(I-100*I1-I2*10)
        I=I1*64+I2*8+I3

        WRITE(10,977) I
977     FORMAT(' BYTE NUMBER IS=',O16)
        CALL PIO(I,0,K)
        KK=(.NOT.(IBITR(K,8))).AND.377K
        WRITE(10,990) K,K,KK
990     FORMAT('0DATA=',I5,O16,'K      COMPLIMENTED REVERSED=',O15)
```

```
      GOTO 5000


 6000 ACCEPT 'WHICH BYTE TO BE LOADED(OCTAL) =',I
      IF (I.EQ.-1) GOTO 10
      IF (I.LT.0.OR.I.GT.377) GOTO 6000
      I1=I/100
      I2=(I-100*I1)/10
      I3=(I-100*I1-I2*10)
      I=I1*64+I2*8+I3
      WRITE(10,978) I
  978 FORMAT(' BYTE TO BE WRITTEN IS ',O18)
      WRITE(10,989)
  989 FORMAT(' DATA (OCTAL) =',Z)
      READ(11) K
      IF (K.LT.0) GOTO 6000
      I1=K/100
      I2=(K-100*I1)/10
      I3=(K-100*I1-I2*10)
      K=I1*64+I2*8+I3

      J=0
 6005 IF (J.GT.10) GOTO 6010
      CALL PIO(I,K,L)
      IF (L.EQ.K) GOTO 6010
      J=J+1
      GOTO 6005

 6010 WRITE(10,988) I,L,J
  988 FORMAT(' WORD',O14,'  WAS LOADED WITH',O14,'   WITH',O14,' RETRIES')
      GOTO 6000

C     READING BEFORE WRITING
 7000 TYPE
      TYPE 'READING BEFORE WRITING'
      TYPE 'CHECK PROGRAMMER..  P1 ON    P2 ON    ..... STRIKE NEWLINE'
      READ(11,998) TEMP(1)
      IF (LENGTH(TEMP).NE.0) GOTO 10
      DO 7005 I=1,256
      EBUF(I)=0
 7005 CONTINUE

      DO 7010 I=1,256
      IM1=I-1
      CALL PIO(IM1,0,RBUF(I))
      IF (RBUF(I).EQ.0) GOTO 7200
      WRITE(10,965) IM1,RBUF(I)
  965 FORMAT('0NON-ZERO DATA IN EMPTY PROM',/,
     1 ' LOCATION',O14,'  HAD',O14,' OCTAL',/,
     1 ' SHOULD I CONTINUE (Y/N) ?',Z)
      READ (11,998) TEMP(1)
      IF (ISEARCH(TEMP,'Y').LE.0) GOTO 10

 7200 CALL PIO(IM1,WBUF(I),RBUF(I))
      IF (RBUF(I).EQ.WBUF(I)) GOTO 7100
```

```
            EBUF(I)=EBUF(I)+1
            WRITE(10,976) IM1,EBUF(I)
            IF (EBUF(I).GT.10) GOTO 3100
            GOTO 7200
 7100       IF (1.EQ.1) GOTO 7010
            IF (EBUF(IM1).GT.0) TYPE
 7010       CONTINUE
            GOTO 10

            END


            SUBROUTINE INDATA(WBUF)
            INTEGER SHFTL
            INTEGER WBUF(256)

 10         READ BINARY(1) I
            IF (I.EQ.0) GOTO 10

 C          SET COUNTER
            J=1

            I2=I.AND.377K
            I=SHFTL(I,8)
            I1=I.AND.377K

            IF (I2.EQ.0) GOTO 15
            READ BINARY(1) I
            I2=I.AND.377K
            J=2
            WBUF(1)=I2

 15         READ BINARY(1) I
            I2=I.AND.377K
            I= SHFTL(I,8)
            I1=I.AND.377K

            IF (J.GT.256) GOTO 30

            WBUF(J)=I1
            J=J+1

            IF (J.GT.256) GOTO 40
            WBUF(J)=I2
            J=J+1
            GOTO 15

 30         ICRC=I1
            GOTO 50

 40         ICRC=I2

 50         J=0

            DO 60 I=1,256
            J=(J+WBUF(I)).AND.377K
```

```
60      CONTINUE

        IF (J.NE.ICRC) TYPE 'CHECKSUM ERROR',ICRC,J
        RETURN
        END


        SUBROUTINE PIO(IA,ID,IO)
        KA=IA.AND.377K
        KD=ID.AND.377K

A       INTDS
A       LDA     0,KA-200,3
A       DOAS    0,51
A       SKPDN   51
A       JMP     .-1
A       LDA     0,KD-200,3
A       DOAS    0,51
A       SKPDN   51
A       JMP     .-1
A       NIOS    50
A       SKPDN   50
A       JMP     .-1
A       DIA     0,50
A       STA     0,KO-200,3
A       DIAC    0,50
A       DOAC    0,51
A       INTEN
A

        IO=KO.AND.377K
        RETURN
        END
```

APPENDIX D:

LISTING OF COMPUTER PROGRAM USED FOR
TESTING MEMORY IN THE NOISE MONITOR

```
                    /
                    /THIS IS A MEMORY TEST ROUTINE WHICH CAN BE RUN
                    /ANYWHERE IN BANK ZERO STARTING AT A PAGE BOUNDARY.
                    /
                    /TO USE, USE THE MONITOR TO LOAD THE FIRST THREE ADDRESSES;
                    /
                    /       XFIELD  THIS SPECIFIES THE MEMORY BANK (0,1,2, OR 3)
                    /       FIRST   FIRST LOCATION TO BE TESTED
                    /       LAST    LAST LOCATION TO BE TESTED
                    /
                    /
                    /SINCE THE ADDRESS POINTER IS INCREMENTED AFTER EACH LOAD,
                    /THE POINTER IS POINTING AT THE START ADDRESS WHEN THE THREE
                    /NUMBERS HAVE BEEN ENTERED.  TO BEGIN EXECUTION, TURN THE
                    /FUNCTION SWITCH TO THE GOTO POSITION AND, WHILE HOLDING
                    /DOWN THE EXECUTE SWITCH, DEPRESS THE DISPLAY SWITCH.
                    /
                    *1000
1000  0000  XFIELD, 0
1001  0000  FIRST,  0
1002  0000  LAST,   0
1003  6002  START,  IOF
1004  3357          DCA PASS          /CLEAR PASS COUNTER
1005  4206          JMS IOUT          /SET UP PAGE INDEPENDENT REFERENCE
1006  0000  IOUT,   0                 /THIS IS POINTER TO OCTOUT
1007  1354          TAD IADDR
1010  1206          TAD IOUT          /THIS COMPUTES THE ABSOLUTE ADDRESS
1011  3206          DCA IOUT
                    /
1012  1200          TAD XFIELD        /CHANGE USER REQUEST TO HARDWARE FORMAT
1013  7106          CLL RTL
1014  7006          RTL
1015  1355          TAD D4000         /KEEP DISPLAY ON
1016  6607          SELMD             /THIS APPLIES TO ENTIRE TEST
                    /
1017  1202          TAD LAST          /GET NUMBER OF LOCATIONS TO CLEAR
1020  7040          CMA
1021  1201          TAD FIRST
1022  3357          DCA PASS
1023  1201          TAD FIRST         /SET ADDRESS POINTER
1024  3360          DCA ADDR
1025  3760          DCA I ADDR        /ZERO IN USER'S FIELD
1026  2360          ISZ ADDR
1027  2357          ISZ PASS
1030  5225          JMP .-3
                    /
1031  3362  ADR1,   DCA DATA          /INITIALIZE DATA
1032  3364          DCA DATA2
1033  7240          STA
1034  3361          DCA FWD           /SET FORWARD SCAN
1035  7240          STA
1036  3365          DCA BIT           /SET COUNTER
1037  1201          TAD FIRST
1040  3360          DCA ADDR          /SET UP POINTER
1041  4276          JMS UPDAT
                    /
1042  1760  RDLP1,  TAD I ADDR        /GET DATA FROM USER'S FIELD
1043  7041          CIA
1044  1363          TAD DATA1         /COMPARE WITH THE EXPECTED VALUE
1045  7440          SZA
```

```
1046  4367          JMS ERROR       /MUST HAVE CHANGED SOMEWHERE
1047  1364          TAD DATA2       /UPDATE CONTENTS
1050  3760          DCA I ADDR
1051  1760          TAD I ADDR      /DID DATA GET THERE?
1052  7041          CIA
1053  1364          TAD DATA2
1054  7440          SZA
1055  4367          JMS ERROR
1056  1361          TAD FWD         /DATA OK.  DIRECTION?
1057  7650          SNA CLA
1060  5326          JMP BKLP
                    /
1061  1360          TAD ADDR        /LAST ADDRESS?
1062  7160          STL CMA
1063  1202          TAD LAST
1064  7630          SZL CLA
1065  5271          JMP LP1
1066  2360          ISZ ADDR        /NOT YET
1067  7000          NOP
1070  5242          JMP RDLP1
                    /
1071  4276  LP1,     JMS UPDAT       /UPDATE DATA WORDS
1072  5320          JMP LP2
1073  1201  LP1A,    TAD FIRST       /AND RESET ADDRESS
1074  3360          DCA ADDR
1075  5242          JMP RDLP1
                    /
1076  0000  UPDAT,   0
1077  2365          ISZ BIT         /ADVANCE BIT COUNTER
1100  5307          JMP UPDAT1
1101  7330          K4000           /COMPLEMENT DATA BIT
1102  1362          TAD DATA
1103  3362          DCA DATA
1104  1356          TAD KM14        /RESET BIT COUNTER
1105  3365          DCA BIT
1106  7410          SKP             /TAKE NORMAL RETURN
1107  2276  UPDAT1, ISZ UPDAT      /THIS IS FOR SKIP RETURN
1110  1362          TAD DATA        /GET VALUE OF NEXT BIT
1111  7104          CLL RAL
1112  1364          TAD DATA2
1113  3363          DCA DATA1       /UPDATE FIRST WORD
1114  1364          TAD DATA2
1115  7010          RAR             /UPDATE SECOND WORD
1116  3364          DCA DATA2
1117  5676          JMP I UPDAT
                    /
1120  1362  LP2,     TAD DATA        /DONE WITH BITS.  WHICH DATA STATE
1121  7710          SPA CLA         /WERE WE IN?
1122  5324          JMP LP3
1123  5273          JMP LP1A
                    /
1124  3361  LP3,     DCA FWD         /SET BACKWARD SEQUENCE
1125  5242          JMP RDLP1       /STARTING AT LAST
                    /
1126  1201  BKLP,    TAD FIRST
1127  7160          STL CMA         /ARE WE BACK AT THE BEGINNING?
1130  1360          TAD ADDR
1131  7630          SZL CLA
1132  5337          JMP BKLP2
1133  7240          STA             /NO.  DECREMENT ADDRESS
```

```
1134  1360                TAD ADDR
1135  3360                DCA ADDR
1136  5242                JMP RDLP1
                  /
1137  4276  BKLP2,   JMS UPDAT      /UPDATE DATA WORDS
1140  5344          JMP BKLP3
1141  1202  BKLP2A, TAD LAST       /YES.  RESET ADDRESS
1142  3360          DCA ADDR
1143  5242          JMP RDLP1
                  /
1144  1362  BKLP3,  TAD DATA       /GET DATA TYPE
1145  7710          SPA CLA
1146  5350          JMP BKLP4
1147  5341          JMP BKLP2A     /FIX ADDRESS
                  /
1150  2357  BKLP4,  ISZ PASS       /MADE IT THROUGH
1151  1357          TAD PASS       /TELL THE WORLD
1152  4606          JMS I IOUT     /IN OCTAL, OF COURSE
1153  5231          JMP ADR1       /TRY AGAIN
                  /
1154  0172  IADDR,  OCTOUT-IOUT
1155  4000  D4000,  4000
1156  7764  KM14,   -14
1157  0000  PASS,   0
1160  0000  ADDR,   0
1161  0000  FWD,    0
1162  0000  DATA,   0
1163  0000  DATA1,  0
1164  0000  DATA2,  0
1165  0000  BIT,    0
1166  0000  ERDAT,  0
1167  0000  ERROR,  0
1170  3366          DCA ERDAT      /SAVE BAD BITS
1171  7330          K4000
1172  6607          SELMD
1173  5774          JMP I .+1      /RETURN TO MONITOR FIELD
1174  7777          7777           /DO A RESTART
                  /
                  /
                  /       PAGE
                  /
1200  0000  OCTOUT, 0
1201  4216          JMS OUT
1202  6604          LODIS
1203  1233          TAD OUTK2
1204  7002          BSW
1205  4216          JMS OUT
1206  6605          HIDIS
1207  6622          FUNLO
1210  7700          SMA CLA        /DO WE WANT OUT?
1211  5600          JMP I OCTOUT
1212  7330          K4000
1213  6607          SELMD
1214  5615          JMP I .+1
1215  7777  KMON,   7777           /YES.  DO A RESTART
                  /
1216  0000  OUT,    0
1217  3233          DCA OUTK2
1220  1233          TAD OUTK2
1221  0230          AND K70
```

```
1222  7104          CLL RAL          /ALIGN HIGH DIGIT
1223  3232          DCA OUTK1
1224  1233          TAD OUTK2
1225  0231          AND K7
1226  1232          TAD OUTK1
1227  5616          JMP I OUT
                    /
1230  0070  K70,    70
1231  0007  K7,     7
1232  0000  OUTK1,  0
1233  0000  OUTK2,  0
                    /
                    /TRANSFER CODE ROUTINE.  USE MONITOR TO LOAD ADDRESS OF
                    /DESTINATION IN OUTK2.  THEN SWITCH TO GOTO SINCE POINTER
                    /IS NOW AT START OF TRANSFER ROUTINE.
                    /
1234  1233  TRAN,    TAD OUTK2
1235  3200          DCA OCTOUT
1236  1254          TAD XFIRST
1237  3216          DCA OUT
1240  1255          TAD KLEN
1241  3232          DCA OUTK1
1242  1616  TRAN1,  TAD I OUT
1243  3600          DCA I OCTOUT
1244  2200          ISZ OCTOUT
1245  2216          ISZ OUT
1246  2232          ISZ OUTK1
1247  5242          JMP TRAN1
1250  1256          TAD JLEN
1251  1233          TAD OUTK2
1252  3200          DCA OCTOUT       /HAVE NEW STARTING ADDRESS
1253  5600          JMP I OCTOUT     /BEGIN TESTING IMMEDIATELY
1254  1000  XFIRST, XFIELD
1255  7521  KLEN,   XFIELD-END
1256  0003  JLEN,   START-XFIELD
              END,
                    /


1000  1175  1177
1200  1257  1377
```

ADDR	1160
ADR1	1031
BIT	1165
BKLP	1126
BKLP2	1137
BKLP2A	1141
BKLP3	1144
BKLP4	1150
DATA	1162
DATA1	1163
DATA2	1164
D4000	1155
END	1257
ERDAT	1166
ERROR	1167
FIRST	1001
FWD	1161
IADDR	1154
IOUT	1006
JLEN	1256
KLEN	1255
KMON	1215
KM14	1156
K7	1231
K70	1230
LAST	1002
LP1	1071
LP1A	1073
LP2	1120
LP3	1124
OCTOUT	1200
OUT	1216
OUTK1	1232
OUTK2	1233
PASS	1157
RDLP1	1042
START	1003
TRAN	1234
TRAN1	1242
UPDAT	1076
UPDAT1	1107
XFIELD	1000
XFIRST	1254

APPENDIX E:

LISTING OF COMPUTER PROGRAM USED TO CHECK MAGNETIC TAPES IN THE FIELD FOR RELIABILITY

Four Functions in Hidden Positions

BLACK	WHITE
EXTERNAL DATA BLOCK	LDN
STARTING LOC	
MEMORY ADDRESS (Both read and write)	MEMORY CONTENTS (both read and write); execute must be pushed first and released last when writing into memory; the monitor increments the address after read or write
Read is load cassette program; write is begin execution of program in field 0 at address currently in memory address register; execute must be pushed first and released last to avoid activating cassette.	Field (both read and write) 0 has ROM and RAM 1 has data RAM 2 has ROM only

To Use Any Program

1. Turn box off and on, if needed. The cassette program is wiped out during data collection

2. Turn function switch to read cassette position (black)

3. Press display

4. Start tape recorder

5. Display blanks out when leader is detected

6. Display returns with check sum shown (should be 0).

To Use Check Tape Program

1. Load as described above

2. Write: 1000 for minisample (or 1002 for block tapes) into memory address

3. Begin execution as described above

4. Start tape recorder. Program will read cassette tape and stuff characters in Field 1 buffer beginning at Loc 1; simultaneously, the same information will be displayed in LED readout; bad bits in leader (377s) and in line feeds (000s) should be watched for -- they can be seen despite the high speed of the incoming data stream; display format:

PE = parity error
FE = framing error
OE = overrun error

5. For minisample tapes, read about 30 blocks or as many as there are (maximum = 69 blocks); for block tapes, read about 30 (maximum = 80)

6. To look at raw data, hit the sample key to return to monitor (see above for memory commands; data in Field 1

7. To evaluate data, hit start; display will blank during processing; display will return when program returns to monitor; display interpretation:

-2 Cannot find such bytes
-1 Cannot find leader

These errors can occur in the middle of a minisample tape when the previous blocks are correct; Loc 23, Field 0 should be examined to get the number of blocks in error before this point.

0 No errors detected
any + number

Number or error correcting code blocks found in error

8. If desired, data may be examined in Field 1; the format of the data is listed in Table 1; data begins at Loc 1, Field 1

9. Program may be restarted at either address as many times as needed without reloading; go to Step 2.

```
                /
                /DEFINITIONS FROM NOISEMON
                OCTBCD=6363
                FUN2AA=5634
                FUN2AB=5636
                KD10=156
                KD20=160
                KD200=173
                KD400=172
                KD3777=7350
                KD7700=165
                KD7760=166
                KM12=155
                MODE=125
                TSW3=210
                TSW5=222
                SUB=115
                GOSUB=JMS SUB
                IRETN=123
                RETURN=JMP I IRETN
                /
                DEFINE CALL XX
                <       GOSUB
                        XX
                >
                /
                /
                /CASSETTE COMMANDS FOR EMULATOR
                CSSF=6061
                CSLS=6066
                /
                /
                /
                *1000
1000  7240      MINI,   STA             /ENTRY FOR MINI SAMPLE
1001  7410              SKP
                /
1002  7200      BLOCK,  CLA             /ENTRY FOR BLOCK TAPES
1003  3253              DCA FLAG
1004  6002              IOF             /READ TAPE UNTIL START SWITCH IS DEPRESSED.
1005  6627              TIN             /THEN JUMP TO ERROR CHECKER
1006  7200              CLA
1007  3010              DCA 10          /SET UP BUFFER STARTING AT LOC. 1
1010  1377              TAD (5000+TSW5  /INHIBIT START AND SAMPLE SWITCHES
1011  3776              DCA TSW3
                /
1012  6622      UIN,    FUNLO           /CHECK FOR ESCAPES
1013  0160              AND KD20        /SAMPLE SWITCH
1014  7640              SZA CLA
1015  5177              JMP 177         /YES.  EXIT TO MONITOR
                /
1016  6622              FUNLO
1017  0156              AND KD10        /START SWITCH?
1020  7640              SZA CLA
1021  5254              JMP INP1        /YES.  EXIT TO TAPE ANALYZER
                /
1022  6626              STATIN          /A CHAR AVAILABLE?
1023  0173              AND KD200
1024  7650              SNA CLA
1025  5212              JMP UIN
```

```
                    /
1026  6626          STATIN          /FOUND A CHAR.  GET ERROR BITS
1027  0375          AND (70)
1030  7002          BSW
1031  3006          DCA 6           /SAVE FOR LATER
1032  6627          TIN             /GET THE CHAR
1033  0374          AND (377)       /MASK OUT UNUSED BITS
1034  1006          TAD 6           /PUT IT ALL TOGETHER
1035  3022          DCA 22
1036  1022          TAD 22
1037  4115          CALL OCTBCD     /DISPLAY
1040  6363
1041  4115          CALL FUN2AB
1042  5636
                    /
1043  1160          TAD KD20        /AND STORE IN FIELD 1
1044  1125          TAD MODE
1045  6607          SELMD           /SET FIELD HARDWARE
1046  1022          TAD 22
1047  3410          DCA I 10        /STUFF DATA--1 BYTE PER WORD
1050  1125          TAD MODE        /RESTORE FIELD HARDWARE
1051  6607          SELMD
1052  5212          JMP UIN         /GET MORE
                    /
                    /
1053  0000  FLAG,   0               /0 FOR BLOCK, -1 FOR MINI
                    /
                    /
1054  7200  INP1,   CLA
1055  3011          DCA 11          /SET UP DATA RETRIEVAL
1056  1011          TAD 11
1057  3012          DCA 12
1060  3023          DCA 23          /ZERO ERROR COUNT
1061  1253          TAD FLAG
1062  7700          SMA CLA
1063  5270          JMP BLK
                    /
1064  4274  MINX,   JMS LEADER      /DECODE MINI
1065  4321          JMS SYNCH
1066  4773          JMS ERRCD
1067  5264          JMP MINX
                    /
1070  4274  BLK,    JMS LEADER      /DECODE BLOCK DATA
1071  4321          JMS SYNCH
1072  4773          JMS ERRCD
1073  5272          JMP .-1         /FOREVER
                    /
1074  0000  LEADER, 0
1075  7350          KD3777          /TURN OFF DISPLAY
1076  0125          AND MODE
1077  3125          DCA MODE
1100  1125          TAD MODE
1101  6607          SELMD
                    /
1102  1165          TAD KD7700      /SET LEADER COUNTER
1103  3007          DCA 7
                    /
1104  1372  LR1,    TAD (-4)
1105  3005          DCA 5
1106  2007  LR2,    ISZ 7           /ONLY UP TO 64 TRIES
```

```
1107  7410          SKP
1110  5771          JMP ERR1          /COULDN'T FIND LEADER!
                    /
1111  4770          JMS TEST          /IS THERE ENOUGH CHARS IN BUFFER?
                    /
1112  4767          JMS GETBYT        /GET A CHAR FROM BUFFER (MASKED)
1113  1366          TAD (-377)        /IS IT AN ERROR FREE LEADER CHAR? (A 377)
1114  7640          SZA CLA
1115  5304          JMP LR1           /NO!  RESET COUNTER AND TRY AGAIN
                    /
1116  2005          ISZ 5             /YES!  ADVANCE COUNTER.  LOOK FOR
1117  5306          JMP LR2           /4 CONSECUTIVE LEADER CHARS
                    /
1120  5674          JMP I LEADER      /OK.  POSITIVE IDENT OF LEADER.  EXIT.
                    /
                    /
1121  0000  SYNCH, 0
1122  1166          TAD KD7760        /SET UP LEADER COUNTER
1123  3007          DCA 7
                    /
1124  3004  SYN1,   DCA 4             /CLEAR ERROR COUNTER
1125  1365          TAD (13)          /LOOK FOR FIRST SYNCH BYTE (13)
1126  4764          JMS SYNS
1127  7346          KM3               /ALLOW UP TO 3 ERRORS
1130  1004          TAD 4
1131  7740          SZA SMA CLA
1132  5324          JMP SYN1          /TOO MANY TRY AGAIN.
                    /
1133  1363          TAD (320)         /LOOK FOR SECOND SYNCH BYTE (320)
1134  4764          JMS SYNS
1135  1362          TAD (-6)          /ALLOW UP TO SIX ERRORS TOTAL
1136  1004          TAD 4
1137  7740          SZA SMA CLA
1140  5324          JMP SYN1
                    /
                    /
1141  1155          TAD KM12          /DONE WITH SYNCH.  SET UP ERRCD
1142  3024          DCA 24            /SET UP 12 BIT WORDS.
1143  3030          DCA 30            /CLEAR RESULT WORD
1144  1361          TAD (-63)         /SET UP 51 BIT DATA STREAM
1145  3025          DCA 25
1146  7240          STA               /INITIALIZE BYTE GETTER
1147  3026          DCA 26
1150  3027          DCA 27            /CLEAR POLYNOMIAL RESULT
                    /
1151  5721          JMP I SYNCH
                    /
1161  7715          PAGE
1162  7772
1163  0320
1164  1200
1165  0013
1166  7401
1167  1216
1170  1420
1171  1401
1172  7774
1173  1256
1174  0377
1175  0070
```

```
1176  0210
1177  5222
                /
                /
1200  0000  SYNS,   0
1201  3005          DCA 5           /SAVE SYNCH BYTE REFERENCE
1202  2007          ISZ 7           /ALLOW ONLY 16 TRIES
1203  7410          SKP
1204  5777          JMP ERR2        /COULDN'T FIND SYNCH BYTES
                /
1205  4776          JMS TEST        /ARE THERE ENOUGH CHARS IN BUFFER?
                /
1206  4216          JMS GETBYT      /GET A CHAR FROM BUFFER (MASKED)
1207  4231          JMS EXOR        /AND COMPARE
1210  7110          CLL RAR         /COUNT THE NUMBER OF DIFFERENCE BITS
1211  7430          SZL
1212  2004          ISZ 4
1213  7440          SZA             /END OF LOOP TEST
1214  5210          JMP .-4
                /
1215  5600          JMP I SYNS      /OK. DONE WITH CHECK
                /
                /
1216  0000  GETBYT, 0
1217  1160          TAD KD20        /DATA IS STORED IN FIELD ONE
1220  1125          TAD MODE
1221  6607          SELMD
1222  1411          TAD I 11
1223  3006          DCA 6
1224  1125          TAD MODE        /RESTORE HARDWARE TO FIELD ZERO
1225  6607          SELMD
1226  1006          TAD 6
1227  0375          AND (377)       /MASK OUT ERROR BITS
1230  5616          JMP I GETBYT    /EXIT WITH CHAR IN ACCUMULATOR
                /
                /
1231  0000  EXOR,   0
1232  3006          DCA 6           /SAVE ONE ARG.  OTHER IS IN 5
1233  1006          TAD 6           /EXOR = X BAR & Y + X & Y BAR
1234  7040          CMA
1235  0005          AND 5
1236  7421          MQL
1237  1005          TAD 5
1240  7040          CMA
1241  0006          AND 6
1242  7501          MQA             /A PDP-8 INCLUSIVE OR
1243  5631          JMP I EXOR      /EXIT WITH RESULT IN ACCUMULATOR
                /
                /
1244  0000  PUTWRD, 0
1245  3006          DCA 6           /SAVE DATA TEMPORARILY
1246  1160          TAD KD20        /DATA IS STORED IN FIELD 1
1247  1125          TAD MODE
1250  6607          SELMD
1251  1006          TAD 6
1252  3412          DCA I 12
1253  1125          TAD MODE        /RESTORE HARDWARE TO FIELD ZERO
1254  6607          SELMD
1255  5644          JMP I PUTWRD
                /
```

```
                /
1256  0000  ERRCD,  0
1257  4776          JMS TEST          /CHECK TO SEE IF THERE ARE 8 BYTES AVAILABLE
                /
                /
1260  4312  CD2,    JMS CHKBIT
1261  1030          TAD 30            /SHIFT BIT INTO DATA
1262  7004          RAL
1263  3030          DCA 30
1264  2024          ISZ 24            /DONE WITH 12 BITS?
1265  5273          JMP CD1
                /
1266  1155          TAD KM12          /YES.  RESET COUNTER
1267  3024          DCA 24
1270  1030          TAD 30            /AND STUFF IN MEMORY
1271  4244          JMS PUTWRD
1272  3030          DCA 30            /CLEAR DATA WORD
                /
1273  2025  CD1,    ISZ 25            /DONE WITH 51 BITS?
1274  5260          JMP CD2
                /
1275  1155          TAD KM12          /DO 12 BIT CHECK CODE
1276  3025          DCA 25
1277  4312  CD3,    JMS CHKBIT        /IGNORE DATA BIT
1300  2025          ISZ 25
1301  5277          JMP CD3
                /
1302  1027          TAD 27            /IS REMAINDER ZERO?
1303  7640          SZA CLA
1304  2023          ISZ 23            /NO? ADVANCE ERROR COUNT
                /
1305  4312          JMS CHKBIT        /CLEAR OUT PARITY.  NOT CHECKED.
1306  1374          TAD (-63)
1307  3025          DCA 25
1310  3027          DCA 27            /RESET COUNTER AND REMAINDER
                /
1311  5656          JMP I ERRCD
                /
                /
1312  0000  CHKBIT, 0
1313  2026          ISZ 26            /DONE WITH 8 BITS?
1314  5321          JMP CHK1
1315  4216          JMS GETBYT        /YES. GET MORE..
1316  3031          DCA 31
1317  1373          TAD (-10)         /RESET COUNTER
1320  3026          DCA 26
                /
1321  1031  CHK1,   TAD 31            /GET BIT FROM BYTE
1322  7104          CLL RAL
1323  3031          DCA 31
1324  1031          TAD 31
1325  0172          AND KD400
1326  3005          DCA 5
1327  7330          K4000             /GET HIGH ORDER BIT OF REMAINDER
1330  0027          AND 27
1331  7112          CLL RTR
1332  7010          RAR               /ALIGN FOR DATA BIT
1333  4231          JMS EXOR
1334  7100          CLL               /IN CASE BIT IS ZERO
1335  7650          SNA CLA
```

```
1336  5345                JMP CHK2          /BIT IS ZERO
                  /
1337  1372                TAD (5234)        /BIT IS ONE.  USE CODE POLYNOMIAL
1340  3005                DCA 5
1341  1027                TAD 27
1342  4231                JMS EXOR
1343  3027                DCA 27
1344  7120                STL
                  /
1345  1027  CHK2,   TAD 27                  /UPDATE REMAINDER
1346  7004                RAL
1347  3027                DCA 27
                  /
1350  1031                TAD 31            /GET BIT IN LINK AGAIN
1351  7100                CLL
1352  0172                AND KD400
1353  7640                SZA CLA
1354  7020                CML
1355  5712                JMP I CHKBIT      /AND EXIT
                  /
1372  5234                PAGE
1373  7770
1374  7715
1375  0377
1376  1420
1377  1400
                  /
                  LNKTAB 1477
                  /
                  /
                  /ERROR EXIT
1400  7001  ERR2,   IAC
1401  7001  ERR1,   IAC
1402  3002          DCA 2
1403  1002          TAD 2
1404  5207          JMP DON1
                  /
                  /
                  /NORMAL EXIT
1405  3002  DONE,   DCA 2                  /SET SIGN +
1406  1023          TAD 23                  /GET NUMBER OF ERRORS
1407  4115  DON1,   CALL FUN2AA             /AND DISPLAY
1410  5634
1411  1125          TAD MODE                /AFTER TURNING DISPLAY ON
1412  7104          CLL RAL
1413  7130          STL RAR
1414  3125          DCA MODE
1415  1125          TAD MODE
1416  6607          SELMD
1417  5177          JMP 177                 /AND RETURN TO MONITOR
                  /
                  /
1420  0000  TEST,   0
1421  1011          TAD 11                  /ARE THERE AT LEAST 8 BYTES AVAILABLE?
1422  7161          CIA STL
1423  1010          TAD 10
1424  1277          TAD (-10)
1425  7620          SNL CLA
1426  5205          JMP DONE
1427  5620          JMP I TEST
1477  7770          PAGE
```

```
/
                *27
                /
0027  0000  DMPFLD, 0                /POINTER FOR FIELD TO BE DUMPED
0030  1000  DMPBEG, 1000             /FIRST ADDRESS TO BE DUMPED
0031  1500  DMPEND, 1500             /LAST ADDRESS TO BE DUMPED
0032  0000  DMPBG,  0                /POINTER FOR ADDRESS TO BE DUMPED
0033  0000  DMPX,   0                /TEMPORARY LOCATION FOR DUMP ROUTINE
0034  0000  DMPCK,  0                /HOLDS CHECKSUM DURING READ AND WRITE CASSETTE
0035  0000  OCT1,   0                /TEMPORARY STORAGE FOR OCTOUT
0036  0000  OCT2,   0                /    "       "     "    "
                /
                /
0037  0000  BUF,    0
0040  0000          0
                /
                /
                /
                /
                *7200
                /   PUNCH OUTPUT PROGRAM
                /
                /         OUTPUT LEADER(16 - '377' AND 1 - '000')
                /
7200  6066  STAR,   CSLS             /INITIALIZE PUNCH
7201  1377          TAD (-20
7202  3037          DCA BUF
7203  1376  ULOOP1, TAD (377        /OUTPUT LEADER '377'
7204  4276          JMS PUNCH       /CALL PUNCH OUTPUT SUBROUTINE
7205  2037          ISZ BUF
7206  5203          JMP ULOOP1
7207  4276          JMS PUNCH       /OUTPUT '000'
                /
                /
                /
                /   MAIN OUTPUT ROUTINE
                /
                /
7210  3034  PUNOUT, DCA DMPCK        /ZERO CHECKSUM
7211  1027          TAD DMPFLD      /SEND FIELD DIGIT FIRST
7212  7421          MQL
7213  1030          TAD DMPBEG      /THEN SEND BEGINNING ADDRESS
7214  4253          JMS ULOOP2
7215  1031          TAD DMPEND      /SEND END ADDRESS
7216  7421          MQL
7217  1031          TAD DMPEND      /COMPUTE NUMBER OF WORDS TO SEND
7220  7040          CMA
7221  1030          TAD DMPBEG
7222  3037          DCA BUF
7223  1037          TAD BUF
7224  4253          JMS ULOOP2
                /
7225  1030          TAD DMPBEG      /SET UP ADDRESS POINTER
7226  3032          DCA DMPBG
                /
7227  1432  PUNDAT, TAD I DMPBG
7230  7421          MQL
7231  2032          ISZ DMPBG
7232  7000          NOP
7233  1432          TAD I DMPBG
```

```
7234  2032          ISZ DMPBG
7235  7000          NOP
7236  3036          DCA OCT2
7237  1036          TAD OCT2
7240  4253          JMS ULOOP2      /SEND DATA
                /
7241  2037          ISZ BUF         /ADVANCE COUNTER
7242  7410          SKP
7243  5246          JMP .+3
7244  2037          ISZ BUF
7245  5227          JMP PUNDAT
7246  1034          TAD DMPCK       /DONE.  SEND CHECKSUM
7247  7041          CIA
7250  4276          JMS PUNCH
7251  7402          HLT
7252  5200          JMP STAR
                /
                /
7253  0000  ULOOP2, 0
7254  7521          SWP             /GET FIRST DATA INTO ACCUMULATOR
7255  3033          DCA DMPX
7256  1033          TAD DMPX        /SEND FIRST 8 BITS
7257  4276          JMS PUNCH
7260  1033          TAD DMPX
7261  0375          AND (7400       /GET REMAINING
7262  3033          DCA DMPX
7263  7501          MQA             /PICK UP HIGH 4 BITS FROM SECOND WORD
7264  0375          AND (7400
7265  7112          CLL RTR
7266  7012          RTR
7267  1033          TAD DMPX        /COMBINE
7270  7112          CLL RTR         /AND ALIGN
7271  7012          RTR
7272  4276          JMS PUNCH       /ALL HIGH BITS
7273  7501          MQA             /GET REMAINING BITS FROM SECOND WORD
7274  4276          JMS PUNCH
7275  5653          JMP I ULOOP2
                /
                /
                /
                /
                /
                /          PUNCH(PUNCH OUTPUT SUBROUTINE)
                /
7276  0000  PUNCH,  0
7277  0376          AND (377        /MASK EXTRANEOUS
7300  3036          DCA OCT2        /SAVE DATA
7301  6061          CSSF            /GET PUNCH STATUS
7302  5301          JMP .-1         /LOOP UNTIL PUNCH IS READY
7303  1036          TAD OCT2        /GET DATA
7304  6066          CSLS            /OUTPUT DATA
7305  7200          CLA
7306  1034          TAD DMPCK       /UPDATE CHECKSUM
7307  1036          TAD OCT2
7310  3034          DCA DMPCK
7311  5676          JMP I PUNCH
                /
                /
7375  7400          $
7376  0377
7377  7760
```

```
0000  0041  0177
1000  1152  1160
1200  1356  1371
1400  1430  1476
7200  7312  7374

BLK      1070      PUTWRD  1244
BLOCK    1002      RETURN  5523
BUF      0037      STAR    7200
CD1      1273      SUB     0115
CD2      1260      SYNCH   1121
CD3      1277      SYNS    1200
CHKBIT   1312      SYN1    1124
CHK1     1321      TEST    1420
CHK2     1345      TSW3    0210
CSLS     6066      TSW5    0222
CSSF     6061      UIN     1012
DMPBEG   0030      ULOOP1  7203
DMPBG    0032      ULOOP2  7253
DMPCK    0034
DMPEND   0031
DMPFLD   0027
DMPX     0033
DONE     1405
DON1     1407
ERRCD    1256
ERR1     1401
ERR2     1400
EXOR     1231
FLAG     1053
FUN2AA   5634
FUN2AB   5636
GETBYT   1216
GOSUB    4115
INP1     1054
IRETN    0123
KD10     0156
KD20     0160
KD200    0173
KD3777   7350
KD400    0172
KD7700   0165
KD7760   0166
KM12     0155
LEADER   1074
LR1      1104
LR2      1106
MINI     1000
MINX     1064
MODE     0125
OCTBCD   6363
OCT1     0035
OCT2     0036
PUNCH    7276
PUNDAT   7227
PUNOUT   7210
```

CERL DISTRIBUTION

HQDA (SGRD-EDE)

Chief of Engineers
ATTN: DAEN-MPE-I/F. P. Beck
ATTN: DAEN-ASI-L (2)
ATTN: DAEN-MPR
ATTN: DAEN-RDL

US Army Medical Bioengineering
R&D Laboratory
Envr Protection Res Div
ATTN: LTC L. H. Reuter
Ft. Detrick
Frederick, MD 21701

US Army Aeromedical Research Lab
ATTN: R. T. Camp, Jr.
ATTN: CPT J. Patterson
Box 577
Ft. Rucker, AL 36360

USA-WES
ATTN: Library
ATTN: J. Stoll/WESSE

Director
6570 AMRL/BBE
ATTN: Dr. H. Von Gierke
ATTN: J. Speakman
ATTN: LTC D. Johnson, BBA
Wright-Patterson AFB, OH 45433

Navy Undersea Center, Code 401
ATTN: B. Gales
ATTN: B. Young
San Diego, CA 92132

Washington, DC
ATTN: Building Research Advisory Board
ATTN: Transportation Research Board
ATTN: Library of Congress (2)
ATTN: Dept of Transportation Library

EPA Noise Office
ATTN: A. Hicks, Rm 2113
ATTN: K. Williams, Rm 109
ATTN: T. O'Hare, Rm 907G

EPA Region III Noise Program
ATTN: P. Anderson

Illinois EPA
ATTN: DNPC/G. Zak
ATTN: B. Hellweg

EPA
ATTN: AW-471/C. Caccavari
ATTN: AW-471/H. Nozick
ATTN: AW-371/A. Konheim
ATTN: R. Marrazzo
ATTN: W. Sperry
ATTN: J. Goldstein
ATTN: D. Gray
ATTN: D. Mudarri
ATTN: R. Hayman
ATTN: R. A. Simmons

Aberdeen Proving Ground, MD 21005
Human Engineering Laboratory
ATTN: J. E. Weisz/AMZHE
ATTN: G. Garinther
Ballistics Research Laboratory
ATTN: B. Taylor
Army Environmental Hygiene Agency
ATTN: Chief, Bio-Acoustics Div

Defense Documentation Center (12)

National Bureau of Standards
ATTN: D. R. Flynn

NASA
ATTN: H. Hubbard

Engineering Societies Library
New York, NY 10017

True-integrating environmental noise monitor and sound exposure level meter / by P. D. Schomer...et al. -- Champaign, IL : Construction Engineering Research Laboratory ; Springfield, VA : available from NTIS, 1978 - 1980
4 v. ; 27 cm. (Technical report ; N-41)

1. Noise-measurement. I. Schomer, Paul D. II. Averbuch, Aaron J. III. Weisberg, M. W. IV. Brown, Ronald. V. Little, Lincoln M. VI. Series: U. S. Army Construction Engineering Research Laboratory. Technical report; N-41.