NBSIR 74-482

An Automated System for Precision Calibration of Accelerometers

B. F. Payne

Vibration Section, Mechanics Division
Institute for Basic Standards
National Bureau of Standards
Washington, D. C. 20234

April 1974

Final Report

Prepared for
Department of Defense
Calibration Coordination Group

NBSIR 74-482

AN AUTOMATED SYSTEM FOR PRECISION CALIBRATION OF ACCELEROMETERS

B. F. Payne

Vibration Section, Mechanics Division
Institute for Basic Standards
National Bureau of Standards
Washington, D. C. 20234

April 1974

Final Report

Prepared for
Department of Defense
Calibration Coordination Group

U. S. DEPARTMENT OF COMMERCE, Frederick B. Dent, Secretary

NATIONAL BUREAU OF STANDARDS, Richard W. Roberts, Director

FORWORD

The development of the automated calibration system for accelerometers was sponsored by the Department of Defense Calibration Coordination Group (DoD/CCG) consisting of: the Aerospace Guidance and Metrology Center, Newark Air Force Station, Newark, Ohio 43055; the Metrology and Calibration Center, Redstone Arsenal, Alabama 35809; and the Metrology Engineering Center, Bureau of Naval Weapons Representative, Pomona, California 91766. The Atomic Energy Commission was represented by an observer from Sandia Laboratory, Albuquerque, New Mexico 87115. The project was coordinated by the Aerospace Guidance and Metrology Center.

The Dod/CCG project number assigned was CCG 69-13 and work was performed under NBS cost center 2130423.

TABLE OF CONTENTS

Page

Abstract

1. Introduction 1-1

2. Hardware 2-1

2.1 Circuit A 2-1
2.2 Circuit B 2-1
2.3 Signal Junction Box 2-2
2.4 Minicomputer 2-2
2.5 Patch Panels 2-2
2.5.1 Patch Panel Circuit A for Manual Operation 2-2
2.5.2 Patch Panel Circuit B for Manual Operation 2-3
2.6 Magnetic Tape Drive 2-12
2.7 Exciters 2-12
2.8 Interfaces 2-12
2.8.1 DVM Interface 2-12
2.8.2 Frequency Counter Interface 2-15
2.8.3 Relay Interface 2-18
2.8.4 Oscillator Interface 2-18
2.8.5 X-Y Plotter Interface 2-23
2.9 Calibration of the System 2-25

3. Software 3-1

3.1 Computer Capability 3-1
3.2 Software Philosophy 3-1
3.3 Data Block 3-2
3.4 Start 1, Start 2 3-2
3.5 Halt and Stop Features 3-6
3.6 Provisions for Automatic Selection of Circuits and Automatic Ranging of Test Equipment 3-6

4. Software Program Descriptions 4-1

4.1 Hexidecimal-to-Decimal Conversion for Test Sensitivity 4-2
4.2 Range Test for Test Accelerometer 4-7
4.3 Check for INVERSE FLAG and Set Relays 4-12
4.4 Check for INVERSE FLAG and Calculate Calibration Factor for Test Accelerometer If Set 4-15
4.5 Clear Tables for Accelerometer Data Block 4-18
4.6 *Round-Off Subroutine 4-21
4.7 *ASCII-to-Decimal Subroutine 4-23
4.8 Check for UNSTABLE FLAG and Type Message If Set . . . 4-26
4.9 *Save Registers Subroutine 4-28
4.10 *Restore Registers Subroutine 4-29
4.11 Calibration Factors Exciter 2 4-30
4.12 Calibration Factors Exciter 1 4-31
4.13 Start 1 4-32
4.14 Start 2 4-34
4.15 Set Relays for Ratio I, Output Oscillator Code, Read Frequency Counter, Type It on TTY 4-35

TABLE OF CONTENTS

		Page
4.15.1	Programming Relay Banks 1 and 2	4-35
4.15.2	Programming the Decade Oscillator	4-38
4.15.3	Reading the Frequency Counter	4-39
4.16	Set Relays for Ratio II, Calculate Test Accelerometer Sensitivity	4-47
4.16.1	Programming Relay Banks 1 and 2	4-47
4.16.2	Calculation of Test Accelerometer Calibration Factor	4-47
4.17	Storage Table for Oscillator Code	4-52
4.18	TTY Printout	4-53
4.19	Table of Defined Constants	4-56
4.20	*Decimal to Hexidecimal Subroutine	4-58
4.21	*Wait Subroutine for TTY	4-61
4.22	*Decimal Point Typeout	4-61
4.23	Storage Table for Exciter Calibration Factors (Hex.)	4-62
4.24	Shut Off Oscillator, Ring Bell, Start Over	4-63
4.25	*Delay for DVM Subroutine	4-65
4.26	Halt, Stop, or Continue	4-66
4.27	*Constant Acceleration Closed Loop Control Subroutine	4-69
4.28	*Scope Scaling Subroutine	4-76
4.29	*Round Off Subroutine (for Constant Acceleration)	4-77
4.30	*Hexidecimal-to-Decimal Subroutine	4-79
4.31	*Calculate Delta V Subroutine	4-83
4.32	Data Block Control Constants (1400 Tables)	4-84
4.33	*Voltage/Frequency Data Modified to Proper Oscillator Code	4-84
4.34	*Set Slow/Fast Code for ac/dc Converters Subroutine	4-90
4.35	Store Calibration Data in Tables	4-92
4.36	Decimal to ASCII Routine (5 digits)	4-96
4.37	*Decimal to ASCII Subroutine (1 digit)	4-98
4.38	Check for Correct Range of Signal ac/dc Converter	4-100
4.39	*Multiple Readings (DVM) and Digital Filter Subroutine	4-103
4.40	*Read Data Subroutine	4-109
4.41	*ASCII to Decimal Subroutine	4-116
4.42	Desired Frequency Table	4-120
4.43	*Check for Proper Frequency Subroutine	4-121
4.44	*Type-Out Subroutine	4-124
4.45	*Message Entry Subroutine	4-130
4.46	Accelerometer Data Block Entry Program	4-135
4.47	Storage Table for Type Out -Constants and Save Locations	4-142
5.	Discussion	5-1
6.	Acknowledgments	6-1
7.	References	7-1

*Indicates subroutines

LIST OF TABLES

		Page
2-1.	Hardware Components	*
2-2.	DVM Cabling	2-14
2-3.	Counter Digital Output Connector Cabling	2-16
2-4.	Remote Preset Connector Cabling	2-17
2-5.	Relay Pin Connections	2-19
2-6.	Relay Control Line Connections	2-20
2-7.	Oscillator Cabling	2-21
4-1.	Relay Bank 1 Coding	4-36
4-2.	Relay Bank 2 Coding	4-37
4-3.	Data File Format	4-95

LIST OF FIGURES

1-1.	Oscillator Driving Voltage vs. Frequency for 10 g Acceleration	1-2
2-1.	Diagram of System Interconnections	2-4
2-2.	Block Diagram of Automated Accelerometer System	2-5
2-3.	Signal Junction Box Part 1	2-6
2-4.	Signal Junction Box Part 2	2-7
2-5.	Impedance Matching Capacitor Bank Schematic	2-8
2-6.	Schematic of Automated Accelerometer System	2-9
2-7.	Automated Calibration System	2-10
2-8.	Exciters 1 and 2	2-11
2-9.	Calibration Circuit Schematic	2-26
3-1.	Sense Status Loop	3-3
3-2.	Data Block Format	3-4
3-3.	Condensed Flow Chart by Program Name	3-5
3-4.	Summary Flow Chart	*
4-1.	Hexidecimal-to-Decimal conversion	4-6
4-2.	Range Test for Test Accelerometer Flowchart	4-11
4-3.	Check for Inverse Flag and Set Relays Flowchart	4-14
4-4.	Check for Inverse Flag and Calculate S_T if Set Flow Chart	4-16
4-5.	Five Byte ASCII-to-Decimal Flowchart	4-25
4-6.	Teletypewriter Printout Flow Chart	4-57
4-7.	Decimal-to-Hexidecimal Flowchart	4-60
4-8.	Halt, Stop, or Continue Flowchart	4-68
4-9.	Constant Acceleration Closed Loop Control Subroutine Flowchart	4-75

*Located in the back; after Page 7-1.

LIST OF FIGURES

PAGE

4-10. Hexidecimal-to-Decimal Subroutine Flowchart 4-81
4-11. Voltage/Frequency Data Modified to Proper Oscillator Code Flowchart 4-89
4-12. Decimal to ASCII Flowchart 4-99
4-13. Multiple DVM Reading and Delay Subroutine Flowchart 4-102
4-14. Read Data Subroutine Flowchart 4-111
4-15. ASCII-to-Decimal Subroutine Flowchart 4-119
4-16. Accelerometer Data Block Entry Program Flowchart 4-141
5-1. History of a Control Accelerometer for 1972 5-2
5-2. History of a Control Accelerometer for 1973 5-3
5-3. Histogram of a Control Accelerometer for 1972 5-4
5-4. Histogram of a Control Accelerometer for 1973 5-5

AN AUTOMATED SYSTEM FOR PRECISION CALIBRATION OF ACCELEROMETERS

by

B. F. Payne

ABSTRACT

The report describes an automated system for accelerometer calibration under real time control by a small, dedicated digital computer. The hardware components of the system are described and the software programs are given. The software automatically regulates the rate and amount of data collected based on analysis of input data. Printouts of the frequency response of test accelerometers is on a teletypewriter and also the response can be stored on a magnetic tape. Manual operation of the system is also described.

KEYWORDS: Acceleration, automation, calibration, measurements, minicomputer, shakers, standards, transducers, vibration, vibration exciters, vibration pickups.

1. INTRODUCTION

An automated process controller for precision accelerometer calibration has been developed by the Vibration Section of the National Bureau of Standards. The system is composed for the most part of commercially available test equipment. This system was designed to meet the need for an accurate automated calibration system for accelerometers. This final report describes the hardware and gives a complete listing of the software. The system is controlled by a minicomputer with 16,384 bytes of core memory (1 byte - 8 bits) and a Teletype (TTY). Since the physical phenomena involved react relatively slowly, the cycle time of the small computer is more than adequate for this purpose. The system controls two electrodynamic vibration exciters. Unlike some commercial exciters, the NBS standard Dimoff exciters [1,2]* have transfer functions which are easily adapted to closed loop acceleration control. Figure 1-1 shows the driving voltage of a Dimoff Type 200 exciter plotted against frequency with and without a capacitive impedance matching network.

The exciters have built-in standard accelerometers. They are first calibrated by absolute methods from 10 Hz to 10 kHz.[3,4] The automated system stores frequency and voltage ratio data at test acceleration levels. The frequencies and acceleration levels are stored in a Data Block table of values in the core of the computer. For maximum usefulness, the program is written so that changes in frequency and acceleration level can be made quickly at the teletype terminal or by loading paper or magnetic tape with no changes in the operating program required.

This system of accelerometer calibration differs from other automatic systems in that the software includes a selective procedure for screening data to minimize the effects of noise and drift. This program is called Multiple Readings (DVM) and Digital Filter Subroutine and it regulates the quantity of the test measurements to obtain data which lie within the established repeatability criteria (Section 4.38).

* The numbers in brackets refer to references found at the end of this report.

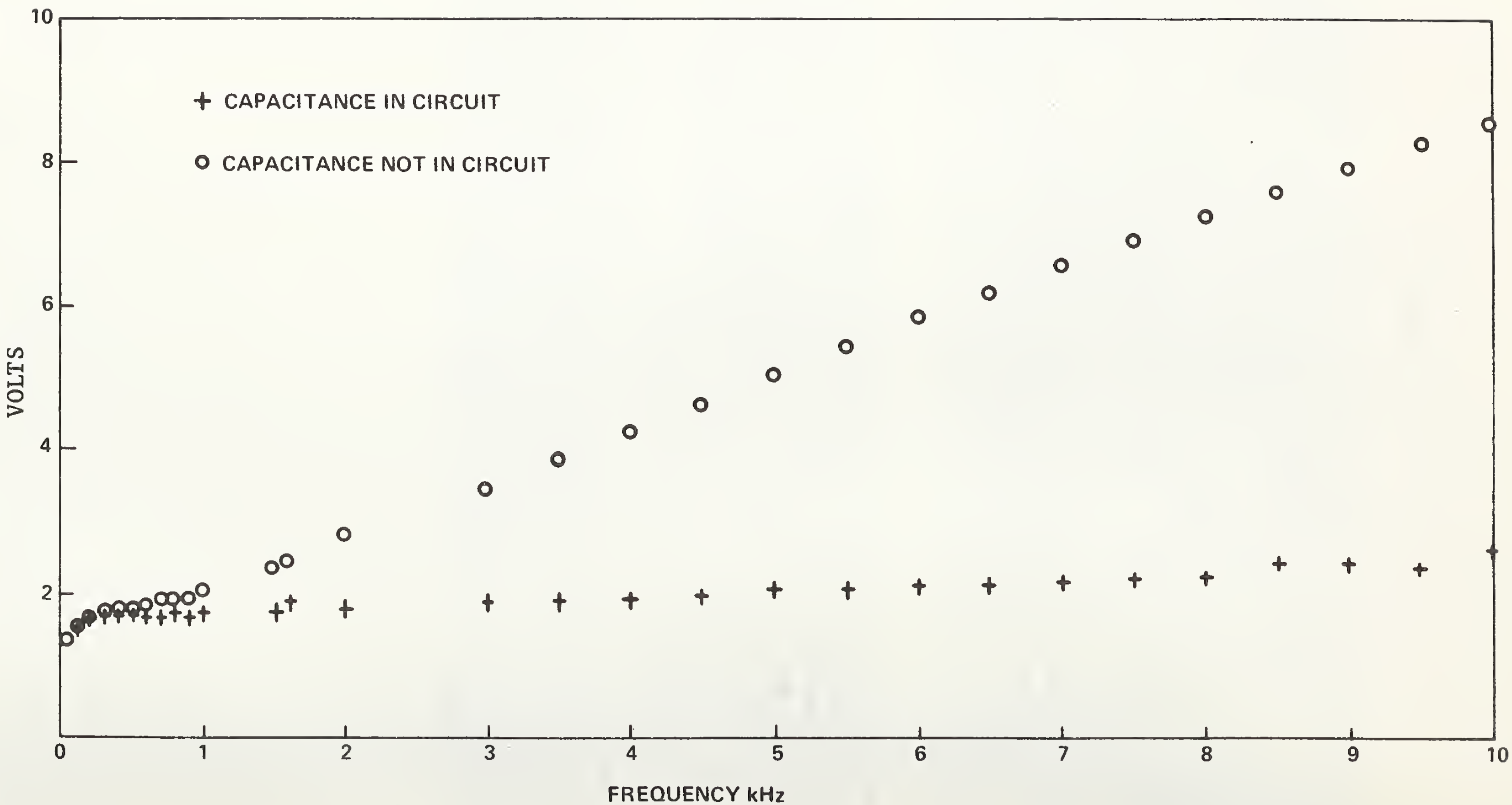


FIGURE 1-1. OSCILLATOR DRIVING VOLTAGE VS. FREQUENCY FOR 10 g ACCELERATION

2. HARDWARE

The hardware components for the automated system are listed in table 2-1 and are shown in a block diagram in figures 2-1* and 2-2 and manufacturer and model numbers are given in [4]. The internal connections of the signal junction box are shown in figures 2-3 and 2-4. The impedance matching capacitor bank schematic is shown in figure 2-5. The electrical schematic for the automated system is shown in figure 2-6. In this schematic, the relay positions are all shown in the normally closed position. Refer to the software description of program Set Relays for Ratio I, Output Oscillator Code for details in programming the relays for the circuit desired (Sect. 4.15).

The circuitry of this system allows voltage ratios to be taken by either of two circuits. The circuit used is chosen by a program based on data taken from the test accelerometer. For test accelerometers whose sensitivity is $\leq$566 mV/g, the circuit used will be called Circuit A. For sensitivities >566 mV/g, the circuit will be called Circuit B. The gain of the power amplifier should be set for approximately 60 percent of full gain for a 250-watt amplifier in order to utilize the full amplitude range of the digital oscillator.

2.1 Circuit A

This is the circuit used for most accelerometer calibrations. Referring to figure 2-6, the relays are set for RI ratio as described in Programming Relay Banks 1 and 2, Section 4.15.1. The program will then cause data to be read with the relays in this position. This is denoted as Ratio I or RI. After this is completed the relays are reset for RII ratio as described in Set Relays for Ratio II, Section 4.16. The program then causes data to be read. This is denoted as Ratio II or RII. The test accelerometer sensitivity can be calculated from these ratios as described in Section 4.16. The data are read under the supervision of the Multiple Readings subroutine, Section 4.39. This program will be described in detail in the Software section of this report.

2.2 Circuit B

This circuit is used for accelerometers whose sensitivity is >566 mV/g. If this circuit is chosen the INVERSE FLAG is set which triggers a program to reset the relays. See Check for INVERSE FLAG and Set Relays, Section 4.3. Referring again to figure 2-6, this program reverses the standard and test signals at the input to the readout circuitry and bypasses the amplifiers. Only Ratio RII is read in this case. See Section 4.3 for details. The sensitivity of the test accelerometer can then be computed as described in Section 4.4.

*This figure is at the end of the report.

2.3 Signal Junction Box

The signal junction box (figures 2-1, 2-3, and 2-4) allows for the test instruments to be connected to a junction box. The back of the junction box is interconnected with the relays controlled by the computer. In this manner the software can control the circuitry in the setup. For example, in circuits A and B described above, the software decides which circuit can be used based upon data taken from the test accelerometer. Once this decision has been made, the software sets the relays for the proper circuit and then proceeds with the calibration. Likewise, the computer can set the range of the ac/dc converters based upon data taken during the test.

Figure 2-7 shows the automated accelerometer calibration system with the associated electronic equipment. Figure 2-8 shows two vibration exciters used with the system.

2.4 Minicomputer

The computer system has 16,384 bytes of core memory and a Teletype (TTY). It includes High Speed Arithmetic and Read/Write Block Instructions. It includes six interface units to control the digital voltmeter (DVM), oscillator, frequency counter, X-Y plotter, a bank of 32 relays (12 double-pole, and 20 single-pole), and a 9-track magnetic drive unit.

2.5 Patch Panels

The only time the patch panels are used in the automatic mode is for selecting exciter 1 or 2 to be used as the calibration exciter. The plugs should connect points 21-22 for exciter 1 and points 45-46 for exciter 2 on Patch Panel 1 (Figure 2-1).

2.5.1 Patch Panel Circuit A for Manual Operation. The system can also be operated in a manual mode by use of the patch panels. For manual operation, the oscillator is in the manual mode with the desired frequency selected on the front panel of the oscillator. The output should be set for 5 to 10 volts. The acceleration level of the exciter is then controlled by the gain potentiometer of the power amplifier. For the manual equivalent of circuit A described above, additional plugs and patch cords are needed in Patch Panel 1: 1-2*, 7-8, and 22-23 for exciter 1, or 46-47 for exciter 2. To obtain RI use a plug to connect 9-10 and for RII connect 33-34. Cable 36 will have to be disconnected from point J on the junction box, the oscillator switched from auto to manual, switches S1 and S2 opened or their cables disconnected, and the DVM switched to ratio on the front panel. By adjusting the volta divider until the RI equals the magnitude of the standard accelerometer sensitivity, the RII will equal the test accelerometer sensitivity. Example: If the standard accelerometer sensitivity S_{Std} = 20.09 mV/g, then by adjusting the voltage divider until RI = 20090 on the DVM, RII will be the test accelerometer sensitivity, RII = 18532, and S_{Test} = 18.53 mV/g.

*1-2, etc., indicates a connection between points 1 and 2 in figure 2-1.

2.5.2 Patch Panel Circuit B for Manual Operation. For circuit B manual operation, the plugs and patch cords are removed. Plugs or patch cords are used to connect points 1-10, 32-33, and 22-23 for exciter 1 or 46-47 for exciter 2. Cable 36 will be disconnected from point J and cable 8 will be disconnected from point I of the junction box to take the computer controlled relays out of the circuit. The oscillator will be switched from auto to manual, switches S1 and S2 opened or their cables disconnected, and the DVM switched to ratio on the front panel. The voltage divider is not used in this circuit. This circuit is set up for RII. Only RII data are taken. The test sensitivity must be computed in this case from the following equation

$$S_{Test} = \frac{S_{Std}}{RII} .$$

TABLE 2-1. Hardware Components

Minicomputer

Teletype

Magnetic Tape Drive

Signal Junction Box (Relay Controlled)

Patch Panels 1 and 2 (Manual)

Test Equipment Controlled by Computer:

1 Digital voltmeter
1 Digital oscillator
1 Frequency counter
1 X-Y Plotter

Support Test Equipment:

3 Panel voltmeters
1 Power amplifier

2 dc voltage amplifiers
1 Manual capacitor bank
1 Voltage divider
1 Cathode follower
1 Phase meter

Test Equipment Controlled by Computer Controlled Relays:

2 ac/dc converters
*1 dc power supply
1 Capacitor bank (16 capacitor)
1 Oscilloscope
*1 Wave analyzer

*Equipment for future expansion of system to measure distortion.

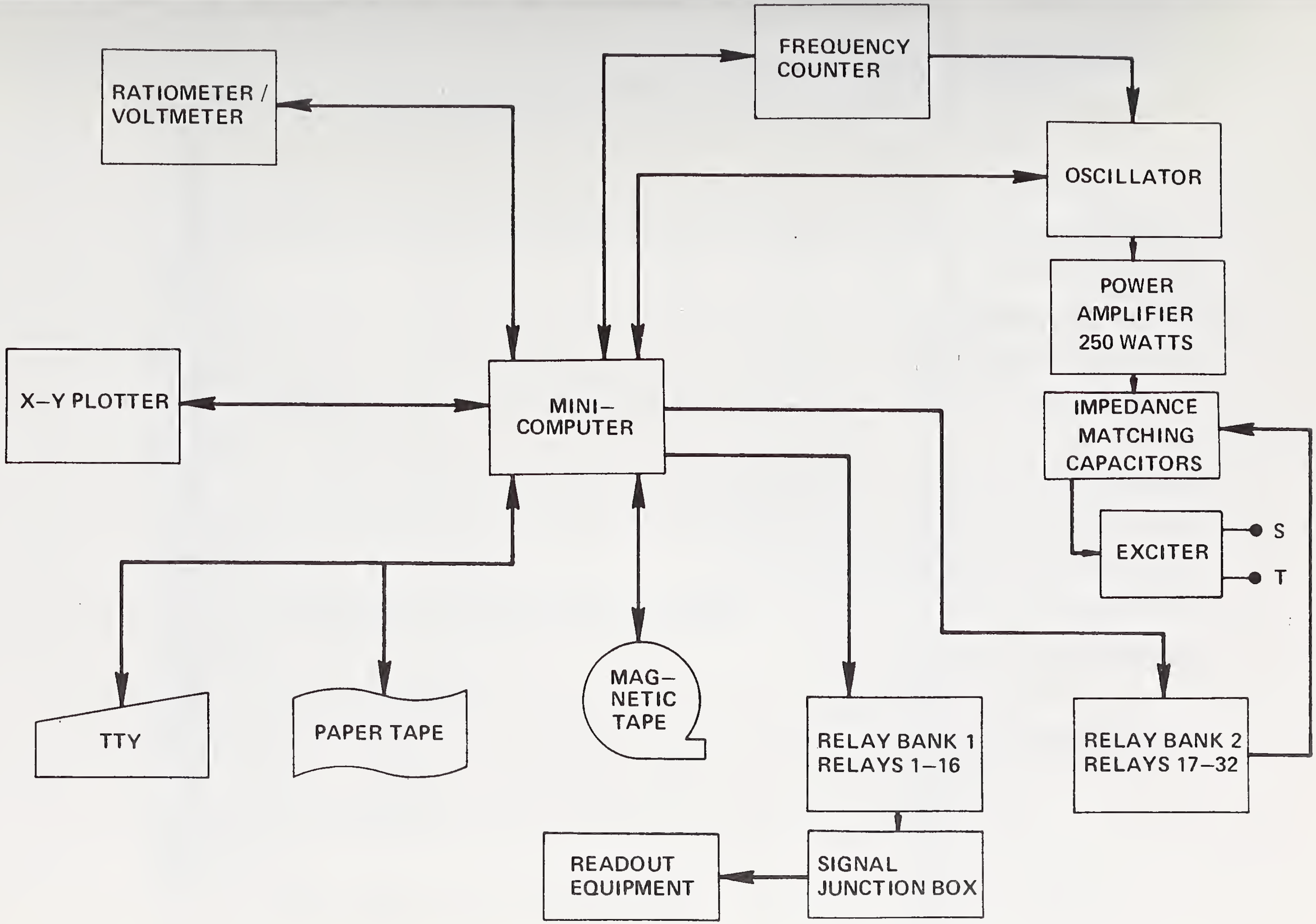
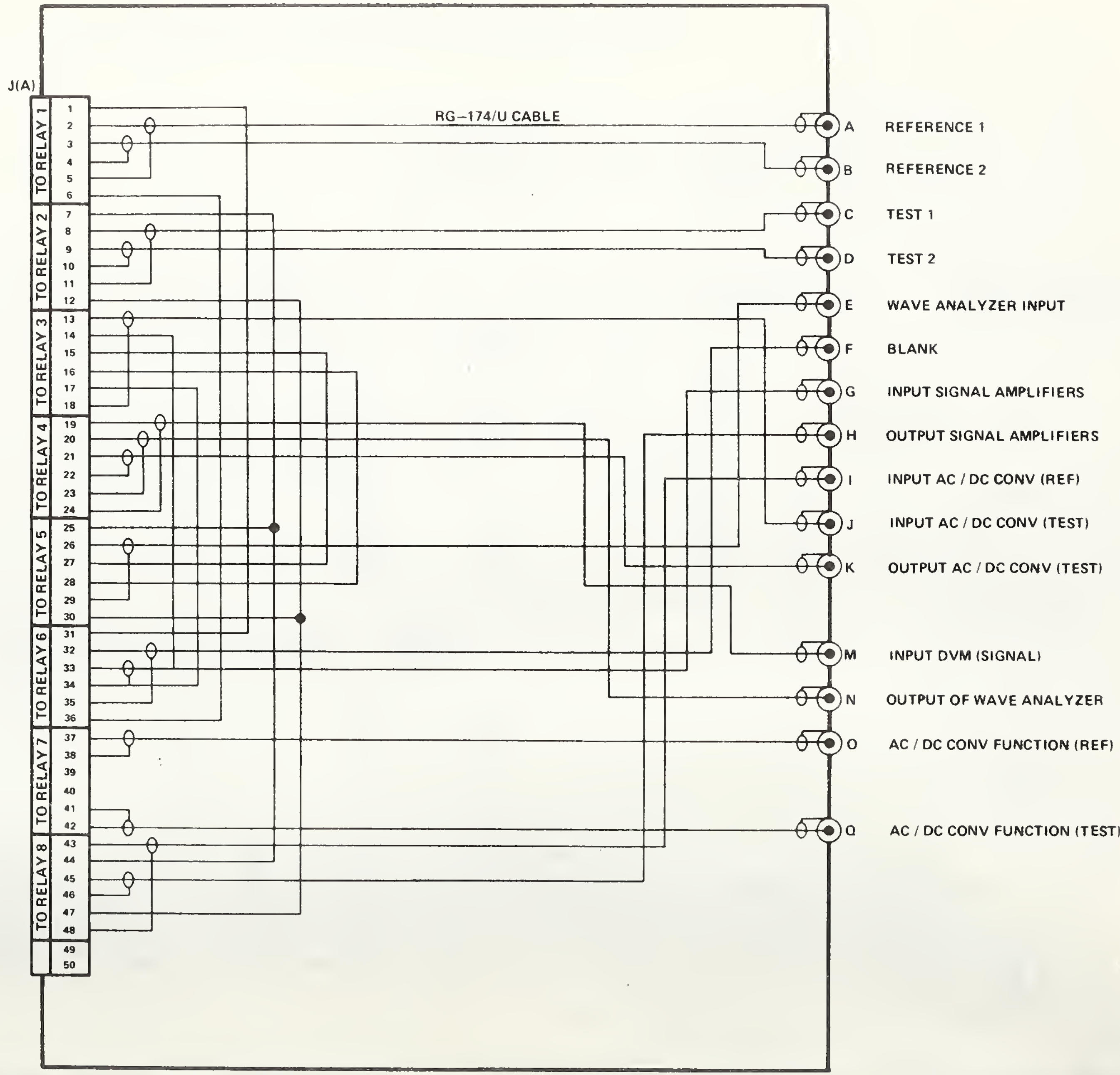


FIGURE 2-2. BLOCK DIAGRAM OF AUTOMATED ACCELEROMETER SYSTEM


J(A)
TO RELAY 1
TO RELAY 2
TO RELAY 3
TO RELAY 4
TO RELAY 5
TO RELAY 6
TO RELAY 7
TO RELAY 8
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
RG–174/U CABLE
A
REFERENCE 1
B
REFERENCE 2
C
TEST 1
D
TEST 2
E
WAVE ANALYZER INPUT
F
BLANK
G
INPUT SIGNAL AMPLIFIERS
H
OUTPUT SIGNAL AMPLIFIERS
I
INPUT AC / DC CONV (REF)
J
INPUT AC / DC CONV (TEST)
K
OUTPUT AC / DC CONV (TEST)
M
INPUT DVM (SIGNAL)
N
OUTPUT OF WAVE ANALYZER
O
AC / DC CONV FUNCTION (REF)
Q
AC / DC CONV FUNCTION (TEST)

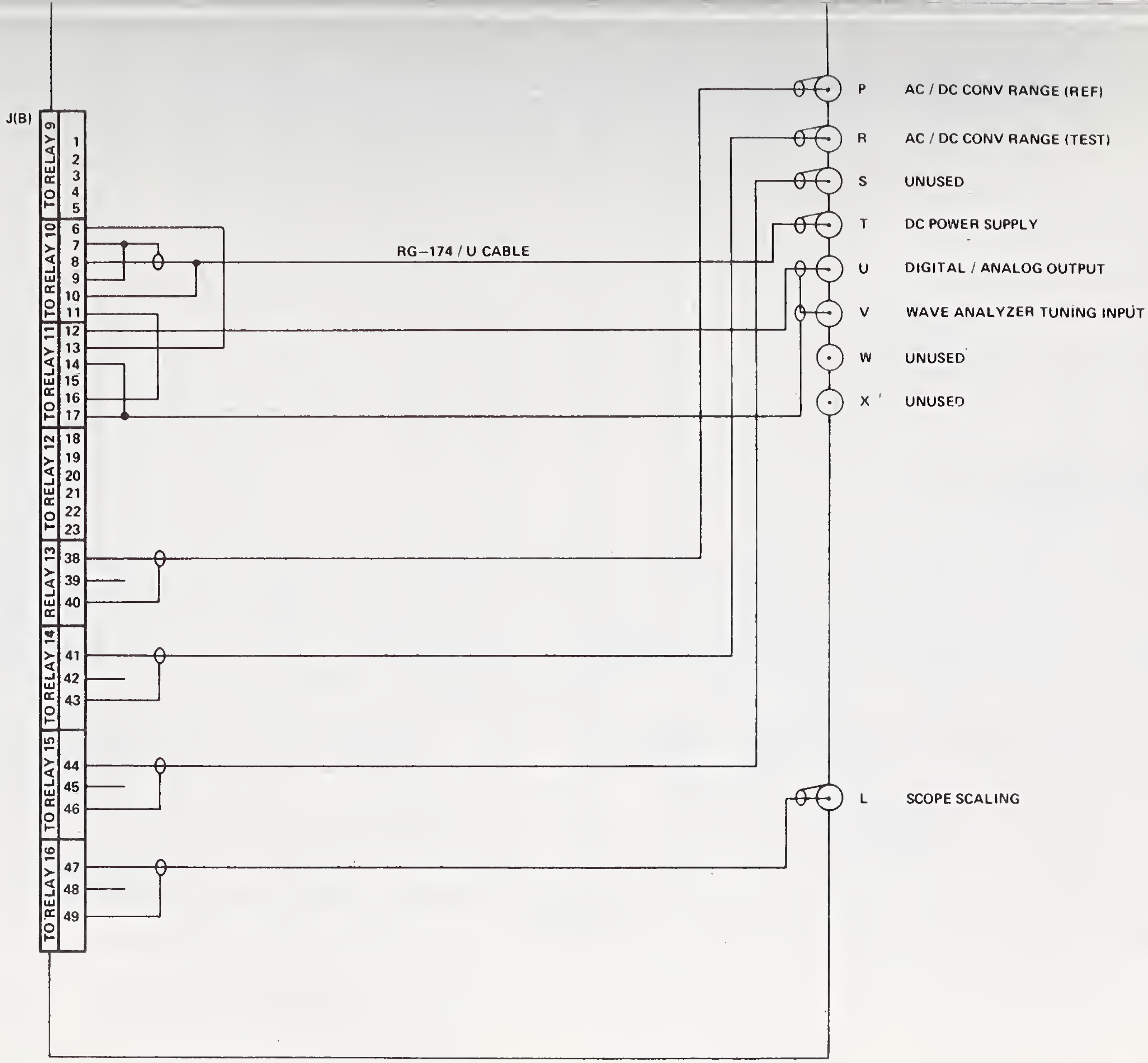


FIGURE 2-4. SIGNAL JUNCTION BOX PART 2

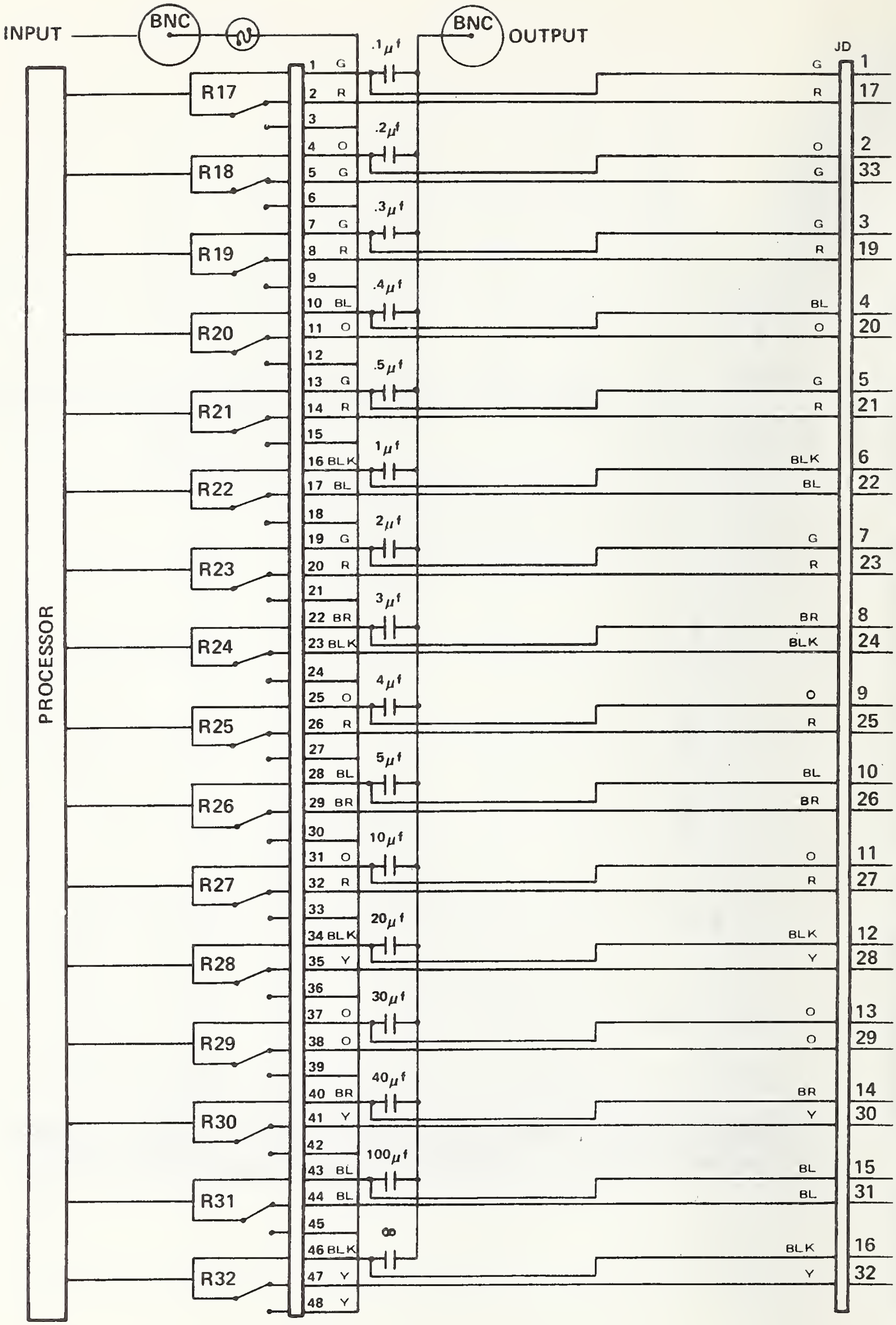


FIGURE 2-5. IMPEDANCE MATCHING CAPACITOR BANK SCHEMATIC

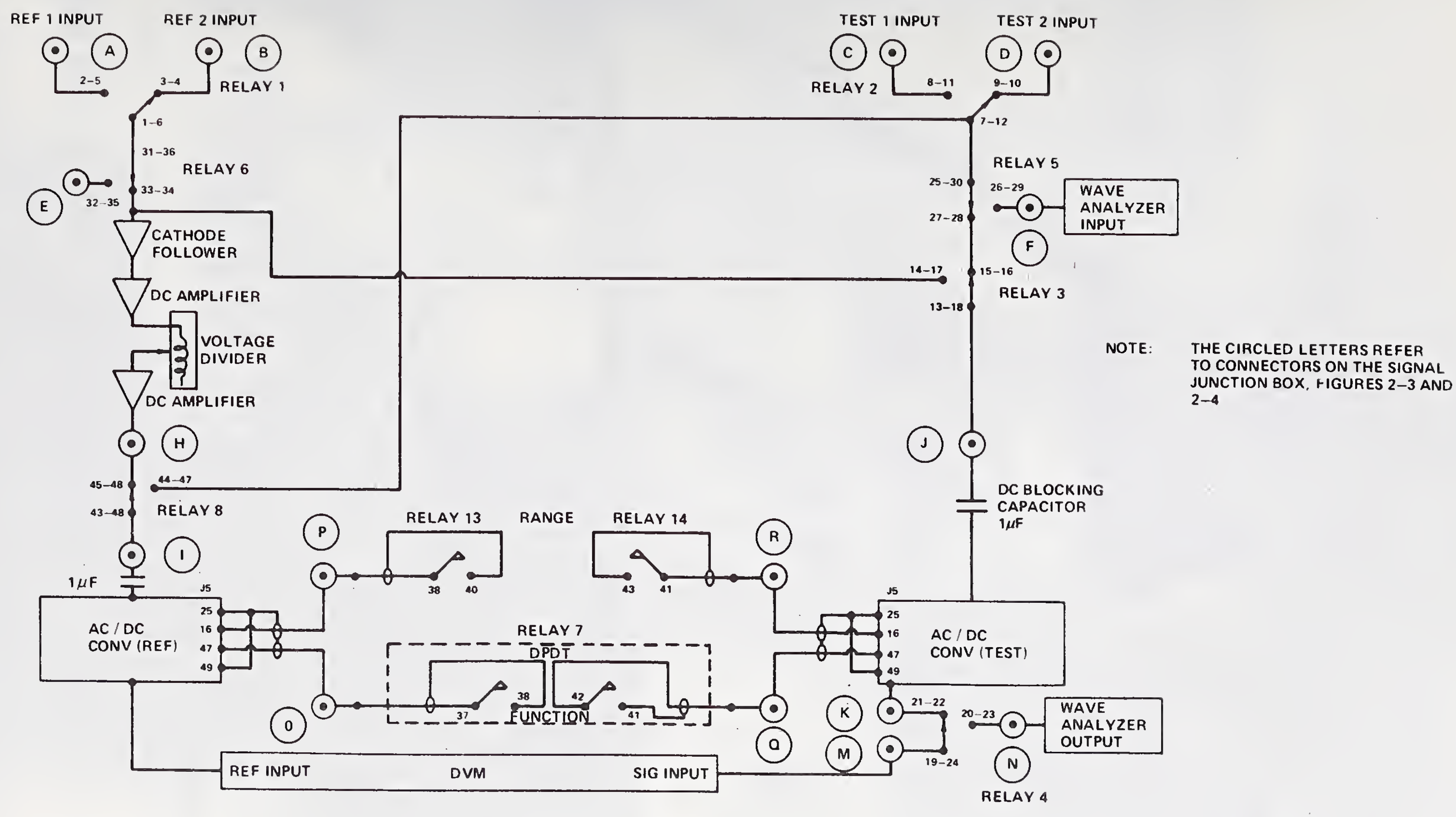


FIGURE 2-6. SCHEMATIC OF AUTOMATED ACCELEROMETER SYSTEM

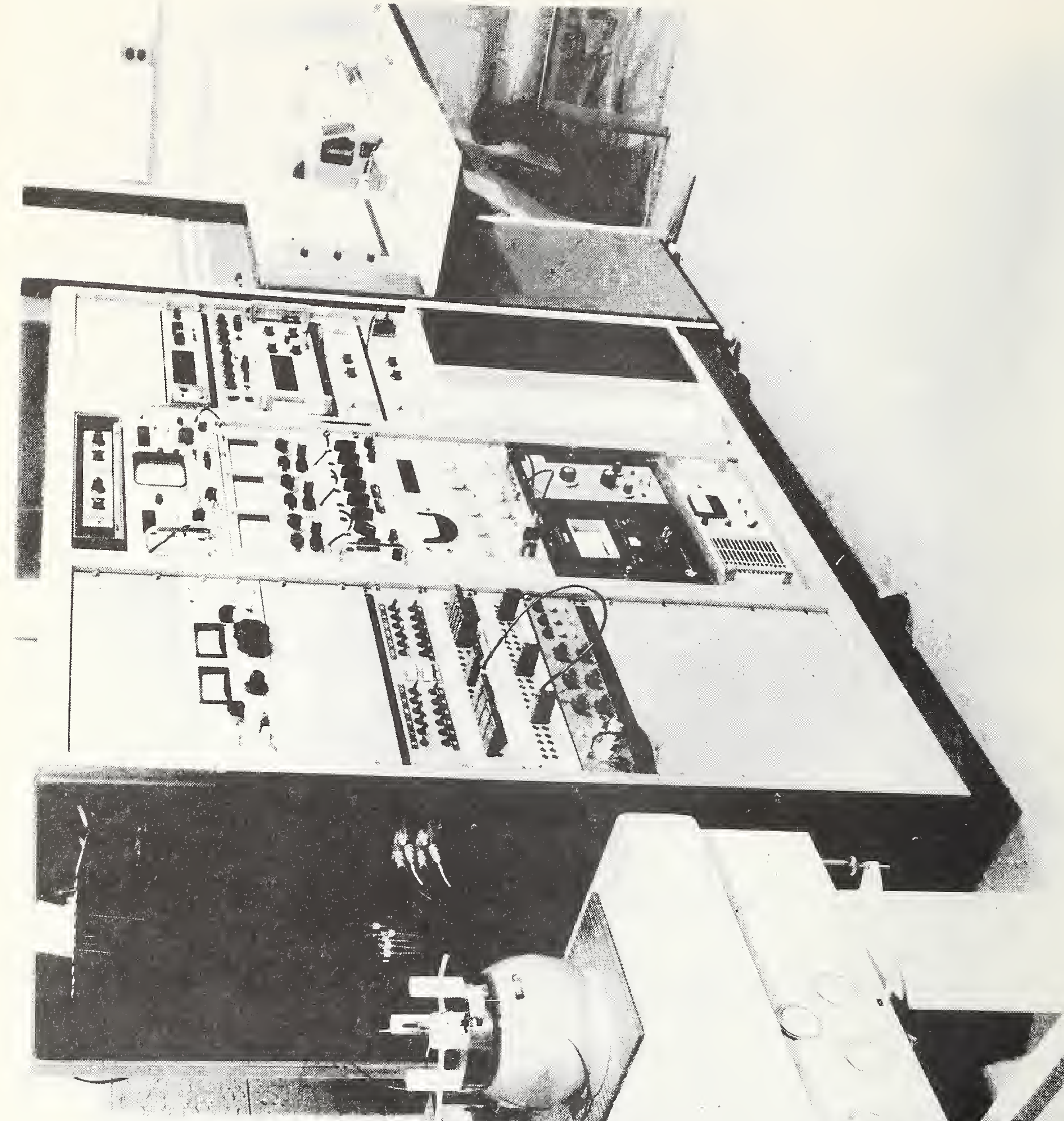


FIGURE 2-7. AUTOMATED CALIBRATION SYSTEM

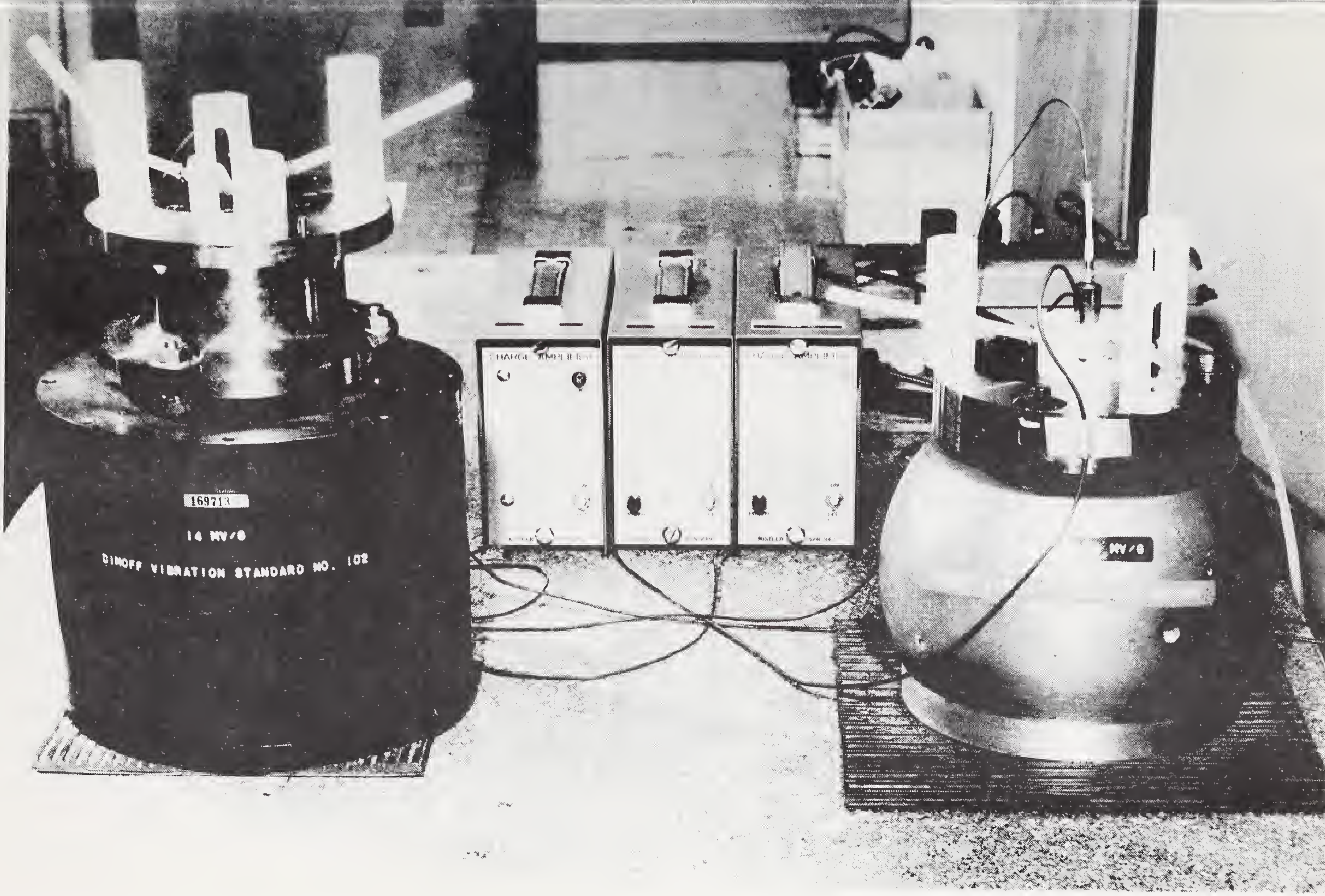


FIGURE 2-8. EXCITERS 1 AND 2

2.6 Magnetic Tape Drive

The digital magnetic tape drive is a nine-track 800-bpi (315-bpc) read after write version. The speed is 25 inches per second (63.5 cm per second) in both the forward and reverse directions. Tape capacity is 2400 feet (731.5 m) of 1/2 inch, 1.5 mil (1.27 cm, 0.038 mm) computer tape on standard 10-1/2 inch (26.67 cm) IBM compatible reels.

2.7 Exciters

Two Dimoff air bearing exciters are shown in figure 2-8. These are described in references 3 and 4. These exciters are calibrated by absolute methods of reciprocity and interferometry as described in references 1, 2, and 5. The exciter 2 has a frequency range of 10 Hz to 10,000 Hz and the exciter 1 has a frequency range of 10 Hz to 5000 Hz. They employ ceramic moving elements, air bearings, and permanent magnets.

2.8 Interfaces

2.8.1 DVM Interface. The interface has the ability, under program control to:

1. Select volt or ratio function,
2. Command DVM to convert, and
3. Read polarity code; five digits, and range code from the DVM. The polarity and data digits are converted, in the interface, to ASCII codes.

A priority interrupt circuit is included which, if enabled, will interrupt the processor when service is required. The Print Command line from the DVM will generate an interrupt when data are available to be read.

Program Notes

1. Strap options provide for the selection of any device address from X'00' to X'FF'.

2. Control functions are executed via OC (Output Command) instructions:

Bit	0	1	2	3	4	5	6	7
Command	DIS	EBL	CNVT	CCTR			RATIO	VOLT

DIS - Disables device interrupt from interrupting the processor. This does not prevent an interrupt from being queued up in the ATN (Attention) FF (Flip-Flop).

EBL - Enables device interrupt.

CNVT - Commands the DVM to perform a conversion of its inputs.

CCTR - Clear the Read Steering FF's (This counter controls the sequence of data being read).

RATIO - Selects ratio function of the DVM.

VOLT - Selects the voltage function of the DVM.

3. The Print Command line can also be interrogated via an SS (Sense Status) instruction. This function appears on the BSY (Busy) line (bit 4). When set to a "1" the DVM will be in the process of a conversion. When set to a "0" the DVM has completed its conversion and data are available to be read.

4. Data can be transferred to the processor via RD (Read Data) instruction. Data are read by five consecutive RD's. The Read Steering flip-flops must be reset before each Data Transfer.

   The order of data transfer is as follows:

   Polarity
   Ten-thousands digit
   Thousands digit
   Hundreds digit
   Tens digit
   Units digit
   Range code

5. The System clear signal will:

   a. Clear address FF,
   b. Clear the ATN FF,
   c. Disable Interrupts,
   d. Set the BSY FF, and
   e. Clear the Read Steering FF's.

Hardware Components

1. The DVM Interface consists of:

1 ea.	NBS DVM	SK-148
2 ea.	I/O Cables (14 pr)	17-002F01
1 ea.	I/O Cables (8pr)	17-037

   Table 2-2 shows the connection pin numbers.

TABLE 2-2. NBS DVM Cabling

Desig	SK-148 MB From	Cinch 15 Pin To	Mating Conn From	Unit To	Command
P15	10-42	P01-1	J01-1	J204-A	
P15	20	2	2	B	
CONVO	30	3	3	E	
RATDC	40-42	4	4	D	
HD10	10-41	P02-1	J02-1	J202-5	100(X1)
HD20	20	2	2	6	(X2)
HD40	30	3	3	30	(X3)
DO80	40	4	4	31	(X4)
TH10	50	5	5	7	1000(X1)
TH20	70	7	7	8	(X2)
TH40	21	9	9	32	(X3)
TH80	41	11	11	33	(X4)
TTH10	60	6	6	9	10,000(X1)
TTH20	11	8	8	10	(X2)
TTH40	31	10	10	34	(X3)
TTH80	51	12	12	35	(X4)
PRINTO	61-41	P02-13	J02-13	48	Print Command
MPOLO	40-40	P03-4	J03-4	11	Sign
UN10	50	5	5	1	1(X1)
UN20	60	6	6	2	(X2)
UN40	70	7	7	26	(X3)
UN80	11	8	8	27	(X4)
TN10	21	9	9	3	10(X1)
TN20	31	10	10	4	(X2)
TN40	41	11	11	28	(X3)
TN80	51	12	12	29	(X4)
RG10	61	13	13	13	Range(X1)
RG20	71	P03-14	14	14	(X2)
GRD	00-40	P03-15	J03-15	25	Reference
GRD	00-41	P02-15	J02-15	25	Reference
GRD	00-42	P01-15	J01-15	25	Reference

2.8.2 Frequency Counter Interface. The counter may be operated in the preset, rate, time or ratio modes by operation of controls on the counter. The interface provides "read into memory" circuitry for the five digits of the display. The "read in" is accomplished by execution of consecutive RD (Read Data) instructions in the program. An "end of count" signal is generated by the counter which is used by the controller to initiate the "read in" sequence or interrupt the processor so that appropriate program strategy may be applied.

Programming Considerations

The following table shows the "Output Command" and status structure of the interface:

Command	Bit	Status
DISABLE INT.	0	
ENABLE INT.	1	
*INITIALIZE	3	
START COUNT	4	** BUSY
STOP COUNT	5	
	6	
	7	

* In the initialize state, the counter is stopped and the display reset.

** The controller recognizes a busy condition during all counter functions.

Hardware Components

The counter interface board may be plugged into any I/O slot in the expansion card file. The strap lead from 214-0 to 114-0 should be removed from the wiring side of the slot chosen. The counter cabling connections are given in Tables 2-3 and 2-4.

TABLE 2-3. Counter Cabling

DIGITAL OUTPUT CONNECTIONS

Desig.	MB Term.	Back Panel Connection		Counter Digital Output Connection
DIU11	10-40	J04-1	P04-1	1
DIU21	20-	-2	-2	2
DIU41	30-	-3	-3	26
DIU81	40-	-4	-4	27
DIT11	50-	-5	-5	3
DIT21	60-	-6	-6	4
DIT41	70-	-7	-7	28
DIT81	11-	-8	-8	29
DIH11	21-	-9	-9	5
DIH21	31-	-10	-10	6
DIH41	41-	-11	-11	30
DIH81	51-	-12	-12	31
DITH11	61-	-13	-13	7
DITH21	71-40	J04-14	P04-14	8
DITH41	10-41	J05-1	P05-1	32
DITH81	20-	-2	-2	33
DITT11	30-	-3	-3	9
DITT21	40-	-4	-4	10
DITT41	50-	-5	-5	34
DITT81	60-	-6	-6	35
EOC1	70-41	J05-7	P05-7	47
GND	00-40	J04-15	P04-15	50

REMOTE RESET CABLE

RST1	11-42	J08	P08	EXT. TEST BNC CONN.

TABLE 2-4. Counter Cabling

REMOTE PRESET CONNECTIONS

Desig.	MB Term.	Back Panel Connection		Remote Preset Connection
U11	10-40	J06-1	P06-1	1
U21	20-	-2	-2	2
U41	30-	-3	-3	26
U81	40-	-4	-4	27
T11	50-	-5	-5	3
T21	60-	-6	-6	4
T41	70-	-7	-7	28
T81	11-	-8	-8	29
H11	21-	-9	-9	5
H21	31-	-10	-10	6
H41	41-	-11	-11	30
H81	51-	-12	-12	31
TH11	61-	-13	-13	7
TH21	71-40	J06-14	P06-14	8
TH41	10-41	J07-1	P07-1	32
TH81	20-	-2	-2	33
TT11	30-	-3	-3	9
TT21	40-	-4	-4	10
TT41	50-	-5	-5	34
TT81	60-	-6	-6	35
P10	70-	-7	-7	44
GND	00-41	J07-15	P05-15	50

2.8.3 Relay Interface. This interface provides computer control of 32 relays via two Control Line Modules. The relays to be controlled consist of 20 single-pole normally open contacts and 12 double-pole normally open contacts. The normally closed contacts of all relays are available on a provided cable.

A 32-bit memory location is loaded with the condition of the 32 relay contacts. For normally open operation, a "one" bit in its associated memory bit location will close the contact. The "memory image" of the relay contacts is updated via the program. A "write data" instruction must be issued for each set of eight relays. Therefore, four consecutive WD (Write Data) instructions will update the 32 contacts. An "output command" instruction to one of the Control Line Modules will open all 16 contacts associated with the module. Where normally closed contacts are used, the OC (Output Command) instruction will leave the contact in a closed state. Since 16 relays are controlled by each Control Line Module, two consecutive device numbers have been chosen for the two modules. See tables 2-5 and 2-6.

Mechanical Considerations

The relays are located in a special unit mounted in the system cabinet. Cables connect the control lines to the relay chassis. The relay chassis contains its own power supply (+24V) for coil voltage and the relays are plugged into sockets for ease of replacement. The contact lines are brought to the back panel of the cabinet to 15 pin connectors. (See table 2-5).

The Control Line Modules may be placed in any two adjacent I/O slots. The device numbers chosen for the two modules are X'71' and X'72'.

2.8.4 Oscillator Interface. The Oscillator Controller is a special controller used exclusively with the oscillator for converting four data bytes to the necessary signals to control amplitude and frequency output of the oscillator. The controller uses a general purpose Input/Output Mother Board SK-174, and a general purpose Mother Board SK-175.

The Controller provides 31 data output lines and one clear line to the oscillator. The 31 data lines each have a storage flip-flop which will hold the data for 15ms after the fourth "write data" instruction. The clear line operates in the same manner if cleared by data configuration.

Also provided are one active status line which is active for 15ms following the fourth data byte, and one priority interrupt line which will be activated by an illegal code only.

TABLE 2-5. Relay Banks' Pin Connectors

Relay	Swinger	Norm. Open	Norm. Closed	Back Panel Connection	Pin No. Swinger	Pin No. NO	Pin No. NC
R1	DPT1	NO1	NC1	J(A)	1	2	3
	DPT2	NO2	NC2		6	5	4
R2	DPT1	NO1	NC1		7	8	9
	DPT2	NO2	NC2		12	11	10
R3	DPT1	NO1	NC1		13	14	15
	DPT2	NO2	NC2		18	17	16
R4	DPT1	NO1	NC1		19	20	21
	DPT2	NO2	NC2		24	23	22
R5	DPT1	NO1	NC1		25	26	27
	DPT2	NO2	NC2		30	29	28
R6	DPT1	NO1	NC1		31	32	33
	DPT2	NO2	NC2		36	35	34
R7	DPT1	NO1	NC1		37	38	39
	DPT2	NO2	NC2		42	41	40
R8	DPT1	NO1	NC1		43	44	45
	DPT2	NO2	NC2		48	47	46
R9	DPT1	NO1	NC1	J(A)	49	1J(B)	2J(B)
	DPT2	NO2	NC2	J(B)	5	4J(B)	3J(B)
R10	DPT1	NO1	NC1		6	7	8
	DPT2	NO2	NC2		11	10	9
R11	DPT1	NO1	NC1		12	13	14
	DPT2	NO2	NC2		17	16	15
R12	DPT1	NO1	NC1		18	19	20
	DPT2	NO2	NC2		23	22	21
R13	SPT1	NO1	NC1		38	40	39
R14	SPT1	NO1	NC1		41	43	42
R15	SPT1	NO1	NC1		44	46	45
R16	SPT1	NO1	NC1	J(B)	47	49	48
R17	SPT1	NO1	NC1	J(C)	1	3	2
R18	SPT1	NO1	NC1		4	6	5
R19	SPT1	NO1	NC1		7	9	8
R20	SPT1	NO1	NC1		10	12	11
R21	SPT1	NO1	NC1		13	15	14
R22	SPT1	NO1	NC1		16	18	17
R23	SPT1	NO1	NC1		19	21	20
R24	SPT1	NO1	NC1		22	24	23
R25	SPT1	NO1	NC1		25	27	26
R26	SPT1	NO1	NC1		28	30	29
R27	SPT1	NO1	NC1		31	33	32
R28	SPT1	NO1	NC1		34	36	35
R29	SPT1	NO1	NC1		37	39	38
R30	SPT1	NO1	NC1		40	42	41
R31	SPT1	NO1	NC1		43	45	44
R32	SPT1	NO1	NC1	J(C)	46	48	47

Note: Pin Number 50 of J(A), J(B), and J(C) is ground.

TABLE 2-6. Relay Banks 1 and 2 Coding Format

CONTROL LINE	RELAY		BIT
Device X'71'			
CL000	R1	DPDT	0
CL010	R2		1
CL020	R3		2
CL030	R4		3
CL040	R5		4
CL050	R6		5
CL060	R7		6
CL070	R8		7
CL080	R9		8
CL090	R10		9
CL100	R11		10
CL110	R12		11
CL120	R13	SPDT	12
CL130	R14		13
CL140	R15		14
CL150	R16		15
Device X'72'			
CL000	R17	SPDT	0
CL010	R18		1
CL020	R19		2
CL030	R20		3
CL040	R21		4
CL050	R22		5
CL060	R23		6
CL070	R24		7
CL080	R25		8
CL090	R26		9
CL100	R27		10
CL110	R28		11
CL120	R29		12
CL130	R30		13
CL140	R31		14
CL150	R32		15

TABLE 2-7. Oscillator Cabling

Signal	MB Location	Back Panel Location		Program Input Connection
OSCLRA	50-42	P8-5	J8-5	2-7-12-17-27-32-37
OPE 311	40-	-4	-4	38
301	30-	-3	-3	39
291	20-	-2	-2	40
281	10-42	P8-1	J8-1	41
271	71-41	P7-14	J7-14	33
261	61-	-13	-13	34
251	51-	-12	-12	35
241	41-	-11	-11	36
231	31-	-10	-10	28
221	21-	-9	-9	29
211	11-	-8	-8	30
201	70-	-7	-7	31
191	60-	-6	-6	18
181	50-	-5	-5	19
171	40-	-4	-4	20
161	30-	-3	-3	21
151	20-	-2	-2	13
141	10-41	P7-1	J7-1	14
131	71-40	P6-14	J6-14	15
121	61-	-13	-13	16
111	51-	-12	-12	8
101	41-	-11	-11	9
091	31-	-10	-10	10
081	21-	-9	-9	11
071	11-40	-8	-8	3
061	70-	-7	-7	4
051	60-	-6	-6	5
041	50-	-5	-5	6
031	40-	-4	-4	24
021	30-	-3	-3	23
OPE 011	20-40	-2	-2	22
Ground	81-40	P6-15	J6-15	1
Ground	81-41	P7-15	J7-15	1
Ground	81-42	P8-15	J8-15	1

Oscillator Interface Specifications

Output Signals

Logical zero is -13.5V ±2V at 0.0ma.
Logical one is 0V ±0.5V at 4ma.

Input Signals

Logical zero is 0V ±0.5V at 1.2ma.
Logical one is 5V ±2V at 0.0ma.

Programming Notes

Any command to the controller will clear the controller resulting in no change in device output. Data are transferred to the device via "write data" or write block instruction of four bytes.

First Half-Byte (least significant four bits of first-byte) This half-byte is variable from X'0' to X'9' to control the oscillator amplitude in steps of 0.01V rms per step. X'0' = no output. X'1' = 0.01V rms output. X'9' = 0.09V rms output. Decodes from this half-byte of X'A' through X'F' are illegal and will result in an interrupt being generated. (See byte four most significant bit for exception).

Second Half-Byte (most significant four bits of first-byte) Used to control the amplitude in steps of 0.1V rms per step. X'0' = no output, X'1' = 0.1V output, and X'9' = 0.9V output. Output is additive to first half-byte. X'A' through X'F' have the same properties as described in first half-byte.

Third Half-Byte (least significant four bit of second-byte) Used to control the output in steps of 1.0V. X'0' = 0V output. X'9' = 9V output. Output is additive to the first byte X'A' through X'F', as previously described.

Fourth Half-Byte (most significant four bits of second-byte) Used to control the output frequency in steps of 0.1 Hz per step from X'0' to X'9'. X'A' through X'F', as previously described.

Fifth Half-Byte (least significant four bits of third-byte) Controls output frequency in steps of 1.0 Hz per step as described above.

Sixth Half-Byte (most significant four bits of third-byte) Controls output frequency in steps of 10.0 Hz per step.

Seventh Half-Byte (least significant four bits of fourth-byte) Controls output frequency in steps of 100.0 Hz per step.

Eighth Half-Byte (most significant four bits of fourth-byte) Controls the multiplication factor of the last four half-bytes. X'1' = multiplication factor of 1.0. X'2' = multiplication factor of 10.0. X'4' = multiplication factor of 100.0. One of these numbers must be used for correct operation.

Any number less than X'8' and not specified here is illegal and will result in an interrupt being generated. If this byte is $\geq$X'8' (most significant bit = one), none of the illegal codes will be recognized, and the oscillator will be cleared resulting in no output. No interrupt will be generated. This bit may, therefore, be used as a wait bit or clear bit. The device will also be cleared by the console initiate. The device may have an address of X'00' through X'FF' by use of the address strap option.

Hardware Components

1 ea.	Oscillator Controller Module	Part SK-174
1 ea.	Oscillator Controller Module	Part SK-175
3 ea.	General Purpose I/O Cables	Part 17-002

Table 2-7 shows the oscillator cabling.

2.8.5 X-Y Plotter Interface. The SK-149 X-Y plotter interface is designed to interface to the X-Y plotter. The interface has the ability, under program control to:

1. Output +10 volts to -10 volts coordinates on the X and Y inputs to the plotter
2. Output Y coordinates, only for the time Y mode
3. Output both X and Y coordinates simultaneously in the X-Y mode
4. Trigger the trace in the time Y mode, and
5. Control the Pen-up/Pen-down function.

The interface also contains a priority interrupt circuit which, if armed, can interrupt the processor for service. Two interrupt functions are provided:

1. Internal clock interrupt which can generate an interrupt approximately every 20ms. This interrupt is under control of the ARM function.
2. An external interrupt line which will interrupt the processor on the transision from the "0" state (0V ±0.5V, 1.2ma) to the "one" state (5V ±0.5V).

This interrupt line is not under control of the ARM function; but the function should be put into the DISARM state to prevent multiple interrupts from the internal clock.

Program Notes

1. Strap options provide for the selection of any device address from X'00' to X'FF'.

2. Control functions are executed via OC (Output Command) instructions. The command structure is as follows:

Bit	0	1	2	3	4	5	6	7
Command	TIME	X-Y	TRG	INIT	DIS	ARM	DOWN	UP

TIME - Puts the controller in the Time-Y mode.
X-Y - Puts the controller in the X-Y mode.
TRG - Generates a Trigger Command to the plotter for the Time-Y mode.
INIT - Initializes the controller: lifts pen; disarms interrupts; clears the write steering FF; clears the ATN FF; puts controller in the T-Y plot mode.
DIS - Disarms internal clock interrupt.
ARM - Arms internal clock interrupt.
DOWN - Lowers the pen to the paper.
UP - Raises the pen from the paper.

3. The X-Y coordinate are controlled by WD (Write Data) instructions. Four consecutive WD instructions (initially the data steering FF must be cleared) are necessary to update the X-Y coordinate:

```
WD    DEV, Y DATA
WD    DEV, Y DATA +1
WD    DEV, X DATA
WD    DEV, X DATA +1
```

The DATA layout is as follows:

	0	1	2	3	4	5	6	7
Y/X DATA	SIGN	MSD →						
	0	1	2	3	4	5	6	7
Y/X DATA + 1	→	LSD						

In the Time-Y mode only two WD's are necessary per point. The ten bit resolution provides plotting steps of 19 millivolts. The DAC is accurate to within the least significant bit.

4. The time interval between the output of points or the pen state changes must be facilitated with the interrupt. For example: the maximum tracking velocity of the plotter is 15 inches per second, or approximately 70 milliseconds per inch. Using the 20-ms interrupt, the pen can traverse approximately 2/7 of an inch per interrupt. If it is desired to plot a line two inches long, it will take eight (7+1, for safety factor) interrupts before the line is completed.

For the pen to go from the down state to the up state, two interrupts are needed for completion. For the pen to go from the up state to the down state, four interrupts are needed.

Hardware Components

1. The SK-149 interface module consists of:

1 ea.	X-Y Plotter Interface Board	SK-149
1 ea.	10 Bit Dual Chan DAC	35-074F04
1 ea.	Cable	17-037
2 ea.	Coax Cables	17-055F01

2. The two boards may be inserted into any two available adjacent I/O slots in the expansion card file, with the DAC to the left (facing front) of the controller board. Rack/Tack strap must be removed from these slots.

3. On the DAC board the upper coaxial connector is the X-Channel and the lower is the Y-Channel.

4. X-Y Plotter Cabling

From	To
J9-8	J102A
J9-9	B
J9-11	C
J9-15	D

2.9 Calibration of the System

For calibration of the ratio readout system, plugs connecting points 3-4, 5-6, 11-12, 17-41, 18-42 are removed. Plugs and patch cords for points 41-8, 42-4, 5-12, and 36-53 are connected. This will give a circuit shown in figure 2-9. Cable 36 from point J and cable 8 from point I of the junction box are disconnected. Switch the oscillator to manual and open switch S1.

The following procedure has proven reliable for calibration purposes. Set the oscillator for 100 Hz, 5 V rms output. Adjust the voltage divider for 70 mV rms on the input to the test converter (meter 1). Using some transfer voltmeter such as a high quality differential voltmeter, check the accuracy of the test converter (1 V range) by switching the DVM to the voltage mode. If the ac/dc converter is in error, adjustments can be made by adjusting R37, R35, C18, and C23 as explained in its instruction manual. After the 100 Hz point is checked, switch the oscillator to 10,000 Hz and check the accuracy at this frequency. High and low frequency adjustments can be

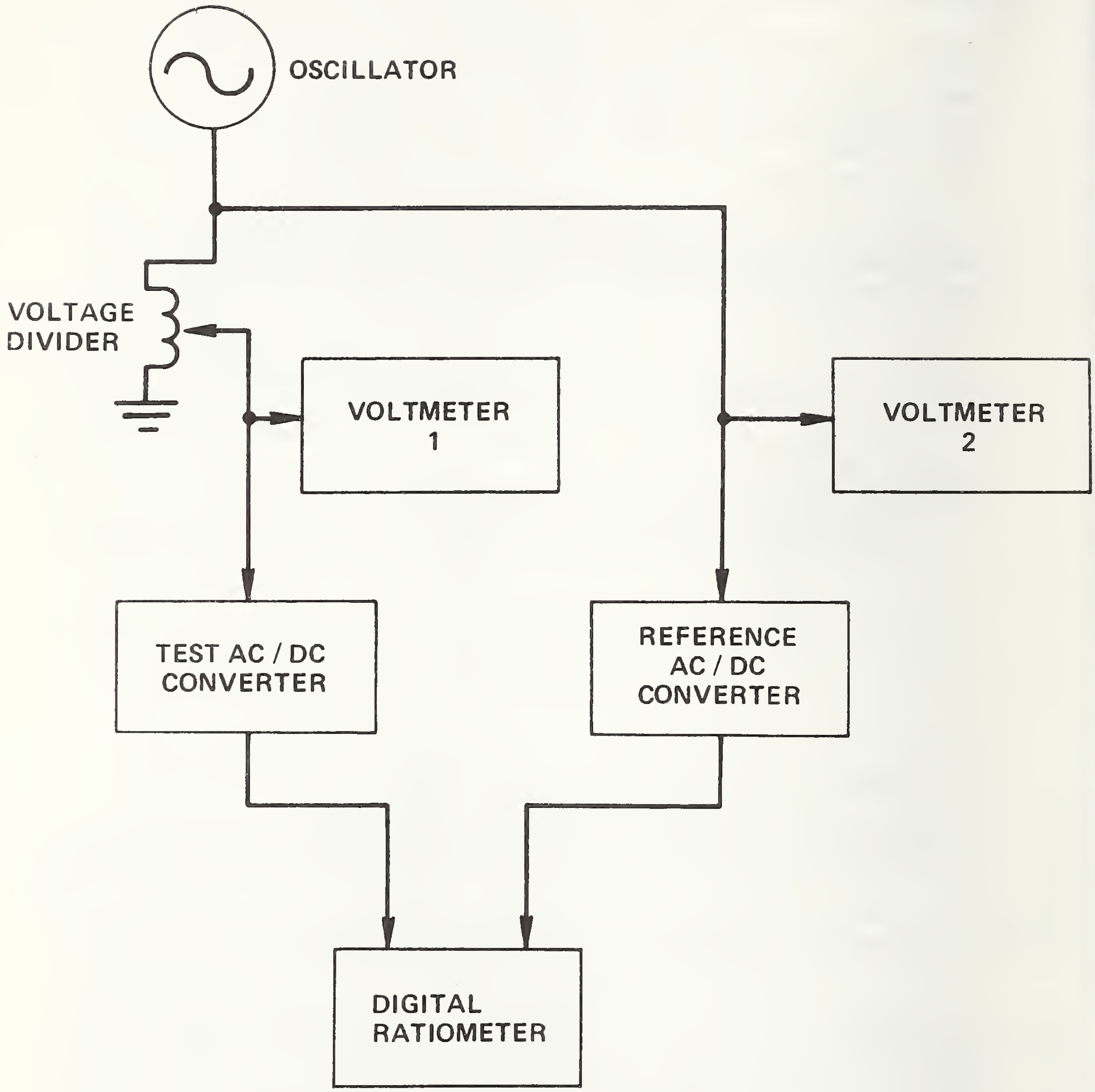


FIGURE 2-9. CALIBRATION CIRCUIT SCHEMATIC

made on the ac/dc converter. After this adjustment, switch back again to the 100 Hz point and recheck the accuracy. Switching back and forth between low and high frequency points several times may be necessary as there is some interaction between the two adjustments. After the accuracy has been established at the two extremes, a check should be made at the midpoints and at 10 Hz, 30 Hz, and 50 Hz to ensure accuracy over the entire frequency range. In case adjustment cannot bring the ac/dc converter into 0.1 percent of reading accuracy with these adjustments, converter performance is no longer acceptable and it will be necessary to repair the converter. The errors in the converters are the limiting factors in the overall accuracy of the system.

Now switch to the ratio mode and compare the DVM ratio with the voltage divider setting. Do the same kind of adjustment on the reference converter (10 V range) at 100 Hz and 10,000 Hz to bring the voltage divider settings and the DVM ratio to 0.1 percent agreement.

In addition to the 70 mV rms on the input to the test converter, the accuracy of the system should also be checked at levels from 14 mV to 1 V rms to ensure amplitude linearity. The 14 mV rms lower level is established for a 10 mV/g accelerometer. For a 2 g calibration, the rms voltage will be approximately 14 mV rms. Now the system should be in calibration on the 1 V rms range. Because the 10-V rms range on the signal converter is used occasionally, the 10-V range should be checked also using the same procedure as given above for the 1-V rms range. This calibration need not be done above 4 V rms because for a voltage greater than approximately 4 V rms the inverse circuit (circuit b) is used.

From a user's standpoint, a calibration once a month is good practice. Time involved for a calibration is usually about 20 minutes.

Also, periodically, the input from the standard accelerometer is connected to both the reference and the test input and a calibration run by computer control. The sensitivities typed out should agree with the calibration factors of the standard accelerometer at the test frequencies.

3. SOFTWARE

3.1 Computer Capability

The minicomputer has 16 general purpose registers, each 16 bits in length. These registers are available for use as accumulators and all sixteen can be used as index registers. The computer has high speed arithmetic hardware options of fixed point Multiply/Divide and Read Block/Write Block capabilities.

Two methods of programming I/O (Input/Output) are available:

1. Interrupt - Controlled by priority level as determined by relative plug position of device mother boards, or

2. Sense - status basis.

Most of the programming for the I/O in the automated system is of the sense-status type since great speed is not required and the software is somewhat easier to write for this type. In this method the device to be serviced is interrogated for a device busy bit and the program continues in this sense-status loop until the device is available as shown in figure 3-1. In the interrupt method, the computer can be interrupted to service a device instead of waiting in a sense status loop.

In the present program, all coding has been in machine language using hexidecimal notation. This makes for efficient use of core space and for ease in programming I/O instructions.

3.2 Software Philosophy

The basic design philosophy in the automated accelerometer calibration system was twofold; first, to provide an accurate and reliable system for accelerometer calibration utilizing state-of-the-art techniques and equipment. Speed of calibration is of some importance since sufficient data for statistical analysis are desirable. However, accuracy must not be sacrificed for the sake of a quick calibration. Secondly, the design should make use of techniques in such a manner that the software of the system assumes the responsibility for control and decisions as much as possible, leaving the operator of the system free for other tasks. The operation of running the system can be carried out at the technician level. The operator technician should be able to make small changes in the system such as changes in the DATA BLOCK described in the next section. He should be able to reload the core from the magnetic tape, and make use of the HALT and STOP features described below. But, he should not have to make major changes in the operating programs. The software should be flexible enough to accommodate any accelerometers that may need calibration.

The quality of data collected may vary with the accelerometer under test. Variations in accelerometer and signal conditioning circuits may result in varying noise conditions and settling times. In the software for this system, this is taken into account and the quantity of data collected will vary from one accelerometer to another. Details of this procedure are found in the Multiple Readings (Section 4.38) program.

3.3 Data Block

The program has been organized about the current test point as found in the Data Block (core location 1400). This data block consists of control constants for all the test points to be performed. Each test point occupies 6 bytes. The organization of this table is shown in figure 3-2. The first four bytes contain the frequency data. The coding for the four bytes to program the frequency is as follows:

0		1		2		3	
	100 Hz	10 Hz	1 Hz	0.1 Hz			Multiplier

Example: 0010 0001 is the code for 10 Hz. The last digit is the multiplier which is either 1 or 2. Example: 0998 2002 is the code for 9982 Hz.

Bytes 4 and 5 shown in figure 3-2give the oscillator voltage necessary for this test point in millivolts for exciter 2. This voltage is trimmed each time the program goes through the Acceleration Level Set program. The new voltage value will then replace the old one in this table. Bytes 6 and 7 are the same information for exciter 1. Bytes 8 and 9 give the capacitor coding to set the relay bank 2 for the proper impedance matching network (see Section 4.15.1) for exciter 2. Bytes C and D are the same information for exciter 1. Bytes A and B and E and F give the desired millivolt output from the standard accelerometer for exciter 2 and 1 respectively. These values may easily be changed by use of the monitor to obtain different acceleration levels. Several sets of Data Block constants are saved on paper tape for convenience in quick changeover of desired acceleration levels.

If the total number of test points is to be changed, the contents of the following core locations must be changed. Location 0E8E contains a constant for exciter 2 and location 0E90 contains a constant for exciter 1. The constant must be four times the number of test points. The constant must be in hexidecimal notation. In the program given in this report, the exciter 1 constant is 0064 (25 test points) and for exciter 2 the constant is 0094 (37 test points).

3.4 Start 1, Start 2

The program for automatic calibration of accelerometers starts with either the Start 1 or Start 2 programs. These correspond to exciter 1 or exciter 2. The summary flow chart for the automated system is shown in figure 3-3. The read Ratio I and Ratio II are handled by the Multiple Readings and Digital Filter Subroutine (see Section 4.39) which regulates the number of readings based upon the scatter in the data. The individual programs are described in the listing at the end of this software section. The more complicated programs have flow charts accompaning them.

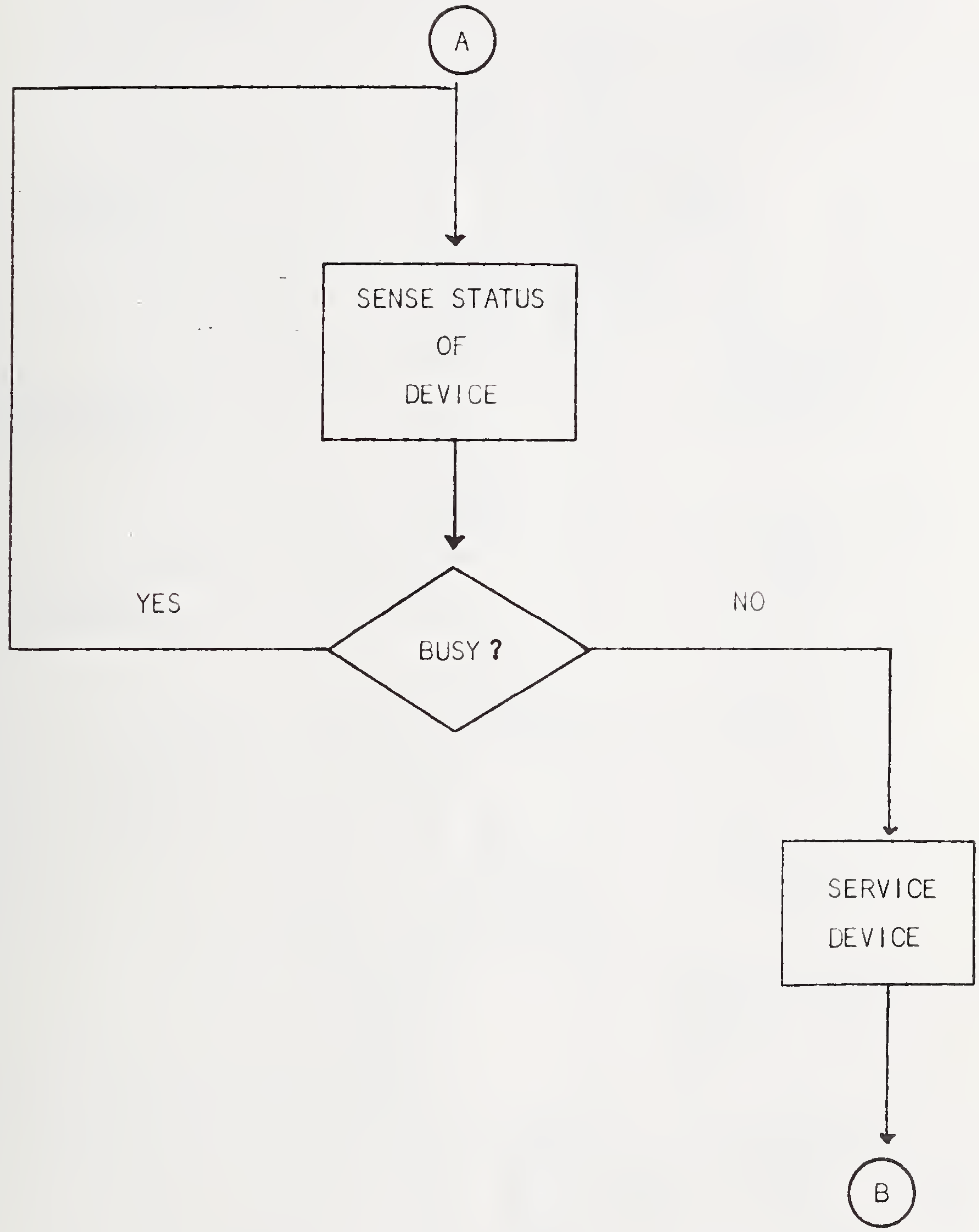


FIGURE 3-1. Sense Status Loop.

Relative byte addresses:		2	4	6	8	A	C	E
1400 Test Pt. 1	First Test Frequency		Exciter 2 Osc. Voltage	Exciter 1 Osc. Voltage	Exciter 2 Capacitor Code	Exciter 2 Desired Voltage	Exciter 1 Capacitor Code	Exciter 1 Desired Voltage
1410 Test Pt. 2		Second Test Frequency						
. . . . .		. . .						
1640 Test Pt. -								

FIGURE 3-2. Data Block Format.

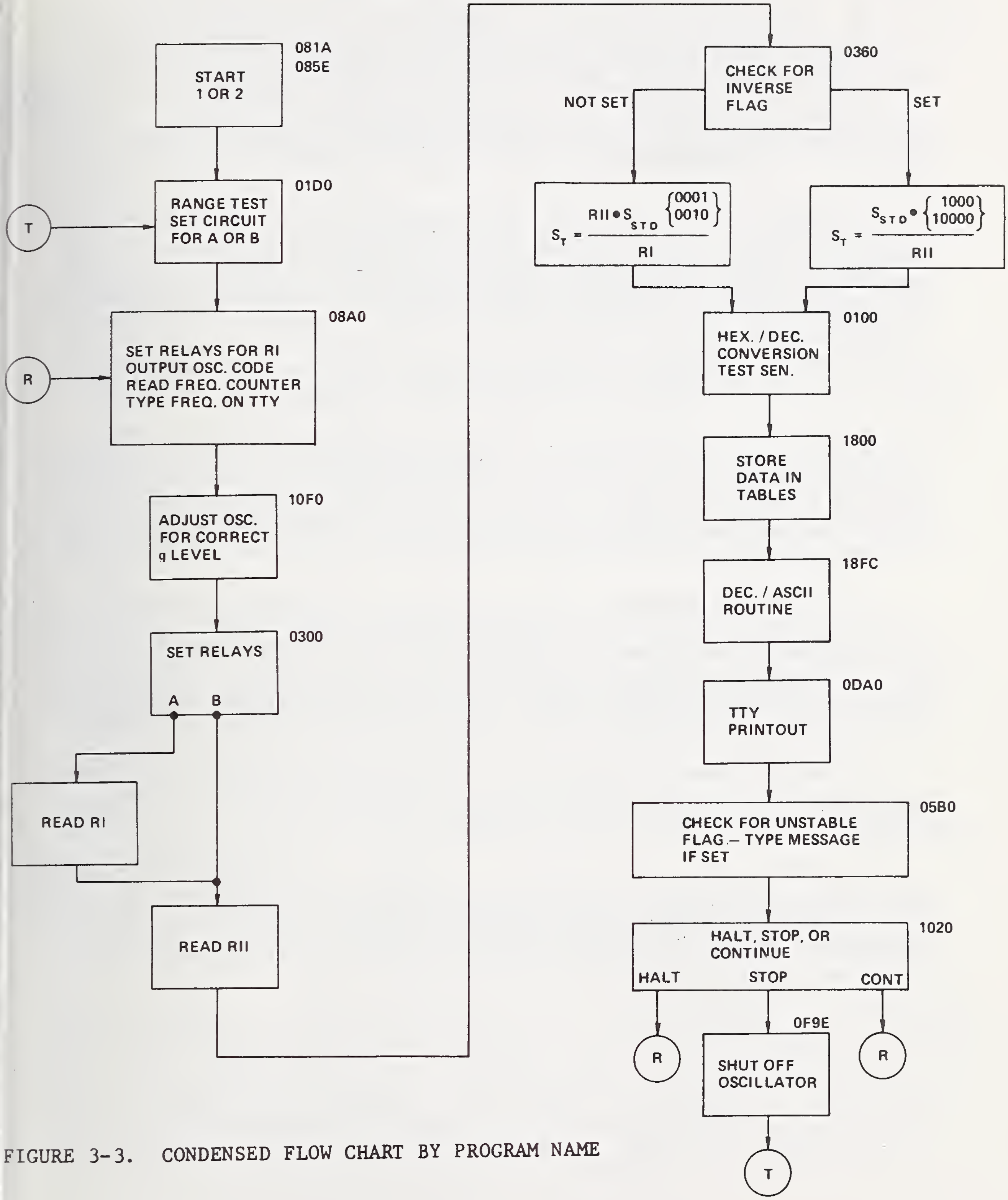


FIGURE 3-3. CONDENSED FLOW CHART BY PROGRAM NAME

3.5 Halt and Stop Features

The HALT feature permits the operator to type an H character on the TTY after a test frequency has been typed by computer control. This activates a program which, after the completion of the current test point, will cause the program to halt but leave the exciter energized. The computer will continue in the HALT mode until the execute button is pushed on the computer console whereupon the program will continue with the next test point.

The STOP feature is similar in that when the S character is typed the computer will halt after completion of the current test point. In this case however, the program will start over with the first test point when the execute button is pressed. Also, an extra line feed is programmed to separate the two sets of data.

3.6 Provisions for Automatic Selection of Circuits and Automatic Ranging of Test Equipment

The voltage ratio circuit can be either circuit a or circuit b as explained in the hardware section. The software selects one of these two circuits to obtain voltage ratio data. The program which selects this circuit is the Check For Inverse Flag and Set Relays. The circuit chosen is based upon the test accelerometer sensitivity as explained in Section 2.

The automatic ranging of the ac/dc voltage converters is also selected by the software. Only the test ac/dc converter range needs to be automatic. The range is normally one volt full scale but switched to ten volts full scale, if needed. The range selection is accomplished by a relay either open or closed (See Section 4.38). The reference converter is always at 10 volts full scale and the DVM is always at 10 volts full scale range.

The ac/dc converters mode (Slow or Fast) is also set by the software. For frequencies greater than 100 Hz, they operate in the Fast mode; otherwise, they operate in the Slow mode.

The DVM function (either ratio or voltage) is selected by the software. The voltage mode is used to monitor accelerometer voltage levels for setting the acceleration levels, whereas the ratio mode is used for all other data collection. The function selection is provided as a part of the DVM interface package (see Section 2.8.1).

4. SOFTWARE PROGRAM DESCRIPTIONS

This section is a listing of programs by core location sequence. A description of each program and a flow chart for the more complicated ones precedes the program listing. The hexidecimal core contents printout was prepared by use of the software monitor. This monitor was developed at NBS for the purpose of programming the subject minicomputer. It occupies approximately 640 bytes of core and is used to enter hexidecimal code directly into core locations. It is also used to punch paper tapes of core locations, read paper tapes, and to type out blocks of core on the TTY.

Figure 3-3 shows a condensed flow chart by program name and location. Figure 3-4* shows a detailed flow chart by type of operation.

The software for the automated system was written in machine language code. At the time this project was started, the higher order languages were not developed for small computers to the extent to make it a practical approach to software. In the present generation of minicomputers, the higher order languages of Fortran and Basic are more fully developed for these machines. In planning for new systems, use of the higher order languages may be practical. However, the system designer should be prepared to invest in larger amounts of memory to handle the higher order languages.

One of the most frequent criticisms of real-time systems using machine language code for programming is poor documentation. Programs written by one person are often quite difficult to understand and modify by another person. For this reason the entire software package is included in this report with documentation for easier understanding of the machine language coding.

The following FLAGS are used throughout the software.

EXCITER (SHAKER) FLAG: This flag tells which exciter is being used for a test. The program must know this in order to know which data to use in the DATA BLOCK since each exciter requires different programming. It is also needed when the test sensitivity is calculated since each exciter has a different calibration for the standard accelerometer sensitivity.

INVERSE FLAG: This flag tells which circuit to use: a or b. For test accelerometers with high sensitivities, circuit b must be used to calibrate the accelerometer.

THRU FLAG: This flag tells the program if the program has cycled "thru" once to check for overranging on the test ac/dc converter. If program has checked for the overrange condition, it does not recheck it again.

LOW/NORM FLAG: If the sensitivity of the test accelerometer is so low that only four digits of significant data can be read (first digit zero on DVM), this flag is set to indicate a LOW signal from the accelerometer. This is used in the computation of the test sensitivity to extend the printout to one extra digit. For example: 9.954 instead of 9.95 mV/g. This feature can be used to regulate the number of digits typed out by adjusting the voltage divider to give only four significant digits on the DVM at the point where an extra digit of typeout is desired.

* This figure is at the end of the report.

STABLE/UNSTABLE FLAG: This flag is set if the data do not meet the program requirements of the Multiple Readings program (see Section 4.39). It indicates the test accelerometer does not meet the stability requirements in this program. In the typeout, it triggers a diagnostic message to be typed beside the sensitivity typeout.

RANGE FLAG: This flag is set for 0001 for a range of one volt on the test ac/dc converter or 0010 for a range of 10 volts on the test ac/dc converter. The range of the converter is needed in the computation of the test sensitivity.

4.1 Hexidecimal-to-Decimal Conversion for Test Sensitivity

This routine takes the test sensitivity in hexidecimal and converts it to decimal for type out. It also calculates where to place the decimal point. Three parameters calculated here are used in the Store Data program. These are:

1. Total number of digits to be typed ≡C1 (01A8)
2. Number of digits typed before decimal point ≡C2 (01A6)
3. C1-C2 = C3 (09E0)

Hexidecimal-to-Decimal Conversion for Test Sensitivity; Set Typeout Constants.

```
0100 4030 09FA     SAVE TEST SENSITIVITY                IN ON R3
                                                        OUT ON RA, RB,
0104 40E0 01A4     SAVE RE                                     RC, RD,
                                                               RE
0108 4200 0000     NOP

010C C890 2710     LOAD CONSTANTS FOR CONVERSION

0110 C880 03E8

0114 C850 0064

0118 C870 000A

011C 0E22          CLEAR R2 FOR DIVISION USE

011E 0200          NOP

0120 0D29          DIVIDE HEX. NO. BY FIRST CONST.

0122 0200          NOP
```

```
0124 08A3            PUT QUOT. IN RA  (FIRST DECIMAL DIGIT)
0126 0200            NOP
0128 0832            PUT REMAINDER IN R3
012A 0B22            CLEAR R2
012C 0D28            DIVIDE BY SECOND CONST.
012E 0200            NOP
0130 08B3            PUT QUOT. IN RB  (2 ND. DECIMAL DIGIT)
0132 0200            NOP
0134 0832            PUT REMAINDER IN R3
0136 0B22            CLEAR R2
0138 4200 0000       NOP
013C 0D25            DIVIDE REMAINDER BY 3 RD CONST.
013E 0200            NOP
0140 08C3            PUT QUOT. IN RC  (3 RD. DECIMAL DIGIT)
0142 0200            NOP
0144 0832            PUT REMAINDER IN R3
0146 0B22            CLEAR R2
0148 0D27            DIVIDE REMAINDER BY 4 TH. CONST.
014A 0200            NOP
014C 08D3            PUT QUOT. IN RD  ( 4 TH DECIMAL DIGIT)
014E 0200            NOP
0150 08E2            PUT REMAINDER IN RE   ( 5 TH DECIMAL DIGIT)
0152 0B44            CLEAR R4
0154 4540 09F6       WAS THE FIRST DIGIT IN RII ZERO?
0158 4330 0168       IF SO GO TO 0168
```

```
015C C820 0003    IF NOT, LOAD 3 INTO R2
0160 4020 01A6    STORE IN STORAGE LOC.  (# DIGITS PRINTED BEFORE DE
                                                                  PT
0164 4300 0170    BRANCH
0168 C820 0002    LOAD 2 INTO R2
016C 4020 01A6    STORE  (# DIGITS PRINTED BEFORE DEC. PT.)
0170 4200 0000    NOP
0174 4200 0000    NOP
0178 C5A0 0000    IS FIRST DECIMAL DIGIT ZERO ?
017C 4330 018C    IF SO BRANCH
0180 C820 0004    IF NOT LOAD 4 INTO R2
0184 4020 01A8    STORE THIS NUMBER  ( # DIGITS TO BE PRINTED OUT)
0188 4300 0198    BRANCH
018C C820 0005    LOAD 5 INTO R2
0190 4020 01A8    STORE THIS NUMBER  ( # DIGITS TO BE PRINTED OUT)
0194 4200 0000    NOP
0198 40E0 01AA    STORE 5 TH. DECIMAL DIGIT FOR LATER USE
019C 4830 01A6     LOAD # DIGITS BEFORE DECIMAL PT. CODE
01A0 4300 01AC    BRANCH
01A4 00B6         STORAGE LOC. (NOT USED)
01A6 0005         STORAGE LOC.  ( # DIGITS BEFORE DEC. PT.)
01A8 0005         STORAGE LOC. ( # OF DIGITS PRINTED OUT)
01AA 0005         STORAGE FOR FITH DECIMAL DIGIT
01AC 4880 09EE    LOAD INVERSE RATIO FLAG
01B0 C580 0000    NOT SET?
01B 4 4330 01C0   IF NOT SET, BRANCH TO CONTINUE
```

```
01B8 C830 0005     IT IS SET, LOAD 5 INTO R3
01BC 4030 01A6     STORE THIS NUMBER (#DIGITS BEFORE DEC. PT.)
01C0 4820 01A8     LOAD CODE FOR # DIGITS TO BE PRINTED OUT
01C4 0B23          SUBTRACT THESE 2 NUMBERS
01C6 0200          NOP
01C8 4020 09E0     STORE THIS NUMBER (USED IN STORE DATA IN TABLES)
01CC 4300 1800     GO TO STORE DATA IN TABLES PROGRAM
>
```

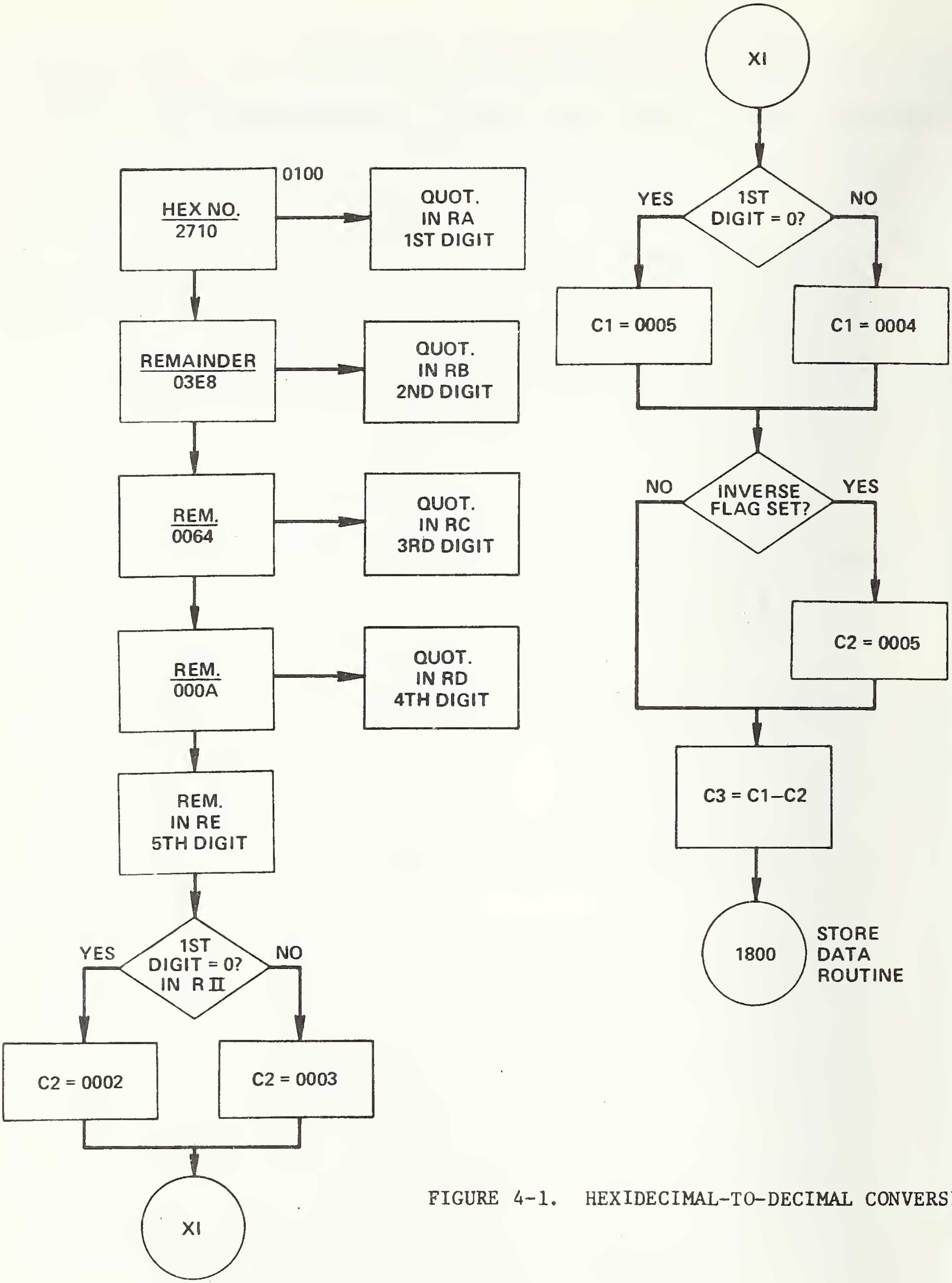


FIGURE 4-1. HEXIDECIMAL-TO-DECIMAL CONVERSION

4.2 Range Test for Test Accelerometer

This routine activates the exciter for 2000 Hz at 2 g, reads the test accelerometer voltage, and compares this reading to an upper limit. For test voltages below this upper limit the INVERSE FLAG is reset (0000 is stored in core location 09EE), and for test voltages above this limit, the INVERSE FLAG is set (0001 in location 09EE). The INVERSE FLAG will be used later on to determine which ratio circuit will be used for the voltage ratio reading of standard-to-test accelerometer output. The upper limit voltage is set to get the maximum accuracy from the voltage ratio circuit. For the present circuit configuration, the upper limit is 800 mV rms for 2 g peak acceleration at 2000 Hz.

Range Test for Test Accelerometer

```
01D0 C850 0072     LOAD RELAY BANK 2 DEVICE #
01D4 DA50 1408     SET CAPITANCE AT ZERO
01D8 DA50 1409
01DC C870 000F     LOAD OSCILLATOR DEVICE #
01E0 C860 0C50     SET INDEX REG. FOR  F=2000 HZ.
01E4 4190 16C0     BAL TO SET UP OSC. CODE
01E8 9D75          SENSE STATUS OF OSC.
01EA 0552          BUSY?
01EC 4230 01E8     IF SO SS AGAIN
01F0 DA76 0C8C     WRITE DATA TO OSC.
01F4 CA60 0001     ADD 1 TO INDEX
01F8 C560 0054     HAVE 4 BYTES BEEN WRITTEN?
01FC 4230 01E8     IF NOT, WRITE AGAIN
0200 4200 0000     NOP
0204 C8D0 0004     LOAD 4 INTO RD
0208 C830 00CA     LOAD DVM DEVICE ADDRESS
```

```
020C DE30 0B24     ENABLE INTERUPTS FOR DVM
0210 9F44          CLEAR INTERUPTS
0212 0200          NOP
0214 C8C0 0002     LOAD VOLTAGE CODE
0218 9F3C          SET DVM FOR VOLTAGE MODE
021A 0200          NOP
021C C8A0 0071     LOAD RELAY BANK I DEVICE ADDRESS
0220 C840 0000     LOAD I VOLT RANGE CODE FOR SIG. CONVERTER
0224 C850 00F2     LOAD CODE FOR RATIO I, FAST MODE
0228 9AA5          SET RELAYS
022A 9AA4            "     "
022C 4190 0E78     DELAY
0230 0896           LOAD INDEX INTO R9
0232 0C8[illegible]           MULT INDEX BY 4, STORE IN R9
0234 4090 09C4      STORE CAPACITANCE INDEX IN CORE
0238 4140 10F0      BAL TO CONST. C SUB.
023C C830 00CA      RESTORE DVM DEVICE ADDRESS
0240 C800 0071      LOAD RELAY BAND I DEVICE ADDRESS
0244 C840 0004      LOAD 10 VOLT RANGE CODE
0248 C850 00C2      LOAD RATIO II CODE
024C 9A02           SET RELAYS
024E 9A04             "     "
0250 4190 0F1E      BAL TO MULTIPLE READINGS OF DVM SUB.
0254 4200 0000      NOP
```

```
0258 4200 0000      NOP
025C 9F95           ACKNOWLEDGE INTERUPTS
025E 0539           IS IT THE DVM?
0260 4230 025C      IF NOT LOOK AGAIN
0264 0B 77          CLEAR R7
0266 0200           NOP
0268 C890 0005      LOAD LIMIT INTO R9
026C C880 0001      LOAD INCREMENT INTO R8
0270 DB37 02B8      READ DATA INTO CORE
0274 4120 1010      DELAY
0278 C170 0270      BXLE (LOOP)
027C 4200 0000      NOP

0280 4190 0500      BAL TO DECODE DVM DATA
0284 4200 0000      NOP
0288 4820 0554      LOAD 1 ST DIGIT INTO R2
028C C520 0000      COMPARE IT TO 0
0290 4230 02A4      BRANCH ON NOT = TO SET INVERSE FLAG
0294 4820 0556      LOAD DIGITS 2-4 INTO R2
0298 C520 8000      COMPARE TO 8000   ( 800 MV + VOLTS OF FIRST DIGIT)
029C 4320 02C8      BRANCH ON NOT + TO RETURN
02A0 4200 0000      NOP
02A4 C860 0000      RESET R6 TO DESIRED STARTING FREQ.
02A8 4060 09C4      CLEAR INDEX STORAGE AREA
02AC C820 0001      LOAD 1 (SETS INVERSE FLAG)
```

```
02B0 4C20 09FE      STORE IN FLAG LOCATION IN CORE

02B4 4300 08BC      RETURN

02B8 A8B0 B0B2      STORAGE

02BC B 710 0000     STORAGE

02C0 0000              "

02C2 0000              "

02C4 4200 0000      NOP

02C8 C820 0000      LOAD 0 INTO R2  (RESETS INVERSE FLAG)

02CC 4020 09EE      STORE FLAG

02D0 C860 0000      CLEAR R6 (OR SET FOR DESIRED STARTING FREQ.)

02D4 4300 08A0      GO TO SET RELAYS

>
```

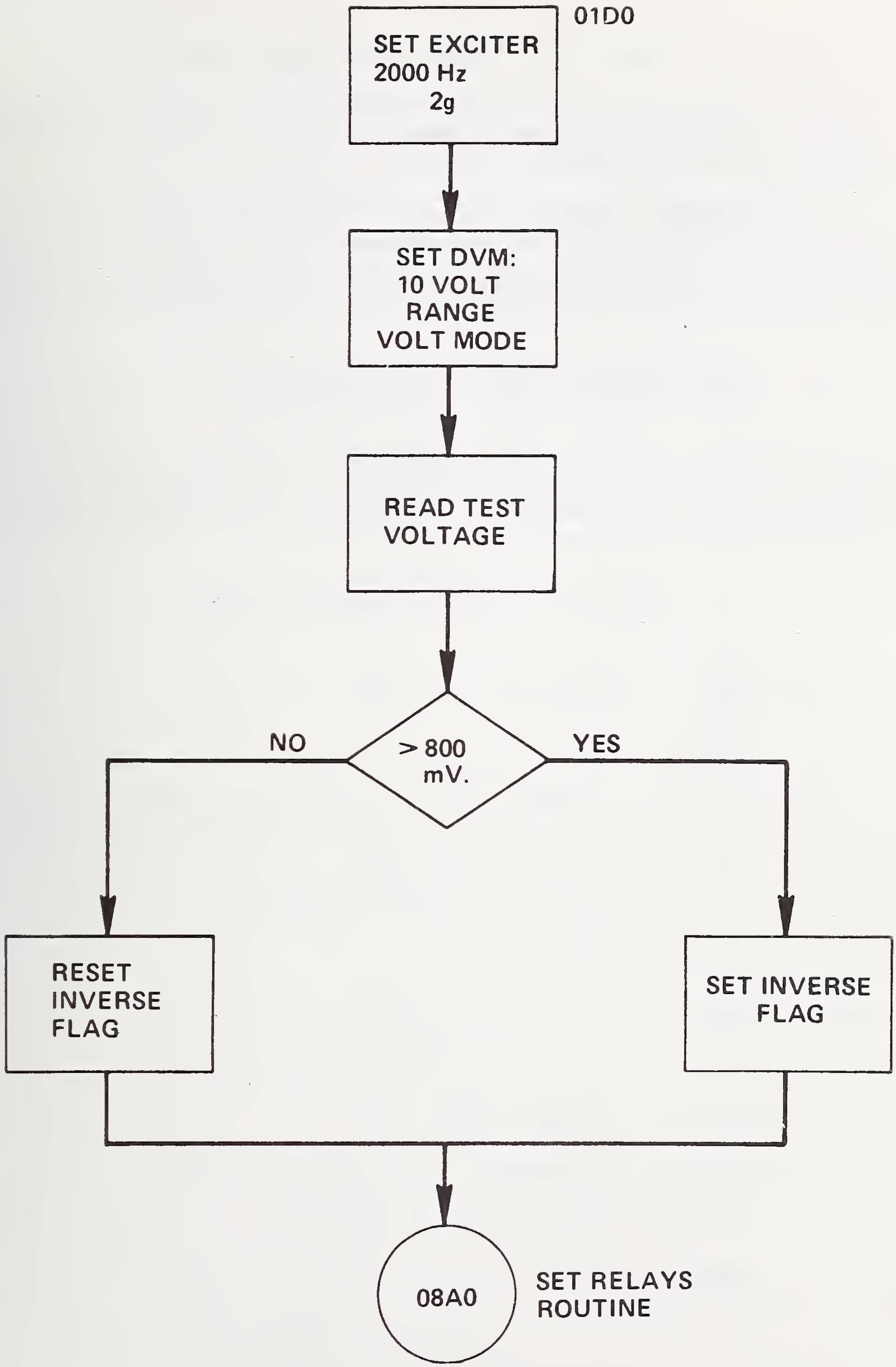


FIGURE 4-2. RANGE TEST FOR TEST ACCELEROMETER FLOWCHART

4.3 Check for INVERSE FLAG and Set Relays

This routine checks the INVERSE FLAG. If it is not set, the circuit remains in the a position (see Hardware Section). In this case, the relays are set as described in Section 4.15.

If the INVERSE FLAG is set, this is because the output of the test accelerometer is too high for circuit a and circuit b must be used as described in the Hardware Section of this report. In this case, relay eight must be activated (closed) and the binary code for the relays will be:

0000 0000 1110 0001 for "SLOW" mode, and
0000 0000 1110 0011 for "FAST" mode.

This can be seen schematically in figure 2-6. The solid line with the arrow at the end represents an 0 condition in the binary coding; a 1 condition in the binary coding is represented with no line connecting the pins. In this case only RII will be read.

Check for Inverse Flag and Set Relays

```
0300 4820 09EE     LOAD INVERSE FLAG

0304 C520 0001     IS IT SET?

0308 4330 0314     BRANCH ON = TO SET RELAY 8 FOR INVERSE RATIO

030C C8C0 0001     LOAD RATIO CODE

0310 4300 0A6A     GO BACK TO MAIN PROGRAM

0314 4200 0000     NOP

0318 C8C0 0001     LOAD RATIO CODE

031C C800 0071     LOAD RELAY BANK 1 DEVICE ADDRESS

0320 C840 0000     LOAD 1 VOLT CODE

0324 4520 0676     RESTORE R2 (FAST/SLOW CODE)

0328 CA20 0001     ADD 1 TO SET RELAY 8

032C 4200 0000     NOP
```

```
0330 9A02            SET RELAY BANK 1
0332 9A04             "     "
0334 4200 0000       NOP
0338 4200 0000       NOP
033C 4300 0AB4       GO TO RATIO II PROGRAM
0340 0000
0342 0000
0344 0000
0346 0000
0348 0000
034A 0000
034C 0000
034E 0000
0350 0000
0352 0000
0354 0000
0356 0000
0358 0000
035A 0000
```

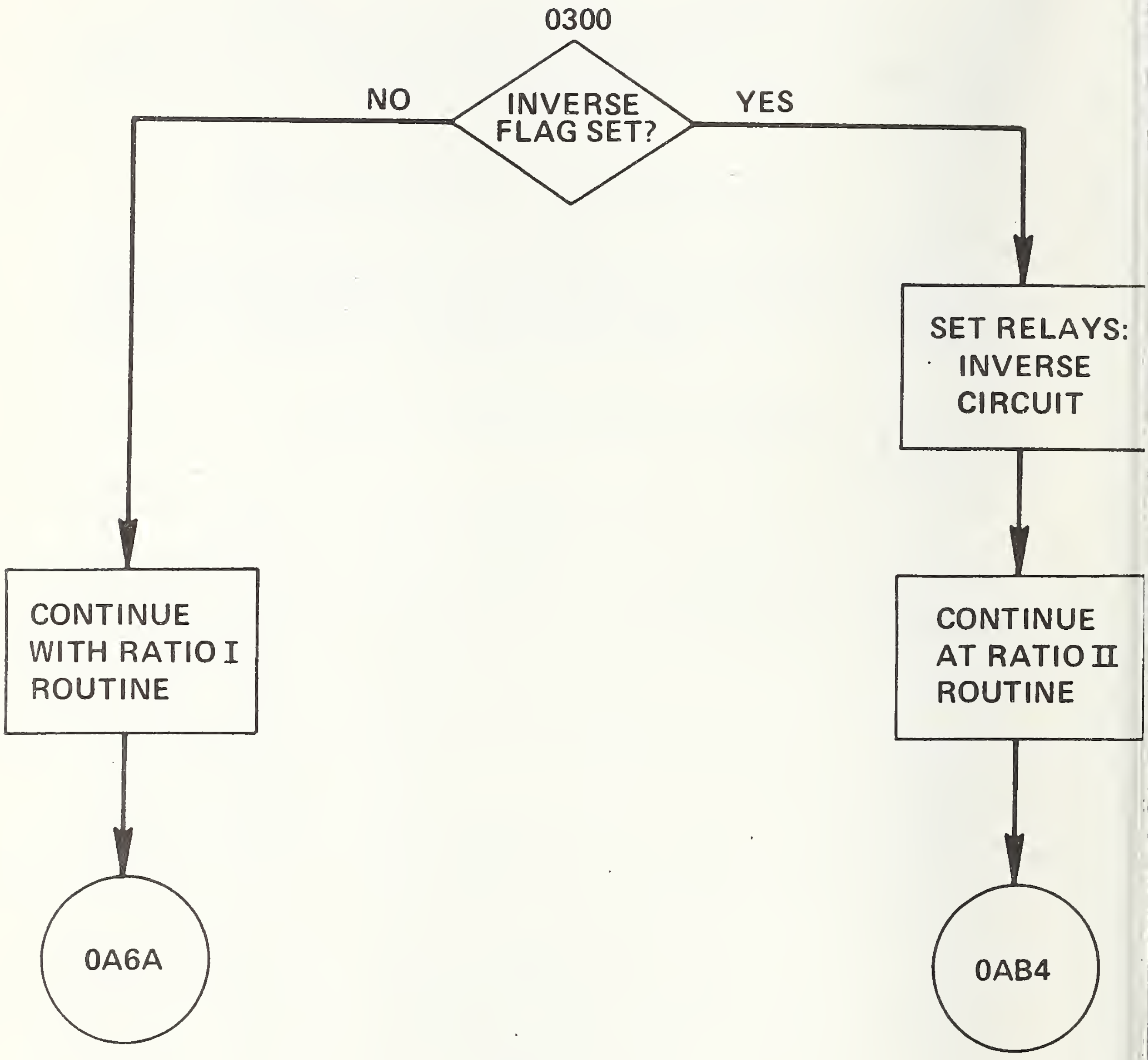


FIGURE 4-3. CHECK FOR INVERSE FLAG AND SET RELAYS FLOWCHART

4.4 Check for INVERSE FLAG and Calculate Calibration Factor for Test Accelerometer if Set

This routine determines if the INVERSE FLAG is set (circuit b as outlined in the Hardware Section). If set, the program will calculate the test accelerometer sensitivity and then go to Hexidecimal-to-Decimal routine. If the INVERSE FLAG is not set the program will continue at OAD4.

Check for INVERSE FLAG and Calculate Calibration Factor

```
0360 4200 0000     NOP
0364 4820 09FF     LOAD INVERSE FLAG
0368 C520 0000     IS IT NOT SET?
036C 4330 0AD4     BRANCH IF NOT SET TO RETURN TO MAIN PROG.
0370 4200 0000     NOP
0374 4830 09F8     LOAD RII
0378 0200          NOP
037A 0200          NOP
037C 4200 0000     NOP
0380 487F 0F54     LOAD CAL. FACTOR OF STANDARD
0384 C850 03E8     LOAD 1000 OR 10000 (THIS IS SET IN DECODING PROG.)
0388 0B44          CLEAR R4
038A 0200          NOP
038C 0C47          MULT. CAL OF STANDARD BY CONSTANT ABOVE
038E 0200          NOP
0390 0D43          DIVIDE THIS BY RII
0392 0200          NOP
0394 0835          SAVE R5 IN R3
0396 0200          NOP
0398 4300 0100     CONVERT TO DECIMAL
039C 4200 0000     NOP
```

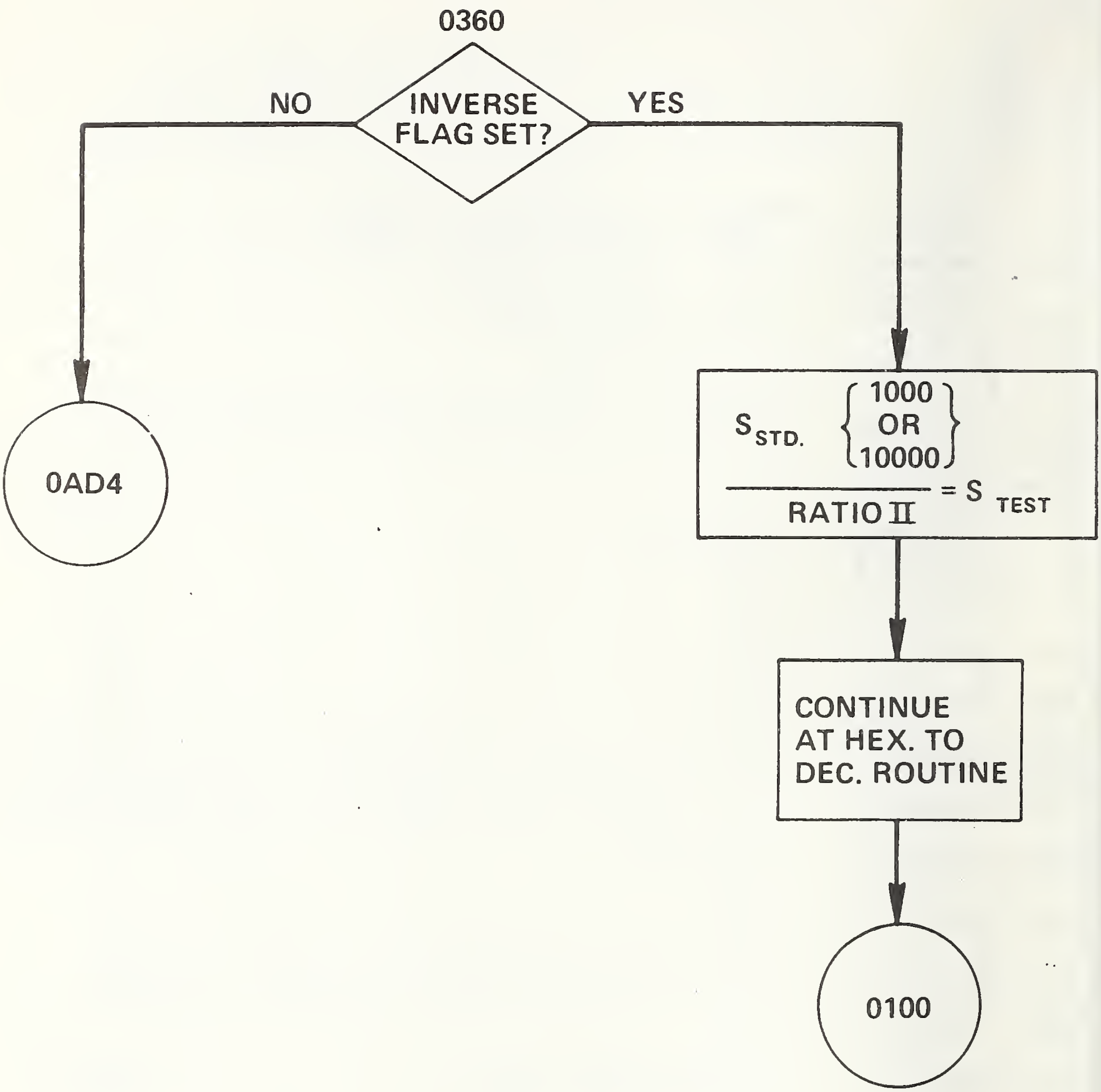


FIGURE 4-4. CHECK FOR INVERSE FLAG AND CALCULATE S_T IF SET FLOW CHART

Table of Constants
Used in Accelerometer Data Block Entry Program

Note: The following code is part of Section 4.46, Accelerometer Data Block Entry Program.

```
03A0 C64E D44E    FN  TN
03A4 C341 504D    CA  PM
03A8 504E 5053    PN  PS
03AC 414D 414E    AM  AN
03B0 4153 C355    AS  CU
03B4 44C3 C5B1    DC  E1
03B8 C5B2 53B1    E2  S1
03BC 53B2 4741    S2  GA
03C0 D748 41D4    WH  AT
03C4 3FA0 8D0A    ?   CR,LF
03C8 D748 C9C3    WH  IC
03CC 483F 8D0A    H?  CR,LF
03D0 50C9 C34B    PI  CK
03D4 5550 A04D    UP   M
03D8 CF44 A04E    OD   N
03DC CF3F 8D0A    O?  CR,LF
03E0 414D 50A0    AM  P
03E4 4DCF 44A0    MO  D
03E8 4ECF 3F8D    NO  ?CR
03EC 0AA0         LF,
03EE 0000
```

4.5 Clear Tables for Accelerometer Data Block

The first program on the following pages is used in the PARTIAL entry mode of the Entry Program (3740). When a parameter is called for in the PARTIAL mode, the existing record space is erased (set equal to 0000 for all bytes) for that parameter. The second program is used in the ALL entry mode of the Entry Program (3740). When ALL is typed in response to the question "All or Partial Changes?", the entire record space is cleared for the parameter entry program.

Clear Tables for Accelerometer Data Block

```
0400 4200 0000     NOP
0404 0B11          CLEAR R1
0406 0B22          CLEAR R2
0408 4200 0000     NOP
040C 4200 0000     NOP
0410 4012 2B00     CLEAR LOCATION (2B00 + (R2) )
0414 C520 0100     X=100?
0418 4330 0424     BRANCH ON = TO EXIT
041C CA20 0002     ADD 2 TO X
0420 4300 0410     CONTINUE
0424 4300 3790     EXIT CLEAR TABLES ROUTINE
0428 0000
042A 0000
042C 0000
042E 0000
0430 4843 38A6     LOAD # BYTES TO BE CLEARED
0434 0B11          CLEAR R1
```

```
0436 0B55           CLEAR R5
0438 4823 38A4      LOAD START LOCATION FOR CLEARING
043C 4200 0000      NOP
0440 D212 0000      CLEAR LOCATION ( (R2) )
0444 CA50 0001      ADD 1 TO COUNTER 1
0448 CA20 0001      ADD 1 TO COUNTER 2
044C 4200 0000      NOP
0450 0554           COMPARE COUNTER TO # BYTES TO CLEAR
0452 0200           NOP
0454 4230 0440      BRANCH ON NOT = TO CONTINUE CLEARING
0458 4300 37BC      EXIT CLEAR TABLES ROUTINE
>
```

Table of Messages

Note: The following memory locations are a storage area for messages used throughout the software programs.

```
     P

0460 8D0A 4ECF

0464 A041 4444

0468 C9D4 C9CF

046C  4E41 CCA0

0470 C6C9 CCC5

0474 A0C6 CF55

0478 4E44 A0C6

047C CFD2 A0D4

0480 48C9 53A0

0484 50C9 C34B

0488 5550 8D0A

048C 0000

048E 0000

0490 0000

0492 0000

0494 0000

0496 0000

0498 0000

>
```

4.6 *Round-Off Subroutine

Called on RE

This subroutine is used for data from the digital voltmeter. The DVM gives five digits of data, of which four digits would overload registers in certain operations (addition for example).

Entry Requirements: Five digit decimal number RO, R1 and core location 0554, 0556.

Output: Hexidecimal number equivalent to the decimal entry number rounded to four digits, output in R3 and core location 26EC.

*Round-Off Subroutine

```
                                        IN ON RO, R1 AND
                                           0554, 0556
049C 40F0 04F4     SAVE RE              OUT ON R3 AND 26EC

04A0 C500 0000     IS FIRST DECIMAL DIGIT ZERO?

04A4 4330 04E0     BRANCH ON =     (<9999)

04A8 CD00 000C     SHIFT LEFT C BITS

04AC CC10 0004     SHIFT RIGHT 4 BITS

04B0 0A10          R1+RO=R1

04B2 0B00          CLEAR RO

04B4 4200 0000     NOP

04B8 4190 0E9A     DECIMAL TO HEX (BACK ON R3)

04BC 4850 0556     LOAD DECIMAL # (LAST 4 DIGITS)

04C0 C450 000F     PICK OFF LAST DIGIT=D

04C4 CB50 0005     D-5=R5

04C8 4210 04D0     BRANCH ON NEGATIVE

04CC CA30 0001     ADD 1 TO HEX. #
```

```
04D0 4030 26EC      SAVE
04D4 C840 000A      LOAD MULTIPLIER CODE
04D8 4040 04F6      SAVE
04DC 4300 04F8      GO TO EXIT
04E0 4190 0E9A      DECIMAL TO HEX (R3)
04E4 4030 26EC      SAVE
04E8 C840 0001      LOAD MULTIPLIER CODE
04EC 4040 04F6      SAVE
04F0 4300 04F8      GO TO EXIT
04F4 26E0 000A      STORAGE FOR RE AND MULTIPLIER CODE
04F8 48E0 04F4      RESTORE RE
04FC 030E           EXIT
```

4.7 *ASCII-to-Decimal Subroutine

Call on R9

This subroutine converts five bytes of ASCII code to two half-word decimal code.

Input Requirements: ASCII code in core locations 02B9-02BD.

Output: Decimal code in core locations 0554 and 0556.

*ASCII-to-Decimal Subroutine

```
0500 4200 0000    NOP
0504 D3A0 02B9    LOAD DATA INTO REG'S RA TO RE
0508 D3B0 02BA                "
050C D3C0 02BB                "
0510 D3D0 02BC
0514 D3E0 02BD
0518 C4A0 000F     PICK OFF LAST 4 BITS
051C C4B0 000F              "
0520 C4C0 000F
0524 C4D0 000F
0528 C4E0 000F
052C CDB0 000C    SHIFT LEFT 12
0530 CDC0 0008    SHIFT LEFT 8
0534 CDD0 0004    SHIFT LEFT 4
0538 0ABC         RB+RC
053A 0ABD         RB+RD
053C 0ABE         RB+RE
```

```
053E 0200
0540 40A0 0554      STORE RA (FIRST DIGIT)
0544 40B0 0556      STORE RB (LAST 4 DIGITS)
0548 4200 0000
054C 4200 0000
0550 0309           RETURN
0552 0200
0554 0000           STORAGE AREA
0556 0136                 "
0558 0000
055A 0000
055C 0000
055E 0000
>
```

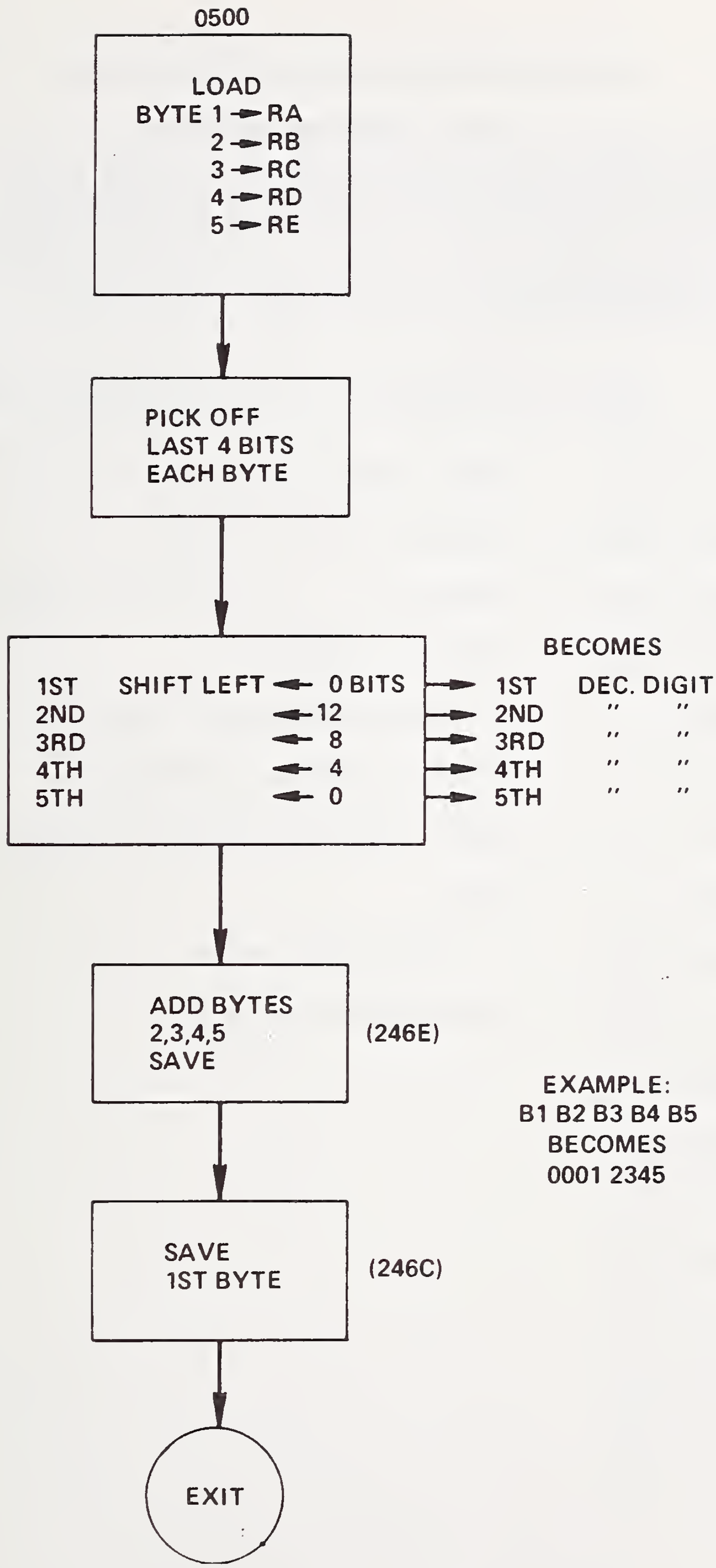


FIGURE 4-5. FIVE BYTE ASCII-TO-DECIMAL FLOWCHART

4.8 Check for UNSTABLE/FLAG and Type Message if Set

This routine checks the STABLE/UNSTABLE FLAG. This flag is set for either 0000 or 0001 in the Multiple Readings program. This flag indicates that the test accelerometer data fall within the stability requirements or that the data do not fall within the requirements. The data must not show a scatter of more than 0.1 percent to meet the requirements in the Multiple Readings program. In the case of unstable data, a diagnostic is typed out after the calibration factor on the TTY.

Check for UNSTABLE/FLAG and Type Message if Set

```
05B0 4810 230E     LOAD STABLE/UNSTABLE FLAG
05B4 C510 0000     COMPARE TO 0 (STABLE?)
05B8 4330 1020     BRANCH ON = TO LOOK FOR H OR S
05BC 41E0 3478     TYPE OUT MESSAGE
05C0 05D0          STARTING ADDRESS OF MESSAGE
05C2 05E7          ENDING ADDRESS OF MESSAGE
05C4 0001          CODE INDICATING ASCII FORMAT
05C6 0200          NOP
05C8 4300 1020     GO LOOK FOR H OR S
05CC A0A0 A0A0     STORAGE FOR MESSAGE
05D0 A0A0 A0A0
05D4 A0A0 A0A0
05D8 554E 53D4
05DC 4142 CCC5
05E0 A053 C947
05E4 4E41 CCA0
05E8 A0A0 A0A0
05EC A0A0 A0A0

>
```

Storage For Save Registers Subroutine

Saves All Registers Except E

Note: The following memory locations are reserved for the Save Registers Subroutine Section 4.9.

```
0670 2900 1010      STORAGE

0674 6300 1510

0678 3101 3010

067C 3901 5010

0680 5571 9910

0684 7670 9910

0688 3070 9910

068C 6771 9911

0690 7351 9912

>
```

4.9 *Save Registers Subroutine

(Call on RE)

```
06D8 4200 0000      NOP
06DC 4200 0000      NOP
06E0 4010 0674      SAVE R1 IN LOCATION 0674
06E4 4020 0676      ETC.
06E8 4030 0678
06EC 4040 067A
06F0 4050 067C
06F4 4060 067E
06F8 4070 0680
06FC 4081 0682
0700 4090 0684
0704 40A0 0686
0708 40B0 0688
070C 40C0 068A
0710 40D0 068C
0714 40E0 068E
0718 40F0 0690
071C 030E           RETURN TO CALL
0720 0200
>
```

4.10 *Restore Registers Subroutine

(Call on RE)

```
0720 4800 0690      RESTORE R0
0724 4810 0674      ETC.
0728 4820 0676
072C 4830 0678
0730 4840 067A
0734 4850 067C
0738 4860 067E
073C 4870 0680
0740 4880 0682
0744 4890 0684
0748 48A0 0686
074C 48B0 0688
0750 48C0 068A
0754 48D0 068C
0758 48E0 068E
075C 030E           RETURN TO CALL
>
```

4.11 Calibration Factors Exciter 2

```
0760 1009 1009
0764 1009 1009
0768 1009 1009
076C 1009 1009
0770 1009 1009
0774 1009 1009
0778 1009 1009
077C 1009 1009
0780 1010 1010
0784 1010 1010
0788 1010 1010
078C 1010 1010
0790 1011 1011
0794 1012 1012
0798 1012 1012
079C 1013 1013
07A0 1013 1014
07A4 1014 1014
07A8 1014 0300
>
```

4.12 Calibration Factors Exciter 1

```
07E0 2004 2004
07E4 2004 2004
07E8 2004 2004
07EC 2004 2004
07F0 2004 2004
07F4 2003 2006
07F8 2008 2010
07FC 2012 2013
0800 2018 2020
0804 2023 2023
0808 2023 2029
080C 2030 2031
0810 2031 2023
>
```

4.13 Start 1

Location 081A is the starting location for the automated accelerometer calibration program for exciter 1. The calibration factors for this exciter are converted to hexidecimal and transferred into the working storage area starting at location 0F54. The RANGE FLAG is reset to 0001 and saved in location 09E4. The SHAKER FLAG is set equal to 140C and saved in location 09C0. Register 6 is cleared for starting the program at the initial frequency.

Start 1

```
*****           START 1  (SHAKER 1)

081A 0BFF            CLEAR RF
081C 0BAA            CLEAR RA
081E 0B00            CLEAR R0
0820 481A 07E0       LOAD SHAKER 1 CAL. FACTOR
0824 4190 0E9A       CONVERT TO HEX.
0828 403A 0F54       STORE IN MAIN PROGRAM
082C CAA0 0002       ADD 2 TO RA
0830 C5A0 0032       COMPARE TO UPPER LIMIT
0834 4230 0820       IF NOT = GO LOAD ANOTHER FACTOR
0838 4820 0E90       LOAD CUT OFF FREQ. CODE
083C 4020 0E8C       STORE CODE
0840 C810 0001       LOAD 1
0844 4010 09E4       SET RANGE FLAG =1
0848 0B22            CLEAR R2
```

```
084A 4020 0A02    CLEAR THIS LOC. FOR MAG. TAPE PROG

084E 0200         NOP

0850 C810 140C    LOAD SHAKER 1 FLAG

0854 4010 09C0    STORE

0858 0B66         CLEAR R6 TO START AT FIRST TEST FREQ.

085A 4300 01D0    GO TO SET RELAYS PROG.

>
```

4.14 Start 2

Location 085E is the starting location for the automated accelerometer calibration program for exciter 2. This program performs the same function as Start 1 except for setting the SHAKER FLAG equal to 1408.

Start 2

```
*****                START 2 (SHAKER 2)

085E 0BFF            CLEAR RF
0860 4820 0E8E       LOAD CUT OFF FREQ. CODE
0864 4020 0E8C       STORE IN MAIN PROG.
0868 0BAA            CLEAR RA
086A 0B00            CLEAR R0
086C 481A 0760       LOAD SHAKER 2 CALIBRATION FACTOR
0870 4190 0E9A       CONVERT IT TO HEX.
0874 403A 0F54       STORE IN MAIN PROG.
0878 CAA0 0002       ADD 2 TO RA
087C C5A0 004A       COMPARE TO UPPER LIMIT
0880 4230 086C       IF NOT = LOAD ANOTHER FACTOR
0884 C810 0001        SET RANGE FLAG =1 AND SAVE
0888 4010 09E4
088C 4200 0000
0890 0B22            CLEAR R2
0892 0200
0894 C810 1408       LOAD SHAKER 2 FLAG
0898 4010 09C0       STORE FLAG
089C 4300 01D0       GO TO RANGE TEST

>
```

4.15 Set Relays for Ratio I, Output Oscillator Code, Read Frequency Counter, Type it on TTY

4.15.1 Programming Relay Banks 1 and 2. This routine sets up relay banks 1 and 2. Relay bank 1 sets up the electrical circuit for either Ratio I or Ratio II (see figure 1-1). In this program, the relays will be set up to read Ratio I. The Ratio I readings will be:

$$RI = \frac{\text{Standard Accelerometer Voltage}}{\text{Standard Accelerometer Voltage} \cdot G},$$

where G = the gain of the amplifying circuit in figure 1-1.

The hexidecimal code, 00E0, is found in location 08B6. Converting this to binary:

0000 0000 1110 0000

Disregarding the first eight bits,because a Write Data statement uses only the last eight bits, the useful part is:

1110 0000.

The first bit on the left represents the position of relay 1, the second bit of relay 2, etcetera up to bit eight which represents the position of relay 8. In figure 1-1, the schematic of the voltage ratio circuit is given. The relays 1 through 14 are shown. The solid line with the arrow at the end represents a 0 condition in the binary coding, and no line represents a 1 condition in the binary coding.

The binary code for relays 9 through 16 is:

0000 0000,

which indicates that in figure 1-1, relays 13 and 14 are set in the solid line position.

The following table summarizes the coding for relays 1 through 16.

TABLE 4-1. Relay Bank 1 Coding

Relay Number		Code 0	Code 1
1	Connects relay 6 to	Ref. signal 2	Ref. signal 1
2	Connects relay 5 to	Test signal 2	Test signal 1
3	Connects test converter to	Test signal	Ref. signal
4	Connects output of test converter to	DVM input	Wave analyzer
5	Connects test signal to	Relay 3	Wave analyzer
6	Connects reference signal to	Relay 6	----
7	Sets ac/dc converters to	"Slow" mode	"Fast" mode
8	Connects reference ac/dc converter input to	Output of amp.	Test signal
9	----	----	----
10	Controls polarity of power supply output to wave analyzer circuit (see fig. 2-4)	Negative	Positive
11	See figure 2-4 Power Supply	Disconnected	Connected
12	----	----	----
13	Reference ac/dc converter range	1 Volt	10 Volts
14	Signal ac/dc converter range	1 Volt	10 Volts
15	----	----	----
16	Contact for "seek" on oscilloscope	Opens	Closes

Relays 17 through 32 control a bank of capacitors. This is for impedance matching of the exciter drive coil to the power amplifier, and is connected in series between the power amplifier output and the exciter drive coil. The following table gives the values of the capacitors which each relay controls.

TABLE 4-2. Relay Bank 2 Coding

Relay Number	Capacitance (μF)*
17	0.1
18	0.2
19	0.3
20	0.4
21	0.5
22	1.0
23	2.0
24	3.0
25	4.0
26	5.0
27	10.0
28	20.0
29	30.0
30	40.0
31	100.0
32	Shunt

*The binary code of 0 disconnects a capacitor; a code of 1 connects a capacitor. (See figure 2-5.)

The following diagram illustrates the binary coding for the two bytes of code which control the capacitance bank. The number in the boxes is the number of the relay.

17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32

A 0 code disconnects a capacitor; a code of 1 connects a capacitor. The following examples of the capacitor bank coding are found in the 1400 Data Block Control Constants.

Example 1: at core location 140C, the hexidecimal coding is 0001. This code will mean all capacitors except number 32 are disconnected from the circuit. On the previous chart, relay number 32 is a shunt.

Example 2: at location 147C, the hexidecimal coding is 00A6. This converted to binary is 0000 0000 1010 0110. From the diagram above this is found to mean capacitors corresponding to relays 25, 27, 30, 31 are connected. The capacitor bank is wired so that these selected capacitors are connected in parallel. The capacitance is then the sum in the above example which is 154μF. The total available capacitance is 221.5μF, excluding the shunt.

4.15.2 <u>Programming the Decade Oscillator</u>. The oscillator has the following decade controls:

Voltage: 1 volt increments from 1 to 9 volts
0.1 volt increments from 1.0 to 0.9
0.01 volt increments from 0.01 to 0.09

Frequency: 100 Hz increments from 100 to 900
10 Hz increments from 10 to 90
1 Hz increments from 1 to 9
0.1 Hz increments from 0.1 to 0.9

Multiplier of any of the above: X1, X10, X100.

The oscillator is programmed by four successive Write Data instructions. The coding must be in the following format:

First byte of coding: 0.1 volt increment
0.01 volt increment

Second byte: 0.1 Hz increment
1 volt increment

Third byte: 10 Hz increment
1 Hz increment

Fourth byte: Multiplier, 100 Hz increment

The subroutine at 16C0 translates the code of the 1400 DATA BLOCK to the previous format. The first two columns in the 1400 DATA BLOCK entitled "FREQ" are reformated for each test frequency prior to programming the oscillator. This routine programs the oscillator initially. The Constant Acceleration Subroutine (see Section 4.27) programs the oscillator for a desired acceleration level.

4.15.3 Reading the Frequency Counter. To read the frequency counter the following operations are required:

1. Execute an output command to start the counter, and

2. Sense status of counter and, when "not busy" condition is reached, execute six Read Data instructions.

3. The counter can be stopped by an output command to stop.

The code read into computer core is in the ASCII format, which is ready to be sent to the teletype. In the present program, it is desirable to start the counter counting after the initial count is finished in order that the operator can see the current test frequency displayed on the counter (see location 0A30).

Set Relays for Ratio I, Output Oscillator Code,
Read Frequency Counter, Type it on TTY

```
08A0 C830 0004    LOAD COUNTER STOP COMMAND
08A4 4060 09C4    CLEAR INDEX STORAGE AREA
08A8 C800 000E    LOAD COUNTER DEVICE ADDRESS INTO R0
08AC 9E03         STOP COUNTER
08AE 0B33         CLEAR R3 FOR RELAY CODE
08B0 C800 0071    LOAD RELAY BANK 1 DEVICE ADDRESS INTO R0
08B4 C820 00E0    LOAD CODE FOR RELAY BANK 1
08B8 9A02         WRITE DATA TO RELAY BANK 1 FOR RI RATIO
08BA 9A03                         "
08BC 0B22         CLEAR R2
```

```
08BE C870 000F      LOAD OSCILLATOR DEVICE ADDRESS
08C2 9D75           SENSE STATUS OF OSCILLATOR
08C4 0552           BUSY?
08C6 4230 08C2      IF SO, SENSE AGAIN
08CA 48E0 09C0      LOAD SHAKER FLAG INTO RE (TO IDENTIFY WHICH SHAKE
08CE DA70 0C80      TURN OFF OSCILLATOR
08D2 DA70 0C80
08D6 DA70 0C80
08DA DA70 0C80
08DE 4020 09E2      STORE 0000 (USED IN CHECK FREQ. ROUTINE)
08E0 0200
08E2 0BAA           CLEAR RA (USED TO OUTPUT CODE TO OSCILLATOR)
08E4 4880 09C4      LOAD CAPACITOR INDEX INTO R8
08E8 40E0 08FE      STORE SHAKER FLAG BELOW TO SET RELAYS
08EC C850 0072      LOAD RELAY BANK 2 DEVICE ADDRESS (FOR CAPACITORS)
08F0 C8D0 0001      LOAD 1 INTO RD
08F4 0ADE           ADD 1 TO SHAKER FLAG TO SET RELAYS
08F6 40D0 0902      STORE BELOW FOR OUTPUT TO RELAY BANK 2
08FA 0200           NOP
08FC DA58 1408      WRITE DATA TO RELAY BANK 2 (CAPACITORS)
0900 DA58 1409         "                  "
0904 4190 16C0      BAL TO SET UP OSCILLATOR CODE FOR THIS FREQ.
0908 9D75           SENSE STATUS OF OSC
090@ 0552           BUSY?
```

```
090C 4230 0908    IF SO, GO SENSE AGAIN
0910 DA76 0C8C    WRITE DATA TO OSCILLATOR
0914 0B88         CLEAR R8
0916 0B00         CLEAR R0
0918 4560 0E8C    IS THIS THE LAST TEST POINT?
091C 4330 0F9E    IF SO, GO RING BELLS,ETC.
0920 CA60 0001    ADD 1 TO INDEX REGISTER
0924 4200 0000    NOP
0928 CAA0 0001    ADD 1 TO OSC. COUNTER INDEX
092C C5A0 0004    HAVE 4 BYTES BEEN WRITTEN TO OSC?
0930 4230 0908    IF NOT, GO WRITE ANOTHER BYTE
0934 0896         LOAD INDEX REG. INTO R9
0936 C8A0 0004    LOAD 4 INTO RA
093A 0C8A         R9*RA PUT RESULTS INTO R8,R9
093C 4090 09C4    STORE CAPACITOR INDEX
0940 C810 0000    LOAD 0
0944 4010 09EA    CLEAR "THRU"FLAG  (USED IN RANGE TEST )
0948 C800 000E    LOAD COUNTER DEVICE ADDRESS INTO R0
094C C810 0002    LOAD TTW DEVICE ADDRESS INTO R1
0950 C830 0004    LOAD STOP COMMAND FOR COUNTER INTO R3
0954 C840 0008    LOAD START COMMAND FOR COUNTER INTO R4
0958 4190 0E78    BAL TO DELAY ROUTINE FOR COUNTER
095C 9E04         STARTS COUNTER
095E 9D09         SENSE STATUS OF COUNTER
0960 4280 095E    IF BUSY SENSE STATUS AGAIN
```

```
0964 0BBB            CLEAR RB
0966 C8C0 0001       LOAD 1 INTO RC   (INCREMENT REGISTER FOR LOOP)
096A 08D4            LOAD 8 INTO RD  (LIMIT REGISTER FOR LOOP)
096C DB0B 09B6       READ COUNTER INTO MEMORY
0970 C1B0 096C       BXLE
0974 9E03            STOPS COUNTER
0976 4300 2560       BRANCH TO CHECK FOR PROPER FREQ.
097A C8D0 0006       LOAD 6 INTO RD (LIMIT REGISTER FOR TTW LOOP)
097E C880 0006       LOAD 6 INTO R8
0982 0BBB            CLEAR RB  (BXLE REGISTER)
0984 DE10 09B2       PUT TTW IN WRITE MODE
0988 DA1B 09B4       OUTPUT TO TTW: CR,LF,FREQ.DATA
098C 9D1E            SENSE STATUS OF TTW
098E 4280 098C       IF BUSY GO SENSE STATUS AGAIN
0992 C1[illegible]C 0988       BXLE (LOOP FOR TTY PRINTOUT)
0996 C830 0002       LOAD 2 INTO R3
099A DE30 09B0       PUT TTY IN READ MODE (TO ACCEPT H OF S)
099E 4300 0A30       CONTINUE WITH MAIN PROGRAM

>
```

```
*****          STORAGE LOCATIONS

09A2 09F2

09A4 0000

09A6 F000 0000

09AA 8000 3340      TEMP. STOP COMMAND

09AE 0000

09B0 9400           READ MODE CODE FOR TTY

09B2 9898           WRITE MODE CODE FOR TTY

09B4 8D8A B0B0      CODE FOR CR,LF, FREQ DATA STORAGE

09B8 B0B5 B0B0      FREQ DATA STORAGE

09BC B0B0 B000          "

09C0 1408 0000      SHAKER FLAG STORAGE 140C=SHAKER 1, 1408=SHAKER 2
                    STORAGE FOR R9 (TTY DATA)
09C4 0040           STORAGE FOR R9 (INDEX FOR CAPACITOR CODES)

09C6 0005           STOR. FOR LOOP COUNTER FOR CONST G. ROUTINE

09C8 0000

09CA 0014           STOR. FOR R5 IN SUB. CONST G

09CC 0000

09CE 0000

09D0 0000

09D2 8000 2864      TEMP. STOP COMMANDS

09D6 8000 3320        "

09DA 0000

09DC 0904           ADDRESS OF PSW

09DE 0A5E           STORAGE FOR R4
```

09E0	0002	STOR. FOR R5 IN DECODE DATA ROUTINE(TELLS WHERE DEC. PT. IS)
09E2	0000	STOR. FOR COUNTER IN CHECK FREQ. ROUTINE
09E4	0001	STOR. FOR RANGE FLAG FOR SIGNAL CONV. 1 OR 10 RANGE
09E6	000A	STOR. FOR RI (RATIO I DECIMAL)
09E8	8677 0000	STORAGE FOR RII (RATIO II DEC.) STOR. FOR # TIMES THRU RANGE CHECK ROUTINE
09EC	055E	STOR. FOR RI (RATIO I HEX.)
09EE	0000	INVERSE FLAG (0=NOT INVERSE; 1=INVERSE)
09F0	0000	
09F2	00E0	STOR. FOR R2,R4 (RELAY CODES)
09F4	0000	"
09F6	0000	STORAGE FOR ZERO/NON ZERO FLAG(FIRST DIGIT OF RII
09F8	1B8D 2756	RATIO II (HEX) TEST CAL FACTOR(HEX)
09FC	03E8	NOT ASSIGNED
09FE	2710 0000	NOT ASSIGNED NOT ASSIGNED
0A02	0000	STORAGE FOR COUNTER IN MAG TAPE PROG.
0A04	A400 F000	TTY UNBLOCK,READ STORAGE FOR R3 IN DEC/HEX /H
0A08	1B8F F000	HEX # IN 09EA ROUTINE NOT ASSIGNED
0A0C	0000	NOT ASSIGNED
0A0E	F000 0000	
0A12	F000 0000	
0A16	F000 0000	
0A1A	F000 0000	
0A1E	F000 0000	
0A22	F000 0000	
0A26	F000 0000	
0A2A	F000 0000	
0A2E	0000	

```
0A30 9E04           TURN COUNTER BACK ON (FOR CONTINUOUS DISPLAY)
0A32 41A0 12D0      BAL TO SCOPE SCALE ROUTINE
0A36 C840 0000      LOAD RELAY CODE IN R4
0A3A C800 0071      LOAD RELAY BANK 1 DEVICE ADDRESS
0A3E 41E0 17B4      BAL TO DETERMINE FAST/SLOW FOR AC/DC CONVERTERS
                    (SETS UP R2)
0A42 4020 09F2      SAVE R2
0A46 4040 09F4      SAVE R4
0A4A 9A02           WRITE DATA TO RELAY BANK 1
0A4C 9A04               "
0A4E DE30 0B24      ENABLE INTERUPT FOR DVM
0A52 9F44           CLEAR PENDING INTERUPTS
0A54 C8C0 0002      LOAD VOLT CODE
0A58 9E3C           SET DVM FOR VOLT READ
0A5A 4140 10F0      BAL TO SET ACCELERATION LEVEL
0A5E 4200 0000      NOP
0A62 C830 00CA      LOAD DVM DEVICE ADDRESS
0A66 4300 0300      GO CHECK FOR INVERSE FLAG
0A6A 9E3C           SET DVM FOR RATIO
0A6C 4190 1A00      BAL TO MULTIPLE READINGS SUBROUTINE
0A70 4810 230C      LOAD AVERAGE RATIO I (HEX)
0A74 4010 09EC      SAVE
0A78 C810 0001      RESET RANGE FLAG = 1
```

```
0A7C 4010 09E4
0A80 4200 0000     NOP
0A84 4300 0AA0     GO TO RII PROGRAM
0A88 0000
0A8A 0000
0A8C 0000
0A8E 0000
0A90 0000
0A92 0000
0A94 0000
0A96 0000
0A98 0000
0A9A 0000
0A9C 0000
0A9E 0000
>
```

4.16 Set Relays for Ratio II, Calculate Test Accelerometer Sensitivity

4.16.1 Programming Relay Banks 1 and 2. This routine sets up relay bank 1 for Ratio II readings.

$$RII = \frac{\text{Test Accelerometer Voltage}}{\text{Standard Accel. Voltage} \cdot G} ,$$

where G is the gain of the amplifying circuit in figure 1-1. The binary code for this will be:

0000 0000 1100 0000 for the "SLOW" position of converters, and
0000 0000 1100 0010 for the "FAST" position of converters.

See figure 1-1 and explanation under Programming Relays for RI for explanations.

4.16.2 Calculation of Test Accelerometer Calibration Factor. The calibration factor for the test accelerometer is given by:

$$S_{Test} = \frac{RII \cdot S(\text{Standard})}{RI} \cdot \text{RANGE FLAG (0001 or 0010)} .$$

The RANGE FLAG is 0001 if test converter was set for 1 volt full scale and 0010 if test converter was set for 10 volts full scale. This converter range is automatically tested at each test point at the routine at 19A0. If the INVERSE FLAG is set (set at Range Test program 01D0) a different formula is used in the computation of the Test Accelerometer Sensitivity. In this case,

$$S_{Test} = \frac{(1000 \text{ or } 10000) \cdot S(\text{Standard})}{RII} .$$

The multiplying constant is set at 1000 if the LOW/NORM FLAG is 1 and is set at 10000 if the LOW/NORM FLAG is 0. The LOW/NORM FLAG is determined when RII is read; a Ratio II with the first digit of zero sets the LOW/NORM FLAG at 0 and a non-zero digit sets the flag at 1-1. (See Read Data Subroutine location 2000).

Set Relays for Ratio II, Read Ratio II,
Calculate Test Accelerometer Sensitivity

```
0AA0 C800 0071     LOAD RELAY BANK 1 DEVICE ADDRESS
0AA4 41E0 17D8     BAL TO DETERMINE FAST/SLOW MODE FOR CONVERTERS
0AA8 41A0 12D4     BAL TO SCOPE SCALE
0AAC C840 0000     RELOAD RELAY CODE
0AB0 9A02          SET RELAY BANK 1
0AB2 9A04           "    "      "
0AB4 C830 00CA     LOAD DVM DEVICE ADDRESS
0AB8 DE30 0B24     ENABLE INTERUPTS FOR DVM
0ABC 9F44          CLEAR INTERUPTS
0ABE 9E3C          SET DVM FOR RATIO
0AC0 4190 1A00     BAL TO MULTIPLE READINGS SUBROUTINE
0AC4 4810 230C     LOAD RATIO II (HEX)
0AC8 4010 09F8     SAVE
0ACC 4200 0000     NOP
0AD0 4300 0B04     CHECK FOR INVERSE FLAG
0AD4 48B0 09F8     LOAD RATIO II (HEX)
0AD8 4300 19A0     CHECK FOR CORRECT RANGE OF TEST CONVERTER
0ADC 4830 09F8     RELOAD RATIO II (HEX)
0AE0 4870 09EC     LOAD RATIO I
0AE4 485F 0F54     LOAD CAL. FACTOR OF STANDARD
0AE8 CAF0 0002     ADD 2 TO INDEX
0AEC 4890 09F4     LOAD RANGE FLAG (0001 OR 000A)
```

```
0AF0 0C25              RATIO II * CAL. FACTOR OF STD.
0AF2 0D27              DIVIDE BY RATIO I
0AF4 0C29              MULT. BY RANGE (0001 OR 000A)
0AF6 0200              NOP
0AF8 4200 0000         NOP
0AFC 4200 0000         NOP
0B00 4300 0100         GO CONVERT CAL. FACTOR TO DECIMAL
0B04 4840 09F6         LOAD LOW/NORM FLAG
0B08 C540 0000         EQUAL TO 0?
0B0C 4330 0B18         BRANCH ON =
0B10 C890 03E8         LOAD CONST.
0B14 4300 0B1C         BRANCH
0B18 C890 2710         LOAD CONST.
0B1C 4090 0386         STORE CONST. IN CALCULATION PROGRAM
0B20 4300 0360         CHECK FOR INVERSE FLAG
0B24 4030 0000         CODE FOR DEVICE INTERUPT ENABLE,OUTPUT CONVERT
                       FOR DVM
0B28 0000
0B2A 0000
0B2C 0000
0B2E 0000
0B30 0000
0B32 0000
0B34 0000
0B36 0000
```

```
0B38 0000
0B3A 0000
0B3C 0000
0B3E 0000
0B40 0000
0B42 0000
0B44 0000
0B46 0000
0B48 0000
0B4A 0000
0B4C 0000
0B4E 0000
>
```

Storage Space for Digital Filter Subroutine

Note: The following memory locations are reserved for the Multiple Readings subroutine Section 4.39.

0B50

•

•

•

•

0C6E

4.17 Storage Table for Oscillator Code

Note: The following memory locations are reserved for storage for oscillator code as calculated in Section 4.33.

```
0C80 0000
0C82 0000
0C84 0000
0C86 0000
0C88 0000
0C8A 0000
0C8C 4000 1010
0C90 8500 1510
0C94 4401 3010
0C98 4901 5010
0C9C 7071 9910
0CA0 8470 9910
0CA4 3370 9910
0CA8 7371 9911
0CAC 9751
0CAE 9912
0CB0 1452 9913
0CB4 3752 9814
0CB8 5842 9815
0CBC 8032 9716
0CC0 0423
0CC2 9717
0CC4 2703 9718
0CC8 5003 0021
0CCC 7194 4921
0CD0 0695
0CD2 6921 7995
0CD6 9921
0CD8 8592 9921
0CDC 1391 9921
0CE0 3477 4922
0CE4 5276 9922
0CE8 0867
0CEA 4923 2767
0CEE 9923
0CF0 0010
0CF2 4924 6101
0CF6 9924
```

```
0CF8 7091 4825
0CFC 7091 9825
0D00 7571 4826
0D04 8161 9826
0D08 1552 4827
0D0C 9041
0D0E 9827
0D10 9041
0D12 4828 9431
0D16 9828
0D18 9841
0D1A 4829 1442
0D1E 9829
>
```

4.18 TTY Printout

This routine types the test accelerometer calibration factor on the teletype. There are three codes which are important here. They are:

Decimal Point Code:	Number of digits to be typed before decimal point is typed,
Digits to be Printed Code:	Total number of digits to be typed,
INVERSE FLAG:	Determined which ratio circuit was used to calculate calibration factor (set in Range Test 01D0). If the flag is set, a different printout routine is used.

The first two codes above were set in the 01D0 Hexidecimal-to-Decimal routine for Test Accelerometer Sensitivity. From TTY Printout routine, the program branches to 0B50 to test for STABLE/UNSTABLE condition.

TTY Printout

```
0DA0 DE10 09B2     PUT TTY IN WRITE MODE
0DA4 4180 0EEC     BAL TO WAIT
0DA8 DA10 0E5C     OUTPUT SPACE
0DAC 4180 0EEC     WAIT
0DB0 DA10 0E5C     OUTPUT SPACE
0DB4 4180 0EEC     WAIT
0DB8 9A1A          OUTPUT FIRST DIGIT
0DBA 0200          NOP
0DBC 4180 0EEC     WAIT
0DC0 9A1B          OUTPUT SECOND DIGIT
0DC2 0200          NOP
0DC4 4180 0EEC     WAIT
0DC8 4820 01A6     LOAD DEC. PT. CODE
0DCC C520 0002     IS IT 2?
```

```
0DD0 4330 0DE0     IF SO, BRANCH
0DD4 4300 0E20     GO CHECK FOR INVERSE FLAG
0DD8 4190 0EF6     OUTPUT DECIMAL PT.
0DDC 4300 0DEC     BRANCH
0DE0 4190 0EF6     OUTPUT DECIMAL PT.
0DE4 4180 0EEC     WAIT
0DE8 9A1C          OUTPUT THIRD DIGIT
0DEA 0200          NOP
0DEC 4180 0EEC     WAIT
0DF0 9A1D          OUTPUT FOURTH DIGIT
0DF2 0200          NOP
0DF4 4820 01A8     LOAD # DIGITS TO BE PRINTED CODE
0DF8 C520 0005     IS IT 5 ?
0DFC 4230 0E10     BRANCH ON NOT =
0E00 4180 0EEC     WAIT
0E04 4200 0000     NOP
0E08 4200 0000     NOP
0E0C 9A1E          OUTPUT FIFTH DIGIT
0E0E 0200          NOP
0E10 C870 000F     LOAD OSC. DEVICE ADDRESS
0E14 4180 0EEC     WAIT
0E18 4300 05B0     LOOK FOR STABLE/UNSTABLE FLAG
0E1C 0000
0E1E 0000
```

```
0E20 4820 09EE      LOAD INVERSE FLAG
0E24 C520 0000      IS IT NOT SET ?
0E28 4330 0E4C      BRANCH ON =
0E2C 9A1C           OUTPUT THIRD DIGIT
0E2E 0200           NOP
0E30 4180 0EEC      WAIT
0E34 9A1D           OUTPUT FOURTH DIGIT
0E36 0200           NOP
0E38 4180 0EEC      WAIT
0E3C 9A1E           OUTPUT FIFTH DIGIT
0E3E 0200           NOP
0E40 4180 0EEC      WAIT
0E44 4190 0EF6      OUTPUT DECIMAL PT.
0E48 4300 0E18      EXIT
0E4C 9A1C           OUTPUT THIRD DIGIT
0E4E 0200           NOP
0E50 4300 0DD8      RETURN
>
```

4.19 Table of Defined Constants

```
0E58 F000 0000
0E5C A0A0 0000     SPACE,SPACE
0E60 8700 0000     BELL
0E64 F000 0000
0E68 F000 0000
0E6C F000 0000
0E70 F000 0000
0E74 F000 0000
```

DELAY ROUTINE FOR FREQ. COUNTER

```
0E78 0PBB          CLEAR RB
0E7A 4200 0000     NOP
0E7E CABO 0001     ADD 1 TO RB
0E82 C5B0 0800     COMPARE COUNTER TO DESIRED LIMIT
0E86 4230 0E7E     IF NOT EQUAL, ADD ONE MORE TO THE COUNT
0E8A 0309          RETURN
```

DEFINED CONSTANTS

```
E8C 0064           CUT OFF FREQ. CODE(PROG. SETS THIS VALUE)
0E8E 0094          CUT OFF FREQ. CODE FOR SHAKER 2
0E90 0064          CUT OFF FREQ. CODE FOR SHAKER 1
0E92 F000 0000
0E96 F000 0000
```

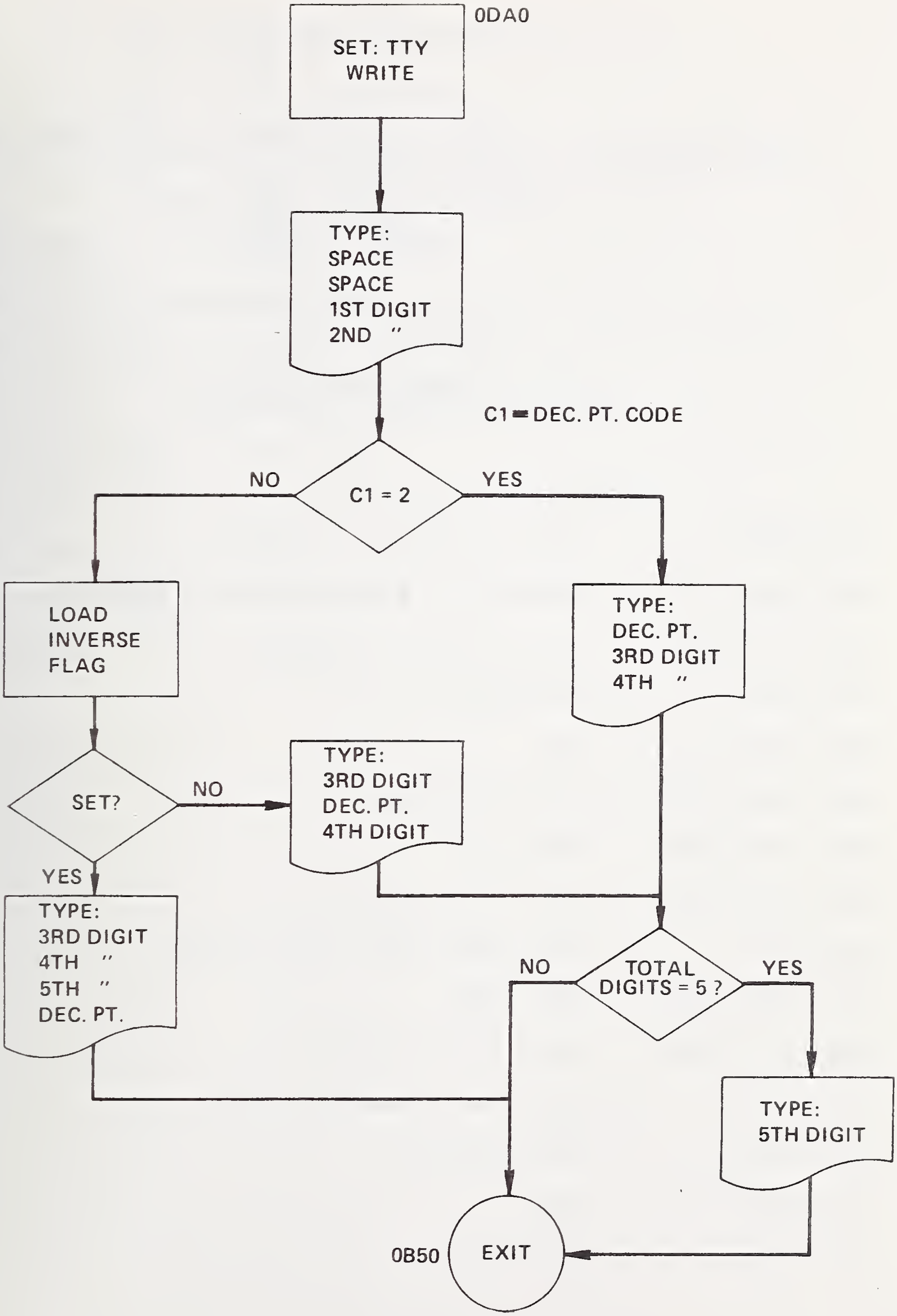


FIGURE 4-6. TELETYPEWRITER PRINTOUT FLOW CHART

4.20 *Decimal-to-Hexidecimal Subroutine

Call on R9

Input requirements: Decimal number up to five digits in R0 and R1.

Example: decimal number: 29345
(R0) = 0002
(R1) = 9345

Output requirements: Four digit hexidecimal number in R3.

Decimal-to-Hexidecimal Subroutine
(Call on R9 in on R0, R1 Out on R3)

```
0E9A 08E9           LOAD R9 INTO RA TO SAVE IT
0E9C 0851           LOAD DEC. # INTO R5 (LAST 4 DIGITS)
0E9E C450 F000      PICK OFF BITS 0-3
0EA2 0891           RELOAD DEC. #
0EA4 0200           NOP
0EA6 C490 0F00      PICK OFF BITS 4-7
0EAA C870 0064      LOAD 64
0EAE CC90 0008      SHIFT R9 RIGHT 8 BITS
0EB2 0C87           MULT THIRD DEC. DIGIT BY 64
0EB4 C830 03E8      LOAD 03E8
0EB8 CC50 000C      SHIFT RIGHT 12 BITS
0EBC 0C25           MULT FOURTH DIGIT BY 03E8
0EBE 0A39           ADD R3 AND R9
0EC0 0200           NOP
0EC2 0891           LOAD DEC. #
```

```
0EC4 C490 00F0     PICK OFF BITS 8-11
0EC8 C870 000A     LOAD A
0ECC CC90 0004     SHIFT RIGHT 4 BITS
0ED0 0C87          MULT SECOND DIGIT BY A
0ED2 0A39          ADD THIS TO SUM
0ED4 0891          LOAD DEC #
0ED6 C490 000F     PICK OFF BITS 12-15
0EDA 0A39          ADD THIS TO SUM
0EDC C890 2710     LOAD 2710
0EE0 0C80          MULT FIFTH DEC. DIGIT BY 2710
0EE2 0A39          ADD THIS TO SUM
0EE4 4030 0A08     SAVE HEX. #
0EE8 030F          RETURN
```

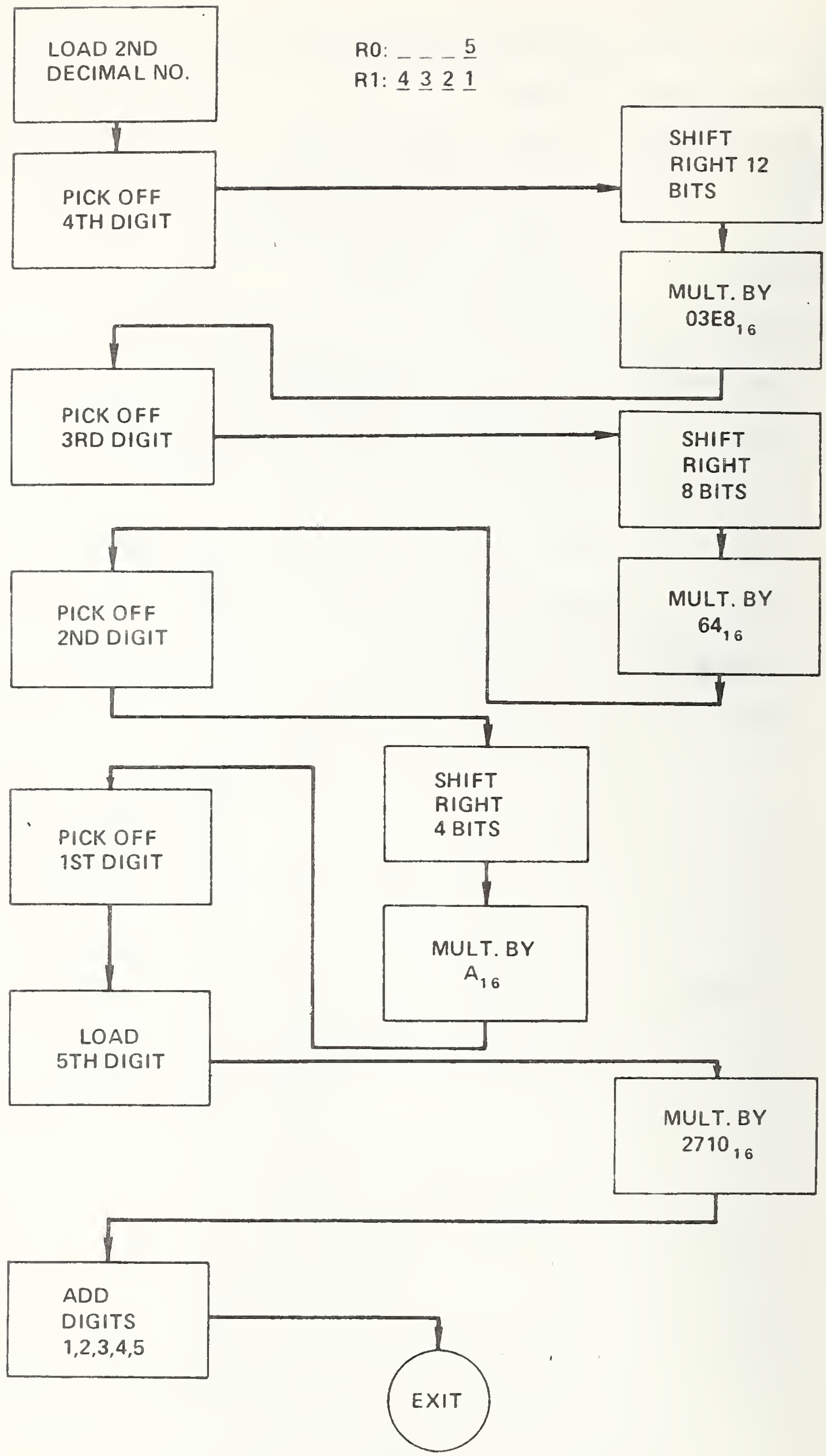


FIGURE 4-7. DECIMAL-TO-HEXIDECIMAL FLOWCHART

4.21 *Wait Subroutine for TTY

(Call on R8)

```
0EEC 9D19           SENSE STATUS OF TTY
0EEE 4280 0EEC      IF BUSY SENSE AGAIN
0EF2 0308           IF NOT BUSY RETURN TO CALL
>
```

4.22 *Decimal Point Typeout

(Call on R9)

```
0EF6 0200           NOP
0EF8 4090 0F0E      SAVE R9
0EFC 4180 0EEC      WAIT
0F00 C830 00AE      LOAD DEC. PT. CODE
0F04 9A13           OUTPUT DEC. PT.
0F06 0200           NOP
0F08 4890 0F0E      RESTORE R9
0F0C 0309           RETURN
0F0E 0DDC           STORAGE
>
```

4.23 Storage Table for Exciter Calibration Factors (Hexidecimal)

This table stores the calibration factors for the current exciter performing a calibration. These hexidecimal numbers are stored in this table by either START 1 (Section 4.13) or START 2 (Section 4.14).

Storage Table for Exciter Calibration Factors (Hexidecimal)

```
0F54 07D4
0F56 07D4
0F58 07D4
0F5A 07D4
0F5C 07D4
0F5E 07D4
0F60 07D4
0F62 07D4
0F64 07D4
0F66 07D4
0F68 07D3
0F6A 07D6
0F6C 07D8
0F6E 07DA
0F70 07DC
0F72 07DD
0F74 07E2
0F76 07E4
0F78 07E7
0F7A 07E7
0F7C 07E7
0F7E 07EE
0F80 07ED
0F82 07D2
0F84 07D4
0F86 07D6
0F88 07D8
0F8A 07DB
0F8C 07DF
0F8E 07E4
0F90 07E9
0F92 07EF
0F94 07F5
0F96 07FC
0F98 0803
0F9A 080B
0F9C 0813
>
```

4.24 Shut Off Oscillator, Ring Bell, Start Over

This routine shuts off the oscillator and rings the teletypewriter bell five times to indicate an end of test. The program is routed here after the last frequency test point in the Data Block has been completed or after a STOP command has been entered on the TTY.

Shut Off Oscillator, Ring Bells, Start Over

```
0F9E C810 0002     LOAD TTY DEVICE ADDRESS INTO R1
0FA2 DE10 09B2     PUT TTY IN WRITE MODE
0FA6 4200 0000     NOP
0FAA 0B22          CLEAR R2
0FAC DA70 0C7A     TURN OFF OSCILLATOR
0FB0 DA70 0C7B
0FB4 DA70 0C7C
0FB8 DA70 0C7D
0FBC 4180 0EFC     SENSE STATUS OF TTY
0FC0 DA10 0E60     RING BELL
0FC4 4180 0EFC     S S
0FC8 DA10 0E60     RING BELL
0FCC 4180 0EFC     S S
0FD0 DA10 0E60     RING BELL
0FD4 4190 0E78     DELAY
0FD8 4180 0EEC     S S
0FDC DA10 0E60     RING BELL
0FE0 4180 0EEC     S S
```

```
OFE4 DA10 0E60    RING BELL
OFE8 4180 0EEC    S S
OFEC DA10 09B5    OUTPUT LINE FEED TO TTY
OFF0 0BFF         CLEAR RF
OFF2 0B 66        CLEAR R6
OFF4 4060 09C4    CLEAR STORAGE AREA OF CAPACITOR CODE INDES
OFF8 C200 OFFC    LPSW
OFFC 0000         DISABLE INTERUPTS CODE
OFFE 01D0         TRANSFER TO RANGE TEST


1000 FFFF FFFF
1004 FFFF FFFF
1008 FFFF FFFF
100C FFFF 0200
>
```

4.25 *Delay for DVM Subroutine

Call on R2

This subroutine is used by the Read Data Subroutine (Section 4.39). Its purpose is to delay the sequential Read instructions of the computer. It was found that without this delay between Read instruction, errors would occur in the recorded data. This was due to timing differences (supposedly) between the DVM and the computer.

*Delay for DVM Subroutine (Used by DVM Read Program)

```
100E 0200
1010 0BBB           CLEAR RB
1012 CAB0 0001      ADD 1 TO RB
1016 C5B0 0100      COMPARE RB TO 0100
101A 4230 1012      IF NOT = GO ADD 1 TO RB
101E 0302           RETURN TO CALL
>
```

4.26 Halt, Stop, or Continue

After the completion of each test point, this routine checks for any operator input to the teletypewriter. This routine allows the operator to type an S or H after the frequency has been typed. This is read and stored in core. This routine checks this and alters the direction of the program accordingly.

S:	STOP	The STOP command halts the program, leaves the exciter energized, and waits for an execute by the operator. When the execute button is pushed, the program is started over at the first test point in the Data Block.
H:	HALT	The HALT command halts the program, leaves the exciter energized, and waits for an execute by the operator. When the execute button is pushed, the program is started at the next test point in line to be executed.

Any character other than H or S is presently ignored by the program. This can be expanded to include other interactions with the operator and computer. For example, a command could be given to go back one or two test points and repeat them.

Halt, Stop, or Continue (Comes Here from OE12)

```
1020 C830 0002     LOAD TTY DEVICE ADDRESS INTO R3

1024 DE30 09B0     OUTPUT TO TTY: DISABLE,BLOCK, AND READ

1028 4200 0000     NOP

102C 0B88          CLEAR R8

102E 0200          NOP

1030 9D34          SENSE STATUS OF TTY INTO R4

1032 0844          LOAD R4 TO SET CONDITION CODE

1034 4330 1044     IF ANY INPUT GO TO READ

1038 4890 09C2     LOAD ANY TTY DATA TAKEN AFTER FREQ. PRINTOUT

103C 4080 09C2     CLEAR OUT STORAGE AREA FOR TTY DATA

1040 4300 1046     GO TO TEST FOR H OR S
```

```
1044 9B39           READ DATA FROM TTY
1046 C590 0048      IS IT H?
104A 4330 1062      IF SO, GO TO HALT ROUTINE
104E C590 0053      IS IT S?
1052 4330 1076      IF SO, GO TO STOP ROUTINE
1056 4300 106E      IF NEITHER, CONTINUE
105A F000 0000
105E 0B66           CLEAR R6 TO START AT FIRST TEST FREQ,
1060 0200           NOP
1062 C200 1066      LPSW
1066 8000 08AE      RETURN TO 08AE AND STOP
106A FFFF
106C FFFF
106E C200 1072      COMES HERE IF INPUT IS NEITHER H OR S
                       LPSW
1072 0000
1074 08AF           DISABLE INTERUPTS AND GO TO 08AE
1076 0B66           **** STOP ROUTINE ****   CLEAR R6 TO START OVER
1078 DE30 09B2      PUT TTY IN WRITE MODE
107C  DA30 09B5     WRITE DATA TO TTY  LINE FEED
1080 0813           LOAD R3 INTO R1 (TO PUT DEVICE ADDRESS IN R1)
1082 4180 0EEC      BAL TO WAIT FOR TTY
1086 C200 108A      LPSW
108A 8000 0FA6      DISABLE INTERUPTS, STOP AT OFAC (RING BELLS,ETC.)
>
```

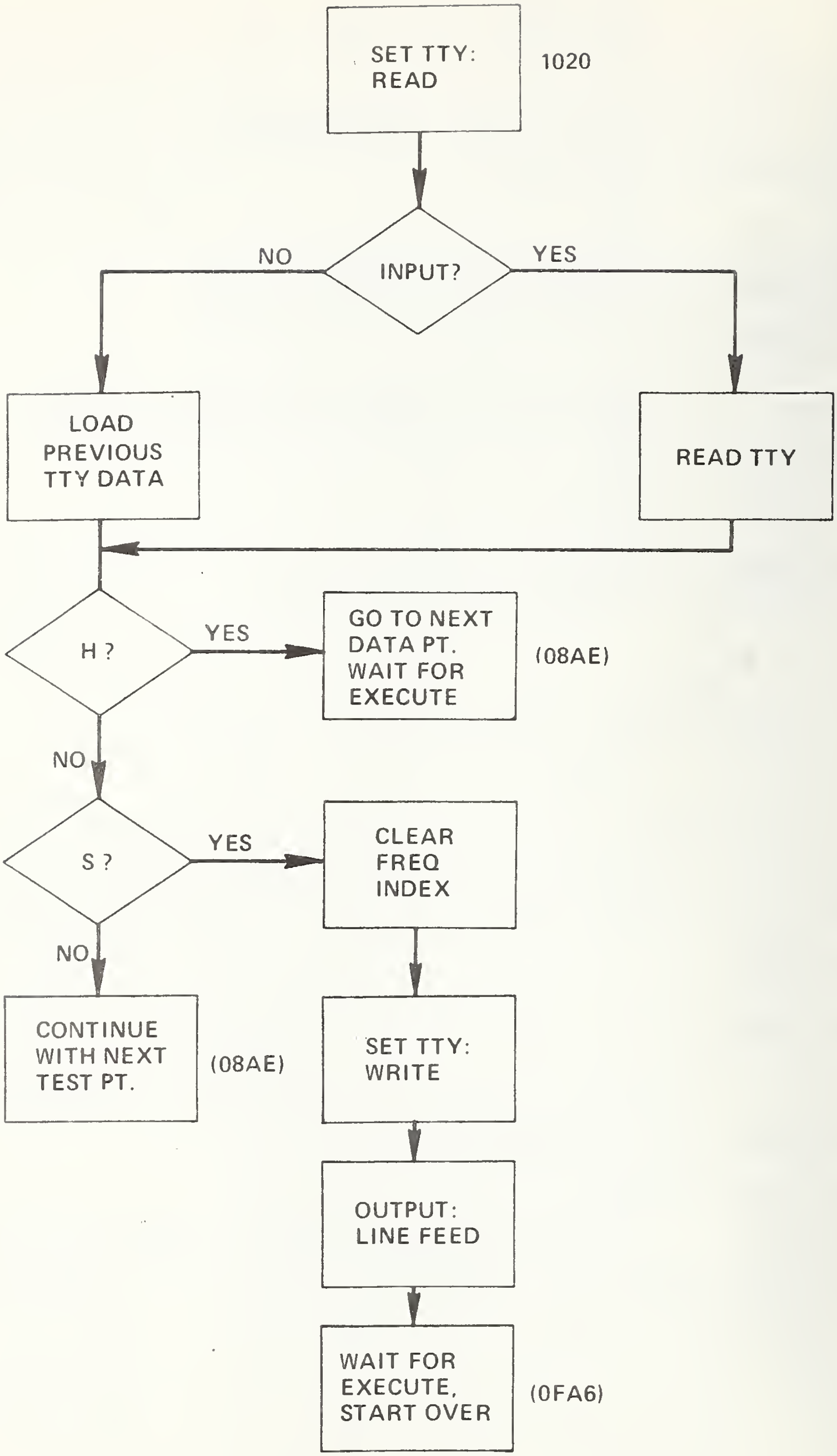


FIGURE 4-8. HALT, STOP, OR CONTINUE FLOWCHART

4.27 *Constant Acceleration Closed Loop Control Subroutine

Call on R4

This is a closed-loop control of acceleration level. The standard accelerometer voltage output is read by the DVM and compared to column 6 (exciter 2) or column 8 (exciter 1) of the 1400 Data Block. These columns give the desired voltage levels for the test point. Any deviation of standard accelerometer voltage from desired level over one millivolt triggers the program to trim the oscillator to obtain the desired value. An upper limit is set on the number of trimming operations to prevent an endless loop in case the desired value cannot be obtained. The trimmed oscillator voltage is stored in the DATA BLOCK at column 4 (exciter 1) or column 3 (exciter 2).

*Constant Acceleration Closed Loop Control

```
10F0 0B88            CLEAR R8
10F2 4080 09C6       STORE 0000 IN LOOP COUNTER
10F6 4040 09DE       SAVE R4
10FA 41E0 06D8       BAL TO SAVE REGISTERS
10FE C200            NOP
1100 0B88            CLEAR R8
1102 0BBB            CLEAR RB
1104 CAB0 0001       ADD 1 TO RB
1108 C560 0004       COMPARE R6(INDEX COUNTER) TO 4 (10 HZ.)
110C 4330 1118       IF = , GO TO 10 HZ. PROGRAM
1110 4300 13CC       GO TO SET SPEED OF DVM READINGS
1114 4300 111C
1118 C5B0 0500       COMPARE COUNTER OT 0500
111C 4230 1104       IF NOT = CONTINUE COUNTING
1120 CA80 0001       ADD 1 TO R8
```

```
1124 DE30 0B25      OUTPUT CONVERT DVM

1128 C580 000A      HAS THE DVM TAKEN 10 READINGS?

112C 4230 1102      IF NOT, GO READ ANOTHER TIME

1130 9F88           CLEAR INTERUPTS

1132 4300 1146      BRANCH

1136 F000 0000

113A F000 0000

113E F000 0000

1142 F000 0000

1146 9F95           ACKNOWLEDGE INTERUPTS

1148 0539           IS IT THE DVM?

114A 4330 1152      IF SO CONTINUE

114E 4300 1146      IF NOT, GO BACK TO AIR

1152 0B77           CLEAR R7
               READS VOLTAGE INTO CORE,ROUNDS OFF
               TO NEAREST MV,LEAVES OBS. MV. IN R1

1154 4200 0000      NOP

1158 C890 0005      LOAD BXLE LIMIT REGISTER FOR READING DATA

115C C880 0001      LOAD BXLE INCREMENT REGISTER

1160 DB37 129C      READ DATA INTO CORE STARTING AT 129C

1164 4120 1010      DELAY

1168 C170 1160      BXLE

116C 4200 0000      NOP

1170 4200 0000      NOP

1174 D3A0 129D      LOAD FIRST DIGIT OF DATA INTO RA

1178 D3B0 129E           2'ND                      RB
```

```
117C D3C0 129F          3'RD                     RC
1180 D3D0 12A0          4'TH                     RD
1184 D3E0 12A1          5'TH                     RE
1188 4190 1300     BAL TO ROUNDOFF ROUTINE
118C 0B11          CLEAR R1
118E 0B00          CLEAR R0
1190 931A          LOAD FIRST DIGIT INTO R1
1192 C410 C00F     PICK OFF BITS 12-15
1196 CD10 000C     SHIFT LEFT 12 BITS
119A 0B00          CLEAR R0
119C 0200          NOP
119E 082B          LOAD 2'ND DIGIT
11A0 C420 000F     PICK OFF BITS 12-15
11A4 CD20 0008     SHIFT LEFT 8 BITS
11A8 0A12          ADD FIRST AND SEC. DIGIT,PUT INTO R1
11AA 933C          LOAD 3'RD DIGIT INTO R3
11AC C430 000F     PICK OFF BITS 12-15
11B0 CD30 0004     SHIFT LEFT 4 BITS
11B4 0A13          ADD ON 3'RD DIGIT TO R1

           GETS DESIRED MV FROM TABLE,CALCULATES
           ΔMV, CALCULATES ΔV FOR OSCILLATOR

11B6 CC10 0004     SHIFT RIGHT 4 BITS (MV OBS. NOW IN R1)
11BA 0B88          CLEAR R8
11BC C8A0 0002     LOAD 2 INTO RA
11C0 4190 0E9A     BAL TO CONVERT OBS. VALUE TO HEX. (BACK IN R3)
```

```
11C4 48C0 09C4     LOAD TABLE OF VALUES INDEX INTO RC
11C8 48B0 09C0     LOAD SHAKER FLAG (ADDRESS OF TABLE) INTO RB
11CC CAB0 0002     ADD 2 TO THIS ADDRESS TO GET MV LOCATION
11D0 CBC0 0010     SUBTRACT 10 FROM RC TO GET CORRECT INDEX
11D4 40B0 11DA     STORE MV. TABLE ADDRESS BELOW
11D8 481C 140A     LOAD MV. DESIRED INTO R1
11DC 0843          LOAD OBS. VALUE(HEX.) INTO R4
11DE CB60 0004     SUBTRACT 4 FROM INDEX
11E2 4190 0E9A     BAL TO CONVERT DES. VALUE TO HEX.
11E6 0823          PUT MV. DESIRED IN R2 (HEX.)
11E8 0B24          MV. DESIRED-MV. OBSERVED = ΔMV.
11EA 0200          NOP
11EC CB20 0001     SUBTRACT 1 FROM ΔMV.
11F0 4320 12AC     IF NOT PLUS GO TO TEST FOR 0
11F4 4300 1284     GO TEST LOOP COUNTER FOR MAX. NO. OF TRIES
11F8 4190 13E4     BAL TO CALCULATE ΔV FOR OSCILLATOR (OUT ON R5)
11FC 4810 09C0     LOAD SHAKER FLAG
1200 C510 1408     IS IT SHAKER 2?
1204 4330 1210     IF SO, BRANCH
1208 481C 1406     LOAD OSC. VOLTAGE INTO R1
120C 4300 1214     BRANCH
1210 481C 1404     COMES HERE FOR SHAKER 2 , LOAD OSC. VOLTAGE IN R
1214 4050 09CA     STORE R5 (ΔV FOR OSCILLATOR)
1218 4080 09C6     STORE LOOP COUNTER
121C 4190 0E9A     BAL TO CONVERT VOLTAGE (OLD) TO HEX.
1220 4850 09CA     RESTORE R5 (ΔV FOR OACILLATOR)
```

CALCULATES CORRECTED VOLTAGE FOR
OSCILLATOR,OUTPUTS CODE TO OSC

```
1224 CD30 0004      SHIFT LEFT 4 BITS
1228 0A35           ADD ΔV TO OLD VOLTAGE VALUE
122A 0200           NOP
122C CC30 0004      SHIFT RIGHT 4 BITS
1230 4190 1370      BAL TO CONVERT NEW VOLTAGE TO DECIMAL (BACK IN R2)
1234 4810 09C0      LOAD SHAKER FLAG
1238 C510 1408      IS IT SHAKER 2
123C 4330 1248      IF SO GO TO 1248
1240 402C 1406      STORE OSC CODE IN TABLE
1244 4300 124C      BRANCH
1248 402C 1404      COMES HERE IF SHAKER 2,STORE OSC CODE IN TABLE
124C 4190 16C0      BAL TO CONVERT TO ORIGINAL OSC. CODE FORMAT
1250 C870 000F      LOAD OSC DEVICE NO.
1254 0B22           CLEAR R2
1256 0BAA           CLEAR RA
1258 9D75           SENSE STATUS
125A 0552           IS OSC. BUSY?
125C 4230 1258      IF SO, SENSE AGAIN
1260 DA76 0C8C      WRITE DATA TO OSC.
1264 CA60 0001      ADD 1 TO INDEX REGISTER
1268 CAA0 0001      ADD 1 TO RA
126C C5A0 0004      HAVE 4 BYTES BEEN WRITTEN?
1270 4230 125A      IF NOT, WRITE AGAIN
1274 4200 0000      NOP
```

```
1278 C830 00CA      LOAD DVM DEVICE ADDRESS
127C 4300 1100      GO READ VOLTAGE AGAIN
1280 4300 12C4      GO TO RESTORE R4 AND EXIT
1284 CA20 0001      ADD ONE TO R2(TO RESTORE TO ORIG. VALUE)
1288 4880 09C6      LOAD LOOP COUNTER

          LOOP COUNTER TEST,ΔMV TESTED FOR
          -1 CONDITION,PREPARE TO LEAVE SUB.

128C CA80 0001      ADD 1 TO LOOP COUNTER
1290 C580 0020      COMPARE TO UPPER LIMIT
1294 4330 12BC      IF = , GO TO EXIT
1298 4300 11F8      GO BACK TO CALCULATE Δ MV.
129C ABB1 B4B0      STORAGE FOR VOLTAGE READINGS OF DVM
12A0 B1B3 0B11         "
12A4 0000
12A6 0000
12A8 F000 0000
12AC 4330 12BC      IF ΔMV. =0 EXIT
12B0 C520 FFFF      COMPARE ΔMV. TO -1
12B4 4330 12BC      IF =, EXIT
12B8 4300 11F4      IF NOT, CONTINUE
12BC CA60 0004       COMES HERE IF LOOP COUNTER IS IN LIMIT
                      ADD 4 TO INDEX
12C0 4300 1280      GO TO EXIT
12C4 4840 09DE      RESTORE R4 (BAL REGISTER FOR THIS SUBROUTINE)
12C8 0304           EXIT (RETURN TO CALL)
```

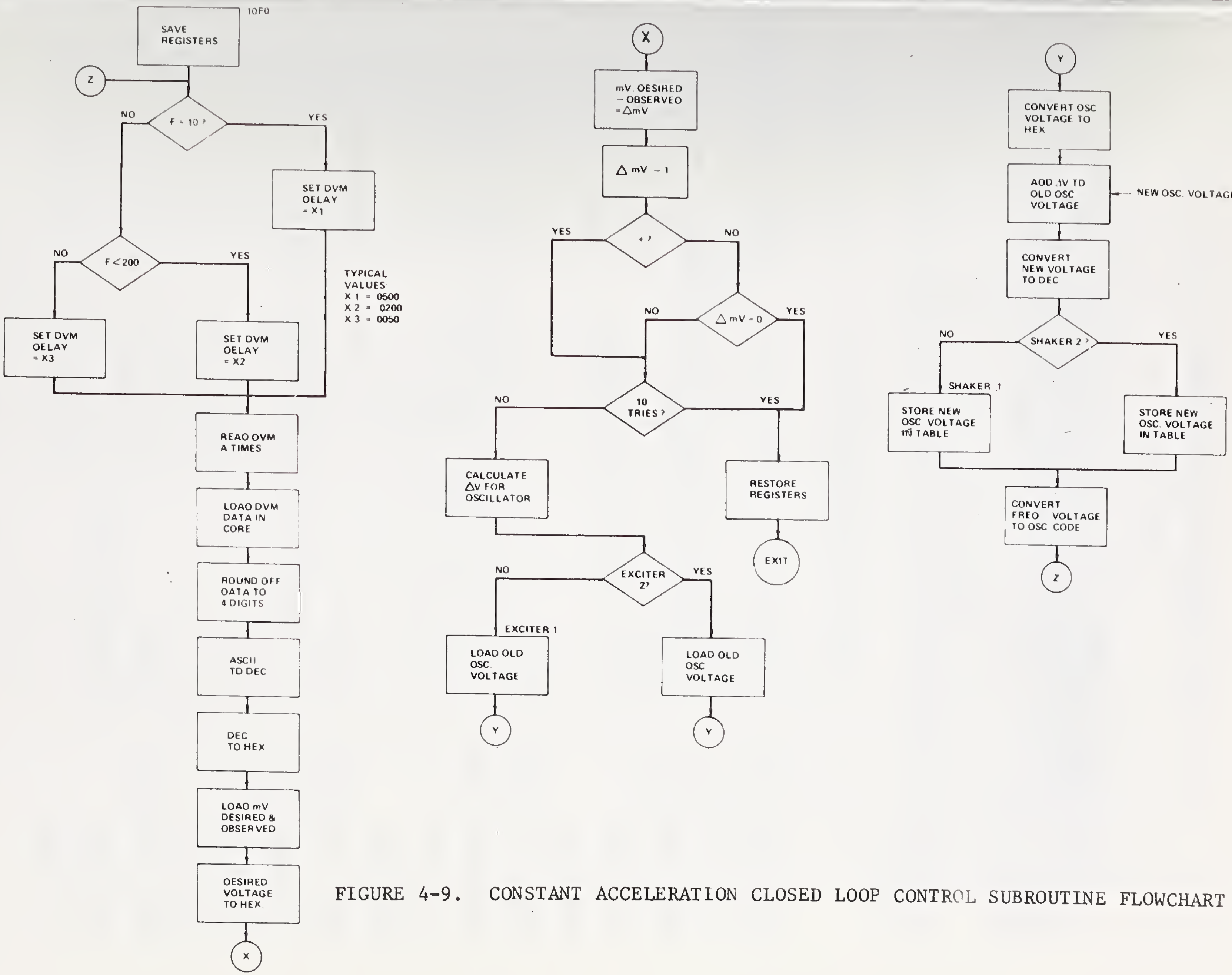


FIGURE 4-9. CONSTANT ACCELERATION CLOSED LOOP CONTROL SUBROUTINE FLOWCHART

4.28 *Scope Scaling Subroutine

Call on RA

This subroutine permits a relay (relay 16) in bank 1 to close and thereby activating the "seek" feature of the oscilloscope. This seek feature triggers the scope and automatically sets the time base and amplifier units for a constant display of number of cycles and amplitude of the display.

In this subroutine, the relay closes and opens twice. This is necessary sometimes because the first closing may not get a proper triggering.

Scope Scaling Subroutine

```
12D0 C820 00E0      LOAD RELAY CODE FOR RATIO I (RATIO I PROG. ENTERS
                                                              HERE)
12D4 C800 0071      LOAD RELAY BANK I DEVICE ADDRESS
                        (RATIO II PROG. ENTERS HERE)
12D8 CA40 0001      ADD I FOR SCOPE RELAY

12DC 9A02           WRITE DATA TO RELAY BANK I

12DE 9A04             "

12E0 4190 0E78      DELAY

12E4 C840 0000      LOAD RELAY CODE TO DEACTIVATE SCOPE RELAY

12E8 9A02           DEACTIVATE SCOPE RELAY

12EA 9A04             "

12EC 4190 0E78      DELAY

12F0 C840 0001      LOAD CODE TO ACTIVATE SCOPE RELAY

12F4 9A02           ACTIVATE SCOPE RELAY

12F6 9A04             "

12F8 4190 0E78      DELAY

12FC 030A           RETURN TO CALL

12FE 0B9A

>
```

4.29 *Round Off Subroutine (for Constant Acceleration)

Call on R9

This subroutine takes a five digit number (base ten) in registers RA, RB, RC, RD, RE and rounds the number to a four digit number in registers RA, RB, RC, RD. This subroutine is used in the Constant Acceleration routine (Section 4.27).

*Round Off Subroutine

```
1300 C4A0 000F     PICK OFF BITS 12-15 OF RA
1304 C4B0 000F                "          RB
1308 C4C0 000F                "          RC
130C C4D0 000F                "          RD
1310 C4E0 000F                "          RE
1314 CBD0 0005     SUBTRACT 5 FROM RD
1318 4220 1320     IF PLUS GO TO 1320
131C 4210 1354     IF MINUS GO TO 1354
1320 CBC0 0009     SUBTRACT 9 FROM RC
1324 4210 1350     IF MINUS GO TO 1350
1328 CBB0 0009     SUBTRACT 9 FROM RB
132C 4210 1344     IF MINUS GO TO 1344
1330 4200 0000     NOP
1334 CAA0 0001     ADD 1 TO RA
1338 C8B0 0000     LOAD 0 INTO RB
133C C8C0 0000     LOAD 0 INTO RC
1340 4300 1354     GO TO 1354
```

```
1344 CAB0 000A      ADD A TO RB (1 PLUS 9 THAT WAS SUBTRACTED)
1348 C8C0 0000      LOAD 0 INTO RC
134C 4300 1354      GO TO 1354
1350 CAC0 000A      ADD A TO RC
1354 4200 0000      NOP
1358 0309           RETURN TO CALL
```

4.30 *Hexidecimal-to-Decimal Subroutine

Call on R9

Input requirements: Hexidecimal number in R3.

Output requirements: Up to four digits decimal number in R2.

Hexidecimal-to-Decimal Subroutine
Converts a Four Digit Hex Number to a Decimal Number In on R3, Out on R2

```
1370 C880 03E8      LOAD 03E8 INTO R8
1374 C850 0064      LOAD 0064 INTO R5
1378 C870 000A      LOAD 000A INTO RA
137C 0B22           CLEAR R2
137E 0D28           DIVIDE HEX NO. BY 03E8,QUOT IN R3, REM. IN R2
1380 08A3           PUT QUOT. IN RA (1'ST DECIMAL DIGIT)
1382 0832           PUT REM. IN R3
1384 0B22           CLEAR R2
1386 0D25           DIVIDE THIS BY 64
1388 08B3           PUT QUOT IN RB  (2'ND DIGIT)
138A 0200           NOP
138C 0832           PUT REM. IN R3
138E 0200           NOP
1390 0B22           CLEAR R2
1392 0D27           DIVIDE THIS BY A
1394 08E3           PUT QUOT. IN RE  (3'RD DIGIT)
1396 08D2           PUT REM. IN RD (4'TH DIGIT)
1398 CDA0 000C      SHIFT RA LEFT 12 BITS
```

```
139C CDB0 0008      SHIFT RB LEFT 8 BITS
13A0 CDE0 0004      SHIFT RE LEFT 4 BITS
13A4 0AAB           ADD RB TO RA
13A6 0AAE           ADD RE TO RA
13A8 0AAD           ADD RD TO RA
13AA 082A           LOAD RA INTO R2
13AC 0309           RETURN TO CALL
>
```

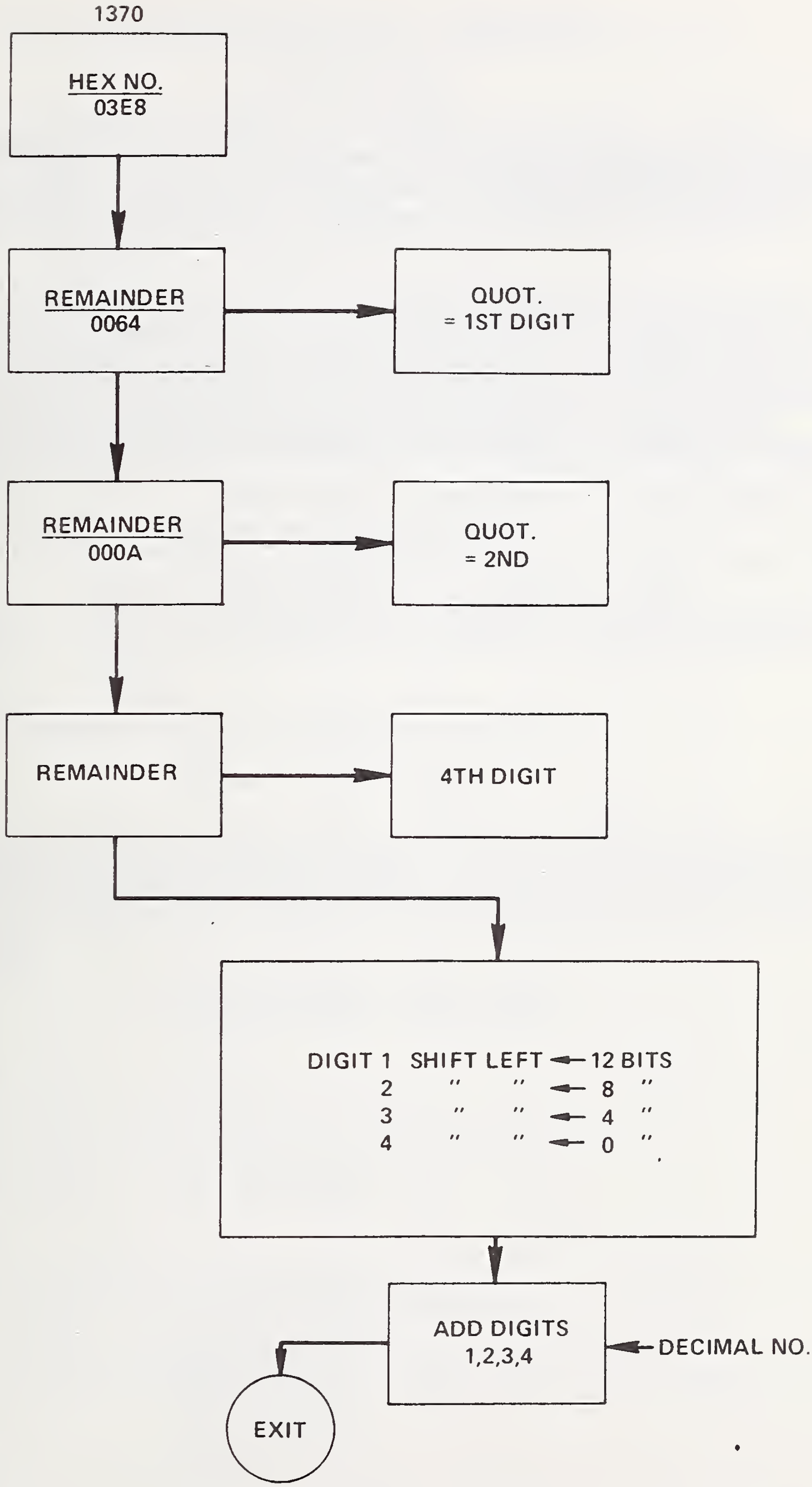


FIGURE 4-10. HEXIDECIMAL-TO-DECIMAL SUBROUTINE FLOWCHART

4.27(con't) *Constant Acceleration Closed Loop Control

Speed Set For DVM

Note: The following is part of Section 4.27, Constant Acceleration Closed Loop Control.

```
13CC C560 001C     FREQ: 200 HZ.
13D0 4320 13DC     BRANCH ON NEG. (<200 HZ.)
13D4 C5B0 0020     COMPARE  COUNTER TO UPPER LIMIT
13D8 4300 1114     RETURN TO MAIN PROG.
13DC C5B0 0200     COMPARE COUNTER TO UPPER LIMIT FOR F<200 HZ.
13E0 4300 1114     RETURN TO MAIN PROG.
```

4.31 *Calculate Delta V Subroutine

Call on R9

This subroutine is used by the Constant Acceleration Control subroutine (Section 4.27) to calculate the trimming voltage for output to the oscillator. At present, the delta millivolt of the standard accelerometer is multiplied by a constant to trim the oscillator to approximately 70 percent of the desired value.

From figure 1-1, it can be seen that approximately two volts is needed for an acceleration of 10 g on exciter 2. A typical value for the standard accelerometer voltage output is 140 mV for one NBS exciter and 70 mV for another exciter at 10 g. For the 140 mV exciter, a change of 14 mV of oscillator voltage will produce one millivolt change in accelerometer voltage. The equation used to compute the trim voltage for this exciter is:

$$\text{Delta V} = \text{Delta mV} \cdot (10)$$

where Delta V: amount of trim voltage for oscillator, and

Delta mV: millivolt standard accelerometer differs from desired value.

For the 70 mV exciter (10 g) the equation used is:

$$\text{Delta V} = \text{Delta mV} \cdot (8).$$

The constant in these equations will vary with the exciter being used in the system and the power amplifier gain. A value for this constant should be used to trim the voltage output of the oscillator from 60 to 80 percent of the desired difference per cycle so as not to overshoot the final desired value.

Calculate Delta V Subroutine

```
13E4 C850 000A      LOAD A
13E8 0C42 48A0      DELTA MV*A  LOAD SHAKER FLAG
13EC C9C0 C5A0      IS IT SHAKER 2?
13F0 1408
13F2 4230 13FC      IF NOT EQUAL, BRANCH
13F6 C850 0008      LOAD 8 INTO R5
13FA 0C42           DELTA MV*8
13FC 0309           RETURN
```

4.32 Data Block Control Constants (1400 Tables)

This table contains the control parameters for the test points. Each row represents a test point and contains the frequency, oscillator voltage for the first try, code for selecting the proper capacitance match, and the desired millivolt output from the standard accelerometer. This table is set up for two exciters, but could be expanded to contain data for additional exciters that may be added to the setup. The code for the capacitor match is explained in Section 4.15.1.

The program selects the test point corresponding to the data in the first row. After completing this test point, the second test point corresponding to the second row is processed, etcetera, until the last test point is finished. Upon finishing the last test point, the program starts over again at the first test point.

The data in columns 3 and 4 are updated each time a calibration is run, as explained in Section 4.27.

4.33 *Voltage/Frequency Data Modified to Proper Oscillator Code

Call on R9

This subroutine converts the 1400 DATA BLOCK data columns 1, 2, 3, 4 which will program the oscillator. See Section 4.15.2 for a description of the proper oscillator format.

Input requirements: R6 must contain proper index number (in hexidecimal) for the desired frequency. The index number is four·(test point number). The test point number starts at zero and ends at N. In the present program N = 38.

Output requirements: The properly coded output is stored in the Storage Table for Oscillator Code (OC8C).

This subroutine is called for each test point processed. It does not set up the entire OC8C table with one call instruction.

Data Block Control Constants

	Frequency		Voltage		Cap.	mV.	Cap.	mV.
			← Exciter					→
			1	2	2	2	1	1
1400	0010	0001	0029	0043	0001	0028	0001	0028
1410	0015	0001	0071	0088	0001	0070	0001	0070
1420	0030	0001	0147	0149	0001	0140	0001	0140
1430	0050	0001	0155	0144	0001	0140	0001	0140
1440	0099	7001	0167	0155	0001	0140	0001	0140
1450	0099	7001	0083	0076	0001	0070	0001	0070
1460	0099	7001	0032	0029	0001	0028	0001	0028
1470	0199	7001	0175	0156	0001	0140	00A6	0140
1480	0299	5001	0187	0171	00A6	0140	0054	0140
1490	0399	5001	0185	0175	020A	0140	00A4	0140
14A0	0498	5001	0187	0187	00C2	0140	0208	0140
14B0	0598	4001	0188	0198	00CC	0140	0210	0140
14C0	0697	3001	0191	0208	00E4	0140	0260	0140
14D0	0797	2001	0191	0220	00C4	0140	0120	0140
14E0	0897	0001	0193	0233	0204	0140	0420	0140
14F0	0100	0002	0191	0247	0088	0140	00C0	0140
1500	0149	9002	0194	0301	0060	0140	0080	0140
1510	0169	9002	0196	0321	0120	0140	0900	0140
1520	0199	9002	0200	0355	0020	0140	1200	0140
1530	0199	9002	0098	0172	0020	0070	1200	0070
1540	0199	9002	0039	0069	0020	0028	1200	0028
1550	.0249	7002	0197	0425	0440	0140	8C00	0140
1560	0299	7002	0206	0473	0080	0140	4400	0140
1570	0349	6002	0205	0000	4100	0140	0000	0000
1580	0399	6002	0208	0000	0A00	0140	0000	0000
1590	0449	1002	0214	0000	0200	0140	0000	0000
15A0	0499	0002	0214	0000	2C00	0140	0000	0000
15B0	0548	9002	0219	0000	1400	0140	0000	0000
15C0	0598	9002	0262	0000	0400	0140	0000	0000
15D0	0648	7002	0229	0000	0400	0140	0000	0000
15E0	0698	6002	0241	0000	2800	0140	0000	0000
15F0	0748	5002	0246	0000	4800	0140	0000	0000
1600	0798	4002	0274	0000	8800	0140	0000	0000
1610	0848	4002	0246	0000	8800	0140	0000	0000
1620	0898	3002	0270	0000	0800	0140	0000	0000
1630	0948	2002	0250	0000	0800	0140	0000	0000
1640	0998	2002	0259	0000	0800	0140	0000	0000

```
16C0 C8B0 0004      LOAD 4 INTO RB
16C4 0CA6           R6*4   RESULT IN RA,RB
16C6 081B           LOAD RB INTO R1
16C8 4200 0000      NOP
16CC 4050 16F8      STORE R5
16D0 4850 09C0      LOAD SHAKER FLAG INTO R5
16D4 C550 1408      IS IT 201?
16D8 4330 16E0      IF SO , GO TO 16E0
16DC 4300 16EC      IF NOT, GO TO 16EC
16E0 4831 1405      LOAD VOLTAGE FROM TABLE INTO R3
16E4 4841 1404                 "
16E8 4300 1700      BRANCH
16EC 4831 1407      LOAD VOLTAGE FROM TABLE INTO R3
16F0 4841 1406                 "
16F4 4300 1700      BRANCH
16F8 0072           STORAGE FOR R5
16FA 0000
16FC 4200 0000      NOP
1700 4200 0000
1704 0200
1706 0200
1708 4200 0000
```

```
170C 08A3           LOAD VOLTAGE INTO RA
170E 0200
1710 C4A0 00F0      MASK OFF 3'RD DIGIT  (.1 VOLTS)
1714 CDA0 0008      SHIFT LEFT 8 BITS
1718 08BA           LOAD VOLTAGE INTO RB
171A 08A3           LOAD VOLTAGE INTO RA
171C 0200           NOP
171E C4A0 000F      MASK OFF 4'TH DIGIT  (.01 VOLTS)
1722 CDA0 0008      SHIFT LEFT 8 BITS
1726 0ABA           ADD THIS TO RB
1728 D3A1 1402      LOAD FREQ. INFORMATION INTO RA
172C C4A0 00F0      MASK OFF 3'RD DIGIT  (.1 HZ.)
1730 0ABA           ADD THIS TO RB
1732 08A4           LOAD VOLTAGE INTO RA
1734 0200           NOP
1736 C4A0 0F00      MASK OFF 2'ND DIGIT  (VOLTS)
173A CCA0 0008      SHIFT RIGHT 8 BITS
173E 0ABA           ADD THIS TO RB
1740 40B6 0C8C      STORE RB IN 0C8C (INDEX R6)
1744 4200 0000      NOP
1748 4200 0000
174C 4200 0000
1750 D3A1 1401      LOAD FREQ. INFOR. INTO RA
1754 C4A0 00F0      MASK OFF 3'RD DIGIT (10 HZ.)
1758 CDA0 0008      SHIFT LEFT 8 BITS
```

```
175C 08CA           PUT THIS IN RC
175E D3A1 1401      LOAD SOME MORE CODE INTO RA
1762 C4A0 000F      MASK OFF 4'TH DIGIT (1 HZ.)
1766 CDA0 0008      SHIFT LEFT 8 BITS
176A 0ACA           ADD THIS TO RC
176C D3A1 1403      LOAD MORE CODE INTO RA
1770 C4A0 000F      MASK OFF 4'TH DIGIT  (MULTIPLIER)
1774 CDA0 0004      SHIFT LEFT 4 BITS
1778 0ACA           ADD THIS TO RC
177A D3A1 1400      LOAD MORE CODE INTO RA
177E C4A0 000F      MASK OFF 4'TH DIGIT (100 HZ.)
1782 0ACA           ADD THIS TO RC
1784 40C6 0C8E      STORE RC IN 0C8E (INDEX R6)
1788 C870 000F      LOAD OSC. DEVICE NO. IN R7
178C 0BAA           CLEAR RA
178E 0B00           CLEAR R0
1790 4850 16F8      RESTORE R5
1794 4200 0000      NOP
1798 4200 0000
179C 4200 0000
17A0 4200 0000
17A4 4200 0000
17A8 4200 0000
17AC 4200 0000
17B0 0309           RETURN TO CALL

NOTE****     NEW CODE NOW LOOKS LIKE THIS:

       ( .1V   .01V   .1HZ   1V   10 HZ   1HZ   X   100HZ  )
```

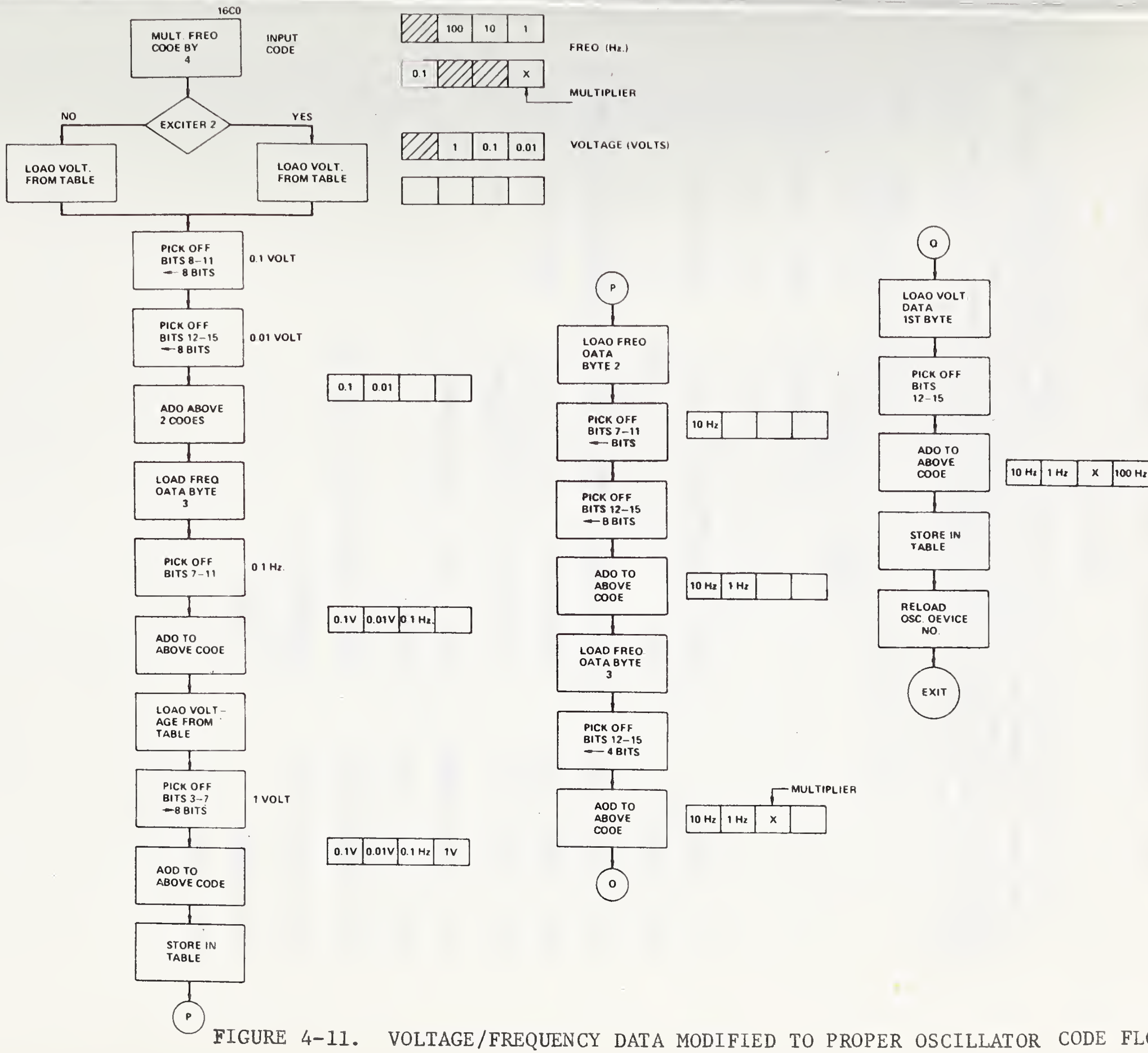


FIGURE 4-11. VOLTAGE/FREQUENCY DATA MODIFIED TO PROPER OSCILLATOR CODE FLOWCHART

4.34 *Set Slow/Fast Code for ac/dc Converters Subroutine

Call on RE

This subroutine sets up the code for either slow (frequency <200 Hz) or fast mode (frequency equal to or greater than 200 Hz) for the ac/dc converters. This mode is controlled by relay 7 (see Section 4.15.1). This subroutine has two entry points, one if RATIO I is being taken and one if Ratio II is being taken.

*Set Slow/Fast Code for ac/dc Converters Subroutine

Ratio I Program

```
17B 4 0836           LOAD R6 INTO R3
17B 6 0200           NOP
17B8 CB30 001C       SUBTRACT 1C FROM R3 (1C=CODE FOR 200 HZ.)
17BC 4220 17C4       BRANCH ON +
17C0 4300 17CC       CONTINUE IN SLOW MODE
17C 4 C820 00E2      LOAD 'FAST' CODE INTO R2
17C8 4300 17D0       BRANCH
17CC C820 00E0       LOAD 'SLOW'CODE INTO R2
17D0 C830 00CA       RESTORE R3
17D4 030E            RETURN
17D6 0200
```

PROGRAM FOR RII

```
17D8 0836          LOAD R6 INTO R3
17DA 0200          NOP
17DC CB30 001C     SUBTRACT 1C FROM R3
17E0 4220 17E8     BRANCH ON +
17E4 4300 17F0     CONT. IN SLOW MODE
17E8 C820 00C2     LOAD 'FAST' CODE INTO R2
17EC 4300 17F4     BRANCH TO EXIT
17F0 C820 00C0     LOAD 'SLOW' CODE INTO R2
17F4 C830 00CA     RESTORE R3
17F8 030E          RETURN
```

4.35 Store Calibration Data in Tables

This routine stores the calibration frequency and sensitivity in core as the test points are being performed. The locations set aside for this purpose are:

Exciter 1: 2C00-2E00
Exciter 2: 2E00-3000

The following examples illustrate the format of the stored data.

0000 0010 1009 0002	10 Hz, 10.09 mV/g
0000 1500 9988 0003	1500 Hz, 9.988 mV/g

The purpose of this storage is to transfer these data to magnetic tape for storage at the end of a test. Additional storage space is set aside as shown in table 4-3.

The data starting at 2B00 are entered into the computer by the Accelerometer Data Block Entry program (see Section 4.45). The data from 2C00-3000 are stored as the calibration program steps through the test points. The summary data starting at 3000 are read in by paper tape or typed in on the TTY. The summary data are these which are reported in a calibration report. It includes all data used to make up a complete calibration. Usually they are prepared by feeding all the data into a time-sharing computer program. This program combines the data by averaging techniques to provide final output report data.

The interferometer data consist of data on three piezoelectric exciters for each accelerometer calibrated. These data are kept on paper tape. It is desirable to keep not only the final summary, but all the data collected on the accelerometer on the magnetic tape files. These data can then easily be recalled for future checking and comparisons.

Store Calibration Data in Tables

```
1800 41E0 06E0     SAVE REGISTERS

1804 41E0 23FC     BAL TO DECODE FREQ. DATA

1808 0200          NOP

180A 081A          LOAD FIRST DIGIT (CAL. FACTOR OF TEST)

180C CD10 000C     SHIFT LEFT C BITS

1810 082B          LOAD 2'ND DIGIT
```

```
1812 0200             NOP
1814 4880 246C        LOAD FIRST FREQ. DIGIT
1818 CD20 0008        SHIFT LEFT 8 BITS
181C 083C             LOAD 3'RD DIGIT
181E 0200             NOP
1820 CD30 0004        SHIFT LEFT 4 BITS
1824 0A12             ADD R1 AND R2
1826 0A13             ADD R1 AND R3
1828 082D             LOAD 4'TH DIGIT
182A 0200             NOP
182C 0A12             ADD THIS TO ABOVE SUM
182E 0200             NOP
1830 C840 0002        LOAD 2
1834 0836             LOAD FREQ INDEX(R6)
1836 0200             NOP
1838 0C24             MULT. BY 2
183A 0200             NOP
183C 4890 246E        LOAD FRIQ. DIGITS 2-5
1840 082A             RELOAD FIRST DIGIT
1842 0200             NOP
1844 4230 1868        BRANCH ON NOT 0
1848 CD10 0004        (RA=0)  SHIFT LEFT 4 BITS
184C 4820 01AA        LOAD DIGIT 5
1850 0A12             ADD IT TO SUM
1852 0200             NOP
```

```
1854 4200 0000      NOP
1858 4200 0000
185C 4200 0000
1860 4200 0000
1864 4200 0000
1868 4840 09F0      LOAD DEC. PT. CODE(TELLS WHERE DEC. PT. IS LOCATED)
186C 4820 09C0      LOAD SHAKER ID
1870 C520 140C      SHAKER 2?
1874 4230 188C      IF NOT,BRANCH
1878 4013 2BFC      STORE TEST CAL. FACTOR (FOR SHAKER 1)
187C 4083 2BF8      STORE FREQ.
1880 4093 2BFA        "
1884 4043 2BFE      STORE DEC. PT. CODE
1888 4300 18A0      BRANCH
188C 4013 2DFC      STORE TEST CAL. FACTOR (FOR SHAKER 2)
1890 4083 2DF8      STORE FREQ. DATA
1894 4093 2DFA        "
1898 4043 2DFE      STORE DEC. PT. CODE
189C 4200 0000      NOP
18A0 41E0 0720      RESTORE REGISTERS
18A4 4300 18FC      GO TO DEC/ASCII ROUTINE
>
```

TABLE 4-3. Data File Format

2900	
.	Interferometer fringe disappearance data
.	
2AF0	2 4 6 8 A C E
	* * * * * * * * *
2B00	/file number/test number / xxx /
2B10	/pickup mfgr. /pickup s.n /
2B20	/pickup model number / xxx /
2B30	/amp. mfgr. /amp. s.n. /
2B40	/amp. model number / xxx /
2B50	/customer /
2B60	/date of calib /exciter 1/exciter 2/xxx /
2B70	/std. 1 /std. 2 /capacitance /
2B80	gain
.	"
.	
2BF0	"
2C00	Exciter 1 data
.	"
.	
2DF0	"
2E00	Exciter 2 data
.	"
.	
2EF0	"
3000	Summary Data (data reported in a calibration)
.	
.	
31F0	"

4.36 Decimal to ASCII Routine (5 digits)

This routine converts five digits of decimal code to five digits of ASCII code. This routine utilized subroutine Section 4.37.

Decimal to ASCII Routine (5 digits)
(In/Out on RA, RB, RC, RD, RE)

```
18FC 48E0 01AA     RESTORE FIFTH DIGIT
1900 080A          LOAD FIRST HEX. #
1902 0200
1904 4190 1954     BAL TO ASCII CONVERT
1908 08A2          PUT ASCII INTO RA
190A 0200
190C 080B          SAME FOR 2'ND #
190E 0200
1910 4190 1954
1914 08B2
1916 080C          SAME FOR 3'RD
1918 4190 1954
191C 08C2
191E 080D          SAME FOR 4'TH
1920 4190 1954
1924 08D2
1926 080E          SAME FOR 5'TH
1928 4190 1954
192C 08E2
```

```
192E 0200
1930 C810 0002     LOAD TTY DEVICE ADDRESS
1934 9D15          SENSE STATUS OF TTY
1936 0855          LOAD TO SET CC
1938 4330 1940     BRANCH ON ZERO
193C 4300 0DA0     BRANCH TO PRINTOUT
1940 9B19          READ TTY
1942 4090 09C2     STORE DATA
1946 4300 0DA0     BRANCH TO PRINTOUT
194A 0200
>
```

4.37 *Decimal -to-ASCII Subroutine (1 Digit)

Call on R9

This subroutine converts a one digit decimal number to ASCII code.

Input requirements: decimal number in R0

Output requirements: ASCII code in R2

*Decimal/ASCII Subroutine (1 Digit)

```
194C 4200 0000
1950 4200 0000
1954 0810        LOAD
1956 0200
1958 CB10 0009   R1-9=R1
195C 4220 1968   BRANCH ON +
1960 C600 00B0   OR
1964 0820        LOAD
1966 0309        RETURN
1968 C610 00C0   OR, MUST BE ALPHA CHAR.
196C 0821        LOAD
196E 0309        RETURN
```

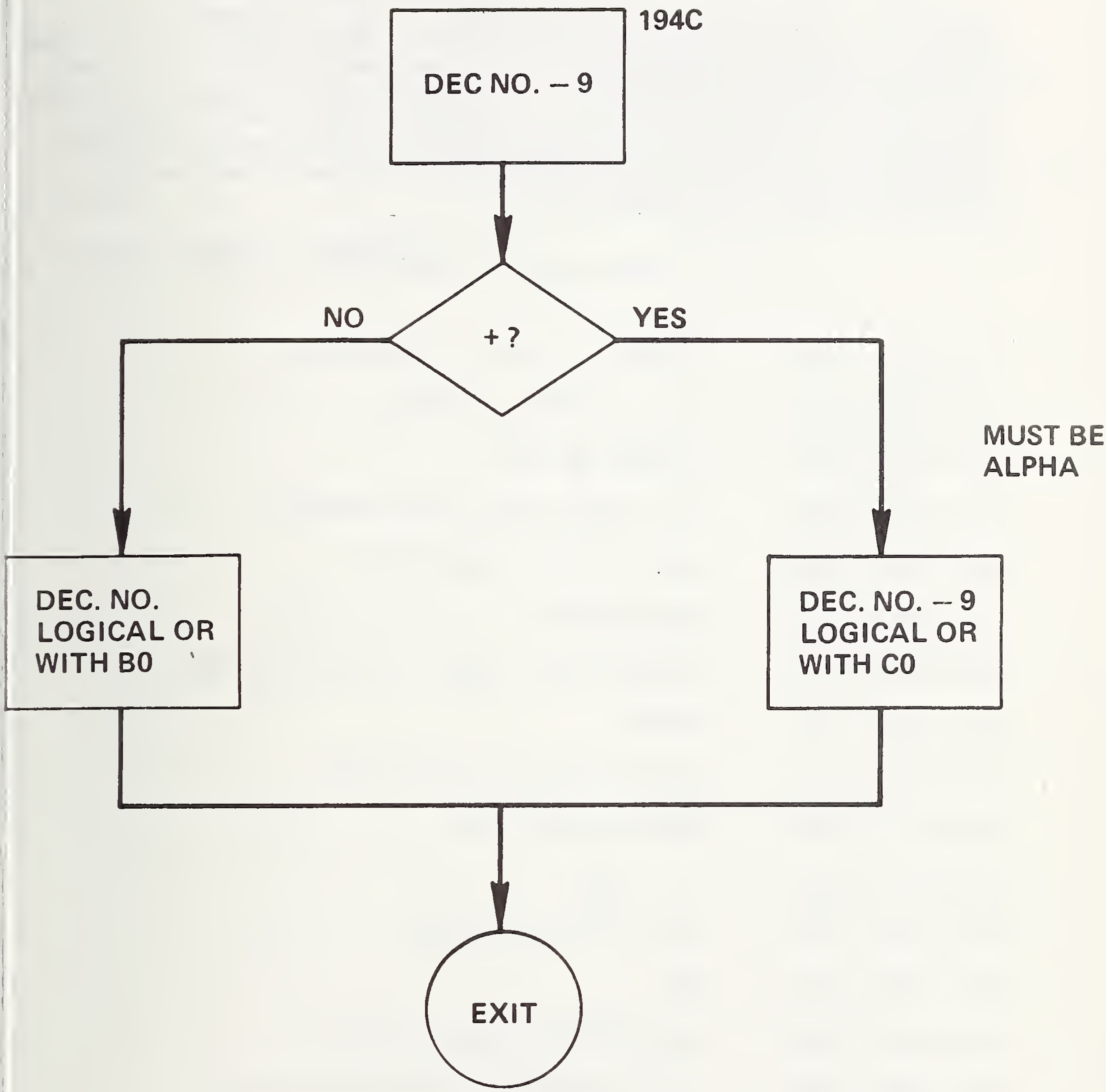


FIGURE 4-12. DECIMAL TO ASCII FLOWCHART

4.38 Check for Correct Range of Signal ac/dc Converter

This routine checks Ratio II for all nines (out of range) and if this condition if found, the code for the relays is set to program the signal converter for 10-volt range. The RANGE FLAG is set for 000A: the THRU FLAG is set for 0001. The THRU FLAG is set to indicate that this routine has been executed once and should not be repeated on the next pass for this test point. If the converter is set for 10-volt range, the program is transferred to read Ratio II again; if the one-volt range is sufficient, the program continues with the caloulation of the test accelerometer sensitivity (0ADC) in Section 4.16.

Check for Correct Range of Signal ac/dc Converter

```
19A0 4300 19E8    BRANCH TO CHECK "THRU FLAG"
19A4 C5B0 2710    ALL 9'S (DEC) IN RII?
19A8 4230 19C0    BRANCH ON NOT =
19AC C840 0004    LOAD CODE FOR 10 VOLT RANGE
19B0 C810 000A    LOAD 10 VOLT RANGE FLAG
19B4 4010 09E4    STORE FLAG
19B8 C800 0071    RELOAD RELAY BANK 1 DEVICE ADDRESS
19BC 4300 19D8    BRANCH
19C0 C810 0001    SET FLAG FOR 1 VOLT RANGE
19C4 4010 09E4    STORE RANGE FLAG
19C8 C810 0000    LOAD 0000
19CC 4010 09EA    STORE "NOT THRU"FLAG
19D0 4200 0000    NOP
19D4 4300 0ADC    CONTINUE WITH MAIN PROG.
19D8 C8C0 0001    RESET RATIO MODE CODE FOR DVM
19DC C810 0001    LOAD 0001
19E0 4010 09EA    STORE "THRU" FLAG
```

```
19E4 4300 19F8      RETURN
19E8 4810 09EA      LOAD "THRU"FLAG
19EC C510 0001      HAVE I BEEN HERE BEFORE?
19F0 4230 19A4      BRANCH IF NOT TO CHECK FOR OVERANGE COND.(9999)
19F4 4300 0ADC      HAS BEEN THRU, BY PASS THIS ROUTINE
19F8 41E0 17D8      BAL TO FAST/SLOW ROUTINE FOR CONVERTERS(FIXES UP R2)
19FC 4300 0AB0      RETURN
>
```

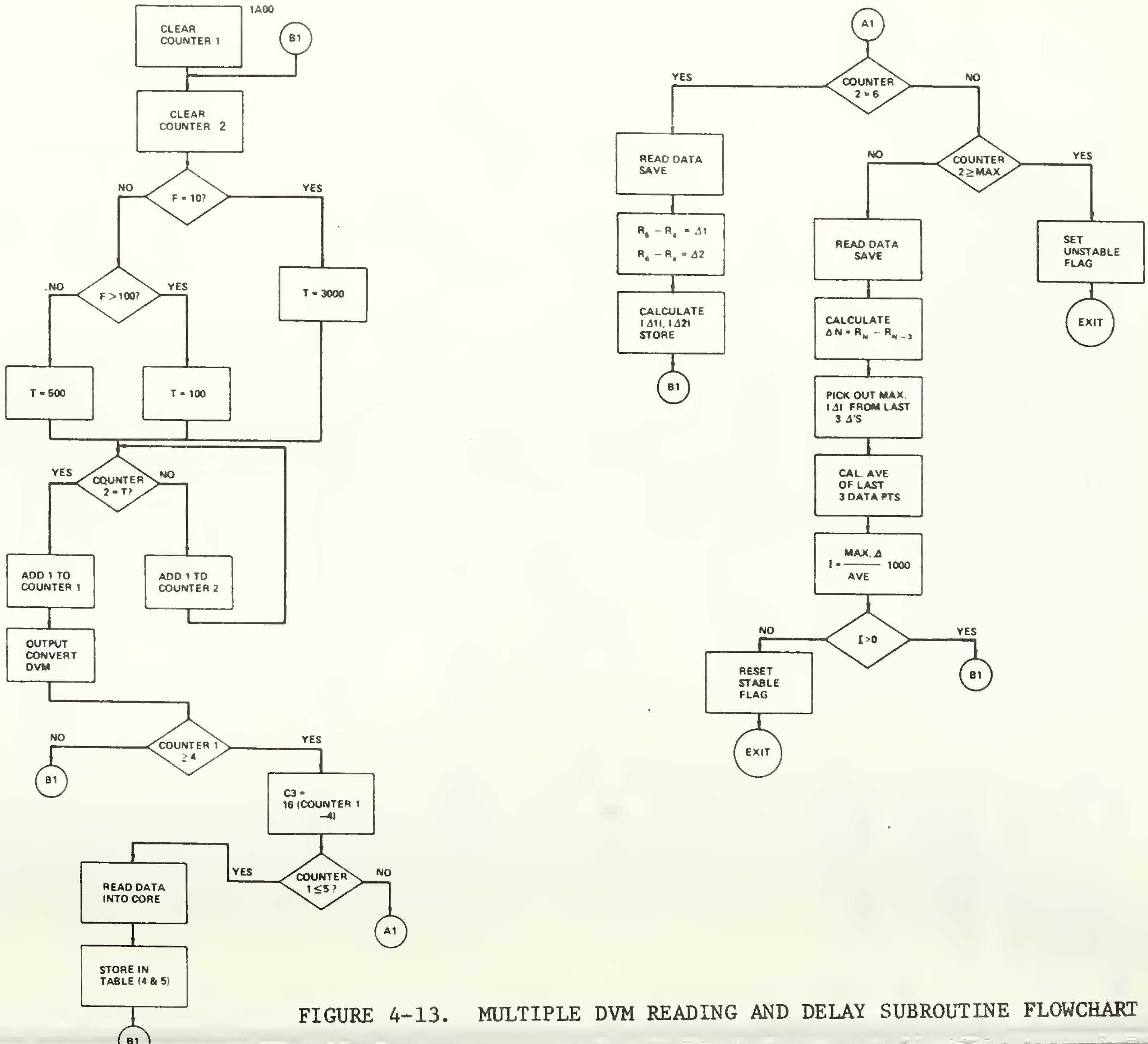


FIGURE 4-13. MULTIPLE DVM READING AND DELAY SUBROUTINE FLOWCHART

4.39 *Multiple Readings (DVM) and Digital Filter Subroutine

Call on R9

This subroutine handles the actual data collection by the DVM. The first part of the subroutine sets up the timing delays for the frequency ranges. The three ranges are:

10 Hz	Typical delay constant =	3000	(1A36)
10<F<100	" " " =	500	(1A2E)
F>100	" " " =	100	(1A3E)

This delay allows for settling time for the exciter and also for the ac/dc converters. The larger the constant, the longer the delay. COUNTER 1 counts up to the delay constant before a ratio is read by the DVM.

After the timing delay is set up, the program executes three output convert instructions. An output convert instruction triggers the DVM to take a reading but does not read the data into core. The reason for executing these initial output convert instructions is to take care of any transient signals that may occur because of the switching circuitry. These readings are not saved and do not enter into the subsequent calculations.

COUNTER 2 keeps track of the number of ratio readings in the subroutine. Starting with reading number four, the data are stored in a table. Beginning with reading number five, deviations are calculated as shown in the accompanying flow chart (see figure 4-13). As can be seen from this flow chart, deviations are calculated from the current ratios as follows:

$$\text{delta } R = R_N - R_{N-3}$$

The reason for going three readings back to calculate the deviations is to catch any slow drift in the system. Since a decision on whether or not to accept a series of readings is based on the percentage deviation of the readings, a slow drift in the readings would not be detected if the deviations were based on a $R_N - R_{N-1}$ deviation calculation. Once seven readings have been taken, the program then has three deviations to look at. These deviations are converted to absolute values, and the maximum of the three is picked out and saved. If this maximum deviation is less than 0.1 percent of the average of the last three readings, this average is saved as the desired ratio and the STABLE/UNSTABLE FLAG is reset (0000). If the deviation is equal or greater than 0.1 percent, another reading is taken. The last three readings are then compared as above. This process is continued until three readings are taken successively which are acceptable or an upper limit is reached of 32 readings. It the upper limit is reached and three acceptable readings were not found, the average of the last three is used and the STABLE/ UNSTABLE FLAG is set (0001). If set, the UNSTABLE FLAG causes a message to be typed after the calibration for this test point: "UNSTABLE SIGNAL".

Important Core Locations:

(2303)	Number of ratios taken to get 0.1 percent maximum delt
(2304)	Ratio (N-2)
(2306)	Ratio (N-1)
(2308)	Ratio (N)
(230A)	Maximum deviation of last three ratios tested
(230C)	Average of last three ratios taken
(230E)	STABLE/UNSTABLE FLAG (0000, stable; 0001, unstable)

Input Requirements:

DVM must be in ratio mode.

*Multiple Reading (DVM) and Digital Filter Subroutine

```
1A00 4200 0000     NOP
1A04 4200 0000     NOP
1A08 0B88          CLEAR R8 (TABLE INDEX)
1A0A 0200          NOP
1A0C C830 00CA     LOAD DVM DEVICE ADDRESS
1A10 0BBB          CLEAR RB (COUNTER INDEX)
1A12 0200          NOP
1A14 4200 0000     NOP
1A18 C560 0004     F=10 HZ.?
1A1C 4330 1A34     IF SO BRANCH
1A20 0876          IF NOT,LOAD FREQ. INDEX REGISTER (R6) INTO R7
1A22 0200          NOP
1A24 CB70 001C     SUBTRACT 1C
```

```
1A28 4220 1A3C      IF F>100 HZ. BRANCH
1A2C C870 0500      10<F<100   SET T=500    T IS A TIMING PARAMETER
1A30 4300 1A40      BRANCH
1A34 C870 3000      F=10 HZ.,    SET T=3000
1A38 4300 1A40      BRANCH
1A3C C870 0100      F>100 HZ.   SET T= 100
1A40 05B7           RB:R7   COMPARE COUNTER TO T
1A42 0200           NOP
1A44 4330 1A50      BRANCH ON =
1A48 CAB0 0001      ADD 1 TO COUNTER
1A4C 4300 1A40      GO TEST FOR UPPER LIMIT
1A50 CA80 0001      ADD 1 TO # OF READINGS INDEX
1A54 DE30 0B25      OUTPUT CONVERT DVM
1A58 0878           LOAD
1A5A 08B8           LOAD
1A5C CB70 0004      SUB. 4 FROM # OF READINGS
1A60 4310 1A68      BRANCH IF #READINGS IS≥4
1A64 4300 1A0C      IF < 4 BRANCH BACK
1A68 C820 0008      LOAD 8
1A6C 0CA2           MULT #READINGS * 8
1A6E 08AB           LOAD INTO RA
1A70 4200 0000      NOP
1A74 CBA0 0020      SUB. 20 FROM ABOVE PRODUCT
1A78 0878           LOAD #READINGS INTO R7
1A7A 0200           NOP
```

```
1A7C CB70 0005     SUBTRACT 5
1A80 4320 1A9C     BRANCH ON NOT + (#EQUAL OR LESS THAN 5)
1A84 C580 0006     # READINGS =6?
1A88 4230 1B0C     IF NOT BRANCH
1A8C 41C0 2000        # READINGS =6   BAL TO READ DATA SUBROUTINE
1A90 4830 26EC     RESTORE RATIO
1A94 4300 1AAC     GO CALCULATE DELTA 1
1A98 4200 0000     NOP
1A9C 41C0 2000     BAL TO READ DATA
1AA0 4830 26EC     RESTORE RATIO
1AA4 403A 0B50     STORE RATIO IN TABLE
1AA8 4300 1A0C     GO TAKE ANOTHER RATIO READING
1AAC 403A 0B50     STORE RATIO 6
1AB0 4840 0B58     LOAD RATIO 5
1AB4 4200 0000     NOP
1AB8 4200 0000     NOP
1ABC 4830 0B50     LOAD RATIO 4
1AC0 4200 0000     NOP
1AC4 4200 0000     NOP
1AC8 0B43          RATIO 5-RATIO 4
1ACA 0834          LOAD
1ACC 4200 0000     NOP
1AD0 4030 0B5A     STORE DELTA 1 IN TABLE
1AD4 4200 0000    NOP
1AD8 4840 0B60     LOAD RATIO 6
```

```
1ADC 4200 0000    NOP
1AE0 4200 0000    NOP
1AE4 4200 0000    NOP
1AE8 4830 0B50    LOAD RATIO 4
1AEC 4200 0000    NOP
1AF0 4200 0000    NOP
1AF4 0B 43        RATIO 6- RATIO 4
1AF6 0200         NOP
1AF8 4200 0000    NOP
1AFC 4040 0B62    STORE DELTA 2
1B00 4200 0000    NOP
1B04 4300 2310    GO TO TAKE ABSOLUTE VALUE OF DELTAS
1B08 4300 1A0C    CONTINUE READINGS
1B0C 0878         LOAD # READINGS
1B0E 0200         NOP
1B10 CB70 0023    SUBTRACT OFF UPPER LIMIT OF # READINGS ALLOWED
1B14 4220 22F0    IF MAXIMUM # HAS BEEN REACHED,BRANCH
1B18 41C0 2000    IF NOT, BAL TO READ DATA
1B1C 4830 26EC    RESTORE RATIO N
1B20 4080 2302    SAVE # TRIES

1B24 403A 0B50    STORE RATIO N
1B28 4200 0000    NOP
1B2C 4200 0000    NOP
1B30 4200 0000    NOP
```

```
1B34 0843             LOAD RATIO N
1B36 0200             NOP
1B38 4200 0000
1B3C 4200 0000
1B40 CBA0 0018        LOAD RATIO(N-3)
1B44 481A 0B50          "
1B48 4200 0000        NOP
1B4C 4200 0000
1B50 0B41             RATIO N- RATIO(N-3)= DELTA N
1B52 0834             LOAD
1B54 4310 1B60        BRANCH ON NOT MINUS
1B58 C850 FFFF        LOAD -1
1B5C 0C44             MULT DELTA N BY -1
1B5E 0845             LOAD
1B60 CAA0 0018        ADD 18 TO RESTORE INDEX
1B64 404A 0B56        STORE ABSOLUTE VALUE OF DELTA N
1B68 4300 2200        GO TO MORE SPACE
1B6C 0872             STORAGE FOR R8
```

4.40 *Read Data Subroutine

Call on RC

This subroutine programs the DVM to read one time. The decimal reading is stored in R9 and core locations 0554 and 0556. If the first digit is zero, the LOW/NORM FLAG is set at 0000 and if the first digit is not zero, the flag is reset at 0001. This subroutine also calls the Round Off Subroutine (049C)(see Section 4.29) which gives, as output, a hexidecimal number equivalent to the decimal entry rounded to four digits. This will be stored in 26EC.

Input Requirements: DVM must be set for Ratio or Voltage mode as desired.

Output Requirements: Decimal data will be in core locations 0554, 0556 and R9. Hexidecimal data equivalent to rounded decimal data will be in core location 26EC.

*Read Data Subroutine
(Data will be in R9 and 0554, 0556)

```
2000 C830 00CA    LOAD DVM DEVICE ADDRESS
2004 DE30 0B24    ENABLE INTERUPTS FOR DVM
2008 41E0 06D8    SAVE REGISTERS
200C 9F95         ACKNOWLEDGE INTERUPTS
200E 0539         IS IT DVM?
2010 4330 2018    IF SO,BRANCH TO READ DATA
2014 4300 2008    IF NOT,LOOK AGAIN FOR INTERUPT
2018 0B77         CLEAR R7
201A 0200         NOP
201C C890 0005    LOAD LIMIT INTO R9
2020 C880 0001    LOAD INCREMENT INTO R8
2024 DB37 02B8    READ DATA INTO CORE
2028 4120 1010    DELAY
202C C170 2024    BXLE
```

```
2030 4200 0000     NOP
2034 4200 0000
2038 4190 0500     BAL TO ASCII/DEC. ROUTINE
203C 4870 0554     LOAD FIRST DEC. DIGIT
2040 4230 205C     BRANCH ON NOT 0
2044 C840 0000     LOAD 0
2048 4040 09F6     SET LOW/NORM FLAG=1
204C 4890 0556     LOAD 2'ND HALFWORD OF DEC. #
2050 4800 0554     LOAD 1'ST HALFWORD
2054 0819          LOAD
2056 0200          NOP
2058 4300 26C8     BRANCH
205C C840 0001     LOAD 1
2060 4040 09F6     SET LOW/NORM FLAG =1
2064 4800 0554     LOAD FIRST HALFWORD
2068 4810 0556     LOAD SECOND HALFWORD
206C 4200 0000     NOP
2070 4200 0000     NOP
2074 4200 0000     NOP
2078 4300 26D0     BRANCH TO NEW SPACE
>
```

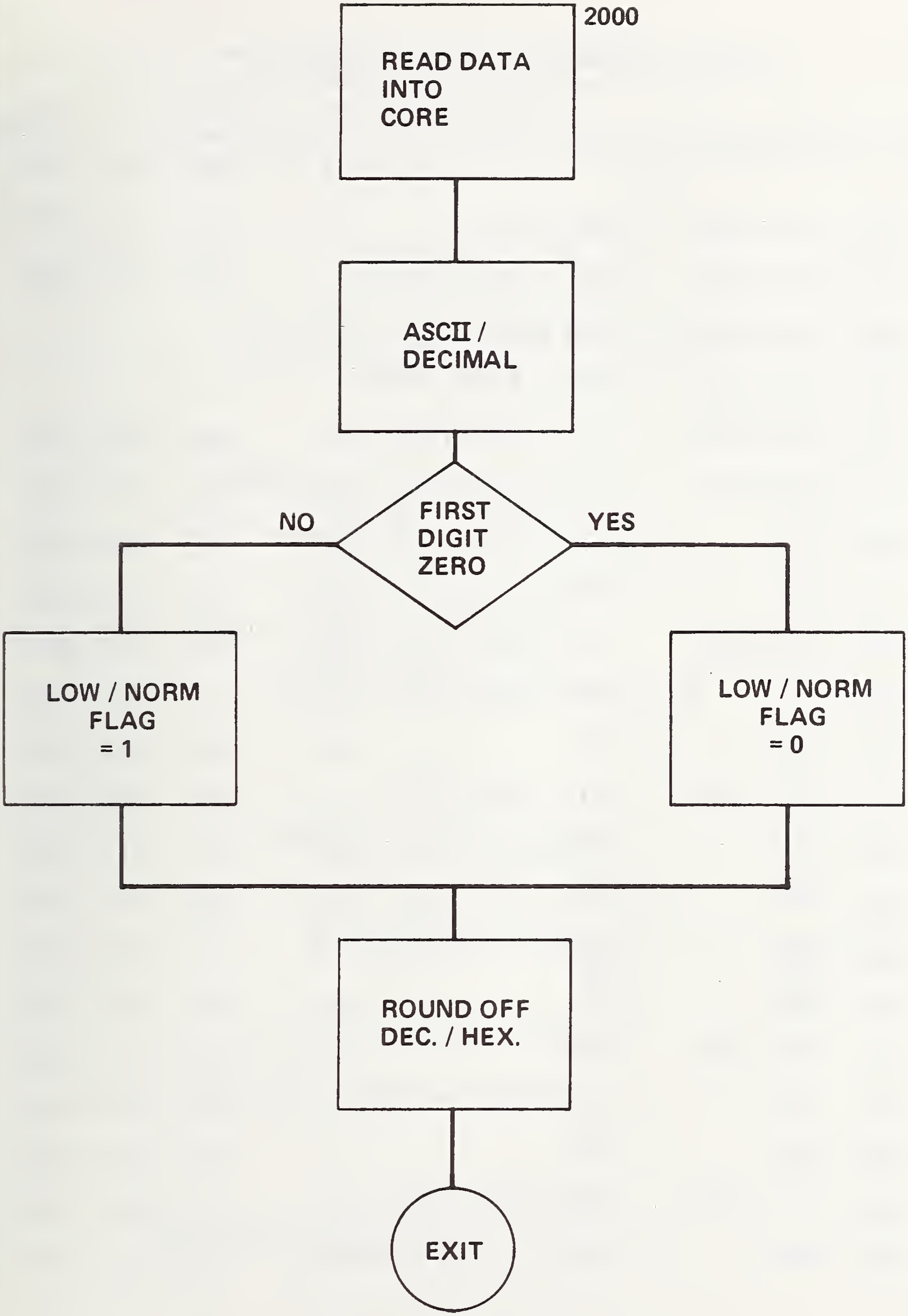


FIGURE 4-14. READ DATA SUBROUTINE FLOWCHART

Multiple Readings and Digital Filter Subroutine
Calculate Max Delta of Last Three Deltas

Note: The following is part of Section 4.39, Multiple Readings Subroutine.

```
2200 483A 0B56      LOAD DELTA N
2204 CBA0 0008      SUB. 8 FROM INDEX
2208 482A 0B56      LOAD DELTA(N-1)
220C CBA0 0008      SUB. 8 FROM INDEX
2210 481A 0B56      LOAD DELTA(N-2)
2214 CAA0 0010      ADD 10 TO INDEX TO RESTORE
2218 0512           COMPARE DELTA(N-2):DELTA (N-1)
221A 0200           NOP
221C 4280 2228      BRANCH ON LOW   DELTA(N-2)<DELTA(N-1)
2220 0841           LOAD DELTA (N-2) AS MAX 1
2222 0200           NOP
2224 4300 222C      SKIP NEXT INSTR.
2228 0842           LOAD DELTA(N-1) AS MAX 1
222A 0200           NOP
222C 0543           COMPARE MAX1:DELTA N
222E 0200           NOP
2230 4280 223C      BRANCH IF MAX 1 < DELTA N
2234 0854           SAVE MAX 1 AS MAX DELTA IN R5
2236 0200           NOP
2238 4300 2240      SKIP NEXT INST.
223C 0853           DELTA N IS MAX DELTA,SAVE IN R5
```

```
223E 0200            NOP
2240 4050 230A       STORE MAX DELTA
2244 4200 0000       NOP
2248 4200 0000
224C 4200 0000
```

Calculate Average of Last Three Ratios

```
2250 483A 0B50       LOAD RATIO N
2254 CBA0 0008       SUB. 8 FROM INDEX
2258 482A 0B50       LOAD RATIO (N-1)
225C CBA0 0008       SUB. 8 FROM INDEX
2260 481A 0B50       LOAD RATIO (N-2)
2264 CAA0 0010       ADD 10 TO INDEX TO RESTORE
2268 4200 0000       NOP
226C 4200 0000
2270 4010 2304       SAVE RATIO  (N-2)
2274 4020 2306       SAVE RATIO  (N-1)
2278 4030 2308       SAVE RATIO N    N WILL BE LAST RATIO READ
227C 4200 0000       NOP
2280 4200 0000
2284 4200 0000
2288 4200 0000
228C 4A30 2306       N+(N-1)
2290 4A30 2304       N+(N-1)+(N-2)
```

```
2294 4200 0000
2298 4200 0000
229C C840 0003     LOAD
22A0 0B22          CLEAR R2
22A2 0200
22A4 0D24          (N+(N-1)+(N-2))/3  QUOT. IN R3 (AVERAGE OF 3 RATIOS
22A6 0200
22A8 4030 230C     SAVE AVERAGE
22AC C830 03E8     LOAD CONST.
22B0 4C20 230A     1000*DELTA(MAX)
22B4 4D20 230C     1000*DELTA(MAX)/AVER. RATIO
22B8 0833          LOAD TO SET CONDITION CODE
22BA 0200
22BC 4220 1A0C     BRANCH ON + (QUOT. > 0)  TO CONTINUE READINGS
22C0 C810 0000     LOAD 0
22C4 4010 230E     STORE "STABLE" FLAG    MAX DELTA < .1% OF RATIO AV
22C8 0309          RETURN TO CALL
22CA 0000
22F0 C810 0001     LOAD 1
22F4 4010 230E     STORE "UNSTABLE FLAG"
22F8 4200 0000
22FC 4200 0000
2300 0309          RETURN TO CALL
2302 0008          STORAGE # OF RATIOS READ TO GET .1% MAX DELTA
```

```
2304 1B8D 1B8D      "    RATIO (N-2)      RATIO (N-1)
2308 1B8E 0001      "    RATIO N          MAX DELTA
230C 1B8D 0000      "    AVERAGE RATIO     STABLE/UNSTABLE FLAG
```

Calculate Absolute Value of Deltas

```
2310 4840 0B62    LOAD DELTA 1
2314 4200 0000
2318 4200 0000
231C 4200 0000
2320 4310 232C    BRANCH ON NOT -
2324 C850 FFFF    LOAD -1
2328 0C44         DELTA 1 *(-1)
232A 0845         LOAD QUOT
232C 4040 0B66    STORE /DELTA 1/ IN TABLE
2330 4200 0000
2334 4840 0B72    LOAD DELTA 2
2338 4200 0000
233C 4200 0000
2340 4200 0000
2344 4310 2350    BRANCH ON NOT +
2348 C850 FFFF    LOAD -1
234C 0C44         DELTA 2 * (-1)
234E 0845         LOAD QUOT.
2350 4040 0B 76   STORE /DELTA 2/ IN TABLE
2354 4200 0000
2358 4300 1B08    RETURN TO MAIN PROG.
>
```

4.41 *ASCII/Decimal Subroutine

Call on RE

This subroutine converts five bytes of ASCII code to decimal code.

Input Requirements: ASCII code in locations 09B6-09BA

Output Requirements: Decimal data in locations 246C-246F

This subroutine saves all registers except RE (see figure 4-15.)

*ASCII/Decimal Subroutine

```
23FC 40F0 2470      SAVE RE
2400 41E0 06E0      BAL TO SAVE REGISTERS
2404 D3A0 09B7      LOAD ASCII #'S INTO REGISTERS
2408 D3B0 09B8
240C D3C0 09B9
2410 D3D0 09BA
2414 D3E0 09B6
2418 0B11           CLEAR R1
241A 0B77           CLEAR R7
241C 0B00           CLEAR R0
241E 0200
2420 930A           LOAD FIRST ASCII #
2422 0200
2424 0810           ALSO INTO R1
2426 0200
2428 C410 000F      PICK OFF BITS 12-15
242C CD10 000C      SHIFT LEFT 12
```

```
2430 930B           LOAD SEC. #
2432 0200
2434 0820           ALSO IN R2
2436 0200
2438 C420 000F      PICK OFF BITS 12-15
243C CD20 0008      SHIFT LEFT 8
2440 0A12           ADD R2 TO R1
2442 0200
2444 933C           LOAD 3'RD #
2446 0200

2448 C430 000F      PICK OFF BITS 12-15
244C CD30 0004      SHIFT LEFT 4 BITS
2450 0A13           ADD R3 TO R1
2452 0200
2454 933D           LOAD 4'TH #
2456 0200
2458 C430 000F      PICK OFF BITS 12-15
245C 0A13           ADD R3 TO R1 (R1 NOW CONTAINS FIRST 4 DIGITS)
245E 0200
2460 933E           LOAD 5 TH DIGIT
2462 0200
2464 C430 000F      PICK OFF BITS 12-15
2468 4300 2474      BRANCH
```

```
246C 0000           STORAGE
246E 3000 1808         "
2472 0000
2474 4200 0000
2478 4200 0000
247C 4030 246C      STORE FIRST DIGIT
2480 4010 246E      STORE  LAST 4 DIGITS
2484 41E0 0720      RESTORE REGISTERS
2488 48E0 2470      RESTORE RE
248C 030E           RETURN
>
```

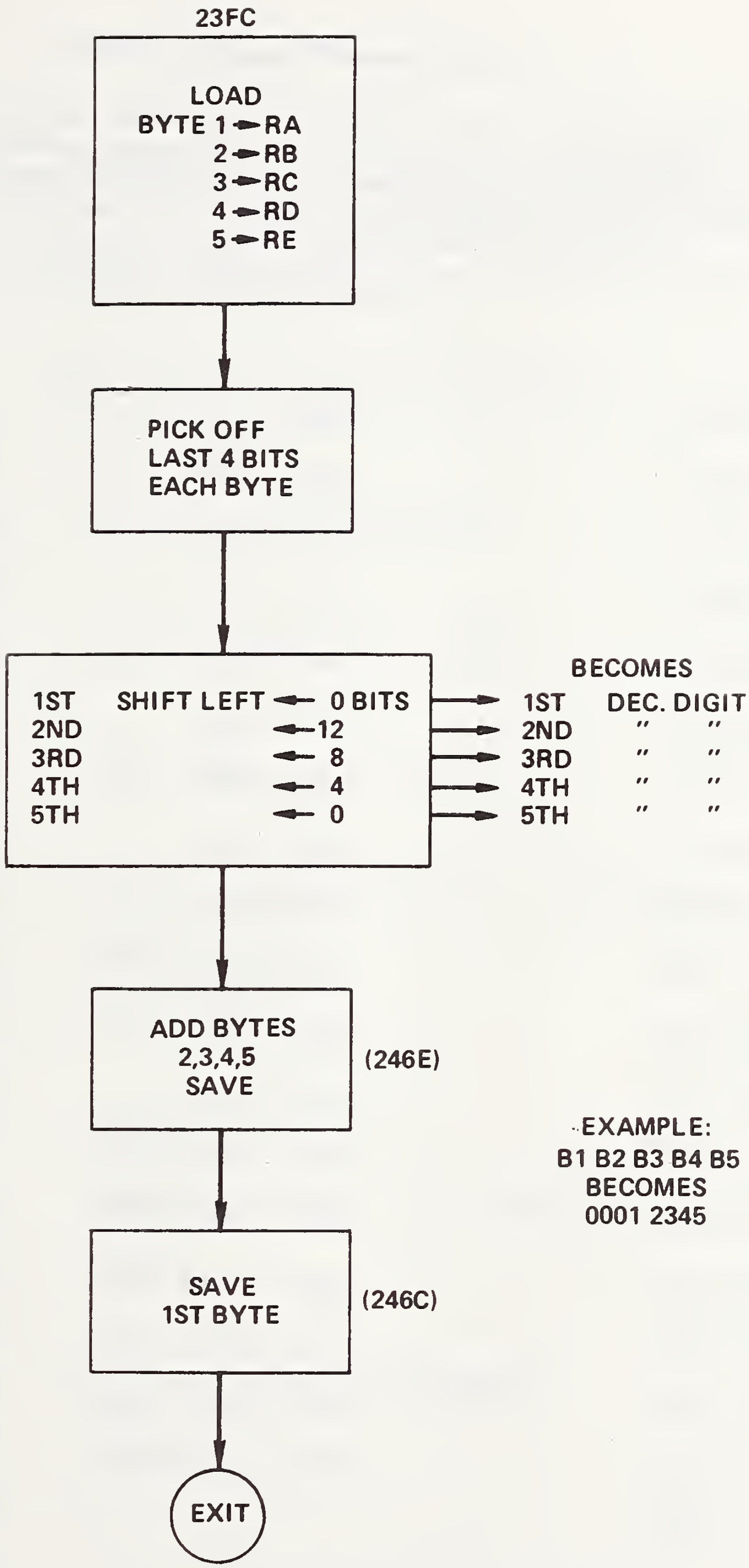


FIGURE 4-15. ASCII-TO-DECIMAL SUBROUTINE FLOWCHART

4.42 Desired Frequency Table

This table gives a set of desired frequency test points which is used to check against the reading of the frequency counter. These desired frequencies are compared to the counter reading in the Check for Proper Frequency program (see Section 4.43).

Desired Frequency Table

24B0 0000	24D6 0400	2506 2500 0000
24B2 0010	24D8 0000	250A 3000 0000
24B4 0000	24DA 0500	250E 3500 0000
24B6 0015	24DC 0000	2512 4000 0000
24B8 0000	24DE 0600	2516 4500 0000
24BA 0030	24E0 0000	251A 5000 0000
24BC 0000	24E2 0700	251E 5500 0000
24BE 0050	24E4 0000	2522 6000 0000
24C0 0000	24E6 0800	2526 6500 0000
24C2 0100	24E8 0000	252A 7000 0000
24C4 0000	24EA 0900	252E 7500 0000
24C6 0100	24EC 0000	2532 8000 0000
24C8 0000	24EE 1000 0000	2536 8500 0000
24CA 0100	24F2 1500 0000	253A 9000
24CC 0000	24F6 1700 0000	253C 0000
24CE 0200	24FA 2000 0000	253E 9500
24D0 0000	24FE 2000 0000	2540 0001
24D2 0300	2502 2000 0000	2542 0000
24D4 0000		>

4.43 *Check for Proper Frequency Subroutine

Call on R8

This subroutine compares the frequency counter reading with a desired frequency and if the two do not agree, reprograms the oscillator again and reads the frequency again. This is repeated up to three times. After three tries, the program continues with whatever frequency exists at that time.

*Check for Proper Frequency Subroutine

```
2560 41E0 06E0     BAL TO SAVE REGISTERS
2564 4200 0000
2568 41E0 23FC     BAL TO DECODE FREQ  DATA
256C 4200 0000
2570 4880 246C     LOAD FREQ. DATA DIGIT 1
2574 4890 246E       "                     2-5
2578 4586 24AC     COMPARE FIRST DIGIT WITH TABLE VALUE
257C 4330 2588     BRANCH ON =
2580 4300 25A0     GO TO TRY AGAIN
2584 4200 0000
2588 4596 24AE     COMPARE DIGITS 2-5
258C 4230 25A0     BRANCH ON NOT= TO TRY AGAIN
2590 4200 0000
2594 4200 0000
2598 41E0 0720     BAL TO RESTORE REGISTERS
259C 4300 097A     CONTINUE
25A0 4200 0000
```

```
25A4 4870 09E2      LOAD COUNTER IN R7
25A8 C570 0003      IS IT 3?
25AC 4330 2598      BRANCH ON = TO RESTORE AND CONT.
25B0 CA70 0001      ADD 1 TO R7
25B4 4070 09E2      STORE COUNTER
25B8 4200 0000
25BC 41E0 0720      BAL TO RESTORE REGISTERS
25C0 C870 000F      LOAD OSC. DEVICE ADDRESS
25C4 C8A0 0000      CLEAR RA
25C8 CB60 0004      SUB. 4 FROM R6
25CC 4300 0908      RETURN
>
```

*Read Data Subroutine (Continued from Section 4.40)

Note: The following is part of Section 4.40 Read Data subroutine.

```
26BC 4200 0000     NOP
26C0 4200 0000
26C4 4200 0000
26C8 4200 0000
26CC 4200 0000
26D0 4200 0000
26D4 4200 0000
26D8 4200 0000
26DC 41E0 049C     BAL TO ROUND OFF,DEC/HEX ROUTINE
26E0 4200 0000     NOP
26E4 41E0 0720     RESTORE REGISTERS
26E8 030C          RETURN TO CALL
26EA 0000          HEX. STORAGE
26EC 1B8E 6356     STORAGE
26F0 B564 3269
26F4 3232 3559
26F8 6935 6435
26FC 3332 6944
```

4.44 *Type-Out Subroutine

Call on RE

This subroutine permits type-out of a message or data by a subroutine call with three transfer parameters. The calling sequence is as follows:

```
xxxx  41E0 3478
      Parameter 1  Parameter 2
      Parameter 3

      Parameter 1:  starting address of type message or data.
      Parameter 2:  ending address of type message or data.
      Parameter 3:  either 0000 for binary data or 0001 for ASCII data.
```

If the data to be typed are in ASCII code, the type-out will be as stored in the core without any additional spaces, carriage returns or line feeds. These must be supplied in the table of constants in core to be typed. Messages to be typed may be entered by using the Message Entry Subroutine(3600),Sect. 4.45

If the data to be typed are in binary, spaces will be placed after every two bytes typed out. A carriage return, line feed will be output after the maximum bytes per line. This constant (maximum bytes per line) is entered in location 340A.

*Type-Out Subroutine

```
3478 4200 0000    NOP
347C 40E0 3554    SAVE RETURN ADDRESS
3480 488E 0000    LOAD ADDRESS OF START
3484 0B55         CLEAR
3486 0200
3488 4050 3558    STORE COUNTER X
348C 4050 355A    STORE COUNTER 1
3490 4050 355C    STORE COUNTER 2
3494 4080 355E    SAVE STARTING ADDRESS
3498 CAE0 0002    ADD 2 TO RETURN ADDRFSS
349C 488E 0000    LOAD END ADDRESS
```

```
34A0 4300 3598     GO CHECK FOR ASCII CODE
34A4 4870 355E     (COMES HERE IF NOT ASCII)  LOAD START ADDRESS
34A8 4A70 3558     ADD X
34AC D307 0000     LOAD BYTE TO BE TYPED
34B0 0830          LOAD
34B2 C400 00F0     PICK OFF 1 ST. CHARACTER
34B6 CC00 0004     SHIFT RIGHT
34BA C430 000F     PICK OFF 2 ND. CHAR.
34BE 4190 1954     BAL. DEC/ASCII
34C2 41E0 3564     TYPE FIRST CHAR.
34C6 0803          LOAD 2 ND. CHAR.
34C8 4190 1954     DEC/ASCII
34CC 41E0 3564     TYPE 2 ND. CHAR.
34D0 4850 355A     LOAD COUNTER 1
34D4 CA50 0001     ADD 1
34D8 4050 355A     STORE COUNTER 1
34DC C550 0002     COUNTER 2 = 2?
34E0 4230 34F4     BRANCH ON NOT =
34E4 D320 0E5C     LOAD SPACE CHAR.
34E8 41E0 3564     TYPE IT
34EC 0B 55         CLEAR COUNTER 1
34EE 0200
34F0 4050 355A     STORE
34F4 48A0 3558     LOAD COUNTER X
34F8 CAA0 0001     ADD 1
```

```
34FC 4850 355C     LOAD COUNTER 2
3500 CA50 0001     ADD 1
3504 4050 355C     STORE COUNTER 2
3508 C550 0008     COUNTER 2 = MAX. BYTES PER LINE?
350C 4230 3548     IF NOT BRANCH
3510 D320 09B4     LOAD CARRIAGE RETURN CODE
3514 41E0 3564     TYPE OUT
3518 D320 09B5     LOAD LINE FEED CODE
351C 41E0 3564     TYPE OUT
3520 4200 0000
3524 4820 3560     LOAD END ADDRESS
3528 4840 355E     LOAD START ADDRESS
352C 0B24          SUBTRACT
352E 0200
3530 CA20 0001     ADD 1 TO ABOVE DIFFERENCE
3534 0B2A          SUBTRACT OFF # TIMES THRU
3536 0200
3538 4220 3540     BRANCH ON + ( NOT FINISHED)
353C 4300 357C     BRANCH
3540 0200
3542 0B77          RETURN
3544 4070 355C     CLEAR COUNTER 2
3548 40A0 3558     SAVE X
354C 4200 0000
3550 4300 34A4     CONTINUE
```

```
3554 05C0           STORAGE FOR RE
3556 0000
3558 0018              "        COUNTER X
355A 0000              "        COUNTER 1
355C 0000              "        COUNTER 2
355E 05D0              "        START
3560 05E7              "        END
3562 0000
3564 C810 0002      LOAD TTY DEVICE ADDRESS
3568 DE10 09B2      PUT TTY IN WRITE MODE
356C 4180 0EEC      WAIT
3570 9A12           TYPE
3572 0200
3574 030E           RETURN
3576 0200
3578 0000
357A 0000
357C 48E0 3554      RESTORE RE
3580 CAE0 0006      ADD 6
3584 030E           RETURN
3586 0200
3588 0000
358A 0000
```

```
3598 4080 3560     SAVE END ADDRESS
359C 48E0 3554     RESTORE RE
35A0 CAE0 0004     ADD 4
35A4 480E 0000     LOAD CODE FOR ASCII OR BINARY IDENTIFICATION
35A8 C500 0000     IS IT 0 ?
35AC 4330 34A4     BRANCH ON = TO CONTINUE
35B0 4870 355E     LOAD START ADDRESS
35B4 4A70 3558     ADD COUNTER X
35B8 D327 0000     LOAD BYTE TO BY TYPED
35BC 08B2          SAVE IT IN RB
35BE 0200
35C0 48A0 3558     LOAD COUNTER X
35C4 4200 0000
35C8 4200 0000
35CC 4200 0000
35D0 4820 3560     LOAD END ADDRESS
35D4 4840 355E     LOAD START ADDRESS
35D8 0B24          SUBTRACT
35DA 0200
35DC CA20 0001     ADD 1
35E0 0200
35E2 0B2A          (END -START)- # TIMES THRU
35E4 4220 35EC     BRANCH ON + (NOT FINISHED)
```

```
35E8 4300 357C     GO TO EXIT
35EC 082B          LOAD
35EE 0200
35F0 41E0 3564     TYPE BYTE
35F4 CAA0 0001     ADD 1 TO COUNTER X
35F8 40A0 3558     SAVE COUNTER X
35FC 4300 35B0     EXIT
>
```

4.45 *Message Entry Subroutine

Call on RE

This subroutine permits entry of messages to be typed out by the Type-Out Subroutine (see Section 4.44). This subroutine has two transfer parameters. The calling sequence is as follows:

```
xxxx  41E0 3600
      Parameter 1  Parameter 2

      Parameter 1:  number of bytes allowed in the message.
      Parameter 2:  starting address for stored message (core location)
```

Upon calling this subroutine, a message can be typed and transferred directly into core (ASCII code). Upon reaching the maximum bytes allowed, a message is typed on the TTY: NO MORE MESSAGE SPACE. If end-of-message is reached before all the message space is used, control A will cause the program to terminate and exit to the monitor. If a mistake is made in typing a character or characters the backward arrow < will backspace one character (one ASCII byte) for each < typed. The correct character or characters then may be typed.

*Message Entry Subroutine

```
3600 4300 36E0     GO TO MORE SPACE
3604 4200 0000
3608 4200 0000
360C 0B00          CLEAR R0
360E 0200
3610 4000 36C8     SAVE COUNTER X
3614 C830 0002     PUT TTY IN READ MODE
3618 DE30 0A04      "
361C 9D34          SENSE STATUS, LOAD COND. CODE
361E 0844
3620 4230 361C     BRANCH ON BUSY TO SENSE STATUS
```

```
3624 9B39           READ DATA

3626 0200

3628 4300 3708      GO CHECK FOR < CHARACTER (A < IS TYPED TO ERACE
                                              A CHAR.)
362C C590 0081      IS THE CHAR. A CONTROL A (CHAR. FOR END OF MESSAGE)?

3630 4330 36D0      IF SO, GO TO EXIT

3634 4B10 36CA      HAS MAXIMUM # OF BYTES BEEN TYPED?

3638 4310 3678      IF SO, TYPE OUT "NO MORE MESSAGE SPACE"

363C C590 0021      IS THE CHAR. "!"?

3640 4330 3664      BRANCH ON =

3644 4810 36C8      LOAD COUNTER

3648 D291 3720      STORE BYTE IN CORE

364C CA10 0001      ADD 1 TO X

3650 4010 36C8      SAVE

3654 4200 0000

3658 4200 0000

365C 4300 361C      CONTINUE

3660 4200 0000

3664 41E0 3478      COMES HERE IF CHAR IS "1"  GO TO TYPE OUT

3668 368E 369F

366C 0001

366E 0200

3670 4300 36D0      GO TO EXIT

3674 4200 0000

3678 41E0 3478      TYPE OUT "NO MORE MESSAGE SPACE"

367C 36A8 36C1
```

```
3680 0001

3682 0200

3684 4300 36D0      GO TO EXIT

3688 0000

368A 0000

368C 0000

368F 8D8A C5CE      STORAGE FOR TTY CODE     CR,LF   END OF MESSAGE

3692 C4A0 CFC6

3696 A0CD C5D3

369A D3C1 C7C5

369E 8D8A 0000

36A2 0000

36A4 0000

36A6 0000

36A8 8D8A A0CE           "                   CR,LF   NO MORE MESSAGE
                                                     SPACE
36AC CFA0 CDCF

36B0 D2C5 A0CD

36B4 C5D3 D3C1

36B8 C7C5 A0D3

36BC D0C1 C3C5

36C0 8D8A 0000      STORAGE FOR TTY CONSTANTS

36C4 0000

36C6 0000
```

```
36C8 0003              STORAGE FOR COUNTER
36CA 0003                 "      MAXIMUM # BYTES ALLOWED IN MESSAGE
36CC 3814 0000            "      RF
36D0 48F0 36CC         RESTORE RF
36D4 CAE0 0004         ADD 4
36D8 030F              RETURN
36DA 0000
36DC 0000
36DE 0000
36E0 40F0 36CC         SAVE RF
36E4 488F 0000         LOAD ((RE)) INTO R6
36E8 4080 36CA         SAVE MAXIMUM # OF BYTES
36EC CAE0 0002         ADD 2 TO RE
36F0 488E 0000         LOAD ((RE)) INTO R8
36F4 4080 364A         SAVE STARTING LOCATION
36F8 4300 3604         GO BACK AND CONTINUE
36FC 73EE 73EC
3700 73EC 73EC
3704 73EC 73EC
3708 4810 36C8         LOAD COUNTER
370C C590 003C         IS IT "<"?
3710 4230 362C         BRANCH ON NOT =
3714 CB10 0001         SUBTRACT 1 FROM COUNTER
3718 4010 36C8         STORE
371C 4300 361C         CONTINUE
```

```
3720 FFD7 D7D4       STORAGE FOR REPLY

3724 C941 CC00

3728 0000

372A 73EA 73EC

372F 73EC 73EC

3732 73EC 41E0       SAMPLE PROG FOR USING MESSAGE ENTRY

3736 3600 0020       BAL TO MESSAGE ENTRY

373A 05D0            FIRST PARAMETER* # OF BYTES ALLOWED
                     SECOND PARAMETER*  STORAGE PLACE FOR ANSWER
373C 4300 1C80       GO TO MONITOR

>
```

4.46 Accelerometer Data Block Entry Program

This program allows the user of the system to enter data about the accelerometer under test such as manufacturer, model number, serial number, amplifier used, etcetera. The program is started at location 3740 (see figure 4-16). The program then asks the operator a series of questions about the accelerometer and associated equipment. As the answers are typed on the TTY by the operator, the ASCII code is stored in proper locations in core (see Section 4.35).

There are two options at the beginning of the program, the "PARTIAL" and the "ALL" modes. If the operator types "ALL" in response to the question "ALL OR PARTIAL", the program then goes through and asks a series of sixteen questions. At the end of the sixteen questions, the program returns to Monitor. For the "PARTIAL" option, the operator must request which parameter he wishes to enter by typing one of the following two letter codes.

FN	for	file number	AS	for	amplifier serial number
TN		test number	CU		customer
CA		capacitance	DC		data of calibration
PM		accelerometer manufacturer	E1		exciter 1
PN		accelerometer model number	E2		exciter 2
PS		accelerometer serial number	S1		standard 1
AM		amplifier manufacturer	S2		standard 2
AN		amplifier model number	GA		gain

The operator must enter one of the codes above. The program then asks the question corresponding to the code which was entered. In the "PARTIAL" mode, the program always types "ANOTHER CHANGE?" after each entry. The proper answer is "YES" or "NO".

The data entered by this program together with the data of the calibration factors form a data package located in 2900-3200 (see Section 4.35). This data package can then be entered into a file on magnetic tape by a subroutine call.

Accelerometer Data Block Entry Program

```
3740 4200 0000
3744 4200 0000
3748 41E0 3478     TYPE OUT "ALL OR PARTIAL CHANGES?"
374C 3920 3939
3750 0001
3752 0200
3754 4200 0000
3758 41E0 3600     BAL TO ENTRY PROGRAM FOR ANSWER
375C 0007          TRANSFER PARAMETERS
375E 3720 D310        "         "            LOAD ENTRY DATA
3762 3720 C410     PICK OFF LAST 4 BITS
3766 000F
3768 0811          LOAD R1 TO SET CONDITION CODE
376A 0200
376C 4330 3780     BRANCH IF ANSWER WAS "PARTIAL"
3770 4200 0000
3774 C800 0001     LOAD 1
3778 4000 378C     STORE "ALL" FLAG
377C 4300 0400     CLEAR STORAGE TABLES
3780 C800 0000     LOAD 0
3784 4000 378C     STORE "PARTIAL" FLAG
3788 4300 3844     BRANCH TO CONTINUE
378C 0000          STORAGE FOR ALL/PARTIAL FLAG
378E 0200
```

```
3790 41C0 3888     OUTPUT CR,LF
3794 0B33          CLEAR R3
3796 0200
3798 4030 389C     CLEAR COUNTER X, SAVE
379C 4823 38A0     LOAD START LOCATION
37A0 4843 38A2     LOAD # BYTES ALLOWED IN ANSWER
37A4 0A42          ADD TO GET ENDING LOCATION
37A6 4020 37B2     STORE START LOCATION
37AA 4040 37B4     STORE END LOCATION
37AE 41E0 3478     TYPE QUESTION
37B2 0000
37B4 0000
37B6 0001
37B8 4300 0430     CLEAR TABLES
37BC 4823 38A4     LOAD STORAGE LOCATION
37C0 4843 38A6     LOAD # BYTES ALLOWED
37C4 4020 37D2     STORE
37C8 4040 37D0     STORE
37CC 41E0 3600     GO TO MESSAGE ENTRY SUBROUTINE
37D0 000C
37D2 2B78 4810     LOAD ALL/PARTIAL FLAG
37D6 378C 4330     BRANCH ON PARTIAL
37DA 3800 4840     LOAD INDEX X
37DE 389C 4830       "
37E2 389C 4200
```

```
37E6 0000
37E8 CB40 0078     HAS LAST QUESTION BEEN ASKED?
37EC 4310 1C80     IF SO GO TO MONITOR
37F0 CA30 0008     IF NOT, ADD 8 TO INDEX X
37F4 4030 389C     STORE
37F8 41C0 3888     OUTPUT CR,LF
37FC 4300 379C     CONTINUE

3800 4200 0000
3804 41E0 3478     TYPE "ANOTHER CHANGE?" (ANSWER MUST BE YES OR NO
3808 39DC 39EF
380C 0001
380F 0200
3810 41E0 3600     BAL TO MESSAGE ENTRY FOR REPLY
3814 0003
3816 3720 4830     LOAD REPLY
381A 3720 C530     WAS IT YES?
381E 59C5 4330     IF SO, BRANCH
3822 3844 4200
3826 0000
3828 C530 4ECF     WAS IT NO?
382C 4330 1C80     IF SO, GO TO MONITOR
3830 41F0 3478     IF NEITHER YES OR NO TYPE "WHAT?"
3834 03C0
```

```
3836 03C5
3838 0001
383A 0200
383C 4300 3810      TRY AGAIN
3840 4200 0000
3844 41E0 3478      TYPE "WHICH?"
3848 03C6
384A 03CF
384C 0001
384E 0200
3850 41F0 3600      BAL TO MESSAGE ENTRY FOR REPLY
3854 0003
3856 3720 4830      LOAD REPLY
385A 3720 41C0      TYPE CR LF
385E 3888 0B55      CLEAR R5
3862 0200
3864 4535 03A0      COMPARE (R3): (03A0+(R5))  WHICH PARAMETER ?
3868 4230 3878      BRANCH IF THIS IS NOT IT
386C C830 0004      LOAD 4
3870 0C25           MULT COUNTER R5 BY 4
3872 0200
3874 4300 379C      GO TO TYPE OUT THE QUESTION
3878 CA50 0002      ADD 2 TO INDEX
387C C550 0080      COMPARE COUNTER R5 TO UPPER LIMIT
```

```
3880 4330 3844     IF IMPROPER PARAMETER WAS TYPED, TYPE "WHICH?"
3884 4300 3864     CONTINUE
3888 41E0 3478     OUTPUT CR,LF SUBROUTINE
388C 03CE
388E 03CF
3890 0001
3892 030C
3894 0200
3896 73EC 73EC
389A 73EA 0079     XXXX, STORAGE FOR COUNTER
389E 73EC 393A
>
```

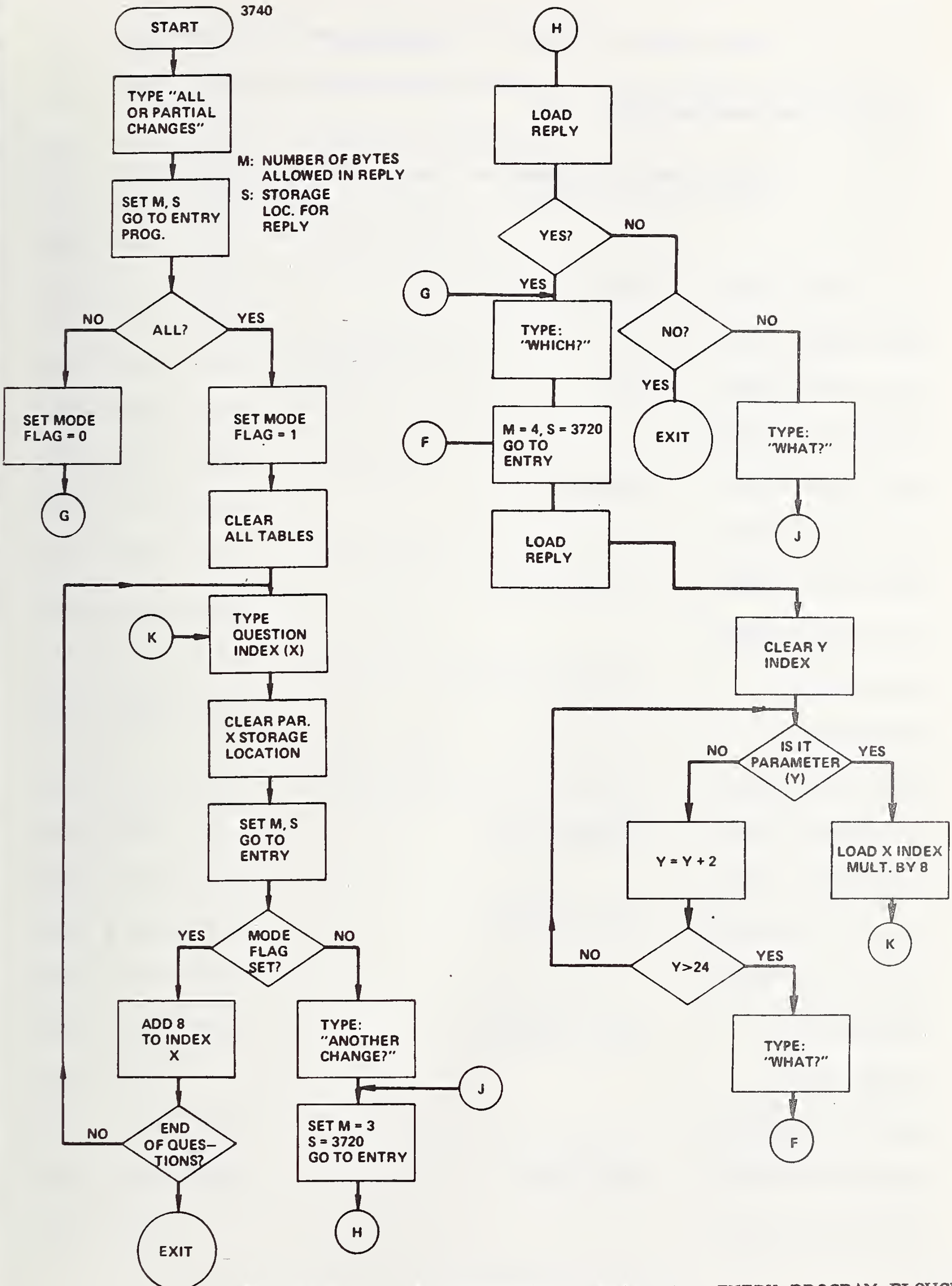


FIGURE 4-16. ACCELEROMETER DATA BLOCK ENTRY PROGRAM FLOWCHART

4.47 Storage Table for Type-Out Constants and Save Locations

This is a storage table for the Accelerometer Data Block Entry Program (see Section 4.46).

Storage Table for Type-Out Constants and Save Locations

```
38A0 393A 0009     FN
38A4 2B00 0004
38A8 3944 0008     TN
38AC 2B 04 0008
38B0 394E 0009     CAPAC.=?
38B4 2B78 000C
38B8 3958 000E     PICKUP MNFG?
38BC 2B10 0006
38C0 03D0          PICKUP MOD NO?
38C2 000F
38C4 2B20 000A
38C8 3968 000C     PICKUP SN=?
38CC 2B16 000A
38D0 3976 000B     AMP MNFG=?
38D4 2B30 0006
38D8 03E0          AMP MOD NO?
38DA 000F
38DC 2B40 000A
38E0 3982 0009     AMP SN=?
38E4 2B36 000A
```

```
38E8 398C 000B    CUSTOMER?
38EC 2B50 0010
38F0 3998 000F    DATE OF CALIB=?
38F4 2B60 0006
38F8 39A8 000D    EXCITER #1=?
38FC 2B66 0004
3900 39B8 000D    EXCITER #2=?
3904 2B6A 0004
3908 39C8 0009    STD #1=?
390C 2B70 0004
3910 39D4 0009    STD #2=?
3914 2B74 0004
3918 39F0 0007    GAIN?
391C 2B84 007C
3920 41CC CCA0    ALL OR PARTIAL CHANGES?
3924 CFD2 A050
3928 41D2 D4C9
392C 41CC A0C3
3930 4841 4E47
3934 C553 3F8D
3938 0AA0
393A C6C9 CCC5    FILE #=?
393E A3BD 3F8D
3942 0AA0
```

```
3944 D4C5 53D4      TEST #=?
3948 A3BD 3F8D
394C 0AA0
394E C341 5041      CAPAC.=?
3952 C32E BD3F
3956 8D0A 50C9      PICKUP MNFG=?
395A C34B 5550
395E A04D 4EC6
3962 47BD 3F8D
3966 0AA0
3968 50C9 C34B      PICKUP SN=?
396C 5550 A053
3970 4EBD 3F8D
3974 0AA0
3976 414D 50A0      AMP MNFG=?
397A 4D4F C647
397E BD3F 8D0A
3982 414D 50A0      AMP SN=?
3986 534E BD3F
398A 8D0A C355      CUSTOMER=?
398E 53D4 CF4D
3992 C5D2 BD3F
3996 8D0A 4441      DATA OF CALIB=?
```

```
399A D4C5 A0CF
399E C6A0 C341
39A2 CCC9 423F
39A6 8D0A C5D8      EXCITER #1=?
39AA C3C9 D4C5
39AE D2A0 A3B1
39B2 BD3F 8D0A
39B6 A0A0 C5D8      EXCITER #2=?
39BA C3C9 D4C5
39BE D2A0 A3B2
39C2 BD3F 8D0A
39C6 A0A0 53D4      STD #1=?
39CA 442E A3B1
39CE BD3F 8D0A
39D2 A0A0 53D4      STD #2=?
39D6 442E A3B2
39DA BD3F 8D0A
39DE 414F CED4      ANOTHER CHANGE?
39E2 48C5 D2A0
39E6 C348 414E
39EA 47C5 3F8D
39EE 0AA0
39F0 4741 C94F      GAIN?
39F4 3F8D 0AA0

>
```

5. DISCUSSION

Accurate reference accelerometers need to be calibrated with a minimum of time spent in recalibration consistent with accuracy requirements. The automated system for accelerometer calibration was set up to provide a precision calibration facility and to shorten the time required to perform calibrations. Another advantage of this type of system is that it is possible to collect more data on control accelerometers. Those control accelerometers which are known to be reliable are maintained as control units for measuring the repeatability of the system. One of the control accelerometers is calibrated each day the system is in use on two exciters and the results are kept on file. Although deviations from the normal sensitivity may be spotted by the operator, checking the deviations based on data collected over a long time frame is desirable. Data have been collected over a four-year period on the control accelerometers using the automated system. Figures 5-1 and 5-2 show the history at 400 Hz over a two-year time span. All the data in figure 5-1 are from one control accelerometer on one exciter. The data in figure 5-2 are for the same accelerometer (number 1) up to day 256. From day 256 on, the data are for a second control accelerometer (number 2) with the same nominal sensitivity. Figure 5-3 shows a histogram for 1972 for control accelerometer 1 in terms of accumulative average percent deviation from the mean for a frequency of 400 Hz on two exciters. This is based on 131 data points on each exciter. This gives a measure of the probability that a given data point will deviate from the mean sensitivity by a certain amount in percent. For example, figure 5-3 indicates that 90 percent of the time the deviation from the mean at 400 Hz should be 0.5 percent or less. Figure 5-4 shows the same histogram for 1973.

By accumulating data over a long period of time, a data bank can be valuable for quality control of the system. This type of information can be incorporated into a software program in the minicomputer to set upper and lower bands for deviations of the control accelerometer. For data falling outside these bands, the computer can be programmed to respond with messages on the TTY or if desirable, to halt the system. Instead of a go or no go diagnostic, an analysis of the current data from a control accelerometer could give deviations from the mean based on the mean in the data bank for each test frequency. These deviations could then be translated into frequency of occurence as seen in figures 5-3 and 5-4. A diagnostic message can be printed to indicate in what region the current data lie in terms of the probability density of the data bank.

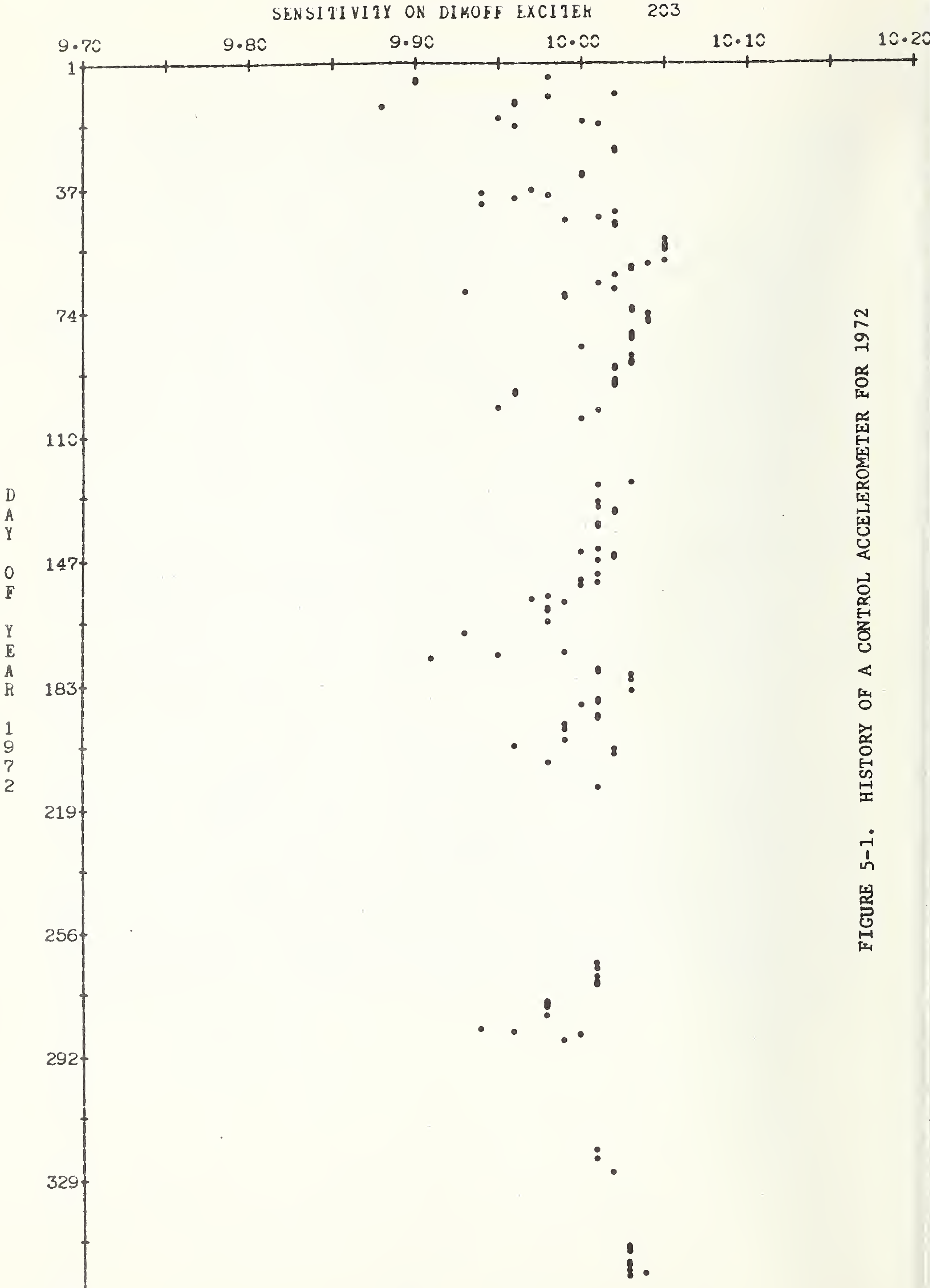


FIGURE 5-1. HISTORY OF A CONTROL ACCELEROMETER FOR 1972

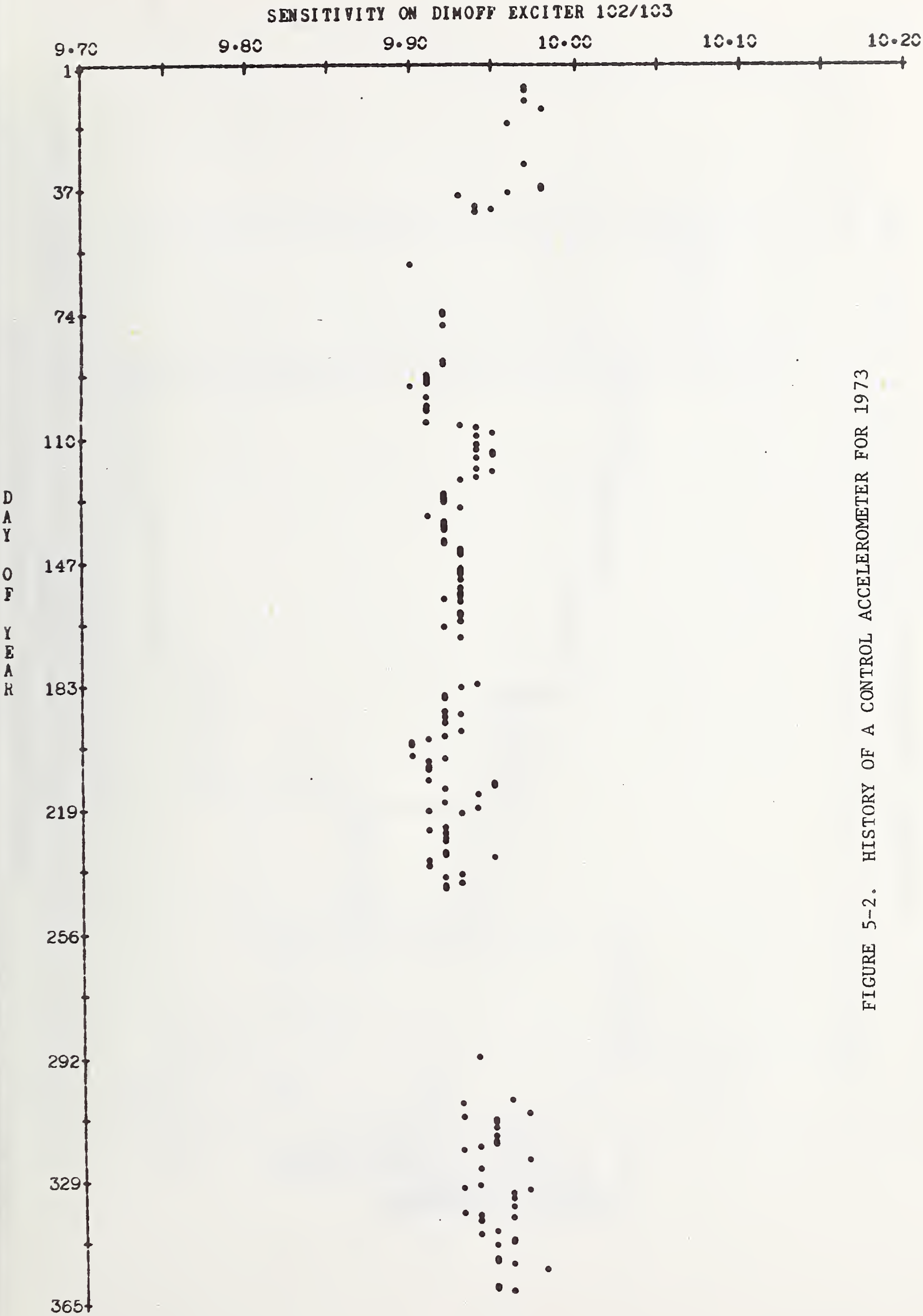


FIGURE 5-2. HISTORY OF A CONTROL ACCELEROMETER FOR 1973

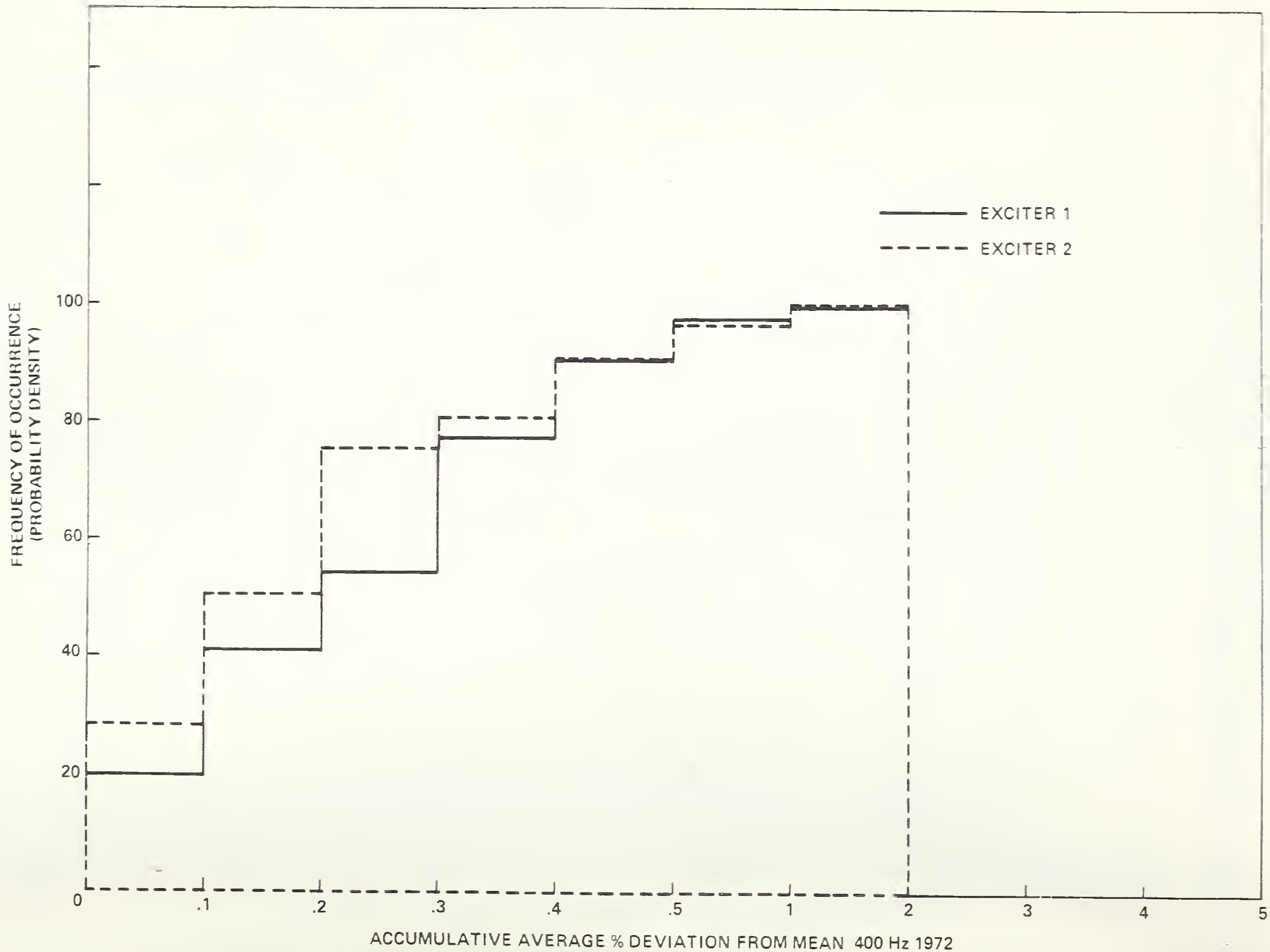


FIGURE 5-3. HISTOGRAM OF A CONTROL ACCELEROMETER FOR 1972

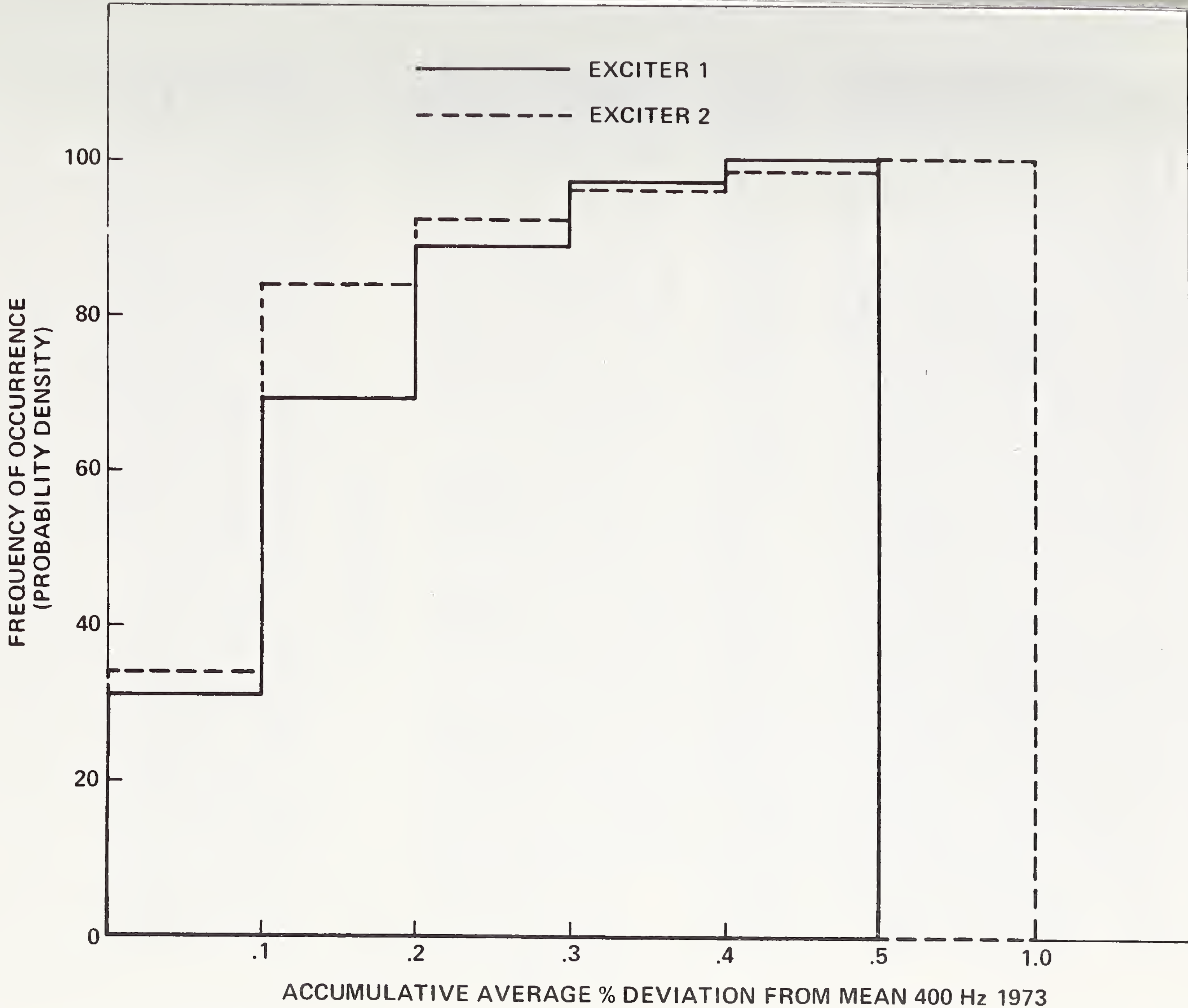


FIGURE 5-4. HISTOGRAM OF A CONTROL ACCELEROMETER FOR 1973

In Section 2 of this report the hardware components listed a wave analyzer and dc power supply as equipment for future expansion of the system. The wave analyzer is tunable by a dc voltage and can be programmed by remotely tuning it to read harmonic distortion. One of the digital to analog converters used with the x-y plotter can be used to tune the analyzer. Since the digital to analog converter has a range of ± 10 volts dc, a dc power supply is needed to supplement it to give ± 20 volts dc. This can be accomplished by switching in either a +10 volts or a -10 volts in series with the analog to digital converter. The software for the analyzer would tune the analyzer for each harmonic of a given test frequency, one at a time. The output from the analyzer would then be read by the DVM and the percentage distortion calculated for each harmonic. This could be printed out after each test point on the TTY or it could be stored and printed in a summary statement at the end of a test. Such a procedure would slow down the calibration process considerably. However, it would add valuable information regarding the calibration process.

An alternative procedure would be the use of a real time programable analyzer now commercially available. This could be programmed to take a spectrum analysis at each test frequency and the harmonic amplitudes could be stored for analysis and for a summary statement.

The major source of failure in the hardware has been in the interface cards. These cards were not as reliable as the other minicomputer hardware. During the course of over four years of operation, the main frame computer hardware has needed only two service calls. The interface cards needed several service calls during the first two years of operation. Apparently, most of the bugs were removed during these two years. The test instruments which are used by the automated system have proven reliable except for the ac/dc converters which have a fairly high rate of failure. A good procedure is to have a backup instrument for each test instrument used by the system. This is necessary if continuous use is to be made of the system.

6. ACKNOWLEDGMENTS

The sponsorship of this project by the Department of the Defense Calibration Coordination Group is gratefully acknowledged. The success of this project is indebted to the following people whose contributions are gratefully acknowledged: Seymour Edelman, Roscoe Bloss, and John Ramboz for planning and encouragement; Carson Meadors and Benton Durley III for component layout and wiring; Charles Federman for assisting in schematic drawings; Jim Pollard for checkout and testing; Phillip Stein for use of his monitor program and magnetic tape drive software; and Linda Ross for aid in manuscript preparation and typing.

USCOMM-NBS-DC

7. REFERENCES

1. B. F. Payne, "Absolute Calibration of Vibration Generators with Time-Sharing Computer as Integral Part of System", The Shock and Vibration Bulletin, 36, Part 6, Feb. 1967, pp. 183-194.

2. Samuel Levy and Raymond R. Bouche, "Calibration of Vibration Pickups by the Reciprocity Method", Journal of Research NBS 57, pp. 227-243 (1956) RP2714.

3. T. Dimoff, "Electrodynamic Vibration Standard with a Ceramic Moving Element", Journal of the Acoustical Society of America, Vol. 40, No. 3, Sept. 1966, pp. 671-676.

4. B. F. Payne, "An Automated Precision Calibration System for Accelerometers", 17th National Aerospace Instrumentation Symposium, May 1971.

5. V. A. Schmidt, S. Edelman, E. R. Smith, and E. T. Pierce, "Modulated Photoelectric Measurement of Vibration", Journal of the Acoustical Society of America, Vol 34, No. 4, pp. 445-458 (1962).

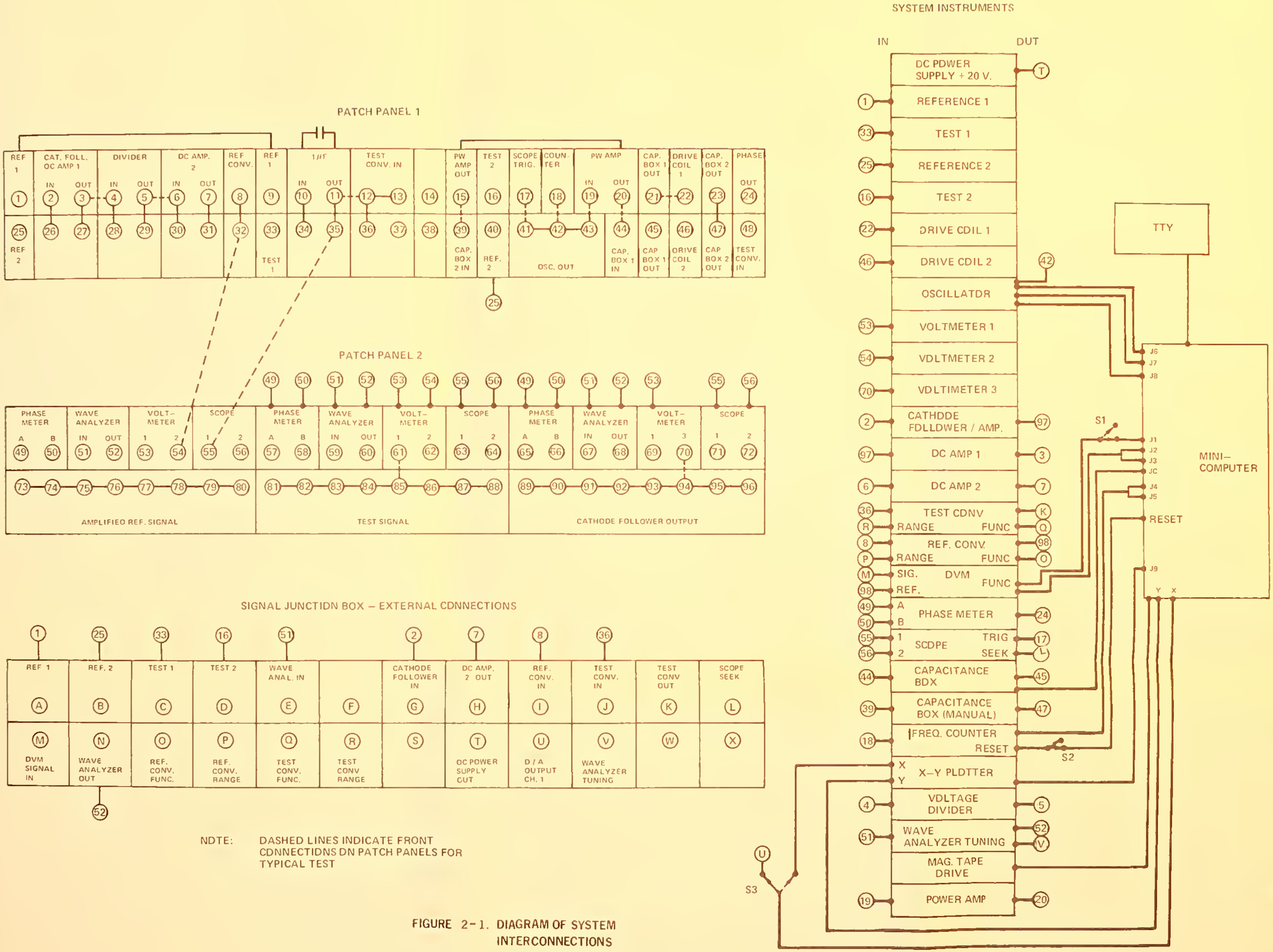


FIGURE 2-1. DIAGRAM OF SYSTEM INTERCONNECTIONS

NBS-114A (REV. 7-73)

U.S. DEPT. OF COMM. BIBLIOGRAPHIC DATA SHEET	1. PUBLICATION OR REPORT NO. NBSIR 74-482	2. Gov't Accession No.	3. Recipient's Accession No.

4. TITLE AND SUBTITLE AN AUTOMATED SYSTEM FOR PRECISION CALIBRATION OF ACCELEROMETERS	5. Publication Date April 1974
	6. Performing Organization Code
7. AUTHOR(S)	8. Performing Organ. Report No.
9. PERFORMING ORGANIZATION NAME AND ADDRESS NATIONAL BUREAU OF STANDARDS DEPARTMENT OF COMMERCE WASHINGTON, D.C. 20234	10. Project/Task/Work Unit No.
	11. Contract/Grant No.
12. Sponsoring Organization Name and Complete Address *(Street, City, State, ZIP)*	13. Type of Report & Period Covered
	14. Sponsoring Agency Code CCG 69-13

15. SUPPLEMENTARY NOTES

16. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here.)*

The report describes an automated system for accelerometer calibration under realtime control by a small, dedicated digital computer. The hardware components of the system and the software programs are given. The software automatically regulates the rate and amount of data collected based on analysis of input data. Printout of the frequency response of test accelerometers is on a teletypewriter and also can be stored on a magnetic tape. Manual operation of the system is also described.

17. KEY WORDS *(six to twelve entries; alphabetical order; capitalize only the first letter of the first key word unless a proper name; separated by semicolons)*

Acceleration, automation, calibration, measurements, minicomputer, shakers, standards, transducers, vibration, vibration exciters, vibration pickups.

18. AVAILABILITY	19. SECURITY CLASS (THIS REPORT)	21. NO. OF PAGES
[X] Unlimited [ ] For Official Distribution. Do Not Release to NTIS	UNCLASSIFIED	
[ ] Order From Sup. of Doc., U.S. Government Printing Office Washington, D.C. 20402, SD Cat. No. C13 [ ] Order From National Technical Information Service (NTIS) Springfield, Virginia 22151	20. SECURITY CLASS (THIS PAGE) UNCLASSIFIED	22. Price

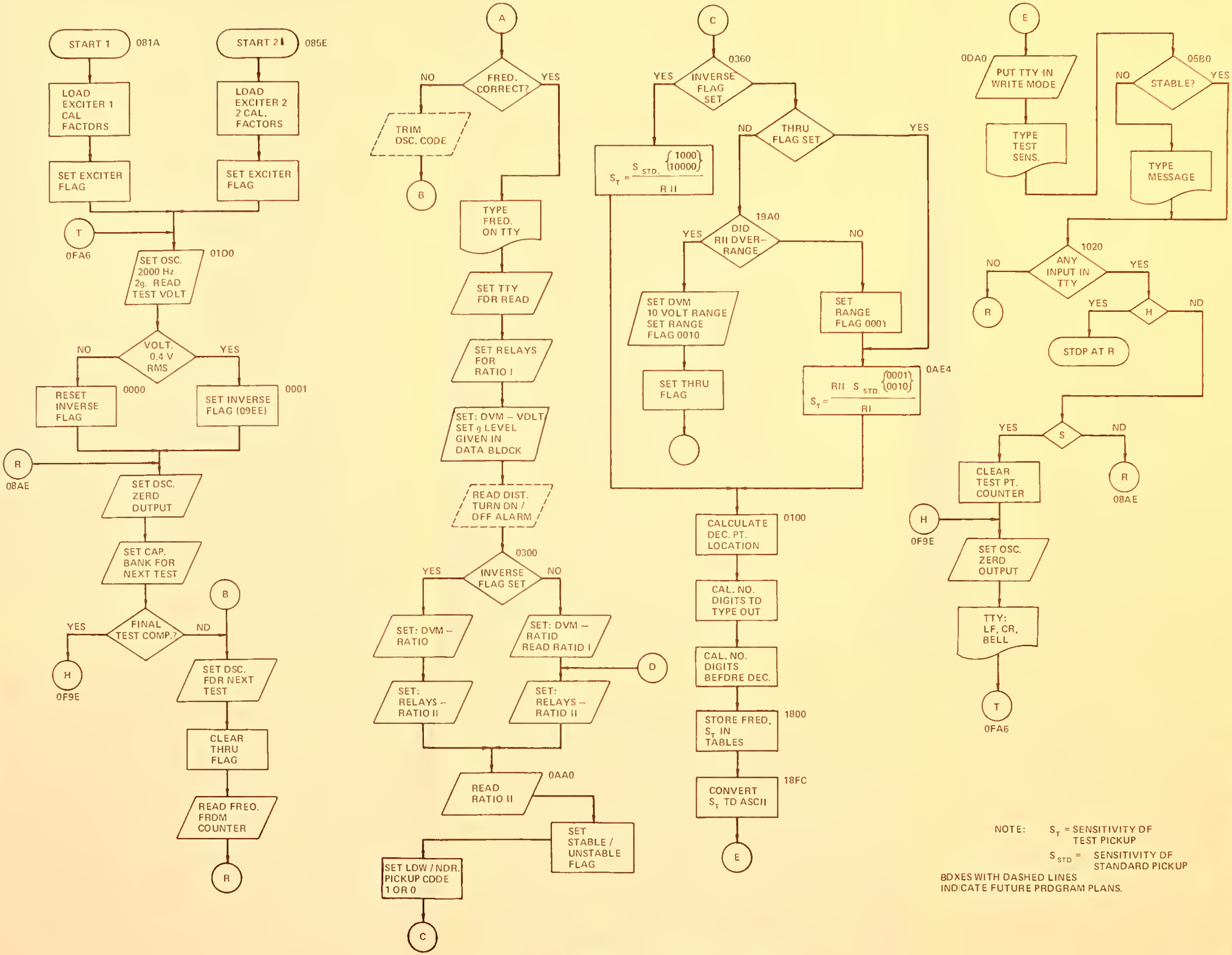


FIGURE 3-4. SUMMARY FLOW CHART