NASA CR-
160335

FINAL REPORT CONTRACT NAS 9-15522

"THE DEVELOPMENT OF SPECIAL INSTRUMENTATION TO SUPPORT CARDIOVASCULAR STUDIES IN SPACE"

Submitted by

Dr. Richard J. Gowen,
Vice President and Dean of Engineering;

Dr. Michael J. Batchelder,
Assistant Professor, Department of Electrical Engineering;

Robert Brown,
Graduate Research Assistant, Department of Electrical Engineering

(NASA-CR-160335) THE DEVELOPMENT OF SPECIAL INSTRUMENTATION TO SUPPORT CARDIOVASCULAR STUDIES IN SPACE (South Dakota School of Mines and Technology) 123 p

N79-80034

Unclas
00/52 45828

RECEIVED
NASA STI FACILITY
ACCESS DEPT.

Division of Engineering

South Dakota School of Mines & Technology

Rapid City, South Dakota 57701

FINAL REPORT CONTRACT NAS 9-15522

"THE DEVELOPMENT OF SPECIAL INSTRUMENTATION TO SUPPORT CARDIOVASCULAR STUDIES IN SPACE"

'Micro-Computer Control of the Lower Body Negative Pressure Device and the Limb Volume Measuring System'

Submitted by

Dr. Richard J. Gowen,
Vice President and Dean of Engineering;

Dr. Michael J. Batchelder,
Assistant Professor,
Department of Electrical Engineering;

and Robert Brown,
Graduate Research Assistant,
Department of Electrical Engineering

Funded under NASA contract number NAS 9-15522

July 26, 1979

1.0 ABSTRACT

The micro-computer controlled Experiment Support System (ESS) containing both a controller for the Lower Body Negative Pressure Device (LBNPD) and the Limb Volume Measure System (LVMS) has been tested; design changes have been made; and the system has been demonstrated to perform in accordance with all specifications. The equipment evaluated, modified, and verified was originally developed as part of the Joint Air Force Academy Project.

2.0 INTRODUCTION

The Experiment Support System hardware developed under the original contract with the Air Force Academy has been transferred to the South Dakota School of Mines and Technology (SDSM&T). When work on the project terminated at the Air Force Academy, the system design and construction had been completed; however, the system had not been tested. The work on the project at SDSM&T has been to test the system, isolate problems, and make design changes so the system performs properly.

Initial progress on the project at SDSM&T was slow due to inevitable difficulties in transferring the project after a one year hiatus. The necessary system design changes have been made; missing elements have been fabricated; and system performance testing has been completed.

The micro-computer controlled Experiment Support System consists of a controller for both the Lower Body Negative Pressure Device (LBNPD) and the Limb Volume Measuring System (LVMS). The micro-computer shown in Figure 2.1 controls not only the LBNPD pressure, but also the LVMS data acquisition and processing.

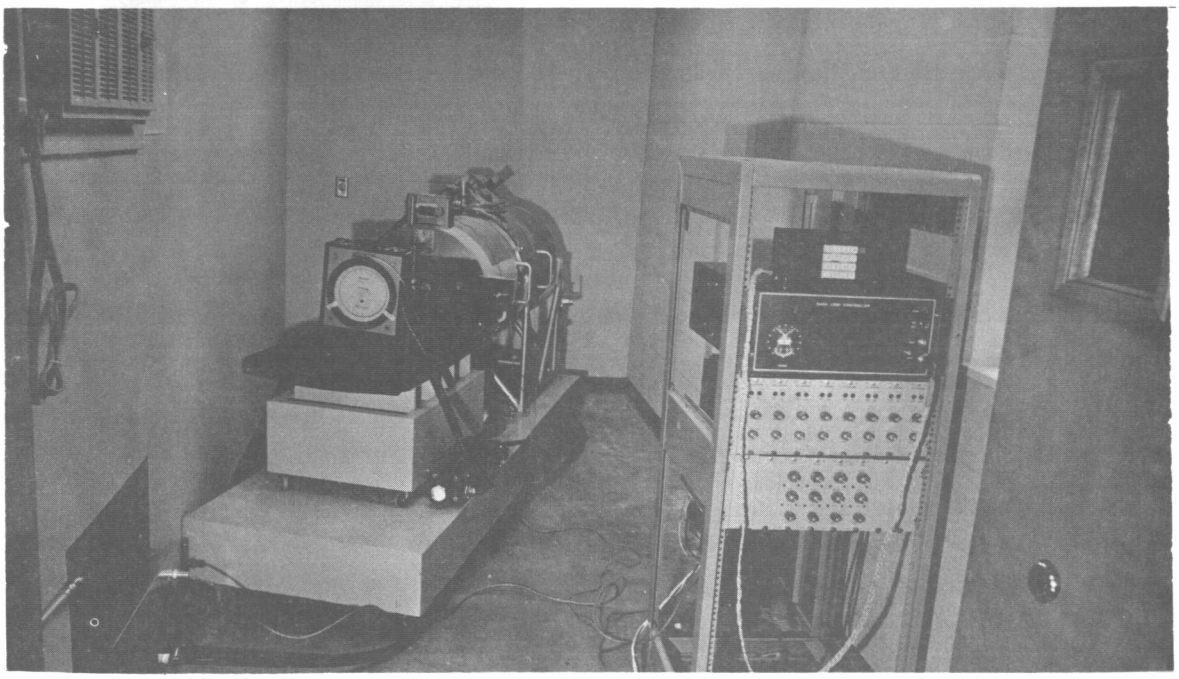


Figure 2.1 LBNPD and micro-computer controller in test cell at SDSM&T. The micro-computer and input keyboard are the top units in the equipment rack. The LBNPD is sealed for controlled pressure verification of the system operation.

The transition of LBNPD pressure from one pressure level to another using a manually controlled valve was a major source of variability in results obtained during the Skylab M092 Cardiovascular Experiments using the LBNPD. The use of the micro-computer provides for repeatability in the application of the vacuum pressure to stress subject in the LBNPD. The use of a micro-computer provides for a marked improvment in the operation of the LVMS over that used in Skylab.

3.0 SYSTEM DESCRIPTION

3.1 HARDWARE

The system, illustrated in Figure 3.1, consists of a micro-computer which controls electrical solenoid values to automatically vary the vacuum pressure applied to the LBNPD. The micro-computer provides automatic calibration and signal processing for the Skylab LVMS.

The operation of the system is controlled from the hand-held operator keyboard and display module. In operation the operator depresses an appropriate key to activate the desired micro-computer controller program as described in Section 4.0. The micro-computer controller program provides for automatic operation of the LBNPD using a vacuum profile identical to that employed in Skylab LBNPD experiments. Additionally, all system functions can be controlled manually from the keyboard through the use of appropriate codes.

The micro-computer software has been developed to provide for the safe operation of the LBNPD in the event of a power failure or a computer malfunction. The operation of the system requires that power be provided to energize the ABORT solenoid valve before vacuum may be applied to the LBNPD. Loss of power will cause this "abort" valve to de-energize and the LBNPD will rapidly return to ambient pressure. A vacuum pressure greater than 60 torr will cause the micro-computer to open the "abort" valve and return the system to ambient pressure. The normal failsafe protective limits of the LBNPD as used in Skylab are still operable and limit vacuum to 70 torr.

The micro-computer automatically completes the calibration sequence for the LVMS leg bands. The operator need only identify the leg bands used to the micro-computer through a coded input of band identification using the keyboard. The calibration curves of the leg bands are stored in the micro-computer and the correct calibration curve is selected automatically by the micro-computer. The electrical output of the leg bands is translated through the calibration curve to provide an output voltage which is displayed as the percent change in leg volume. Thus, the operator need only place the leg bands on the subject, key in the band identifier; and the micro-computer then completes the calibration using the standard Skylab procedure. The output of the leg bands is then presented as percent volume change.

The micro-computer provides outputs both in the form of digital displays and as voltages proportional to the two leg bands and the LBNPD vacuum pressure.

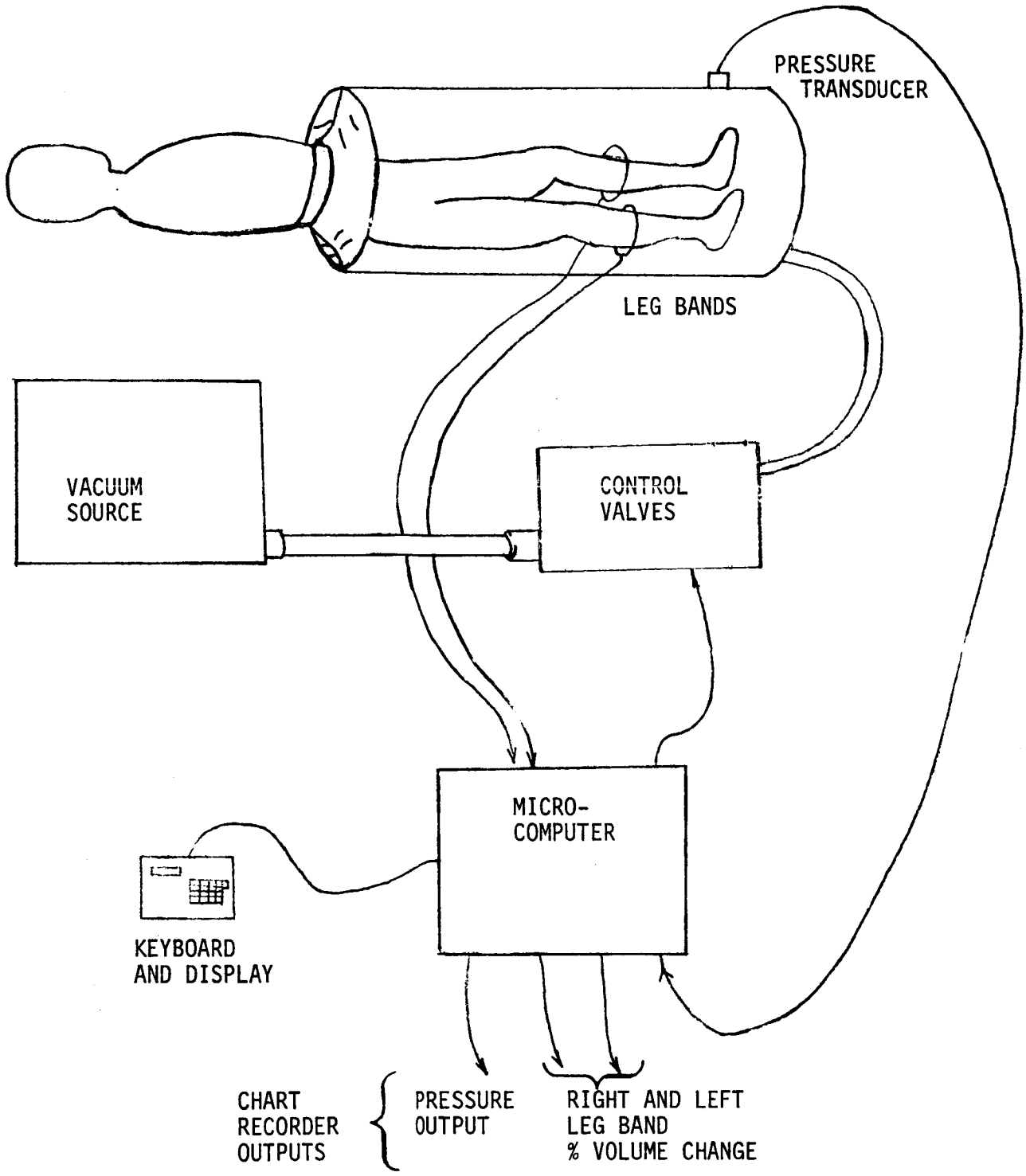


FIGURE 3.1 SYSTEM DIAGRAM

3.1.1 PRESSURE CONTROL

The system source of vacuum is two vacuum cleaner motors shown in Figure 3.2. The vacuum source is discussed further in Section 5.0 (System Results). The vacuum source is connected to the valve manifold consisting of five computer controlled valves shown in Figures 3.3 and 3.4. The valves are binary; each is completely open or closed. Four of the valves in the original design controled flow through the calibrated orifices connected to the manifold. There orifices were adjusted to a 1, 2, 4, 8 flow ratio. The fifth valve is a failsafe feature which allows the LBNPD pressure to rapidly reach ambient pressure. This can be performed manually by the operator using the 'A' command or automatically by the computer if the vacuum pressure exceeds 60 torr.

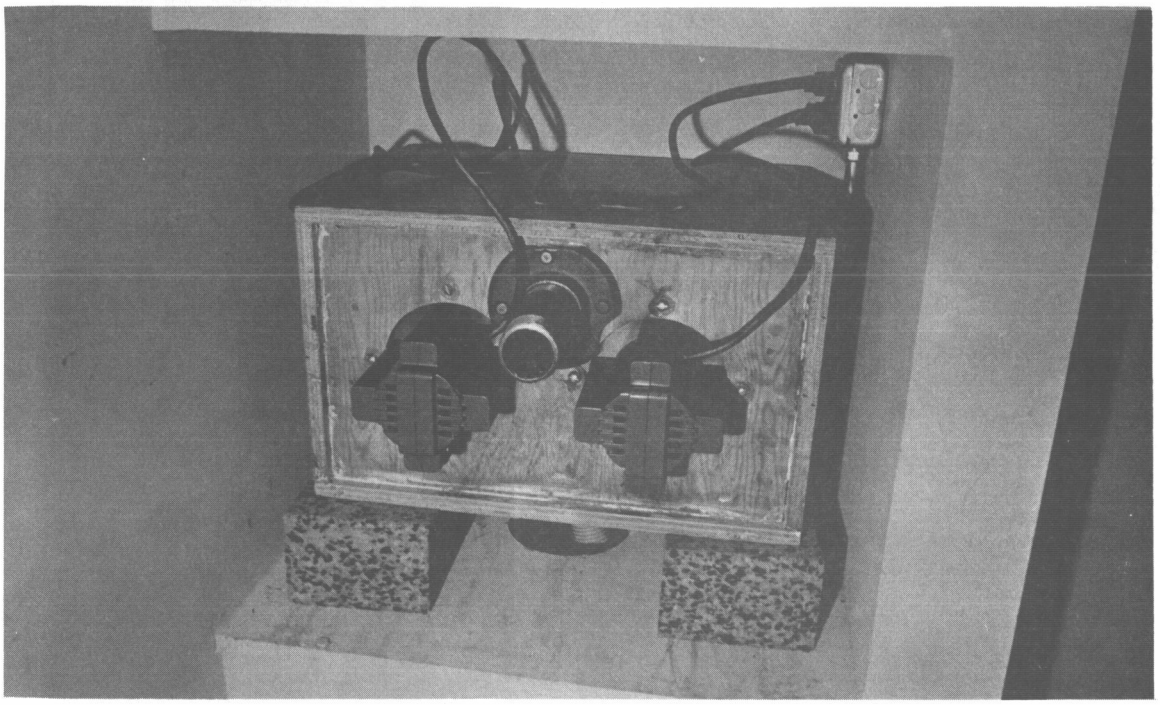
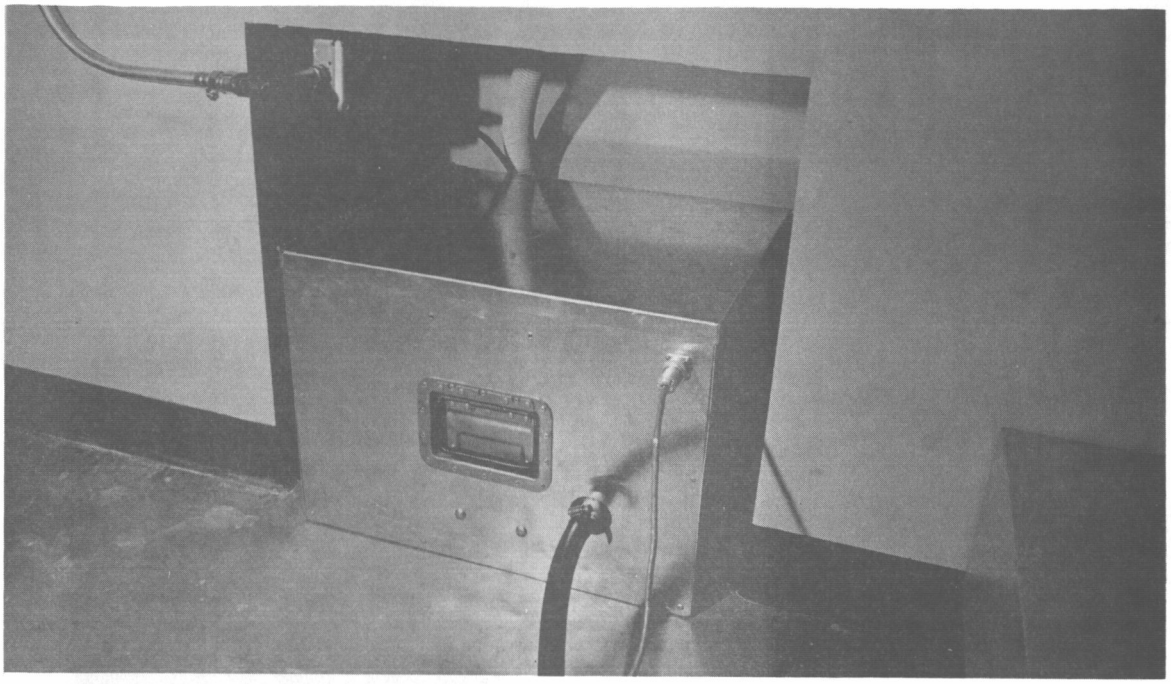


Figure 3.2 The vacuum source (two Sears A1-2555-01 Motor Blower Assemblies for shop vacuum 758-17821)

The original system was designed using four valves to achieve 16 levels of effective valve control positions. However, changes in the system that have resulted during the test and evaluation portion of the work of this contract have demonstrated that acceptable results can be achieved by using only one of the valves in combination with a special control algorithm as described in Section 6.0.

The LBNPD also contains a mechanical failsafe valve which automatically opens at -70 torr or can be manually operated by the subject in the tank.



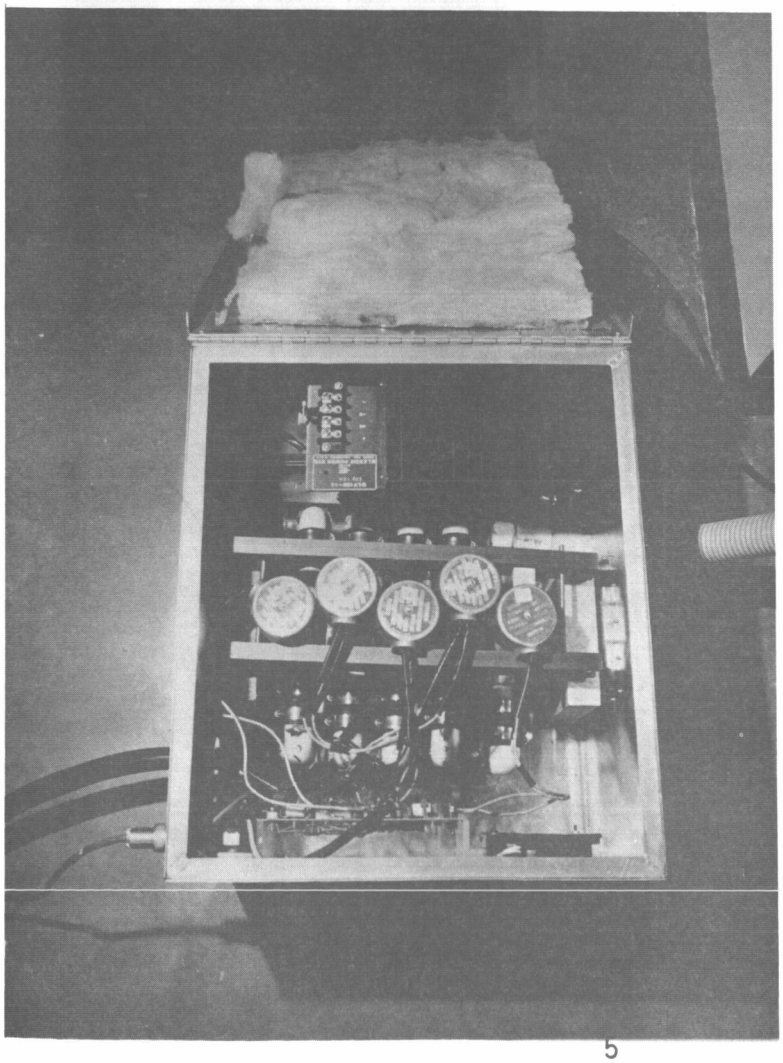


Figure 3.3 (Above) Valve Assembly (vacuum input shown on the top; outputs at the bottom; control cable at the top right.)

Figure 3.4 (Left) Valve Assembly (vacuum manifold at the top, LBNPD manifold at the bottom.)

The pressure transducer in the LBNPD closes the pressure control loop. The original pressure transducer mounted on the LBNPD was apparently damaged in the transfer of the equipment from the Air Force Academy and is nonfunctional. It was replaced by a transducer and amplifier, details are shown in Appendix 2, Figure 2. The replacement transducer requires a 10 volt dc excitation and a gain of approximately 200. The transducer and amplifier feed into the pressure A/D converter in the computer as shown in Appendix 2, Figure 3 and are equivalent to the standard Skylab LBNPD pressure transducer output.

The operator sets a desired pressure with the 'D' command or starts the automatic pressure profile sequence with the 'B' command from the keyboard. Using the actual pressure in the tank derived from the pressure transducer through the A/D converter, the computer determines the valve setting necessary to bring the LBNPD pressure to the desired pressure.

Both pressure level and rate of change in pressure are controlled by the micro-computer. While the present hardware was constructed using four control valves and the "abort" valve, improvements in the system timing and control sensitivity levels have resulted in improved performance through the control of only one valve. Thus, the final system uses the "abort" valve and one control valve, the other three valves are deactivated. To control the vacuum pressure at a desired level, the vacuum is applied to the LBNPD whenever the actual pressure differs by more than .25 torr. Vacuum pressure is controlled as vacuum is increased to within .25 torr. When the level of vacuum is to be returned toward ambient pressure, then either the natural leakage of the subject's waist seal is used or the "abort" valve is activated. In the automatic mode, the "abort" valve is employed to return pressure to ambient levels at the conclusion of the Skylab profile. The pressure is read and the valve adjusted every 50 ms. With this control system the pressure is maintained constant within approximately $\pm$ 1 torr.

3.1.2 LEG BAND DATA ACQUISITION AND PROCESSING

The second major system function is processing leg band data. Two leg bands are used during an experiment: leg band one (right) and leg band two (left). There is a single leg band oscillator-amplifier card and two leg band controller cards, one for right and one for left. The block diagram is shown in Figure 3.5; the detailed schematics are found in Appendix 2, Figures 4 and 5.

The simplified leg band schematic is found in Figure 3.6; the detailed schematic is found in Appendix 2, Figure 7. The sine wave input from the oscillator-amplifier card is clipped to a square wave of precise amplitude and fed to an RC voltage divider. As the leg (one plate of the capacitor) expands, the capacitance increases, increasing the voltage output. When the calibrate relay is activated, the calibration plate introduces a fixed capacitance into the circuit, and a known calibration voltage change in the output voltage occurs.

The leg band data processing system must be initialized before data can be acquired; this is automatically performed by the computer. Each leg band has been calibrated for percent volume change as a function of output voltage from the oscillator-amplifier card. A typical characteristic plot is shown in Figure 3.7. Before using a leg band the equipment must be initialized to give the proper percent volume change output for a given voltage input. There are two steps to this process which determine two points on the calibration curve.

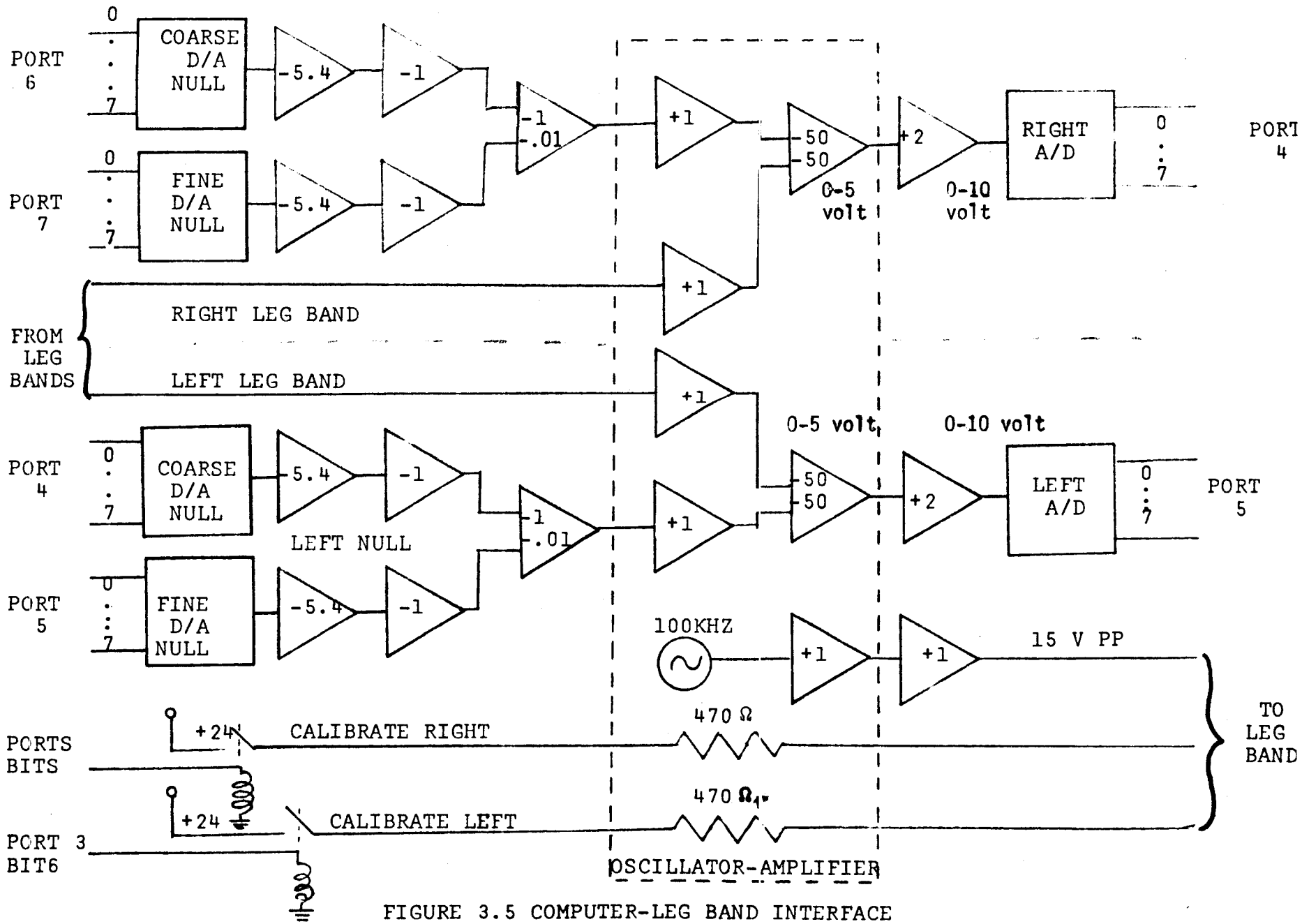


FIGURE 3.5 COMPUTER-LEG BAND INTERFACE
(3 Cards: Oscillator-Amplifier; LBNPD Controller right; LBNPD Controller left)

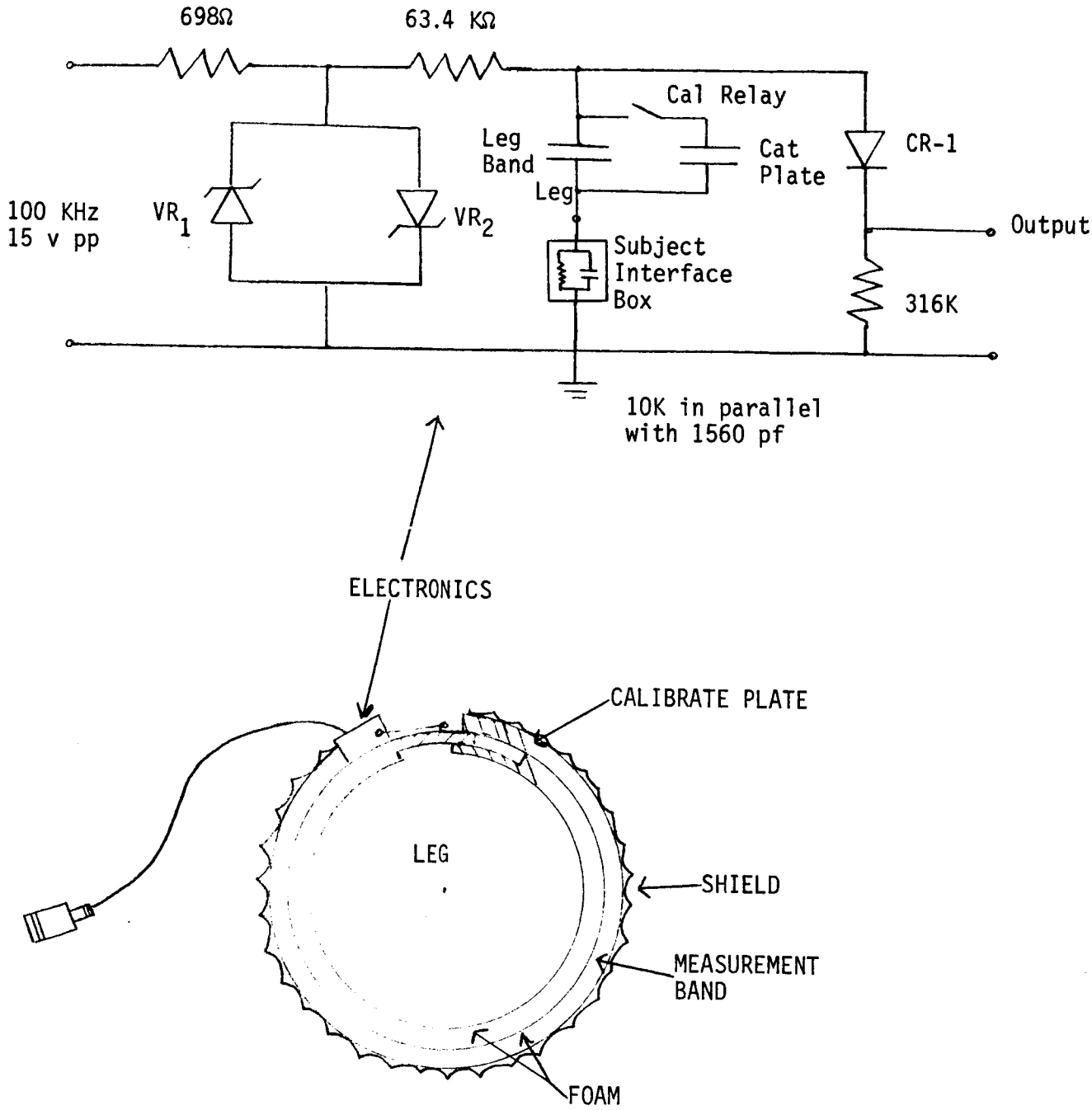
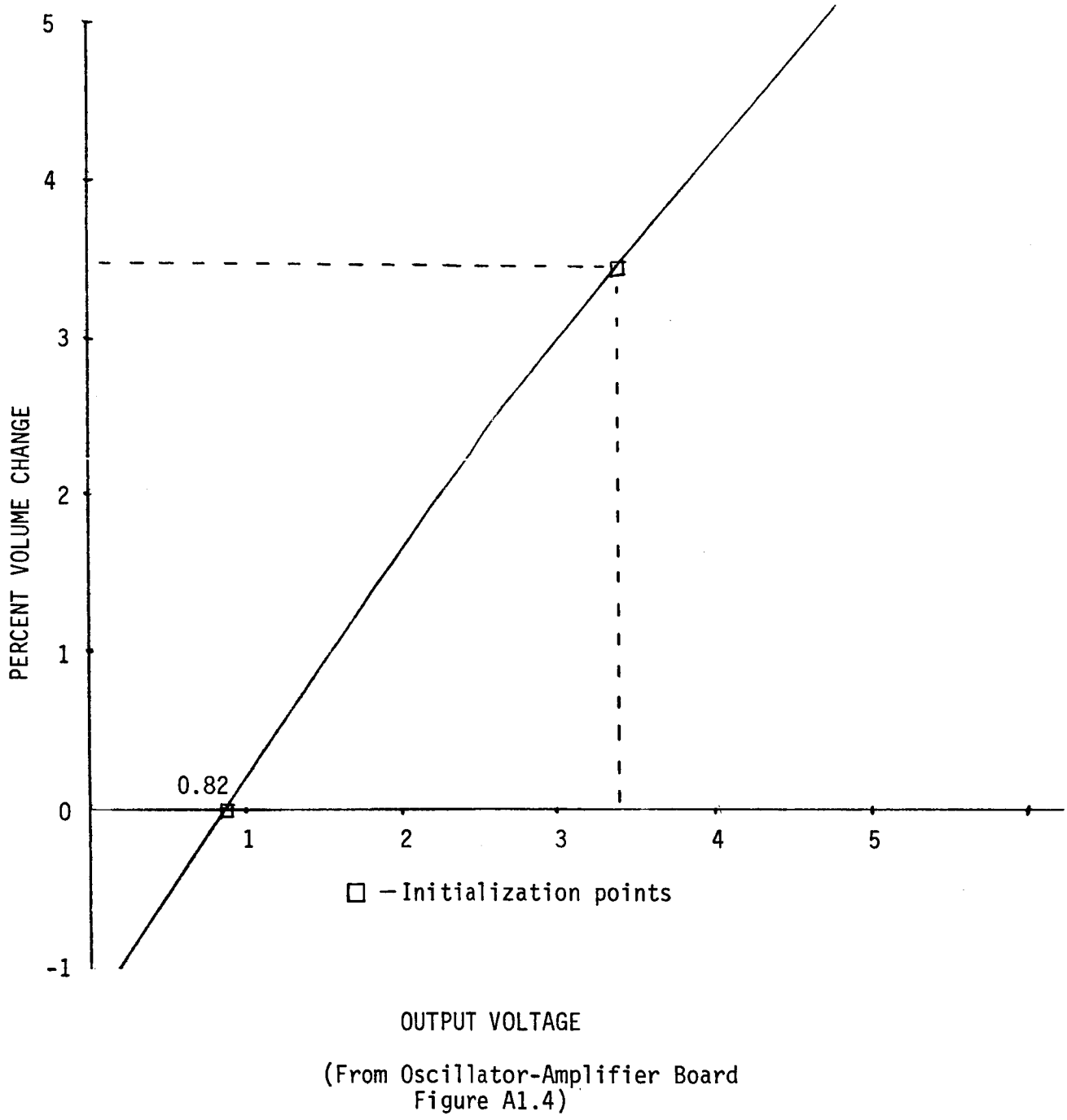


FIGURE 3.6 SIMPLIFIED LEG BAND SCHEMATIC

The electronics are potted in the module of the leg band. The ground return is through the subject interface box simulated for tests by a 10K resistor in parallel with 1560 pf.



(From Oscillator-Amplifier Board
Figure A1.4)

FIGURE 3.7 TYPICAL LEG BAND CHARACTERISTIC

First, the zero percent volume change point is established by connecting a null voltage to the leg band card and varying this voltage until the corresponding voltage of 0.82 volts is established at the output of the leg band card. This essentially moves the characteristic plot back and forth until the zero percent point corresponds to 0.82 volts. Next, the slope of the characteristic plot is adjusted until the upper initialization point on the curve is established. This point is determined by closing the calibrate relay in the leg band which adds a predetermined capacitance in the leg band itself. This calibration plate capacitance produces a predetermined volume change reflected as a known output voltage. Thus, each leg band has a calibration number which is the percent volume change produced when the calibration relay is closed.

In the original manual system, two potentiometers were used for each leg band. One for adjusting the null and one for adjusting the gain. Adjusting the gain is merely an adjustment of the slope of the characteristic plot. The micro-computer controlled system has the advantage that this calibration procedure can be performed automatically without any manual intervention from the operator. The operator merely identifies which leg bands are being used and the micro-computer selects the prestored characteristic plot for the leg bands to perform the automatic calibration.

In the computer operated system the null voltage is generated by D/A converters. The gain is not adjusted physically, (it is constant at 50); however, the equivalent result of adjusting the gain is achieved by software.

This automatic initialization process is performed each time the Begin sequence is selected by the operator using the 'B' command. After initialization is completed, leg band data is acquired and processed.

The 0-5 volt data from the oscillator-amplifier card is amplified to a 0-10 volt signal for A/D conversion. Every 50ms the A/D converters are read, the data is converted to percent volume change; then displayed on the operator panel and sent to the chart recorder outputs.

The 50ms period for both the pressure control and leg band data processing is established by a 100 Hz clock shown in Appendix 2, Figure 3(b). This clock drives the interrupt line of the computer and provides the time source for the automatic pressure sequences. The software does data acquisition and processing every fifth interrupt for an effective 50ms interval.

During each 50ms interval the processed data are sent to three D/A converters to be available for driving chart recorders. The Chart Recorder Driver schematics are found in Appendix 2, Figure 6. These outputs are right and left leg band data, and pressure.

3.2 SOFTWARE

The software operates at two levels: background and foreground. Foreground operates when an interrupt is processed (the only interrupt in this system is the 100 Hz clock). Background operates at all other times. Appendix 3 contains flow charts for all routines and Appendix 4 contains a summary calling sequence.

The background scans the keyboard waiting for the operator to enter a command. The operator can always regain control of the system by pushing the reset key on the keyboard. RESET is also called when the system power is switched on. RESET initializes the system to manual control and maximum pressure rate (15 torr/sec), sets the display to pressure, and then enters the keyboard scan loop. The background normally just waits for the operator to push a key. Thus, under normal operation, all commands are available at any time, regardless of what the system is doing in the foreground. The commands available are described in Section 4.1 and summarized in Appendix 1. The flow charts for the command routines are found in Appendix 3.

Of special interest are the commands which initiate the foreground operation: select desired pressure, and begin automatic sequence. Selecting a desired pressure (command 'D') sets the pressure control profile to manual; BEGIN (command 'B') sets the pressure control to automatic, generating the pressure profile stored in the ROM as shown in Figure 3.8. Both commands enable interrupts which initiate foreground operation.

BEGIN also performs automatic calibration of both left and right leg bands so the data acquired in the foreground can be processed correctly. The only operator input BEGIN requires is the ID numbers of the leg bands used. The calibration procedure consists of nulling the leg band so that its current size represents zero percent volume change, then closing the calibrate relay in the leg band and calculating the gain factor. The NULL and GAIN routine flow charts are found in Appendix 3. The NULL values (BIAS1, BIAS2) and gain (GAIN1, GAIN2) are used in the foreground to process the leg band data.

The foreground program is activated everly 10ms. When the clock interrupt occurs, the CPU automatically disables further interrupts (until an enable instruction is executed) and the interrupt program saves all CPU registers (to be restored before returning to the background). The variable INTCT is used to cause actual processing every fifth interrupt i.e., every 50ms. The foreground maintains pressure control and processes leg band data. The variable VACLK determines how often the pressure control valves are adjusted. Currently it is set to call VADJ, the valve adjust routine, every 50ms.

The VADJ routine reads the pressure transducer input from the A/D. For safety, if the vacuum pressure is greater than or equal to 60 torr, it jumps to the IABORT routine with the same result as if the operator had entered the "A" (ABORT) command. The pressure is brought from the old desired pressure level to the new desired pressure level at a constant rate. When the real pressure reaches the desired pressure, the level control segment of the program is entered. Otherwise, it continues to use rate control to maintain the rate of change at 5 torr/sec or 15 torr/sec as selected by the operator (with 'E1' or 'E2' command).

The next function of the foreground pressure control is maintaining the time for the automatic pressure control sequence initiated by the BEGIN command. Time is maintained by three separate software clocks. ATIM is a count-down timer which holds the dwell time at a given pressure. PTIM is a count-down timer which holds the time at the current desired pressure. PTIME is PTIM converted to an up-counter for display.

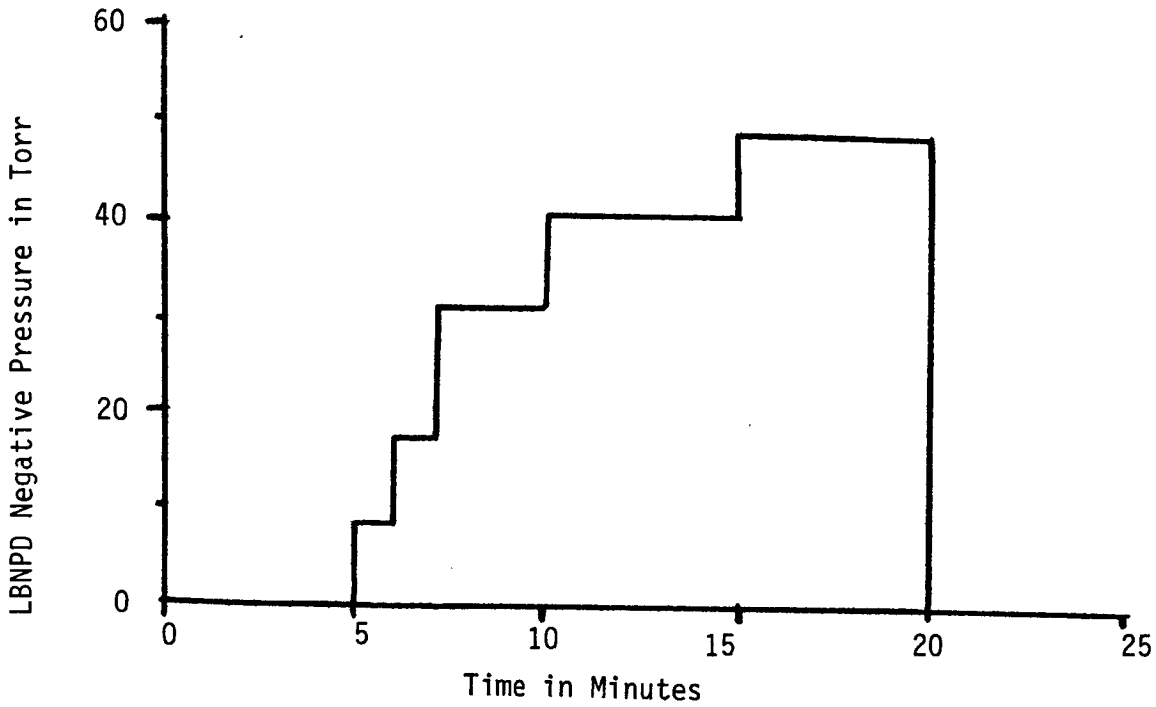


FIGURE 3.8 LBNPD Negative Pressure Profile

If ATIM, the dwell time down-counter, is decremented to a negative value, it is time for the next level in the automatic sequence and PAUTO is called. It resets ATIM to the next dwell time, sets the desired pressure to the value from the table, and then clears PTIM, the time at current pressure down-counter.

Every second (every 20 processed interrupts) PTYME is called to update all three timers.

At this point in the interrupt sequence the valve setting, VALVE, calculated earlier is output to the valve drivers to set the valves.

Now the second major foreground function of processing leg band date is performed. The routine LBRDR performs this function by reading the leg bands and converting them to a binary percent volume change. The leg band value from the A/D is adjusted using the BIAS and GAIN calculated earlier in the NULL and CALIB routines of the BEGIN command. The COMP (compensation) routine performs this by calculating [(INPUT - BIAS) * GAIN] + BIAS. This compensated value should be the same value that would have been read from the original manual unit. It is not yet a volume change value, but a voltage adjusted for gain and offset by software instead of hardware.

This compensated voltage from the leg band is converted to percent volume change by the routine APPROX which calls VALFD. VALFD does a linear interpolation from a characteristic curve table (unique to each leg band) relating voltage to percent volume change.

The results generated during this interrupt period are then output through three 0 to 10 volt D/A converters to the chart recorder connectors on the computer rear bulkhead. These results are right and left leg band percent volume change and the LBNPD pressure.

Finally, the display is updated according to the display mode selected by the operator. Either the leg band percent volume change, the time, or the actual and desired pressured are displayed depending on which display mode was selected by the operator (commands F0, F1, F2). Then the registers are restored to their original values when the interrupt occured, interrupts are enabled, and control returns to the background.

4.0 SYSTEM USE

Once the system has been set up and all calibrations performed the keyboard commands described in the next section can be used to operate the system. The system initial set up procedure is found in Appendix 6.

4.1 KEYBOARD COMMANDS

The system is controlled by commands entered on the keyboard by the operator. The console with the 17 key keyboard and 5 character display is shown in figure 4.1. The commands are: 'A', 'B', 'C', 'D', 'E', 'F', and RESET (the 17th key at the upper right of the keyboard). The commands normally can be activated at any time during operation of the system.

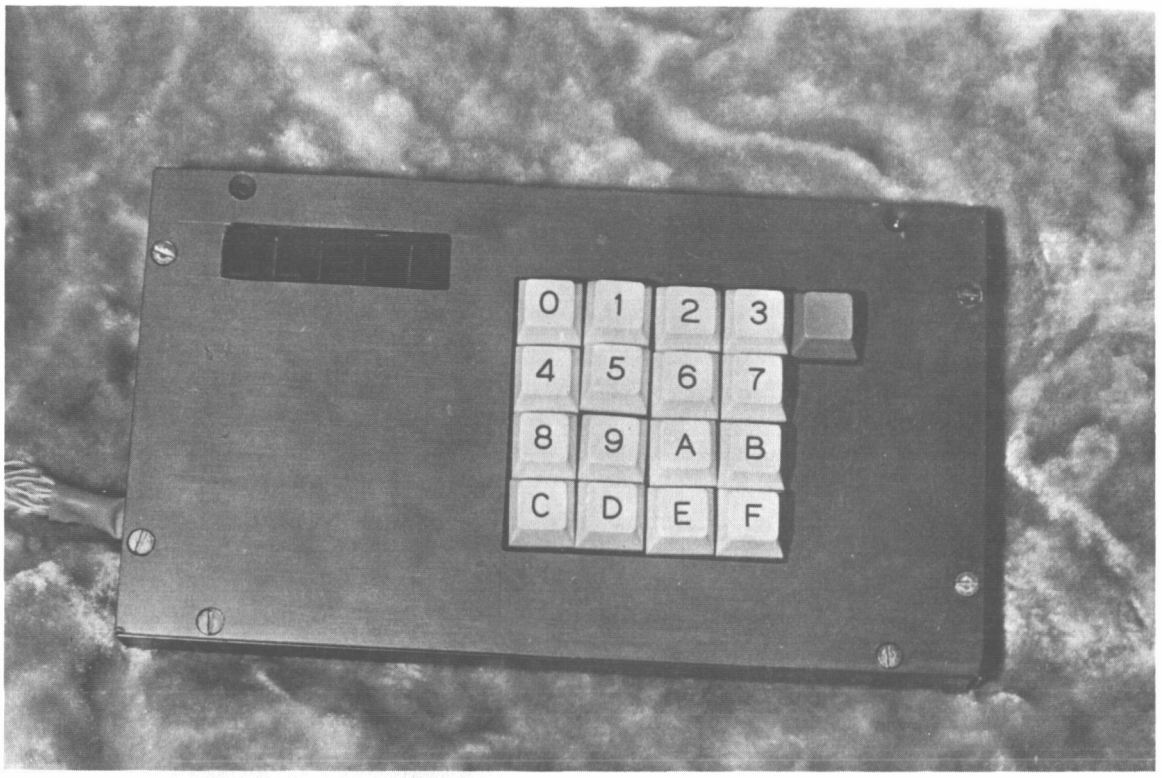


Figure 4.1 Operator Console

The RESET command can be used to return the system to a known condition. It closes and deactivates all valves, resets and deactivates all timers, resets the display to all zeros, and resets the temporary storage locations.

The 'A' (ABORT) command can be entered at any time to terminate the experiment quickly. The command brings the LBNPD to ambient pressure as rapidly as possible by closing the four control vavles and opening the abort valve. The display shows "dEAdd" and the operator must use the reset command to reactivate the system. This command continues the transmission of data to chart recorder outputs.

A normal experiment begins with the 'B' (BEGIN) command which starts the automatic control sequence. After the 'B' command is entered the display shows 'A100' and the system waits for the operator to enter the two digit leg band ID. These ID's are found in Table 4.1. After the Operator enters the ID for leg band one (the right leg band) the display shows 'bA200' and waits for leg band two (the left leg band) ID. If both are valid ID numbers the micro-computer performs the automatic calibration of the leg bands by nulling the output of the leg band oscillator-amplifier card and calculating the software gain factor. If the gain calculated is less than one the display will show 'bAd01' or 'bad02'. This may indicate problems such as entering the wrong ID for a leg band, improper grounding, incorrect adjustments in the leg band signal processing cards.

Leg Band	Keyboard Code	Leg Band	Keyboard Code	Leg Band	Keyboard Code
AF	00	AU	19	CI	32
AG	01	CY	1A	CJ	33
AM	02	BA	1B	CO	34
AR	03	BB	1C	CP	35
AS	04	BC	1D	CR	36
AT	05	BD	1E	CS	37
AV	06	BE	1F	CT	38
AY	07	BG	20	CU	39
AZ	08	BH	21	CV	3A
BF	09	CC	22	CX	3B
BL	0A	CE	23	CY	3C
BP	0B	CG	24	AA	3D
BQ	0C	AC	25	AB	3E
BS	0D	AD	26	AH	3F
BT	0E	AJ	27	AI	40
BX	0F	AN	28	AK	41
BY	10	AQ	29	AO	42
CB	11	AX	2A	BI	43
CD	12	BJ	2B	CZ	44
CE	13	BK	2C	BN	45
CG	14	BU	2D	BO	46
CL	15	BV	2E	CF	47
DB	16	BZ	2F	CK	48
AL	17	CA	30	CN	49
AP	18	CH	31		

AA	3D	BB	1C	CD	12
AB	3E	BC	1D	CE	13
AC	25	BD	1E	CE	23
AD	26	BE	1F	CF	47
AF	00	BF	09	CG	24
AG	01	BG	20	CG	14
AH	3F	BH	21	CH	31
AI	40	BI	43	CI	32
AJ	27	BJ	2B	CJ	33
AK	41	BK	2C	CK	48
AL	17	BL	0A	CL	15
AM	02	BN	45	CN	49
AN	28	BO	46	CO	34
AO	42	BP	0B	CP	35
AP	18	BQ	0C	CR	36
AQ	29	BS	0D	CS	37
AR	03	BT	0E	CT	38
AS	04	BU	2D	CU	39
AT	05	BV	2E	CV	3A
AU	19	BX	0F	CX	3B
AV	06	BY	10	CY	3C
AX	2A	BZ	2F	CY	1A
AY	07	CA	30	CZ	44
AZ	08	CB	11	DB	16
BA	1B	CC	22		

TABLE 4.1 LEG BAND CODES

The automatic pressure control sequence has been shown in figure 3.8. The sequence terminates by generating an ABORT. At any point during the sequence the operator can terminate with an ABORT or RESET command. The operator can also change the automatic pressure sequence by using the 'D' command.

The 'C' command is used to select one of four control routines specified by the second digit of the command.

The 'C1' command activates a separate monitor system useful for system maintenance and problem isolation. The commands for this separate monitor allow the operator to enter programs into memory, read the contents of memory, and execute program beginning anywhere in memory. These commands are found in the comments of the software listing of Appendix 5. The 'C1' command will not be used in normal circumstances.

The second control command 'C2' causes the system to null the right leg band amplifier, then continuously display the output of the A/D converter. This permits testing and using the leg bands on the calibrated cylinders.

The 'CA' command is essentially the BEGIN command without pressure control.

The final control command, 'CC' (Continue), allows the automatic pressure sequence to be resumed after it has been interrupted.

The 'E' command selects one of two rates at which pressure level transitions are to be made: 'E1' selects 5 torr/sec, 'E2' selects 15 torr/sec.

The final command, 'F' selects the display format: 'F0' selects the leg band format; 'F1' selects the time format; 'F2' selects the pressure format.

4.2 CONSOLE DISPLAYS

The five digit console display uses 7-segment LED readouts to display numbers. Three display formats are available: leg band format; time format; and pressure format. These formats are diagramed in Appendix 1. The leg band format displays left and right leg band percent volume change. The time format shows the time at the current desired pressure and the desired pressure. The pressure format shows the number of minutes at the current desired pressure, the actual pressure, and the desired pressure.

5.0 SYSTEM RESULTS

The results of system pressure control and leg band data processing are found in Figures 5.1 through Figure 5.16.

Figure 5.1 shows data plotted for controlled LBNPD pressure versus leak rate. For these tests the LBNPD was sealed with the end plate containing a valve adjusted to a given leak rate. This configuration is shown in Figure 5.2. The plots produced by the equipment shown in Figure 5.4 are given for desired pressures of 10, 20, 30, 40, 50 torr. As the leak rate is increased, eventually the system cannot maintain pressure at the desired level. This is due to the characteristics of the vacuum source as seen in Figure 5.3, not due to the operation of the control system.

Figures 5.5 through 5.11 show pressure profiles with leak rates from 1 through 6 CFM. These plots were made with an operator manually entering a new pressure using the 'D' keyboard command every 20 seconds. The rate of pressure change was the default value of 15 torr/sec. Except when the vacuum source is no longer able to provide sufficient vacuum for the higher leak rates, the pressure is maintained with +-1.0 torr of the desired pressure, occasionally ranging to +-1.5 torr.

The results of the pressure rate control tests are shown in Figures 5.12 and 5.13. The rate of pressure change was selected by the operator with the 'E' command. The only choices are 5 torr/sec and 15 torr/sec.

The results of the leg band data acquisition and processing tests are shown in Figures 5.14 and 5.15. Data for 10 tests, the averages, and the standard deviations for two leg bands are shown. These tests were conducted using the calibrated cylinders which give a known percent volume change.

Figure 5.16 shows pressure and leg band data output for a subject in an experimental run.

6.0 SDSM&T DESIGN CONTRIBUTIONS

The Experiment Support System was designed and constructed at the Air Force Academy. After the project was transferred to SDSM&T the system was tested. Several problems became apparent which required design changes. The two major system functions are pressure control and acquisition and processing of leg band data. Both areas needed design changes.

6.1 PRESSURE CONTROL

The pressure control system was completely redesigned and simplified dramatically both in hardware and software. The original design used five valves: four for pressure control; and one failsafe abort valve. The new design uses two valves: one for pressure control and one for the failsafe pressure abort. This drastically simplifies the design with three fewer valves and their drivers, simplified valve mainfolds and elimination of the calibrated orifices. In addition, the power supply for the valve drivers could be replaced by a smaller, cheaper power supply. Physically the valve system could be much smaller with these reductions.

The pressure control system software was also simplified. Both rate control and level control were simplified with only one pressure control valve to manipulate. Due to these and other changes the new design requires one less ROM to hold the program. With these changes in the pressure control algorithm the resolution of the pressure output to the chart recorder was increased by a factor of four.

As part of these changes the rate interrupts are processed was doubled. This doubled the resolution on the time axis of the data plotted on the chart recorder. Data are acquired and processed at twice the original rate.

The ABORT routine was changed to maintain leg band data processing and chart recorder outputs even though the vacuum is aborted. The original system stopped everything during an abort.

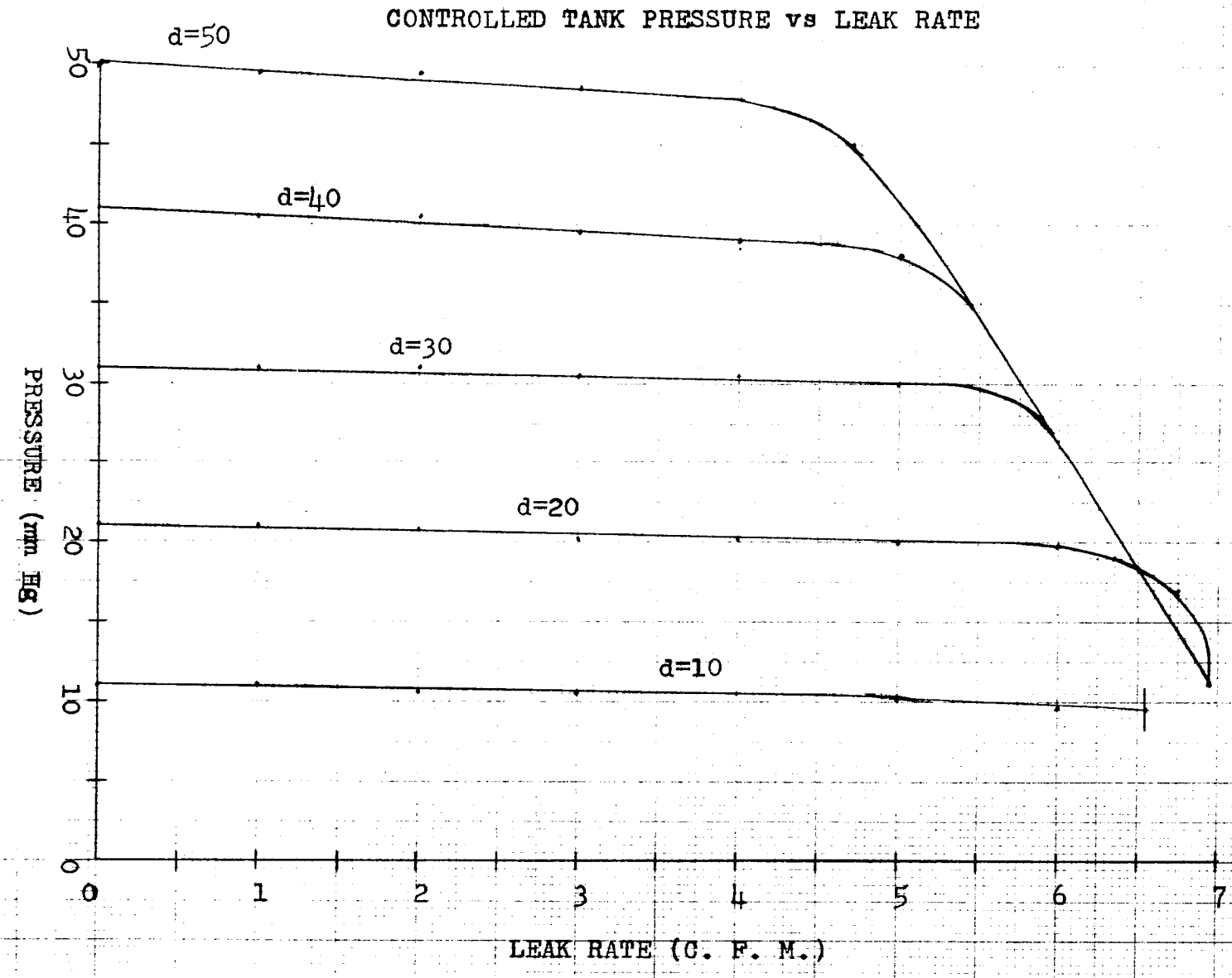


FIGURE 5.1 CONTROLLED LBNPD PRESSURE VS. LEAK RATE

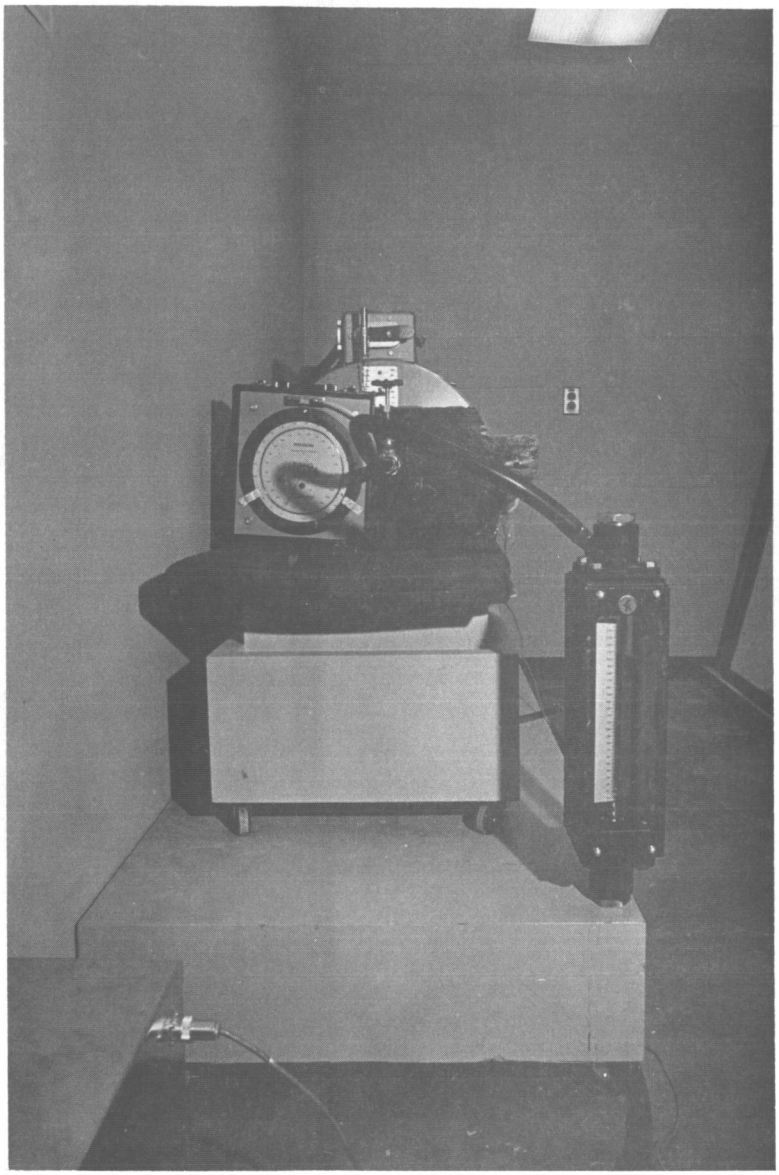


Figure 5.2 Leak rate flow meter and pressure meter set up to generate data plotted in Figures 5.1 and 5.3.

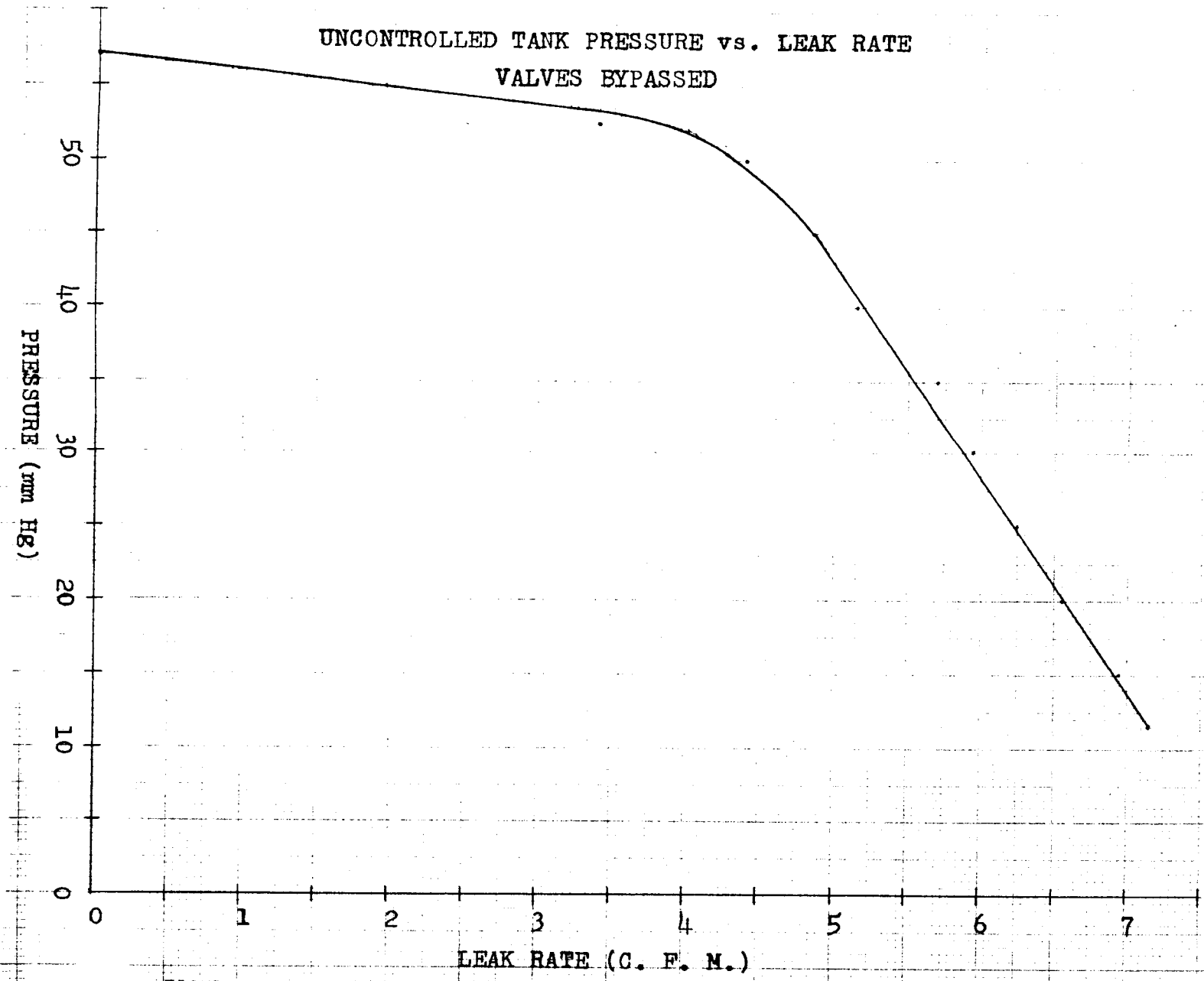


FIGURE 5.3 UNCONTROLLED TANK PRESSURE VS. LEAK RATE WITH VALVES BYPASSED

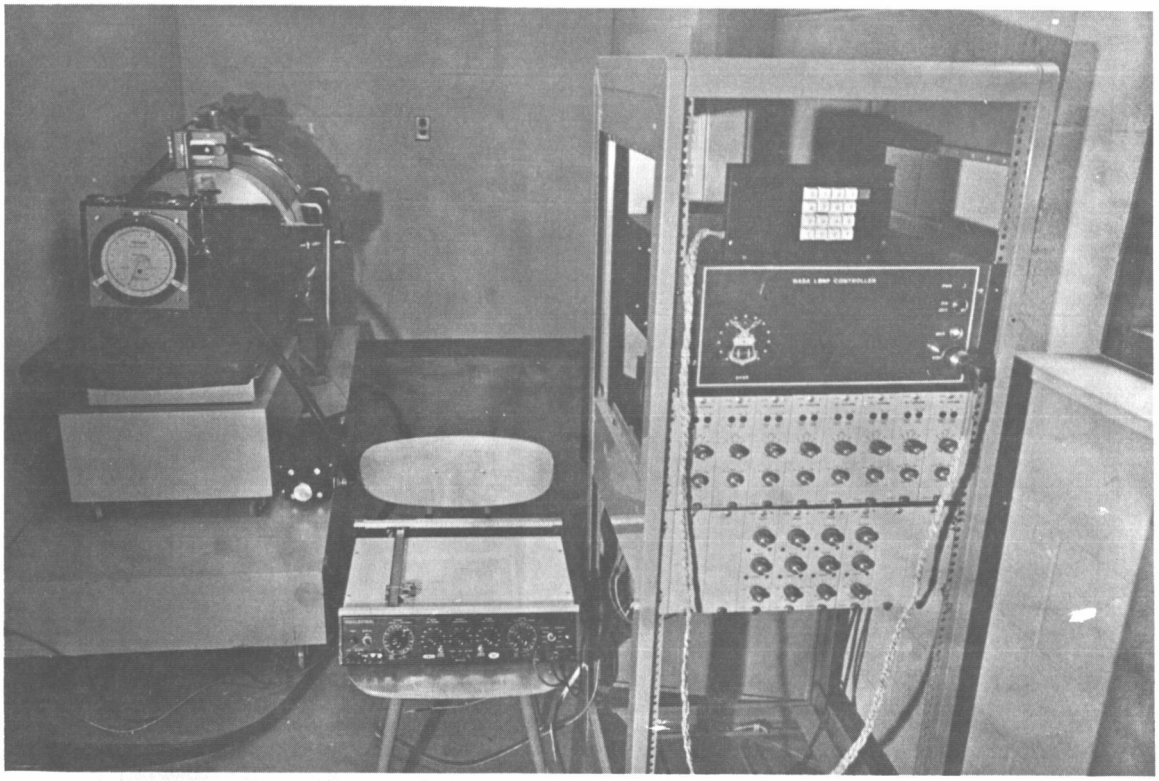


Figure 5.4 Plotter set up to produce Figures 5.5 through 5.13.

20 TO THE INCH 46 1242
MADE IN U.S.A.
KEUFFEL & ESSER CO.

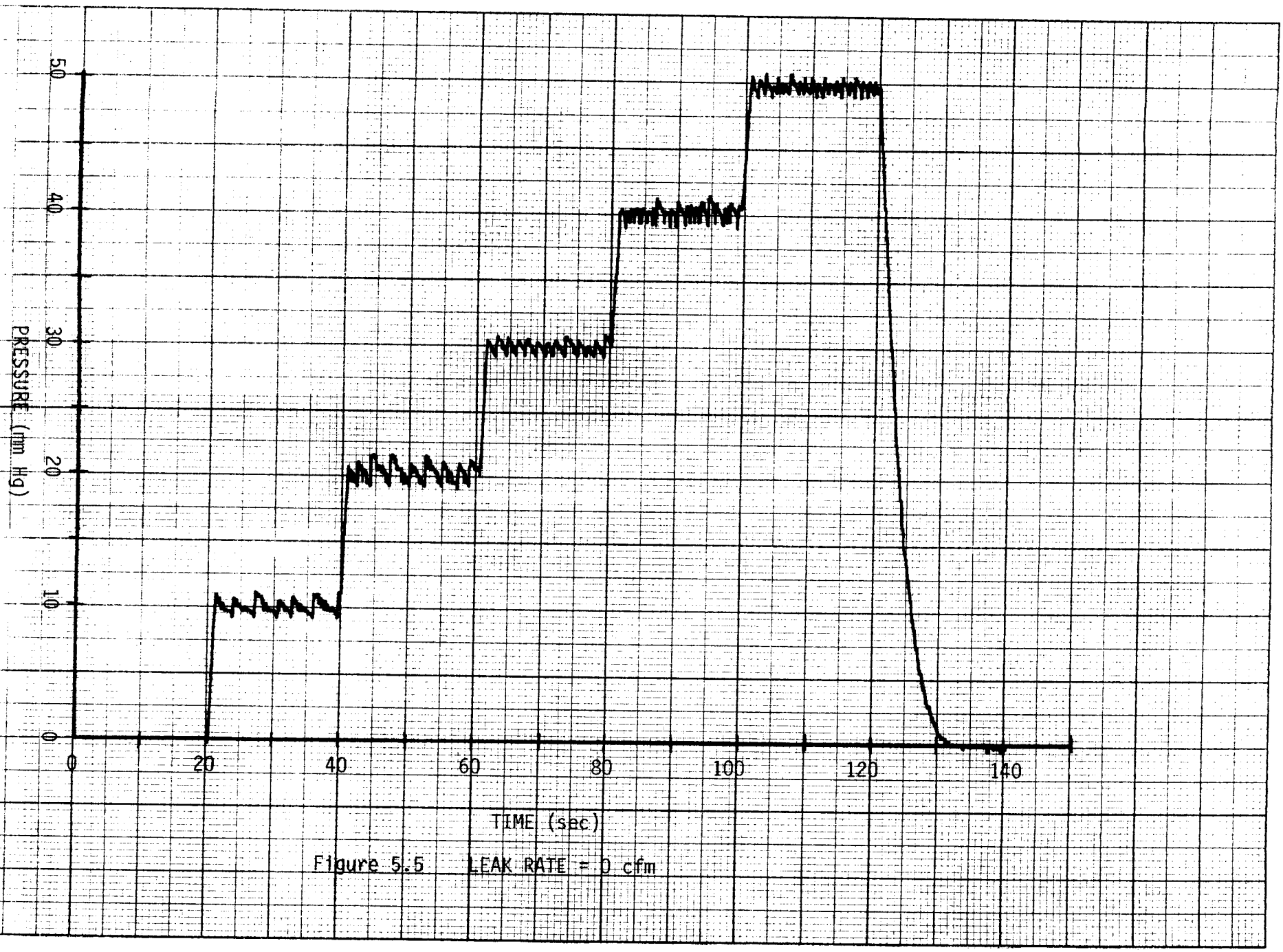


Figure 5.5 LEAK RATE = 0 cfm

K+E 20 X 20 TO THE INCH 46 1242
7 X 10 INCHES MADE IN U.S.A.
KEUFFEL & ESSER CO.

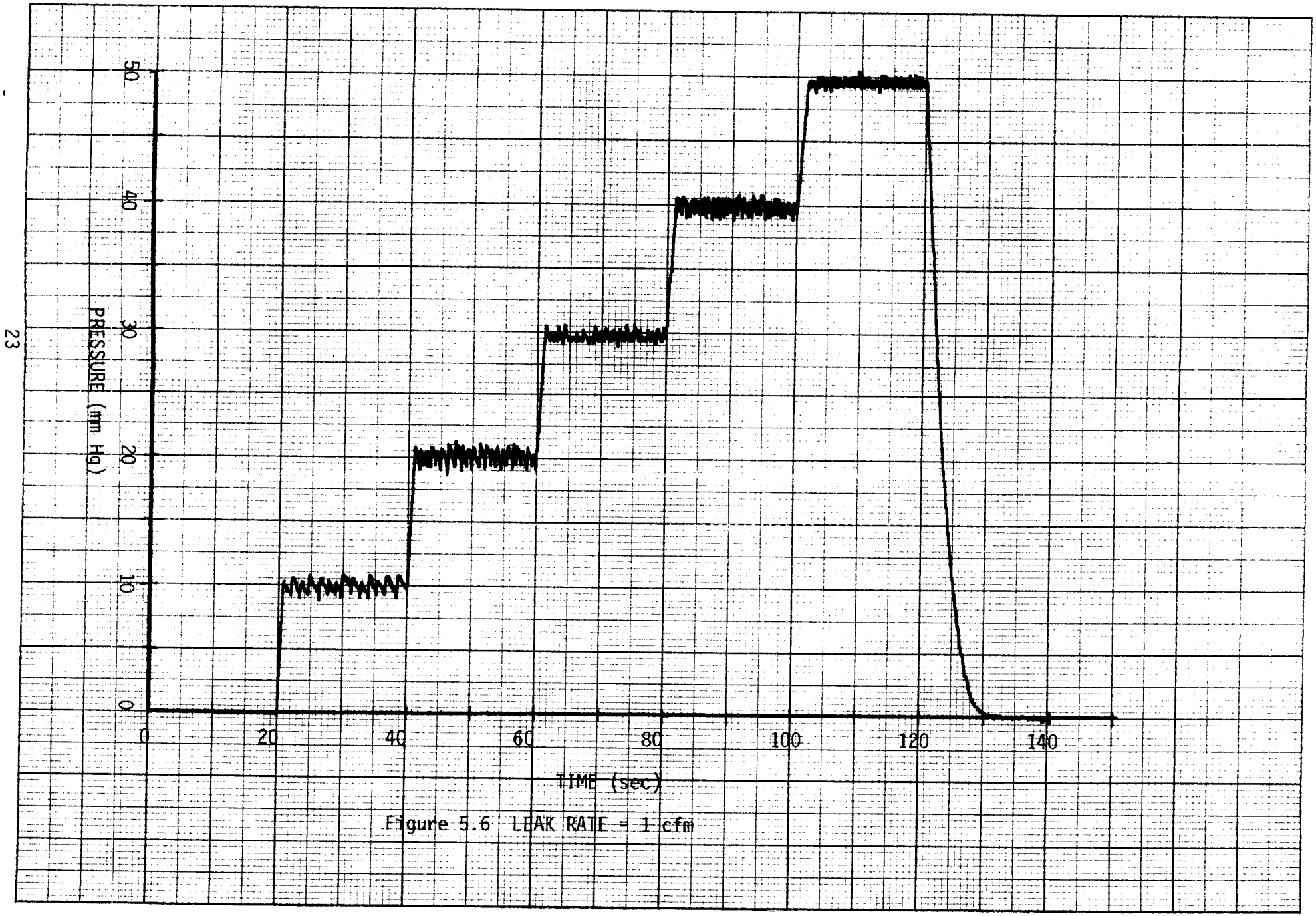


Figure 5.6 LEAK RATE = 1 cfm

K+E 20 X 20 TO THE INCH 46 1242
KEUFFEL & ESSER CO.

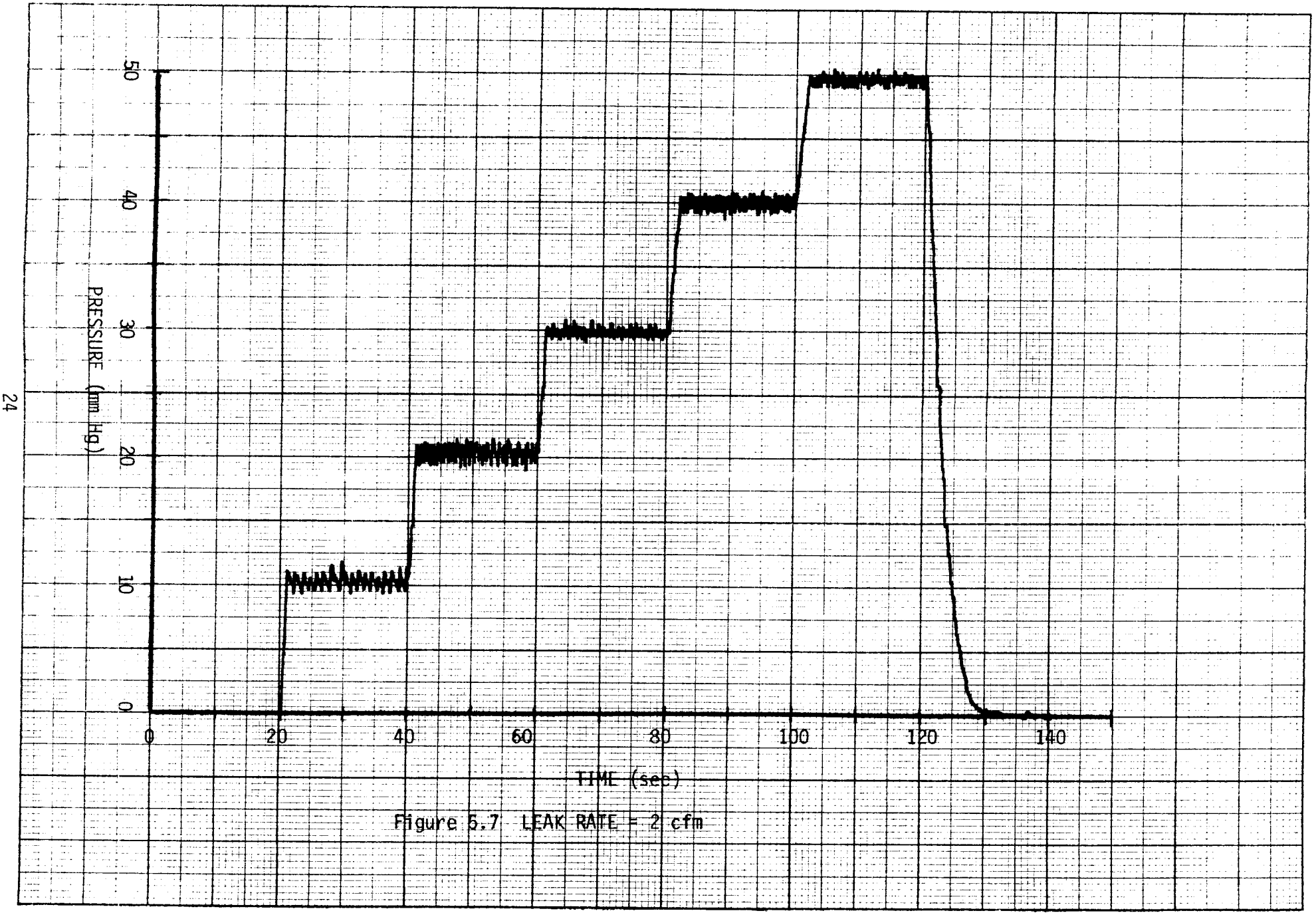


Figure 5.7 LEAK RATE = 2 cfm

K+E 20 X 20 TO THE INCH 46 1242
7 X 10 INCHES MADE IN U.S.A.
KEUFFEL & ESSER CO.

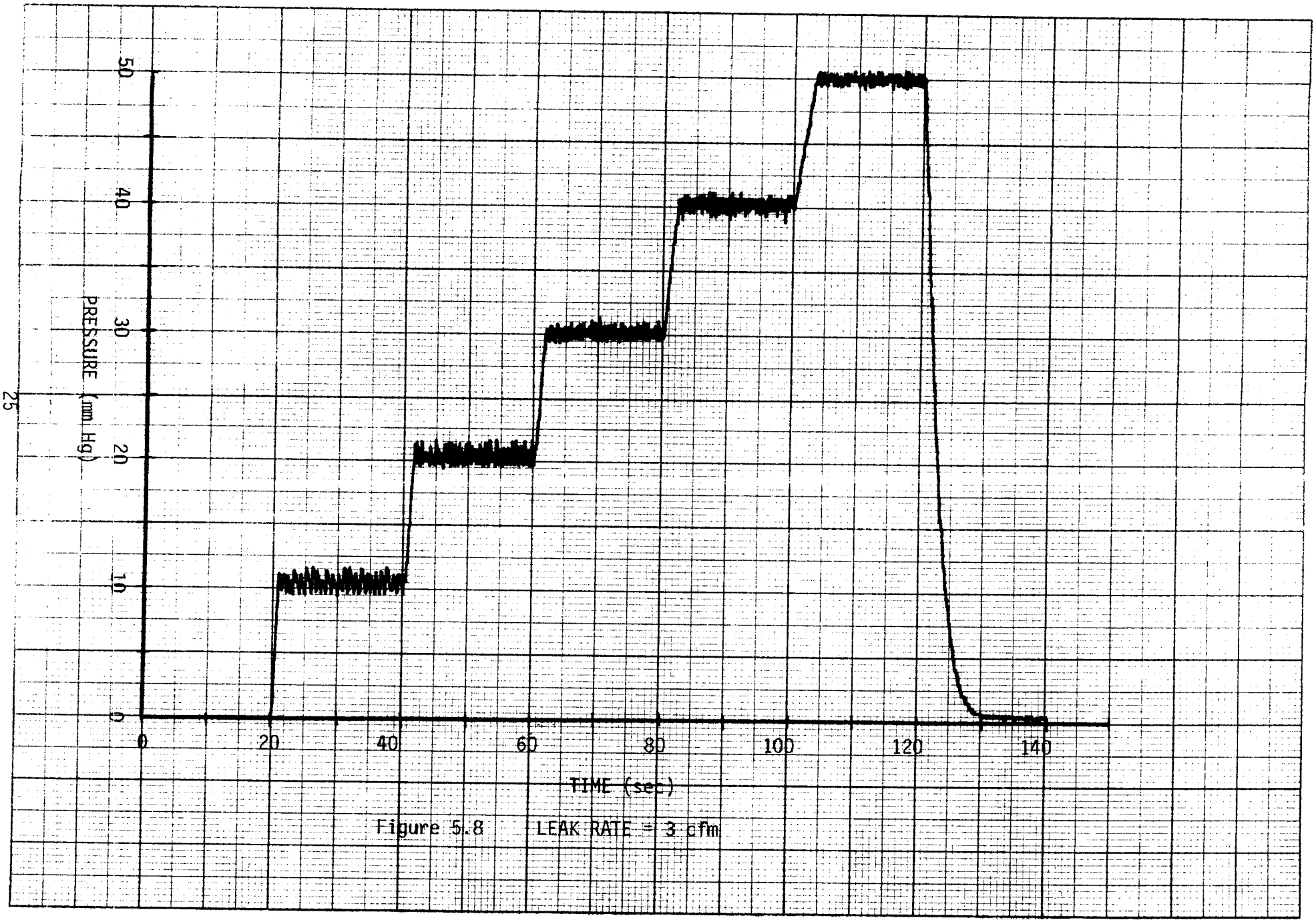


Figure 5.8 LEAK RATE = 3 cfm

K+E 20 X 20 TO THE INCH 46 1242
7 X 10 INCHES MADE IN U.S.A.
KEUFFEL & ESSER CO.

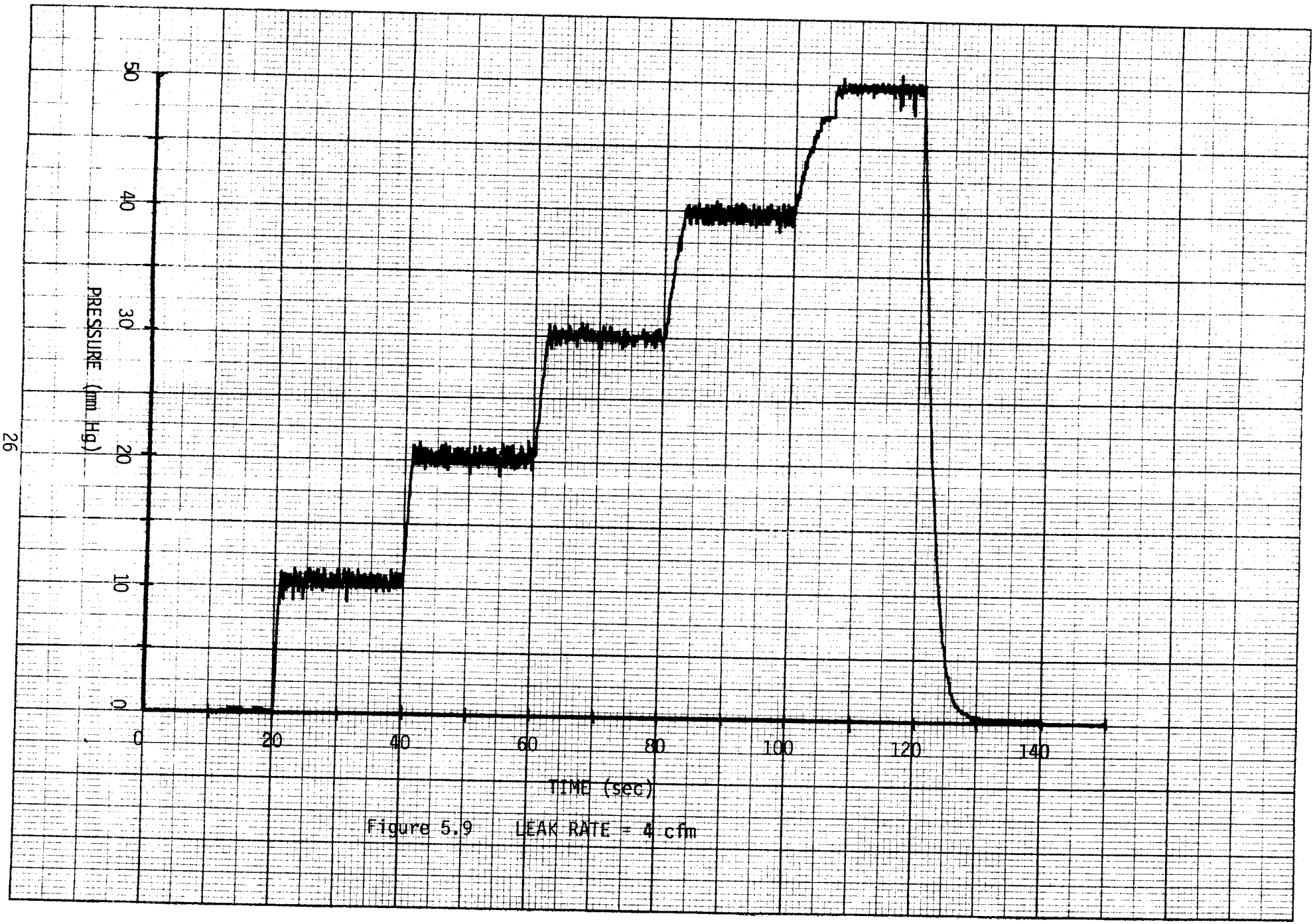


Figure 5.9 LEAK RATE = 4 cfm

K+E 20 X 20 TO THE INCH 46 1242
7 X 10 INCHES MADE IN U.S.A.
KEUFFEL & ESSER CO.

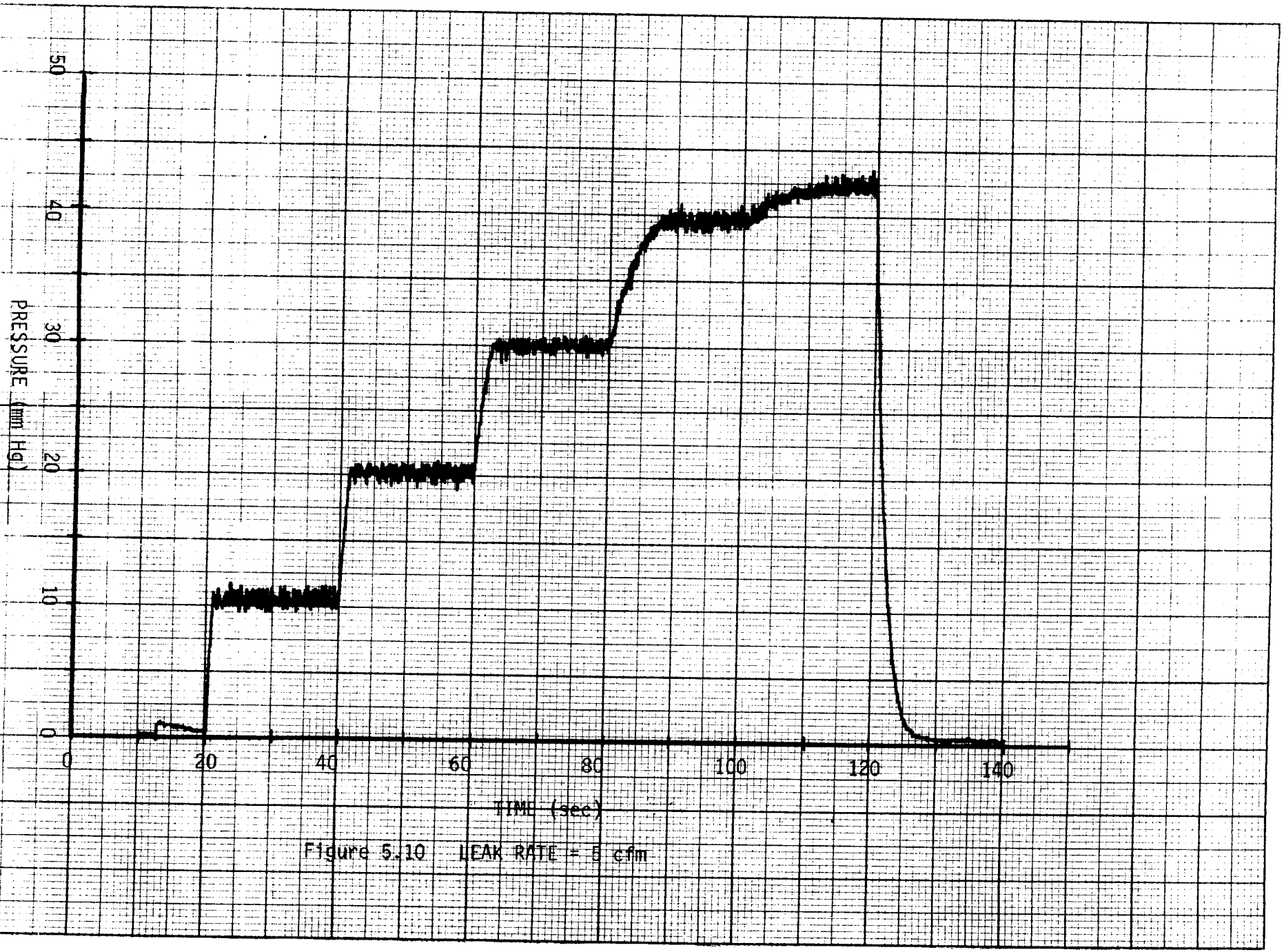


Figure 5.10 LEAK RATE = 5 cfm

K+E 20 X 20 TO THE INCH 46 1242
7 X 10 INCHES
MADE IN U.S.A.
KEUFFEL & ESSER CO.

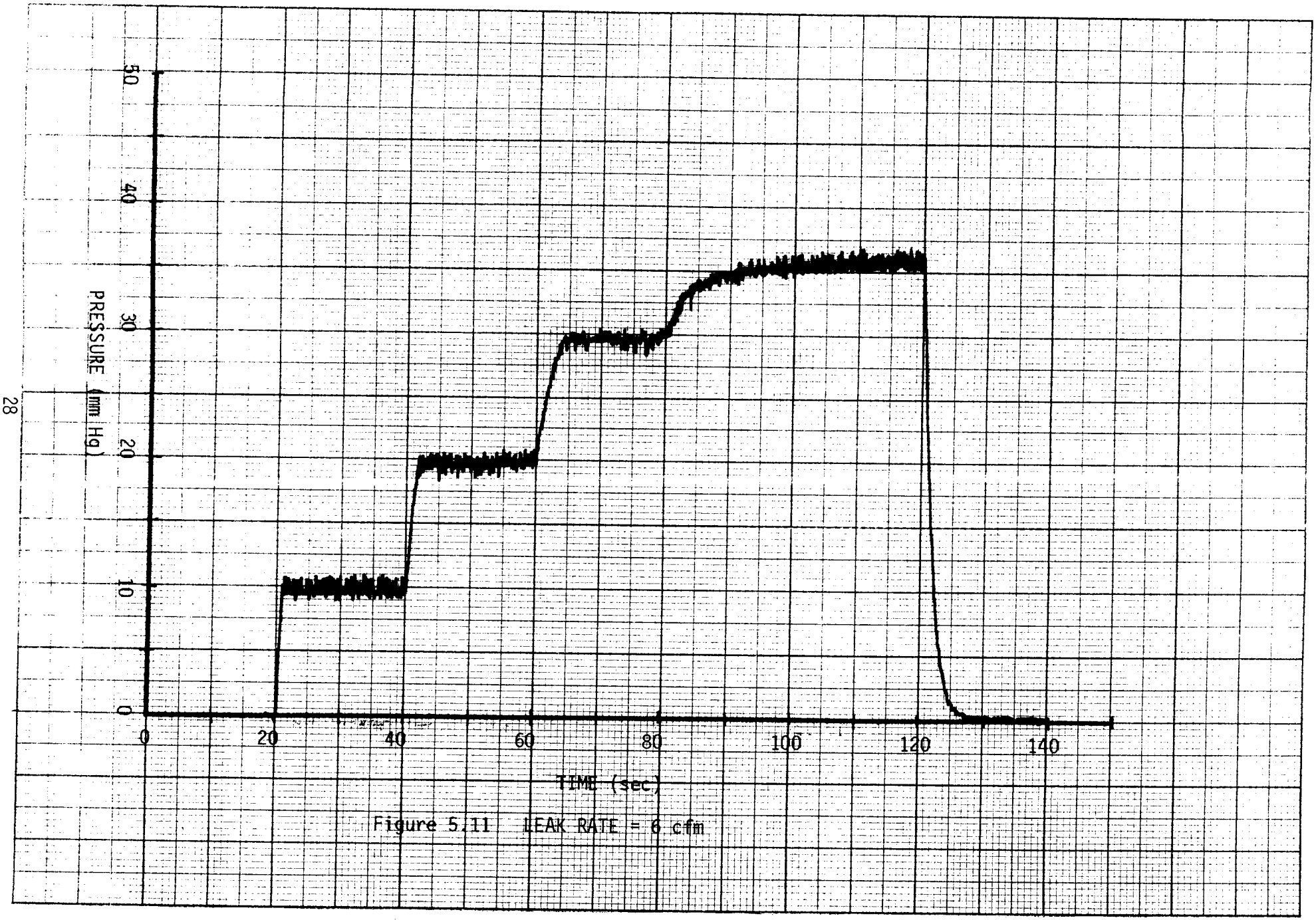


Figure 5.11 LEAK RATE = 6 cfm

THE INCH 46 1242
ESSER CO.

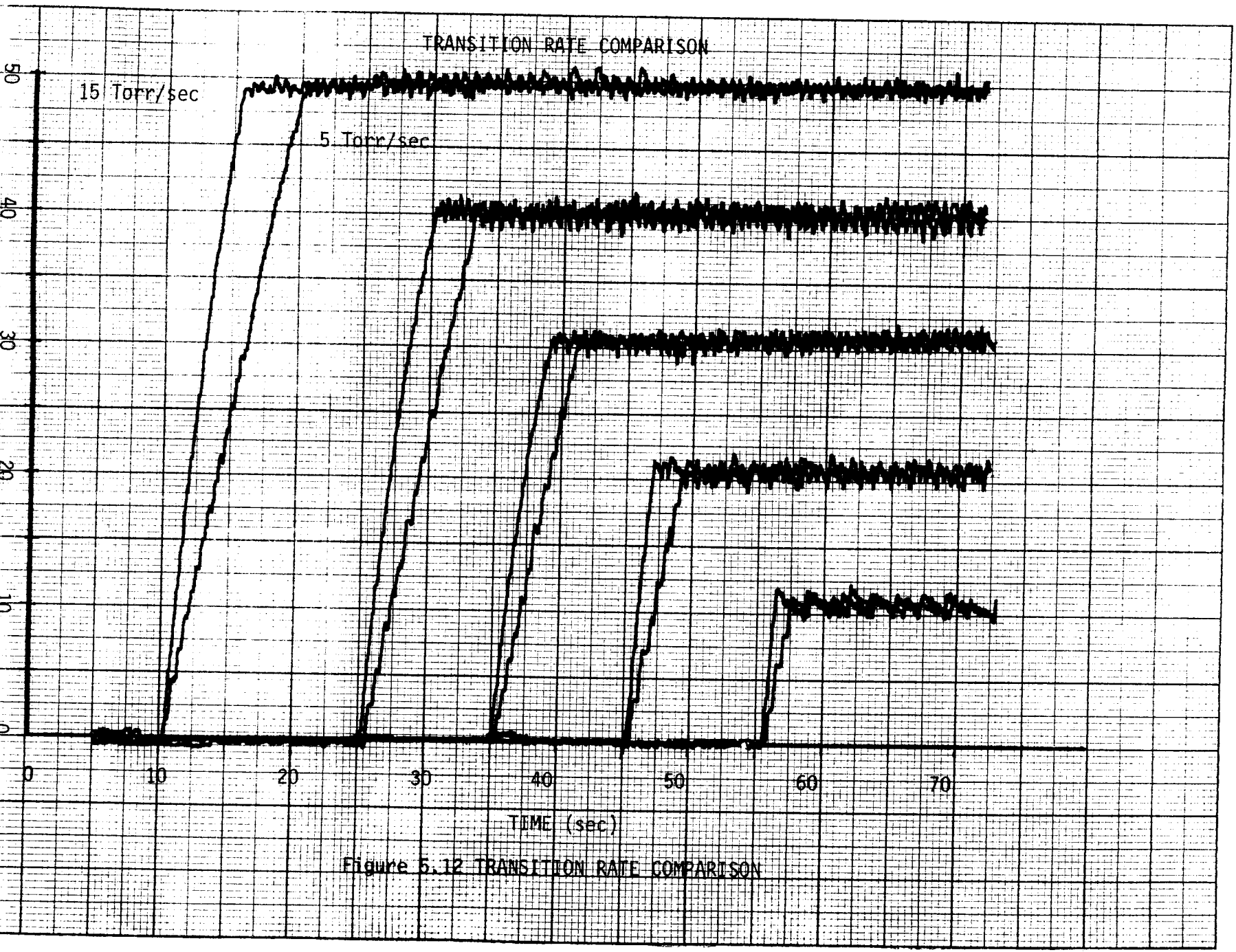


Figure 5.12 TRANSITION RATE COMPARISON

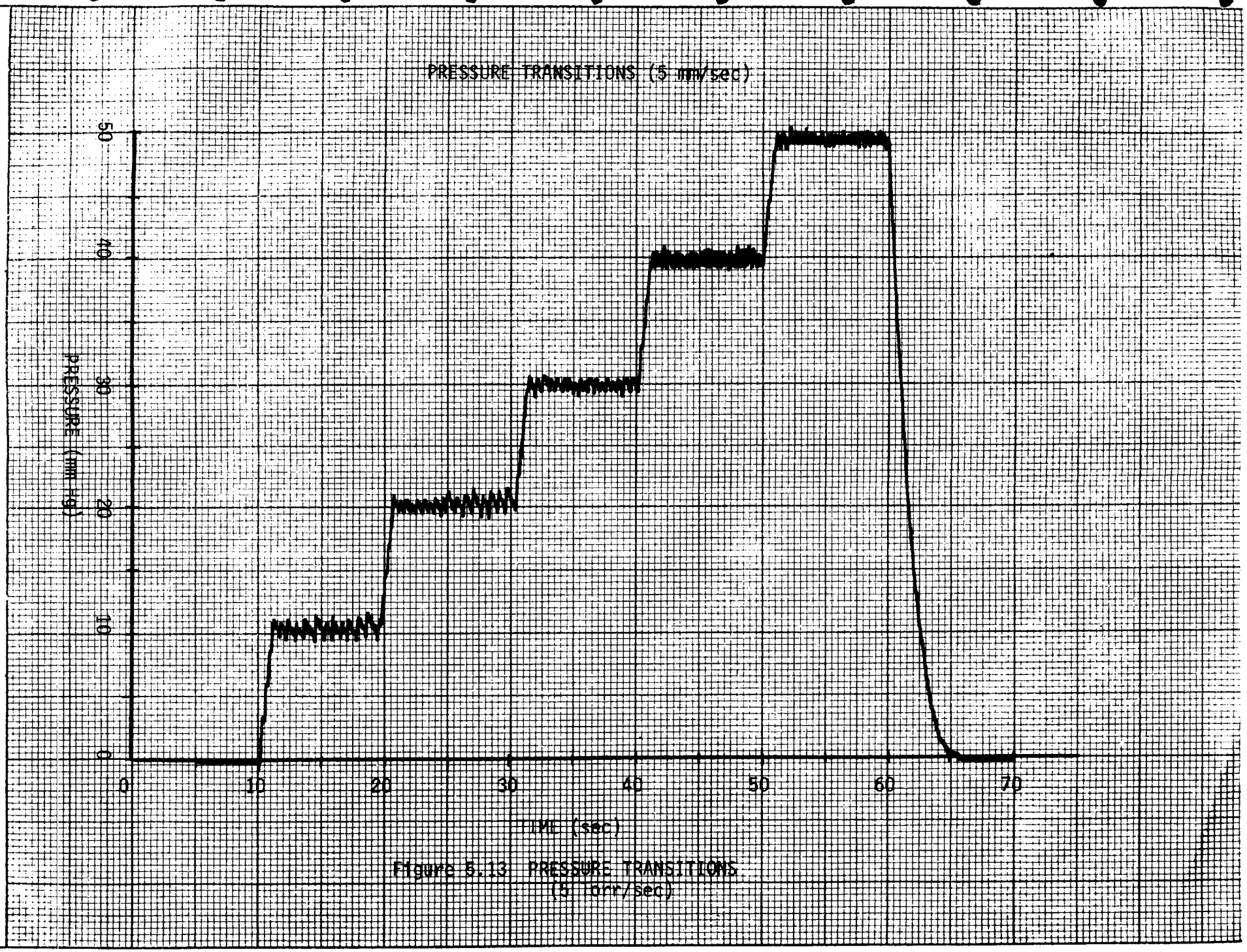


Figure 5.13 PRESSURE TRANSITIONS
(5 torr/sec)

	Trial	-1% Ind.%	0%	1%	2%	3%	4%	5%
Right	1	-1.0	.1	1.1	2.0	3.1	4.1	5.0
	2	-1.0	0.0	0.9	2.0	2.9	3.8	4.8
	3	-1.0	0.1	1.1	2.0	3.0	4.0	5.0
	4	-1.0	-0.1	1.0	2.2	3.1	4.2	5.0
	5	-1.0	0.0	1.1	2.1	3.1	4.1	5.0
	Mean	-1.0	.02	1.04	2.06	3.04	4.04	4.96
	S. Dev.	0.0	.08	.09	.09	.09	.15	.09
Left	1	-1.0	0.0	1.0	2.1	3.1	3.9	5.0
	2	-1.0	0.0	.9	2.1	3.1	4.1	5.0
	3	-1.0	0.0	1.0	2.1	3.1	4.1	5.0
	4	-1.0	.1	1.1	2.1	3.1	4.2	5.0
	5	-1.0	0.0	.9	2.0	3.0	4.1	5.0
	Mean	-1.0	.02	.98	2.08	3.08	4.08	5.0
	S. Dev.	0.0	.04	.08	.04	.04	.11	0.0
Indicated % Mean all 10 trials		-1.0	.02	1.01	2.07	3.06	4.06	4.98
	S. Dev.	0.0	.06	.09	.07	.07	.13	.06

Figure 5.14 Leg Band BI Results on Calibrated Cylinder

	Trial	-1%	0%	1%	2%	3%	4%	5%
Right	1	-1	-.1	1.0	2.1	3.1	4.0	5.0
	2	-1	.1	1.2	2.2	3.3	4.2	5.0
	3	-1	0.0	1.0	2.0	3.2	4.2	5.0
	4	-1	-.1	.9	1.9	3.1	4.0	4.9
	5	-1	0.0	1.0	2.0	3.0	4.0	5.0
	Mean	-1	-.02	1.02	2.04	3.14	4.08	4.98
	S. Dev.	.00	.08	.11	.11	.11	.11	.04
Left	1	-.8	.2	1.0	2.2	3.0	4.2	5.0
	2	-.8	0.0	1.1	2.2	3.1	4.2	5.0
	3	-.9	-.1	.9	1.8	3.0	3.9	4.8
	4	-1	0.0	.8	1.9	3.0	4.1	5.0
	5	-.9	0.0	1.1	2.1	3.0	4.1	5.0
	Mean	-.88	.02	.98	2.04	3.02	4.1	4.96
	S. Dev.	.08	.11	.13	.18	.04	.12	.09
Indicated % Mean all 10 trials		-.94	0.0	1.00	2.04	3.08	4.09	4.97
	S. Dev.	.08	.09	.12	.14	.10	.11	.07

Figure 5.15 Leg Band AI Results on Calibrated Cylinder

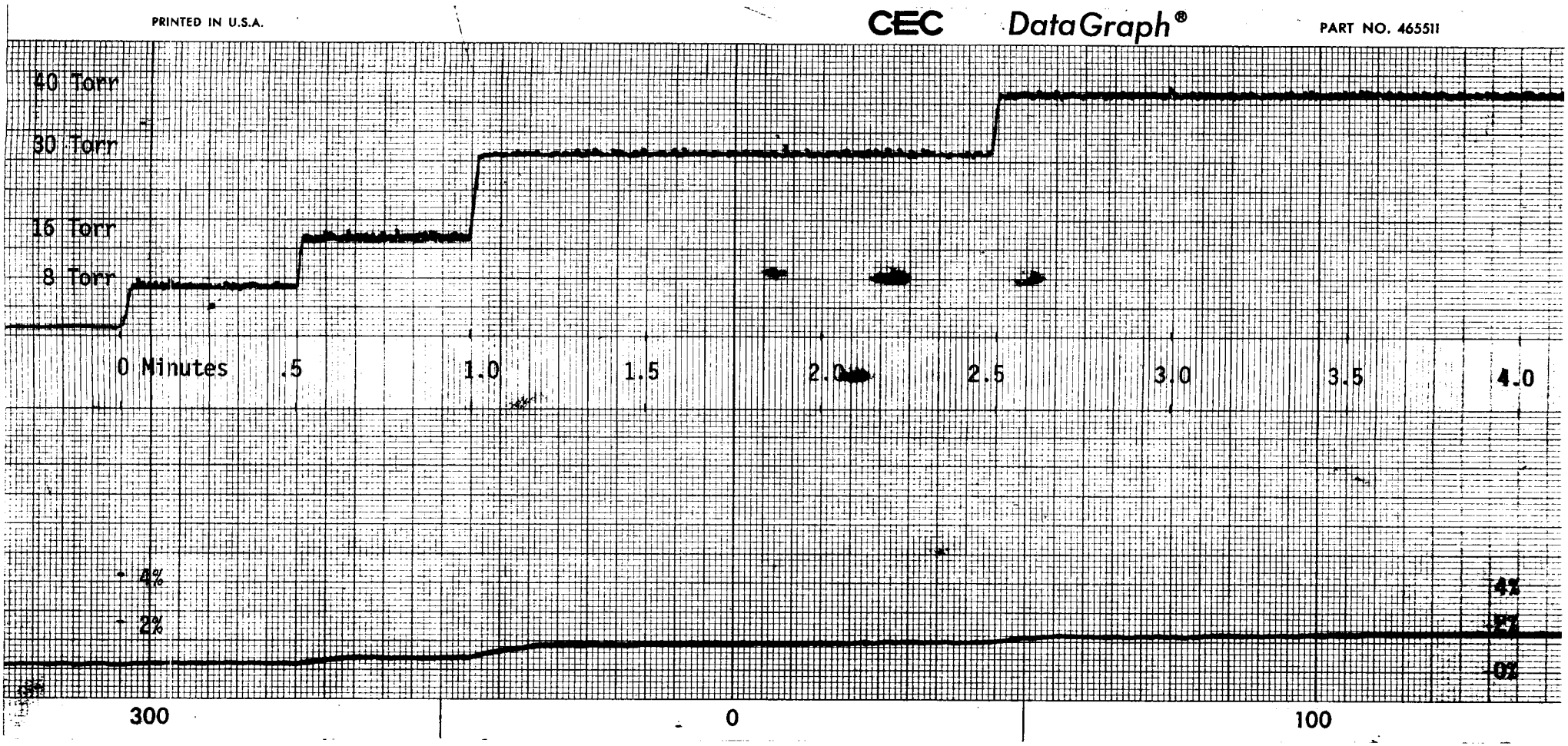


Figure 5.16 Test of LBNPD Pressure Control and LVMS Leg Band data processing. Pressure in Torr and Percent Volume Change vs time in Minutes.

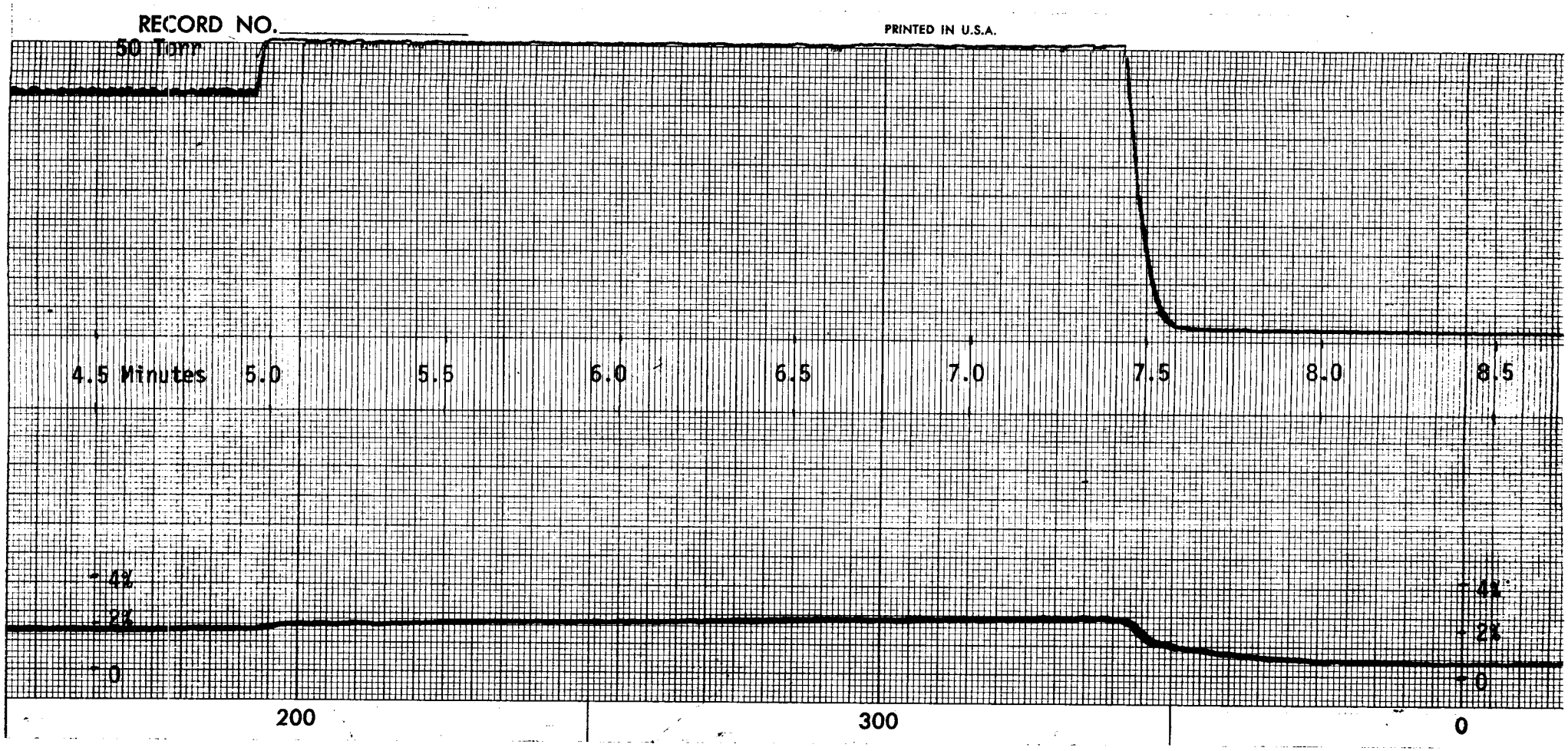


Figure 5.16 (Con't) Test of LBNPD Pressure Control and LVMS Leg Band data processing. Pressure in Torr and Percent Volume Change vs Time in Minutes

Figure 6.1 shows that the substitue pressure transducer which is necessary for the LBNPD transducer was not functioning when the system arrived at SDSM&T.

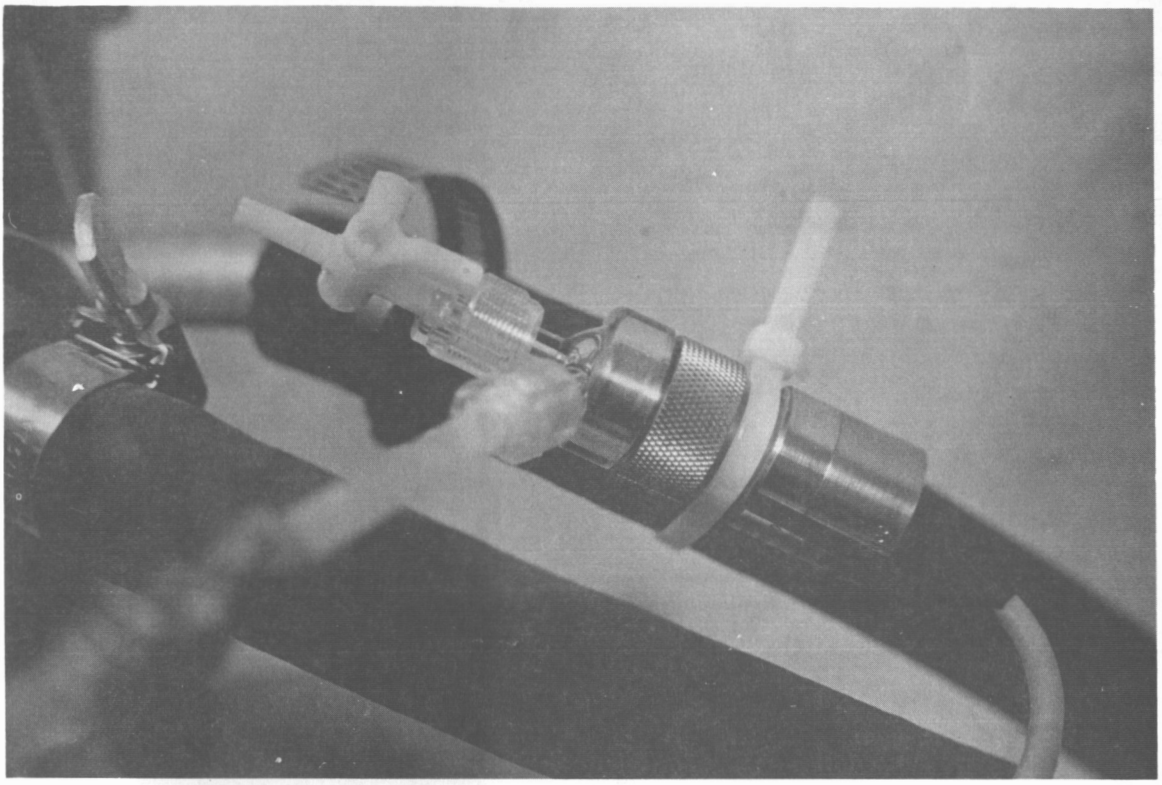


Figure 6.1 The substitue pressure transducer.

6.2 LEG BAND DATA ACQUISITION AND PROCESSING

The design changes in this case did not result in any simplification but were necessary for the system to function.

For an undetermined reason there was an incompatibility between the leg band card output (0-5 volts) and the A/D converter (0-10 volts). A gain of two amplifiers was added to each channel to resolve this problems. These can be seen in Figure 6.2; the schematic diagram is in Appendix 2, Figure 8.

The next modification was in the filtering of the power supply on the oscillator-amplifier card and the right and left channel leg band controller cards. The inductors intended for filtering were removed becuase they caused supply voltage changes with the load. The inductors removed had a resistance of 80 ohms which caused a change in voltage as the current changed. Although this change in voltage was only 10 mv, it had a large effect on the leg band card output since the output gain is high. The gain of the oscillator-amplifier card was changed from 100 to 50, to allow the data to be processed more accurately.

The rest of the changes were software modifications. The first software change was correcting an error in the COMP routine which caused the left GAIN to be used for processing right leg band data.

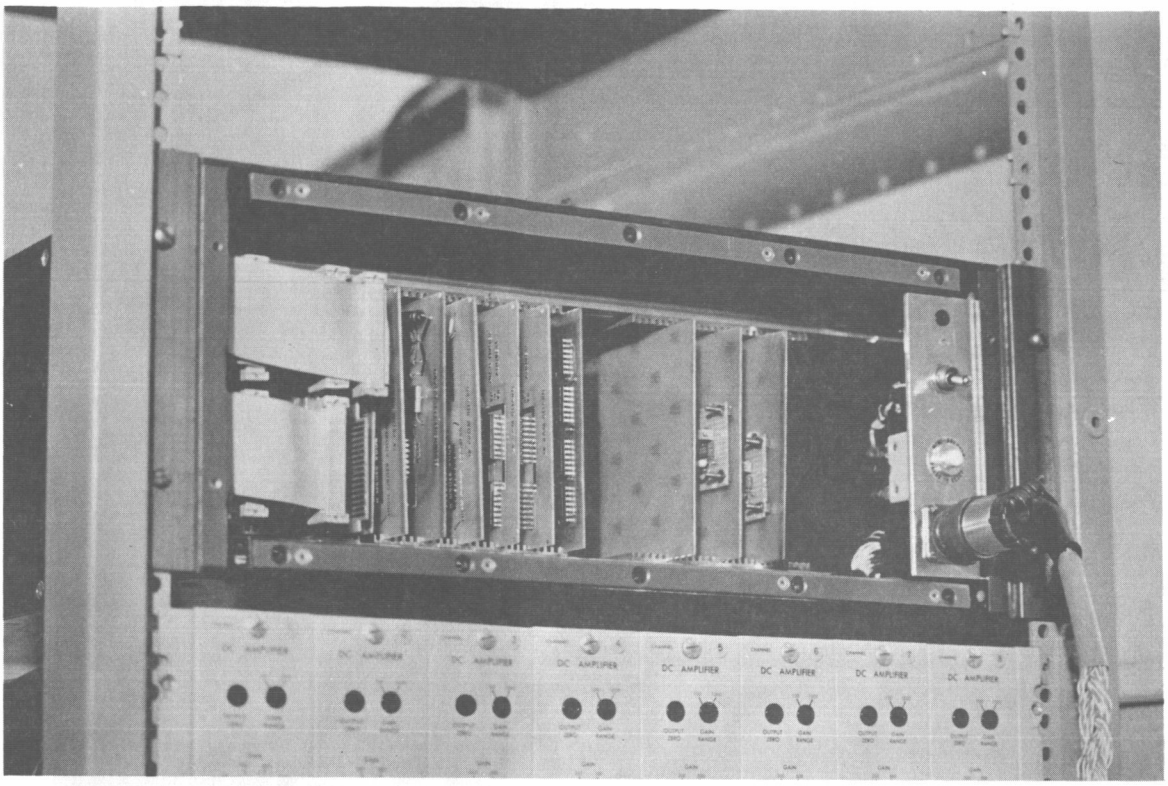


Figure 6.2 The gain of two amplifiers (shown mounted on spacers on the leg band controller cards).

The second software change was in the GAINR and GAINL rountine. These routines close the calibrate relays, read the voltages, and compute GAIN1 and GAIN2 to be used in processing leg band data. A low calibrate voltage was read because the 0.25 second delay between closing the calibrate relay and reading the signal was inadequate. This was increased to consistently give good calibrate readings.

6.3 SDSM&T SUPPORT FACILITY

6.3.1 LABORATORY

To support this project a lab was constructed with special partitions for the system components. This provided a good working environment, well organized and free of interference.

6.3.2 SOFTWARE SUPPORT

To aid the extensive software development several aids were developed. A cross-assembler running on the CDC-6400 was adapted to generate code for the INTEL 8080 microprocesser in the ESS. A ROM programmer for the 1702 ROM used in the ESS was adapted for use in this project. To increase the efficiency of generating new ROM'S during development, the ROM programmer was interfaced to a time-sharing port on the CDC-6400. This allowed changes to be made in the ESS software, and a new ROM programmed in just a few minutes. The assembler output comes directly from the CDC-6400 to the ROM programmer. This avoids frustrating and time consuming manual loading of code to be programmed.

APPENDIX 1

OPERATOR COMMANDS

COMMAND	RESULTS
Power On	Generates a computer RESET (See "R" below)
"R" (RESET)	Generates a computer RESET a) Closes and deactivates all valves b) Resets and deactivates all timers c) Resets display to all zeros d) Resets temporary storage locations
"A" (ABORT)	Generates a program ABORT a) Closes four control valves and opens abort valve b) Sets display to "dEAdd" c) Halts computer: operator must RESET to reactivate
"B" (BEGIN)	BEGINS automatic control sequence a) Displays "BA100" and waits for two-digit leg band ID entry (echos entry) b) Displays "BA200" and waits for two-digit leg band ID entry (echos entry) c) Nulls both analog amplifiers if leg band ID's were valid, otherwise requests ID's again d) Calculates software gain factor for each leg band by activating leg band cal relays e) BEGINs controlling chamber pressure according to the stored pressure profile: 1) The automatic pressure profile can be interrupted by entering a specific desired pressure (see "D" below), then resumed by commanding "CC" (CONTINUE). (The interrupted phase of the automatic sequence will be completed using the new desired pressure, but following phases will be completed at the normal auto pressure levels.) 2) The automatic sequence will terminate by generating a program ABORT. 3) The automatic sequence can be manually terminated by RESET or ABORT (see above).
"Cx" (CONTROL x)	Allows user to select individual CONTROL routines a) "C0" has no effect b) "C1" begins execution of the normal EDUMICRO operating system. (This is useful in debugging the system, but of no use in normal NASA operation.)

	c) "C2" causes the system to null one leg band amplifier, then continuously display the output of the A/D converter. Using this command permits mounting a leg band on one of the calibrated cylinders, reading the leg band data for known "leg" volumes, then entering that information into the leg band calibration tables (stored as part of the program). d) "CA" causes the system display "BA100" and "BA200" and wait for the appropriate leg-band ID's (see "BEGIN" above). 1) Nulls both analog amplifiers if leg band ID's were valid, otherwise requests ID's again. 2) Calculates software gain factor for each leg band by activating leg band cal relays. 3) Returns to scan keyboard without controlling pressure/valves. e) "CC" is a "CONTINUE" command which allows the automatic pressure sequence to be <u>resumed</u> after it has been interrupted (see "BEGIN" above). "CC" should not be used to <u>initiate</u> automatic operation because it does not call the necessary calibration routines.
"Dxx" (DESIRED PRESSURE)	Allows manual entry of new value of DESIRED PRESSURE a) The DESIRED PRESSURE must be a two-digit entry representing mm Hq. b) Can be used to interrupt the automatic sequence (must be followed by a "CONTINUE" command if the auto sequence is to be resumed; see BEGIN and "CC" above).
"Ex" (SELECT RATE)	SELECTs one of two RATES at which pressure level transitions are to be made. a) "E1" selects the low rate (5 mm Hq/sec) b) "E2" selects the high rate (15 mm Hq/sec)
"Fx" (DISPLAY FORMAT SELECT)	Selects one of three possible display FORMATs. a) "F0" selects "leg band" format b) "F1" selects "time" format c) "F2" selects "pressure" format

INVALID COMMANDS	Changes display to "6000F" a) Normal display operation is interrupted for approx. 2 sec. b) Pressure/valve control is maintained using the most recent valid value of "desired pressure".

DISPLAY FORMATS

Leg Band Format:

LEFT PERCENT (ONES)	LEG VOLUME (TENTHS)	SIGN INDICATOR A Both + B R- L+ D R+ L- E both -	RIGHT PERCENT (ONES)	LEG VOLUME (TENTHS)

Time Format:

TIME AT (MIN)	CURRENT DESIRED PRES. (SECONDS)	DESIRED PRESSURE (MM HG)

Pressure Format:

Time at Current Desired Pressure (minutes)	ACTUAL PRESSURE (MM HG)	DESIRED PRESSURE (MM HG)

APPENDIX II

HARDWARE SCHEMATICS

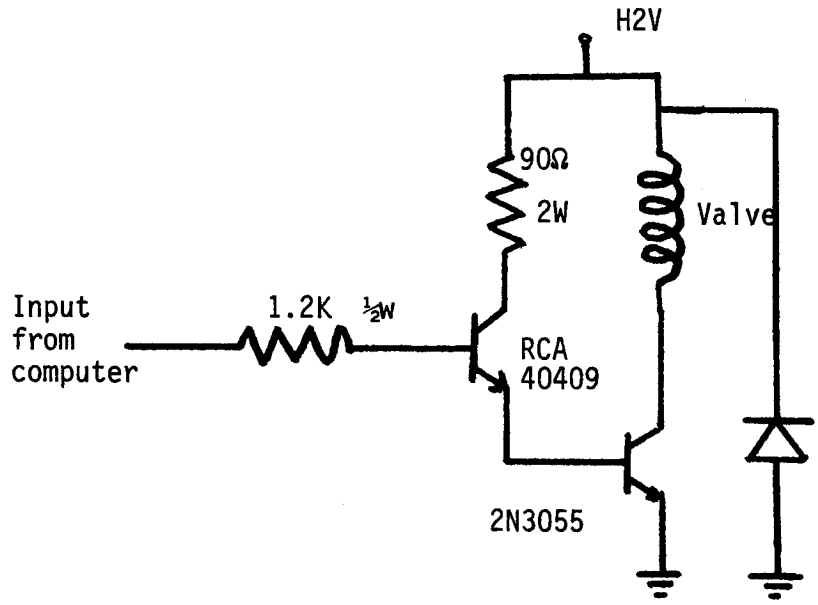


FIGURE A2.1 VALVE DRIVER SCHEMATIC

This is one of five identical circuits for the five valves. The valves are ATKOMATIC VALVE COMPANY MODEL 15448.

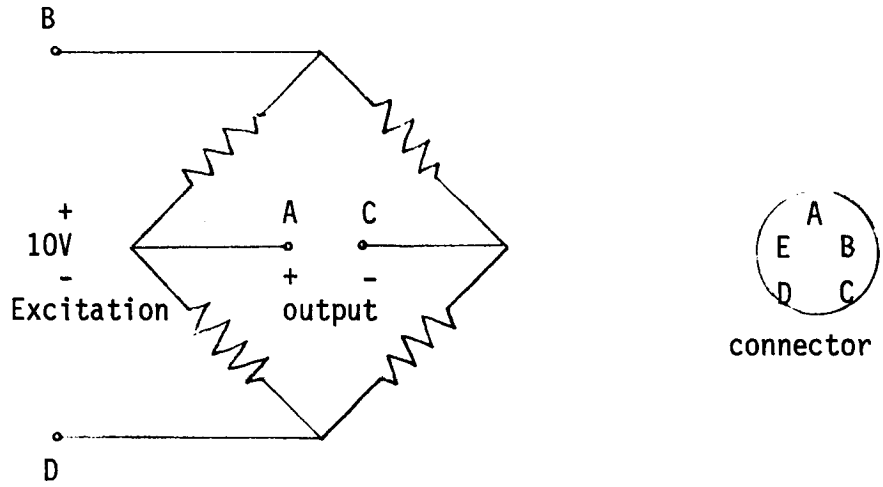


FIGURE A2.2 Substitute pressure transducer,

Stratham model P23De with characterstic

134.6 $\frac{\mu v}{v}$ /10 Torr

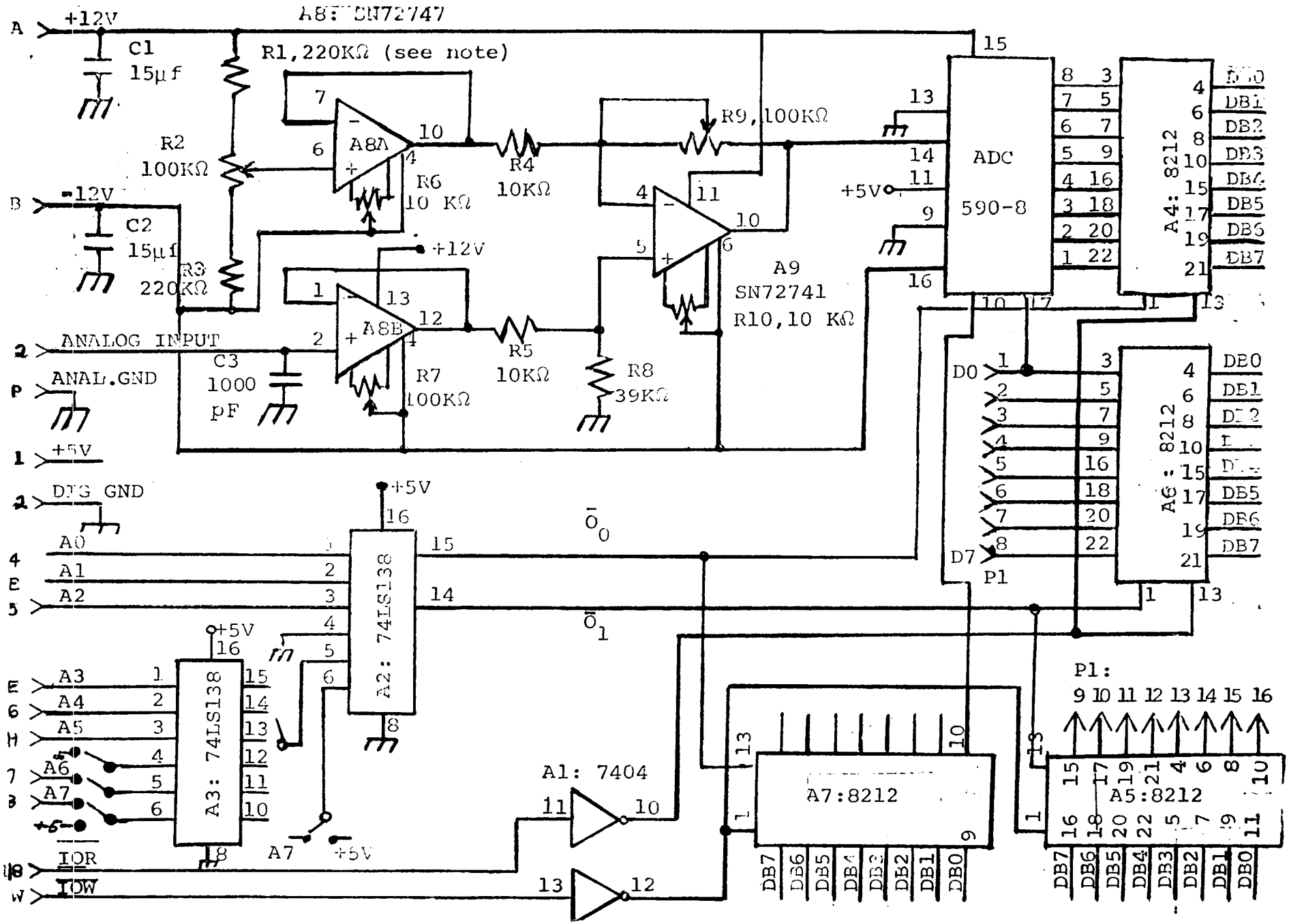
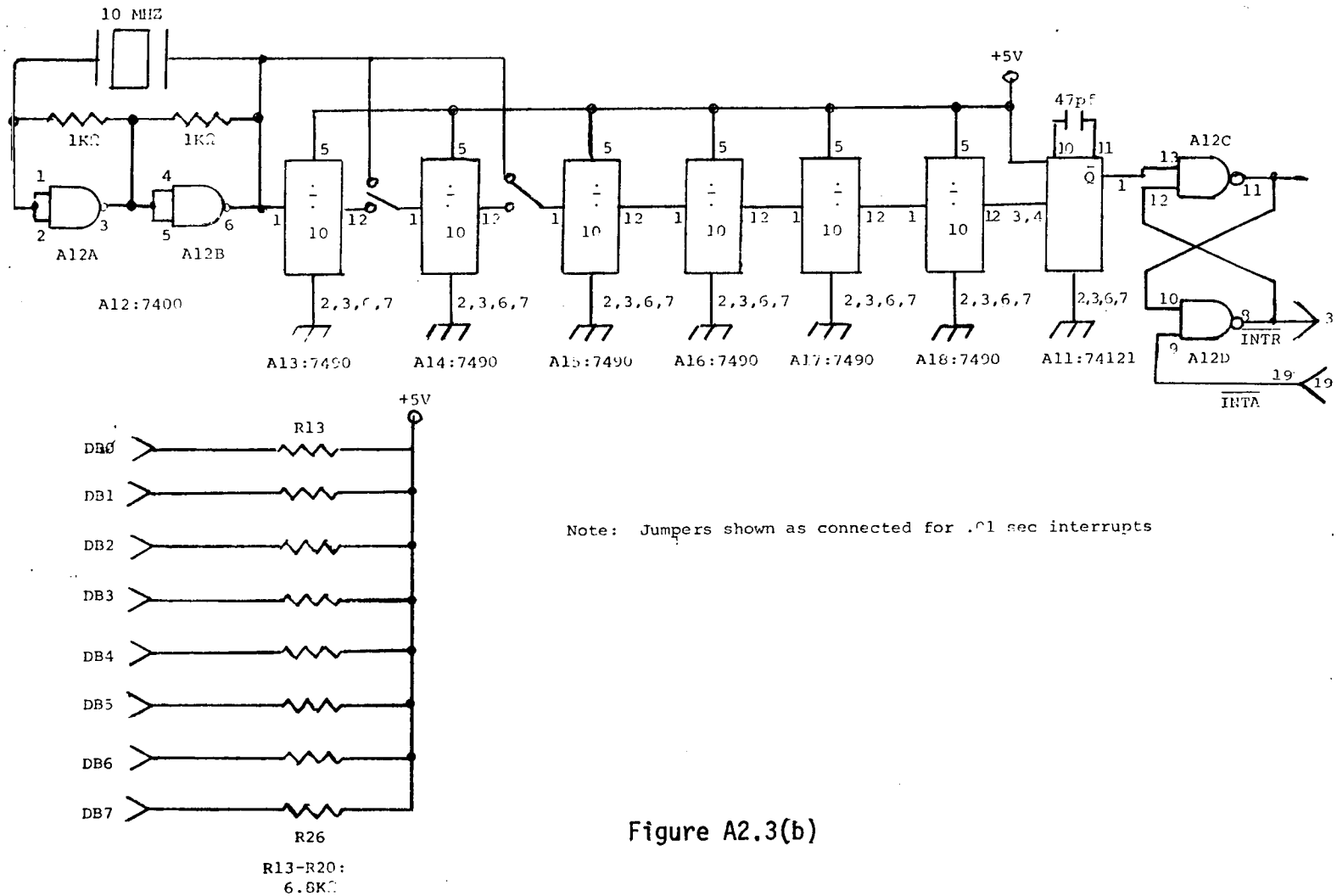


FIGURE A2.3(a) NASA LBNP CONTROLLER ANALOG/INTERRUPT BOARD (PRESSURE A/D CARD)

NASA LBNP CONTROLLER ANAL/INTERRUPT BOARD



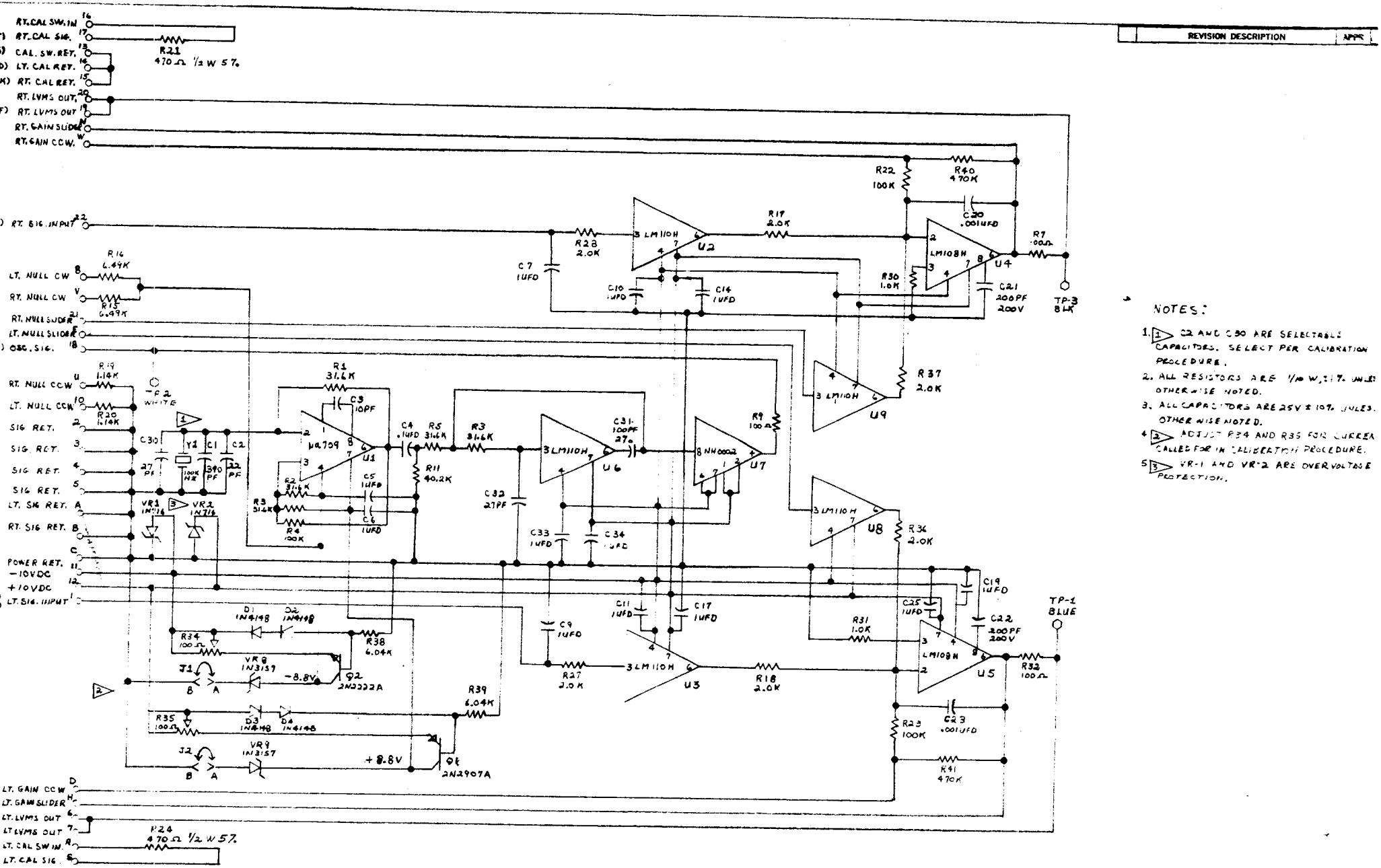
Note: Jumpers shown as connected for .01 sec interrupts

Figure A2.3(b)

NASA LBNP CONTROLLER
A/D CONVERTER BOARD PARTS LIST

A1	7404
A2,A3	74LS138
A4-A7	8212
A8	72747
A9	72741
A10	ADC 590-8, Hybrid Systems A/D Converter
A11	74121
A12	7400
A13-A18	7490
C1,C2	15μf, 25 VDC
C3	.001 μf
C4,C5,C6	.1 μf
C7	47 pf
R1,R3	220 KΩ
R2,R9	100 KΩ pot., Bourns 215P-1-104
R4,R5	10 KΩ
R6,R7,R10	10 KΩ pot.
R8	39 KΩ
R11,R12	1 KΩ
R13-R20	6.8 KΩ
XTAL1	10 MHZ

Figure A2.3(c)



NOTES:

1. 1 C2 AND C30 ARE SELECTABLE CAPACITORS. SELECT PER CALIBRATION PROCEDURE.
2. ALL RESISTORS ARE 1/4 W, ±1% UNLESS OTHERWISE NOTED.
3. ALL CAPACITORS ARE 25V ± 10% UNLESS OTHERWISE NOTED.
4. 2 ADJUST R34 AND R35 FOR CURRENT CALLED FOR IN CALIBRATION PROCEDURE.
5. 3 VR-1 AND VR-2 ARE OVERVOLTAGE PROTECTION.

Figure A2.4 Oscillator-Amplifier Card
(extracted from "Report on Leg Volume Measuring System Electronic Control Unit" by R. W. Radloff, US Air Force Academy.)

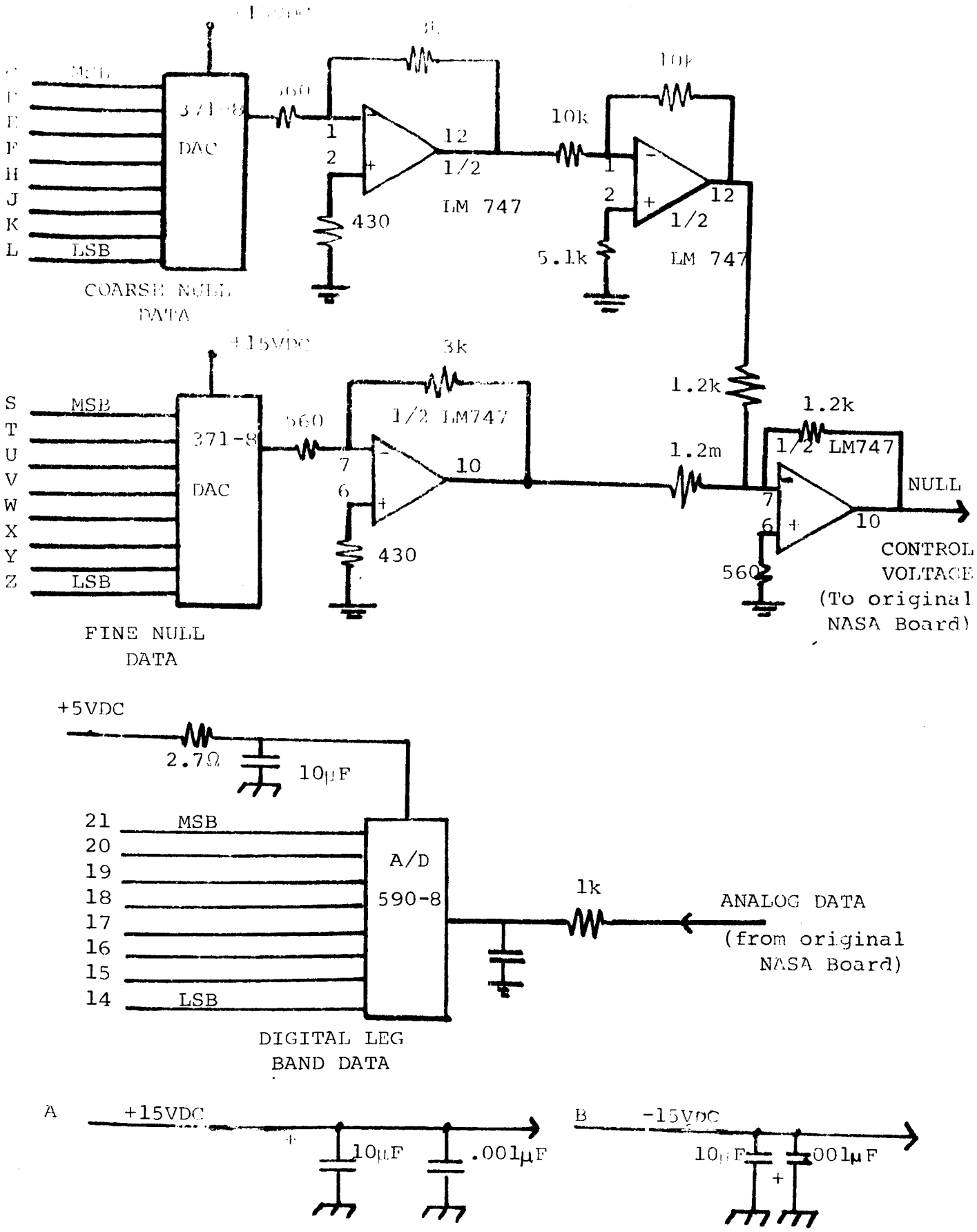


Figure A2.5(a) LBNP CONTROLLER ANALOG AMP BOARD SCHEMATIC DRAWING

Figure A2.5(b)

EDGE CONNECTOR PIN ASSIGNMENTS FOR

LNBP CONTROLLER ANALOG AMP BOARD

1	+5V		A	+15V	
2	A/D Strobe		B	-15V	
3	A/D Busy		C	MSB	Coarse Nulling Input (Output Port 4 or 6)
4	N/C		D		
5	N/C		E		
6	N/C		F		
7	N/C		H		
8	N/C	*See Below	J		
9	N/C	*See Below	K		
10	N/C	*See Below	L	LSB	
11	N/C	*See Below	M	N/C	
12	Nulling Analog Output		N	N/C	
13	A/D Input		P	Analog Gnd	
14	LSB	A/D Output (Input Port 5 or 4)	R	N/C	
15	A/D Output (Input Port 5 or 4)		S	MSB	Fine Nulling Input (Output Port 5 or 7)
16	A/D Output (Input Port 5 or 4)		T		Fine Nulling Input (Output Port 5 or 7)
17	A/D Output (Input Port 5 or 4)		U		Fine Nulling Input (Output Port 5 or 7)
18	A/D Output (Input Port 5 or 4)		V		Fine Nulling Input (Output Port 5 or 7)
19	A/D Output (Input Port 5 or 4)		W		Fine Nulling Input (Output Port 5 or 7)
20	A/D Output (Input Port 5 or 4)		X		Fine Nulling Input (Output Port 5 or 7)
21	MSB	A/D Output (Input Port 5 or 4)	Y		Fine Nulling Input (Output Port 5 or 7)
22	Dig. Gnd		Z	LSB	Fine Nulling Input (Output Port 5 or 7)

On Backplane wiring for the "right" analog Amp Board, pins 8-11 are used:

8 Rt Cal Out
9 (Tie Point for Cal RElay) Bit 6 of Port 3
10 Lt Cal Out
11 (Tie Point for Call Relay) Bit 7 of Port 3

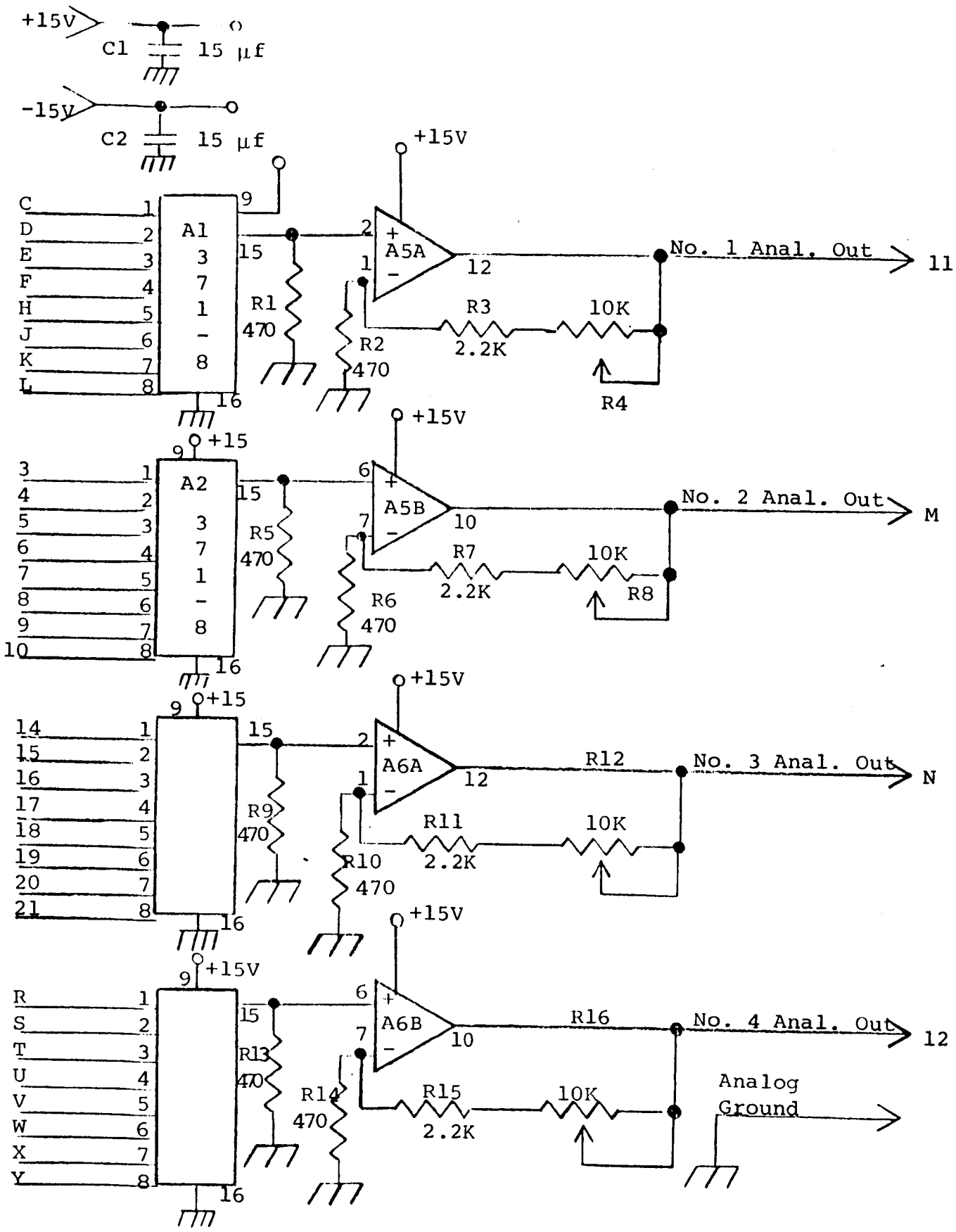
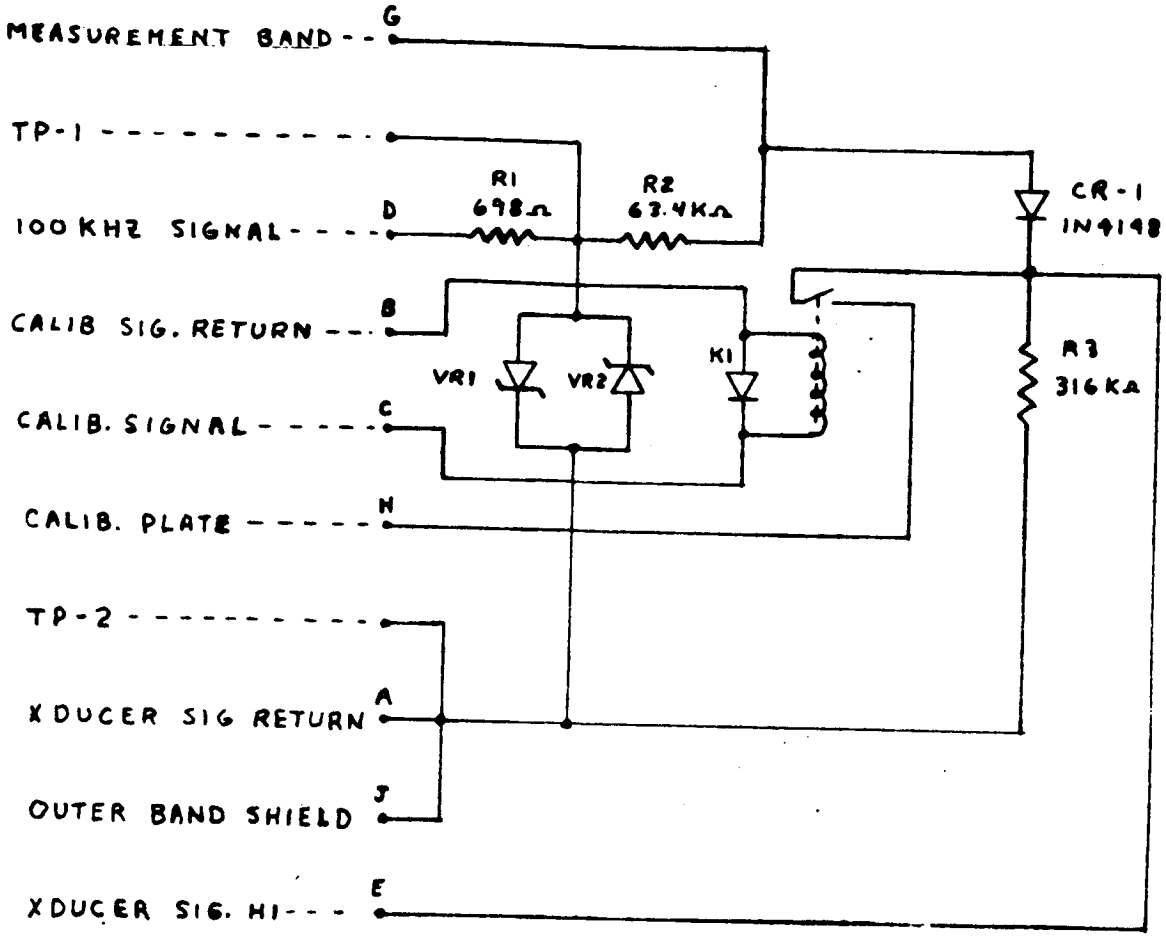


Figure A2.6(a)
LBNP Controller Chart Recorder Driver Schematic Diagram

EDGE CONNECTOR PIN ASSIGNMENTS FOR
LBNP CONTROLLER CHART RECORDER DRIVER

1	N/C		A	+15V	
2	N/C		B	-15V	
3	MSB	Driver No. 2 Input	C	MSB	Driver No. 1 Input
4			D		
5			E		
6			F		
7			H		
8			J		
9			K	LSB	
10	LSB		L		
11	No. 1 Anal. Out		M	No. 2 Anal. Out	
12	No. 4 Anal. Out		N An	No. 3 Anal. Out	
13	Anal. Gnd		P	Anal. Gnd	
14	MSB	Driver No. 3 Input	R	MSB	Driver No. 4 Input
15			S		
16			T		
17			U		
18			V		
19			W		
20			X		
21	LSB		Y	LSB	
22	Dig. Gnd		Z		

Figure A2.6(b)



NOTE:

ALL ELECTRONIC COMPONENTS ARE POTTED AND AND FIXED TO THE LEG BAND.

FIGURE A2.7 CAPACITIVE PLETHYSMOGRAPH ELECTRONICS

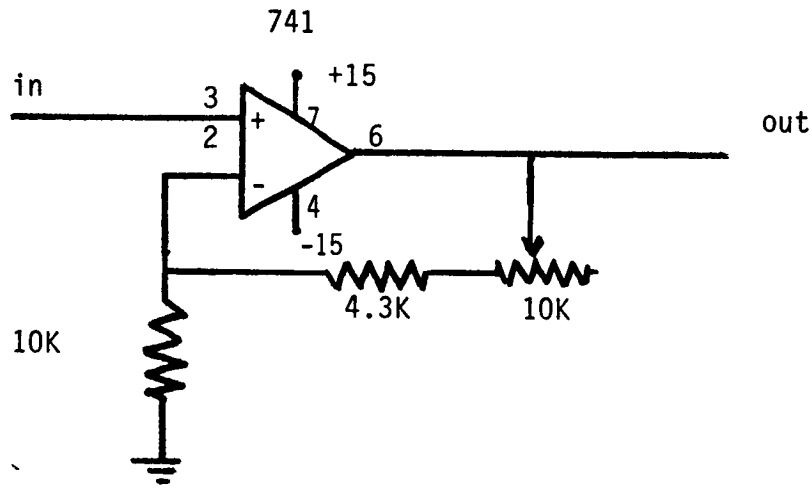


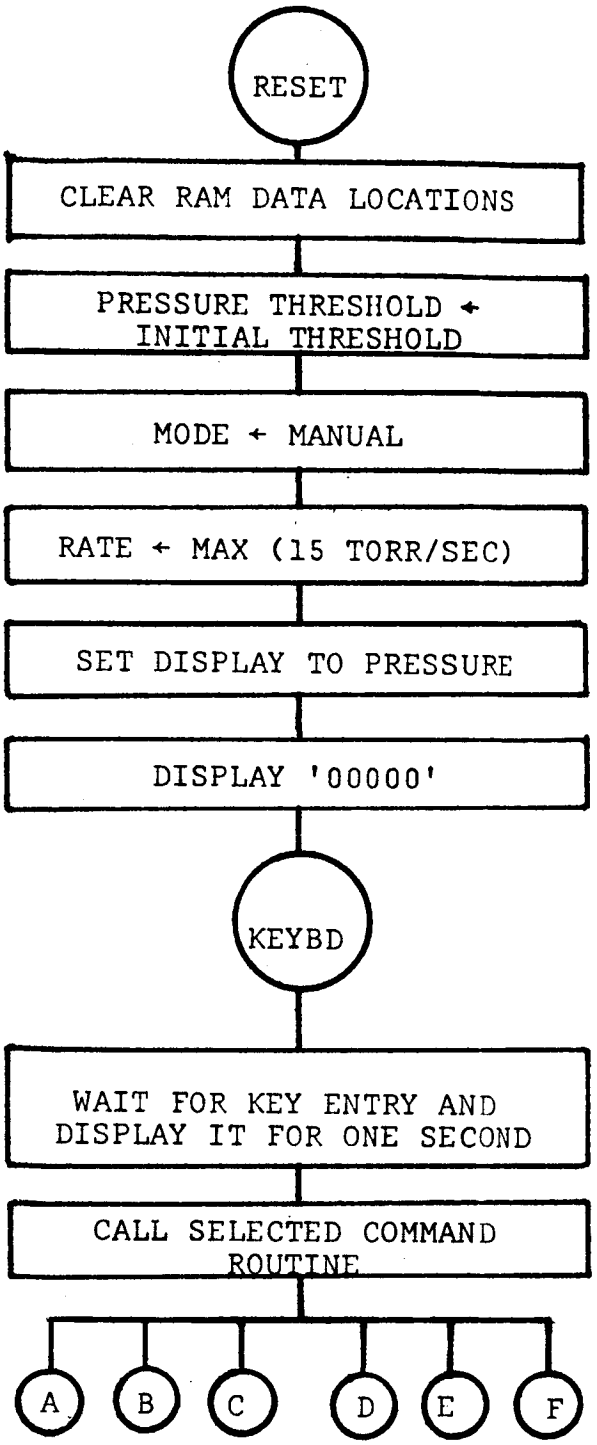
Figure A2.8 Leg Band Data A/D Input Amplifier

Two of these are used; one for the right channel and one for the left channel.

APPENDIX III

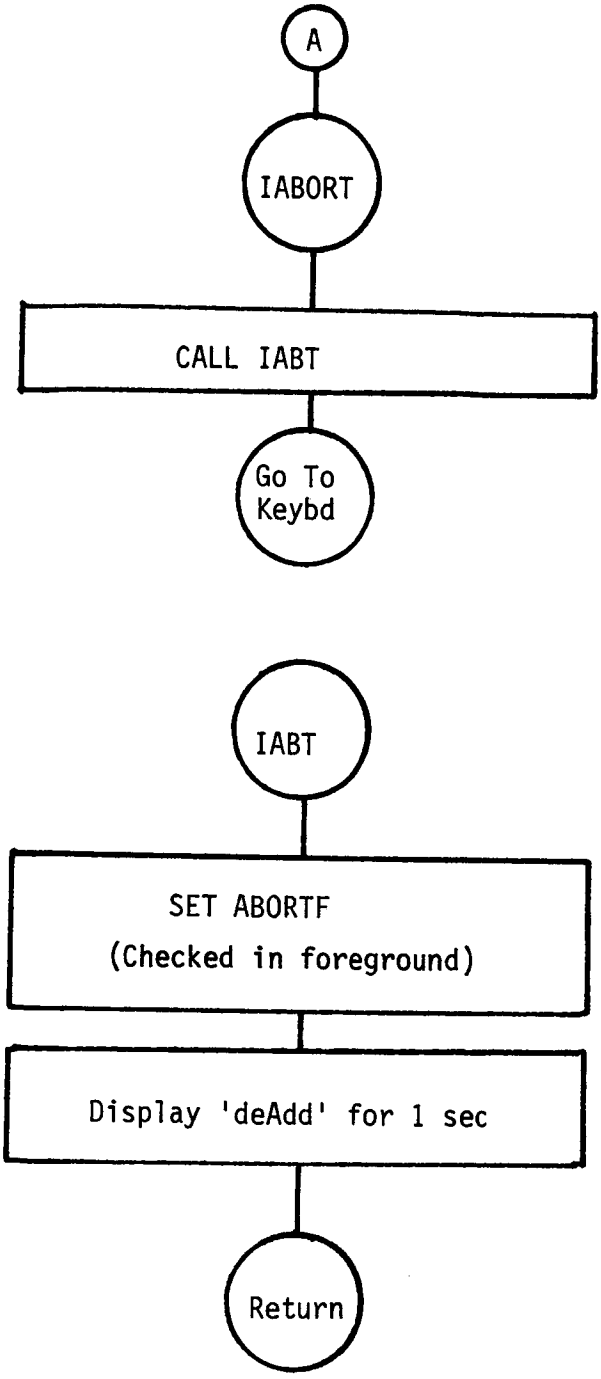
FLOW CHARTS

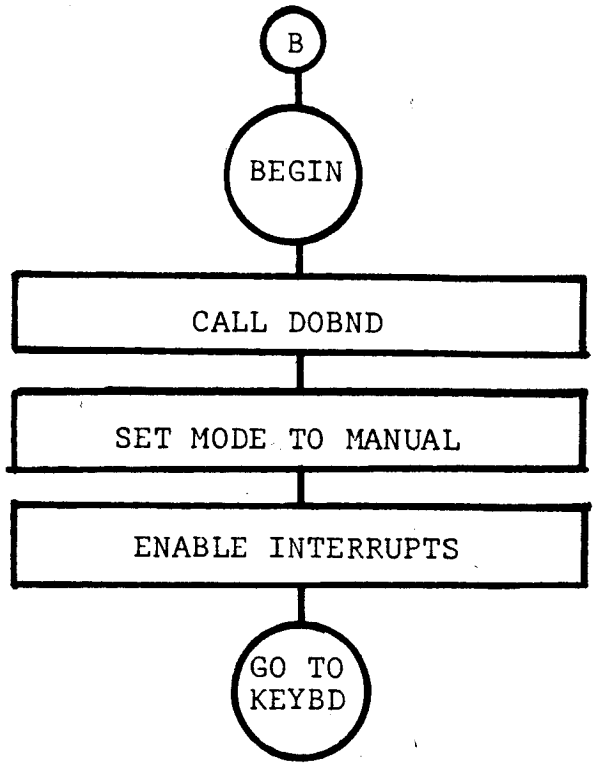
BACKGROUND



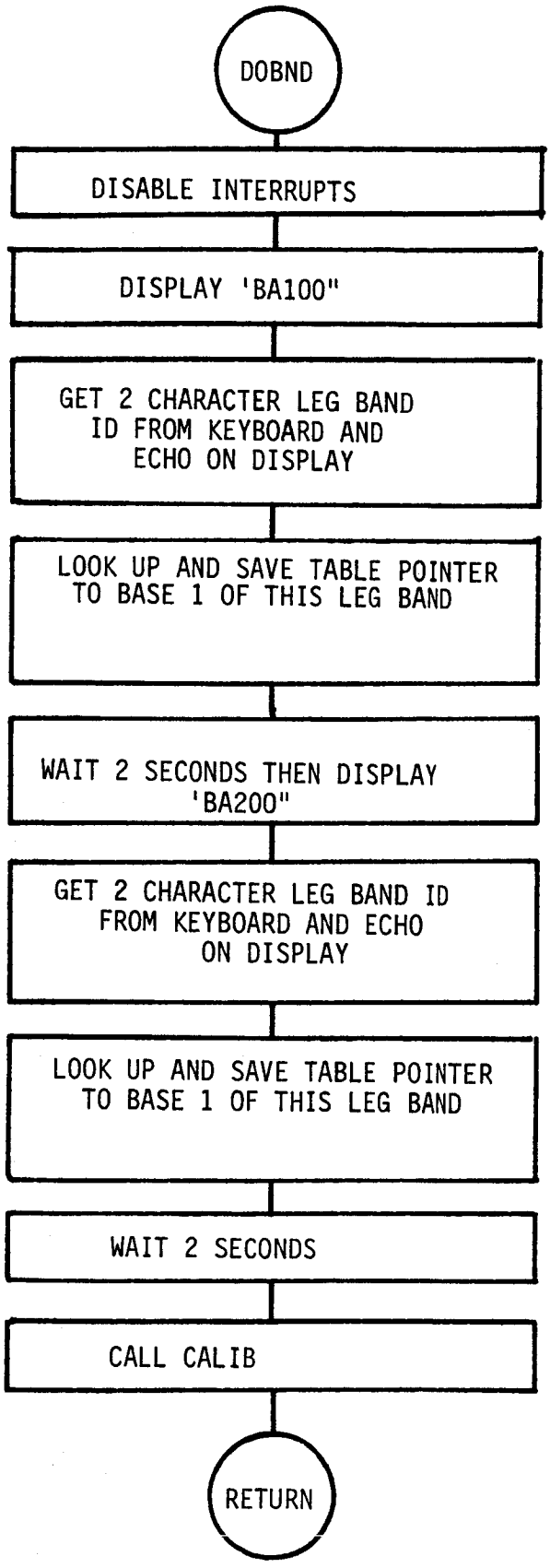
(RESET KEY OR POWER ON CALLS THIS ROUTINE)

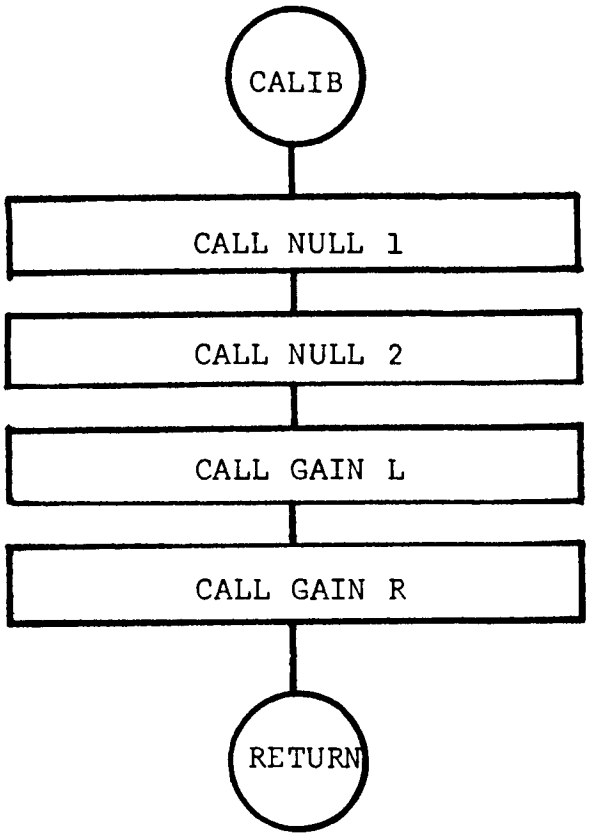
(THIS ROUTINE EXECUTES WHEN NO INTERRUPTS ARE ACTIVE)

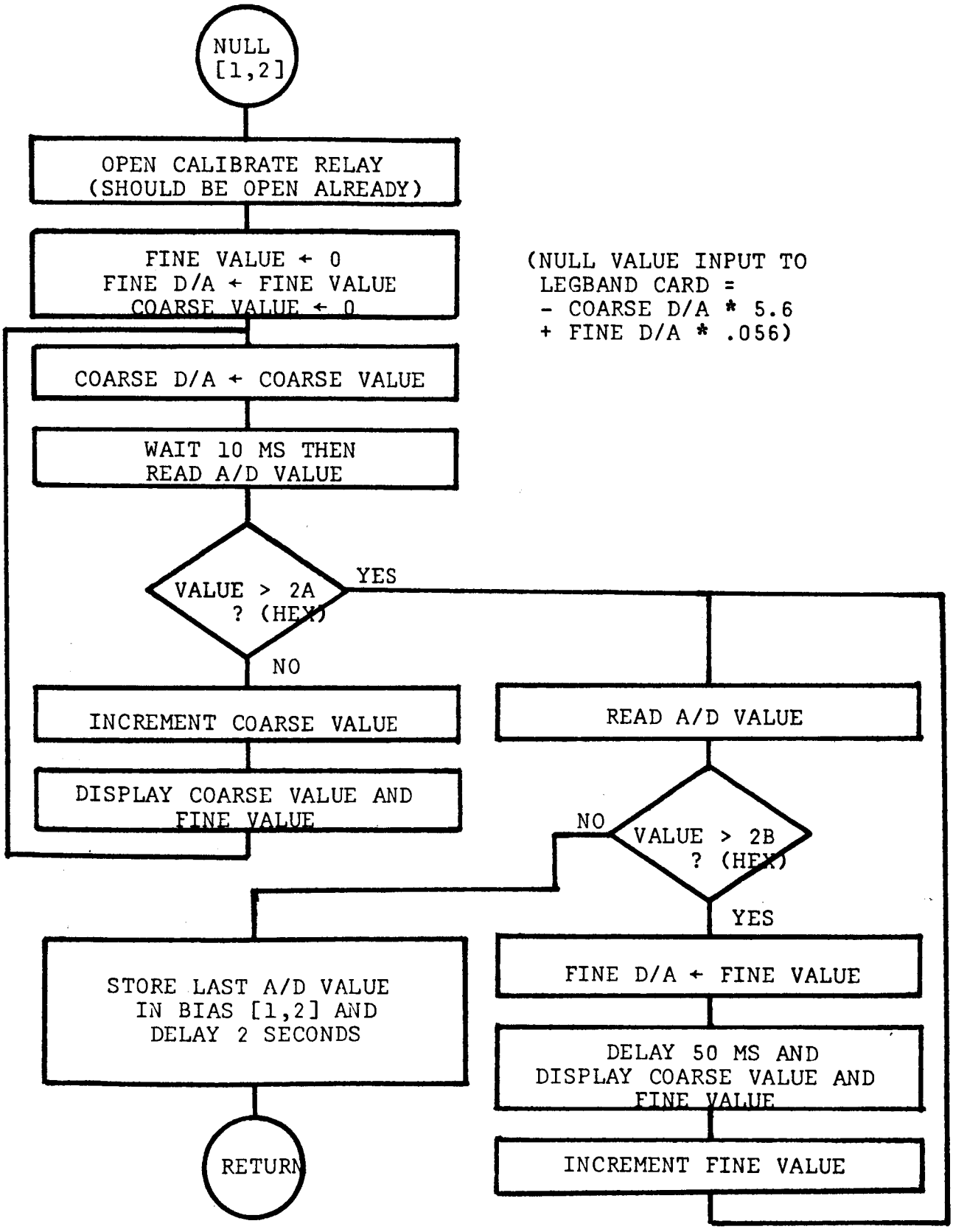

A
IABORT
CALL IABT
Go To
Keybd
IABT
SET ABORTF
(Checked in foreground)
Display 'deAdd' for 1 sec
Return

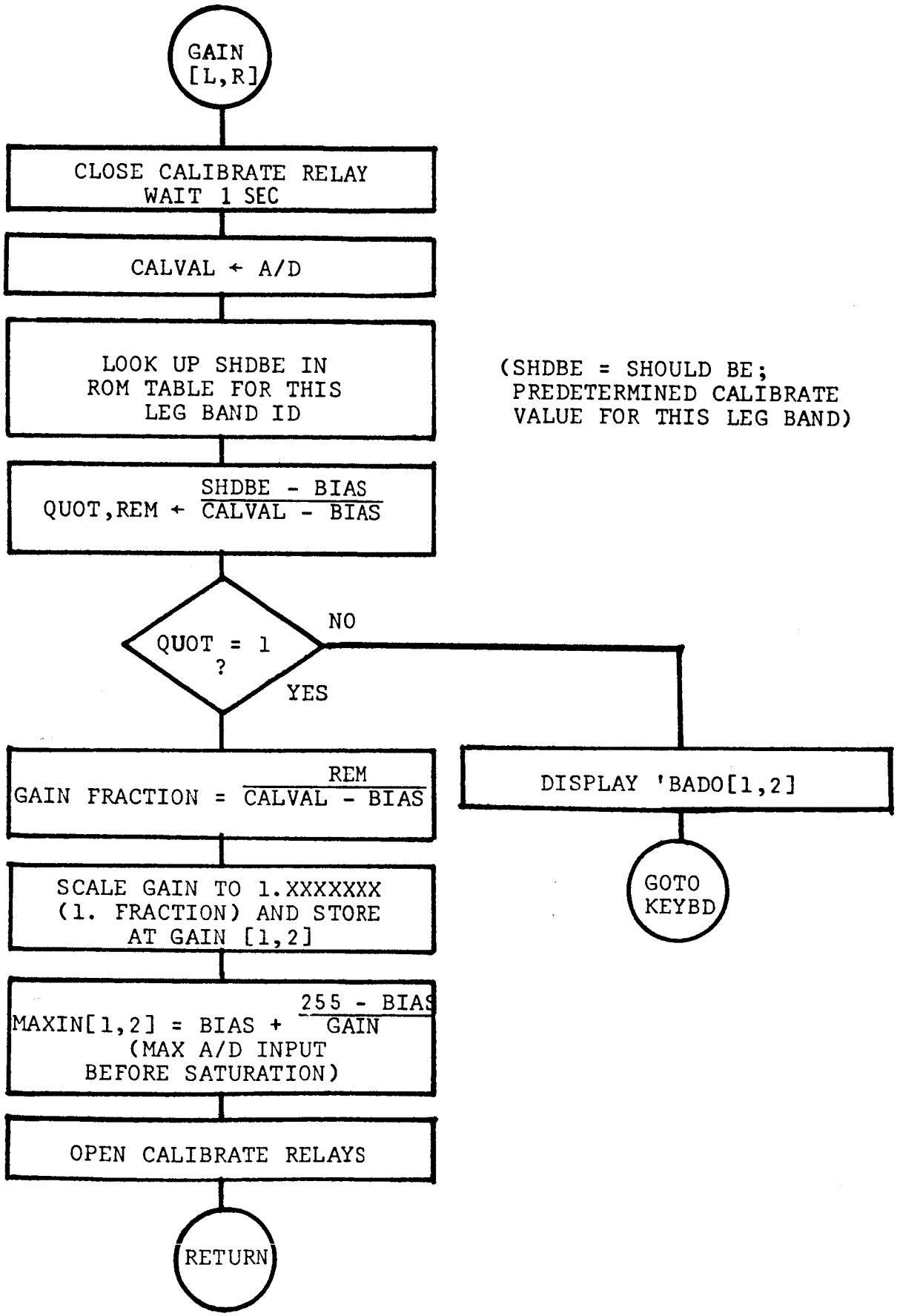


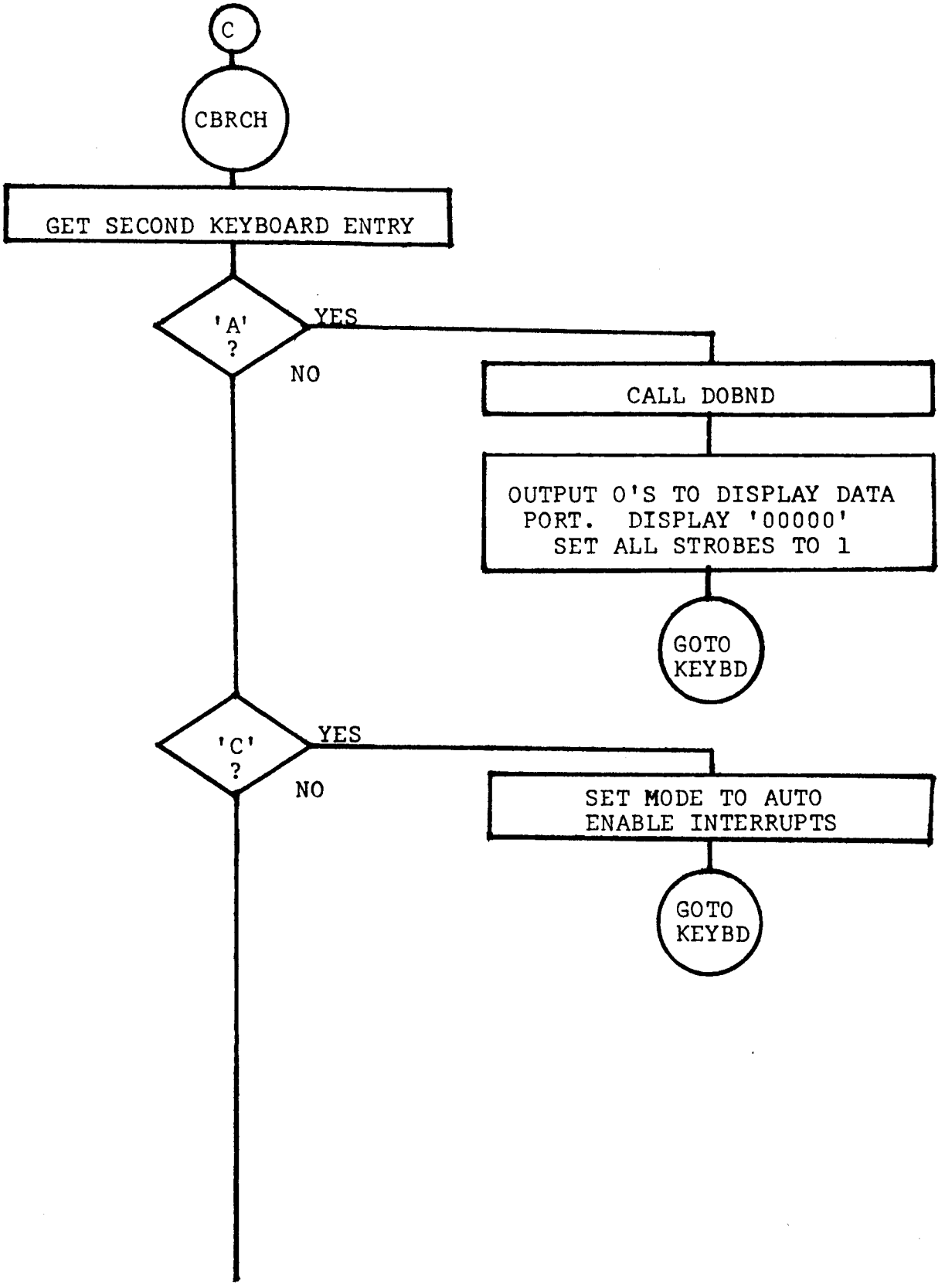
(STARTS PRESSURE CONTROL
AND LEG BAND DATA ACQUISITION)

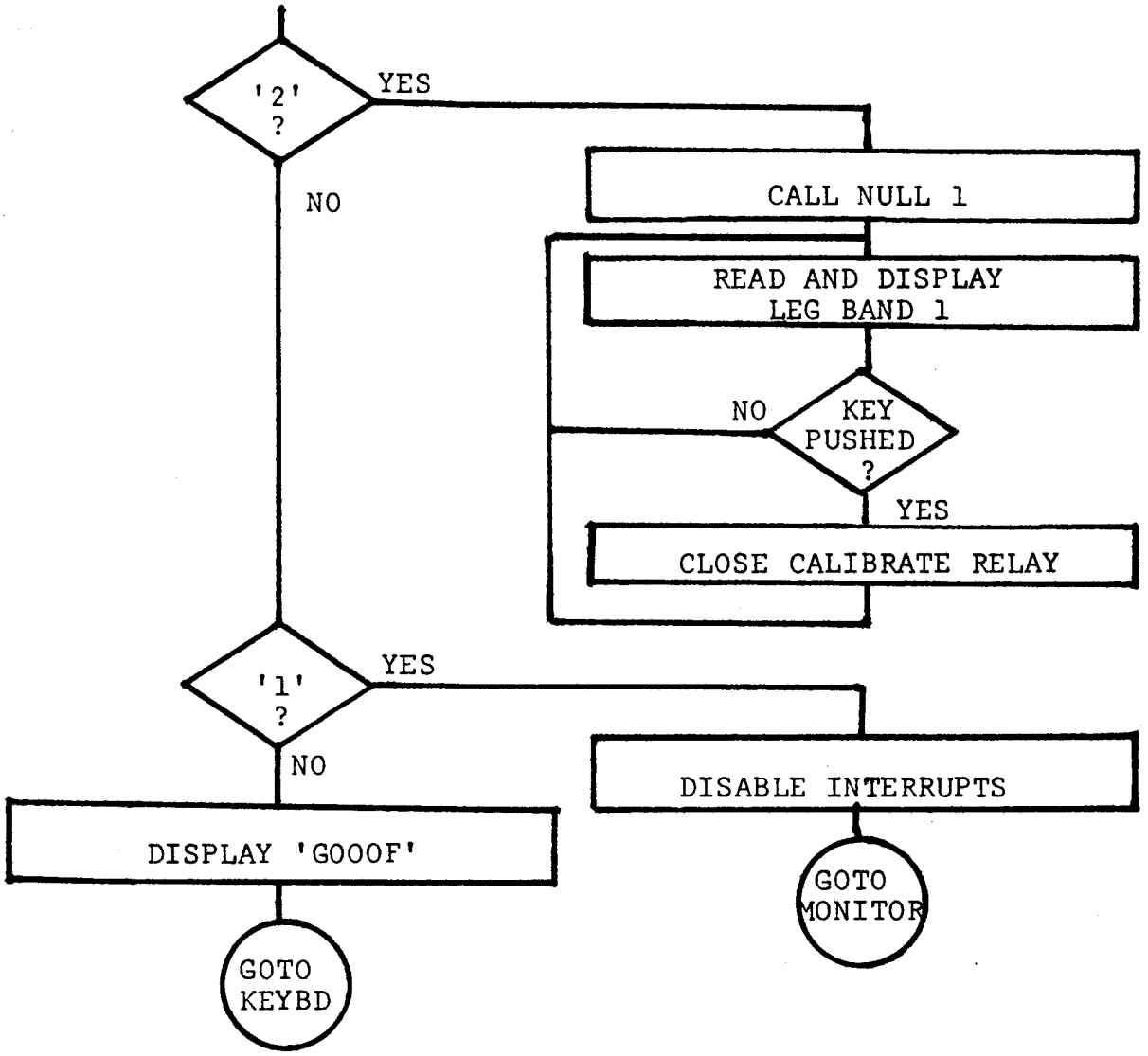

DOBND
DISABLE INTERRUPTS
DISPLAY 'BA100"
GET 2 CHARACTER LEG BAND ID FROM KEYBOARD AND ECHO ON DISPLAY
LOOK UP AND SAVE TABLE POINTER TO BASE 1 OF THIS LEG BAND
WAIT 2 SECONDS THEN DISPLAY 'BA200"
GET 2 CHARACTER LEG BAND ID FROM KEYBOARD AND ECHO ON DISPLAY
LOOK UP AND SAVE TABLE POINTER TO BASE 1 OF THIS LEG BAND
WAIT 2 SECONDS
CALL CALIB
RETURN

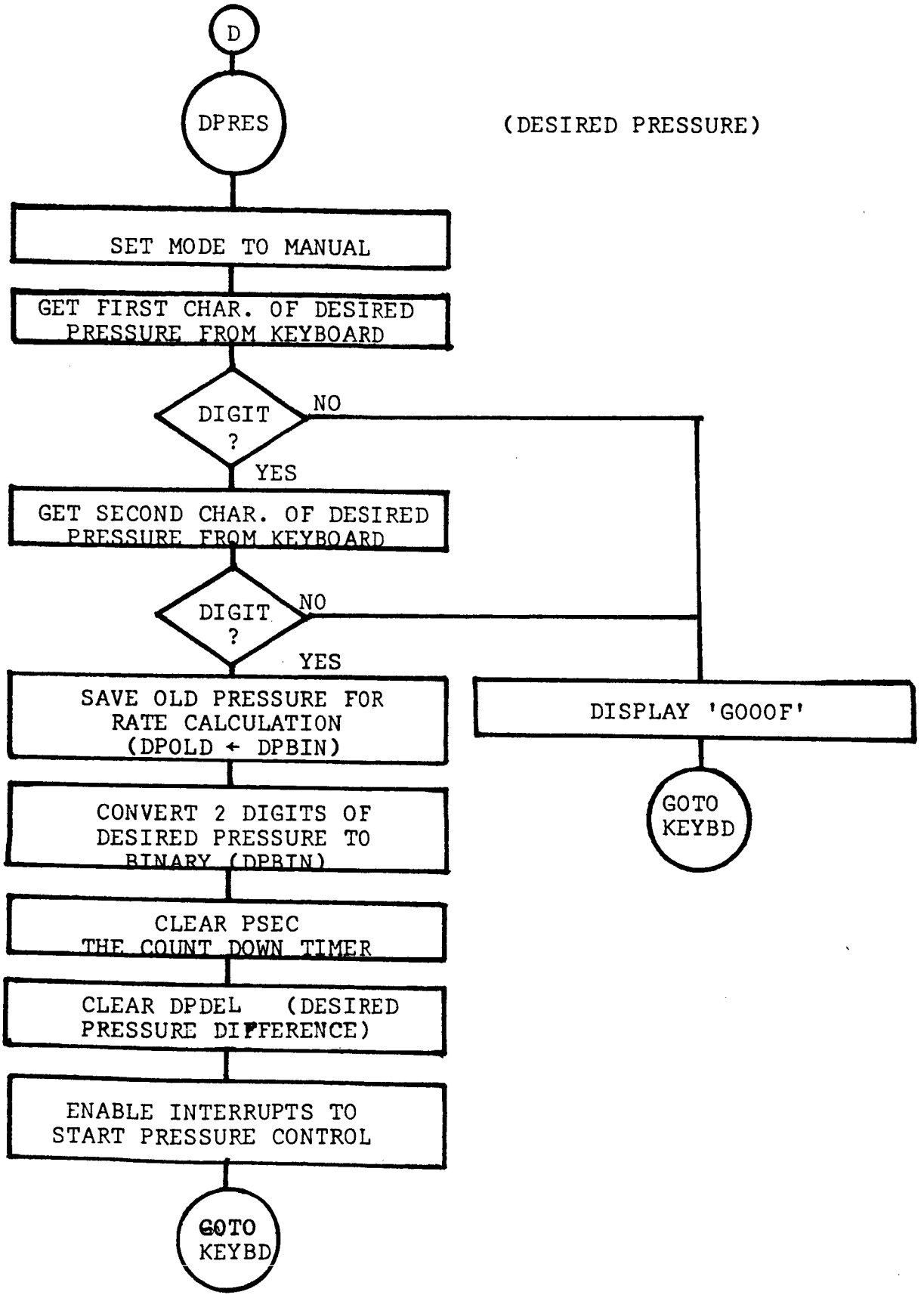

CALIB
CALL NULL 1
CALL NULL 2
CALL GAIN L
CALL GAIN R
RETURN

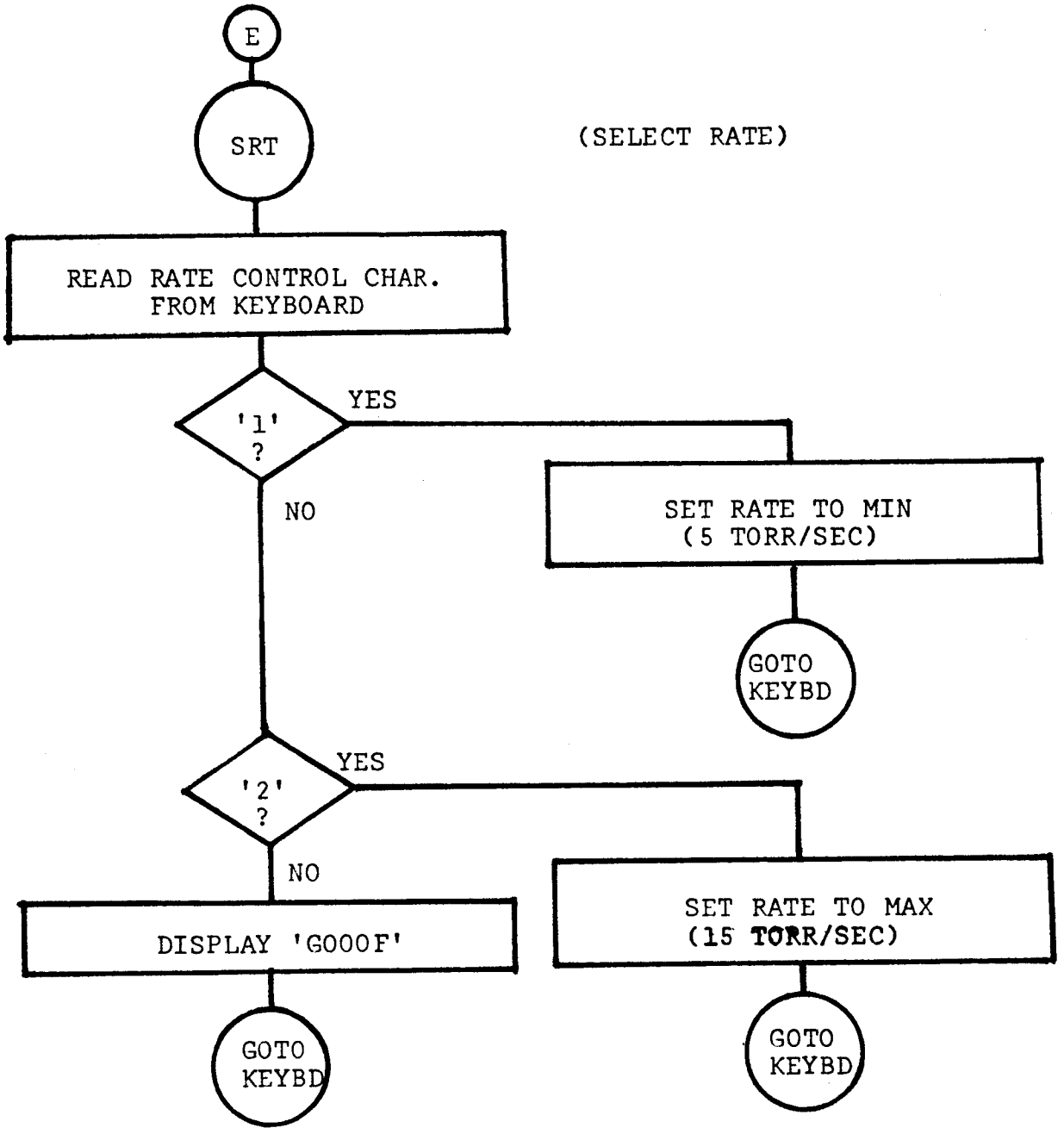

NULL
[1,2]
OPEN CALIBRATE RELAY
(SHOULD BE OPEN ALREADY)
FINE VALUE ← 0
FINE D/A ← FINE VALUE
COARSE VALUE ← 0
(NULL VALUE INPUT TO
LEGBAND CARD =
- COARSE D/A * 5.6
+ FINE D/A * .056)
COARSE D/A ← COARSE VALUE
WAIT 10 MS THEN
READ A/D VALUE
VALUE > 2A
? (HEX)
YES
NO
INCREMENT COARSE VALUE
DISPLAY COARSE VALUE AND
FINE VALUE
READ A/D VALUE
NO
VALUE > 2B
? (HEX)
YES
STORE LAST A/D VALUE
IN BIAS [1,2] AND
DELAY 2 SECONDS
FINE D/A ← FINE VALUE
DELAY 50 MS AND
DISPLAY COARSE VALUE AND
FINE VALUE
RETURN
INCREMENT FINE VALUE

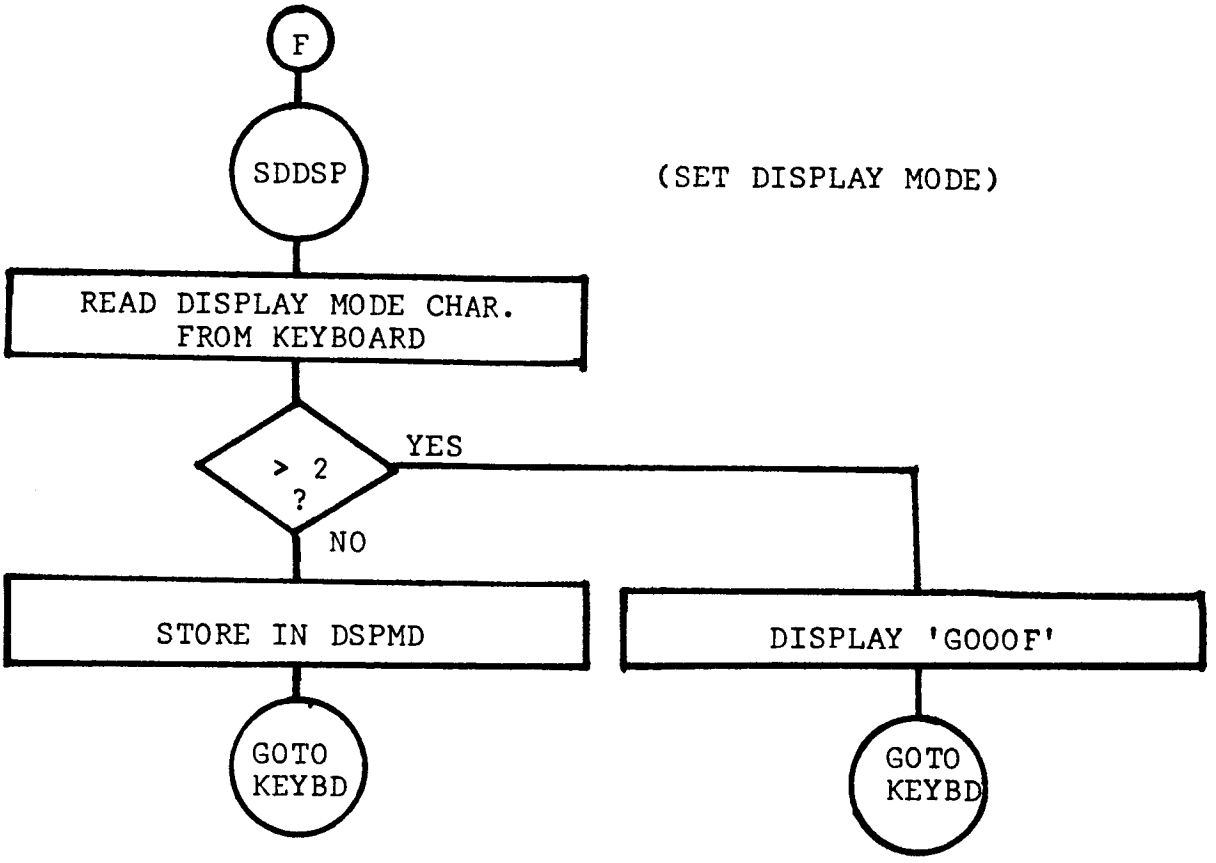

GAIN [L,R]
CLOSE CALIBRATE RELAY
WAIT 1 SEC
CALVAL ← A/D
LOOK UP SHDBE IN ROM TABLE FOR THIS LEG BAND ID
(SHDBE = SHOULD BE; PREDETERMINED CALIBRATE VALUE FOR THIS LEG BAND)
QUOT,REM ← (SHDBE - BIAS) / (CALVAL - BIAS)
QUOT = 1 ?
NO
YES
DISPLAY 'BADO[1,2]
GOTO KEYBD
GAIN FRACTION = REM / (CALVAL - BIAS)
SCALE GAIN TO 1.XXXXXXX (1. FRACTION) AND STORE AT GAIN [1,2]
MAXIN[1,2] = BIAS + (255 - BIAS) / GAIN
(MAX A/D INPUT BEFORE SATURATION)
OPEN CALIBRATE RELAYS
RETURN

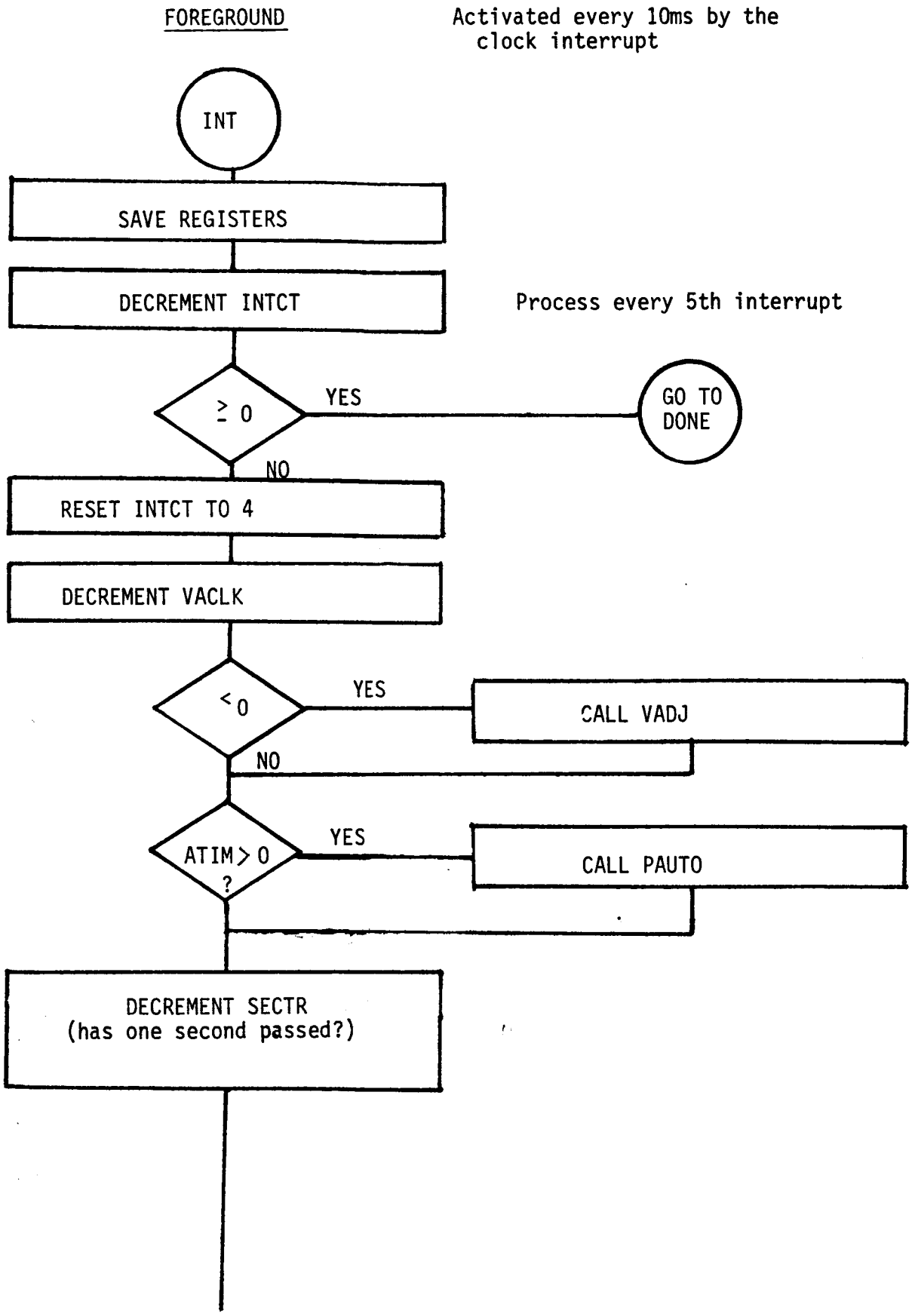

C
CBRCH
GET SECOND KEYBOARD ENTRY
'A' ?
YES
NO
CALL DOBND
OUTPUT 0'S TO DISPLAY DATA PORT. DISPLAY '00000' SET ALL STROBES TO 1
GOTO KEYBD
'C' ?
YES
NO
SET MODE TO AUTO ENABLE INTERRUPTS
GOTO KEYBD

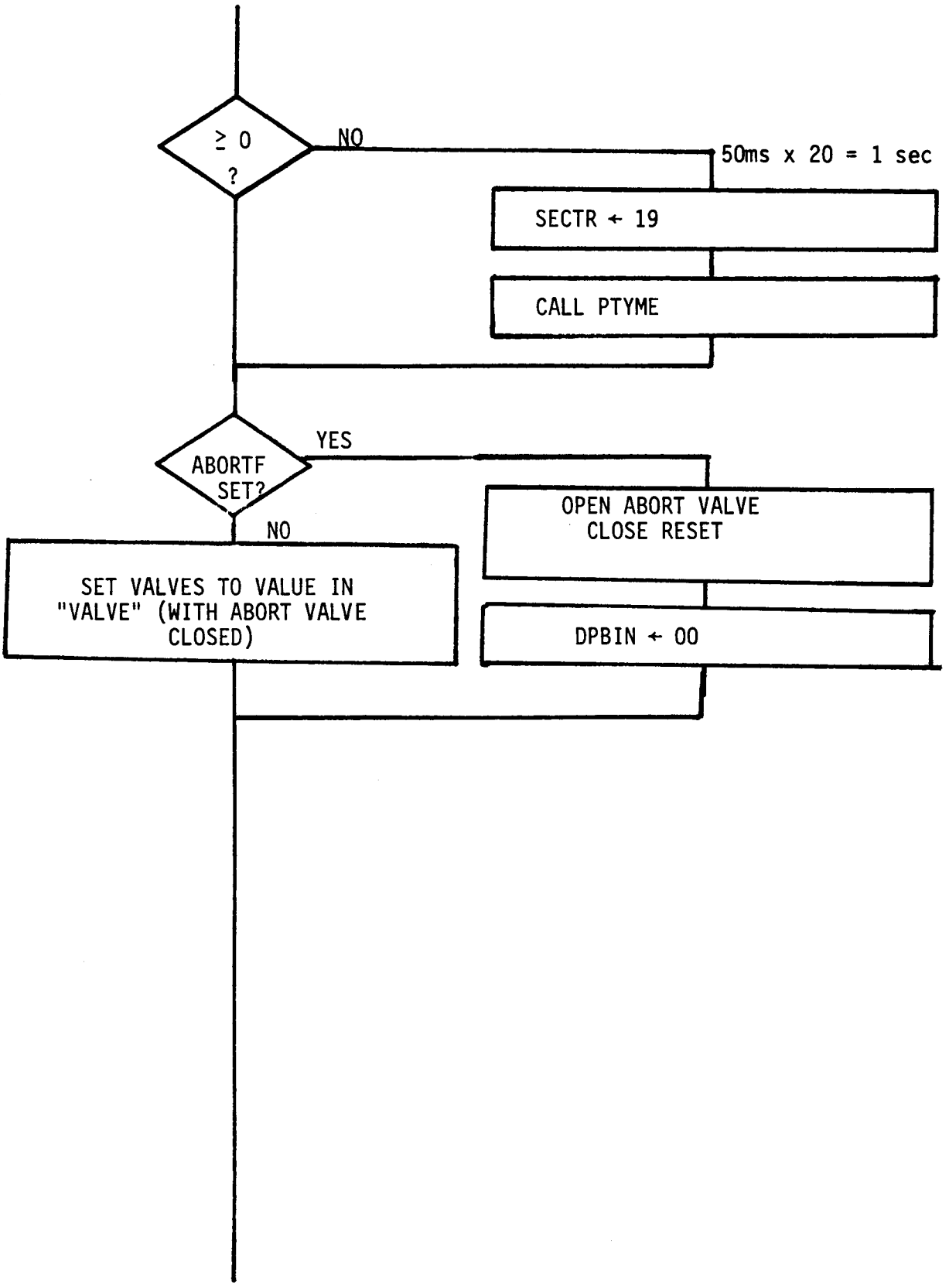

'2' ?
YES
NO
CALL NULL 1
READ AND DISPLAY LEG BAND 1
NO
KEY PUSHED ?
YES
CLOSE CALIBRATE RELAY
'1' ?
YES
NO
DISABLE INTERRUPTS
DISPLAY 'GOOOF'
GOTO MONITOR
GOTO KEYBD

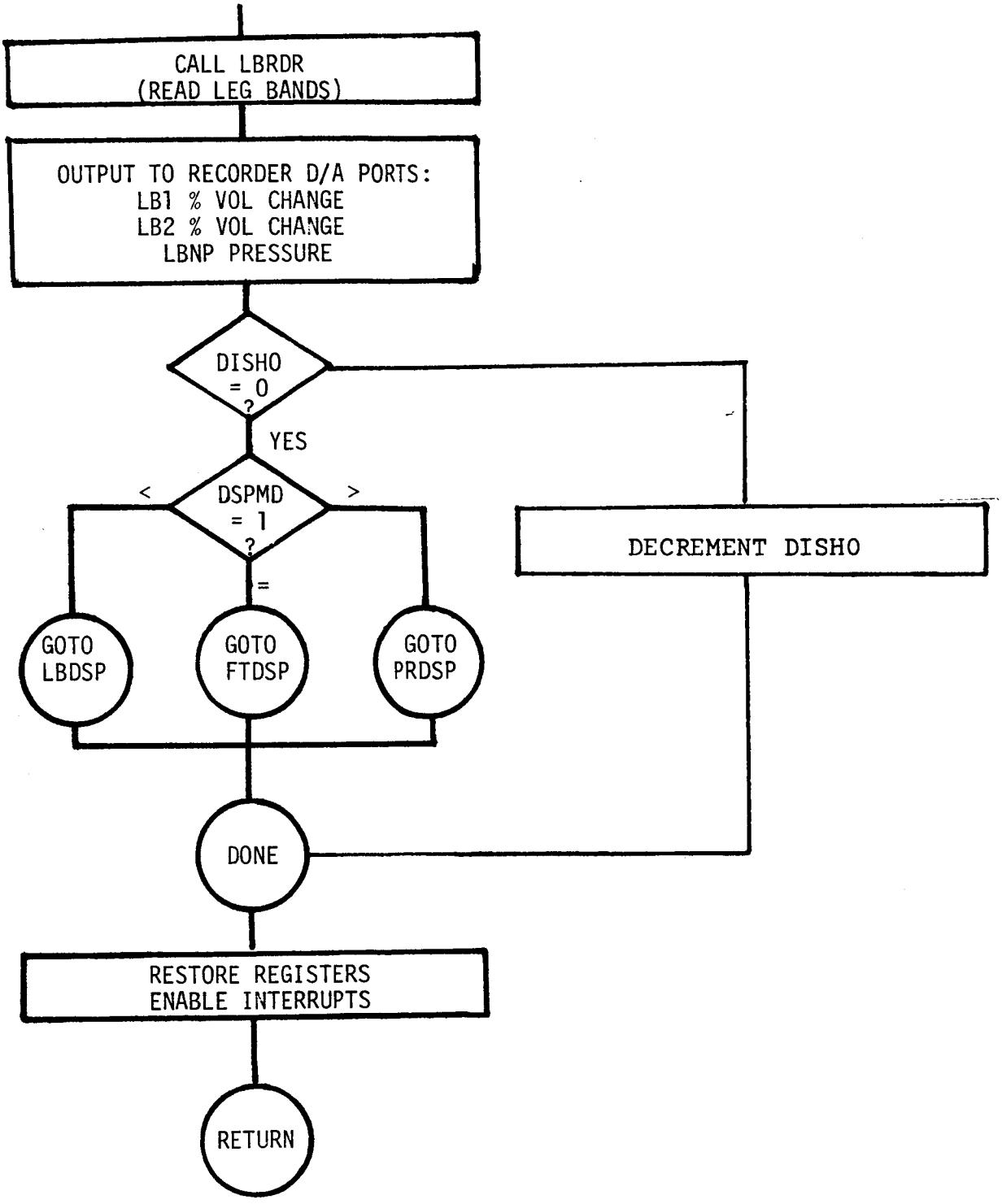

D
DPRES
(DESIRED PRESSURE)
SET MODE TO MANUAL
GET FIRST CHAR. OF DESIRED PRESSURE FROM KEYBOARD
DIGIT ?
NO
YES
GET SECOND CHAR. OF DESIRED PRESSURE FROM KEYBOARD
DIGIT ?
NO
YES
SAVE OLD PRESSURE FOR RATE CALCULATION (DPOLD ← DPBIN)
DISPLAY 'GOOOF'
GOTO KEYBD
CONVERT 2 DIGITS OF DESIRED PRESSURE TO BINARY (DPBIN)
CLEAR PSEC THE COUNT DOWN TIMER
CLEAR DPDEL (DESIRED PRESSURE DIFFERENCE)
ENABLE INTERRUPTS TO START PRESSURE CONTROL
GOTO KEYBD

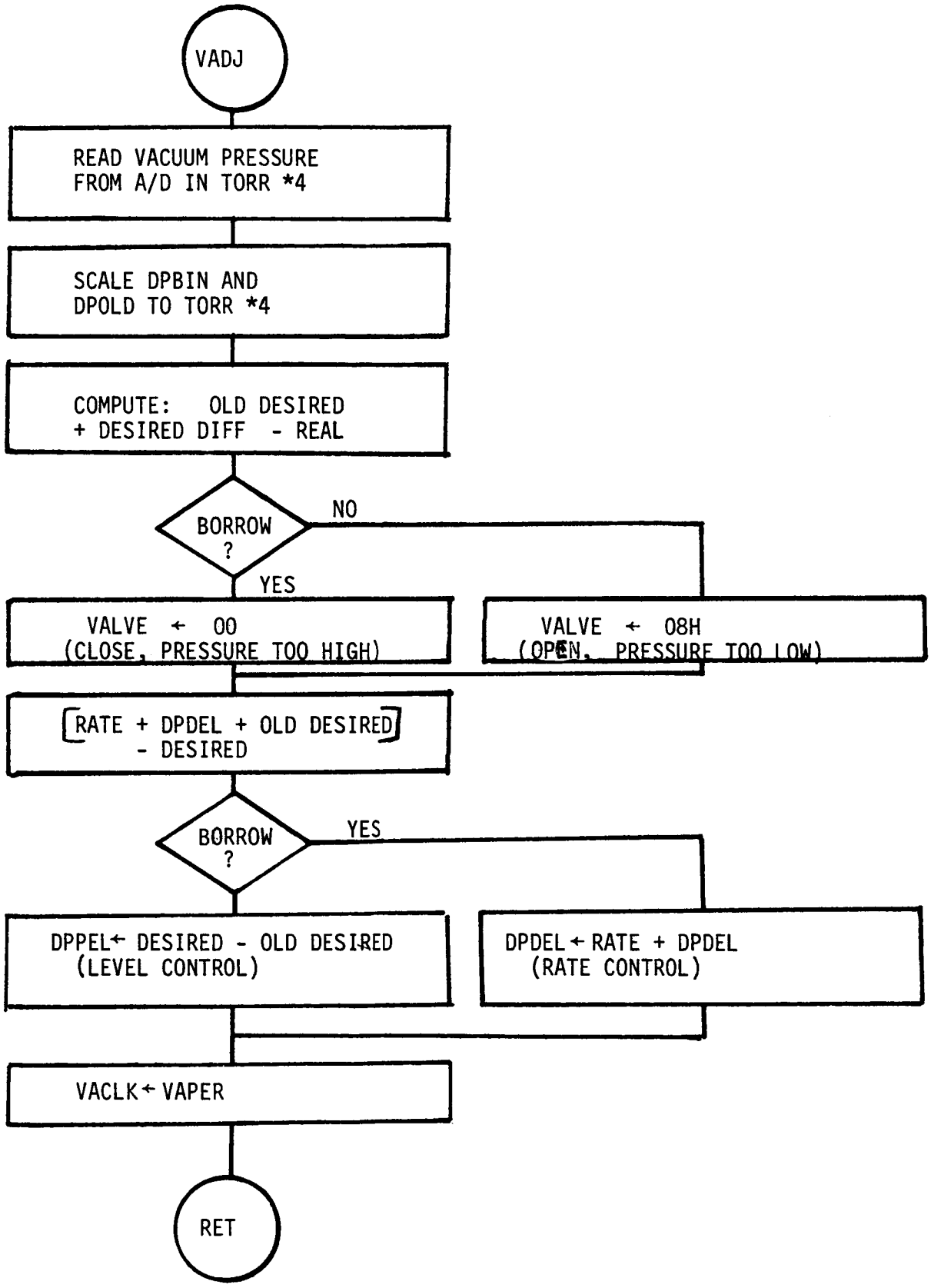

E
SRT
(SELECT RATE)
READ RATE CONTROL CHAR.
FROM KEYBOARD
'1'
?
YES
NO
SET RATE TO MIN
(5 TORR/SEC)
GOTO
KEYBD
'2'
?
YES
NO
DISPLAY 'G000F'
SET RATE TO MAX
(15 TORR/SEC)
GOTO
KEYBD
GOTO
KEYBD

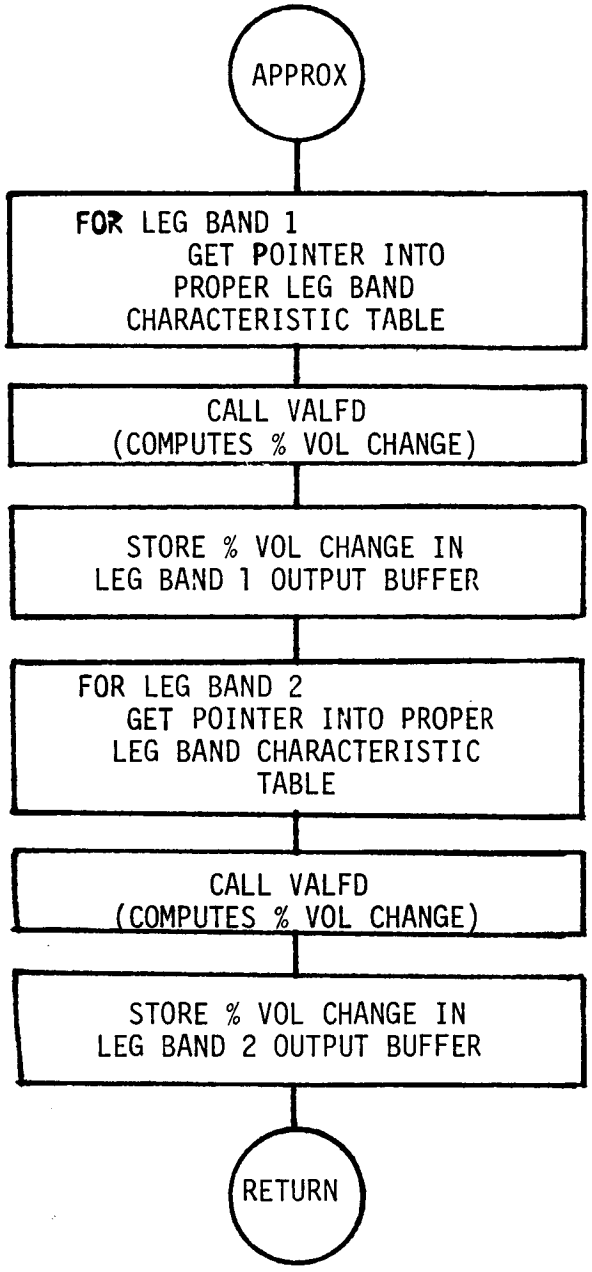

F
SDDSP
(SET DISPLAY MODE)
READ DISPLAY MODE CHAR.
FROM KEYBOARD
> 2
?
YES
NO
STORE IN DSPMD
DISPLAY 'G000F'
GOTO
KEYBD
GOTO
KEYBD

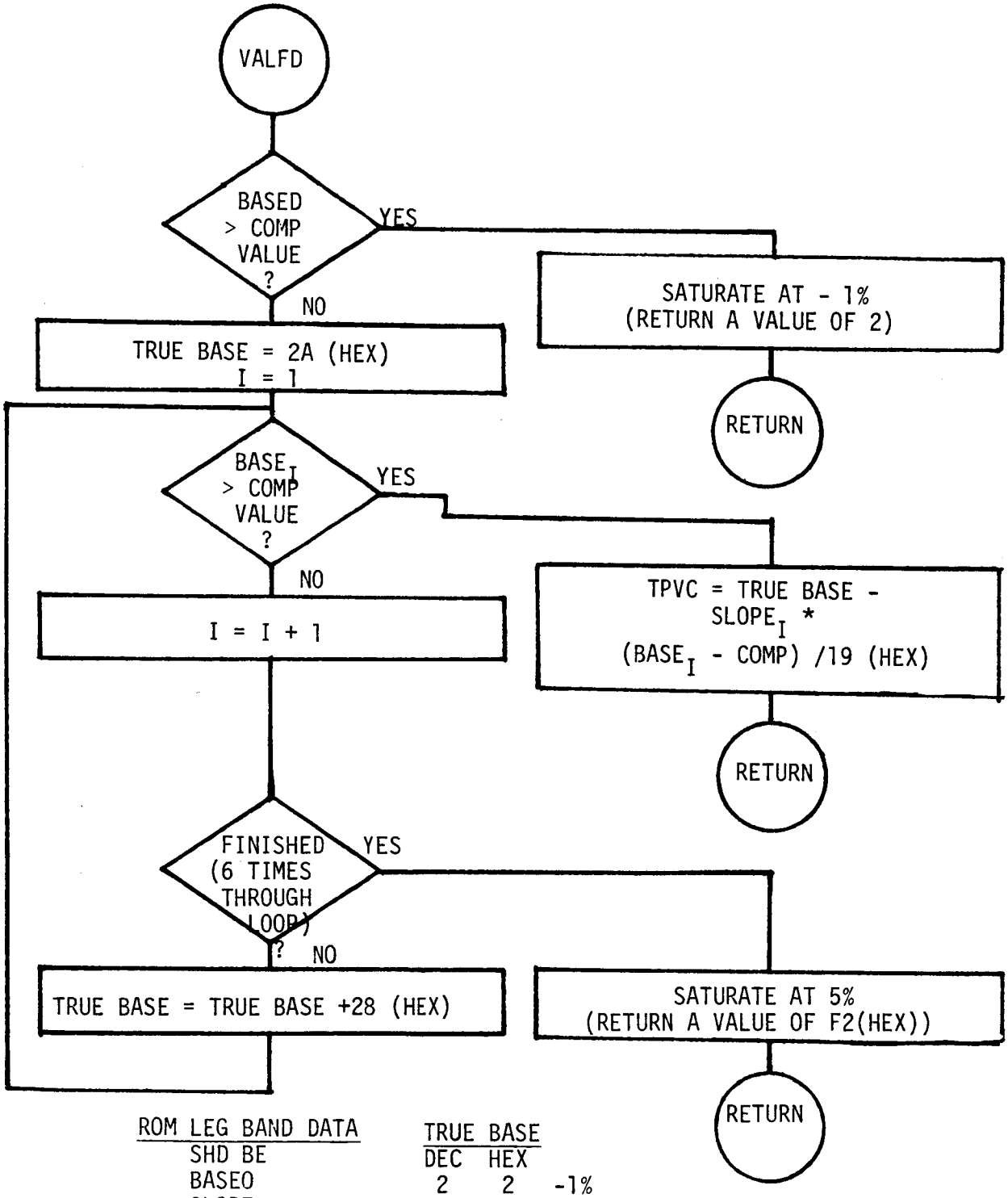

FOREGROUND
Activated every 10ms by the clock interrupt
INT
SAVE REGISTERS
DECREMENT INTCT
Process every 5th interrupt
≥ 0
YES
GO TO DONE
NO
RESET INTCT TO 4
DECREMENT VACLK
< 0
YES
CALL VADJ
NO
ATIM > 0 ?
YES
CALL PAUTO
DECREMENT SECTR
(has one second passed?)


≥ 0 ?
NO
50ms x 20 = 1 sec
SECTR ← 19
CALL PTYME
ABORTF SET?
YES
NO
OPEN ABORT VALVE
CLOSE RESET
SET VALVES TO VALUE IN "VALVE" (WITH ABORT VALVE CLOSED)
DPBIN ← 00


CALL LBRDR
(READ LEG BANDS)
OUTPUT TO RECORDER D/A PORTS:
LB1 % VOL CHANGE
LB2 % VOL CHANGE
LBNP PRESSURE
DISHO
= 0
?
YES
DSPMD
= 1
?
<
>
=
DECREMENT DISHO
GOTO
LBDSP
GOTO
FTDSP
GOTO
PRDSP
DONE
RESTORE REGISTERS
ENABLE INTERRUPTS
RETURN

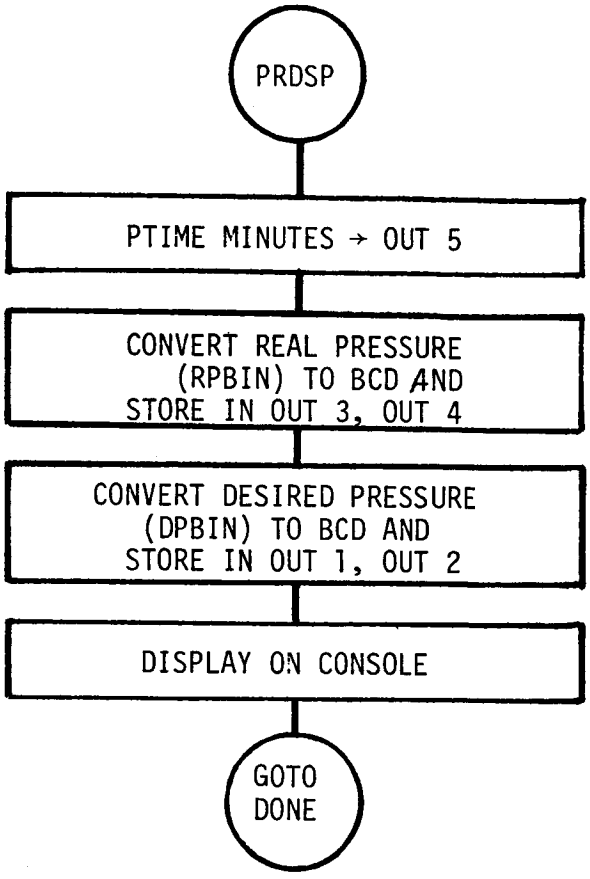

VADJ
READ VACUUM PRESSURE
FROM A/D IN TORR *4
SCALE DPBIN AND
DPOLD TO TORR *4
COMPUTE: OLD DESIRED
+ DESIRED DIFF - REAL
BORROW
?
NO
YES
VALVE ← 00
(CLOSE, PRESSURE TOO HIGH)
VALVE ← 08H
(OPEN, PRESSURE TOO LOW)
[RATE + DPDEL + OLD DESIRED]
- DESIRED
BORROW
?
YES
DPPEL ← DESIRED - OLD DESIRED
(LEVEL CONTROL)
DPDEL ← RATE + DPDEL
(RATE CONTROL)
VACLK ← VAPER
RET


APPROX
FOR LEG BAND 1
GET POINTER INTO
PROPER LEG BAND
CHARACTERISTIC TABLE
CALL VALFD
(COMPUTES % VOL CHANGE)
STORE % VOL CHANGE IN
LEG BAND 1 OUTPUT BUFFER
FOR LEG BAND 2
GET POINTER INTO PROPER
LEG BAND CHARACTERISTIC
TABLE
CALL VALFD
(COMPUTES % VOL CHANGE)
STORE % VOL CHANGE IN
LEG BAND 2 OUTPUT BUFFER
RETURN

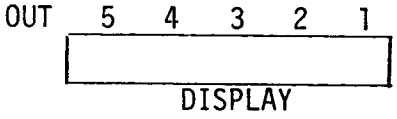


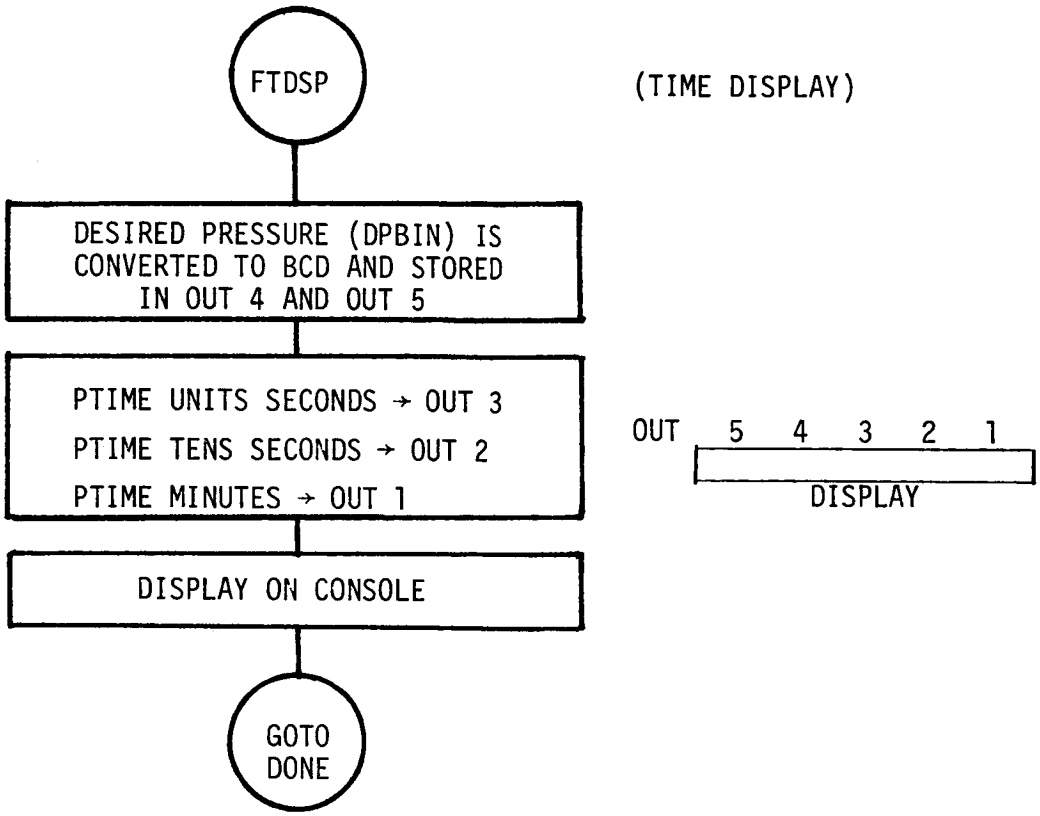
ROM LEG BAND DATA
SHD BE
BASE0
SLOPE
BASE1
SLOPE
BASE2
SLOPE
BASE3
SLOPE
BASE4
SLOPE
BASE5
SLOPE
BASE6
SLOPE

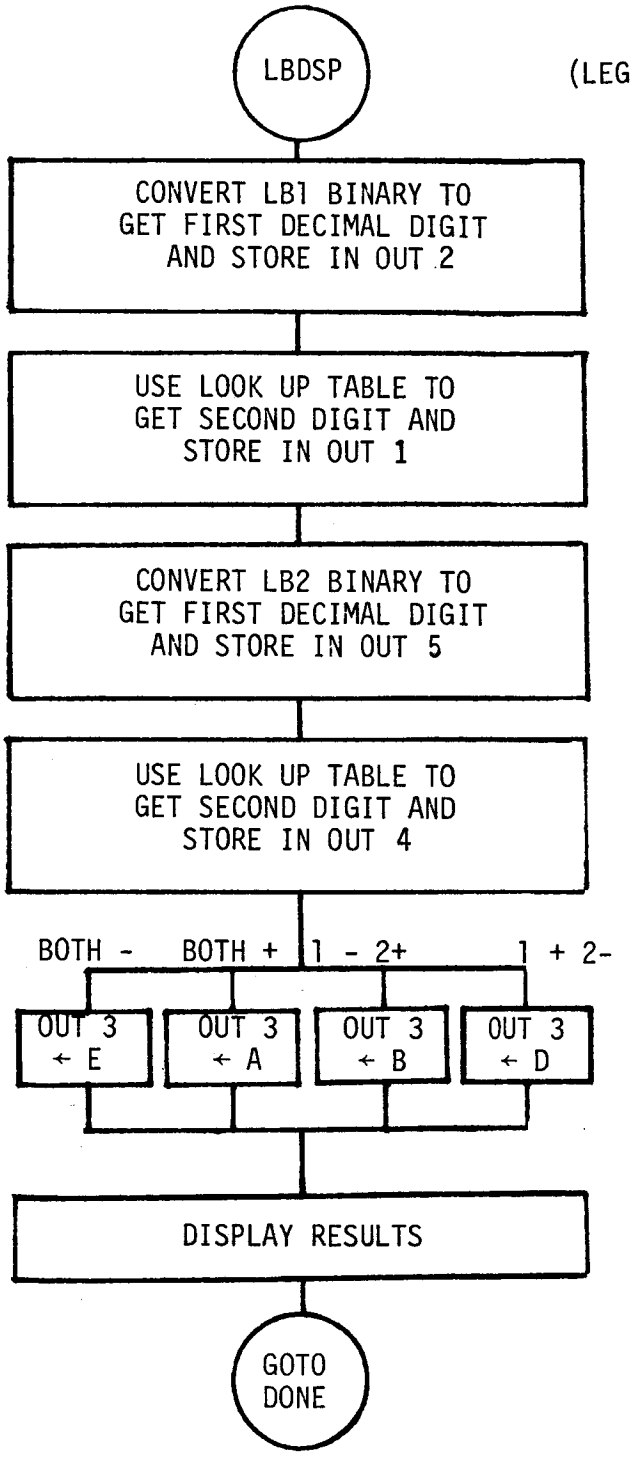
TRUE BASE		
DEC	HEX	
2	2	-1%
42	2A	0%
02	52	1%
122	7A	2%
162	A2	3%
203	CA	4%
242	F2	5%

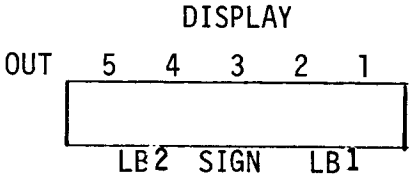


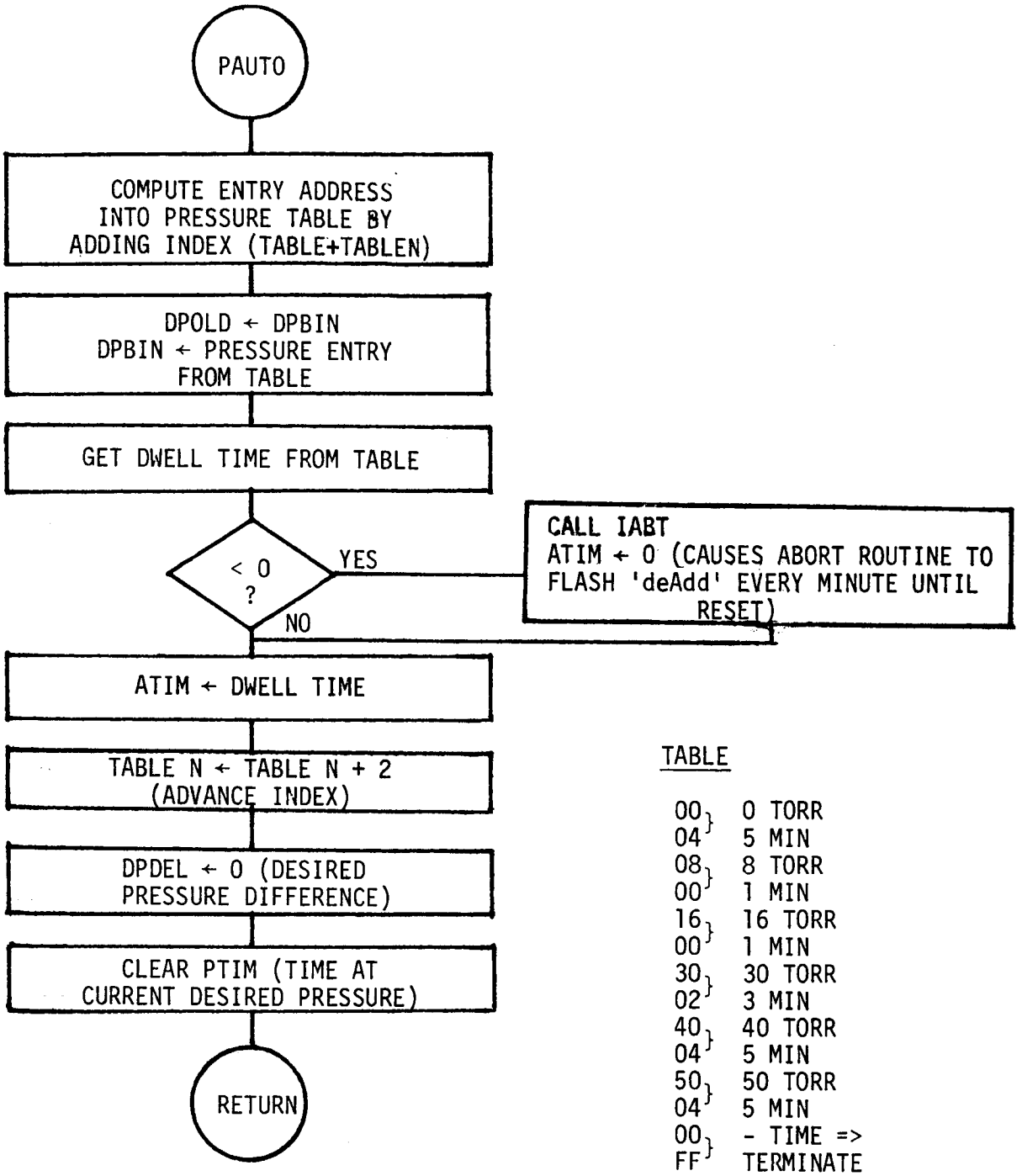
(PRESSURE DISPLAY)




FTDSP
(TIME DISPLAY)
DESIRED PRESSURE (DPBIN) IS CONVERTED TO BCD AND STORED IN OUT 4 AND OUT 5
PTIME UNITS SECONDS → OUT 3
PTIME TENS SECONDS → OUT 2
PTIME MINUTES → OUT 1
OUT 5 4 3 2 1
DISPLAY
DISPLAY ON CONSOLE
GOTO DONE

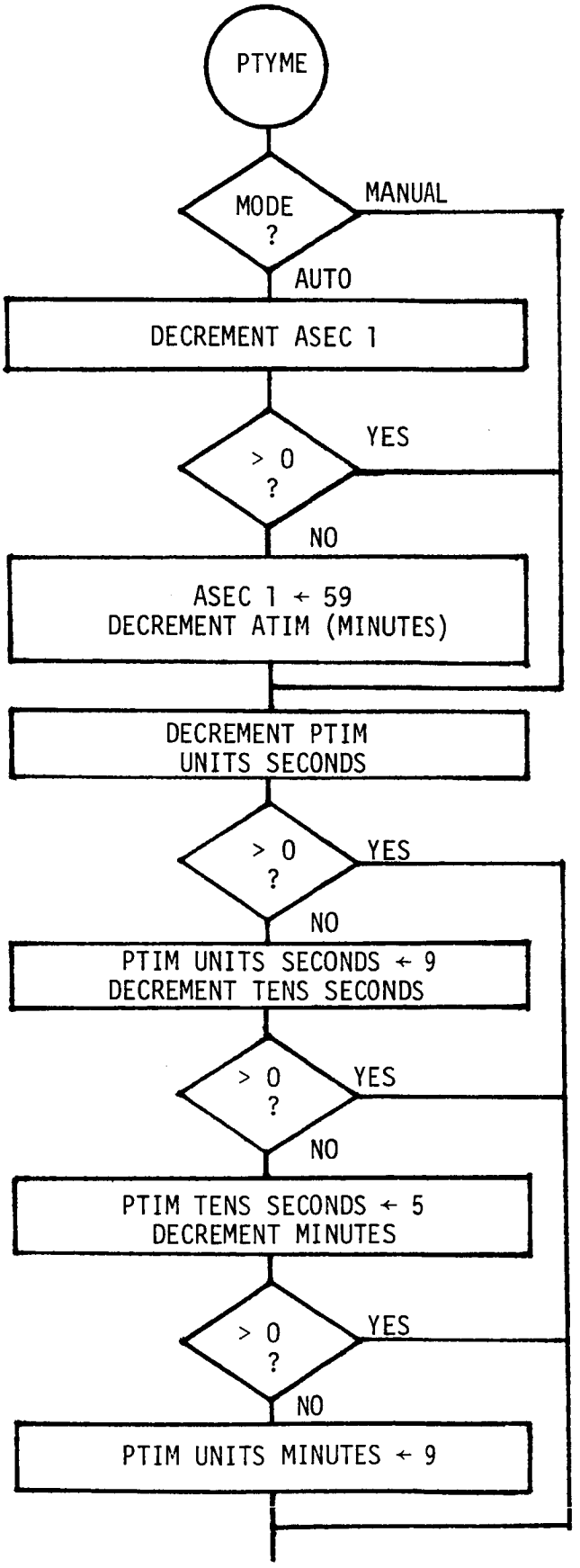

LBDSP
(LEG BAND DISPLAY)
CONVERT LB1 BINARY TO GET FIRST DECIMAL DIGIT AND STORE IN OUT 2
USE LOOK UP TABLE TO GET SECOND DIGIT AND STORE IN OUT 1
CONVERT LB2 BINARY TO GET FIRST DECIMAL DIGIT AND STORE IN OUT 5
USE LOOK UP TABLE TO GET SECOND DIGIT AND STORE IN OUT 4
BOTH -
BOTH +
1 - 2+
1 + 2-
OUT 3 ← E
OUT 3 ← A
OUT 3 ← B
OUT 3 ← D
DISPLAY RESULTS
GOTO DONE


DISPLAY
OUT 5 4 3 2 1
LB2 SIGN LB1



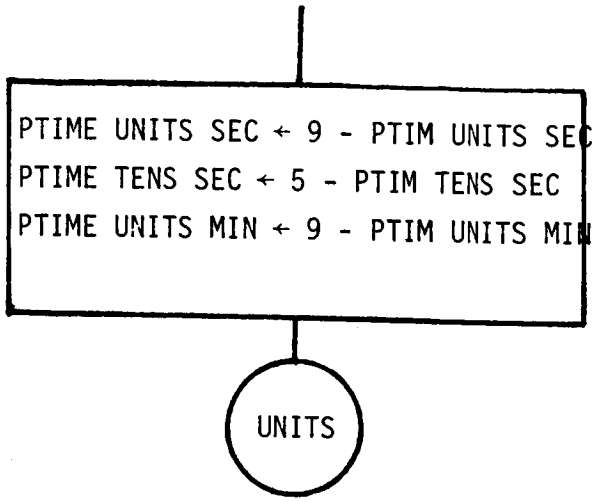
TABLE

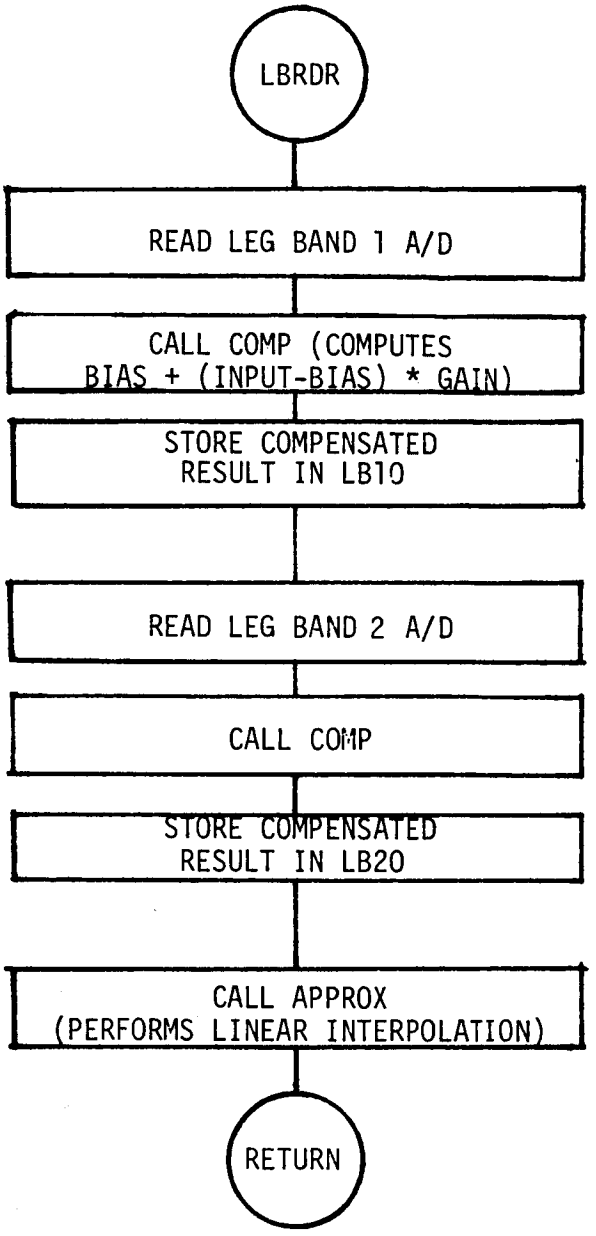
00	0 TORR
04	5 MIN
08	8 TORR
00	1 MIN
16	16 TORR
00	1 MIN
30	30 TORR
02	3 MIN
40	40 TORR
04	5 MIN
50	50 TORR
04	5 MIN
00	- TIME =>
FF	TERMINATE



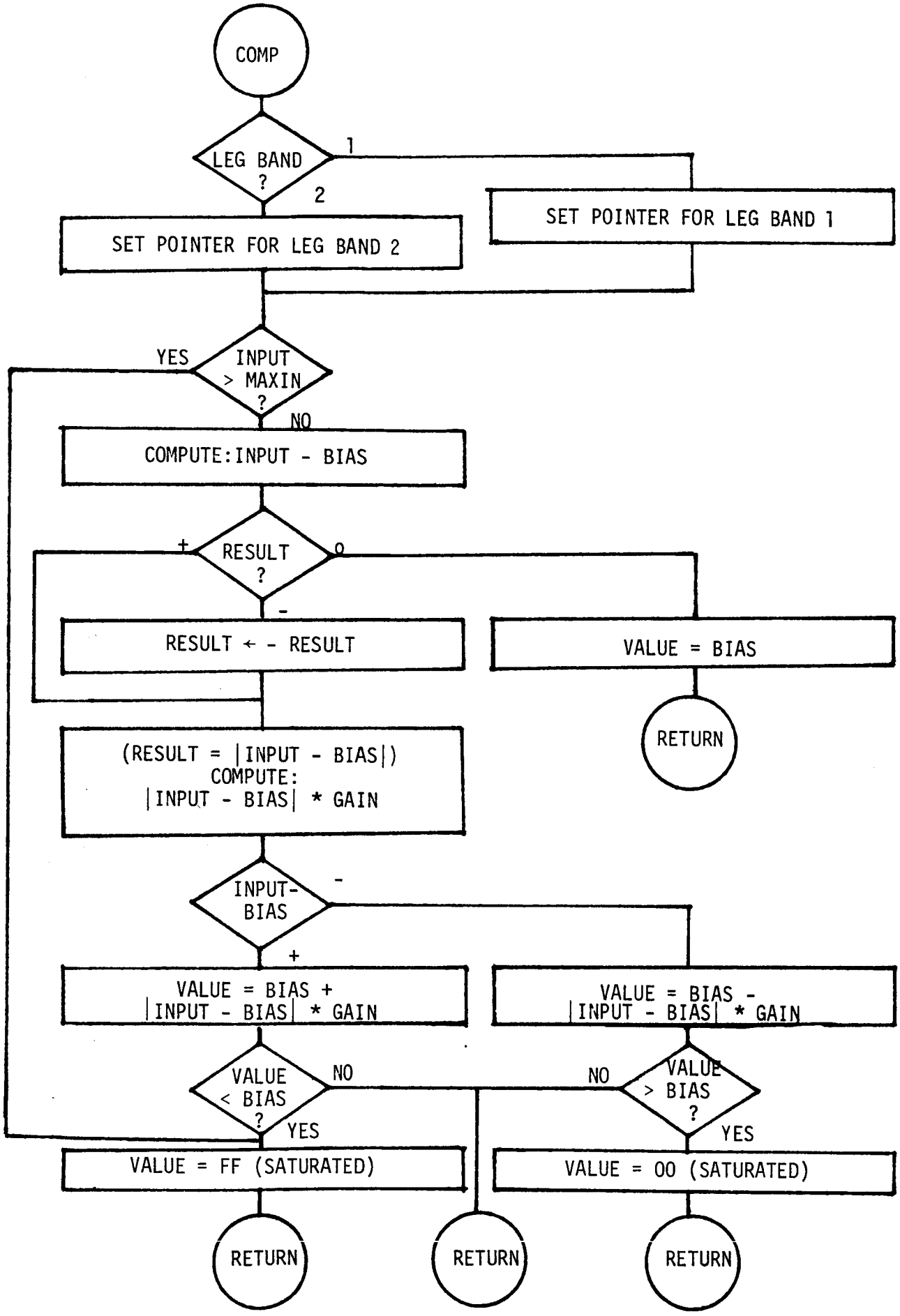
(MAINTAINS SYSTEM TIME)

(DWELL TIME)


PTIME UNITS SEC ← 9 - PTIM UNITS SEC
PTIME TENS SEC ← 5 - PTIM TENS SEC
PTIME UNITS MIN ← 9 - PTIM UNITS MIN
UNITS

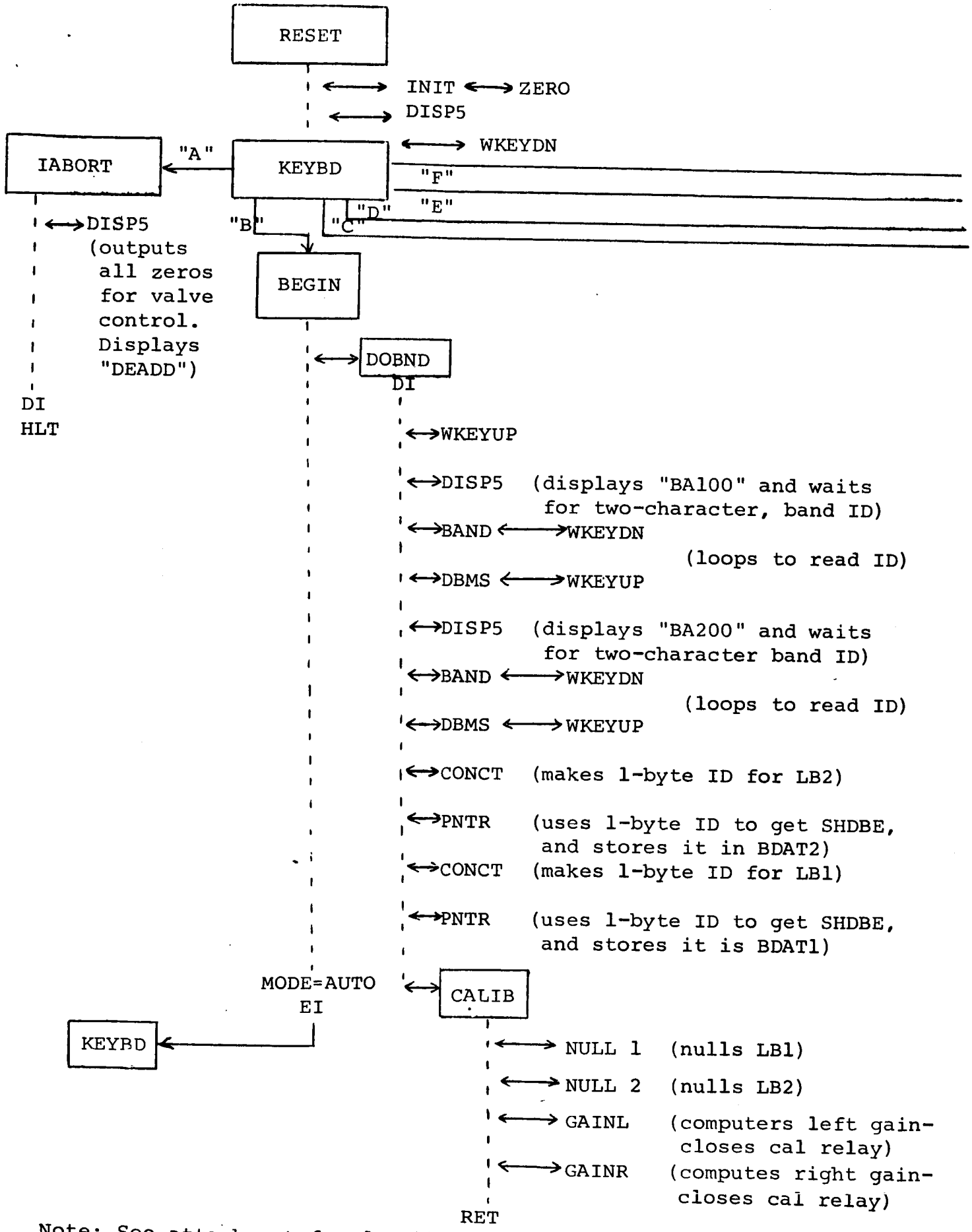


(READS LEG BANDS AND
PROCESSES DATA)

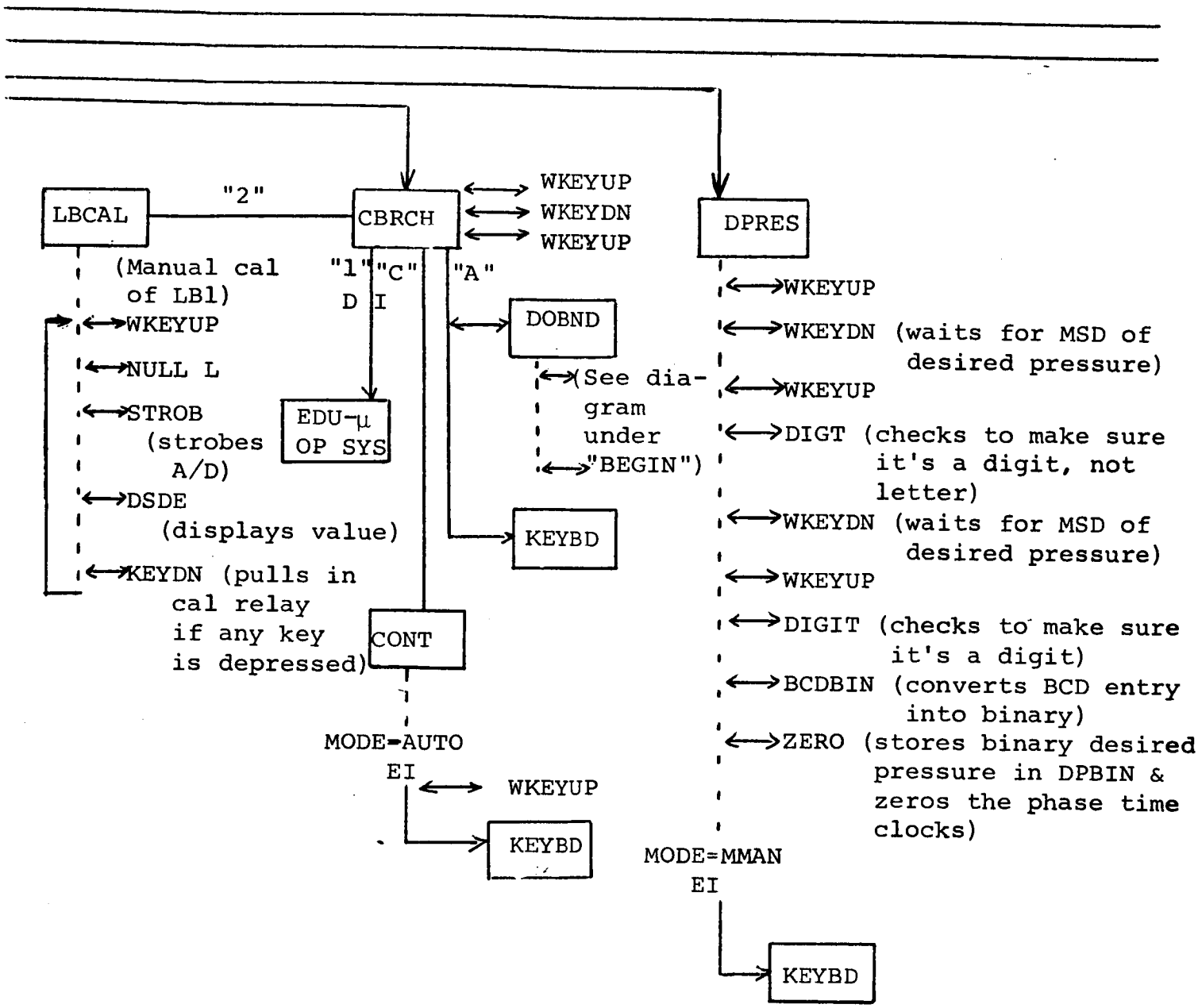

COMP
LEG BAND ?
1
2
SET POINTER FOR LEG BAND 1
SET POINTER FOR LEG BAND 2
YES
INPUT > MAXIN ?
NO
COMPUTE:INPUT - BIAS
+
RESULT ?
0
-
RESULT ← - RESULT
VALUE = BIAS
RETURN
(RESULT = |INPUT - BIAS|)
COMPUTE:
|INPUT - BIAS| * GAIN
INPUT-BIAS
-
+
VALUE = BIAS +
|INPUT - BIAS| * GAIN
VALUE = BIAS -
|INPUT - BIAS| * GAIN
VALUE < BIAS ?
NO
YES
NO
VALUE > BIAS ?
YES
VALUE = FF (SATURATED)
VALUE = 00 (SATURATED)
RETURN
RETURN
RETURN

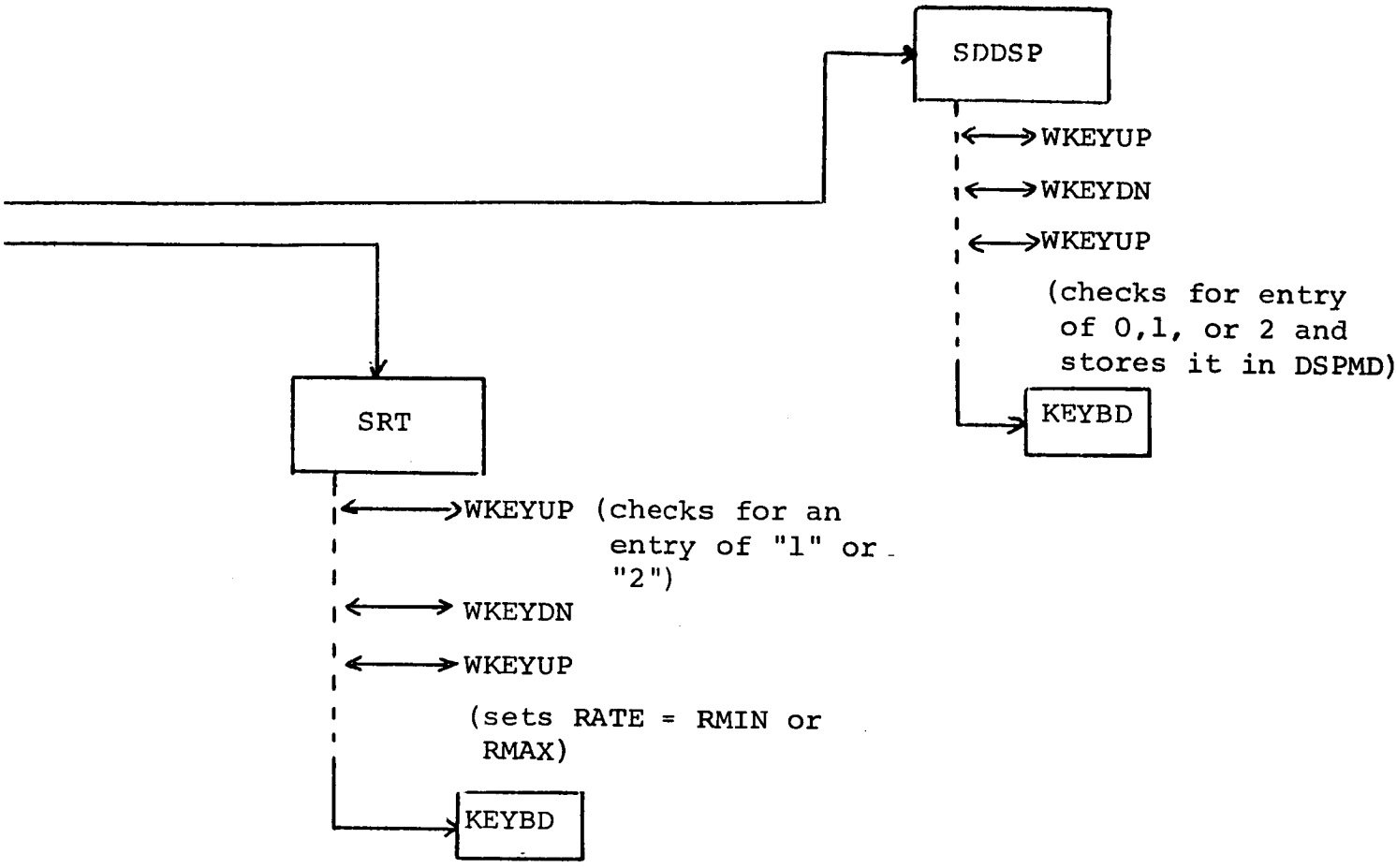
APPENDIX IV

CALLING SEQUENCE



Note: See attachment for detailed "call" sequences under CALIB.


LBCAL
"2"
CBRCH
WKEYUP
WKEYDN
WKEYUP
DPRES
(Manual cal of LB1)
"1"
"C"
"A"
D
I
WKEYUP
NULL L
STROB (strobes A/D)
DSDE (displays value)
KEYDN (pulls in cal relay if any key is depressed)
EDU-μ OP SYS
DOBND
(See diagram under "BEGIN")
KEYBD
CONT
MODE=AUTO
EI
WKEYUP
KEYBD
WKEYUP
WKEYDN (waits for MSD of desired pressure)
WKEYUP
DIGT (checks to make sure it's a digit, not letter)
WKEYDN (waits for MSD of desired pressure)
WKEYUP
DIGIT (checks to make sure it's a digit)
BCDBIN (converts BCD entry into binary)
ZERO (stores binary desired pressure in DPBIN & zeros the phase time clocks)
MODE=MMAN
EI
KEYBD



SUBROUTINES

BAND
BCDBIN
CONCT
DBMS
DELAY
DIGT
DISP5
DSDE
GAINL
GAINR
INIT
KEYDN
NULL 1
NULL 2
PNTR
STROB
WKEYDN
WKEYUP
ZERO

RAM LOCATIONS

LB1O	LB1 OUTPUT BUFFER
LB2O	LB2 OUTPUT BUFFER
BIAS1	LB1 BIAS
	LB1 MAXIN
	LB1 GAIN
BIAS2	LB2 BIAS
	LB2 MAXIN
	LB1 GAIN
BDAT1	LB1 SHDBE
BDAT2	LB2 SHDBE
HOLD1	USED BY "BAND" TO HOLD 2 CHARACTER LB1 ID
HOLD2	USED BY "BAND" TO HOLD 2 CHARACTER LB2 ID
LB1ID	LB1 ID
LB2ID	LB2 ID
BISTA	NBIAS STATUS
DSPMD	DISPLAY MODE

```
CALIB
|
|<-->NULL 1 (nulls LB1)
|      |
|      |<-->DBMS
|      |
|      |<-->STROB (strobes A/D & displays D/A nulling
|      |           values)
|      |
|      |<-->DSDE
|      |
|      |<-->DBMS
|      |
|<-->NULL 2 (nulls LB2)
|      |
|      |<-->DBMS
|      |
|      |<-->STROB (strobes A/D & displays D/A nulling
|      |           values)
|      |
|      |<-->DSDE
|      |
|      |<-->DBMS
|      |
|<-->GAINL (computes left gain-closes cal relay)
|      |
|      |<-->DBMS
|      |
|      |<-->STROBE
|      |
|      |<-->PNTR
|      |
|      |<-->TBYTD
|      |
|      |<-->TBYTD
|      |
|      |<-->TBYTD
|      |
|   OPENS RELAY
|      |
|<-->GAINR (computes right gain-closes cal relay)
|      |
|      |<-->DBMS
|      |
|      |<-->STROB
|      |
|      |<-->PNTR
|      |
|      |<-->TBYTD (divide to get integer part of SHDBE/
|      |           CALVAL)
|      |
|      |<-->TBYTD (divide to get fractional part of
|      |           SHDBE/CALVAL)
|      |
|      |<-->TBYTD (computes max input value)
|      |
|   OPENS RELAY
|
|
|
|
RET
```

INT (decrements VACLK and calls VADJ if countdown complete)

VADJ

READ (reads pressure A/D, compares to DPBIN, and stores necessary value control word in VALUE)

Checks MODE and calls PAUTO if MODE=AUTO and time at correct pressure has elapsed.

PAUTO (gets new pressure and dwell time from table)

ZERO (zeros time words when new pressure is selected)

PTYME (advance DTIME clock advance ATIME clock if MODE= AUTO)

Gets VALUE and outputs value control word

LBRDR

Sets up mask & strobe for LB1

STROB (reads LB1)

COMP

Takes uncorrected LB input and subtracts BIAS

Stores corrected input from LB1 in LB10

2BYTM (gets NUMER to perform gain adjust. Computes ADDTO, puts it in C, adds to original input leaves with corrected input in ACC)

2BYTD

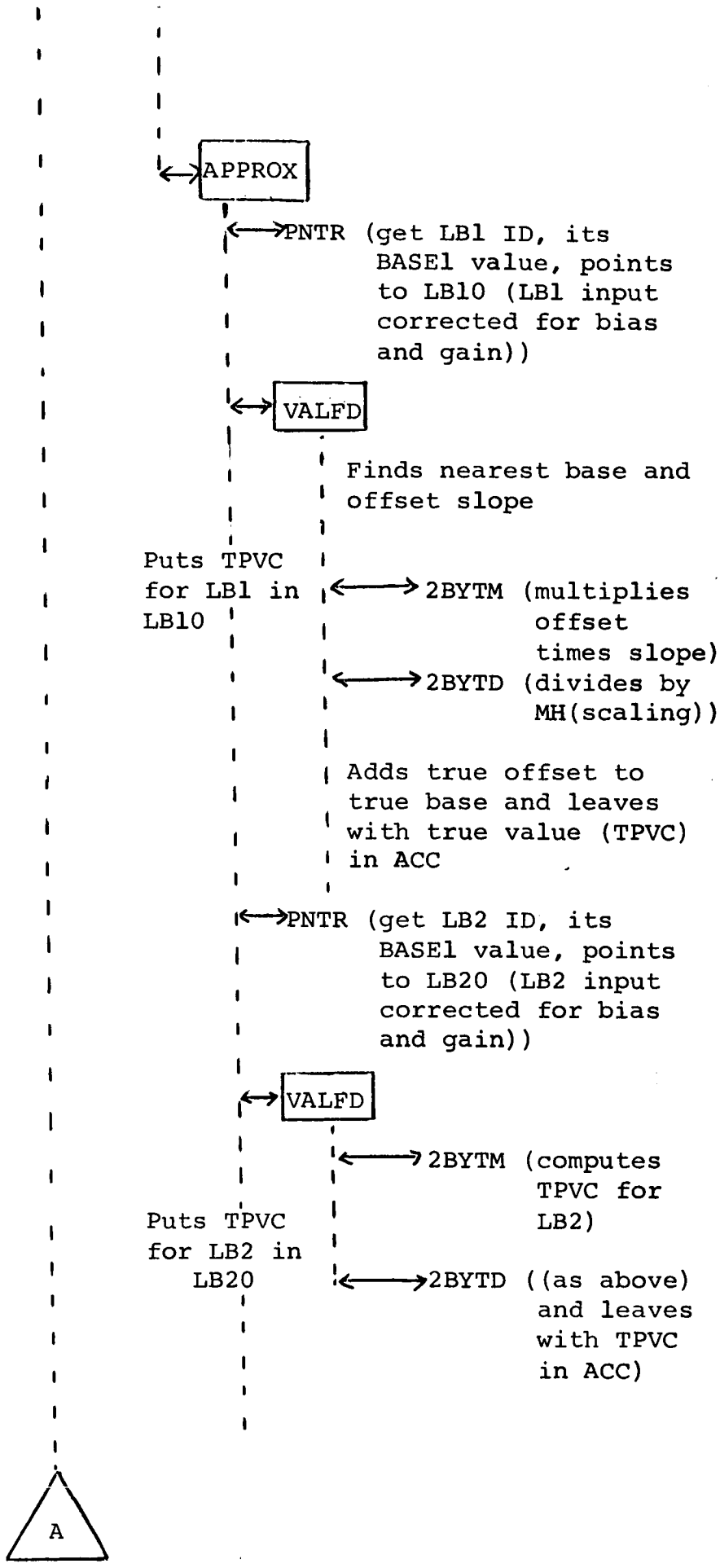
STROB (reads LB2)

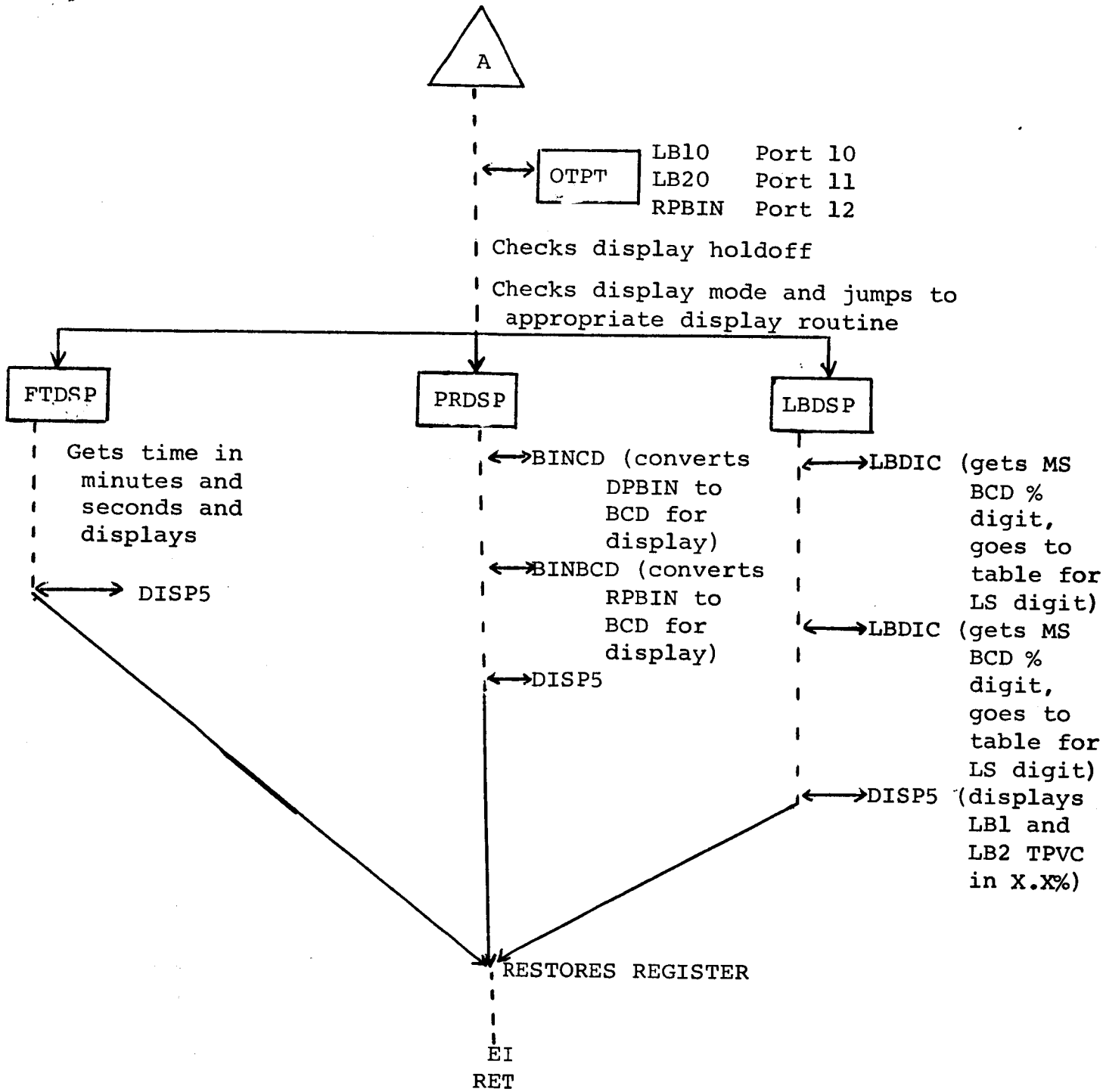
COMP (compensates LB input for BIAS & gain (as above))

Stores corrected input from LB2 in LB20

2BBYTM

2BYTD


APPROX
PNTR (get LB1 ID, its BASE1 value, points to LB10 (LB1 input corrected for bias and gain))
VALFD
Finds nearest base and offset slope
Puts TPVC for LB1 in LB10
2BYTM (multiplies offset times slope)
2BYTD (divides by MH(scaling))
Adds true offset to true base and leaves with true value (TPVC) in ACC
PNTR (get LB2 ID, its BASE1 value, points to LB20 (LB2 input corrected for bias and gain))
VALFD
2BYTM (computes TPVC for LB2)
Puts TPVC for LB2 in LB20
2BYTD ((as above) and leaves with TPVC in ACC)
A


A
OTPT
LB10 Port 10
LB20 Port 11
RPBIN Port 12
Checks display holdoff
Checks display mode and jumps to appropriate display routine
FTDSP
PRDSP
LBDSP
Gets time in minutes and seconds and displays
DISP5
BINCD (converts DPBIN to BCD for display)
BINBCD (converts RPBIN to BCD for display)
DISP5
LBDIC (gets MS BCD % digit, goes to table for LS digit)
LBDIC (gets MS BCD % digit, goes to table for LS digit)
DISP5 (displays LB1 and LB2 TPVC in X.X%)
RESTORES REGISTER
EI
RET

APPENDIX V

ROM LISTINGS

```
                        ;*****************************************************************************
                        ;
                        ;NASA LBNP CHAMBER CONTROLER - WITH PLETHYS CAL AND MONITOR.
                        ;WRITTEN BY USAFA/DFEE/CSL AUG 76 MODIFIED JUL 77
                        ;MODIFIED AT SDSM+T 20 JULY 1979
                        ;MODIFICATIONS BY:
                        ;   M. J. BATCHELDER, PHD
                        ;   R. A. BROWN, BSEE
                        ;
                        ;*****************************************************************************
0000                            ORG     0           ;MONITOR - MODIFIED EDUMICRO
                        ;*****************************************************************************
        E010            BASE    EQU     10E0H
        8000            SETTL   EQU     128
        FF10            MASK1   EQU     10FFH
        FD10            DSPLA   EQU     10FDH
        FE10            DSPMS   EQU     10FEH
        FC10            LDFLG   EQU     10FCH
        F710            BPADR   EQU     10F7H
        FA10            ADDR    EQU     10FAH
        F910            BPTEM   EQU     10F9H
        F510            BPSTR   EQU     10F5H
                        ;*****************************************************************************
                        ;       ROUTINE START AND CMRD
                        ;
                        ;
                        ;       THIS ROUTINE INITIALIZES ALL
                        ;       REQUIRED VARIABLES AND THEN
                        ;       WAITS FOR AN OPERATING SYSTEM
                        ;       COMMAND TO BE ENTERED.  WHEN
                        ;       A VALID COMMAND IS ENTERED, A
                        ;       BRANCH TABLE (LOCATED STARTING AT C BASE)
                        ;       IS USED TO JUMP TO THE APPROPRIATE
                        ;       ROUTINE TO PERFORM THE COMMANDED
                        ;       OPERATION.  THE ROUTINE WHICH
                        ;       PERFORM THE COMMANDS JUMP
                        ;       BACK TO THE MASTER ROUTINE AFTER
                        ;       COMPLETION
                        ;
                        ;       PARAMETERS:         IN:     A KEYCODE IN E REGISTER
                        ;                                   (FROM KEYIN)
                        ;                           OUT:    PC LOADED WITH COMMAND ROUTINE ADDRESS
                        ;       REGISTERS AFFECTED:         ALL
                        ;       MEMORY USED:                MASK1
                        ;       I/O PORTS:                  NONE DIRECTLY
                        ;       OUTPUTS PORTS:              0,1 FOR DISPLAY
                        ;       STACK SPACE:                2 LOCATIONS PRIOR TO
                        ;                                   JUMP TO COMMAND ROUTINES
                        ;                                   USE BY THEM VARIES
                        ;
                        ;       PROGRAMMER:  AL LARSON
                        ;*****************************************************************************
                        ;
0000    31E010          START:  LXI     SP,BASE     ;INITIALIZE THE STACK POINTER
0003    C3B402                  JMP     RESET       ;JUMP TO NASA SYSTEM
0006    00                      NOP
```

```
0007    00                      NOP
0008    C3D000                  JMP     BKPNT       ;RST 1 TO BKPNT
0008    00                      NOP
000C    00                      NOP
000D    31E010      MONITOR:    LXI     SP,BASE     ;INITIALIZE THE STACK POINTER
0010    3E00        CLR:        MVI     A,0
0012    21FF10                  LXI     H,MASK1     ;CLEAR MEMORY SLOTS
0015    D301                    OUT     1
0017    77                      MOV     M,A         ;CLEAR MASK1
0018    2B                      DCX     H
0019    77                      MOV     M,A         ;CLEAR DSPLA
001A    2B                      DCX     H
001B    77                      MOV     M,A
001C    3E00        GETCM:      MVI     A,0         ;CLEAR DISPLAY TO 00000
001E    061F                    MVI     B,01FH
0020    CD1501                  CALL    DISPL
0023    CD2F01      CMRD:       CALL    KEYIN       ;GET COMMAND
0026    0610                    MVI     B,010H
0028    CD1501                  CALL    DISPL
002B    1600                    MVI     D,0         ;CLEAR D
002D    214100                  LXI     H,CBASE     ;GET BRANCH TABLE BASE
0030    19                      DAD     D           ;ADD 2 TIMES E TO BASE
0031    19                      DAD     D
0032    5E                      MOV     E,M         ;GET LS PART OF ADDRESS
0033    23                      INX     H
0034    56                      MOV     D,M         ;GET MS PART
0035    EB                      XCHG                ;PUT IN H,L
0036    E9                      PCHL                ;BRANCH THERE
0037    00                      NOP
                        ;       THE FOLLOWING INSTRUCTION IS EXECUTED BY AN RST 7
                        ;
0038    C30008                  JMP     INT
                        ;
                        ;*******************************************************************
                        ;       ECALL EXECUTES THE CALL COMMAND
                        ;       BY CALL THE ROUTINE WHICH
                        ;       EXECUTES A JUMP.  WHEN RETURN
                        ;       IS MADE, ECALL RESETS TO START
                        ;
                        ;       REGISTERS:          ALL
                        ;       MEMORY USED:        DSPLA,DSPMS,MASK1
                        ;       STACK USED:         10 LEVELS, ALL RESET
                        ;       OUTPUT PORTS:       0,1
                        ;       INPUT PORTS:        0,1
                        ;       PROGRAMMER:         AL LARSON
                        ;
                        ;*******************************************************************
                        ;
003B    CD6800      ECALL:      CALL    EJUMP       ;SAVE PC
003F    C30D00                  JMP     MONITOR     ;RETURN TO MONITOR
                        ;*******************************************************************
                        ;
                        ;       THE FOLLOWING IS THE COMMAND BRANCH TABLE
                        ;
0041    0000        CBASE:      DW      START       ;#0# JUMP TO NASA RESET
0043    BD02                    DW      KEYBD       ;#1# JUMP TO NASA - NO RESET
```

```
0045    0000                DW      MONITOR     ;#2# ILLEGAL COMMAND
0047    0000                DW      MONITOR     ;#3# ILLEGAL COMMAND
0049    0000                DW      MONITOR     ;#4# ILLEGAL COMMAND
004B    C000                DW      SETBP       ;#5# SET BP
004D    6800                DW      EJUMP       ;#6# GO TO
004F    0000                DW      MONITOR     ;#7# ILLEGAL COMMAND
0051    0000                DW      MONITOR     ;#8# ILLEGAL COMMAND
0053    0000                DW      MONITOR     ;#9# ILLEGAL COMMAND
0055    0000                DW      MONITOR     ;#A# ILLEGAL COMMAND
0057    0801                DW      BPCLR       ;#B# CLEAR BREAKPOINT
0059    3800                DW      ECALL       ;#C# CALL
005B    F000                DW      BPDIS       ;#D# DISPLAY BP STACK
005D    7000                DW      EXAMN       ;#E# EXAMINE
005F    6C00                DW      LOADR       ;#F# FILL
                    ;*****************************************************************
                    ;
                    ;       RD2HL READS IN FOUR KEYSTROKES
                    ;       AS AN ADDRESS AND PUTS THEM LEFT TO RIGHT
                    ;       IN HAL IN ORDER ENTERED.
                    ;       ENTRIES ARE ECHOED ON DISPLAY*S
                    ;       FOUR RIGHTMOST DIGITS AND
                    ;       ALSO WIND UP IN MEMORY AT DSPLA
                    ;       AND DSPLA PLUS 1 = DSPMS
                    ;
                    ;       THIS IS A GENERAL ROUTINE AVAILABLE
                    ;       TO THE USER:  JUST SAVE YOUR REGISTER,
                    ;       CALL, KEYIN 4 DIGITS, AND FIND
                    ;       THEM IN HL
                    ;
                    ;       REGISTERS:          ALL
                    ;       MEMORY USED;        DSPLA,DSPMS,MASK1
                    ;       STACK SPACE:        8 LEVELS, ALL RESET
                    ;       INPUT PORTS:        0,1
                    ;       OUTPUT PORTS:       0,1
                    ;       PROGRAMMER:         AL LARSON
                    ;*****************************************************************
                    ;
0061    CDAA01      RD2HL:  CALL    RD2BY
0064    2AFD10              LHLD    DSPLA
0067    C9                  RET
                    ;*****************************************************************
                    ;
                    ;       EJUMP EXECUTES A JUMP TO A
                    ;       SPECEFIED ROUTINE.  IT CALLS
                    ;       RD2HL TO GET AND ECHO
                    ;       THE JMP ADDRESS
                    ;
                    ;       REGISTERS:          ALL
                    ;       MEMORY USED:        MONITOR WORK SPACE
                    ;       STACK SPACE:        8 LEVELS, ALL RESET
                    ;       INPUT PORTS:        0,1
                    ;       OUTPUT PORTS:       0,1
                    ;       PROGRAMMER:         AL LARSON
                    ;
                    ;*****************************************************************
                    ;
```

```
0068    CD6100    EJUMP:  CALL   RD2HL      ;GET ADDRESS
006B    E9                PCHL
                  ;*****************************************************************
                  ;
                  ;       THE FOLLOWING IS A MERGED LOADER
                  ;       AND EXAMINE ROUTINE USED
                  ;       TO EXAMINE AND OR MODIFY
                  ;       ROUTINE USES SUB BLOCKS
                  ;       LABELED FRWD, BKWRD, CHANG
                  ;       TO IMPLEMENT THE AUXILLARY
                  ;       FORWARD, BACKWARD, AND CHANGE
                  ;       COMMANDS.  THE AUXILLARY COMMAND
                  ;       DECODER IS FOUND IN THE
                  ;       BLOCK OF CODE STARTING AT ENTER
                  ;
                  ;
                  ;       REGISTERS:          ALL
                  ;       MEMORY USED:        MONITOR WORK SPACE
                  ;       STACK SPACE:        10 LEVELS, ALL RESET
                  ;       INPUT PORTS:        0,1
                  ;       OUTPUT PORTS:       0,1
                  ;       PROGRAMMER:         AL LARSON
                  ;
                  ;
                  ;       LOADR DIFFERS FROM EXAMINE ONLY IN USE OF
                  ;       A FLAG WORD (LDFLG) WHICH =0 FOR
                  ;       A CHANGE TO THIS LOCATION OR =1 FOR
                  ;       NO CHANGE.  IN LOADER THE FLAG
                  ;       IS SET TO =0 ALL THE TIME.  IN EXAMINE
                  ;       IT IS SET =0 BY ENTERING THE C = CHANGE
                  ;       AUXILLARY COMMAND
                  ;*****************************************************************
                  ;
                  ;       SET UP FLAG WORD
006C    AF        LOADR:  XRA    A           ;SET LOAD FLAG
006D    C37200            JMP    STFG        ;GO STORE IT
0070    3EFF      EXAMN:  MVI    A,FFH       ;CLEAR LOAD FLAG
0072    32FC10    STFG:   STA    LDFLG       ;STORE LOAD FLAG
                  ;       GET ADDRESS FOR START OF EXAMINE OR FILL
0075    CD6100    GETAD:  CALL   RD2HL       ;GET START ADDRESS
0078    22FA10            SHLD   ADDR
007B    2AFA10    GETBY:  LHLD   ADDR        ;FETCH BYTE
                  ;       FETCH AND DISPLAY ADDRESSED BYTE OF MEMORY
007E    5E                MOV    E,M         ;PUT LOWER ADDRESS AND DATA
007F    65                MOV    H,L         ;IN DISPLA
0080    68                MOV    L,E
0081    22FD10            SHLD   DSPLA
0084    CD6601            CALL   DS2BY       ;DISPLAY THEM
0087    3AFC10            LDA    LDFLG       ;CHECK LOAD FLAG
008A    FE00              CPI    00          ;GO CHANGE BYTE
008C    CABD00            JZ     CHANG       ;IF SET
008F    CD2F01    ENTER:  CALL   KEYIN       ;IF NOT, GET COMMAND
                  ;       GET AUXILLARY COMMAND AND DECODE IT
0092    FE0F              CPI    0FH         ;TEST FOR #F# = FORWARD
0094    CAAC00            JZ     FRWRD
0097    FE0B              CPI    0BH         ;#B# = BACKWARD
```

```
0099    CAB600          JZ      BKWRD
009C    FE0C            CPI     0CH         ;#C# = CHANGE
009E    CAPD00          JZ      CHANG
00A1    FE0E            CPI     0EH         ;#E# = EXIT
00A3    CA1000          JZ      CLR
00A6    CD6001          CALL    BADD        ;NO JUMPS - ERROR
00A9    C3BF00          JMP     ENTER
                        ;       MOVE FORWARD IN MEMORY
00AC    2AFA10  FRWRD:  LHLD    ADDR        ;GET ADDRESS
00AF    23              INX     H           ;INCREMENT
00B0    22FA10  COM:    SHLD    ADDR        ;STORE
00B3    C37B00          JMP     GETBY       ;GET NEXT BYTE
                        ;       MOVE BACKWARD IN MEMORY
00B6    2AFA10  BKWRD:  LHLD    ADDR        ;GET ADDRESS
00B9    2B              DCX     H           ;DECREMENT
00BA    C3F000          JMP     COM         ;GO TO COM
                        ;       BEING EXAMINED IF LDFLG=0
00BD    21FD10  CHANG:  LXI     H,DSPLA     ;READ 1 BYTE
00C0    CDBA01          CALL    RD1BY       ;INTO DISPLA
00C3    3AFD10          LDA     DSPLA       ;MOVE BYTE TO M(ADDR)
00C6    2AFA10          LHLD    ADDR
00C9    77              MOV     M,A
00CA    C3BF00          JMP     ENTER
                        ;*******************************************************************
                        ;
                        ;       SETBP SETS UP A SINGLE BREAKPOINT
                        ;       BY WRITING A RST 1 INTO THE
                        ;       ADDRESSED OPCODE SPACE AND
                        ;       SAVES THE BREAKPOINT ADDRESS
                        ;       AND OPCODE FOR RESTORATION
                        ;       BY BPCLR
                        ;
                        ;       REGISTERS:          ALL
                        ;       MEMORY USED:        DSPLA,DSPMS,MASK1,BPADR,BPTEM
                        ;       STACK USED:         10 BYTES
                        ;       INPUT PORTS:        0,1
                        ;       OUTPUT PORTS:       0,1
                        ;       PROGRAMMER:         AL LARSON
                        ;*******************************************************************
                        ;
00CD    CD6100  SETBP:  CALL    RD2HL       ;GET BP ADDRESS
00D0    22F710          SHLD    BPADR       ;SAVE IT
00D3    7E              MOV     A,M         ;GET CONTENTS
00D4    32F910          STA     BPTEM       ;SAVE IN MEMORY
00D7    3ECF            MVI     A,CFH       ;WRITE A RST 1 IN PLACE OF CONTENTS
00D9    77              MOV     M,A
00DA    C30D00          JMP     MONITOR     ;RETURN TO MONITOR
                        ;*******************************************************************
                        ;
                        ;       BKPNT IS THE ROUTINE WHICH IS
                        ;       EXECUTED WHEN A BREAKPOINT IS
                        ;       HIT.  IT IS REACHED VIA
                        ;       FOLLOWING ROUTE.
                        ;         1.  PROGRAM HITS RST 1
                        ;         2.  PROGRAM MAKES CALL TO 0008H
                        ;         3.  JUMP STORED AT 0008H JUMPS TO THIS ROUTINE
```

```
                        ;          4.  THIS ROUTINE SAVES ALL REGISTERS ON CURRENT STACK
                        ;          5.  THIS ROUTINE JUMPS TO MONITOR FOR COMMANDS
                        ;
                        ;        REGISTERS ARE STORED A FLAGS, B, C, D, E,
                        ;        H, L, SP(H), SP(L) IN REVERSE ORDER,
                        ;        THEY ARE LOCATED BY THE ≠D≠ COMMAND
                        ;
                        ;        REGISTERS:          ALL
                        ;        MEMORY USED:        DSPLA,MASK1,BPSTB
                        ;        STACK USED:         14 BYTES
                        ;        INPUT PORTS:        NONE
                        ;        OUTPUT PORTS:       0,1
                        ;        PROGRAMMER:         AL LARSON
                        ;
                        ;*****************************************************************************
                        ;
00DD    F5              BKPNT:  PUSH    PSW         ;SAVE REGISTERS
00DE    C5                      PUSH    B
00DF    D5                      PUSH    D
00E0    E5                      PUSH    H
00E1    210800                  LXI     H,0008H
00E4    39                      DAD     SP
00E5    E5                      PUSH    H
00E6    2B                      DCX     H           ;SAVE LOCATION OF BP STACK BOTTOM
00E7    22F510                  SHLD    BPSTB
00EA    2AF710                  LHLD    BPADR       ;DISPLAY BP ADDRESS
00ED    22FD10                  SHLD    DSPLA       ;IN DIGITS 3-0
00F0    CD6601                  CALL    DS2BY
00F3    3E0B                    MVI     A,0BH       ;DISPLAY ≠B≠
00F5    0610                    MVI     B,10H       ;IN DIGIT 4
00F7    CD1501                  CALL    DISPL
00FA    C32300                  JMP     CMPD        ;GO TO MONITOR FOR COMMANDS
                        ;*****************************************************************************
                        ;
                        ;        BPDIS LOCATES THE BREAKPOINT
                        ;        STACK AND STARTS DISPLAYING IT
                        ;        BY SETTING UP PARAMETERS AND JUMPING
                        ;        INTO THE APPROPRIATE POINT IN THE LOADR
                        ;        THE ≠B≠ AUXILLRY COMMAND IS
                        ;        USED TO DISPLAY THE REGISTERS.
                        ;        NO RESTORE AND PROCEDE FROM
                        ;        THIS BREAKPOINT CAPABILITY IS PROVIDED.
                        ;        USER MUST SET BREAKPOINT, AT THE START OF PROGRAM, HIT BD,
                        ;        EXAMINE REGISTERS (BUT NOT CHANGE THEM).
                        ;        IF REGISTERS ARE OK HE CLEARS
                        ;        THE CURRENT BREAKPOINT WITH ≠B≠
                        ;        COMMAND AND SETS A NEW ONE
                        ;        FURTHER DOWN IN PROGRAM.
                        ;
                        ;        REGISTERS:          A,H,L
                        ;        MEMORY USED:        ADDR,LDFLG
                        ;        STACK SPACE:        NONE
                        ;        INPUT PORTS:        NONE
                        ;        OUTPUT PORTS:       NONE
                        ;        PROGRAMMER:         AL LARSON
                        ;
```

```
                        ;*****************************************************************************
                        ;
00F0    2AF510          BPDIS:   LHLD   BPSTB       ;SET UP EXAMINE
0100    22FA10                   SHLD   ADDR        ;TO LOOK AT BP STACK
0103    3EFF                     MVI    A,FFH       ;CLEAR THE LOAD FLAG
0105    32FC10                   STA    LDFLG
0108    C37B00                   JMP    GETBY
                        ;*****************************************************************************
                        ;
                        ;        BPCLR IS THE ROUTINE WHICH
                        ;        CLEARS THE CURRENT BREAKPOINT
                        ;        RESTORING THE OPCODE ORIGINALLY THERE.
                        ;
                        ;        REGISTERS:          A
                        ;        MEMORY USED:        BPADR
                        ;        STACK SPACE:        NONE
                        ;        INPUT PORTS:        NONE
                        ;        OUTPUT PORTS:       NONE
                        ;        PROGRAMMER:         AL LARSON
                        ;
                        ;*****************************************************************************
                        ;
0108    3AF910          BPCLR:   LDA    BPTEM       ;GET CONTENTS
0105    2AF710                   LHLD   BPADR       ;GET BP ADDRESS
0111    77                       MOV    M,A         ;RESTORE
0112    C30D00                   JMP    MONITOR     ;RETURN TO MONITOR
                        ;*****************************************************************************
                        ;
                        ;        DISPL DISPLAYS A(3:0) IN KEYBOARD DISPLAY LOACATIONS
                        ;        DESIGNATED BY THE B REGISTER.  THE LSB OF B CORRESPONDS TO
                        ;        THE LSB IN THE DISPLAY LIGHTS AND SO ON TO THE LEFT.  A ≠1≠ IN
                        ;        A BIT POSITION CAUSES A(3:0) TO BE DISPLAYED IN THE
                        ;        CORRESPONDING DISPLAY POSITION.  A(3:0) MAY BE DISPLAYED
                        ;        IN UP TO 5 POSITIONS SIMULTANEOUSLY.  A ≠0≠ IN B CAUSES
                        ;        THE CORRESPONDING DISPLAY POSITION TO BE UNCHANGED.
                        ;
                        ;        REGISTERS:          A,C,H,L
                        ;        MEMORY USED:        MASK1
                        ;        STACK SPACE:        NONE
                        ;        INPUT PORTS:        NONE
                        ;        OUTPUT PORTS:       0,1
                        ;        PROGRAMMER:         AL LARSON
                        ;
                        ;*****************************************************************************
                        ;
0115    F6F0            DISPL:   ORI    F0H         ;SET UPPER 4 BITS TO UNBLANK
0117    4F                       MOV    C,A         ;SAVE IN THE C REGISTER
0118    21FF10                   LXI    H,MASK1     ;GET PORT 1 COPY
011B    7E                       MOV    A,M
011C    E6F0                     ANI    F0H         ;ZERO THE LOWER FOUR BITS
011E    B1                       ORA    C           ;MERGE IN THE DATA BITS
011F    D301                     OUT    1           ;SEND TO THE DISPLAY
0121    77                       MOV    M,A         ;SAVE A COPY OF OUTPUT PORT 1 IN MEMORY
0122    3EFF                     MVI    A,FFH         ;SEND THE LOAD COMMAND TO PORT 0
0124    D300                     OUT    0           ;SEND THE ZERO VALUE
0126    78                       MOV    A,B         ;GET THE LOAD WORD
```

```
0127    2F                  CMA                 ;COMPLEMENT IT
0128    D300                OUT     0           ;SEND IT
012A    3EFF                MVI     A,FFH
012C    D300                OUT     0           ;RESEND THE ZERO VALUE
012E    C9                  RET                 ;ENDD OF DISPLAY DRIVER
                    ;*******************************************************************
                    ;
                    ;       KEYIN SENSES 1 KEY DOWN ON KEYBOARD, PERFORMS DEBOUNCE, DECODES
                    ;       KEY, AND PUTS THE HEX VALUE IN REGISTER A.
                    ;
                    ;       REGISTERS:          A,B,C,D,E
                    ;       MEMORY USED:        NONE
                    ;       STACK SPACE:        NONE
                    ;       INPUT PORTS:        0,1
                    ;       OUTPUT PORTS:       NONE
                    ;       PROGRAMMER:         AL LARSON
                    ;
                    ;*******************************************************************
                    ;
012F    DB00        KEYIN:  IN      0           ;READS KEYS 7-0
0131    2F                  CMA                 ;INVERT
0132    4F                  MOV     C,A         ;SAVE A COPY IN C
0133    DB01                IN      1           ;READ KEYS F-8
0135    2F                  CMA                 ;INVERT THEM TOO
0136    47                  MOV     B,A         ;SAVE A COPY IN B
0137    B1                  ORA     C           ;CHECK FOR A KEY DOWN IN 7-0 OR F-8
0138    CA2F01              JZ      KEYIN       ;NONE DOWN SO RESTART
013B    1E00        KEYCD:  MVI     E,0         ;INITIALZE THE HEXCODE
013D    79                  MOV     A,C         ;GET KEYS 7-0
013E    1608        LOOP:   MVI     D,8         ;SET UP D AS A BIT COUNTER
0140    0F          ROTAT:  RRC                 ;PUT NEXT LSB INTO THE CARRY
0141    DA4D01              JC      KEYUP       ;DONE IF THIS BIT IS A ONE-DONE
0144    1C                  INR     E           ;ADVANCE HEXCODE FOR NEXT BIT
0145    15                  DCR     D           ;SEE IF THIS WORD IS DONE
0146    C24001              JNZ     ROTAT       ;NO KEEP SHIFTING FOR THE ONE
0149    78                  MOV     A,B         ;PUT KEYS F-8 IN THE ACC
014A    C33E01              JMP     LOOP        ;LOOK FOR THE ONE IN THE SECOND WORD
014D    1680        KEYUP:  MVI     D,SETTL     ;SET DELAY COUNT
014F    DB00        READU:  IN      0           ;READ KEYS 0-7
0151    47                  MOV     B,A         ;SAVE IN B
0152    DB01                IN      1           ;READ KEYS 8-F
0154    A0                  ANA     B           ;CHECK ALL UP IN 0-7 AND 8-F
0155    FEFF                CPI     FFH
0157    C24D01              JNZ     KEYUP       ;A KEY IS DOWN-RESTART
015A    15                  DCR     D           ;ALL UP SO DECREASE DELAY COUNT ONE
015B    C24F01              JNZ     READU       ;LOOP UNTIL DELAY IS DONE
015E    78                  MOV     A,E
015F    C9                  RET                 ;KEYS UP FOR DELAY PERIOD SO DONE
                    ;*******************************************************************
                    ;
                    ;       THE FOLLOWING ROUTINE DISPLAYS VARIOUS THINGS DEPENDING UPON
                    ;       THE ENTRY POINT
                    ;       BADD:       #BADD#
                    ;       DS2BY:      CONTENTS OF DSPLA AND DSPMS ARE DISPLAYED IN (3:0)
                    ;       DSDE:       CONTENTS OF THE DE REGISTER PAIR ARE DISPLAYED (3:0)
                    ;       DS1BY:      CONTENTS OF REGISTER E IS DISPLAYED IN (1:0)
```

```
                                ;
                                ;       REGISTERS:          ALL
                                ;       MEMORY USED:        DSPLA,DSPMS,MASK1
                                ;       STACK SPACE:        2 BYTES
                                ;       INPUT PORTS:        NONE
                                ;       OUTPUT PORTS:       0,1
                                ;       PROGRAMMER:         AL LARSON
                                ;
                                ;*******************************************************************
                                ;
0160    21DDBA          BADD:   LXI     H,BADDH     ;SET DISPLAY
0163    22FD10                  SHLD    DSPLA       ;TO BAD
0166    2AFD10          DS2BY   LHLD    DSPLA       ;GET LOCATIONS
0169    EB                      XCHG                ;PUT IN D-E PAIR
016A    7A              DSDE:   MOV     A,D         ;DISPLAY DIGIT 2
016B    0604                    MVI     B,4
016D    CD1501                  CALL    DISPL
0170    7A                      MOV     A,D         ;DISPLAY DIGIT 3
0171    0F                      RRC
0172    0F                      RRC
0173    0F                      RRC
0174    0F                      RRC
0175    0608                    MVI     B,8
0177    CD1501                  CALL    DISPL
017A    7B              DS1BY:  MOV     A,E         ;DISPLAY DIGIT 0
017B    0601                    MVI     B,1
017D    CD1501                  CALL    DISPL
0180    7B                      MOV     A,E         ;DISPLAY DIGIT 1
0181    0F                      RRC
0182    0F                      RRC
0183    0F                      RRC
0184    0F                      RRC
0185    0602                    MVI     B,2
0187    C31501                  JMP     DISPL
                                ;*******************************************************************
                                ;
                                ;       RD1BY USES KEYIN TWICE TO READ 2 KEY CHARACTERS AND STORES
                                ;       THEM IN REGISTER E AND IN M(H,L)
                                ;
                                ;       REGISTERS:          A,B,C,D,E
                                ;       MEMORY USED:        M(H,L),MASK1,DSPLA,DSPMS
                                ;       STACK SPACE:        6 BYTES
                                ;       INPUT PORTS:        0,1
                                ;       OUTPUT,PORTS:       0,1
                                ;       PROGRAMMER:         AL LARSON
                                ;
                                ;*******************************************************************
                                ;
018A    E5              RD1BY:  PUSH    H           ;SAVE ADDRESS
018B    CD2F01                  CALL    KEYIN       ;GET MS CHARACTER
018E    0F                      RRC                 ;ROTATE IT TO MS POSITION
018F    0F                      RRC
0190    0F                      RRC
0191    0F                      RRC
0192    5F                      MOV     E,A         ;SAVE IN E
0193    7E                      MOV     A,M         ;GET DISPLAY WORD
```

```
0194    E60F        ANI   0FH      ;INSERT NEW MS CHARACTER
0196    B3          ORA   E
0197    77          MOV   M,A      ;SAVE IN MEMORY
0198    CD6601      CALL  DS2BY    ;DISPLAY
019B    CD2F01      CALL  KEYIN    ;GET LS CHARACTER
019E    E1          POP   H        ;GET ADDRESS
019F    E5          PUSH  H        ;RESAVE ADDRESS
01A0    7E          MOV   A,M      ;GET DISPLAY WORD
01A1    E6F0        ANI   F0H      ;MASK LS CHARACTER
01A3    B3          ORA   E        ;INSERT NEW ONE
01A4    77          MOV   M,A      ;SAVE IN MEMORY
01A5    CD6601      CALL  DS2BY    ;DISPLAY
01A8    E1          POP   H        ;RESTORE H,L
01A9    C9          RET
                ;*****************************************************************
                ;
                ;       RD2BY USES RD1BY TWICE TO READ 4 KEYBOARD CHARACTERS AND STORES
                ;       THEM IN DSPMS AND DSPLA.
                ;
                ;       REGISTERS:          ALL
                ;       MEMORY USED:        MASK1,DSPLA,DSPMS
                ;       STACK SPACE:        8 BYTES
                ;       INPUT PORTS:        0,1
                ;       OUTPUT PORTS:       0,1
                ;       PROGRAMMER:         AL LARSON
                ;
                ;*****************************************************************
                ;
01AA    21FF10  RD2BY:  LXI   H,DSPMS
01AD    CD8A01          CALL  RD1BY
01B0    2B              DCX   H
01B1    CD8A01          CALL  RD1BY
01B4    C9              RET
                ;*****************************************************************
                ;
                ;       RD2A READS 2 CHARACTERS FROM THE KEYBOARD, DISPLAYS THEM AND
                ;       STORES THEM IN REGISTER A
                ;
                ;       REGISTERS:          ALL
                ;       MEMORY USED:        MASK1,DSPLA,DSPMS
                ;       STACK SPACE:        8 BYTES
                ;       INPUT PORTS:        0,1
                ;       OUTPUT PORTS:       0,1
                ;       PROGRAMMER:         AL LARSON
                ;
                ;*****************************************************************
                ;
01B5    21FD10  RD2A:   LXI   H,DSPLA  ;READ IN ONE BYTE FROM KEYBOARD
01B8    CD8A01          CALL  RD1BY    ;AND DISPLAY AND PUT INTO A REGISTER
01BB    3AFD10          LDA   DSPLA
01BE    C9              RET
                ;*****************************************************************
                ;
                ;       DBMS DELAYS #B# MILLISECONDS WHERE B IS THE VALUE IN REGISTER B
                ;
                ;       REGISTERS:          B,C
```

```
                        ;       MEMORY USED:        NONE
                        ;       STACK SPACE:        NONE
                        ;       INPUT PORTS:        NONE
                        ;       OUTPUT PORTS:       NONE
                        ;       PROGRAMMER:         AL LARSON
                        ;
                        ;*******************************************************************
                        ;
01BF    0E3D            DRMS:   MVI     C,3DH
01C1    0D              D1MS:   DCR     C
01C2    C2C101                  JNZ     D1MS
01C5    05                      DCR     B
01C6    C2BF01                  JNZ     DRMS
01C9    C9                      RET
                        ;*******************************************************************
                        ;
                        ;       MOVER MOVES AN N BYTE STRING FROM A SOURCE
                        ;       ARRAY TO A DESTINATION ARRAY.
                        ;       PRIOR TO CALLING SET:
                        ;               C  = NUMBER OF BYTES IN STRING (LESS THAN 256)
                        ;               DE = STARTING ADDRESS OF DESINATION ARRAY
                        ;               HL = STARTING ADDRESS OF SOURCE ARRAY
                        ;
                        ;       REGISTERS:          A,C,D,E,H,L
                        ;       MEMORY USED:        M(H,L)UP
                        ;       STACK SPACE:        NONE
                        ;       INPUT PORTS:        NONE
                        ;       OUTPUT PORTS:       NONE
                        ;       PROGRAMMER:         AL LARSON
                        ;
                        ;*******************************************************************
                        ;
01CA    7E              MOVER:  MOV     A,M         ;GET SOURCE DATA
01CB    12                      STAX    D           ;STORE AT DESTINATION
01CC    13                      INX     D           ;MOVE DESTINATION POINTER
01CD    23                      INX     H           ;MOVE SOURCE POINTER
01CE    0D                      DCR     C           ;COUNT THAT BYTE
01CF    C2CA01                  JNZ     MOVER       ;GO MOVE ANOTHER BYTE
01D2    C9                      RET
```

```
0200                          ORG     0200H       ;NASA LBNPD CONTROLLER WITH PLETHYS CAL
                              ;
                              ;
                              ;****************** SYMBOL EQUATIONS ********************************
                              ;
          2010                MODE    EQU     1020H
          0200                MAUTO   EQU     2           ;AUTOMATIC MODE BIT
          0100                MMAN    EQU     1           ;MANUAL MODE FLAG BIT
          0900                INCTRL  EQU     9           ;CONTROL INPUTS,BIT 0 IS A/D BUSY BIT
          0800                INPRES  EQU     8           ;PRESSURE TRANSDUCER INPUT PORT
          0800                OTCTRL  EQU     8           ;CONTROL OUTPUTS,BIT 0 IS A/D STROBE
          0200                OTVALV  EQU     2           ;VACUUM VALVE CONTROL OUTPUT PORT
          0300                OCTPT   EQU     03H         ;CONTROL WORD OUTPUT PORT(A/D STROBES)
          4A00                LBNBR   EQU     74D         ;NUMBER OF VALID LBID'S
          0400                LB1     EQU     04H         ;RIGHT ADC PORT ADDRESS
          0500                LB2     EQU     05H         ;LEFT ADC PORT ADDRESS
          0100                LB1CT   EQU     01H         ;ALL ZEROS EXCEPT RIGHT ADC STROBE(B0,P3)
          0200                LB2CT   EQU     02H         ;ALL ZEROS EXCEPT LEFT ADC STROBE(B1,P3)
          0600                LB1C    EQU     06H         ;RIGHT COURSE CONTROL DAC
          0400                LB2C    EQU     04H         ;LEFT COURSE CONTROL DAC
          0700                LB1F    EQU     07H         ;RIGHT FINE CONTROL DAC
          0500                LB2F    EQU     05H         ;LEFT FINE CONTROL DAC
          1200                L1OUT   EQU     12H         ;RIGHT OUTPUT PORT ADDRESS
          1100                L2OUT   EQU     11H         ;LEFT OUTPUT PORT ADDRESS
          1000                POUT    EQU     10H         ;PRESSURE OUTPUT PORT ADDRESS
          0F00                LUTB    EQU     0FH         ;LUTAB HIGH ADDRESS
0200        01                RMIN:   DB      01          ;RMIN - LOW RATE (MM HG)/INT SCALED XXXXXX.XX
0201        03                RMAX:   DB      03          ;RMAX - HIGH RATE (MM HG)/INT SCALED XXXXXX.XX
0202        00                VAPER:  DB      0           ;VAPER - VACLK CALLING PERIOD (1/20 SEC)-1
                              ;
                              ;*************************************************************
                              ;THE FOLLOWING LOCATIONS ARE RAM REQUIREMENTS.  THESE LOCATIONS ARE
                              ;INITIALIZED TO ZERO.
                              ;
                              ;THE FOLLOWING LOCATIONS MUST BE SEQUENTIAL.
                              ;
          2110                INTVAR  EQU     1021H       ;
          2110                VACLK   EQU     1021H       ;COUNTDOWN VALUE FOR REP RATE OF VADJ
          2210                ATIM    EQU     1022H       ;AMIN - ATIME(MINUTES)
          2310                ASEC1   EQU     1023H       ;ASEC1 - ATIME(SECONDS,BINARY)
          2410                PTIM    EQU     1024H       ;PSEC1 - PTIME SECONDS(ONES) COUNTDOWN
                              ;                            PSEC2 - PTIME SECONDS(TENS) COUNTDOWN
                              ;                            PMIN1 - PTIME MINUTES(ONES) COUNTDOWN
                              ;
                              ;
          2710                PTIME   EQU     1027H       ;PSECU - PTIME SECONDS(ONES)
                              ;                            PSECT - PTIME SECONDS(TENS)
          2910                PMINU   EQU     1029H       ;PMINU - PTIME MINUTES(ONES)
                              ;
                              ;
          2A10                RPBIN   EQU     102AH       ;REAL PRESSURE (BINARY) MM HG * 4
          2B10                DPBIN   EQU     102BH       ;DESIRED PRESSURE (BINARY) MM HG
          2C10                DPOLD   EQU     102CH       ;LAST VALUE OF DPBIN
          2D10                DPDFL   EQU     102DH       ;DESIRED DIFFERENCE BETWEEN REAL PRESSURE
                              ;                            AND LAST DESIRED PRESSURE (MM HG) X 4
          2E10                INTCT   EQU     102EH       ;INTERUPT COUNTER
```

```
                2F10          ABORTF    EQU    102FH       ;ABORT FLAG
                              ;
                              ;
                3010          OUT1      EQU    1030H       ;OUT1 - LS DIGIT FOR DISPLAY
                3110          OUT2      EQU    1031H       ;OUT2 - SECOND DIGIT FOR DISPLAY
                              ;                            OUT3 - THIRD DIGIT FOR DISPLAY
                              ;                            OUT4 - FOURTH DIGIT FOR DISPLAY
                3410          OUT5      EQU    1034H       ;OUT5 - MS DIGIT FOR DISPLAY
                              ;
                              ;
                3510          VALVE     EQU    1035H       ;VALVE - VACUUM VALVE OUTPUT CONTROL WORD
                3610          TABEN     EQU    1036H       ;TABEN - AUTOMATIC TABLE INDEX
                3710          DISHO     EQU    1037H       ;DISHO - DISPLAY HOLDOFF WORD
                3810          SECTR     EQU    1038H       ;SECTR - SECOND COUNTDOWN USED BY INT
                3910          RATE      EQU    1039H       ;RATE - RATE SELECT (EQUALS RMIN OR RMAX)
                              ;
                              ;*****************************************************
                              ;
                              ;THE FOLLOWING ARE MEMORY REQUIREMENTS FOR THE
                              ;LEG BAND SOFTWARE.
                              ;*****************************************************
                              ;THE FOLLOWING MEMORY IS SEQUENTIAL RAM
                              ;
                4010          LB10      EQU    1040H       ;LB1 OUTPUT BUFFER
                4110          LB20      EQU    1041H       ;LB2 OUTPUT BUFFER
                4210          BIAS1     EQU    1042H       ;LB1 BIAS
                4310          MAXIN1    EQU    1043H       ;LB1 MAX INPUT
                4410          GAIN1     EQU    1044H       ;LB1 GAIN ,SCALED 1.XXXXXXX
                4510          BIAS2     EQU    1045H       ;LB2 BIAS
                4610          MAXIN2    EQU    1046H       ;LB2 MAX INPUT
                4710          GAIN2     EQU    1047H       ;LB2 GAIN,SCALED 1.XXXXXX
                4A10          HOLD1     EQU    104AH       ;LB1 PNTR
                4C10          HOLD2     EQU    104CH       ;LB2 PNTR
                5010          BISTA     EQU    1050H       ;NBIAS STATUS
                5110          DSPMD     EQU    1051H       ;DISPLAY MODE
                              ;
                              ;
                              ;
                              ;
                              ;*****************************************************************
                              ;
                              ;
                              ;TBYTM IS A TWO BYTE MULTIPLY ROUTINE. IT IS ENTERED
                              ;WITH THE MULTIPLICAND IN THE D REGISTER AND THE
                              ;MULTIPLIER IN THE C REGISTER. THE RESULT IS A 16 BIT
                              ;NUMBER IN THE BC REGISTER PAIR. ALL REGISTERS AND
                              ;FLAGS ARE AFFECTED EXCEPT THE HL RP. THE PROGRAM IS
                              ;FROM THE INTEL 8080 USERS MANUAL.
                              ;
                              ;
0203          0600            TBYTM:    MVI    B,0H        ;INITIALIZE MOST SIGNIFICANT BYTE OF RESULT
0205          1E09                      MVI    E,9         ;BIT COUNTER
0207          79              MULT0:    MOV    A,C         ;ROTATE LEAST SIGNIFICANT BIT
0208          1F                        RAR                ;OF MULTIPLYER TO CARRY AND
0209          4F                        MOV    C,A         ;SHIFT LOW-ORDER BYTE OF RESULT
020A          1D                        DCR    E           ;
```

```
020B    C8                      RZ                  ;RETURN IF DONE
020C    78                      MOV     A,B         ;
020D    D21102                  JNC     MULT1       ;
0210    82                      ADD     D           ;ADD MULTIPLYER TO HIGH BYTE IF BIT = 1
0211    1F              MULT1:  RAR                 ;CARRY = 0 , SHIFT HIGH BYTE OF RESULT
0212    47                      MOV     B,A         ;
0213    C30702                  JMP     MULT0       ;
                        ;
                        ;
                        ;TBYTD IS A TWO BYTE DIVIDE ROUTINE. IT IS ENTERED
                        ;WITH THE DIVIDEND IN THE BC REGISTER PAIR AND THE
                        ;DIVISOR IN THE D REGISTER. THE 8 BIT QUOTIENT IS
                        ;RETURNED IN THE C REGISTER AND THE REMAINDER IN THE
                        ;B REGISTER. ALL REGISTERS AND FLAGS ARE AFFECTED EXCEPT
                        ;THE HL REGISTER PAIR. THE PROGRAM IS TAKEN FROM THE
                        ;INTEL 8080 USERS MANUAL.
                        ;
                        ;
0216    1E09            TBYTD:  MVI     E,9         ;BIT COUNTER
0218    78                      MOV     A,B         ;
0219    47              DIV0:   MOV     B,A         ;
021A    79                      MOV     A,C         ;ROTATE CARRY INTO C. ROTATE
021B    17                      RAL                 ;NEXT MOST SIGNIFICANT BIT
021C    4F                      MOV     C,A         ;TO CARRY
021D    1D                      DCR     E           ;
021E    CA3202                  JZ      DIV2        ;
0221    78                      MOV     A,B         ;ROTATE MOST SIGNIFICANT
0222    17                      RAL                 ;BIT TO HIGH-ORDER
0223    D22A02                  JNC     DIV1        ;QUOTIENT
0226    92                      SUB     D           ;SUBTRACT DIVISOR AND LOOP
0227    C31902                  JMP     DIV0        ;
022A    92              DIV1:   SUB     D           ;SUBTRACT DIVISOR. IF LESS THAN
022B    D21902                  JNC     DIV0        ;HIGH-ORDER QUOTIENT, LOOP
022E    82                      ADD     D           ;ELSE, ADD IT BACK
022F    C31902                  JMP     DIV0        ;
0232    17              DIV2:   RAL                 ;
0233    5F                      MOV     E,A         ;
0234    3EFF                    MVI     A,0FFH      ;COMPLEMENT THE QUOTIENT
0236    A9                      XRA     C           ;
0237    4F                      MOV     C,A         ;
0238    7B                      MOV     A,E         ;
0239    1F                      RAR                 ;
023A    C9                      RET                 ;DONE
                        ;
                        ;
                        ;BCDBIN CONVERTS 2 DIGIT BCD NUMBER TO BINARY.  REGISTER B
                        ;CONTAINS THE MSD OF BCD AND REGISTER C CONTAINS THE LSD.
                        ;THE RESULT IS RETURNED IN REGISTER A.
                        ;
                        ;
023B    04              BCDBIN: INR     B           ;INCREMENT NUM OF 10*S
023C    3E00                    MVI     A,0         ;ZERO NUM OF ACCUM 10*S
023E    D5                      PUSH    D           ;SAVE REG D*E
                        ;
023F    05              ACC10:  DCR     B           ;DEC NUM 10*S LEFT
0240    CA4802                  JZ      SETBIN      ;TEST FOR LAST 10
```

```
0243    C60A                ADI   10        ;ADD THE 10
0245    C33F02              JMP   ACC10     ;GET NEXT 10
                    ;
0248    81          SETBIN: ADD   C         ;ADD LSD TO BINARY NUMBER
0249    D1                  POP   D         ;RESTORE D,E REGISTERS
024A    C9                  RET
                    ;
                    ;
                    ;BINBCD CONVERTS A BINARY NUMBER IN REGISTER A TO BCD.
                    ;IF THE NUMBER EXCEEDS 99 THE CARRY BIT IS SET.  OTHERWISE
                    ;THE MSD IS IN REGISTER B AND THE LSD IS IN REGISTER C.
                    ;THE BINARY NUMBER IS LEFT IN REGISTER A.
                    ;
                    ;
024B    37          BINBCD: STC             ;SET ERROR FLAG
024C    F5                  PUSH  PSW       ;SAVE REG A,CONDITION CODES
024D    0600                MVI   B,0       ;CLEAR 10+S COUNTER
                    ;
024F    FE64                CPI   100       ;COMPARE BIN NUM TO 100.
0251    DA5602              JC    CONV      ;IF BIN < 100, DO CONVERSION
0254    F1                  POP   PSW       ;IF BIN NOT < 100, RESTORE REG A
0255    C9                  RET             ;RETURN
                    ;
0256    D60A        CONV:   SUI   10        ;SUBTRACT 10 FROM BIN NUM
0258    FA5F02              JM    SETBCD    ;IF NEGATIVE, ALL 10+S COUNTED
025B    04                  INR   B         ;INCREMENT 10+S COUNTER
025C    C35602              JMP   CONV      ;CONTINUE COUNTING 10+S
                    ;
025F    C60A        SETBCD: ADI   10        ;GET UNITS VALUE IN REG A
0261    4F                  MOV   C,A       ;SET LSD OF BCD IN REG C
0262    F1                  POP   PSW       ;RESTORE BIN NUM IN REG A
0263    3F                  CMC             ;CLEAR ERROR FLAG
0264    C9                  RET             ;RETURN
                    ;
                    ;
                    ;DISP5 USES THE OPSYS UTILITY DISPL TO DISPLAY 5 CHARACTERS.
                    ;THE LOCATION OF THE LEAST SIGNIFICANT CHARACTER (D0)
                    ;IS STORED IN H,L. D1-D4 ARE IN SUCCESSIVELY HIGHER LOCATIONS:
                    ;M(H,L+1)=D1, ETC.
                    ;
                    ;
                    ;
0265    F5          DISP5:  PUSH  PSW       ;SAVE REGISTERS
0266    C5                  PUSH  B         ;SAVE REGISTERS
0267    0601                MVI   B,1H      ;SET UP DO LOAD STROBE
0269    7E          DNXDIG: MOV   A,M       ;GET NEXT DISPLAY DIGIT
026A    E5                  PUSH  H         ;SAVE LOC OF DIGITS
026B    CD1501              CALL  DISPL     ;DISPLAY DIGIT
026E    78                  MOV   A,B       ;PUT LOAD STROBE IN ACCUM
026F    07                  RLC             ;MOVE STROBE TO NEXT DIGIT
0270    FE11                CPI   11H       ;SEE IF ALL DIGITS DISPLAYED
0272    DA7902              JC    DMORE     ;BRANCH FOR MORE DIGITS
0275    E1                  POP   H         ;
0276    C1                  POP   B         ;RESTORE REGISTERS
0277    F1                  POP   PSW
0278    C9                  RET             ;RETURN
```

```
                        ;
0279    E1              DMORE:  POP     H           ;RESTORE DIGIT ADDRESS
027A    23                      INX     H           ;INCREMENT DIGIT ADDRESS
027B    47                      MOV     B,A         ;STORE NEXT DIGIT STROBE
027C    C36902                  JMP     DNXDIG      ;LOOP FOR NEXT DIGIT
                        ;
                        ;
                        ;ZERO INITIALIZES A BLOCK OF MEMORY TO 0.
                        ;THE C REGISTER CONTAINS THE BYTE COUNT AND THE H,L RP
                        ;CONTAINS THE STARTING ADDRESS.
                        ;
                        ;
027F    C5              ZERO:   PUSH    B           ;SAVE ORIG B,C
0280    0600                    MVI     B,0         ;NEED A ZERO TO STORE
0282    70              ZLOOP:  MOV     M,B         ;STORE ZERO TO MEM
0283    23                      INX     H           ;NEXT MEM LOC
0284    0D                      DCR     C           ;DECREMENT COUNT
0285    C28202                  JNZ     ZLOOP       ;IF NOT ZERO, LOOP AGAIN
0288    C1                      POP     B           ;RESTORE B,C
0289    C9                      RET
                        ;
                        ;
                        ;DIGT TESTS A CHARACTER IN A FOR BEING A DECIMAL DIGIT (0-9).
                        ;IF IT IS, RETURNS WITH REGISTERS UNCHANGED
                        ;IF NOT, ALTERNATE RETURN TO KEYBD.
                        ;
                        ;
028A    F5              DIGT:   PUSH    PSW         ;MUST BE ABLE TO RETURN W/REG SAME
028B    D60A                    SUI     10          ;IF A GE 10, RESULT IS POSITIVE
028D    FAA002                  JM      DIGT1       ;OK IF NEGATIVE
0290    F1                      POP     PSW         ;CLEAR A FROM STACK
0291    F1                      POP     PSW         ;CLEAR RETURN ADDRESS FROM STACK
0292    21110E          GOOFY:  LXI     H,GOOF
0295    3E40                    MVI     A,64
0297    323710                  STA     DISHO       ;HOLD OFF NORMAL DISPLAY FOR 3 SEC
029A    CD6502                  CALL    DISP5       ;DISPLAY *GOOOF*
029D    C38D02                  JMP     KEYBD       ;ABNORMAL RETURN
02A0    F1              DIGT1:  POP     PSW         ;RESTORE A
02A1    C9                      RET
                        ;
                        ;
                        ;THE FOLLOWING IS SEQUENTIAL ROM
                        ;
                        ;
02A2    0A              COMCD:  DB      0AH         ;ABORT COMMAND
02A3    0803                    DW      IABORT      ;ABORT ADDRESS (HIGH,LOW)
02A5    0B                      DB      0BH         ;BEGIN COMMAND
02A6    2603                    DW      BEGIN       ;
02A8    0C                      DB      0CH         ;CONTINUE, CALIBRATE, OR RETURN TO OPSYS
02A9    9503                    DW      CBRCH       ;
02AB    0D                      DB      0DH         ;DESIRED PRESSURE COMMAND
02AC    6905                    DW      DPRES       ;
02AE    0E                      DB      0EH         ;SELECT RATE COMMAND
02AF    9B05                    DW      SRT         ;
02B1    0F                      DB      0FH         ;SELECT DISPLAY COMMAND
02B2    E005                    DW      SDDSP       ;
```

```
                        ;
                        ;
02B4    CDEA02          RESET:  CALL    INIT        ;UPON RESET, DO INITIALIZATION
02B7    213010          CLRIT:  LXI     H,OUT1      ;LOADS POINTER TO INITIALIZE
02BA    CD6502                  CALL    DISP5       ;DISPLAY TO #00000#.
                        ;
                        ;
                        ;COMMAND PROCESSOR LOOP.
                        ;THIS CODE EXECUTES WHENEVER AN INTERRUPT HAS NOT BEEN TRIGGERED (BKGD)
                        ;THE KEYBOARD IS READ AND THE APPROPRIATE ROUTINE IS CALLED
                        ;
                        ;
02BD    CD2F01          KEYRD:  CALL    KEYIN       ;GET FOR KEY ENTRY
02C0    323410                  STA     OUT5        ;STORES ENTRY IN MS DIGIT OF DISPLAY
02C3    213010                  LXI     H,OUT1      ;SETS POINTER FOR DISPLAY
02C6    CD6502                  CALL    DISP5       ;ECHOS COMMAND IN MS DISPLAY DIGIT
02C9    3E14                    MVI     A,20D       ;HOLD ECHO FOR
02CB    323710                  STA     DISHO       ;ONE SECOND
02CE    3A3410                  LDA     OUT5        ;GETS ENTRY TO CHECK COMMAND CODE
02D1    21A202                  LXI     H,COMCD     ;POINT TO COMMAND CODES
02D4    0606                    MVI     B,06H       ;COMMAND LOOP COUNTER
02D6    BE              TEST1:  CMP     M           ;COMMAND↓
02D7    CAE402                  JZ      EXIT        ;GO TO COMMAND ROUTINE
02DA    23                      INX     H           ;POINT TO NEXT COMMAND
02DB    23                      INX     H           ;
02DC    23                      INX     H           ;
02DD    05                      DCR     B           ;ANY VALID COMMANDS LEFT↓
02DE    C2D602                  JNZ     TEST1       ;YES-TEST NEXT COMMAND
02E1    C39202                  JMP     GOOFY       ;INVALID COMMAND
02E4    23              EXIT:   INX     H           ;POINT TO LOW ADDRESS
02E5    5E                      MOV     E,M         ;PUT IN D REGISTER
02E6    23                      INX     H           ;POINT TO HIGH ADDRESS
02E7    56                      MOV     D,M         ;PUT IN E REGISTER
02E8    EB                      XCHG                ;PUT ADDRESS IN HL
02E9    E9                      PCHL                ;JUMP TO INSTRUCTION
                        ;
                        ;
                        ;INIT IS CALLED EACH TIME THE KEYBOARD RESET IS HIT
                        ;
02EA    0E19            INIT:   MVI     C,25        ;COUNT OF NUMBER OF LOCATIONS TO ZERO
02EC    212110                  LXI     H,INTVAR    ;START OF BLOCK TO ZERO
02EF    CD7F02                  CALL    ZERO
02F2    3E01                    MVI     A,MMAN      ;SET MODE TO MANUAL
02F4    322010                  STA     MODE
02F7    3A0102                  LDA     RMAX        ;SET RATE=MAX
02FA    323910                  STA     RATE
02FD    3E02                    MVI     A,02H       ;SET DISPLAY TO PRESSURE
02FF    325110                  STA     DSPMD       ;
0302    3E00                    MVI     A,00H       ;SET NBIAS STATUS TO DO
0304    325010                  STA     BISTA       ;
0307    323710                  STA     DISHO       ;ENABLE DISPLAY
030A    C9                      RET                 ;
                        ;
                        ;
                        ;ABORT IS ACTIVATED BY A ↑A↑ COMMAND OR FROM THE READ PRESSURE
                        ;SUBROUTINE IF PRESSURE IS GREATER THAN OR EQUAL TO 60 TORR.
```

```
                        ;OR IF THE END OF THE AUTOMATIC SEQUENCE IS REACHED.
                        ;IABORT CAUSES THE ABORT VALVE TO OPEN AND CLOSES THE REST TO BRING
                        ;THE PRESSURE TO AMBIENT AS RAPIDLY AS POSSIBLE.  THIS ROUTINE
                        ;SETS THE ABORT FLAG WHICH IS RECOGNIZED IN THE FOREGROUND WHERE
                        ;PRESSURE CONTROL IS ABORTED.  RESET IS THE ONLY WAY TO CLEAR
                        ;THE ABORT FLAG
                        ;
                        ;
030B    CD1103          IABORT:  CALL   IABT        ;ENTRY FOR ↑A↑ COMMAND
030E    C3BD02                   JMP    KEYBD
0311    3E01            IABT:    MVI    A,01H       ;ENTRY FOR CALL FROM READ PRESSURE
0313    322E10                   STA    ABORTF
0316    3E00                     MVI    A,0         ;OPEN ABORT VALVE, CLOSE OTHERS
0318    D302                     OUT    OTVALV
031A    3E14                     MVI    A,20        ;DISPLAY ≠DEADD≠ FOR 1 SECOND
031C    323710                   STA    DISHO
031F    210C0E                   LXI    H,DEAD
0322    CD6502                   CALL   DISP5
0325    C9                       RET
                        ;
                        ;
                        ;BEGIN REQUIRES THE LEG BAND IDENTIFIERS TO
                        ;BE ENTERED. IT THEN CALIBRATES THE LEG
                        ;BANDS AND BEGINS AUTO CONTROL OF THE VACUUM
                        ;CHAMBER AND WILL ALSO CAUSE LEG BAND DATA
                        ;TO BE PROCESSED. ALL REGISTERS AND FLAGS
                        ;ARE AFFECTED.
                        ;
                        ;
0326    CD3203          BEGIN:   CALL   DOBND       ;CALLS SUBR TO READ IN LEG BAND ID≠S
0329    3E02                     MVI    A,MAUTO     ;SET MODE TO AUTO
032B    322010                   STA    MODE        ;
032E    FB                       EI                 ;ENABLES INTERRUPT
032F    C3BD02                   JMP    KEYBD       ;RETURNS TO SCAN KEYBOARD
                        ;
                        ;
0332    F3              DOBND:   DI                 ;DISABLES INT TO PERMIT SPECIAL DISPLAY
0333    3E0B                     MVI    A,0BH       ;LOAD MSD WITH ≠B≠
0335    0610                     MVI    B,10H
0337    CD1501                   CALL   DISPL
033A    2100A1                   LXI    H,A100H     ;LOAD LB1 DISPLAY
033D    22FD10                   SHLD   DSPLA       ;DISPLAY
0340    CD6601                   CALL   DS2BY
0343    CDB501                   CALL   RD2A        ;GET LB1 ID
0346    CD7A03                   CALL   PNTR        ;GET DATA POINTER
0349    EB                       XCHG               ;PUT IN HL
034A    224A10                   SHLD   HOLD1       ;SAVE IT
034D    1E08                     MVI    E,8         ;LOADS COUNTDOWN FOR DELAY
034F    06FF                     MVI    B,FFH       ;LOADS DELAY COUNTER
0351    CDBF01          XLUPE:   CALL   DBMS        ;CALLS 250 MS DELAY
0354    1D                       DCR    E           ;DECREMENTS COUNTDOWN
0355    C25103                   JNZ    XLUPE       ;HANG IN THERE FOR TWO SECONDS
035B    2100A2                   LXI    H,A200H     ;LOAD LB2 DISPLAY
035E    22FD10                   SHLD   DSPLA       ;DISPLAY
035E    CD6601                   CALL   DS2BY
0361    CDB501                   CALL   RD2A        ;GET LB2 ID
```

```
0364   CD7A03          CALL   PNTR      ;GET DATA POINTER
0367   EB              XCHG             ;PUT IN HL
0368   224C10          SHLD   HOLD2     ;SAVE IT
036B   1E08            MVI    E,8       ;LOADS COUNTDOWN FOR DELAY
036D   06FF            MVI    B,0FFH    ;LOADS DELAY COUNTER
036F   CDBF01  YLUPE:  CALL   DBMS      ;CALLS 250 MS DELAY
0372   1D              DCR    E         ;DECREMENTS COUNTDOWN
0373   C26F03          JNZ    YLUPE     ;HANG IN THERE FOR TWO SECONDS
0376   CDBF03          CALL   CALIB     ;CALIBRATE LEG BANDS
0379   C9              RET              ;
                ;
                ;
                ;PNTR CHANGES THE LEG BAND ID (LBID) TO THE BASE
                ;1 POINTER FOR THAT LEG BAND. IT IS ENTERED WITH
                ;THE LBID IN THE A REGISTER AND LEAVES THE BASE 1
                ;POINTER IN THE DE REGISTER PAIR. IF AN INVALID
                ;LBID IS ATTEMPTED TO BE PROCESSED, BOTH LEG BAND
                ;IDAS MUST BE RE-ENTERED. THIS IS USED PRIMARILY WITH
                ;THE BEGIN SUBROUTINE. ALL REGISTERS AND FLAGS ARE
                ;AFFECTED
                ;
                ;
037A   FE4A     PNTR:  CPI    LBNBR
037C   D29003          JNC    NTVLD     ;IF NOT ASK AGAIN
037F   E6FF            ANI    0FFH      ;SET FLAGS FOR LBID=0
0381   11020A          LXI    D,LBSD+2  ;POINT TO FIRST BASE1
0384   C8              RZ               ;QUIT IF LBID=0
0385   EB      BLOOP: XCHG              ;PUT BASE POINTER IN HL
0386   110E00          LXI    D,14      ;ADD 14 TO GET NEXT
0389   19              DAD    D         ;BASE1 ADDRESS
038A   EB              XCHG             ;PUT NEW POINTER IN DE
038B   3D              DCR    A         ;COUNT DOWN LBID
038C   C8              RZ               ;DONE WHEN ZERO
038D   C38503          JMP    BLOOP     ;ELSE LOOP
0390   33      NTVLD: INX    SP        ;SIMULATED RETURN
0391   33              INX    SP        ;
0392   C32603          JMP    BEGIN     ;RE-ENTER LBID
                ;
                ;
                ;CBRCH IS THE SUBROUTINE THAT WAITS FOR A KEY
                ;ENTRY AFTER A ≠C≠ HAS BEEN ENTERED AND THEN
                ;TRANSFERS CONTROL TO THE PROPER ROUTINE.
                ;AN INVALID ENTRY SENDS CONTROL TO GOOFY.
                ;
                ;
0395   CD2F01  CBRCH: CALL   KEYIN      ;WAIT FOR KEY ENTRY
0398   FE0A            CPI    0AH       ;COMMAND CA↓
039A   CC3203          CZ     DOBND     ;YES-DO CALIBRATE
039D   FE00            CPI    00H       ;RETURN FROM CALIB↓
039F   C2AC03          JNZ    CKCONT    ;YES-CONTINUE KEYBOARD SCAN
03A2   D301            OUT    1         ;OUTPUTS ZEROS TO DISPLAY DATA PORT
03A4   D300            OUT    0         ;STROBES ALL ZEROS INTO DISPLAY
03A6   2F              CMA              ;LOADS ALL ONES INTO STROBES
03A7   D300            OUT    0         ;SETS ALL STROBES TO ≠1≠
03A9   C3BD02          JMP    KEYBD     ;CONTINUE KEYBOARD SCAN
03AC   FE0C    CKCONT: CPI    0CH       ;COMMAND CC↓
```

```
03AE    CA6005                  JZ      CONT        ;YES-DO CONTINUE
03B1    FE02                    CPI     02H         ;COMMAND C2↓
03B3    CA3E05                  JZ      LBCAL       ;YES-GO NULL LB1 FOR MANUAL CALIBRATION
03B6    FE01                    CPI     01H         ;COMMAND C1↓
03B8    C29202                  JNZ     GOOFY       ;NO-DO GOOFY
03BB    F3                      DI                  ;DISABLES INT FOR RETURN TO MONITOR
03BC    C30D00                  JMP     MONITOR     ;RETURNS TO OPERATING SYSTEM
                        ;
                        ;
                        ;CALIB IS THE LEG BAND CALIBRATION SUBROUTINE.
                        ;ALL REGISTERS AND FLAGS ARE AFFECTED.
                        ;
                        ;
03BF    CDCC03          CALIB:  CALL    NULL1       ;NULLS THE FIRST LEG BAND
03C2    CD1F04                  CALL    NULL2       ;NULLS THE SECOND LEG BAND
03C5    CDD704                  CALL    GAINL       ;COMPUTE LEFT GAIN
03C8    CD7004                  CALL    GAINR       ;COMPUTE RIGHT GAIN
03CB    C9                      RET                 ;
                        ;
                        ;
03CC    AF              NULL1:  XRA     A           ;CLEAR A
03CD    D303                    OUT     3           ;INSURE CAL RELAYS OPEN
03CF    5F                      MOV     E,A         ;INITIALIZE FINE DAC COUNT
03D0    57                      MOV     D,A         ;INITIALIZE COURSE DAC COUNT
03D1    0600                    MVI     B,0         ;CLEAR  B
03D3    0E01                    MVI     C,LB1CT     ;SET UP STROBE FOR ADC1
03D5    C5                      PUSH    B           ;SAVE ADC CONTROL ON STACK
03D6    D307                    OUT     LB1F        ;ZERO LB1 FINE DAC
03D8    D306            CR1LP:  OUT     LB1C        ;ZERO LB1 COARSE DAC
03DA    060A                    MVI     B,10        ;DELAY 10 MSEC
03DC    CDBF01                  CALL    DBMS        ;
03DF    C1                      POP     B           ;GET ADC CONTROL WORDS
03E0    C5                      PUSH    B           ;RESAVE FOR NEXT USE
03E1    CDAC06                  CALL    STROB       ;DO A/D CONVERSION
03E4    DB04                    IN      LB1         ;READ IN A/D VALUE
03E6    FE2A                    CPI     2AH         ;SEE IF OVER NULL VALUE
03E8    D2F303                  JNC     FNE1        ;GO DO FINE DAC IF SO
03EB    14                      INR     D           ;ELSE COUNT UP COURSE DAC
03EC    CD6A01                  CALL    DSDE        ;DISPLAY THE COUNT
03EF    7A                      MOV     A,D         ;PUT DAC COUNT IN ACC
03F0    C30803                  JMP     CR1LP       ;AND LOOP AGAIN
03F3    DB04            FNE1:   IN      LB1         ;READ IN ADC VALUE
03F5    FE2B                    CPI     2BH         ;SEE IF AT NULL VALUE
03F7    DA0E04                  JC      DONE1       ;GET LB1 INPUT
03FA    7B                      MOV     A,E         ;PUT FINE COUNT IN ACC
03FB    D307                    OUT     LB1F        ;AND OUTPUT IT
03FD    0632                    MVI     B,50        ;DELAY 50 MSEC
03FF    CDBF01                  CALL    DBMS        ;
0402    CD6A01                  CALL    DSDE        ;DISPLAY THE DAC COUNTS
0405    1C                      INR     E           ;COUNT UP FINE COUNT
0406    C1                      POP     B           ;GET ADC CONTROL WORDS
0407    C5                      PUSH    B           ;RESAVE IT
0408    CDAC06                  CALL    STROB       ;DO A/D CONVERSION
040B    C3F303                  JMP     FNE1        ;GO TO TOP OF LOOP
040E    C1              DONE1:  POP     B           ;CLEAR ADC CONTROL FROM STACK
040F    214210                  LXI     H,BIAS1     ;POINT TO NULL VALUE LOC
```

```
0412    77                  MOV   M,A       ;STORE THE NULL VALUE
0413    1E08                MVI   E,8       ;DELAY 2 SEC FOR DISPLAY
0415    06FF        ILOOP:  MVI   B,0FFH    ;SET .25 SEC DELAY
0417    CDBF01              CALL  DBMS      ;DO IT
041A    1D                  DCR   E         ;COUNT DOWN LOOP COUNT
041B    C21504              JNZ   ILOOP     ;LOOP 4 TIMES
041E    C9                  RET             ;
                    ;
                    ;
041F    AF          NULL2:  XRA   A         ;CLEAR A
0420    5F                  MOV   E,A       ;INITIALIZE FINE DAC COUNT
0421    57                  MOV   D,A       ;INITIALIZE COURSE DAC COUNT
0422    0600                MVI   B,0       ;CLEAR B
0424    0E02                MVI   C,LB2CT   ;SET UP STROBE FOR ADC2
0426    C5                  PUSH  B         ;SAVE ADC CONTROL ON STACK
0427    D305                OUT   LB2F      ;ZERO LB2 FINE DAC
0429    D304        CR2LP:  OUT   LB2C      ;ZERO LB2 COURSE DAC
042B    060A                MVI   B,10      ;DELAY 10 MSEC
042D    CDBF01              CALL  DBMS      ;
0430    C1                  POP   B         ;GET ADC CONTROL WORDS
0431    C5                  PUSH  B         ;RESAVE FOR NEXT USE
0432    CDAC06              CALL  STROB     ;DO A/D CONVERSION
0435    DB05                IN    LB2       ;READ IN A/D VALUE
0437    FE2A                CPI   2AH       ;SEE IF OVER NULL VALUE
0439    D24404              JNC   FNE2      ;GO DO FINE DAC IF SO
043C    14                  INR   D         ;ELSE COUNT UP COURSE DAC
043D    CD6A01              CALL  DSDE      ;DISPLAY THE COUNT
0440    7A                  MOV   A,D       ;PUT DAC COUNT IN ACC
0441    C32904              JMP   CR2LP     ;AND LOOP AGAIN
0444    DB05        FNE2:   IN    LB2       ;READ IN ADC VALUE
0446    FE2B                CPI   2BH       ;SEE IF AT NULL VALUE
0448    DA5F04              JC    DONE2     ;GET LB2 INPUT
044B    7B                  MOV   A,E       ;PUT FINE COUNT IN ACC
044C    D305                OUT   LB2F      ;AND OUTPUT IT
044E    0632                MVI   B,50      ;DELAY 50 MSEC
0450    CDBF01              CALL  DBMS      ;
0453    CD6A01              CALL  DSDE      ;DISPLAY THE DAC COUNTS
0456    1C                  INR   E         ;COUNT UP FINE COUNT
0457    C1                  POP   B         ;GET ADC CONTROL WORDS
0458    C5                  PUSH  B         ;RESAVE IT
0459    CDAC06              CALL  STROB     ;DO A/D CONVERSION
045C    C34404              JMP   FNE2      ;GO TO TOP OF LOOP
045F    C1          DONE2:  POP   B         ;CLEAR ADC CONTROL FROM STACK
0460    214510              LXI   H,BIAS2   ;POINT NULL VALUE LOC.
0463    77                  MOV   M,A       ;STORE THE NULL VALUE
0464    1E08                MVI   E,8       ;SET UP FOR 2 SEC DELAY
0466    06FF        KLOOP:  MVI   B,0FFH    ;DELAY .25 SECONDS
0468    CDBF01              CALL  DBMS
046B    1D                  DCR   E         ;DO N TIMES
046C    C26604              JNZ   KLOOP
046F    C9                  RET             ;
                    ;
                    ;
                    ;GAINR COMPUTES THE GAIN FOR THE RIGHT (1) LEGBAND
                    ;AND ALSO COMPUTES A MAXIMUM INPUT NUMBER
                    ;
```

```
                        ;
0470    3E40            GAINR:  MVI     A,40H       ;CLOSE RT CAL RELAY
0472    D303                    OUT     3
0474    1E04                    MVI     E,4
0476    06FF                    MVI     B,FFH       ;WAIT 1 SEC FOR SETTLING
0478    CDBF01          GR:     CALL    DRMS
047B    1D                      DCR     E
047C    C27804                  JNZ     GR
047F    014140                  LXI     B,4041H     ;SET A/D STROBE CONTROL WORD
0482    CDAC06                  CALL    STROB       ;COMMAND A/D CONVERSION
0485    DB04                    IN      LB1         ;READ IN THE CAL VALUE
0487    214210                  LXI     H,BIAS1     ;TAKE OUT THE NULL BIAS
048A    96                      SUB     M
048B    57                      MOV     D,A         ;SAVE CAL VALUE AS DIVISOR
048C    D5                      PUSH    D           ;SAVE CALVAL ON THE STACK
048D    2A4A10                  LHLD    HOLD1       ;GET LEGBAND POINTER
0490    EB                      XCHG                ;POINT DE TO BASE 1 VALUE
0491    1B                      DCX     D           ;MOVE IT TO BASE 0
0492    1B                      DCX     D           ;AND THEN TO SHDBE
0493    1A                      LDAX    D           ;GET SHDBE=WHAT CAL VALUE SHOULD BE
0494    214210                  LXI     H,BIAS1     ;TAKE OUT THE NULL BIAS
0497    96                      SUB     M
0498    4F                      MOV     C,A         ;SET UP SHDBE AS DIVIDEND
0499    0600                    MVI     B,0         ;IN BC PAIR
049B    D1                      POP     D           ;GET THE CAL VALUE FROM THE STACK
049C    D5                      PUSH    D           ;AND THEN RESAVE IT
049D    CD1602                  CALL    TRYTD       ;COMPUTE INTEGER PART OF SHDBE/CAL VAL
04A0    79                      MOV     A,C         ;CHECK THAT INTEGER PART IS ONE
04A1    FE01                    CPI     1
04A3    C22505                  JNZ     LBER1       ;BETWEEN 1.000 AND 1.999 AS IT SHOULD BE
04A6    0E00                    MVI     C,0         ;NOW REMAINDER*256 IS IN BC AS DIVIDEND
04A8    D1                      POP     D           ;AND CALVAL IS IN D AS DIVISOR
04A9    CD1602                  CALL    TRYTD       ;COMPUTE FRACTION PART OF GAIN
04AC    79                      MOV     A,C         ;GET GAIN FRACTION IN ACC
04AD    37                      STC                 ;PUT INTEGER PART IN CARRY
04AE    1F                      RAR                 ;RIGHT SHIFT TO GET GAIN AS 1.XXXXXXX
04AF    324410                  STA     GAIN1       ;STORE IT
04B2    1F                      RAR                 ;SCALE GAIN FOR LATER USE
04B3    E67F                    ANI     7FH         ;BY DIVIDING BY 2
04B5    57                      MOV     D,A         ;THEN SAVE AS DIVISOR FOR MAXIN
04B6    3EFF                    MVI     A,255       ;THIS IS MAX POSSIBLE INPUT
04B8    214210                  LXI     H,BIAS1     ;POINT TO NULL VALUE LOC
                        ;
                        ;NOW COMPUTE THE MAXIMUM INPUT VALUE BEFORE SATURATION
                        ;MAXIN= BIAS + (255-BIAS)/GAIN
                        ;
04BB    96              GAINC:  SUB     M           ;SUBTRACT TO GET THE AVAILABLE RANGE
04BC    1F                      RAR                 ;PUT LS BIT INTO THE CARRY
04BD    E67F                    ANI     7FH         ;SET MS BIT TO 0
04BF    47                      MOV     B,A         ;SAVE AS DIVIDEND FOR DIVISION
04C0    3E00                    MVI     A,0         ;PUT LS BIT INTO BIT 7 WITH
04C2    1F                      RAR                 ;REST OF NUMBER ALL 0
04C3    4F                      MOV     C,A         ;BC NOW HAS 255-BIAS *128
04C4    37                      STC                 ;NOW SHIFT BC RIGHT 1 BIT FOR SCALIN
04C5    3F                      CMC                 ;CARRY=0
04C6    78                      MOV     A,B         ;GET MS BYTE
```

```
04C7    1F                  RAR                 ;SHIFT IT. BIT 1 GOES TO CARRY
04C8    47                  MOV     B,A         ;STORE MS BYTE
04C9    79                  MOV     A,C         ;GET LOWER BYTE
04CA    1F                  RAR                 ;SHIFT,FILLING BIT 7 FROM CARRY
04CB    4F                  MOV     C,A         ;THEN STORE THE LS BYTE
04CC    CD1602              CALL    TBYTD       ;DIVIDE BY GAIN
04CF    79                  MOV     A,C         ;GET RESULT AND UNSCALE
04D0    86                  ADD     M           ;ADD BIAS TO GET MAXIN
04D1    23                  INX     H           ;POINT TO MAXIN LOC
04D2    77                  MOV     M,A         ;STORE MAX INPUT
04D3    AF                  XRA     A           ;RELEASE THE CAL RELAYS
04D4    D303                OUT     3
04D6    C9                  RET
                    ;
                    ;
                    ;GAINL COMPUTES THE GAIN FOR THE LEFT (2) LEGBAND
                    ;AND ALSO COMPUTES A MAXIMUM INPUT NUMBER
                    ;
                    ;
04D7    3E80        GAINL:  MVI     A,80H       ;CLOSE LT CAL RELAY
04D9    D303                OUT     3
04DB    1E04                MVI     E,4
04DD    06FF                MVI     B,FFH       ;WAIT 1 SEC FOR SETTLING
04DF    CDBF01      GL:     CALL    DRMS
04E2    1D                  DCR     E
04E3    C2DF04              JNZ     GL
04E6    018280              LXI     B,8082H     ;SET A/D STROBE CONTROL WORD
04E9    CDAC06              CALL    STROB       ;COMMAND A/D CONVERSION
04EC    DB05                IN      LB2         ;READ IN THE CAL VALUE
04EE    214510              LXI     H,BIAS2     ;TAKE OUT THE NULL BIAS
04F1    96                  SUB     M
04F2    57                  MOV     D,A         ;SAVE CAL VALUE AS DIVISOR
04F3    D5                  PUSH    D           ;SAVE CALVAL ON THE STACK
04F4    2A4C10              LHLD    HOLD2       ;GET LEGBAND POINTER
04F7    EB                  XCHG                ;POINT DE TO BASE 1 VALUE
04F8    1B                  DCX     D           ;MOVE IT TO BASE 0
04F9    1B                  DCX     D           ;AND THEN TO SHDBE
04FA    1A                  LDAX    D           ;GET SHDBE=WHAT CAL VALUE SHOULD BE
04FB    214510              LXI     H,BIAS2     ;TAKE OUT THE NULL BIAS
04FE    96                  SUB     M
04FF    4F                  MOV     C,A         ;SET UP SHDBE AS DIVIDEND
0500    0600                MVI     B,0         ;IN BC PAIR
0502    D1                  POP     D           ;GET CAL VAL
0503    D5                  PUSH    D           ;RESAVE IT
0504    CD1602              CALL    TBYTD       ;COMPUTE INTEGER PART OF SHDBE/CALVAL
0507    79                  MOV     A,C         ;CHECK THAT INTEGER PART IS ONE
0508    FE01                CPI     1
050A    C22E05              JNZ     LBER2       ;BETWEEN 1.000 AND 1.999 AS IT SHOULD BE
050D    0E00                MVI     C,0         ;NOW REMAINDER*256 IS IN BC AS DIVIDEND
050F    D1                  POP     D           ;AND CALVAL IS IN D AS DIVISOR
0510    CD1602              CALL    TBYTD       ;COMPUTE FRACTION PART OF GAIN
0513    79                  MOV     A,C         ;GET GAIN FRACTION IN ACC
0514    37                  STC                 ;PUT INTEGER PART IN CARRY
0515    1F                  RAR                 ;RIGHT SHIFT TO GET GAIN AS 1.XXXXXX
0516    324710              STA     GAIN2       ;STORE IT
0519    1F                  RAR                 ;SCALE GAIN FOR LATER USE
```

```
051A    E67F                ANI   7FH        ;BY DIVIDING BY 2
051C    57                  MOV   D,A        ;THEN SAVE AS DIVISOR FOR MAXIN
051D    3EFF                MVI   A,255      ;THIS IS MAX POSSIBLE INPUT
051F    214510              LXI   H,BIAS2    ;POINT TO NULL VALUE LOC
0522    C3BB04              JMP   GAINC      ;GO TO GAINR FOR COMMON FINISH
0525    213405    LBER1:    LXI   H,BND1E    ;POINT TO MESSAGE
0528    CD6502    LECOM:    CALL  DISP5      ;DISPLAY IT
052B    C3BD02              JMP   KEYBD
052F    213905    LBER2:    LXI   H,BND2E    ;ANNOUNCE BAD BAND 2
0531    C32805              JMP   LECOM
0534    01        BND1E:    DB    1
0535    00                  DB    0
0536    0D                  DB    0DH
0537    0A                  DB    0AH
0538    0B                  DB    0BH
0539    02        BND2E:    DB    02H
053A    00                  DB    00H
053B    0D                  DB    0DH
053C    0A                  DB    0AH
053D    0B                  DB    0BH
                  ;
                  ;
                  ;LBCAL NULLS THE RIGHT LEG BAND, THEN CONTINUOUSLY
                  ;READS AND DISPLAYS THE ANALOG INPUT FROM THAT BAND. AFTER NULLING,
                  ;THE OPERATOR MAY PRESS ANY KEYBOARD KEY TO PULL IN THE CAL RELAY AND
                  ;AND READ THE CORRESPONDING ANALOG INPUT FROM THE DISPLAY. THIS CAN
                  ;BE USED TO CALIBRATE LEG BANDS FOR FUTURE USE. THE OPERATOR MAY MOVE
                  ;THE BAND ALONG THE CYLINDERS AND OBSERVE EACH OUTPUT VALUE. THE ONLY
                  ;WAY TO EXIT THE ROUTINE IS TO PUSH THE RESET KEY.
                  ;
                  ;
053F    CDCC03    LBCAL:    CALL  NULL1      ;NULLS LEG BAND AMP
0541    010100    NOCAL:    LXI   B,0001H    ;LOADS A/D STROBE WORD
0544    CDAC06    RDAGN:    CALL  STROB      ;STROBES A/D
0547    DB04                IN    LB1        ;READ LEG BAND INPUT
0549    5F                  MOV   E,A        ;MOVES VALUE TO E FOR DISPLAY
054A    1600                MVI   D,0        ;CLEARS D FOR DISPLAY
054C    CD6A01              CALL  DSDE       ;DISPLAYS VALUE IN TWO LS DIGITS
054F    DB00                IN    0          ;CHECKS TO SEE IF KEY IS DEPRESSED
0551    4F                  MOV   C,A
0552    DB01                IN    1
0554    A1                  ANA   C
0555    FEFF                CPI   FFH
0557    CA4105              JZ    NOCAL      ;IF NOT, RETURNS TO REREAD LEG BAND
055A    014140              LXI   B,4041H    ;IF SO, LOADS STROBE PLUS CAL RELAY
055D    C34405              JMP   RDAGN      ;RETURNS TO READ LEG BAND
                  ;
                  ;
                  ;CONT IS THE CONTINUE SUBROUTINE. THE A REGISTER AND FLAGS
                  ;ARE AFFECTED.
                  ;
                  ;
0560    3E02      CONT:     MVI   A,MAUTO    ;SET MODE TO AUTO
0562    322010              STA   MODE       ;
0565    FB                  EI               ;ENABLE INTERRUPTS
0566    C3BD02              JMP   KEYBD      ;CONTINUE KEYBOARD SCAN
```

```
                        ;
                        ;
                        ;DPRES READS A NEW DESIRED PRESSURE (2 DIGITS)
                        ;IT ALSO IMPLIES MANUAL MODE.
                        ;
                        ;
0569    3E01            DPRES:  MVI     A,MMAN      ;SET MODE TO MANUAL
056B    322010                  STA     MODE
056E    CD2F01                  CALL    KEYIN       ;FIRST CHAR DEPRESSED↓
0571    47                      MOV     B,A         ;SAVE FIRST CHAR FOR BCD-TO-BIN
0572    C5                      PUSH    B           ;
0573    CD8A02                  CALL    DIGT        ;SHOULD BE A DIGIT, NOT A LETTER
0576    CD2F01                  CALL    KEYIN       ;SECOND CHAR DEPRESSED↓
0579    C1                      POP     B           ;RETRIEVES MSD OF BCD INPUT VALUE
057A    4F                      MOV     C,A         ;SAVE CHAR FOR BCD-TO-BIN
057B    CD8A02                  CALL    DIGT        ;SHOULD BE A DIGIT, NOT A LETTER
057E    3A2B10                  LDA     DPBIN       ;GET OLD PRESSURE
0581    322C10                  STA     DPOLD       ;AND SAVE FOR RATE USE.
0584    CD3B02                  CALL    BCDBIN      ;BCD IN B(MSD),C(LSD),  RESULT IN A
0587    322B10                  STA     DPBIN       ;STORE DESIRED PRESSURE
058A    0E03                    MVI     C,3         ;FOR 3 BYTES
058C    212410                  LXI     H,PTIM      ;BEGINNING AT PTIM
058F    CD7F02                  CALL    ZERO        ;ZERO PHASE TIME
0592    3E00                    MVI     A,0
0594    322D10                  STA     DPDEL
0597    FB                      EI                  ;ENABLES INTERRUPT TO BEGIN VALVE CONTROL
0598    C3BD02                  JMP     KEYBD
                        ;
                        ;
                        ;SRT READS A RATE CODE FROM THE CONSOLE AND
                        ;STORES IT IN RAM TO CONTROL THE RATE ADJUST ROUTINE.
                        ;       RATES: 1=5MM/SEC,       3=15 MM/SEC
                        ;
                        ;
059B    CD2F01          SRT:    CALL    KEYIN       ;RATE NUMBER READ TO A
059E    FE01                    CPI     1           ;IF RATE CODE IS A 1
05A0    CAAB05                  JZ      IRAT5       ;GO SET RATE=MIN
05A3    FE02                    CPI     2           ;IF RATE CODE IS 2
05A5    CAB405                  JZ      IRAT6       ;GO SET RATE=MAX
05A8    C39202                  JMP     GOOFY
05AB    3A0002          IRAT5:  LDA     RMIN        ;SET RATE TO MIN
05AE    323910                  STA     RATE
05B1    C3BD02                  JMP     KEYBD
05B4    3A0102          IRAT6:  LDA     RMAX        ;SET RATE TO MAX
05B7    323910                  STA     RATE
05BA    C3BD02                  JMP     KEYBD
                        ;
                        ;
                        ;FTDSP DISPLAYS THE ELAPSED TIME ON THE CURRENT PHASE IN FORMAT:
                        ;<MIN> <SECONDS2> <PRESSURE2>
                        ;
                        ;
05BD    213010          FTDSP:  LXI     H,OUT1      ;HL POINTS AT OUTPUT BUFFER
05C0    112710                  LXI     D,PTIME     ;DE POINTS AT PHASE TIMES (BCD)
05C3    3A2B10                  LDA     DPBIN       ;GETS BINARY DESIRED PRESSURE
05C6    CD4B02                  CALL    BINBCD      ;CONVERTS IT TO TWO DIGIT BCD
```

```
05C9    71                      MOV     M,C         ;STORES LSD IN FIFTH DISPLAY
05CA    23                      INX     H           ;DIGIT, ADVANCES POINTER TO FOURTH
05CB    70                      MOV     M,B         ;DISPLAY DIGIT AND STORES MSD
05CC    23                      INX     H           ;AS FOURTH DISPLAY
05CD    1A                      LDAX    D           ;GET SECONDS, UNITS
05CE    77                      MOV     M,A         ;SAVE AS THIRD DIGIT
05CF    13                      INX     D           ;
05D0    23                      INX     H           ;
05D1    1A                      LDAX    D           ;GET SECONDS, TENS
05D2    77                      MOV     M,A         ;SAVE AS SECOND DIGIT
05D3    13                      INX     D           ;
05D4    23                      INX     H
05D5    1A                      LDAX    D           ;GET MINUTES
05D6    77                      MOV     M,A         ;SAVE AS FIRST DIGIT
05D7    213010                  LXI     H,OUT1      ;POINT AT BUFFER
05DA    CD6502                  CALL    DISP5       ;AND SEND TO DISPLAY
05DD    C36108                  JMP     DONE
                        ;
                        ;
                        ;SDDSP IS THE SUBROUTINE THAT SETS THE DISPLAY MODE.
                        ;ALL REGISTERS AND FLAGS ARE AFFECTED.
                        ;
                        ;
05E0    CD2F01          SDDSP:  CALL    KEYIN       ;GET KEY ENTRY
05E3    FE03                    CPI     03H         ;TEST FOR VALID ENTRY
05E5    D29202                  JNC     GOOFY       ;INVALID ENTRY
05E8    325110                  STA     DSPMD       ;STORE MODE
05EB    C38D02                  JMP     KEYBD       ;CONTINUE KEYBOARD SCAN
                        ;
                        ;
                        ;LBRDP IS A SUBROUTINE THAT WILL READ THE LEG
                        ;BAND INPUTS AND LOAD THE OUTPUT BUFFER. IT ALSO
                        ;ADJUSTS THE INPUTS FOR BIAS AND GAIN. IT AFFECTS ALL
                        ;REGISTERS AND FLAGS EXCEPT THE DE REGISTER PAIR
                        ;
                        ;
05EE    010100          LBRDP:  LXI     B,1         ;SET LB A/D CONTROL
05F1    CDAC06                  CALL    STROB       ;WAIT FOR INPUT
05F4    DB04                    IN      LB1         ;GET LB1 INPUT
05F6    CDBE06                  CALL    COMP        ;COMPENSATE FOR BIAS AND GAIN
05F9    324010                  STA     LB10        ;LOAD OUTPUT BUFFER
05FC    010200                  LXI     B,2         ;SET LB A/D FOR LEFT LB
05FF    CDAC06                  CALL    STROB       ;WAIT FOR INPUT
0602    DB05                    IN      LB2         ;GET LB2 INPUT
0604    CDBE06                  CALL    COMP        ;COMPENSATE FOR BIAS AND GAIN
0607    324110                  STA     LB20        ;LOAD LB2 OUTPUT BUFFER
060A    CD1107                  CALL    APPROX      ;DO APPROXIMATION
060D    C9                      RET                 ;DONE
                        ;
                        ;
                        ;PRDSP IS THE PRESSURE DISPLAY SUBROUTINE.  ALL REGISTERS AND
                        ;FLAGS ARE AFFECTED
                        ;
                        ;
060F    213410          PRDSP:  LXI     H,OUT5      ;SET POINTER TO MS BCD DIGIT
0611    3A2910                  LDA     PMINU       ;GET DIGIT 1 OF MINUTES
```

```
0614   77                  MOV   M,A       ;STORE IN OUT5
0615   3A2A10              LDA   RPBIN     ;GET REAL PRESSURE IN BINARY
0618   1F                  RAR             ;SCALED IN TORR*4 (XXXXXX.XX)
0619   1F                  RAR
061A   E63F                ANI   3FH       ;SCALE IN TORR (00XXXXXX.)
061C   CD4B02              CALL  BINBCD    ;CONVERT TO 2 DIGIT BCD
061F   2B                  DCX   H         ;POINT TO OUT4
0620   70                  MOV   M,B       ;STORE MS DIGIT IN OUT4
0621   2B                  DCX   H         ;POINT TO OUT3
0622   71                  MOV   M,C       ;STORE LS DIGIT IN OUT3
0623   3A2B10              LDA   DPBIN     ;GET BINARY DESIRED PRESSURE
0626   CD4B02              CALL  BINBCD    ;IN TORR. CONVERT TO 2 DIGIT BCD
0629   2B                  DCX   H         ;POINT TO OUT2
062A   70                  MOV   M,B       ;STORE MS DIGIT IN OUT2
062B   2B                  DCX   H         ;POINT TO OUT1
062C   71                  MOV   M,C       ;STORE LS DIGIT IN OUT1
062D   CD6502              CALL  DISP5     ;DISPLAY
0630   C36108              JMP   DONE      ;RETURN TO INT
                    ;
                    ;
                    ;LBDSP IS THE PERCENT CHANGE IN LEG VOLUME
                    ;DISPLAY ROUTINE. THE DISPLAY IS ROUNDED OFF
                    ;TO THE NEAREST .1%. ALL REGISTERS AND FLAGS
                    ;ARE AFFECTED.
                    ;
                    ;
0633   060F         LBDSP:  MVI   B,LUTB    ;POINT TO LOOK UP TABLE
0635   114010              LXI   D,LB10    ;POINT TO LB1 OUTPUT BUFFER
0638   213110              LXI   H,OUT2    ;POINT TO OUT2
063B   1A                  LDAX  D         ;GET LB1 BINARY
063C   4F                  MOV   C,A       ;SAVE LB1 BINARY
063D   CD8506              CALL  LBD1C     ;GET BCD OF LB1,DIGIT 1
0640   77                  MOV   M,A       ;STORE IN OUT2
0641   2B                  DCX   H         ;POINT TO OUT1
0642   0A                  LDAX  B         ;GET LB1 BCD,DIGIT 2
0643   77                  MOV   M,A       ;STORE IN OUT1
0644   13                  INX   D         ;POINT TO LB2 OUTPUT BUFFER
0645   1A                  LDAX  D         ;GET LB2 BINARY
0646   4F                  MOV   C,A       ;SAVE LB1 BINARY IN C
0647   CD8506              CALL  LBD1C     ;GET BCD OF LB2,DIGIT 1
064A   213410              LXI   H,OUT5    ;POINT TO OUT 5
064D   77                  MOV   M,A       ;STORE IN OUT 5
064E   2B                  DCX   H         ;POINT TO OUT 4
064F   0A                  LDAX  B         ;GET LB2 DIGIT 2
0650   77                  MOV   M,A       ;STORE IN OUT 4
0651   1A                  LDAX  D         ;GET LB2 BINARY
0652   FE2A                CPI   2AH       ;LB2 NEGATIVE↓
0654   DA6906              JC    LB2N      ;YES-LB2 NEG
0657   1B           LB1T:   DCX   D         ;POINT TO LB1 BINARY
0658   1A                  LDAX  D         ;GET LB1 BINARY
0659   FE2A                CPI   2AH       ;LB1 NEGATIVE↓
065B   DA6E06              JC    LB1N      ;YES-LB1 NEG
065E   78                  MOV   A,B       ;GET LB2 INDICATOR
065F   FE0D                CPI   0DH       ;LB2 NEG↓
0661   CA7B06              JZ    DUN       ;YES-2 NEG, 1 POS
0664   060A                MVI   B,0AH     ;BOTH POS
```

```
0666    C37B06                  JMP     DUN         ;
0669    060D        LB2N:       MVI     B,0DH       ;LB2 NEGATIVE INDICATOR
066B    C35706                  JMP     LB1T        ;
066E    78          LB1N:       MOV     A,B         ;GET LB2 INDICATOR
066F    FE0D                    CPI     0DH         ;LB2 NEGATIVE↓
0671    CA7906                  JZ      TWONG       ;YES-BOTH NEGATIVE
0674    060B                    MVI     B,0BH       ;1 NEG, 2 POS
0676    C37B06                  JMP     DUN         ;
0679    060E        TWONG:      MVI     B,0EH       ;BOTH NEG INDICATOR
067B    2B          DUN:        DCX     H           ;POINT TO OUT 3
067C    70                      MOV     M,B         ;STORE INDICATOR IN OUT 3
067D    2B                      DCX     H           ;POINT TO OUT 1
067E    2B                      DCX     H           ;
067F    CD6502                  CALL    DISP5       ;DISPLAY
0682    C36108                  JMP     DONE        ;RETURN TO INT
                    ;
                    ;
                    ;LBD1C TAKES A LEG BAND READING IN THE A
                    ;REGISTER AND RETURNS THE MOST SIGNIFICANT
                    ;DIGIT IN THE A REGISTER. ONLY THE A REGISTER
                    ;AND FLAGS ARE AFFECTED.
                    ;
                    ;
0685    C5          LBD1C:      PUSH    B           ;SAVE BC REGISTER PAIR
0686    FE05                    CPI     05H         ;LB BINARY < 05H
0688    DAA706                  JC      L01         ;YES-SET BCD TO 01
068B    0650                    MVI     B,50H       ;FIRST TEST WORD-2
068D    0E00                    MVI     C,00H       ;FIRST RETURN WORD
068F    B8          TEST:       CMP     B           ;LB BINARY < TEST WORD↓
0690    DAA406                  JC      END1        ;YES-DO END
0693    F5                      PUSH    PSW         ;SAVE LB INPUT
0694    78                      MOV     A,B         ;GET TEST WORD
0695    C628                    ADI     40D         ;INCREMENT TO NEXT TEST WORD
0697    47                      MOV     B,A         ;PUT BACK IN TEST REGISTER
0698    0C                      INR     C           ;INCREMENT RETURN WORD
0699    3E05                    MVI     A,05H       ;CHECK FOR MAX VALUE
069B    B9                      CMP     C           ;RETURN WORD MAX↓
069C    CAA306                  JZ      END2        ;YES-TEST DONE
069F    F1                      POP     PSW         ;GET LB INPUT
06A0    C38F06                  JMP     TEST        ;CONTINUE TEST
06A3    F1          END2:       POP     PSW         ;GET LB IDENT
06A4    79          END1:       MOV     A,C         ;LOAD RETURN WORD
06A5    C1                      POP     B           ;RESTORE BC PP
06A6    C9                      RET                 ;DONE
06A7    0E01        L01:        MVI     C,01H       ;RETURN WORD IS 01
06A9    C3A406                  JMP     END1        ;
                    ;
                    ;
                    ;STROB IS A SUBROUTINE THAT STROBES THE LEG BAND
                    ;ADC↑S AND WAITS FOR THE ADC BUSY BIT TO GO LOW.
                    ;IT MUST BE ENTERED WITH THE B REGISTER CONTAINING
                    ;THE PROPER STROBE CONTROL WORD AND THE C REGISTER
                    ;CONTAINING THE PROPER MASK. THE A REGISTER AND
                    ;FLAGS ARE AFFECTED.
                    ;
                    ;
```

```
06AC    78          STROB:  MOV   A,B     ;GET A ZERO
06AD    D303                OUT   OCTPT   ;SEND IT TO ADC PORT
06AF    79                  MOV   A,C     ;GET THE STROBE BIT
06B0    D303                OUT   OCTPT   ;SEND IT
06B2    78                  MOV   A,B     ;GET A ZERO
06B3    D303                OUT   OCTPT   ;OUTPUT IT
06B5    DB06        WAITA:  IN    6       ;WAIT ON THE BUSY
06B7    A1                  ANA   C       ;MASK THE BIT
06B8    E60F                ANI   0FH     ;MASK OFF THE RELAY BITS
06BA    C2B506              JNZ   WAITA   ;WAIT FOR NOT BUSY
06BD    C9                  RET
                    ;
                    ;
                    ;COMP IS THE BIAS AND GAIN COMPENSATION
                    ;SUBROUTINE. IT MUST BE ENTERED WITH THE
                    ;C REGISTER INDICATING WHICH LEG BAND IS BEING
                    ;ADJUSTED. IF THE C REGISTER CONTAINS LB1CT-
                    ;LEG BAND 1 IS SERVICED. ANY OTHER DATA IN C
                    ;WILL CAUSE COMP TO ASSUME LEG BAND 2. THE
                    ;UNCORRECTED INPUT MUST BE IN THE A REGISTER.
                    ;ONLY THE A REGISTER AND FLAGS ARE AFFECTED.
                    ;
                    ;
06BE    E5          COMP:   PUSH  H       ;SAVE REGISTERS
06BF    D5                  PUSH  D       ;
06C0    C5                  PUSH  B       ;
06C1    F5                  PUSH  PSW     ;SAVE LEG BAND INPUT
06C2    79                  MOV   A,C     ;GET LEG BAND IDENT
06C3    FE01                CPI   LB1CT   ;LB1 OR LB2?
06C5    CAE806              JZ    LBA1    ;DO LEG BAND
06C8    214510              LXI   H,BIAS2 ;POINT TO LB2 BIAS
06CB    F1          CMPP:   POP   PSW     ;GET LEG BAND INPUT
06CC    23                  INX   H       ;POINT TO MAXIN
06CD    BE                  CMP   M       ;CHECK FOR MAX INPUT EXCEEDED
06CE    D2F206              JNC   MSAT    ;SATURATE IF IT IS
06D1    2B                  DCX   H       ;POINT TO BIAS
06D2    96                  SUB   M       ;CORRECT INPUT FOR BIAS
06D3    CA0207              JZ    LAB0    ;DONE IF INPUT=BIAS
06D6    F5                  PUSH  PSW     ;SAVE CARRY FLAG
06D7    DAFF06              JC    DONEG   ;NEGATE MAGNITUDE IF .LT. BIAS
06DA    57          CMUL:   MOV   D,A     ;MULTIPLICAND = ABS(INPUT-BIAS)
06DB    23                  INX   H       ;POINT TO MAXIN
06DC    23                  INX   H       ;POINT TO GAIN/2
06DD    4E                  MOV   C,M     ;MULTIPLIER = GAIN/2
06DE    CD0302              CALL  TRYTM   ;BC = (INPUT-BIAS)*GAIN/2
06E1    79                  MOV   A,C     ;GET LSB OF PRODUCT
06E2    07                  RLC           ;PUT MS BIT INTO CARRY
06E3    78                  MOV   A,B     ;GET MS BYTE OF PRODUCT
06E4    17                  RAL           ;A = (INPUT-BIAS)*GAIN
06E5    2B                  DCX   H       ;POINT TO MAXIN
06E6    2B                  DCX   H       ;POINT TO BIAS
06E7    47                  MOV   B,A     ;SAVE THE ADJUSTED INPUT
06E8    F1                  POP   PSW     ;GET THE CARRY FLAG SAVED EARLIER
06E9    DAC307              JC    DOSUB   ;SUBTRACT BIAS IF CARRY SET
06EC    78                  MOV   A,B     ;RECOVER THE ADJUSTED INPUT
06ED    86                  ADD   M       ;ELSE ADD THE BIAS BACK IN
```

```
06EE    BE                      CMP     M           ;SEE IF RESULT < BIAS. INDICATES OVERFLOW
06EF    D2F406                  JNC     LAB         ;IF NC OVERFLOW, DONE
06F2    3EFF            MSAT:   MVI     A,0FFH      ;IF OVF,SATURATE AT 255
06F4    C1              LAB:    POP     B           ;RESTORE REGISTERS
06F5    D1                      POP     D           ;
06F6    E1                      POP     H           ;
06F7    C9                      RET                 ;DONE
06F8    214210          LBA1:   LXI     H,BIAS1     ;POINT TO LB1 BIAS
06FB    C3CB06                  JMP     CMPP        ;DO COMPENSATION
                        ;
06FE    2F              DONEG:  CMA                 ;DO 1+S COMPLEMENT OF INPUT-BIAS
06FF    3C                      INR     A           ;ADD 1 TO GET 2+S COMPLEMENT
0700    C3DA06                  JMP     CMUL        ;NOW GO MULTIPLY
                        ;
0703    7E              DOSUB:  MOV     A,M         ;GET THE BIAS
0704    90                      SUB     B           ;SUBTRACT THE ADJUSTED INPUT
0705    BE                      CMP     M           ;SEE IF GREATER THAN BIAS
0706    DAF406                  JC      LAB         ;IF NOT, THEN DONE
0709    AF                      XRA     A           ;IF SO,SATURATE AT 0
070A    C3F406                  JMP     LAB         ;GO COMPLETE THE COMPENSATION
070D    7E              LAB0:   MOV     A,M         ;GET BIAS AS VALUE
070E    C3F406                  JMP     LAB         ;AND TERMINATE
                        ;
                        ;
                        ;APPROX IS THE PIECE-WISE LINEAR CALIBRATION CURVE
                        ;APPROXIMATOR. IT TAKES THE INDICATED VALUE FROM
                        ;THE LEG BAND OUTPUT BUFFER AND REPLACES IT WITH THE
                        ;TRUE PERCENT VOLUME CHANGE. NO REGISTERS ARE AFFECTED
                        ;EXCEPT THE A REGISTER AND THE FLAGS.
                        ;
                        ;
0711    C5              APPROX: PUSH    B           ;SAVE REGISTERS
0712    D5                      PUSH    D           ;
0713    E5                      PUSH    H           ;
0714    2A4A10                  LHLD    HOLD1       ;GET LB1 POINTER
0717    EB                      XCHG                ;LOAD DE WITH LB1 BASE 1
0718    214C10                  LXI     H,LB1O      ;POINT TO LB1 OUTPUT BUFFER
071B    CD2E07                  CALL    VALFD       ;COMPUTE TRUE VALUE(TPVC)
071E    77                      MOV     M,A         ;LOAD BUFFER WITH TPVC
071F    2A4C10                  LHLD    HOLD2       ;GET LB2 POINTER
0722    EB                      XCHG                ;LOAD DE WITH LB2 BASE 1
0723    214110                  LXI     H,LB2O      ;POINT TO LB2 OUTPUT BUFFER
0726    CD2E07                  CALL    VALFD       ;COMPUTE TPVC
0729    77                      MOV     M,A         ;LOAD BUFFER WITH TPVC
072A    E1                      POP     H           ;RESTORE REGISTERS
072B    D1                      POP     D           ;
072C    C1                      POP     B           ;
072D    C9                      RET                 ;DONE
                        ;
                        ;
                        ;VALFD IS A SUBROUTINE THAT WILL CALCULATE THE TRUE
                        ;PERCENT VOLUME CHANGE, GIVEN THE INDICATED VALUE.
                        ;IT MUST BE ENTERED WITH THE DE RP POINTING AT THE
                        ;BASE 1 VALUE FOR THE LEG BAND BEING USED AND THE
                        ;HL RP POINTING AT THE OUTPUT BUFFER OF THE LEG BAND
                        ;THAT IS TO BE CONVERTED. ALL REGISTERS AND FLAGS ARE
```

```
                        ;AFFECTED EXCEPT THE HL REGISTER PAIR.
                        ;
                        ;
072F   062A             VALFD:  MVI     B,2AH       ;TRUE BASE REGISTER
0730   0E06                     MVI     C,06H       ;SATURATION COUNTER
0732   1B                       DCX     D           ;GET BASED VALUE
0733   1A                       LDAX    D           ;INTO THE ACC
0734   13                       INX     D           ;RESTORE DE TO BE ADR OF BASE1
0735   BE                       CMP     M           ;COMPARE WITH INDICATED
0736   D26107                   JNC     VMIN        ;SET TO MIN VAL IF IND < OR = BASE 0
0739   1A               NBASE:  LDAX    D           ;GET BASE
073A   BE                       CMP     M           ;INDICATED < BASE↓
073B   D24B07                   JNC     TWEEK       ;CALCULATE TPVC
073E   13                       INX     D           ;POINT TO NEXT BASE
073F   13                       INX     D           ;
0740   0D                       DCR     C           ;BASE = BASE 6↓
0741   CA5E07                   JZ      SAT         ;YES - SATURATED
0744   78                       MOV     A,B         ;GET TRUE BASE
0745   C628                     ADI     28H         ;MAKE TRUE BASE NEXT TRUE BASE
0747   47                       MOV     B,A         ;SAVE TRUE BASE
0748   C33907                   JMP     NBASE       ;CHECK NEXT BASE
074B   C5               TWEEK:  PUSH    B           ;SAVE TRUE BASE
074C   96                       SUB     M           ;CALCULATE OFFSET
074D   4F                       MOV     C,A         ;SAVE OFFSET AS MULTIPLYER
074E   13                       INX     D           ;POINT TO SLOPE
074F   1A                       LDAX    D           ;GET SLOPE
0750   57                       MOV     D,A         ;SAVE SLOPE AS MULTIPLICAND
0751   CD0302                   CALL    TRYTM       ;CALCULATE EXCESS TPVC
0754   1619                     MVI     D,19H       ;SCALE CONSTANT
0756   CD1602                   CALL    TRYTD       ;SCALE EXCESS TPVC
0759   51                       MOV     D,C         ;SAVE EXCESS TPVC
075A   C1                       POP     B           ;GET TRUE BASE
075B   78                       MOV     A,B         ;
075C   92                       SUB     D           ;CALCULATE TPVC
075D   C9                       RET                 ;DONE
075E   3EF2             SAT:    MVI     A,0F2H      ;SATURATE AT 5%
0760   C9                       RET                 ;DONE
0761   3E02             VMIN:   MVI     A,2         ;SATURATE AT -1%
0763   C9                       RET
                        ;
                        ;
                        ;OTPT IS THE DATA OUTPUT ROUTINE. ONLY THE A
                        ;REGISTER AND FLAGS ARE AFFECTED.
                        ;
                        ;
0764   3A4010           OTPT:   LDA     LB10        ;GET LB1 OUTPUT
0767   0312                     OUT     L1OUT       ;SEND
0769   3A4110                   LDA     LB20        ;GET LB2 OUTPUT
076C   0311                     OUT     L2OUT       ;SEND
076E   3A2A10                   LDA     RPBIN       ;GET PRESSURE OUTPUT
0771   0310                     OUT     POUT        ;SEND
0773   C9                       RET                 ;DONE
```

```
0800                                ORG     0800H       ;INTERUPT PROCESSOR
                                ;
                                ;INT IS THE MAIN INTERRUPT SERVICE ROUTINE. IT SAVES ALL PROGRAM
                                ;REGISTERS UPON ENTRY AND RESTORES THEM UPON COMPLETION. IT CALLS
                                ;THE VACUUM ADJUST ROUTINE (VADJ) AT AN INTERVAL SPECIFIED BY
                                ;VAPER (TIME IN 1/20 SECONDS). IT CALLS THE AUTOMATIC PRESSURE
                                ;PROFILE ROUTINE (PAUTO) WHENEVER THE AUTOMATIC TIMER ATIME (TIME IN
                                ;MINUTES) INDICATES THAT A NEW PHASE IS TO BE STARTED. IT PROVIDES
                                ;DISPLAY WORDS FOR 1)DESIRED PRESSURE, 2)REAL PRESSURE, AND
                                ;3)TIME AT CURRENT PRESSURE. IT OUTPUTS A CONTROL WORD TO
                                ;OPEN/CLOSE VACUUM VALVES ACCORDING TO DESIRED PRESSURE AND
                                ;PRESSURE CHANGE RATES.
                                ;
                                ;
        0400                    INTPER    EQU     4
        1300                    SECPER    EQU     19
                                ;
                                ;
                                ;THE INTERRUPTS OCCUR AT 100/SEC. INTCT IS USED TO PROCESS ONLY
                                ;20 INT/SEC. VACLK IS USED TO DECIDE HOW OFTEN TO READ AND ADJUST
                                ;THE PRESSURE.  SECTR COUNTS SECONDS TO MAINTAIN THE TIME(BY
                                ;CALLING PTYME).  IF ANY CHANGES ARE MADE IN THE INTERRUPT RATE
                                ;THE FOLLOWING VARIABLES MUST BE ADJUSTED: INPER(WHICH RESETS
                                ;INTCT); VAPER(WHICH RESETS VACLK); SECPER(WHICH RESETS INTCT);
                                ;RMIN AND RMAX (WHICH SET THE RATE CONTROL)
                                ;
                                ;
0800    F5                      INT:      PUSH    PSW
0801    C5                                PUSH    B
0802    D5                                PUSH    D
0803    E5                                PUSH    H           ;SAVES ALL REGISTERS FOR RETRUN TO BACKGROUND
0804    212E10                            LXI     H,INTCT
0807    35                                DCR     M
0808    F26108                            JP      DONE        ;RETURN WITHOUT PROCESSING
080B    3604                              MVI     M,INTPER    ;RESET INTCT TO INT. PERIOD
080D    212110                            LXI     H,VACLK     ;DECREMENTS VACLK AND CHECKS TO SEE
0810    35                                DCR     M           ;IF COUNTDOWN COMPLETE. IF SO VADJ IS
0811    FC6708                            CM      VADJ        ;CALLED BEFORE CONTINUING
0814    23                                INX     H           ;MOVES POINTER TO ATIME TO SEE IF NEW
0815    7E                                MOV     A,M         ;AUTO PHASE IS TO BE STARTED. IF SO
0816    B7                                ORA     A           ;(SET FLAGS)
0817    FCBB08                            CM      PAUTO       ;PAUTO IS CALLED BEFORE CONTINUING
                                ;
                                ;
                                ;IF ONE SECOND HAS PASSED CALL PTYME TO UPDATE THE TIME FOR BOTH
                                ;PHASE TIME (PTIME) AND AUTO TIME (ATIME).
                                ;
                                ;
081A    213810                            LXI     H,SECTR     ;POINT TO SECOND COUNTER
081D    35                                DCR     M           ;DECREMENT IF ONE SECOND HAS PASSED
081E    F22608                            JP      OUTPT       ;NO
0821    3613                              MVI     M,SECPER    ;YES, RESET COUNTER TO SECOND PERIOD AND
0823    CDFA08                            CALL    PTYME       ;CALL PTYME TO UPDATE TIME
0826    3A2F10                  OUTPT:    LDA     ABORTF      ;CHECK ABORT FLAG SET BY #A# CMD
0829    B7                                ORA     A           ;OR READ SUBR IF PRESSURE >= 60 TORR
```

```
082A    CA3708          JZ      VLV         ;NO ABORT IF 0
082D    3E00            MVI     A,0         ;OPEN ABORT VALVE,CLOSE REST
082F    D302            OUT     OTVALV
0831    322B10          STA     DPBIN       ;SET DESIRED PRESSURE TO 0 MM HG
0834    C33E08          JMP     LB
0837    3A3510  VLV:    LDA     VALVE       ;LOADS VACUUM CONTROL WORD FROM VADJ
083A    F610            ORI     10H         ;MAKE CERTAIN ABORT VALVE IS CLOSED
083C    D302            OUT     OTVALV      ;SEND TO VALVE CONTROL PORT
083F    CDEE05  LB:     CALL    LBRDR       ;READ LEG BANDS
0841    CD6407          CALL    OTPT        ;AND PRESSURE
0844    3A3710          LDA     DISHO       ;DISPLAY DISABLED↓
0847    FE00            CPI     00H
0849    CA530B          JZ      DSP         ;DO DISPLAY
084C    3D              DCR     A           ;COUNT DOWN HOLD OFF
084D    323710          STA     DISHO
0850    C36108          JMP     DONE        ;DON↑T DISPLAY
0853    3A5110  DSP:    LDA     DSPMD       ;GET DISPLAY MODE
0856    FE01            CPI     01H         ;DECODE MODE
0858    DA3306          JC      LBDSP       ;DISPLAY LEGBANDS
085B    CA8D05          JZ      FTDSP       ;DISPLAY FAST TIME
085E    C30E06          JMP     PRDSP       ;DISPLAY PRESSURE
0861    E1      DONE:   POP     H           ;RESTORE REGISTERS, ENABLE
0862    D1              POP     D           ;INTERRUPTS, AND RETURN
0863    C1              POP     B           ;TO THE BACKGROUND
0864    F1              POP     PSW
0865    FB              EI
0866    C9              RET
                ;
                ;
                ;VADJ IS THE CONTROL ROUTINE WHCIH GENERATES A VACUUM
                ;VALVE CONTROL WORD (VALVE) TO EITHER 1)MAINTAIN A
                ;CONSTANT DESIRED PRESSURE, OR 2)MAINTAIN A CONSTANT (PRESET)
                ;RATE OF PRESSURE CHANGE WHEN GOING FROM ONE PRESSURE
                ;LEVEL TO ANOTHER.  IT CONTROLS RATE BY MEASURING THE PRESSURE
                ;CHANGE AFTER EACH SUBROUTINE CALLING INTERVAL (VAPER) AND COMPARING
                ;THE ACTUAL CHANGE (FROM THE LAST CONSTANT PRESSURE LEVEL) TO
                ;THE DESIRED CHANGE.  THE DESIRED CHANGE IS A RAMP GENERATED BY
                ;INCREMENTING DPDEL BY RMIN OR RAMX EACH TIME VADJ IS CALLED
                ;(INTERVAL↑VAPER , THE VALVE PERIOD).  WHEN THE RAMP PRESSURE
                ;REACHES THE NEW DESIRED PRESSURE, DPDEL IS SET TO
                ;NEW DESIRED - OLD DESIRED FOR A LEVEL RAMP
                ;ALL CONTROL IS EFFECTED BY OPENING OR CLOSING A SINGLE VALVE.
                ; FIXED LIMIT OF + OR - 0.25 TORR DEFINES THE ACCEPTABLE ERROR
                ;BETWEEN DESIRED AND ACTUAL. IF ACTUAL IS > DESIRED↑0.25 TORR
                ;THE VALVE IS CLOSED; IF ACTUAL IS <DESIRED-0.25 TORR THE VALVE
                ;IS OPENED. VADJ PRESERVES HL
                ;
                ;
0867    E5      VADJ:   PUSH    H           ;SAVE HL
0868    CDA408          CALL    READ        ;READ PRESSURE FROM A/D SCALED XXXXXX.XX
086B    212A10          LXI     H,RPBIN     ;POINT TO REAL PRESSURE BINARY
086E    77              MOV     M,A         ;SAVE FOR OUTPUT TO CHART RECORDER AND DISPLAY
086F    47              MOV     B,A         ;SAVE RPBIN IN B
0870    23              INX     H           ;POINT TO DPBIN SCALED 00XXXXXX.
0871    7E              MOV     A,M
0872    87              ADD     A
```

```
0873   87                  ADD   A          ;SCALE DPBIN TO XXXXXX.00
0874   4F                  MOV   C,A        ;SAVE SCALED DPBIN IN C
0875   23                  INX   H          ;POINT TO DPOLD
0876   7E                  MOV   A,M        ;GET DPOLD SCALED 00XXXXXX.
0877   87                  ADD   A
0878   87                  ADD   A          ;SCALE XXXXXX.00
0879   57                  MOV   D,A        ;AND SAVE IN D
087A   23                  INX   H          ;POINT TO DPDEL
087B   5F                  MOV   E,M        ;SAVE IN E
087C   7A                  MOV   A,D        ;GET OLD DESIRED
087D   83                  ADD   E          ;OLD DESIRED+DESIRED DIFF
087E   90                  SUB   B          ;(OLD DESIRED+DESIRED DIFF)-REAL
087F   CA8E08              JZ    RAMP       ;IF ZERO DON+T CHANGE VALVE
0882   D28908              JNC   FASTR      ;RATE TOO LOW+
0885   AF                  XRA   A          ;NO, CLOSE VALVE
0886   C38B08              JMP   RSET
0889   3E08        FASTR:  MVI   A,08H      ;YES, OPEN VALVE
088B   323510      RSET:   STA   VALVE
088E   3A3910      RAMP:   LDA   RATE       ;RMIN(5 TORR/SEC) OR RMAX(15 TORR/SEC)
0891   83                  ADD   E          ;RATE+DPDEL(GENERATES RAMP,THE DESIRED RATE)
0892   5F                  MOV   E,A        ;PUT NEW DESIRED DIFF IN E
0893   82                  ADD   D          ;CHECK TO SEE IF RAMP > DESIRED
0894   B9                  CMP   C
0895   DA9B08              JC    LEEV       ;NO, FINISHED
0898   79                  MOV   A,C        ;YES, LEVEL OFF AT DESIRED
0899   92                  SUB   D
089A   5F                  MOV   E,A        ;SAVE CALCULATED DPDEL
089B   73          LEEV:   MOV   M,E        ;SAVE NEW DESIRED DIFFERENCE
089C   E1                  POP   H          ;RESTORE HL FOR RETURN
089D   3A0202              LDA   VAPER      ;GET VALVE PERIOD AND
08A0   322110              STA   VACLK      ;RESET COUNTDOWN CLOCK
08A3   C9                  RET
                   ;
                   ;
                   ;READ IS A SUBROUTINE WHICH STROBES THE A/D CONVERTER AND
                   ;READS THE CONVERTED PRESSURE WHEN CONVERSION IS COMPLETE.
                   ;THE PRESSURE IS RETURNED IN A SCALED XXXXXX.XX TORR.
                   ;
                   ;
08A4   3E03        READ:   MVI   A,03H      ;LOAD A/D STROBE AND ABORT DISABLE
08A6   D308                OUT   OTCTRL     ;SEND CONTROL WORD TO OUTPUT PORT
08A8   3E02                MVI   A,02H      ;CLEAR A/D STROBE, LEAVE ABORT DISABLE
08AA   D308                OUT   OTCTRL
08AC   DB09        IMPT:   IN    INCTRL     ;CHECK A/D BUSY BIT
08AE   E601                ANI   01H
08B0   C2AC08              JNZ   IMPT       ;WAIT FOR CONVERSION COMPLETE
08B3   DB08                IN    INPRES     ;READ A/D
08B5   FEF0                CPI   240        ;ABORT IF >60 TORR
08B7   D41103              CNC   IABT
08BA   C9                  RET
                   ;
                   ;
                   ;PAUTO IS THE AUTOMATIC PRESSURE SEQUENCING ROUTINE WHICH RETRIEVES
                   ;A DESIRED PRESSURE AND DESIRED DWELL TIME FROM A TABLE IN PROM.
                   ;PAUTO IS CALLED BY INT WHEN THE DWELL TIME AT THE CURRENT PRES-
                   ;SURE HAS ELAPSED. PAUTO THEN PROVIDES A NEW PRESSURE AND DWELL
```

```
                        ;TIME. WHEN PAUTO SENSES A NEGATIVE VALUE FOR DWELL TIME IN THE
                        ;TABLE, THE AUTOMATIC SEQUENCING IS TERMINATED AND CONTROL JUMPS
                        ;TO THE ABORT SUBROUTINE. PAUTO PRESERVES HL.
                        ;
                        ;
08BB    E5              PAUTO:  PUSH    H           ;SAVES HL FOR RETURN TO INT
08BC    21160E                  LXI     H,TABLE     ;SETS POINTER AT THE START OF TABLE
08BF    3A3610                  LDA     TABEN       ;GETS THE INDEX, ADDS IT TO THE
08C2    0600                    MVI     B,00H       ;STARTING ADDRESS TO GET THE NEW
08C4    4F                      MOV     C,A         ;TABLE POINTER.
08C5    09                      DAD     B           ;SETS HL TO NEW ENTRY
08C6    3A2B10                  LDA     DPBIN       ;GETS CURRENT VALUE OF DPBIN AND
08C9    322C10                  STA     DPOLD       ;MOVES IT TO DPOLD
08CC    7E                      MOV     A,M         ;GETS NEW DESIRED PRESSURE(BINARY)
08CD    322B10                  STA     DPBIN       ;STORES IT IN DPBIN
08D0    23                      INX     H           ;MOVES POINTER TO NEW DWELL TIME
08D1    7E                      MOV     A,M         ;CHECKS TO SEE IF IT IS NEGATIVE.
08D2    B7                      ORA     A           ;
08D3    F2E108                  JP      NOABT
08D6    CD1103                  CALL    IABT        ;IF SO, CALLS THE ABORT ROUTINE.
08D9    3E00                    MVI     A,0         ;REPEAT ABORT IN ONE MINUTE
08DB    322210                  STA     ATIM
08DE    C3F808                  JMP     PEXIT
08E1    322210          NOABT:  STA     ATIM        ;IF NOT, STORES DWELL TIME IN ATIM
08E4    3A3610                  LDA     TABEN       ;GET TABLE INDEX, ADVANCE IT, AND
08E7    C602                    ADI     02H         ;SAVE IT FOR NEXT TABLE ACCESS
08E9    323610                  STA     TABEN       ;
08EC    AF                      XRA     A           ;CLEARS ACCUMULATOR
08ED    322D10                  STA     DPDEL       ;ZEROES DESIRED PRESSURE CHANGE TO INITIAL-
08F0    0E03                    MVI     C,3         ;IZE RATE COMPUTATIONS IN VADJ AND
08F2    212410                  LXI     H,PTIM      ;ZEROES PTIME WHENEVER A NEW DPBIN
08F5    CD7F02                  CALL    ZERO        ;IS SELECTED.
08F8    E1              PEXIT:  POP     H           ;
08F9    C9                      RET                 ;
                        ;
                        ;
                        ;PTYME IS THE TIMER SUBROUTINE WHICH UPDATES THE CLOCK COUNTERS
                        ;FOR PTIME(TIME AT CURRENT DESIRED PRESSURE) AND ATIME(DWELL TIME
                        ;AT EACH PRESSURE LEVEL WHEN IN THE AUTOMATIC MODE). BOTH TIMES
                        ;ARE KEPT AT A ONE SECOND RESOLUTION (I.E. PTIME IS CALLED AT
                        ;ONE SECOND INTERVALS BY INT). PTIME IS A THREE DIGIT VALUE
                        ;STORED AT PTIME - SECONDS(UNITS),SECONDS(TENS), AND MINUTES(UNITS)
                        ;ATIME IS STORED AT ATIM - MINUTES(UNITS),SECONDS(UNITS,BINARY) -
                        ;ATIME IS NOT FOR DISPLAY, IT IS A COUNTDOWN VALUE.
                        ;
                        ;
08FA    212310          PTYME:  LXI     H,ASEC1     ;SETS POINTER AT ATIME SECONDS
08FD    3A2010                  LDA     MODE        ;CHECKS MODE WORD TO SEE IF AUTO MODE
0900    FE01                    CPI     MMAN        ;IS SELECTED.
0902    CA0E09                  JZ      CLOCK       ;IF NOT, GO TO ADVANCE PTIME
0905    35                      DCR     M           ;IF SO, ADVANCE ATIME (DECREMENT ASEC1)
0906    F20E09                  JP      CLOCK       ;CKECK IF ATIME(MINUTES) SHOULD ADVANCE
0909    363B                    MVI     M,3BH       ;IF SO, RESET ASEC1 TO 59,
090B    2B                      DCX     H           ;MOVE POINTER TO ATIME(MINUTES),
090C    35                      DCR     M           ;AND DECREMENT ATIM
090D    23                      INX     H           ;MOVES POINTER TO ASEC1
```

```
                        ;
                        ;
                        ;CLOCK UPDATES THE COUNTDOWN VALUES FOR THE PTIME CLOCK.  THESE
                        ;COUNTDOWN VALUES ARE PSEC1, PSEC2, AND PMIN1, AND ARE STORED AT PTIM.
                        ;
                        ;
090B    23              CLOCK:  INX     H       ;MOVES POINTER TO PSEC1
090C    35                      DCR     M       ;DECREMENTS PSEC1 AND CHECKS TO SEE IF
090D    F22309                  JP      NEWCK   ;COUNTDOWN COMPLETE. IF NOT, GOES TO
0910    3609                    MVI     M,09H   ;UPDATE PTIME. IF SO, RESETS PSEC1 AND
0912    23                      INX     H       ;MOVES POINTER TO PSEC2.
0913    35                      DCR     M       ;DECREMENTS PSEC2 AND CHECKS TO SEE IF
0914    F22309                  JP      NEWCK   ;COUNTDOWN COMPLETE. IF NOT, GOES TO
0917    3605                    MVI     M,05H   ;UPDATE PTIME. IF SO, RESETS PSEC2 AND
0919    23                      INX     H       ;MOVES POINTER TO PMIN1.
091A    35                      DCR     M       ;DECREMENTS PMIN1 AND CHECKS TO SEE IF
091B    F22309                  JP      NEWCK   ;COUNTDOWN COMPLETE. IF NOT, GOES TO
091E    3609                    MVI     M,09H   ;UPDATE PTIME. IF SO, RESETS PMIN1.
                        ;
                        ;
                        ;NEWCK USES THE COUNTDOWN VALUES (PSEC1,PSEC2,PMIN1)
                        ;TO FORM PTIME(PSECU,PSECT,PMINU) FOR DISPLAY.
                        ;
                        ;
0920    012710          NEWCK:  LXI     B,PTIME ;SETS POINTER TO PTIME(PSECU)
0923    212410                  LXI     H,PTIM  ;SETS POINTER TO PTIM(PSEC1)
0926    3E09                    MVI     A,09H   ;
0928    96                      SUB     M       ;COMPUTES 9-PSEC1
0929    02                      STAX    B       ;STORES RESULT IN PSECU
092A    03                      INX     B       ;
092B    23                      INX     H       ;ADVANCES BOTH POINTERS
092C    3E05                    MVI     A,05H   ;
092E    96                      SUB     M       ;COMPUTES 5-PSEC2
092F    02                      STAX    B       ;STORES RESULT IN PSECT
0930    03                      INX     B       ;
0931    23                      INX     H       ;ADVANCES BOTH POINTERS
0932    3E09                    MVI     A,09H   ;
0934    96                      SUB     M       ;COMPUTES 9-PMIN1
0935    02                      STAX    B       ;STORES RESULT IN PMINU
0936    C9                      RET             ;
```

```
0A00                        ORG     0A00H       ;TABLES
                    ;
                    ;
                    ;NASA LEGBAND DATA
                    ;
                    ;
                    ;THE FOLLOWING DATA IS FOR LB AF
0A00    C4          LBSD:   DB      0C4H
0A01    0D                  DB      00DH
0A02    2A                  DB      02AH
0A03    23                  DB      023H
0A04    49                  DB      049H
0A05    21                  DB      021H
0A06    6C                  DB      06CH
0A07    1D                  DB      01DH
0A08    97                  DB      097H
0A09    18                  DB      018H
0A0A    BF                  DB      0BFH
0A0B    19                  DB      019H
0A0C    E9                  DB      0E9H
0A0D    18                  DB      018H
                    ;THE FOLLOWING DATA IS FOR LB AG
0A0E    C4                  DB      0C4H
0A0F    10                  DB      010H
0A10    2A                  DB      02AH
0A11    27                  DB      027H
0A12    44                  DB      044H
0A13    27                  DB      027H
0A14    62                  DB      062H
0A15    22                  DB      022H
0A16    B4                  DB      0B4H
0A17    1E                  DB      01EH
0A18    A7                  DB      0A7H
0A19    1D                  DB      01DH
0A1A    CD                  DB      0CDH
0A1B    18                  DB      018H
                    ;THE FOLLOWING DATA IS FOR LB AM
0A1C    BB                  DB      0BBH
0A1D    10                  DB      010H
0A1E    2A                  DB      02AH
0A1F    27                  DB      027H
0A20    43                  DB      043H
0A21    28                  DB      028H
0A22    60                  DB      060H
0A23    22                  DB      022H
0A24    7C                  DB      07CH
0A25    24                  DB      024H
0A26    9E                  DB      09EH
0A27    1D                  DB      01DH
0A28    BE                  DB      0BEH
0A29    1F                  DB      01FH
                    ;THE FOLLOWING DATA IS FOR LB AR
0A2A    B3                  DB      0B3H
0A2B    0C                  DB      00CH
0A2C    2A                  DB      02AH
0A2D    22                  DB      022H
```

```
0A2E    49          DB      049H
0A2F    20          DB      020H
0A30    69          DB      069H
0A31    1F          DB      01FH
0A32    8D          DB      08DH
0A33    1C          DB      01CH
0A34    B6          DB      0B6H
0A35    18          DB      018H
0A36    E0          DB      0E0H
0A37    18          DB      018H
                    ;THE FOLLOWING DATA IS FOR LB AS
0A38    BB          DB      0BBH
0A39    0F          DB      00FH
0A3A    2A          DB      02AH
0A3B    26          DB      026H
0A3C    46          DB      046H
0A3D    24          DB      024H
0A3E    64          DB      064H
0A3F    22          DB      022H
0A40    83          DB      083H
0A41    21          DB      021H
0A42    A3          DB      0A3H
0A43    20          DB      020H
0A44    C8          DB      0C8H
0A45    18          DB      018H
                    ;THE FOLLOWING DATA IS FOR LB AT
0A46    D0          DB      0D0H
0A47    0B          DB      00BH
0A48    2A          DB      02AH
0A49    21          DB      021H
0A4A    4B          DB      04BH
0A4B    1E          DB      01EH
0A4C    6C          DB      06CH
0A4D    1F          DB      01FH
0A4E    91          DB      091H
0A4F    18          DB      018H
0A50    B7          DB      0B7H
0A51    18          DB      018H
0A52    E0          DB      0E0H
0A53    18          DB      018H
                    ;THE FOLLOWING DATA IS FOR LB AV
0A54    BB          DB      0BBH
0A55    0D          DB      00DH
0A56    2A          DB      02AH
0A57    23          DB      023H
0A58    48          DB      048H
0A59    22          DB      022H
0A5A    66          DB      066H
0A5B    21          DB      021H
0A5C    87          DB      087H
0A5D    1E          DB      01EH
0A5E    AB          DB      0ABH
0A5F    1C          DB      01CH
0A60    D5          DB      0D5H
0A61    18          DB      018H
                    ;THE FOLLOWING DATA IS FOR LB AY
```

```
0A62    CC          DB      0CCH
0A63    14          DB      014H
0A64    2A          DB      02AH
0A65    2F          DB      02FH
0A66    43          DB      043H
0A67    28          DB      028H
0A68    5E          DB      05EH
0A69    26          DB      026H
0A6A    79          DB      079H
0A6B    25          DB      025H
0A6C    97          DB      097H
0A6D    21          DB      021H
0A6E    8B          DB      08BH
0A6F    1E          DB      01EH
                ;THE FOLLOWING DATA IS FOR LB AZ
0A70    CC          DB      0CCH
0A71    12          DB      012H
0A72    2A          DB      02AH
0A73    2B          DB      02BH
0A74    44          DB      044H
0A75    26          DB      026H
0A76    60          DB      060H
0A77    24          DB      024H
0A78    7F          DB      07FH
0A79    21          DB      021H
0A7A    A0          DB      0A0H
0A7B    1E          DB      01EH
0A7C    C2          DB      0C2H
0A7D    1D          DB      01DH
                ;THE FOLLOWING DATA IS FOR LB BF
0A7E    AF          DB      0AFH
0A7F    11          DB      011H
0A80    2A          DB      02AH
0A81    29          DB      029H
0A82    46          DB      046H
0A83    24          DB      024H
0A84    64          DB      064H
0A85    21          DB      021H
0A86    85          DB      085H
0A87    1E          DB      01EH
0A88    AC          DB      0ACH
0A89    1A          DB      01AH
0A8A    D4          DB      0D4H
0A8B    19          DB      019H
                ;THE FOLLOWING DATA IS FOR LB BL
0A8C    CC          DB      0CCH
0A8D    15          DB      015H
0A8E    2A          DB      02AH
0A8F    31          DB      031H
0A90    41          DB      041H
0A91    2B          DB      02BH
0A92    5B          DB      05BH
0A93    26          DB      026H
0A94    79          DB      079H
0A95    21          DB      021H
0A96    95          DB      095H
```

```
0A97    24          DB      024H
0A98    B3          DB      0B3H
0A99    21          DB      021H
                    ;THE FOLLOWING DATA IS FOR LB BP
0A9A    AB          DB      0ABH
0A9B    0E          DB      00EH
0A9C    2A          DB      02AH
0A9D    24          DB      024H
0A9E    46          DB      046H
0A9F    24          DB      024H
0AA0    64          DB      064H
0AA1    21          DB      021H
0AA2    B6          DB      0B6H
0AA3    1D          DB      01DH
0AA4    A9          DB      0A9H
0AA5    1D          DB      01DH
0AA6    CE          DB      0CEH
0AA7    1B          DB      01BH
                    ;THE FOLLOWING DATA IS FOR LB BQ
0AA8    B3          DB      0B3H
0AA9    10          DB      010H
0AAA    2A          DB      02AH
0AAB    27          DB      027H
0AAC    46          DB      046H
0AAD    24          DB      024H
0AAE    66          DB      066H
0AAF    1F          DB      01FH
0AB0    8C          DB      08CH
0AB1    1A          DB      01AH
0AB2    B1          DB      0B1H
0AB3    1B          DB      01BH
0AB4    D9          DB      0D9H
0AB5    19          DB      019H
                    ;THE FOLLOWING DATA IS FOR LB BS
0AB6    BB          DB      0BBH
0AB7    12          DB      012H
0AB8    2A          DB      02AH
0AB9    2B          DB      02BH
0ABA    43          DB      043H
0ABB    29          DB      029H
0ABC    62          DB      062H
0ABD    21          DB      021H
0ABE    B5          DB      0B5H
0ABF    1D          DB      01DH
0AC0    A7          DB      0A7H
0AC1    1E          DB      01EH
0AC2    CD          DB      0CDH
0AC3    1B          DB      01BH
                    ;THE FOLLOWING DATA IS FOR LB BT
0AC4    AB          DB      0ABH
0AC5    11          DB      011H
0AC6    2A          DB      02AH
0AC7    29          DB      029H
0AC8    45          DB      045H
0AC9    25          DB      025H
0ACA    66          DB      066H
```

```
0ACB    1E          DB      01EH
0ACC    8A          DB      08AH
0ACD    1C          DB      01CH
0ACE    AF          DB      0AFH
0ACF    18          DB      018H
0AD0    D6          DB      0D6H
0AD1    1A          DB      01AH
                    ;THE FOLLOWING DATA IS FOR LB BX
0AD2    87          DB      087H
0AD3    0D          DB      00DH
0AD4    2A          DB      02AH
0AD5    23          DB      023H
0AD6    49          DB      049H
0AD7    20          DB      020H
0AD8    70          DB      070H
0AD9    1A          DB      01AH
0ADA    9A          DB      09AH
0ADB    18          DB      018H
0ADC    C3          DB      0C3H
0ADD    18          DB      018H
0ADE    EA          DB      0EAH
0ADF    1A          DB      01AH
                    ;THE FOLLOWING DATA IS FOR LB BY
0AE0    C4          DB      0C4H
0AE1    14          DB      014H
0AE2    2A          DB      02AH
0AE3    2F          DB      02FH
0AE4    43          DB      043H
0AE5    29          DB      029H
0AE6    60          DB      060H
0AE7    22          DB      022H
0AE8    81          DB      081H
0AE9    1F          DB      01FH
0AEA    A1          DB      0A1H
0AEB    1F          DB      01FH
0AEC    C4          DB      0C4H
0AED    1D          DB      01DH
                    ;THE FOLLOWING DATA IS FOR LB CB
0AEE    87          DB      087H
0AEF    0C          DB      00CH
0AF0    2A          DB      02AH
0AF1    22          DB      022H
0AF2    4A          DB      04AH
0AF3    1F          DB      01FH
0AF4    6D          DB      06DH
0AF5    1D          DB      01DH
0AF6    91          DB      091H
0AF7    1C          DB      01CH
0AF8    8C          DB      08CH
0AF9    17          DB      017H
0AFA    E8          DB      0E8H
0AFB    17          DB      017H
                    ;THE FOLLOWING DATA IS FOR LB CD
0AFC    83          DB      083H
0AFD    0C          DB      00CH
0AFE    2A          DB      02AH
```

```
0AFF    22          DB      022H
0B00    4D          DB      04DH
0B01    1D          DB      01DH
0B02    6E          DB      06EH
0B03    1F          DB      01FH
0B04    95          DB      095H
0B05    1A          DB      01AH
0B06    BF          DB      0BFH
0B07    18          DB      018H
0B08    EC          DB      0ECH
0B09    16          DB      016H
                    ;THE FOLLOWING DATA IS FOR LB CE
0B0A    9E          DB      09EH
0B0B    0A          DB      00AH
0B0C    2A          DB      02AH
0B0D    20          DB      020H
0B0E    4C          DB      04CH
0B0F    1E          DB      01EH
0B10    6F          DB      06FH
0B11    1D          DB      01DH
0B12    98          DB      098H
0B13    19          DB      019H
0B14    C6          DB      0C6H
0B15    16          DB      016H
0B16    F1          DB      0F1H
0B17    18          DB      018H
                    ;THE FOLLOWING DATA IS FOR LB CG
0B18    AF          DB      0AFH
0B19    0F          DB      00FH
0B1A    2A          DB      02AH
0B1B    26          DB      026H
0B1C    47          DB      047H
0B1D    23          DB      023H
0B1E    68          DB      068H
0B1F    1E          DB      01EH
0B20    87          DB      087H
0B21    21          DB      021H
0B22    B0          DB      0B0H
0B23    19          DB      019H
0B24    D7          DB      0D7H
0B25    1A          DB      01AH
                    ;THE FOLLOWING DATA IS FOR LB CL
0B26    C0          DB      0C0H
0B27    12          DB      012H
0B28    2A          DB      02AH
0B29    2B          DB      02BH
0B2A    44          DB      044H
0B2B    26          DB      026H
0B2C    63          DB      063H
0B2D    20          DB      020H
0B2E    7F          DB      07FH
0B2F    24          DB      024H
0B30    A3          DB      0A3H
0B31    1C          DB      01CH
0B32    C7          DB      0C7H
0B33    1C          DB      01CH
```

```
                        ;THE FOLLOWING DATA IS FOR LB DB
0B34    9E                      DB      09EH
0B35    12                      DB      012H
0B36    2A                      DB      02AH
0B37    28                      DB      028H
0B38    43                      DB      043H
0B39    28                      DB      028H
0B3A    61                      DB      061H
0B3B    21                      DB      021H
0B3C    82                      DB      082H
0B3D    1E                      DB      01EH
0B3E    A4                      DB      0A4H
0B3F    1E                      DB      01EH
0B40    C7                      DB      0C7H
0B41    10                      DB      010H
                        ;THE FOLLOWING DATA IS FOR LB AL
0B42    A2                      DB      0A2H
0B43    08                      DB      008H
0B44    2A                      DB      02AH
0B45    21                      DB      021H
0B46    4A                      DB      04AH
0B47    1F                      DB      01FH
0B48    68                      DB      068H
0B49    1F                      DB      01FH
0B4A    8F                      DB      08FH
0B4B    1C                      DB      01CH
0B4C    B8                      DB      0B8H
0B4D    18                      DB      018H
0B4E    E3                      DB      0E3H
0B4F    18                      DB      018H
                        ;THE FOLLOWING DATA IS FOR LB AP
0B50    A7                      DB      0A7H
0B51    13                      DB      013H
0B52    2A                      DB      02AH
0B53    2C                      DB      02CH
0B54    43                      DB      043H
0B55    29                      DB      029H
0B56    5C                      DB      05CH
0B57    28                      DB      028H
0B58    79                      DB      079H
0B59    23                      DB      023H
0B5A    9A                      DB      09AH
0B5B    1F                      DB      01FH
0B5C    B6                      DB      0B6H
0B5D    24                      DB      024H
                        ;THE FOLLOWING DATA IS FOR LB AU
0B5E    AF                      DB      0AFH
0B5F    0D                      DB      00DH
0B60    2A                      DB      02AH
0B61    23                      DB      023H
0B62    49                      DB      049H
0B63    21                      DB      021H
0B64    68                      DB      068H
0B65    20                      DB      020H
0B66    90                      DB      090H
0B67    19                      DB      019H
```

```
0B68    B8          DB      0B8H
0B69    19          DB      019H
0B6A    E1          DB      0E1H
0B6B    18          DB      018H
                    ;THE FOLLOWING DATA IS FOR LB CY
0B6C    C0          DB      0C0H
0B6D    12          DB      012H
0B6E    2A          DB      02AH
0B6F    2B          DB      02BH
0B70    44          DB      044H
0B71    26          DB      026H
0B72    5F          DB      05FH
0B73    25          DB      025H
0B74    7D          DB      07DH
0B75    22          DB      022H
0B76    9C          DB      09CH
0B77    20          DB      020H
0B78    BF          DB      0BFH
0B79    1D          DB      01DH
                    ;THE FOLLOWING DATA IS FOR LB BA
0B7A    AB          DB      0ABH
0B7B    10          DB      010H
0B7C    2A          DB      02AH
0B7D    27          DB      027H
0B7E    45          DB      045H
0B7F    25          DB      025H
0B80    60          DB      060H
0B81    25          DB      025H
0B82    81          DB      081H
0B83    1E          DB      01EH
0B84    A3          DB      0A3H
0B85    1E          DB      01EH
0B86    C6          DB      0C6H
0B87    1D          DB      01DH
                    ;THE FOLLOWING DATA IS FOR LB BB
0B88    B3          DB      0B3H
0B89    11          DB      011H
0B8A    2A          DB      02AH
0B8B    29          DB      029H
0B8C    44          DB      044H
0B8D    26          DB      026H
0B8E    63          DB      063H
0B8F    20          DB      020H
0B90    B8          DB      0B8H
0B91    1B          DB      01BH
0B92    AE          DB      0AEH
0B93    1A          DB      01AH
0B94    D4          DB      0D4H
0B95    1B          DB      01BH
                    ;THE FOLLOWING DATA IS FOR LB BC
0B96    C0          DB      0C0H
0B97    0D          DB      00DH
0B98    2A          DB      02AH
0B99    23          DB      023H
0B9A    4A          DB      04AH
0B9B    1F          DB      01FH
```

```
0B9C    70          DB      070H
0B9D    1A          DB      01AH
0B9E    9C          DB      09CH
0B9F    17          DB      017H
0BA0    C6          DB      0C6H
0BA1    18          DB      018H
0BA2    EB          DB      0EBH
0BA3    1B          DB      01BH
                ;THE FOLLOWING DATA IS FOR LB BD
0BA4    B7          DB      0B7H
0BA5    0D          DB      00DH
0BA6    2A          DB      02AH
0BA7    23          DB      023H
0BA8    4C          DB      04CH
0BA9    1E          DB      01EH
0BAA    6E          DB      06EH
0BAB    1E          DB      01EH
0BAC    93          DB      093H
0BAD    1B          DB      01BH
0BAE    BB          DB      0BBH
0BAF    19          DB      019H
0BB0    E6          DB      0E6H
0BB1    18          DB      018H
                ;THE FOLLOWING DATA IS FOR LB BE
0BB2    A7          DB      0A7H
0BB3    0E          DB      00EH
0BB4    2A          DB      02AH
0BB5    24          DB      024H
0BB6    46          DB      046H
0BB7    24          DB      024H
0BB8    65          DB      065H
0BB9    21          DB      021H
0BBA    85          DB      085H
0BBB    1F          DB      01FH
0BBC    A9          DB      0A9H
0BBD    1C          DB      01CH
0BBE    CE          DB      0CEH
0BBF    1B          DB      01BH
                ;THE FOLLOWING DATA IS FOR LB BG
0BC0    AE          DB      0AEH
0BC1    0E          DB      00EH
0BC2    2A          DB      02AH
0BC3    24          DB      024H
0BC4    48          DB      048H
0BC5    21          DB      021H
0BC6    68          DB      068H
0BC7    1F          DB      01FH
0BC8    8B          DB      08BH
0BC9    1D          DB      01DH
0BCA    81          DB      081H
0BCB    1B          DB      01BH
0BCC    D7          DB      0D7H
0BCD    1B          DB      01BH
                ;THE FOLLOWING DATA IS FOR LB BH
0BCE    C0          DB      0C0H
0BCF    0B          DB      00BH
```

```
0BD0    2A          DB      02AH
0BD1    21          DB      021H
0BD2    4B          DB      04BH
0BD3    1F          DB      01FH
0BD4    71          DB      071H
0BD5    1B          DB      01BH
0BD6    9A          DB      09AH
0BD7    19          DB      019H
0BD8    CC          DB      0CCH
0BD9    14          DB      014H
0BDA    F4          DB      0F4H
0BDB    19          DB      019H
                    ;THE FOLLOWING DATA IS FOR LB CC
0BDC    A7          DB      0A7H
0BDD    0D          DB      00DH
0BDE    2A          DB      02AH
0BDF    23          DB      023H
0BE0    4A          DB      04AH
0BE1    20          DB      020H
0BE2    6D          DB      06DH
0BE3    1D          DB      01DH
0BE4    90          DB      090H
0BE5    1D          DB      01DH
0BE6    BA          DB      0BAH
0BE7    18          DB      018H
0BE8    E4          DB      0E4H
0BE9    18          DB      018H
                    ;THE FOLLOWING DATA IS FOR LB CE
0BEA    C0          DB      0C0H
0BEB    15          DB      015H
0BEC    2A          DB      02AH
0BED    31          DB      031H
0BEE    44          DB      044H
0BEF    26          DB      026H
0BF0    60          DB      060H
0BF1    24          DB      024H
0BF2    78          DB      078H
0BF3    26          DB      026H
0BF4    9A          DB      09AH
0BF5    21          DB      021H
0BF6    C3          DB      0C3H
0BF7    18          DB      018H
                    ;THE FOLLOWING DATA IS FOR LB CG
0BF8    CC          DB      0CCH
0BF9    15          DB      015H
0BFA    2A          DB      02AH
0BFB    31          DB      031H
0BFC    43          DB      043H
0BFD    29          DB      029H
0BFE    5C          DB      05CH
0BFF    28          DB      028H
0C00    79          DB      079H
0C01    23          DB      023H
0C02    99          DB      099H
0C03    20          DB      020H
0C04    BB          DB      0BBH
```

```
0C05    1E          DB      01EH
                $THE FOLLOWING DATA IS FOR LB AC
0C06    E5          DB      0E5H
0C07    18          DB      018H
0C08    2A          DB      02AH
0C09    39          DB      039H
0C0A    43          DB      043H
0C0B    29          DB      029H
0C0C    50          DB      050H
0C0D    27          DB      027H
0C0E    75          DB      075H
0C0F    28          DB      028H
0C10    94          DB      094H
0C11    21          DB      021H
0C12    89          DB      089H
0C13    18          DB      018H
                $THE FOLLOWING DATA IS FOR LB AD
0C14    87          DB      087H
0C15    0F          DB      00FH
0C16    2A          DB      02AH
0C17    26          DB      026H
0C18    44          DB      044H
0C19    27          DB      027H
0C1A    68          DB      068H
0C1B    1A          DB      01AH
0C1C    87          DB      087H
0C1D    24          DB      024H
0C1E    AA          DB      0AAH
0C1F    1D          DB      01DH
0C20    D0          DB      0D0H
0C21    1A          DB      01AH
                $THE FOLLOWING DATA IS FOR LB AJ
0C22    BB          DB      0BBH
0C23    18          DB      018H
0C24    2A          DB      02AH
0C25    39          DB      039H
0C26    41          DB      041H
0C27    28          DB      028H
0C28    58          DB      058H
0C29    2C          DB      02CH
0C2A    70          DB      070H
0C2B    28          DB      028H
0C2C    8C          DB      08CH
0C2D    24          DB      024H
0C2F    AF          DB      0AFH
0C2F    1D          DB      01DH
                $THE FOLLOWING DATA IS FOR LB AN
0C30    C4          DB      0C4H
0C31    15          DB      015H
0C32    2A          DB      02AH
0C33    31          DB      031H
0C34    41          DB      041H
0C35    2C          DB      02CH
0C36    5C          DB      05CH
0C37    26          DB      026H
0C38    77          DB      077H
```

```
0C39    26          DB      026H
0C3A    93          DB      093H
0C3B    24          DB      024H
0C3C    81          DB      081H
0C3D    21          DB      021H
                    ;THE FOLLOWING DATA IS FOR LB AQ
0C3E    AF          DB      0AFH
0C3F    07          DB      007H
0C40    2A          DB      02AH
0C41    1D          DB      01DH
0C42    4F          DB      04FH
0C43    1B          DB      01BH
0C44    76          DB      076H
0C45    1A          DB      01AH
0C46    9C          DB      09CH
0C47    1A          DB      01AH
0C48    C4          DB      0C4H
0C49    19          DB      019H
0C4A    F0          DB      0F0H
0C4B    17          DB      017H
                    ;THE FOLLOWING DATA IS FOR LB AX
0C4C    C4          DB      0C4H
0C4D    15          DB      015H
0C4E    2A          DB      02AH
0C4F    31          DB      031H
0C50    42          DB      042H
0C51    2A          DB      02AH
0C52    5E          DB      05EH
0C53    24          DB      024H
0C54    7A          DB      07AH
0C55    24          DB      024H
0C56    9A          DB      09AH
0C57    1F          DB      01FH
0C58    BC          DB      0BCH
0C59    1F          DB      01FH
                    ;THE FOLLOWING DATA IS FOR LB BJ
0C5A    AF          DB      0AFH
0C5B    17          DB      017H
0C5C    2A          DB      02AH
0C5D    36          DB      036H
0C5E    41          DB      041H
0C5F    2B          DB      02BH
0C60    5D          DB      05DH
0C61    24          DB      024H
0C62    7C          DB      07CH
0C63    20          DB      020H
0C64    9B          DB      09BH
0C65    20          DB      020H
0C66    BC          DB      0BCH
0C67    1F          DB      01FH
                    ;THE FOLLOWING DATA IS FOR LB BK
0C68    C0          DB      0C0H
0C69    15          DB      015H
0C6A    2A          DB      02AH
0C6B    31          DB      031H
0C6C    41          DB      041H
```

```
0C6D    2C          DB      02CH
0C6E    5C          DB      05CH
0C6F    26          DB      026H
0C70    7A          DB      07AH
0C71    22          DB      022H
0C72    98          DB      098H
0C73    21          DB      021H
0C74    B8          DB      0B8H
0C75    1F          DB      01FH
                ;THE FOLLOWING DATA IS FOR LB BU
0C76    BB          DB      0BBH
0C77    11          DB      011H
0C78    2A          DB      02AH
0C79    29          DB      029H
0C7A    45          DB      045H
0C7B    25          DB      025H
0C7C    63          DB      063H
0C7D    22          DB      022H
0C7F    85          DB      085H
0C7F    1D          DB      01DH
0C80    A6          DB      0A6H
0C81    1E          DB      01EH
0C82    CB          DB      0CBH
0C83    1B          DB      01BH
                ;THE FOLLOWING DATA IS FOR LB BV
0C84    C0          DB      0C0H
0C85    15          DB      015H
0C86    2A          DB      02AH
0C87    31          DB      031H
0C88    41          DB      041H
0C89    2C          DB      02CH
0C8A    5A          DB      05AH
0C8B    29          DB      029H
0C8C    79          DB      079H
0C8D    21          DB      021H
0C8E    97          DB      097H
0C8F    22          DB      022H
0C90    BC          DB      0BCH
0C91    1B          DB      01BH
                ;THE FOLLOWING DATA IS FOR LB BZ
0C92    B3          DB      0B3H
0C93    10          DB      010H
0C94    2A          DB      02AH
0C95    27          DB      027H
0C96    46          DB      046H
0C97    24          DB      024H
0C98    63          DB      063H
0C99    23          DB      023H
0C9A    83          DB      083H
0C9B    20          DB      020H
0C9C    A7          DB      0A7H
0C9D    1C          DB      01CH
0C9E    CF          DB      0CFH
0C9F    19          DB      019H
                ;THE FOLLOWING DATA IS FOR LB CA
0CA0    C4          DB      0C4H
```

```
0CA1    0D          DB      00DH
0CA2    2A          DB      02AH
0CA3    23          DB      023H
0CA4    49          DB      049H
0CA5    21          DB      021H
0CA6    69          DB      069H
0CA7    1F          DB      01FH
0CA8    8E          DB      08EH
0CA9    18          DB      018H
0CAA    B6          DB      0B6H
0CAB    19          DB      019H
0CAC    E2          DB      0E2H
0CAD    17          DB      017H
                    ;THE FOLLOWING DATA IS FOR LB CH
0CAE    AF          DB      0AFH
0CAF    14          DB      014H
0CB0    2A          DB      02AH
0CB1    2F          DB      02FH
0CB2    42          DB      042H
0CB3    2A          DB      02AH
0CB4    58          DB      058H
0CB5    28          DB      028H
0CB6    75          DB      075H
0CB7    26          DB      026H
0CB8    93          DB      093H
0CB9    21          DB      021H
0CBA    B5          DB      0B5H
0CBB    1E          DB      01EH
                    ;THE FOLLOWING DATA IS FOR LB CI
0CBC    E5          DB      0E5H
0CBD    12          DB      012H
0CBE    2A          DB      02AH
0CBF    28          DB      028H
0CC0    43          DB      043H
0CC1    29          DB      029H
0CC2    5C          DB      05CH
0CC3    29          DB      029H
0CC4    76          DB      076H
0CC5    27          DB      027H
0CC6    93          DB      093H
0CC7    22          DB      022H
0CC8    B4          DB      0B4H
0CC9    1E          DB      01EH
                    ;THE FOLLOWING DATA IS LB CJ
0CCA    BB          DB      0BBH
0CCB    1A          DB      01AH
0CCC    2A          DB      02AH
0CCD    41          DB      041H
0CCE    3D          DB      03DH
0CCF    33          DB      033H
0CD0    54          DB      054H
0CD1    2C          DB      02CH
0CD2    6B          DB      06BH
0CD3    28          DB      028H
0CD4    87          DB      087H
0CD5    24          DB      024H
```

```
0CD6    A3          DB    0A3H
0CD7    24          DB    024H
                ;THE FOLLOWING DATA IS FOR LB CO
0CD8    C0          DB    0C0H
0CD9    1B          DB    01BH
0CDA    2A          DB    02AH
0CDB    45          DB    045H
0CDC    43          DB    043H
0CDD    29          DB    029H
0CDE    5D          DB    05DH
0CDF    27          DB    027H
0CE0    74          DB    074H
0CE1    2B          DB    02BH
0CE2    94          DB    094H
0CE3    20          DB    020H
0CE4    B7          DB    0B7H
0CE5    1D          DB    01DH
                ;THE FOLLOWING DATA IS FOR LB CP
0CE6    C0          DB    0C0H
0CE7    11          DB    011H
0CE8    2A          DB    02AH
0CE9    29          DB    029H
0CEA    45          DB    045H
0CEB    26          DB    026H
0CEC    62          DB    062H
0CED    22          DB    022H
0CEE    B4          DB    0B4H
0CEF    1E          DB    01EH
0CF0    A9          DB    0A9H
0CF1    1B          DB    01BH
0CF2    D0          DB    0D0H
0CF3    1A          DB    01AH
                ;THE FOLLOWING DATA IS FOR LB CR
0CF4    C0          DB    0C0H
0CF5    10          DB    010H
0CF6    2A          DB    02AH
0CF7    27          DB    027H
0CF8    47          DB    047H
0CF9    22          DB    022H
0CFA    64          DB    064H
0CFB    22          DB    022H
0CFC    B6          DB    0B6H
0CFD    1E          DB    01EH
0CFE    AA          DB    0AAH
0CFF    1C          DB    01CH
0D00    CF          DB    0CFH
0D01    1B          DB    01BH
                ;THE FOLLOWING DATA IS FOR LB CS
0D02    C8          DB    0C8H
0D03    16          DB    016H
0D04    2A          DB    02AH
0D05    33          DB    033H
0D06    44          DB    044H
0D07    26          DB    026H
0D08    5E          DB    05EH
0D09    26          DB    026H
```

```
0D0A    79          DB      079H
0D0B    25          DB      025H
0D0C    97          DB      097H
0D0D    21          DB      021H
0D0E    86          DB      086H
0D0F    21          DB      021H
                    ;THE FOLLOWING DATA IS FOR LB CT
0D10    E9          DB      0E9H
0D11    17          DB      017H
0D12    2A          DB      02AH
0D13    36          DB      036H
0D14    42          DB      042H
0D15    2A          DB      02AH
0D16    5A          DB      05AH
0D17    2A          DB      02AH
0D18    72          DB      072H
0D19    2A          DB      02AH
0D1A    8C          DB      08CH
0D1B    26          DB      026H
0D1C    A7          DB      0A7H
0D1D    26          DB      026H
                    ;THE FOLLOWING DATA IS FOR LB CU
0D1E    AF          DB      0AFH
0D1F    0F          DB      00FH
0D20    2A          DB      02AH
0D21    26          DB      026H
0D22    47          DB      047H
0D23    23          DB      023H
0D24    66          DB      066H
0D25    21          DB      021H
0D26    88          DB      088H
0D27    1D          DB      01DH
0D28    80          DB      080H
0D29    19          DB      019H
0D2A    DA          DB      0DAH
0D2B    18          DB      018H
                    ;THE FOLLOWING DATA IS FOR LB CV
0D2C    AB          DB      0ABH
0D2D    15          DB      015H
0D2E    2A          DB      02AH
0D2F    31          DB      031H
0D30    43          DB      043H
0D31    29          DB      029H
0D32    5C          DB      05CH
0D33    28          DB      028H
0D34    7A          DB      07AH
0D35    21          DB      021H
0D36    9D          DB      09DH
0D37    1D          DB      01DH
0D38    BE          DB      0BEH
0D39    1E          DB      01EH
                    ;THE FOLLOWING DATA IS FOR LB CX
0D3A    87          DB      087H
0D3B    17          DB      017H
0D3C    2A          DB      02AH
0D3D    36          DB      036H
```

```
0D3E    3E          DB      03EH
0D3F    32          DB      032H
0D40    56          DB      056H
0D41    2A          DB      02AH
0D42    72          DB      072H
0D43    24          DB      024H
0D44    8D          DB      08DH
0D45    26          DB      026H
0D46    A8          DB      0A8H
0D47    25          DB      025H
                    ;THE FOLLOWING DATA IS FOR LB CY
0D48    B7          DB      0B7H
0D49    17          DB      017H
0D4A    2A          DB      02AH
0D4B    36          DB      036H
0D4C    44          DB      044H
0D4D    27          DB      027H
0D4E    5F          DB      05FH
0D4F    26          DB      026H
0D50    7B          DB      07BH
0D51    24          DB      024H
0D52    9C          DB      09CH
0D53    1E          DB      01EH
0D54    C5          DB      0C5H
0D55    19          DB      019H
                    ;THE FOLLOWING DATA IS FOR LB AA
0D56    BB          DB      0BBH
0D57    11          DB      011H
0D58    2A          DB      02AH
0D59    29          DB      029H
0D5A    43          DB      043H
0D5B    28          DB      028H
0D5C    5F          DB      05FH
0D5D    24          DB      024H
0D5E    79          DB      079H
0D5F    26          DB      026H
0D60    98          DB      098H
0D61    20          DB      020H
0D62    B7          DB      0B7H
0D63    20          DB      020H
                    ;THE FOLLOWING DATA IS FOR LB AB
0D64    B7          DB      0B7H
0D65    10          DB      010H
0D66    2A          DB      02AH
0D67    27          DB      027H
0D68    45          DB      045H
0D69    25          DB      025H
0D6A    65          DB      065H
0D6B    20          DB      020H
0D6C    82          DB      082H
0D6D    22          DB      022H
0D6E    A5          DB      0A5H
0D6F    1D          DB      01DH
0D70    CE          DB      0CEH
0D71    19          DB      019H
                    ;THE FOLLOWING DATA IS FOR LB AH
```

```
0D72    9A          DB      09AH
0D73    09          DB      009H
0D74    2A          DB      02AH
0D75    1F          DB      01FH
0D76    4F          DB      04FH
0D77    1B          DB      01BH
0D78    77          DB      077H
0D79    19          DB      019H
0D7A    9E          DB      09EH
0D7B    1A          DB      01AH
0D7C    CC          DB      0CCH
0D7D    16          DB      016H
0D7E    F9          DB      0F9H
0D7F    16          DB      016H
                    ;THE FOLLOWING DATA IS FOR LB AI
0D80    B3          DB      0B3H
0D81    0D          DB      00DH
0D82    2A          DB      02AH
0D83    23          DB      023H
0D84    49          DB      049H
0D85    20          DB      020H
0D86    6C          DB      06CH
0D87    1D          DB      01DH
0D88    90          DB      090H
0D89    1C          DB      01CH
0D8A    BA          DB      0BAH
0D8B    18          DB      018H
0D8C    E6          DB      0E6H
0D8D    17          DB      017H
                    ;THE FOLLOWING DATA IS FOR LB AK
0D8E    AF          DB      0AFH
0D8F    10          DB      010H
0D90    2A          DB      02AH
0D91    27          DB      027H
0D92    47          DB      047H
0D93    23          DB      023H
0D94    64          DB      064H
0D95    22          DB      022H
0D96    86          DB      086H
0D97    1E          DB      01EH
0D98    AD          DB      0ADH
0D99    1A          DB      01AH
0D9A    D1          DB      0D1H
0D9B    1C          DB      01CH
                    ;THE FOLLOWING DATA IS FOR LB AO
0D9C    D4          DB      0D4H
0D9D    16          DB      016H
0D9E    2A          DB      02AH
0D9F    33          DB      033H
0DA0    41          DB      041H
0DA1    2B          DB      02BH
0DA2    5A          DB      05AH
0DA3    29          DB      029H
0DA4    74          DB      074H
0DA5    26          DB      026H
0DA6    93          DB      093H
```

```
0DA7    20          DB      020H
0DA8    B3          DB      0B3H
0DA9    1F          DB      01FH
                ;THE FOLLOWING DATA IS FOR LB BI
0DAA    B3          DB      0B3H
0DAB    0C          DB      00CH
0DAC    2A          DB      02AH
0DAD    22          DB      022H
0DAE    4C          DB      04CH
0DAF    1E          DB      01EH
0DB0    70          DB      070H
0DB1    1C          DB      01CH
0DB2    9A          DB      09AH
0DB3    18          DB      018H
0DB4    C8          DB      0C8H
0DB5    16          DB      016H
0DB6    F0          DB      0F0H
0DB7    19          DB      019H
                ;THE FOLLOWING DATA IS FOR LB CZ
0DB8    A7          DB      0A7H
0DB9    11          DB      011H
0DBA    2A          DB      02AH
0DBB    29          DB      029H
0DBC    45          DB      045H
0DBD    25          DB      025H
0DBE    67          DB      067H
0DBF    1D          DB      01DH
0DC0    90          DB      090H
0DC1    19          DB      019H
0DC2    B4          DB      0B4H
0DC3    1C          DB      01CH
0DC4    DD          DB      0DDH
0DC5    19          DB      019H
                ;THE FOLLOWING DATA IS FOR LB BN
0DC6    BB          DB      0BBH
0DC7    0F          DB      00FH
0DC8    2A          DB      02AH
0DC9    26          DB      026H
0DCA    4B          DB      04BH
0DCB    1E          DB      01EH
0DCC    6C          DB      06CH
0DCD    1E          DB      01EH
0DCE    98          DB      098H
0DCF    17          DB      017H
0DD0    C2          DB      0C2H
0DD1    18          DB      018H
0DD2    E2          DB      0E2H
0DD3    1F          DB      01FH
                ;THE FOLLOWING DATA IS FOR LB BO
0DD4    BB          DB      0BBH
0DD5    12          DB      012H
0DD6    2A          DB      02AH
0DD7    2B          DB      02BH
0DD8    46          DB      046H
0DD9    24          DB      024H
0DDA    68          DB      068H
```

```
0DDB    1E          DB      01EH
0DDC    8D          DB      08DH
0DDD    1B          DB      01BH
0DDE    B2          DB      0B2H
0DDF    1B          DB      01BH
0DE0    D5          DB      0D5H
0DE1    1D          DB      01DH
                    ;THE FOLLOWING DATA IS FOR LB CF
0DE2    C4          DB      0C4H
0DE3    0C          DB      00CH
0DE4    2A          DB      02AH
0DE5    22          DB      022H
0DE6    4A          DB      04AH
0DE7    1F          DB      01FH
0DE8    6D          DB      06DH
0DE9    1D          DB      01DH
0DEA    9B          DB      09BH
0DEB    16          DB      016H
0DEC    C9          DB      0C9H
0DED    16          DB      016H
0DEF    F3          DB      0F3H
0DEF    1B          DB      01BH
                    ;THE FOLLOWING DATA IS FOR LB CK
0DF0    AB          DB      0ABH
0DF1    0F          DB      00FH
0DF2    2A          DB      02AH
0DF3    26          DB      026H
0DF4    4B          DB      04BH
0DF5    22          DB      022H
0DF6    6A          DB      06AH
0DF7    1D          DB      01DH
0DF8    93          DB      093H
0DF9    1B          DB      01BH
0DFA    BA          DB      0BAH
0DFB    1A          DB      01AH
0DFC    E3          DB      0E3H
0DFD    19          DB      019H
                    ;THE FOLLOWING DATA IS FOR LB CN
0DFE    C4          DB      0C4H
0DFF    10          DB      010H
0E00    2A          DB      02AH
0E01    27          DB      027H
0E02    47          DB      047H
0E03    23          DB      023H
0E04    6B          DB      06BH
0E05    1C          DB      01CH
0E06    95          DB      095H
0E07    18          DB      018H
0E08    BB          DB      0BBH
0E09    1B          DB      01BH
0E0A    E0          DB      0E0H
0E0B    1B          DB      01BH
                    ;
```

```
                        ;
0E0C    00              DEAD:   DB      0DH
0E0D    00                      DB      0DH
0E0E    0A                      DB      0AH
0E0F    0E                      DB      0EH
0E10    00                      DB      0DH
                        ;
                        ;
0E11    0F              GOOF:   DB      0FH
0E12    00                      DB      0
0E13    00                      DB      0
0E14    00                      DB      0
0E15    06                      DB      6
                        ;
                        ;THE FOLLOWING LOCATIONS MUST BE SEQUENTIAL.
                        ;NOTE THAT PRESSURE ENTRIES ARE IN MM HG. WHILE DWELL TIME ENTRIES ARE
                        ;IN (MIN)-1
                        ;
0E16    00              TABLE:  DB      00D             ;VACUUM = 0 MM HG
0E17    04                      DB      04D             ;DWELL TIME = 5 MIN
0E18    08                      DB      08D             ;VACUUM = 8 MM HG
0E19    00                      DB      00D             ;DWELL TIME = 1 MIN
0E1A    10                      DB      16D             ;VACUUM = 16 MM HG
0E1B    00                      DB      00D             ;DWELL TIME = 1 MIN
0E1C    1E                      DB      30D             ;30 MM HG
0E1D    02                      DB      02D             ;3 MIN
0E1E    28                      DB      40D             ;40 MM HG
0E1F    04                      DB      04D             ;5 MIN
0E20    32                      DB      50D             ;50 MM HG
0E21    04                      DB      04D             ;5 MIN
0E22    00                      DB      00D             ;VACUUM = 0 MM HG
0E23    FF                      DB      0FFH            ;NEGATIVE TIME - TERMINATE SEQUENCE
                        ;
```

```
0F00                                ORG     0F00H
0F00    01          LUTAB:          DB      01H         ;LOOK UP TABLE USED BY LBDSP
0F01    0000                        DW      00H
0F03    0000                        DW      00H
0F05    0909                        DW      909H
0F07    0909                        DW      909H
0F09    0808                        DW      808H
0F0B    0808                        DW      808H
0F0D    0707                        DW      707H
0F0F    0707                        DW      707H
0F11    0606                        DW      606H
0F13    0606                        DW      606H
0F15    0505                        DW      505H
0F17    0505                        DW      505H
0F19    0404                        DW      404H
0F1B    0404                        DW      404H
0F1D    0303                        DW      303H
0F1F    0303                        DW      303H
0F21    0202                        DW      202H
0F23    0202                        DW      202H
0F25    0101                        DW      101H
0F27    0101                        DW      101H
0F29    0000                        DW      000H
0F2B    00                          DB      000H
0F2C    0101                        DW      101H
0F2E    0101                        DW      101H
0F30    0202                        DW      202H
0F32    0202                        DW      202H
0F34    0303                        DW      303H
0F36    0303                        DW      303H
0F38    0404                        DW      404H
0F3A    0404                        DW      404H
0F3C    0505                        DW      505H
0F3E    0505                        DW      505H
0F40    0606                        DW      606H
0F42    0606                        DW      606H
0F44    0707                        DW      707H
0F46    0707                        DW      707H
0F48    0808                        DW      808H
0F4A    0808                        DW      808H
0F4C    0909                        DW      909H
0F4E    0909                        DW      909H
0F50    0000                        DW      000H
0F52    0000                        DW      000H
0F54    0101                        DW      101H
0F56    0101                        DW      101H
0F58    0202                        DW      202H
0F5A    0202                        DW      202H
0F5C    0303                        DW      303H
0F5E    0303                        DW      303H
0F60    0404                        DW      404H
0F62    0404                        DW      404H
0F64    0505                        DW      505H
0F66    0505                        DW      505H
0F68    0606                        DW      606H
0F6A    0606                        DW      606H
```

```
0F6C    0707    DW    707H
0F6E    0707    DW    707H
0F70    0808    DW    808H
0F72    0808    DW    808H
0F74    0909    DW    909H
0F76    0909    DW    909H
0F78    0000    DW    000H
0F7A    0000    DW    000H
0F7C    0101    DW    101H
0F7E    0101    DW    101H
0F80    0202    DW    202H
0F82    0202    DW    202H
0F84    0303    DW    303H
0F86    0303    DW    303H
0F88    0404    DW    404H
0F8A    0404    DW    404H
0F8C    0505    DW    505H
0F8E    0505    DW    505H
0F90    0606    DW    606H
0F92    0606    DW    606H
0F94    0707    DW    707H
0F96    0707    DW    707H
0F98    0808    DW    808H
0F9A    0808    DW    808H
0F9C    0909    DW    909H
0F9E    0909    DW    909H
0FA0    0000    DW    000H
0FA2    0000    DW    000H
0FA4    0101    DW    101H
0FA6    0101    DW    101H
0FA8    0202    DW    202H
0FAA    0202    DW    202H
0FAC    0303    DW    303H
0FAE    0303    DW    303H
0FB0    0404    DW    404H
0FB2    0404    DW    404H
0FB4    0505    DW    505H
0FB6    0505    DW    505H
0FB8    0606    DW    606H
0FBA    0606    DW    606H
0FBC    0707    DW    707H
0FBE    0707    DW    707H
0FC0    0808    DW    808H
0FC2    0808    DW    808H
0FC4    0909    DW    909H
0FC6    0909    DW    909H
0FC8    0000    DW    000H
0FCA    0000    DW    000H
0FCC    0101    DW    101H
0FCE    0101    DW    101H
0FD0    0202    DW    202H
0FD2    0202    DW    202H
0FD4    0303    DW    303H
0FD6    0303    DW    303H
0FD8    0404    DW    404H
0FDA    0404    DW    404H
```

```
0FDC    0505    DW    505H
0FDE    0505    DW    505H
0FE0    0606    DW    606H
0FE2    0606    DW    606H
0FE4    0707    DW    707H
0FE6    0707    DW    707H
0FE8    0808    DW    808H
0FEA    0808    DW    808H
0FEC    0909    DW    909H
0FEE    0909    DW    909H
0FF0    0000    DW    000H
0FF2    0000    DW    000H
0FF4    0101    DW    101H
0FF6    0101    DW    101H
0FF8    0202    DW    202H
0FFA    0202    DW    202H
0FFC    0303    DW    303H
0FFE    0303    DW    303H
                END
```

SYMBOL	VALUE	SYMBOL	VALUE	SYMBOL	VALUE	SYMBOL	VALUE	SYMBOL	VALUE	SYMBOL	VALUE
ABORTF	2F10	CR1LP	D803	FASTR	8908	LBCAL	3E05	MSAT	F206	RPBIN	2A10
ACC10	3F02	CR2LP	2904	FNE1	F303	LBDSP	3306	MULT0	0702	RSET	8B08
ADDR	FA10	DBMS	BF01	FNE2	4404	LBD1C	8506	MULT1	1102	SAT	5E07
APPROX	1107	DEAD	0C0E	FRWRD	AC00	LBER1	2505	NBASE	3907	SDDSP	E005
ASEC1	2310	DIGT1	A002	FTDSP	BD05	LBER2	2E05	NEWCK	2309	SECPER	1300
ATIM	2210	DIGT	8A02	GAINC	BB04	LBNBR	4A00	NOABT	E108	SECTR	3810
BADD	6001	DISHO	3710	GAINL	D704	LBRDR	EF05	NOCAL	4105	SETBCD	5F02
BASE	E010	DISPL	1501	GAINR	7004	LBSD	000A	NTVLD	9003	SETBIN	4802
BCDBIN	3B02	DISP5	6502	GAIN1	4410	LB1CT	0100	NULL1	CC03	SETBP	CD00
BEGIN	2603	DIV0	1902	GAIN2	4710	LB1C	0600	NULL2	1F04	SETTL	8000
BIAS1	4210	DIV1	2A02	GETAD	7500	LB1F	0700	OCTPT	0300	SRT	9B05
BIAS2	4510	DIV2	3202	GETBY	7B00	LB1N	6F06	OTCTRL	0800	START	0000
BINBCD	4B02	DMORE	7902	GETCM	1C00	LB10	4010	OTPT	6407	STEG	7200
BISTA	5010	DNXDIG	6902	GL	DF04	LB1T	5706	OTVALV	0200	STROB	AC06
BKPNT	DD00	DOBND	3203	GOOFY	9202	LB1	0400	OUTPT	2608	TABEN	3610
BKWRD	B600	DONEG	FF06	GOOF	110E	LB2CT	0200	OUT1	3010	TABLE	160E
BLOOP	8503	DONE1	0E04	GR	7B04	LB2C	0400	OUT2	3110	TBYTD	1602
BND1F	3405	DONE2	5F04	HOLD1	4A10	LB2F	0500	OUT5	3410	TBYTM	0302
BND2F	3905	DONE	6108	HOLD2	4C10	LB2N	6906	PAUTO	BB08	TEST1	D602
BPADR	F710	DOSUB	0307	IABORT	0B03	LB20	4110	PEXIT	F808	TEST	8F06
BPCLR	0B01	DPBIN	2B10	IABT	1103	LB2	0500	PMINU	2910	TWEEK	4B07
BPDIS	FD00	DPDEL	2D10	ILOOP	1504	LB	3F08	PNTR	7A03	TWONG	7906
BPSTB	F510	DPCLD	2C10	IMPT	AC08	LDFLG	FC10	POUT	1000	VACLK	2110
BPTEM	F910	DPRES	6905	INCTRL	0900	LECOM	2B05	PRDSP	0E06	VADJ	6708
CALIB	BF03	DSDE	6A01	INIT	EA02	LEEV	9B08	PTIME	2710	VALFD	2F07
CBASE	4100	DSPLA	FD10	INPRES	0800	LOADR	6C00	PTIM	2410	VALVE	3510
CBRCH	9503	DSPMD	5110	INTCT	2F10	LOOP	3F01	PTYME	FA08	VAPER	0202
CHANG	BD00	DSPMS	FE10	INTPER	0400	LUTAB	000F	RAMP	8E08	VLV	3708
CKCONT	AC03	DSP	5308	INTVAR	2110	LUTB	0F00	RATE	3910	VMIN	6107
CLOCK	0E09	DS1BY	7A01	INT	0008	L01	A706	RDAGN	4405	WAITA	B506
CLRIT	B702	DS2BY	6601	IRAT5	AB05	L10UT	1200	RD1BY	8A01	XLUPE	5103
CLR	1000	DUN	7B06	IRAT6	B405	L20UT	1100	RD2A	B501	YLUPE	6F03
CMPP	CB06	D1KS	C101	KEYBD	BD02	MASK1	FF10	RD2BY	AA01	ZERO	7F02
CMRD	2300	ECALL	3B00	KEYCD	3B01	MAUTO	0200	RD2HL	6100	ZLOOP	8202
CMUL	DA06	EJUMP	6800	KEYIN	2F01	MAXIN1	4310	READU	4F01		
COMCD	A202	END1	A406	KEYUP	4D01	MAXIN2	4610	READ	A408		
COMP	BE06	END2	A306	KLOOP	6604	MMAN	0100	RESET	B402		
COM	B000	ENTER	8F00	LAB0	0D07	MODE	2010	RMAX	0102		
CONT	6005	EXAMN	7000	LAB	F406	MONITOR	0D00	RMIN	0002		
CONV	5602	EXIT	E402	LBA1	F806	MOVER	CA01	ROTAT	4001		

```
PASS ONE
 18.73 SECONDS EXECUTION TIME
     0 ERRORS DETECTED

PASS TWO
 23.97 SECONDS EXECUTION TIME
     0 ERRORS DETECTED
  3493 BYTES ASSEMBLED

TOTAL ASSEMBLY TIME  43.32 CP SECONDS
```

APPENDIX VI

PROCEDURE FOR SYSTEM INITIALIZATION

Procedure for initial system set-up (refer to Figure A6.1):

1. Connect the vacuum source hose to the input of the valve manifold.

2. Connect the vacuum hose from the output of the valve manifold to the LBNPD.

3. Verify that the vacuum switch lever on the LBNPD is open and the manual pressure control valve on the LBNPD is turned fully open.

4. Connect the valve control cable from the computer rear bulkhead connector to the connector on the valve assembly box.

5. Connect the pressure transducer cable from the LBNPD connector to the rear bulkhead connector on the computer.

6. Connect the leg band cable from the LBNPD connector to the computer rear bulkhead at one of the two connectors depending on whether the Air Force Academy or NASA cable is used. Be sure the ground cable in the cable bundle is connected securely to the LBNPD frame and the computer chassis.

7. Run the procedure to calibrate the computer power supplies.

8. Run the procedure to calibrate the A/D leg band data amplifiers.

9. Verify the mechanical abort valve on the LBNPD is closed.

10. Run the procedure to calibrate the pressure A/D data amplifier.

11. Connect chart recorder inputs to the pressure, left and right leg band data outputs on the computer rear bulkhead connectors. All outputs are 0 to 5 volts.

12. When a subject is in the LBNPD, he must be connected to ground through the cable which simulates the subject interface box. This contains a 10K resistor in parallel with a 1560pF capacitor.

Procedure to calibrate the computer power supplies equipment:

1 - DVM

1 - Screwdriver

Procedure:

1. Verify the computer power is off.

2. Remove the top of the computer cabinet.

3. The +5, -5 +12, and -12 power supplies are at the rear of the computer cabinet. Locate each of the supplies.

4. Turn on the computer and use the DVM to verify that the voltages for -5, +12, -12 supplies are within 5%. The +5 supply should be +5.0 to +5.25 volts.

5. Use the screwdriver to bring the voltages within the desired range if necessary.

Procedure to calibrate A/D leg band data amplifiers:

Equipment:

1 - Screwdriver

1 - Power supply 0-5 VDC

1 - DVM

1 or more Card Extenders

Procedure:

1. Turn off computer and power supply.

2. Remove the leg band amplosc card.

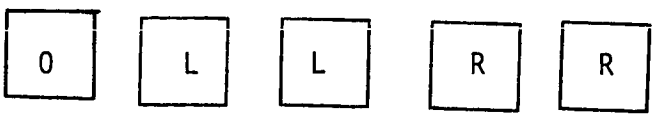
3. Use card extenders to connect power supply and DVM. Negative lead to pin 4 positive lead to pins 7 and 20.

4. Turn on power supply and adjust to 3.5I.01V.

5. Turn on computer and load and execute the following program using the 'C1' command:

LOC	CONTENTS	SYM	OP	ADDR
1100	01 01 00	READ:	LXI	B,0001H
1103	CD 63 07		CALL	STROB
1106	DB 04		IN	LBI
1108	5F		MOV	E,A
1109	01 02 00		LXI	B,0002H
110C	CD 63 07		CALL	STROB
110F	DB 05		IN	LB2
1111	57		MOV	D,A
1112	CD 60 01		CALL	DSDE
1115	C3 00 11		JMP	READ

Output Format:



LL = Left ADC output

RR = Right ADC output

6. Adjust right A/D amp for a reading of b3.

7. Repeat Step 6 for left ADC amp.

8. Adjust power supply to .82 $\pm$ .01V. Verify that output is 2A on both left and right.

9. Other points may be verified using the following equation:

$$OUT = Vin \times (245/5)^*$$

*NOTE: This value is decimal and display is Hex.

Procedure to calibrate pressure A/D data amplifier:

Equipment:

1 - Pressure meter

1 - Card extender

1 - Screwdriver

Procedure:

1. Turn off computer power supply.

2. Remove the pressure A/D card.

3. Connect a card extender and reinsert.

4. Turn on the computer power supply.

5. Push Reset, then 'D00' on the keyboard.

6. Adjust the offset control (R2) until the actual pressure on the display reads zero.

7. Turn on the valve power supply and the vacuum source.

8. Enter 'D50' on the keyboard.

9. Adjust the gain control (R9) until the pressure meter connected to the LBNPD reads 50 TORR.

10. For fine adjustments enter several desired pressures adjusting the offset and gain controls for the best match between the actual pressure display and the pressure meter across the pressure range of zero to 50 TORR.

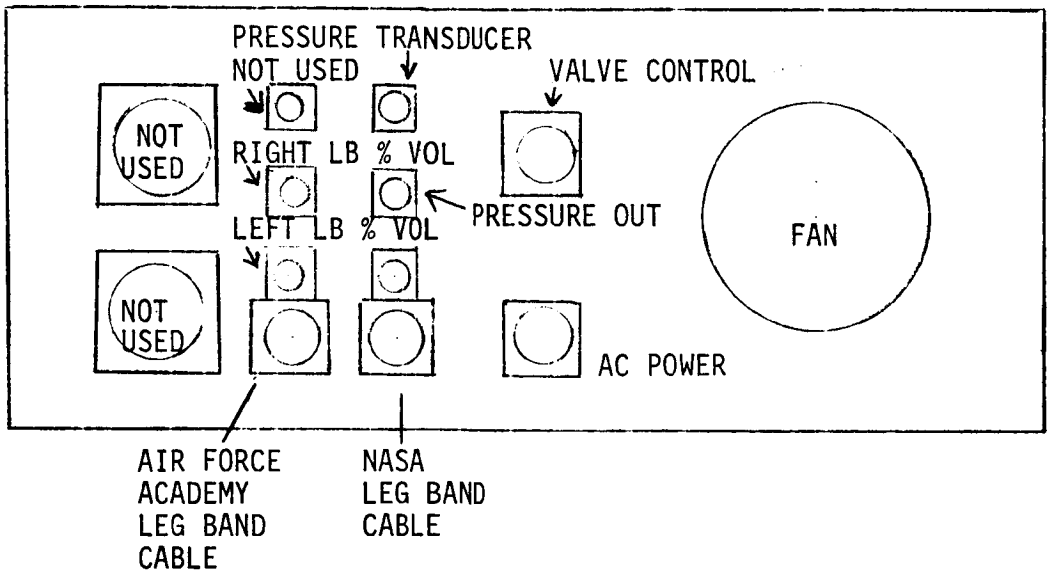


FIGURE A6.1 COMPUTER REAR BULKHEAD CONNECTORS