CONTACT PRESSURE MEASUREMENT

WITH PRESSURIZED FORCE SWITCHES

by

Manuel Vega Perez

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master of Engineering.

McGill University
Montreal, Canada

October, 1983

ABSTRACT

CONTACT PRESSURE MEASUREMENT WITH PRESSURIZED FORCE SWITCHES

A novel, automatic contact pressure measurement system was conceived, implemented, calibrated and analyzed. This system makes use of an array of miniature pistons, pressurized from a common plenum. Contact force impinges on the external, unpressurized face of the pistons which are mounted flush with the surface of the pressure plate containing the array of cylinders in which the pistons slide. As pressure is gradually raised -or reduced- in the plenum, a given piston moves a very small distance, against the local contact force. The pressure that prevails at the instant of the movement is used to determine the contact pressure at the piston that just moved. Because only one pressure transducer is required, this design is quite inexpensive. The system was devised for biomedical application, i.e., contact pressure measurement on a wheel-chair seat. A matrix of 256 piston sensors can be scanned in about 0.4 seconds. A microprocessor controls the pressurization/decompression cycle and takes pressure measurements every time a piston rises, during pressure increase, and a every time a piston falls, while pressure is decreasing.

RESUME

LA MESURE DE LA PRESSION DE CONTACT A L'AIDE D'INTERRUPTEURS DE FORCE PRESSURISES.

Un nouveau systeme automatique de mesure de la pression de contact a ete concu, construit, calibre et analyse. Ce systeme utilise un arrangement de pistons miniatures, pressurises a partir d'une chambre commune. La force de contact agit sur les faces externes, non-pressurises des pistons. Les faces externes, des pistons sont au-niveau avec la surface de la plaque qui contient l'arrangement de cylindres dans lesquels les pistons glissents. Lorsque la pression de la chambre commune est augmentee -ou reduite- chaque piston se deplace par une petite distance dans la direction de la force de contact appliquee. La pression presente a l'instant du mouvement est utilisee pour determiner la pression de contact du piston qui vient juste de se deplacer. Puisque seulement un transducteur de pression est requis, ce systeme est peu couteux. Ce systeme a ete concu pour des applications biomedicales, i.e., mesurer la pression de contact sur le siege d'une chaise roulante. Une matrice de 256 detecteurs a pistons peut etre parcourue dans approximativement 0.4 secondes. Le cycle de compression/decompression qui mesure la pression chaque fois qu'un piston monte lorsque la pression accroit et chaque fois qu'un piston baisse lorsque la pression decroit.

ACKNOWLEDGEMENTS

The work presented in this thesis was carried out under the direction of Dr. Paul J. Zsombor-Murray. The author wishes to express his deepest gratitude to him, for the constant guidance and encouragement received during the course of this study.

Considerable thanks are also due to Mr. Louis J. Vroomen who has always been willing to examine and offer constructive criticisms on the progress of this work; to Mr. George Dedic, Mr. George Tewfik and Mr. Jack Kelly for their valuable help with the experimental tests; to Mr. Louis C. Vroomen for his collaboration with the software development. Finally, the financial support, during all this time, of the Universidad de Santiago de Chile and of the Oficina de Planificacion Nacional is gratefully acknowledged.

To my wife Beatriz and my children Michele and Carlos for providing their unfailing love and support during these two beautiful years.

TABLE OF CONTENTS

PAGE

PAGE

PAGE

CHAPTER 4 EXPERIMENTAL MODEL

PAGE

LIST OF FIGURES

PAGE

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

A	cell area where the air pressure is applied = cell area where external pressure is applied
a	auxiliary parameter = $1/R_1C_1$
b	auxiliary parameter = K/M
c	auxiliary parameter = $X_1K/2(F_1+M_2g)$
d	auxiliary parameter = $X_1KR_1C_1/(F_1+M_2g)$
C_1	plenum capacitance
dm	small change in air mass
dp	small change in air pressure
$d(\Delta p)$	small change in air pressure difference
dp_o	small change in plenum air pressure
dq	small change in air flow
dt	small change in time
e	natural logarithm base
F_1	external force exerted on a cell
F_2	cell weight
F_3	dynamic friction force
g	gravity acceleration
GF	gage factor
K	viscous friction coefficient
K_o	constant value equal to : $a + b + X_1ab/K_1$
K_1	constant value equal to : $(F_1 + M_2g)/MR_1C_1$
L	length

M	cell mass plus equivalent mass of the patient mass to be accelerated, i.e., M_1+M_2
m	plenum air mass
M_1	equivalent mass of the patient mass to be accelerated
M_2	cell mass
p	steady state air pressure
P_1	constant value of the input air pressure
p_1	small change in inflow air pressure
p_o	small change in plenum air pressure
$P_1(s)$	Laplace transform of the input air pressure
P_{max}	maximum air pressure applied to the sensor plenum
$P_o(s)$	Laplace transform of the plenum air pressure
$P_o(t)$	pressure inside the plenum
q	air flow rate
R	electrical resistance
R_1	resistance of the valve restriction
s	complex variable
t	time
t_o	impending motion time at which the external applied pressure is equal to the plenum air pressure
V	plenum volume
x	distance, cell displacement
X_1	constant distance moved by a cell, i.e., trigger distance
$X(s)$	Laplace transform of the distance

Greek symbols

ΔL	change in length
ΔR	change in electrical resistance
ε	strain
δ	density

Abbreviations

ADC	analog to digital converter
DATACQ	program to perform the data acquisition process
DISPLY	program to perform the data displaying process
EOC	end of conversion signal
PIA	peripheral interface adapter
SAVE	subprogram to create a file On diskette with the results of program DATACQ
SOC	start of conversion signal

CHAPTER 1 INTRODUCTION

1.1 General.

During the last 10 years, there has been considerable effort in applying microcomputers to the solution of different medical problems. This tendency can be attributed to three main characteristics of microcomputers: small size, computation capacity and low cost. Applications range from the measurement of a few simple physical variables to the control of complicated apparatus.

An interesting problem in this field is the measurement of the pressure distribution exerted upon supporting regions of the human body. Pressure measurement has been extensively studied and many kinds of sensors based on a number of different physical effects have been developed. However, when the pressure exerted on a surface is not uniform, it is necessary to use many sensors, distributed over the surface, in order to obtain a pressure distribution contour map. Knowledge of this pressure distribution is clearly important in the case of patients who are partially or totally immobilized and must remain in bed or in a wheel chair for extended periods. If regions of excessive contact pressure exist, unrelieved, ulcers or bed sores are inevitably developed on the skin of the patient.

It has been noted that these ulcers are mainly due to high local pressures. Therefore this work is concerned with the prototype development of an economic and efficient microcomputer based method to automatically measure, process and display the pressure distribution exerted by a hypothetical patient on a wheelchair.

1.2 Review of Previous Research.

Ischemic ulcers are a permanent risk for people who have to remain sitting or recumbent for long periods of time(7). These ulcers may lead to sepsis, osteomylitis, mutilating amputations and often death(5). Treatment of decubitus or ischemic ulcers is complex, costly and time consuming; therefore efforts to prevent them have become very important(9). Kosiak(8), described ischemic ulcers as localized areas of necrosis that develop over bony prominences subjected to pressures exceeding capillary pressure for varying periods of time. The areas most commonly affected when patients are seated are the ischial tuberosities and sacrum. Kosiak(9), found that a pressure as low as 70 mm Hg (9.4 kPa) applied to tissue for two hours have produced irreversible changes in cells that ultimately lead to their death. Several efforts to measure contact pressure distribution upon the body have been made during the past few years.

One of the first reports was published in 1958 by Kosiak(7). He placed twelve flat rubber butterfly valves, under a seat, connected to a compressed air reservoir. The pressure exerted at each point was considered to be equal to the minimum pressure at which air escaped from a given valve.

Lindan(10), in 1965, measured contact pressure on the human body by utilizing a "bed of springs and nails". The device consisted of a large flat surface with up to 1000 nails. When the patient's body impinged upon any of these nails it compressed the steel spring beneath each one. Two types of steel springs, with spring constants of 0.165 N/mm and 0.08 N/mm, respectively, were used. A measurement of the pressure on each point was obtained by measuring the displacement of each nail or spring. These measurements were taken, manually, with a millimeter scale. The data from all nails were then used to map pressure distribution contours produced by a patient in the lying or sitting position. In the latter position, pressures of up to 130 mm Hg (17.33 kPa) were observed. Each test lasted between 1 and 1.5 hours. Figure 1-1 shows a patient lying on Lindan's bed.

In 1969 Bush(1) used two commercial transducers to measure the contact pressure over the ischial tuberosities of patients sitting in a wheelchair with hard seats. Each transducer consisted of a small variable capacitor. An electronic circuit detected the change of capacitance. A current proportional to the pressure was read on a microammeter. The transducer had a non-linear response and therefore, a non-linear microammeter readout was used. The

FIGURE 1-1 BED OF SPRINGS AND NAILS USED BY LINDAN

instrument had two scales, 0 to 40 PSI (276 kPa) and 0 to 100 PSI (690 kPa), with a maximum error of ±10 PSI (69 kPa).

Houle(5), in 1969, used a system similar to that utilized by Kosiak in 1958. Hand molded butterfly valves were connected by plastic tubing to a compressed air tank. The pressure necessary to force air through the butterfly valves was considered to be equal to the lateral pressure compressing the valves. The pressure at the reduction valve was kept at about 600 mm Hg (80 kPa) and the flow rate was adjusted with micrometer control to about 7 cc. per minute (0.12 ml/s). Each butterfly valve was connected to the air supply through a needlevalve. Brass manifolds, each accomodating four valves, were arranged at the sides and back of the wheelchair. Pressure on the butterfly valves was sensed by a commercial pressure transducer connected to a recorder. The pressure range used was 0 to 160 mm Hg (21 kPa).

Knapp(6), in 1970, presented a pressure measuring instrument based on the change of electrical capacitance produced when a pressure is exerted on the transducer. This transducer consisted of two copper plates separated by a foam plastic dielectric. Changes in capacitance were detected by an electronic circuit which gave a voltage proportional to the pressure. This instrument provided conveniently high signal levels. 10 PSI (69 kPa) produced an output of 18 V. The pressure signal was read on a voltmeter and the accuracy obtained was ±5%.

Frisina(3) published a preliminary report, on a chemical means for graphically quantifying static pressures, in 1970. The device employed a controlled reaction between an acid indicator and a mild acid. Each constituent was suspended in an absorbent flexible sheet and the treated sheets were stacked to form a composite sandwich. When pressure was applied, the reactants combined. A colour change occurred as a function of pressure and could be sensed visually to obtain immediate qualitative information or it could be recorded with a filtered densitometer if quantitative data were desired.

Kosiak(9), in 1976, published a report concerning a dynamic resting surface which provided sitting patients with local relief of pressure, at regular intervals, for short periods of time. The seat consisted of a set of rollers operating on a continuous belt assembly. The system used a small direct-current motor powered by four rechargeable D cells. Two commercial pressure sensors were placed beneath each ischial tuberosity and the pressures were recorded on chart recorders. Beneath the ischial tuberosities pressure extremes ranged from 0 mm Hg to 160 mm Hg (0 - 21 kPa).

Garfin(4), in 1980, used 65 water filled bladders or balloons connected to pressure transducers to measure the pressure distribution of the human body in the recumbent position. The small (25 mm x 15 mm) rubber balloons were connected to polyethylene tubing by bonding cement and overtied with silk ligatures. The balloons were calibrated and filled with water to a pressure of 10 mm Hg (1.33 kPa).

Until 1982, most methods for measuring seated and recumbent pressure distribution used only one or a few commercial transducers. Simultaneous measurement at many points usually incurs a very high cost in transducers. Some researchers, like Lindan and his associates, built their own fairly inexpensive sensor cells but their accuracy was low. In addition, the results of these tests were generally processed by hand. This is a very tedious and time consuming procedure, when applied to a large array of pressure sensors.

In September 1982, Drummond(2) and his associates presented the first report of a computerized system to measure the pressure distribution of sitting patients. The system used a hard surface with 64 resistive strain gage transducers. These transducers were connected to a LSI-11/02 microcomputer and the results were printed on a LA-36 DEC printer. Bidimensional and tridimensional views of the pressure distribution were obtained on a Calcomp plotter. Obviously, this was a new generation of instrument capable of yielding fast, accurate results with minimum effort. This has been brought about by microcomputer systems whose cost have dramatically decreased during the last few years, making them more affordable for many new applications.

1.3 Scope of the Research Project.

The investigation reported in this work comprises the theoretical and experimental analysis of a novel automated contact pressure measurement system. This report consists of three main sections: the hardware construction, the software development and the performance analysis of a prototype. The hardware construction deals with the sensor system and its theoretical analysis and with the computer hardware. The software must:

-Scan all sensor cells to acquire data and post-process it, and

-Display the pressure distribution graphically.

In general, this system was developed so as to be inexpensive and easily operated by people who are not familiar with computers.

CHAPTER 2 SYSTEM STRUCTURE

2.1 General.

A good system must give accurate and precise measurement of the pressure distribution exerted by a patient on the seat and other surfaces of a wheelchair. This information is usually required in the form of listings and in graphic form showing two and three dimensional views of the distribution. The system must be easy to use and inexpensive.

A conventional approach to the measurement of this pressure distribution is to use many pressure cells distributed uniformly over the contact surfaces. The larger the number of cells, the more detailed the results will be. This number depends on the dimensions of the seat and the dimensions of the sensing surface of each cell. The dimensions of the wheelchair seat range from 380 to 510 mm in both length and width. According to Kosiak and other researchers(5,9,10) the maximum local pressure exerted by sitting patients is less than 300 mm Hg (40 kPa). No correlation has been found between local pressures and body weight or sex.

A microcomputer provides a good way to scan a large number of sensor cells. Microcomputers make digital or analog measurements quickly and accurately.

This chapter describes the development and analysis of a minimum configuration instrument system. The principle of a novel sensor and the design of its scanning circuit are presented in detail.

2.2 A Minimum Configuration.

Every microcomputer configuration requires a microprocessor, memory (ROM, RAM, etc.), support circuitry (power supply, clock, etc.) and various peripheral equipment. Inexpensive, single chip microprocessors, consisting of 20 by 60 mm packages, are currently available. Large capacity, inexpensive ROM and RAM memories, similarly packaged, are also available. Therefore, the microprocessor, ROM and RAM memories and support circuits occupy little space and they are the less expensive part of a complete computer system. Peripheral equipment such as CRT display, keyboard, printer, plotter, diskdrive, etc, are the more expensive parts and they are usually the only components visible to the user.

Two system configuration alternatives were considered and the simpler was implemented.

2.2.1 Option A.

An instrument intended for laboratory use will be installed, more or less permanently where all tests are performed. It requires the following peripherals:

- A keyboard to allow the user to communicate with the computer, i.e., to issue commands and to input test parameters.

- A printer to produce hard copy of results viz., pressure values at each point being measured.
- A CRT display and/or plotter to get a graphic representation of the pressure distribution.
- A disk or diskette unit to provide permanent bulk storage of programs and data.
- The sensor unit.

Figure 2-1 shows a typical microcomputer system architecture with the peripheral equipment indicated above.

2.2.2 Option B.

A small instrument will be installed in wheel chairs to make long duration tests. It will only acquire and store data for subsequent processing.

Option B is more suited to an exhaustive testing program. Several patients, each provided with a separate, microprocessor equipped chair, can be tested simultaneously.

The cost of a computer system for each chair is minimal. Because it is intended only for the acquisition and storage of data its configuration can be simple. The only peripheral, a data storage device, could be a cassette tape recorder. The system must, of course, work automatically. All its programs would be ROM resident and it would have simple front-panel controls. Figure 2-2 shows the configuration of a dedicated system.

Data processing operations are the same in both options

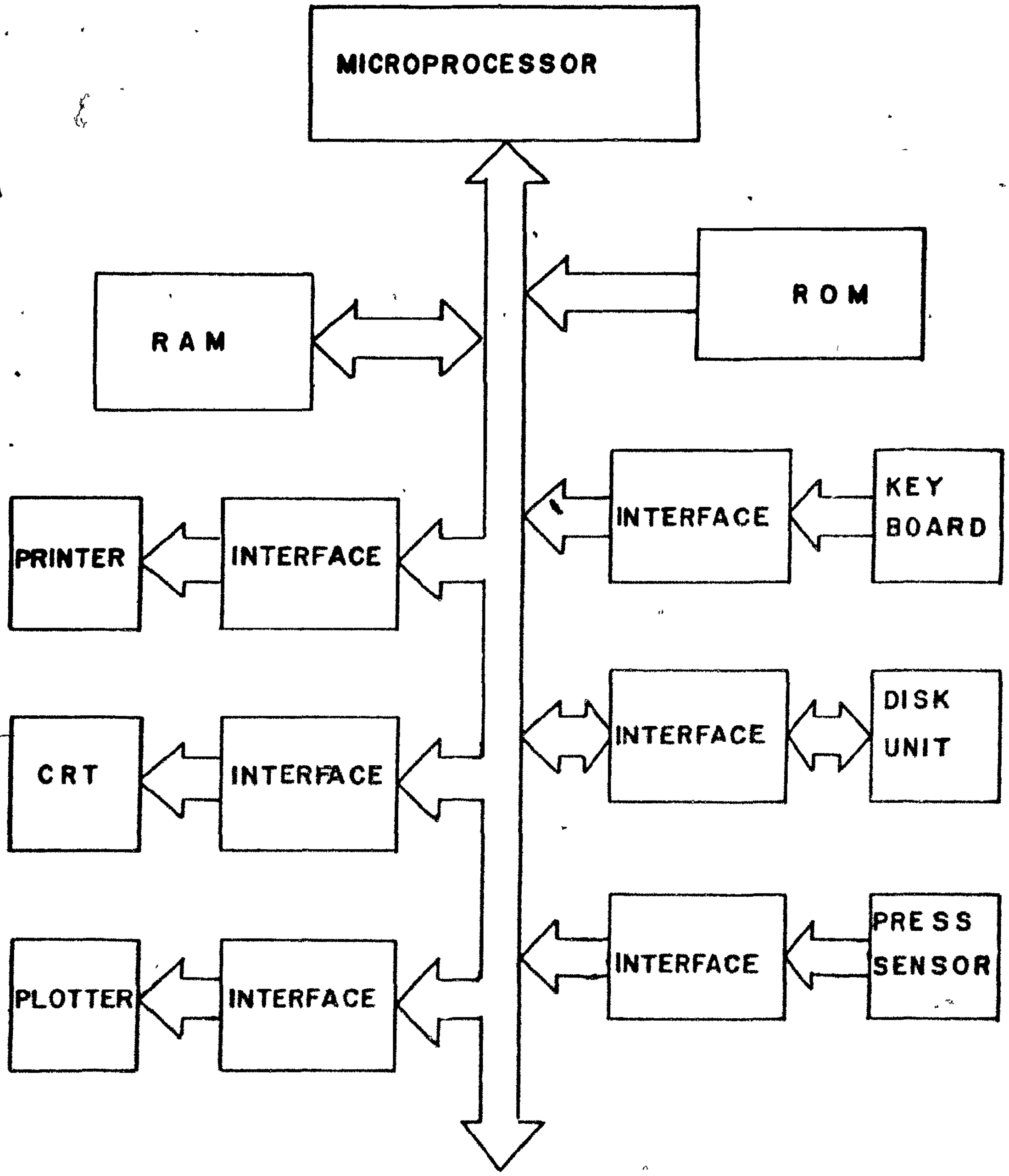

FIGURE 2-1 STRUCTURE OF A MICROCOMPUTERIZED LABORATORY INSTRUMENT

but in option B postprocessing is carried out seperately, in a second stage. On-line postprocessing is sacrificed in favour of data acquisition capacity. Nevertheless, the minimal cost deditated systems each incur the cost of a power supply, possibly with rechargeable batteries. Furthermore, the second, postprocessing stage, is also necessary to transform acquired data into readable form. There exist, too, the additional complications posed by using untried equipment in a clinical environment. It was therefore deemed more convenient to begin by implementing option A.

2.3 The Sensor.

A review of previous research revealed that many kinds of sensors have been used to measure the pressure distribution of body weight. These range from Lindan's bed of nails and springs to Drummond's strain gage sensor.

Some of these used variable capacitance to detect the pressure, others used rubber butterfly valves, while still others tried chromochemical reactions. Most sensors or transducers provide an electrical signal proportional to the pressure measured.

Usually this analog electrical signal has to be amplified so as to drive an instrument or to be measured by a computer controlled ADC. The use of commercial transducers is generally expensive, especially when many

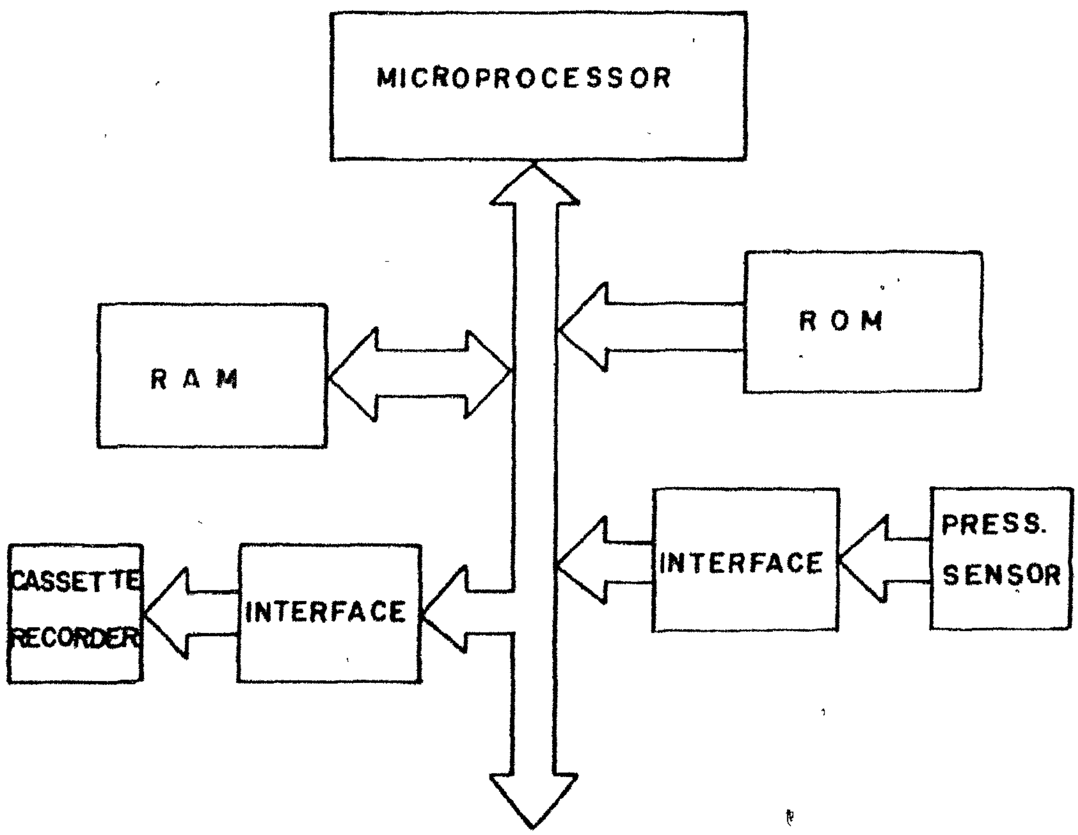

FIGURE 2-2 STRUCTURE OF A MICROCOMPUTERIZED PORTABLE INSTRUMENT

are needed to simultaneously sense a large number of points.

2.3.1 Number of Sensor Cells.

Figure 2-3 shows how the number of pressure sensors to be placed in a seat, was arrived at. Five was considered to be the minimum number which should be placed parallel to the front line of the seat so as to lie within the contact region of each thigh. It was decided to add to this row of five sensors one on the seat edge side and two on the inner side, near the anterior/posterior line of symmetry. This gives 16 cells parallel to the front line of the seat, counting both halves. If the seat is square and a uniform distribution of the sensor cells is desired, the minimum total number of cells is 16 x 16 = 256 cells.

2.3.2 Strain Gages.

Strain gages are widely used in various pressure sensor applications. However, when used singly, they are strongly affected by changes in environmental temperature.

Therefore, it is frequently necessary to arrange two or four gages in a bridge to compensate the effect of temperature changes. There are two main types of strain gages: metallic and semiconductor.

When a strain gage is strained, a change in its electrical resistance occurs. Gage factor is an important strain gage property. It is defined as the ratio of the dimensionless change in resistance to the dimensionless

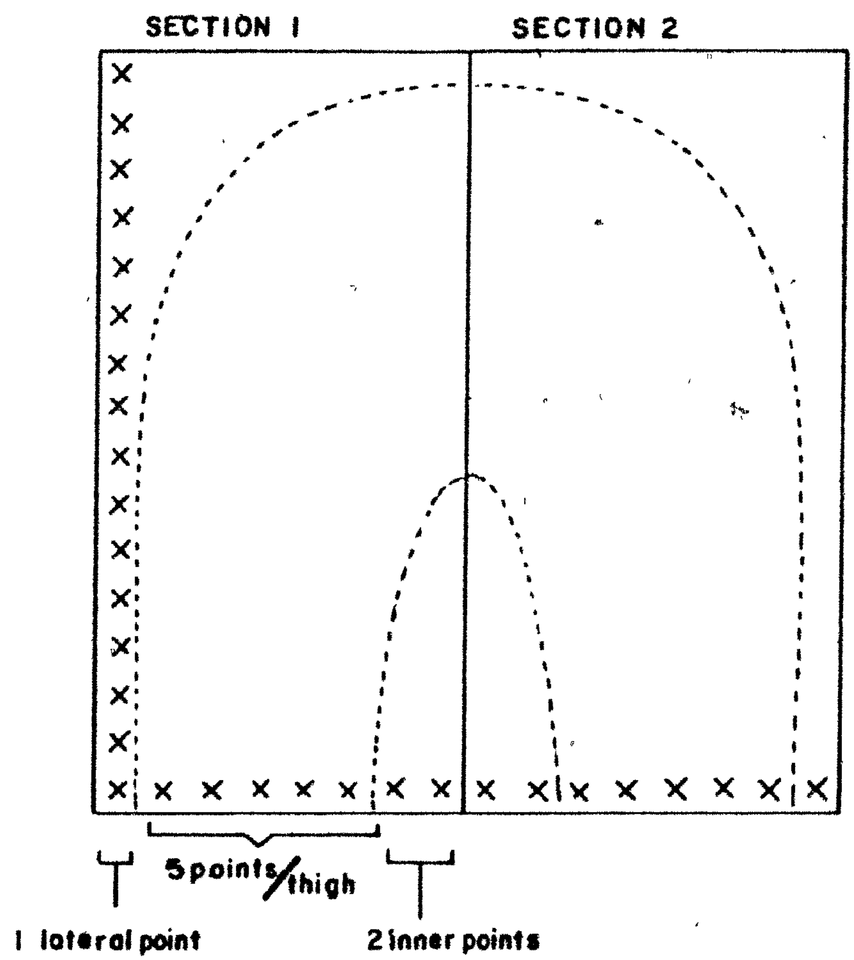

FIGURE 2-3 PATIENT CONTACT SURFACE ON A SEAT

change in length, i.e., strain. Gage factor is therefore a dimensionless quantity and a larger value implies a more sensitive strain gage. Gage factor is expressed in equation form as:

$$GF = \frac{\Delta R/R}{\Delta L/L} = \frac{\Delta R/R}{\epsilon} \qquad (1)$$

Metallic strain gages have a low gage factor. Two is a typical gage factor for commercial types. This means that for a small strain, the output voltage is usually too low and therefore, the signal-to-noise ratio is often unacceptable. Semiconductor strain gages on the other hand, have a higher gage factor. Typical values are 130 to 150. Sensors designed with these gages provide very good resolution even at low strain values. But unfortunately these devices cost over $10 each in 1983, and they are particularly prone to temperature drift.

When the number of cells is large, the system becomes very expensive. Besides, the process of cementing the strain gages to the sensor support is critical and requires careful surface preparation and adhesive techniques. Improper cementing is a common cause of measurement problems. Commercially mounted gages are expensive. Therefore a novel method to measure pressure distribution has been proposed. It promises good resolution, is not affected by noise and is inexpensive.

Since this system is to be computer controlled, a digital pressure transducer, if such were available, would be an attractive way to acquire pressure distribution data. The proposed sensor is almost digital in the sense that the output of each sensor cell is a binary variable, i.e., a low or high voltage signal. An analog pressure transducer is nevertheless required. A good pressure transducer is expensive but the system requires only one. It is highly desirable that the numerous digital sensors be inexpensive.

2.3.3 The Proposed Sensor.

The principle of the proposed sensor cell is that of a pneumatic pressure switch. Pressure switches are normally designed to close a contact when a specified pressure is applied. They are often used as safety devices. Figure 2-4 shows a sensor plate which consists of several pistons or cells distributed uniformly, on a square pitch, over its surface. Each cell consists of a piston/cylinder assembly which operates a displacement detector. Each piston can move up and down through a very small displacement; the minimum required to operate a motion detector.

Figure 2-5 shows a diagram of a typical sensor cell. It operates in the following way. At the beginning of a measurement an air pressure greater than the maximum local pressure to be measured is applied to the plenum under the sensor plate. This causes all pistons to rise and to remain in this raised position. Then, a solenoid valve closes the compressed air admission port and opens a restricted vent through which the air bleeds to atmosphere. The capacitance of the sensor plate plenum, defined on p.24, and the resistance of the restriction, defined on p.22, form an approximate first order system. The time constant is given by the product of the capacitance of the plenum and the resistance of the restriction. Therefore, the air pressure decreases exponentially from the precharged maximum value to a minimum value, which corresponds to the atmospheric pressure.

During the test, the computer continuously scans the air pressure in the plenum. If the air pressure is decreasing, there will be an instant at which the external pressure exerted on a particular cell, is equal to the air pressure. Therefore, this air pressure measurement is the pressure exerted on that cell. Afterwards the air pressure falls below the external pressure and the piston starts to move down. If the elapsed time, between the instant that the movement begins until the instant when it is detected, is known, then the corrected pressure exerted on a particular cell can be obtained. A similar situation, requiring a slightly different correction procedure, occurs if the air pressure is increasing, i.e., during a charging cycle.

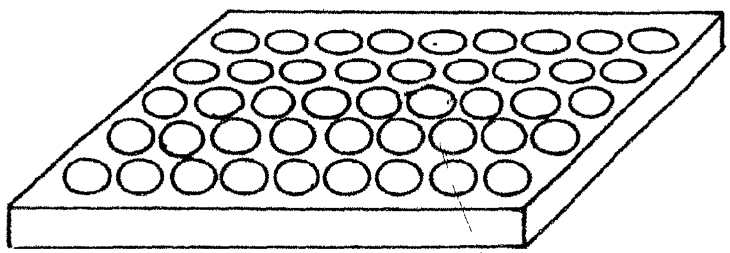

FIGURE 2-4 SENSOR PLATE

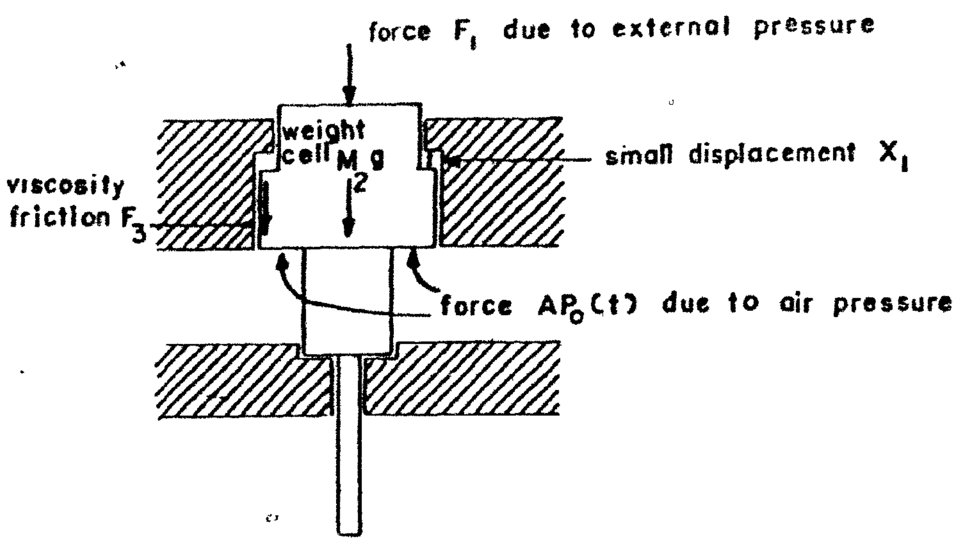

FIGURE 2-5 SENSOR CELL

2.4 Dynamic Analysis of a Cell.

A sensor cell is shown in Figure 2-5. The air pressure exerts an internal force equal to $AP_o(t)$ which is opposite to the external force F_1, exerted by the patient, the cell weight F_2 and the dynamic friction force F_3.

By applying Newton's second law, the motion of a cell may be expressed by the following equations:

$$AP_o(t)-F_1-F_2-F_3 = (M_1+M_2)\frac{d^2x(t)}{dt^2} \qquad (2)$$

when pressure is increasing and:

$$F_1+F_2-F_3-AP_o(t) = (M_1+M_2)\frac{d^2x(t)}{dt^2} \qquad (3)$$

when air pressure is decreasing.

Force F_1 is considered to be constant during a test. Force F_2 is equal to M_2g. M_1 is included to acknowledge that some small region of tissue must be moved by the piston face. The modeling of this difficult aspect of sensor dynamics was not attempted. However its effect, though not negligible, is believed to be small.

If the external force is approximated by the weight of a single rigid body, e.g., a calibrating weight placed on a

sensor cell, then the mass of this body undergoes accelleration together with that of the piston.
Finally the dynamic friction F_3 is equal to:

$$F_3 = K\frac{dx(t)}{dt} \qquad (4)$$

Therefore equations (2) and (3) may be rewritten as:

$$AP_o(t)-F_1-M_2g = M\frac{d^2x(t)}{dt^2} + K\frac{dx(t)}{dt} \qquad (5)$$

$$F_1+M_2g-AP_o(t) = M\frac{d^2x(t)}{dt^2} + K\frac{dx(t)}{dt} \qquad (6)$$

2.4.1 Plenum Air Pressure.

Consider the derivation of the expression for the air pressure $P_o(t)$. According to the pressure system model shown in Figure 2-6, the air flow, q, through the restriction is a function of the pressure difference p_1-p_o. The air flow resistance is defined as:

$$R_1 = \frac{\text{change in air pressure difference}}{\text{change in air flow rate}}$$

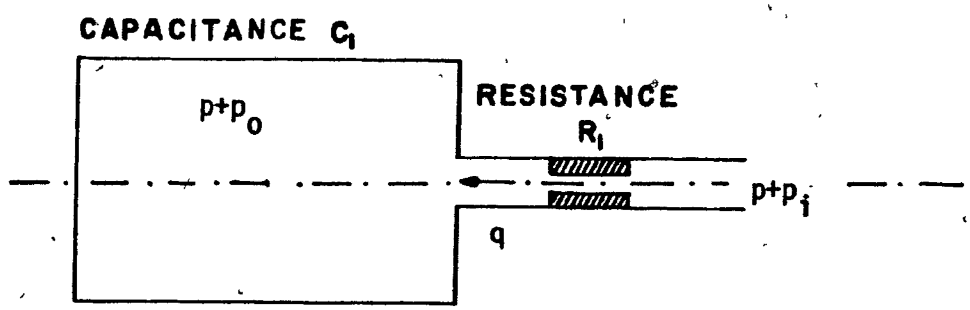

FIGURE 2-6 PRESSURE SYSTEM MODEL

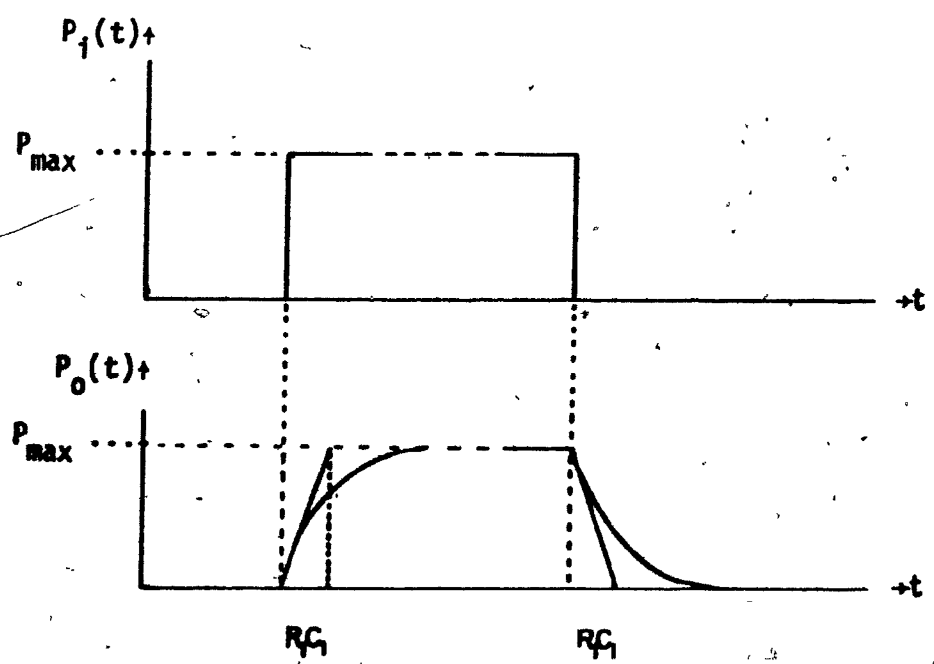

FIGURE 2-7 VARIATION OF PLENUM AIR PRESSURE WHEN A STEP FUNCTION IS APPLIED

$$R_1 = \frac{d(\Delta p)}{dq} \qquad (7)$$

where $d(\Delta p)$ is a small change in the air pressure and dq is a small change in the air flow. The capacitance of the plenum is defined as:

$$C_1 = \frac{\text{change in mass of air stored}}{\text{change in air pressure}}$$

$$C_1 = \frac{dm}{dp} = \frac{Vd\delta}{dp} \qquad (8)$$

The capacitance of the pressure system depends on the type of expansion process involved but it may be considered constant since the pressure changes are small and the process is approximately isothermal.

The system shown Figure 2-6 may be considered linear if small deviations in the variables from their respective steady-state values are assumed. In this case, the resistance R_1 may be considered constant and therefore may be expressed as :

$$R_1 = \frac{p_1 - p_o}{q} \qquad (9)$$

$$dq = \frac{dm}{dt} \qquad (10)$$

$$C_s dp_o = dm = dq \cdot dt$$

$$C_s \frac{dp_o}{dt} = dq = \frac{d(\Delta p)}{R_s} = \frac{p_i - p_o}{R_s}$$

$$R_s C_s \frac{dp_o}{dt} + p_o = p_i \qquad (11)$$

Therefore, the transfer function becomes:

$$\frac{P_o(s)}{P_i(s)} = \frac{1}{R_s C_s s + 1} \qquad (12)$$

where $P_o(s)$ is the Laplace transform of the time function of the internal air pressure and $P_i(s)$ is the Laplace transform of the time function of the external air pressure.

By assuming that solenoid valves open or close instantaneously, the air pressure applied to the system may be validly approximated by a step function. Then the air pressure inside the plenum is:

$$P_i(s) = P_i/s \qquad (13)$$

P_i = constant

Replacing this value in equation 12:

$$P_o(s) = \frac{P_i}{s(R_iC_is+1)} \qquad (14)$$

and therefore:

$$P_o(t) = P_i(1-e^{-t/R_iC_i}) \qquad (15)$$

Figure 2-7 shows the variation of the plenum air pressure when a step function is applied to the input.

One way to establish the external pressure exerted on a cell is to determine the time, t_o, at which the air pressure is equal to the external pressure. At t_o the piston movement is impending, but the piston is stationary. Therefore, it is highly desirable that the movement detector operates in a minimum distance of piston travel, i.e., within a minimum delay from t_o.

Since piston motion, however small, requires some time to occur, the air pressure measured at this time, after motion has taken place and has been detected, will be different from the external pressure exerted on the cell at t_o. Nevertheless, if the measured air pressure is proportional to the external pressure, then this external or contact pressure can be readily inferred from the air pressure which prevails when piston motion is detected. It is shown in what follows that this proportionality exists and the factors that affect it can be determined. The following analysis considers that the air pressure is decreasing. Therefore, the air pressure is:

$$P_o(t) = P_{max} e^{-t/R_1C_1} \tag{16}$$

2.4.2 Time of Impending Motion.

The time, t_o, at which the motion begins may be determined starting from the following condition:

AIR PRESSURE = EXTERNAL PRESSURE

$$P_{max} e^{-t_o/R_1C_1} = \frac{F_1+M_2g}{A} \tag{17}$$

The dynamic friction force is 0 at time t_o, then:

$$e^{-t_o/R_1C_1} = (F_1+M_2g)/AP_{max} \tag{18}$$

$$-t_o/R_1C_1 = \log((F_1+M_2g)/AP_{max})$$

and finally

$$t_o = R_1C_1 \log \frac{AP_{max}}{F_1+M_2g} \tag{19}$$

The time, t_o, for a particular cell depends on the time constant of the pressure system, the maximum air pressure P_{max} and the external force F_1 applied to that cell ,pl 41 plus the cell weight. t_o is independent of A because this area is the same on the external as well as the internal piston face.

2.4.3 Distance Moved by a Cell.

The distance moved by any particular cell may be obtained by considering that movement begins at t = 0. Equation 6 becomes:

$$F_1+M_2g-AP_{max}e^{-(t_o+t)/R_1C_1} = M\frac{d^2x(t)}{dt^2} + K\frac{dx(t)}{dt} \qquad (20)$$

$$F_1+M_2g-AP_{max}e^{-t_o/R_1C_1}e^{-t/R_1C_1}$$

$$= M\frac{d^2x(t)}{dt^2} + K\frac{dx(t)}{dt} \qquad (21)$$

replacing the value of e^{-t_o/R_1C_1} obtained in equation 18:

$$F_1+M_2g-AP_{max}\frac{F_1+M_2g}{AP_{max}}e^{-t/R_1C_1} =$$

$$M\frac{d^2x(t)}{dt^2} + K\frac{dx(t)}{dt} \qquad (22)$$

then

$$(F_1+M_2g)(1-e^{-t/R_1C_1}) = M\frac{d^2x(t)}{dt^2} + K\frac{dx(t)}{dt} \qquad (23)$$

at t=0, x(t)=0, x'(t)=0 and x"(t)=0.

The Laplace transform of equation 18 is

$$\frac{(F_1+M_2g)/R_1C_1}{s(s+1/R_1C_1)} = s(Ms+K)\ X(s) \qquad (24)$$

and finally

$$X(s) = \frac{(F_1+M_2g)/MR_1C_1}{s^2(s+1/R_1C_1)(s+K/M)} \qquad (25)$$

replacing:

$K_1 = (F_1+M_2g)/MR_1C_1$

$a = 1/R_1C_1$

$b = K/M$

X(s) becomes

$$X(s) = \frac{K_1}{s^2(s+a)(s+b)} \qquad (26)$$

The distance moved by a piston is obtained, by applying the inverse Laplace transform, as:

$$x(t) = \frac{K_1}{a^2b^2}\left[\frac{1}{a-b}\ (a^2e^{-bt}-b^2e^{-at}) + abt-a-b\right] \tag{27}$$

2.4.4 Time to Trigger the Motion Detector.

The time to move a cell the minimum distance to trigger the motion detector may be obtained from equation 27 as:

$$\frac{X_1a^2b^2}{K_1} + a + b = \frac{1}{a-b}(a^2e^{-bt}-b^2e^{-at}) + abt \tag{28}$$

replacing the left hand side by

$$K_o = \frac{X_1a^2b^2}{K_1} + a + b \tag{29}$$

$$K_o = \frac{a^2}{a-b}\ e^{-bt} - \frac{b^2}{a-b}\ e^{-at} + abt \tag{30}$$

and multiplying this equation by e^{at}

$$K_oe^{at} = \frac{a^2}{a-b}\ e^{at-bt} - \frac{b^2}{a-b}\ e^{at-at} + abte^{at} \tag{31}$$

Equation 31 may be solved by using a Taylor series:

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \ldots$$

For small x, e^x may be approximated by: $e^x = 1 + x$

For $x < 0.1$ this approximation sustains an error of less than 0.5%. Therefore, the following constraints must be satisfied:

i. $0 < at < 1/10$ $\qquad a = 1/R_1C_1$

ii. $0 < (at-bt) < 1/10$ $\qquad b = K/M$

The first constraint states that the time delay t must be less than $0.1R_1C_1$. The time constant is positive by definition and the series approximation of e^x implies that the distance required to trigger the movement detector must be small. The second constraint establishes a relation between two first order system time constants, i.e., that of the plenum pressure, R_1C_1, and the dynamic constant of a piston, M/K. It will be shown that the experimental prototype easily satisfies this condition.

This relationship is :

$(at-bt) > 0$ implies $a > b$ which means $R_1C_1 > M/K$

From the first constraint, $at < 1/10$ so if $a > b$ then,

$at - bt < 1/10$

The approximate series expansion of equation 31 becomes:

$$K_o(1+at) = \frac{a^2}{a-b}(1+at-bt) - \frac{b^2}{a-b} + abt\ (1+at) \qquad (32)$$

Rearranging equation 32:

$$a^2bt^2 + t\left(ab - \frac{a^2b}{a-b} + \frac{a^3}{a-b} - aK_o\right) + \frac{a^2-b^2}{a-b} - K_o = 0 \qquad (33)$$

which can be reduced to:

$$t^2 + t\,\frac{a+b-K_o}{ab} + \frac{a+b-K_o}{a^2b} = 0 \qquad (34)$$

replacing K_o:

$$t^2 - t\,\frac{X_1ab}{K_1} - \frac{X_1b}{K_1} = 0 \qquad (35)$$

and replacing the value of K_1, a and b:

$$t = \frac{X_1K}{2(F_1+M_2g)} \pm \sqrt{\frac{X_1^2K^2}{4(F_1+M_2g)^2} + \frac{X_1K}{F_1+M_2g}R_1C_1} \qquad (36)$$

t cannot be negative and therefore,

$$t = \frac{X_1K}{2(F_1+M_2g)} + \sqrt{\frac{X_1^2K^2}{4(F_1+M_2g)^2} + \frac{X_1K}{F_1+M_2g}R_1C_1} \qquad (37)$$

2.4.5 Pressure at Instant of Movement Detector Operation.

The air pressure measured when the movement detector is triggered, is the pressure at time t_o+t, where t_o represents the time elapsed from the instant that the air pressure is applied to the system until the air pressure is equal to the external pressure exerted on the cell, including the weight of the cell. t represents the time elapsed from that instant until the cell movement is detected. Then the air pressure which exists when the movement detector triggers is:

$$P_o(t) = P_{max}\, e^{-(t_o+t)/R_1C_1} \tag{38}$$

or

$$P_o(t) = P_{max} e^{-t_o/R_1C_1}\, e^{-t/R_1C_1} \tag{39}$$

but according to equation 18:

$$e^{-t_o/R_1C_1} = (F_1+M_2g)/AP_{max}$$

so:

$$P_o(t) = \frac{(F_1+M_2g)}{A}\, e^{-t/R_1C_1} \tag{40}$$

and

Eq. 37

$$t = \frac{X_1 K}{2(F_1+M_2g)} + \sqrt{\frac{X_1^2K^2}{4(F_1+M_2g)^2} + \frac{X_1K}{F_1+M_2g} R_1C_1}$$

Therefore, the measured pressure depends on the external force F_1 plus the cell weight, the viscous friction coefficient K, the trigger distance X_1 and the time constant of the pressure system R_1C_1.

A similar analysis for the case of increasing pressure yields the following expressions which are derived in Appendix A.

$$P_o(t) = P_{max}(1 - e^{-(t_o+t)/R_1C_1}) \quad (41)$$

$$P_o(t)=P_{max} - P_{max}e^{-t_o/R_1C_1}e^{-t/R_1C_1} \quad (42)$$

$$e^{-t_o/R_1C_1} = (AP_{max}-F_1-M_2g)/AP_{max} \quad (43)$$

then

$$P_o(t) = P_{max} - \frac{AP_{max}-F_1-M_2g}{A} e^{-t/R_1C_1} \quad (44)$$

and

$$t=\frac{X_1K}{2(AP_{max}-F_1-M_2g)}+\sqrt{\frac{X_1^2K^2}{4(AP_{max}-F_1-M_2g)^2}+\frac{X_1K}{AP_{max}-F_1-M_2g}R_1C_1} \quad (45)$$

Measured pressure for the increasing case depends on the same factors as for the decaying case. It also depends on the value of the maximum air pressure since the exponential pressure/time characteristic is asymptotic to P_{max}.

2.4.6 Analysis of the Pressure Expression.

Air pressure at time t_o is equal to the external pressure. However, due to the time delay required to trigger the motion detector, the measured pressure is different from the applied pressure. Suppose that the time delay is the same for all values of pressure to be measured.

Then, the difference between the external applied pressure and the pressure measured at the instant of motion detector triggering will be higher for higher pressure values. This is shown in Figure 2-8.

However, the time delay is not constant but varies inversely with the applied pressure, or force F_1. This is described by equation 37, for decaying pressure and by equation 45 for increasing pressure.

Eq. 37

$$t=\frac{X_1K}{2(F_1+M_2g)}+\sqrt{\frac{X_1^2K^2}{4(F_1+M_2g)^2}+\frac{X_1K}{F_1+M_2g}R_1C_1}$$

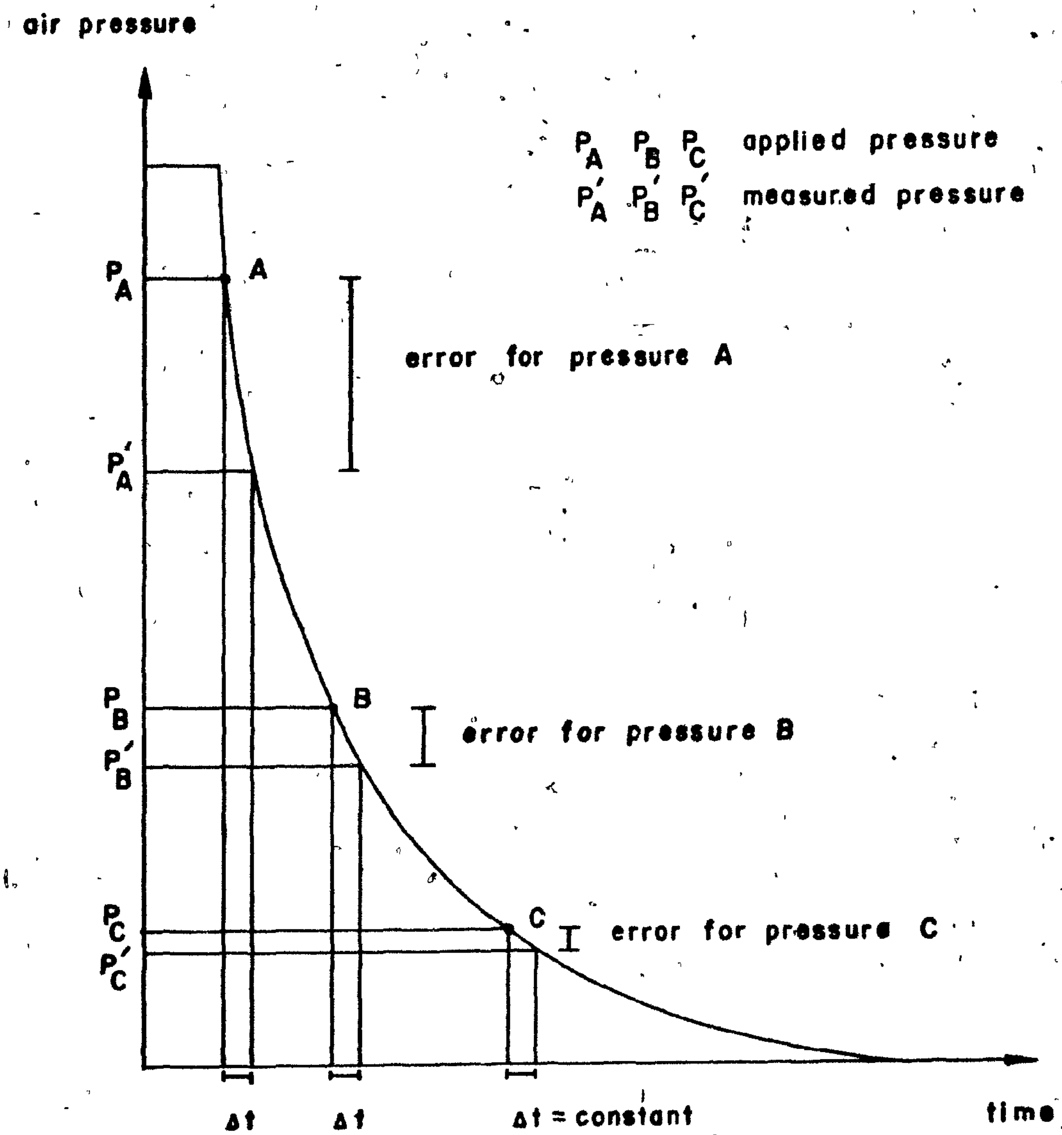

FIGURE 2-8 EFFECT OF A CONSTANT TIME DELAY

$$c = \frac{X_1 K}{2(F_1 + M_2 g)}$$

$$d = \frac{X_1 K}{F_1 + M_2 g} R_1 C_1$$

This equation shows that when the external pressure increases, i.e., F_1 increases, the time delay decreases.

Typical values for c and d are in the range of 10^{-3}, therefore, the value c^2 is very small compared with d and it may be neglected. In the same way, c is very small compared with d and it may also be neglected. Then, the expression for the time delay can be simplified to:

$$t = \sqrt{\frac{X_1 K}{F_1 + M_2 g} R_1 C_1} \qquad (48)$$

This means that the time delay varies approximately as the inverse of the square root of the external force or pressure. Evaluation of this expression shows that the air pressure at the instant of the movement detector operation is approximately proportional to the external applied pressure. Deviation from linearity is less than 0.5% for practical values of the system parameters.

2.5 The Scanning Circuit.

To detect when a cell triggers and to associate the prevailing air pressure to that cell, the air pressure must be scanned continuously during a test. It is possible to use a variety of scanning circuits to determine the state, i.e., open or closed, of the movement detectors. One of the simplest methods makes use of a 256:1 multiplexer. This method checks the state of the detectors sequentially, one at a time. The data required to control the multiplexer may be generated by software. However, as an 8-bit micro-computer with 8- and 16-bit registers is being used, it is more efficient to check the state of 8 or 16 detectors at a time.

Another alternative is to arrange the sensor cells in a matrix, for example 16 x 16, as is frequently done for a keyboard decoder. This solution requires 16 lines for addressing the matrix and the output is obtained on another 16 lines.

A circuit, to scan the state of 256 detectors using multiplexers, is shown in Figure 2-9. This circuit requires sixteen 16:1 multiplexers. To control the scanning of detector data, a 4-bit binary counter is used.

Therefore, by using the 16-bit indexing registers of the microprocessor to acquire all status information, the movement detector states will be contained in 16 sequential, 2-byte words each time an entire scan is completed. It was decided to vary the scanning rate by software which alternates pressure measurements and cell status scan cycles. In this way it was possible to tailor the scanning rate so that acquired pressure and status data will be stored in a table of fixed size at approximately equal intervals of pressure from P_{max} to 0. The choice of the best scanning procedure depends on a tradeoff between scanning speed and the most effective use of memory available for data storage. For optimum resolution of transient pressure distributions, entire scans must be done at the fixed, maximum rate possible. This results in data acquired at equal intervals of time, rather than pressure, and pressure intervals between successive scans would become larger as the test progressed.

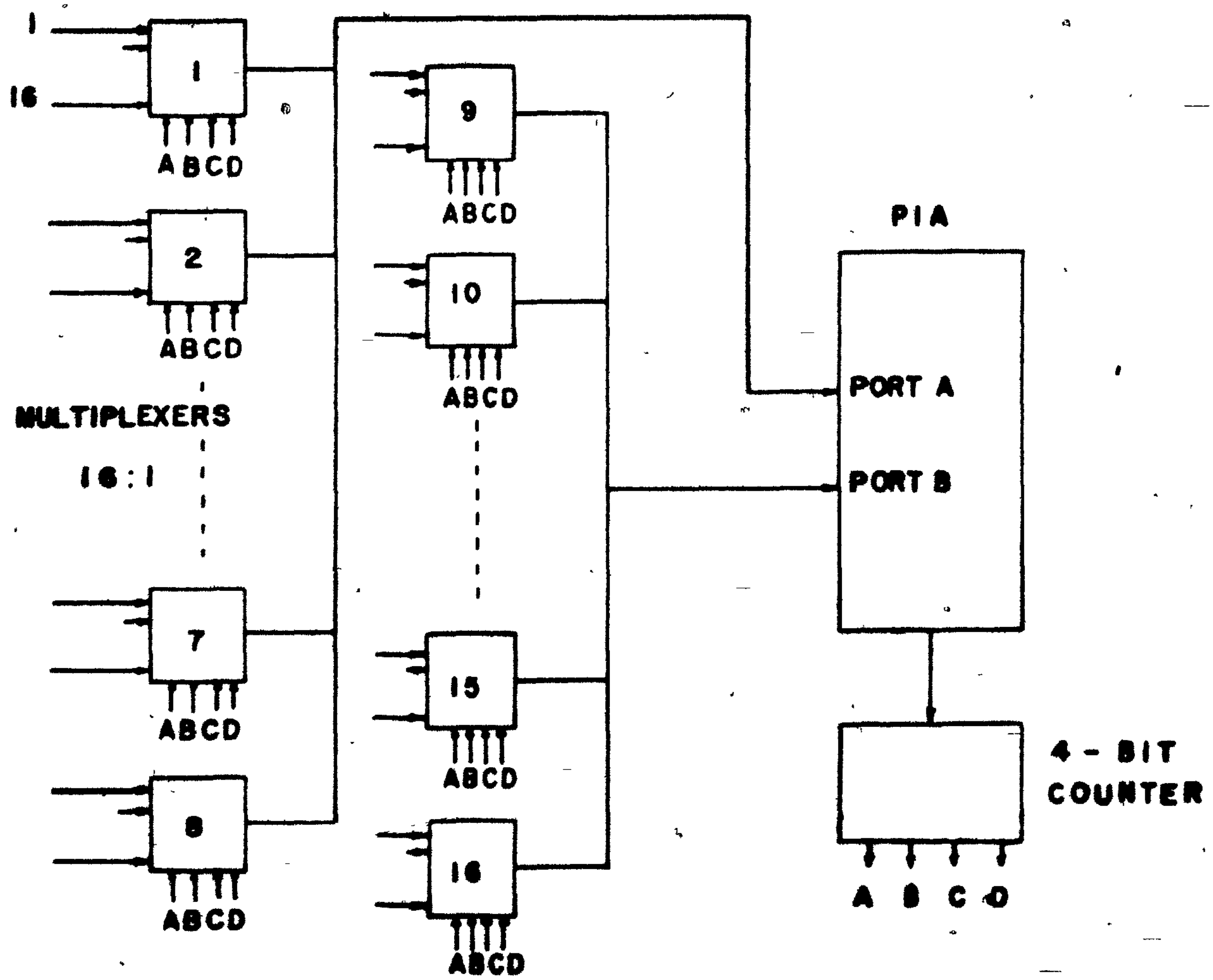

FIGURE 2-9 SCANNING CIRCUIT

CHAPTER 3 SOFTWARE STRUCTURE

3.1 Introduction.

A test consists of two separate procedures:

- THE DATA ACQUISITION PROCESS scans the status of all sensor cells to detect when a particular cell undergoes a transition and to associate it with the prevailing air pressure at the instant of transition. As a result of this process, 256 sequential memory locations contain 256 values of pressure corresponding to the pressure exerted on each of the sensor cells. The program, DATACQ, which performs this task was written in Assembly language so as to achieve the fast, efficient bit manipulation which is necessary if useful data are to be obtained with this instrument.

- THE DATA DISPLAY PROCESS uses the result of the data acquisition process to present the pressure distribution to the operator. The program DISPLY performs this task and it was written in BASIC. BASIC conveniently implements graphics and is much simpler to modify and debug than Assembly language.

3.2 The Data Acquisition Process.

This process can be divided into two phases :

- DATA READING, which involves the examination of the status of the cells and the reading of the air pressure.

- **DATA PROCESSING**, which involves the detection, using cell status data, of cell transitions and the association of the appropriate air pressure reading with each transition.

Two alternative implementations of this process are considered as follows:

3.2.1 Alternative A.

Read and process data, one byte at a time: This means that a byte, containing bit images of the status of eight cells out of 256, is read and immediately compared with the most recent, previous, image of the same eight cells in order to detect any cell transition which may have taken place in the interval between successive examinations. The air pressure is read and stored in one memory location. Every time the ADC generates an EOC signal, the program, in response to the ensuing interrupt, updates that memory location. In this procedure the cells are scanned for possible transition events between succesive pressure measurements.

Optionally, the air pressure may be updated more slowly, and at a non-uniform rate, by not using the EOC interrupt but by programming a fixed scan cycle loop which identifies cell transitions. This procedure measures pressures in response to one or more cell transitions which have been detected during a scan cycle. The cell scan/measurement process continues until the air pressure has decayed to a minimum value.

This solution requires minimum data memory because pressure measurements are associated with cell transitions in real time. Cell transition data are processed immediately and therefore, it is not necessary to store successive bit maps of all cell states, only a single cell transition pressure matrix is required.

3.2.2 Alternative B.

Read and store the matrix of 256 cell state bit images in successive blocks of 32 bytes. In this procedure each block may correspond to a specific, implicit plenum pressure. On the other hand, a pressure reading may be taken and stored before and after each block scan. A certain preselected number of block scans may be taken so as to constitute a complete test cycle between two preselected extremes of plenum air pressure. After the test cycle is completed, these data are postprocessed so as to associate the individual cell transition with the appropriate plenum pressure. Recall that in Alternate A the pressure/cell transition association was carried out in real time. Alternate B clearly requires far more memory space, but the time required to complete a test is much shorter. The data processing is done after the test is finished. It is important to carry out a test cycle as quickly as possible in order to reduce transient effects, i.e., movement of the patient and the interval between the detection of a cell transition and the acquisition of the corresponding pressure data. Alternative B was selected as the more appropriate because the spurious influence of the transient effects were considered more critical than was the disadvantage of limited memory test data storage. Notice that alternative B strikes a compromise between A, i.e.,slowest testing with minimum memory required and a third alternative; scanning at fixed time interval, as described in 2.5 above,i.e., fastest testing with maximum memory required.

3.2.3 Program Design.

Alternative B acquires data first and processes it later. The system scans the status of all the cells 400 times; each scan yields 32 bytes of cell transition data. The program must complete a scan in the shortest time possible. A Motorola 6809 microcomputer was used and the programs were written in Motorola 6809 Assembly language.

Figure 3-1 shows a flowchart of program DATACQ. The program consists of three main parts: the Initialization routine, the Data Reading routine and the Data Processing routine.

A.-Initialization Routine. During the initialization stage, variables, pointers and reference values are defined. PIA interfaces are initialized. The counter is reset and the first SOC signal is sent. Then, the first pressure reading is taken. If the air pressure is at its nominal initial value, a command to open the output valve is issued. A delay, required to execute this command, completes the

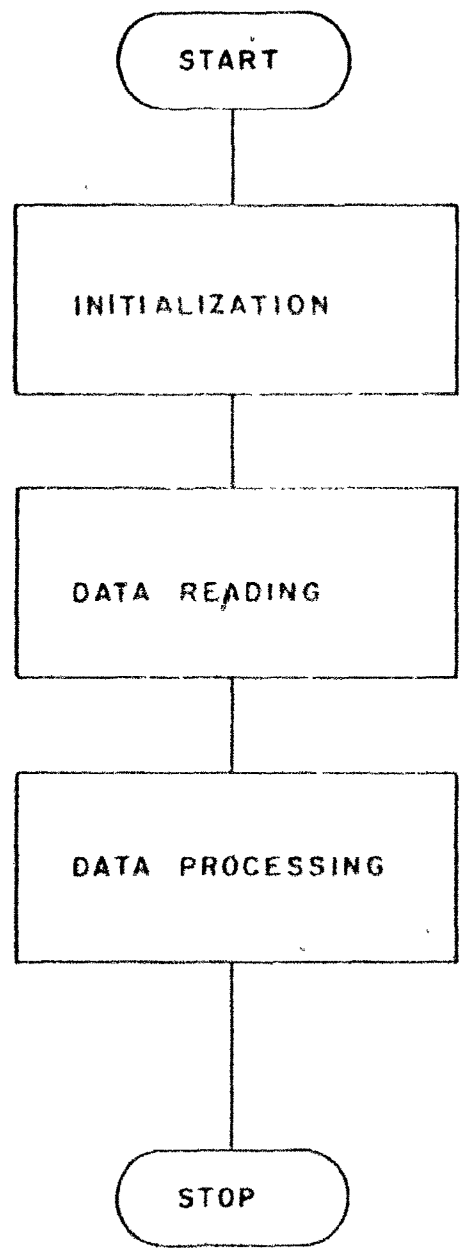

FIGURE 3-1 FLOWCHART OF PROGRAM DATACQ

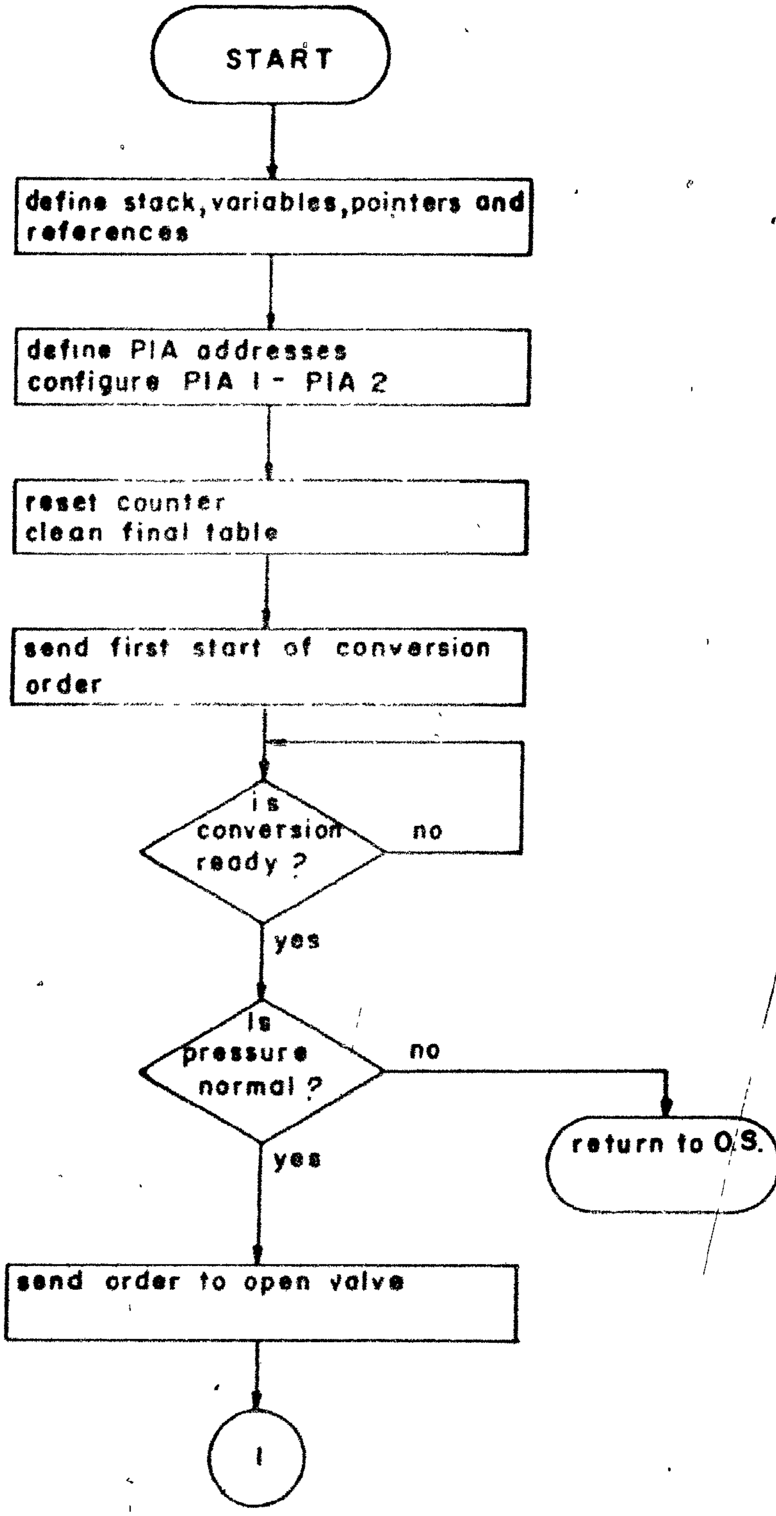

FIGURE 3-2 FLOWCHART OF INITIALIZATION ROUTINE

initialization stage. If the air pressure is not at its nominal initial value, a message, indicating this malfunction condition is printed and the program returns control to the operator.

Figure 3-2 shows a flowchart of the initialization routine.

B.-Data Reading Routine. The data reading phase begins with an SOC signal. The program reads the status of cells through port A and B of PIA #2. This data is stored in the address given by indexing register Y and the counter that controls the multiplexers, shown in Figure 2-9, is incremented. The time required by the counter to increment and for the multiplexer to respond is less than 1 us, therefore it is not necessary to use a wait loop. The next two bytes of cell transition data are read and stored sequentially. This process is repeated 16 times in order to perform a whole scan cycle. Figure 3-3 shows a flowchart of the data reading routine.

After a complete scan, the air pressure is read and stored. Indexing register X contains the air pressure address. The elapsed time between issuing the SOC signal and reading the air pressure is substantially longer than the time required for the ADC to make a conversion. The program issues a new SOC command only after reading and storing all switch position images. Therefore, updated air pressure data will always be ready and available for storage. The test is finished when the air pressure has been read 400

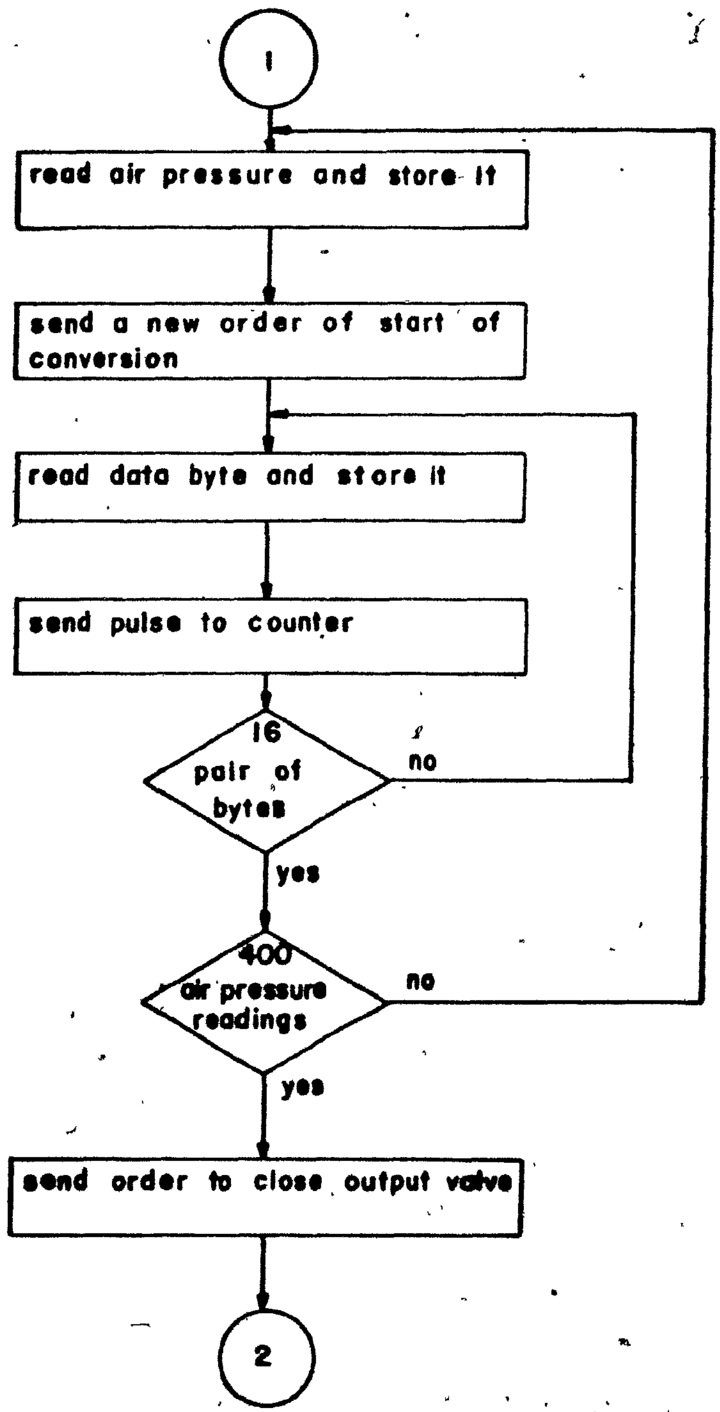

FIGURE 3-3 FLOWCHART OF DATA READING ROUTINE

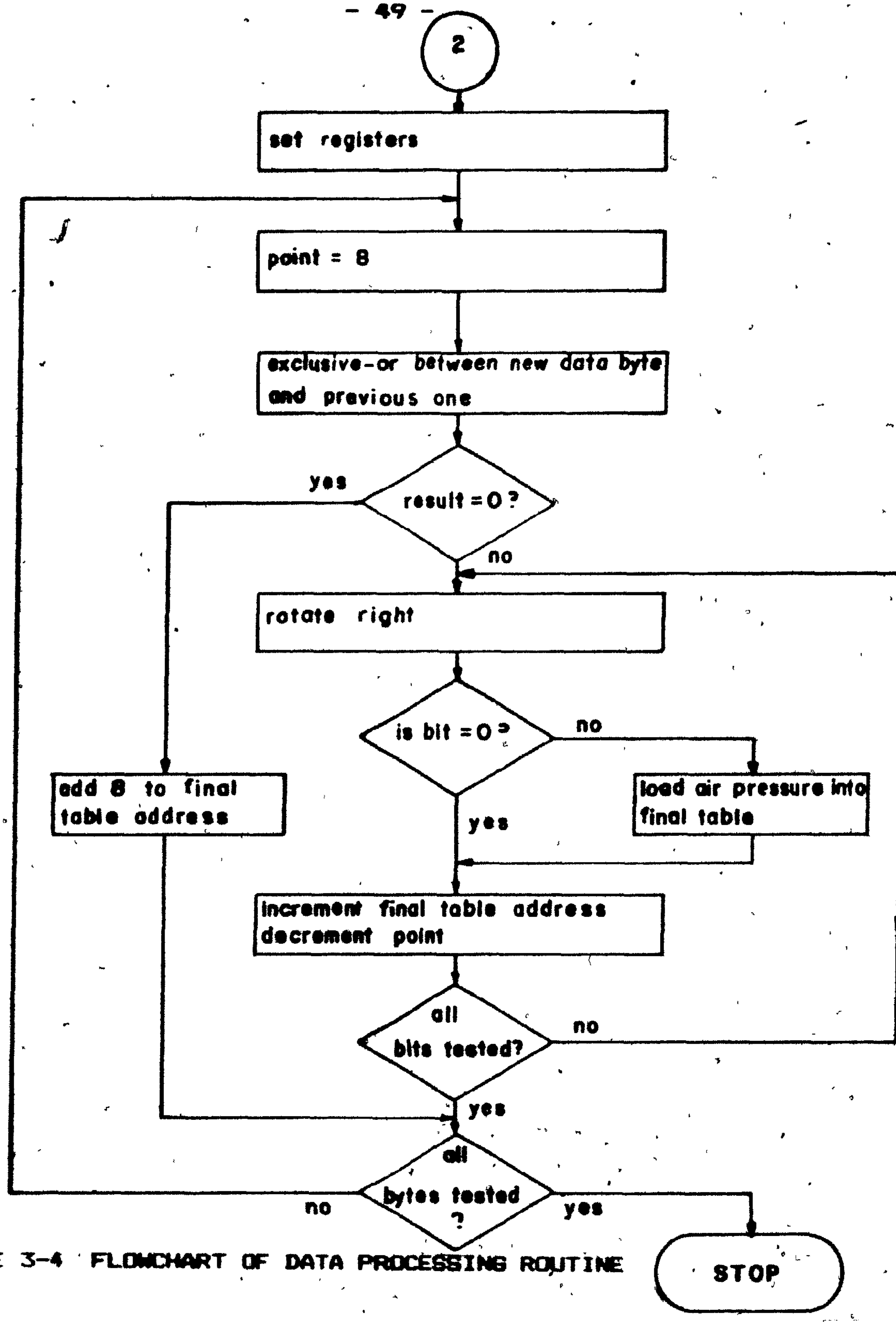

FIGURE 3-4 FLOWCHART OF DATA PROCESSING ROUTINE

times, i.e., 400 complete scans. At this point, the program has sequentially stored 400 x 32 = 12800 bytes of cell data and 400 bytes of air pressure data. Appendix B contains a listing of program DATACQ.

Not all instructions are executed in the same number of clock cycles. Typically, a Motorola 6809 has a clock cycle of 1 us. Instructions take between 2 and 20 cycles to be executed. About 1 ms is required to complete an entire scan. The time required to carry out the data reading process corresponding to the determination of pressure distribution among 256 cells, is approximately 400 ms.

C.-Data Processing Routine. After the data reading process is finished, the program executes the data processing routine, which detects particular cell transitions and associates them with the appropriate pressure. The principle of this routine is the comparison of a given cell data byte with its next value, obtained during the following scan. If values differ, it means that one or more of the eight cells corresponding to that byte have been operated. The comparison is made through an EXCLUSIVE-OR function. Therefore, this operation produces a byte in which the ONE bits correspond to those cells that have undergone transition during the interval between the acquisition of the sequential byte pair that was Exclusive-Ored.

Figure 3-4 shows a flowchart of the data processing routine. The first step is to load the pointer, POINT, with

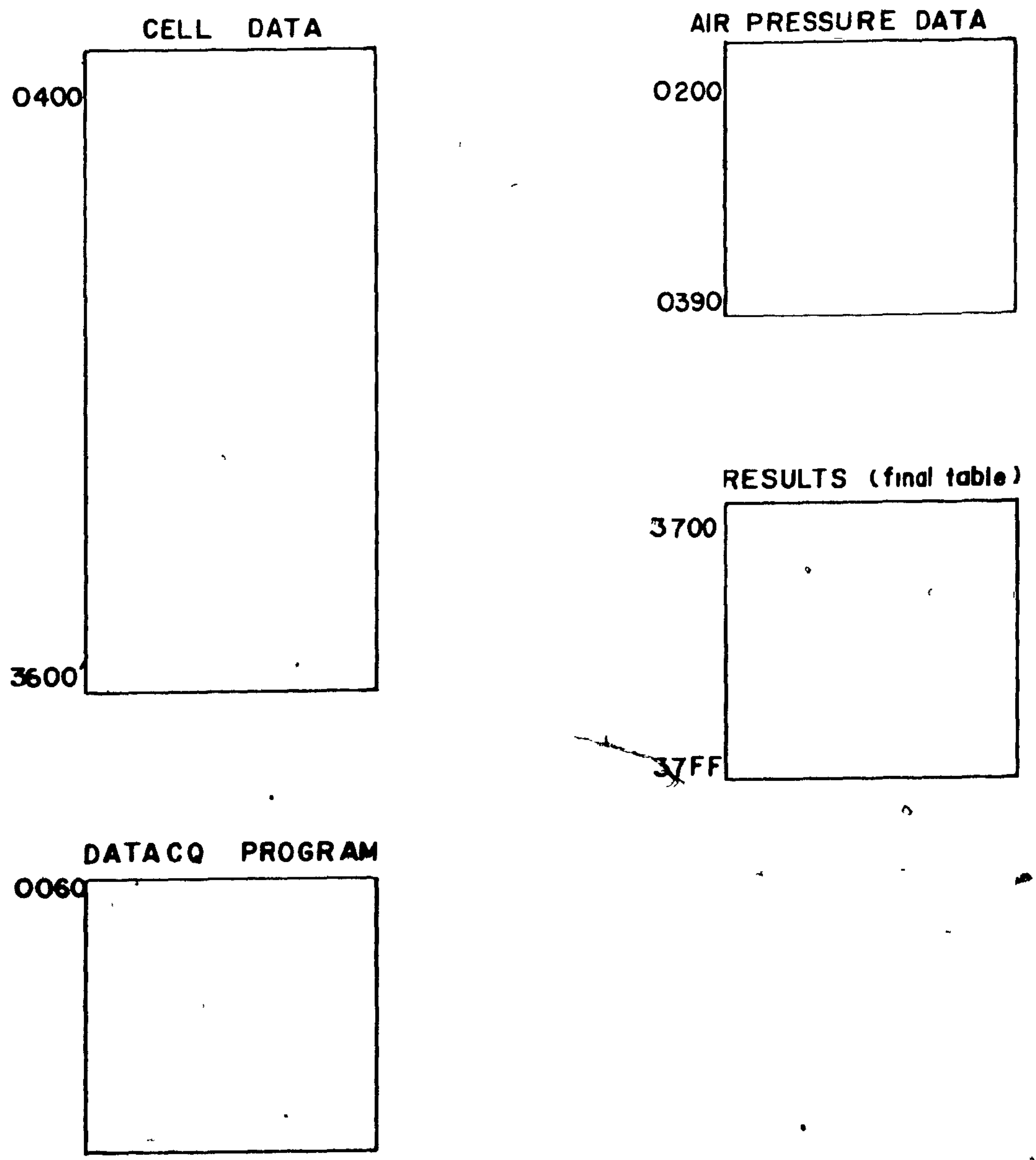

FIGURE 3-5 MEMORY MAP

the value eight, the number of bits to be shifted.

Then, the data byte and its next sequential value are Exclusive-Ored. If the resulting byte is ZERO, it means that there was no change in the status of the cells during the interval implied by the sequential byte pairs and the pointers which address bytes in the two switch image maps being compared, are incremented. If the resulting byte is not ZERO, it means that one or more cells have changed state. To identify these cells, the resulting byte is rotated one bit to the right into the carry bit. If this bit is ONE, then the air pressure corresponding to that scan is loaded in the position corresponding to that cell in the final data table. If the shifted bit is ZERO, then the pointers are updated. If all the bits have not been tested, the next bit is shifted. If all eight bits of the resulting byte have been tested, the process is repeated as the program goes on to detect changes in the next pair of bytes. When all data bytes have been checked, a final table with the pressure corresponding to each cell is obtained. Figure 3-5 shows the memory map assigned to this program.

3.3 The Data Displaying Process.

The final result of program DATACQ is a 256 byte table of pressure values exerted on each of the 256 sensor cells. This test record can be stored in a file on diskette. Subprogram SAVE creates a file with these values. A program listing of this subprogram is included in Appendix C.

Program DISPLY displays and lists the results of a test filed by SAVE. The program consists of the following stages:

- Recover the data from the file created by subprogram SAVE and store it in an array.
- Display the 8-bit integer pressure values.
- Print the cell number and its pressure, converted to engineering units.

Figure 3-6 shows a flowchart of program DISPLY and Figure 3-7 shows a very simple view of the pressure distribution. The program listing is included in Appendix D. More suitable graphic display procedures are available, e.g., the routine developed by Louis C. Vroomen(11) that plots a tridimensional view of the pressure distribution exerted on the sensor plate. This routine uses an isometric coordinate representation to indicate the position of the cells and their pressure value.

This program incorporates a hidden line removal feature when depicting a "net" of the pressure distribution in the pictorial plane. Coordinates X and Y identify the position of a cell on the sensor plate and coordinate Z identifies the pressure exerted on it. The program starts with the lowest value of Y, plotting the pressure corresponding to all the values of X.

Then, the Z coordinates for the next higher Y level are calculated and compared on a point to point basis with those

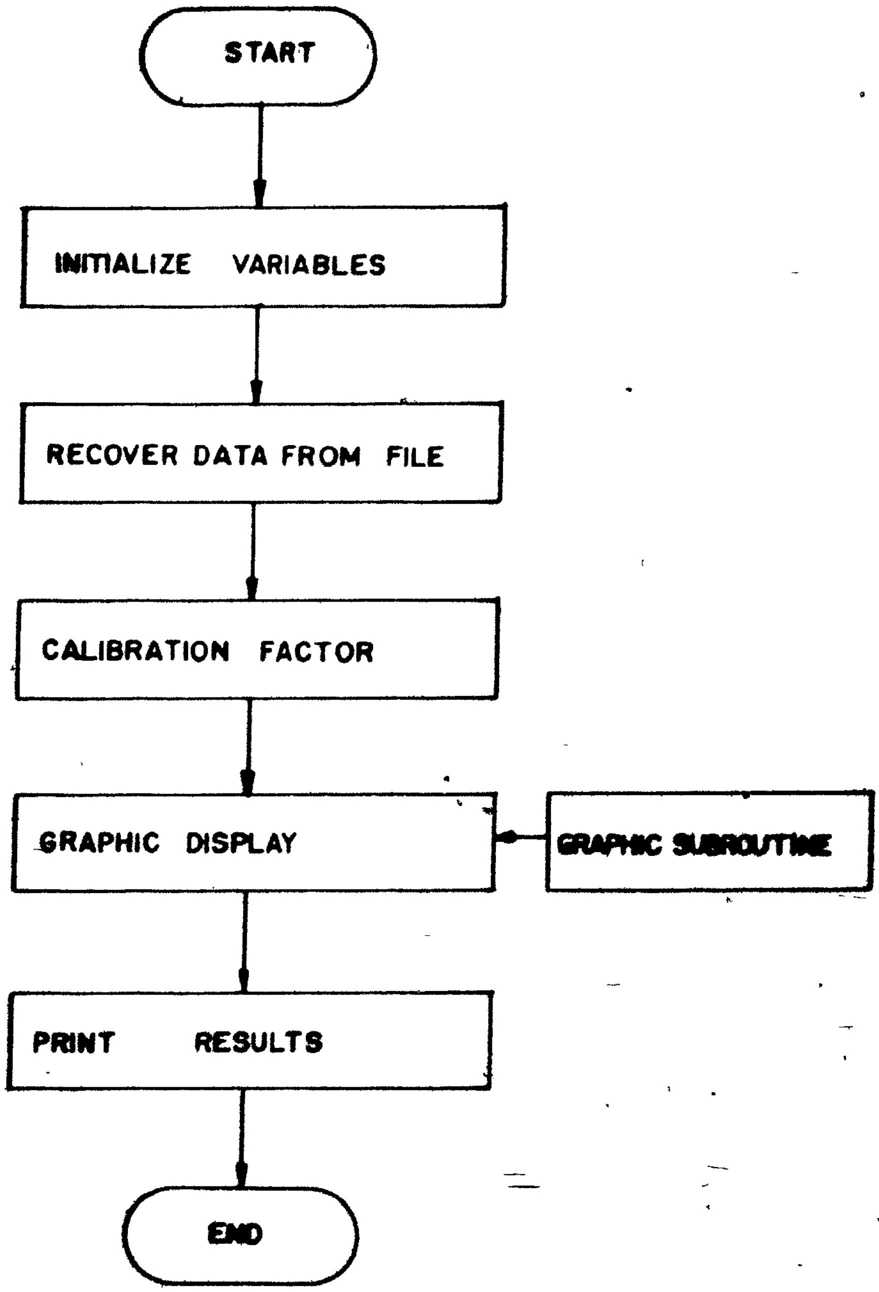

FIGURE 3-6 FLOWCHART OF PROGRAM DISPLY

of the previous curve. If the point is visible i.e., current Z larger than previous Z, then the X-Z coordinate is plotted. If the point is invisible, i.e., current Z lower than previous Z, it is ignored. As each subsequent slice is plotted "displaced" according to the selected skew and elevation angle, with respect to the previous one, the Z datapoint comparison takes place using two linear sliding arrays. The displacement index is a function of these angles, the number of cuts in the Y direction and the length of the Y axis.

Figure 3-8 shows graphics obtained with this routine using arbitrary pressure data. A program listing is included in Appendix E.

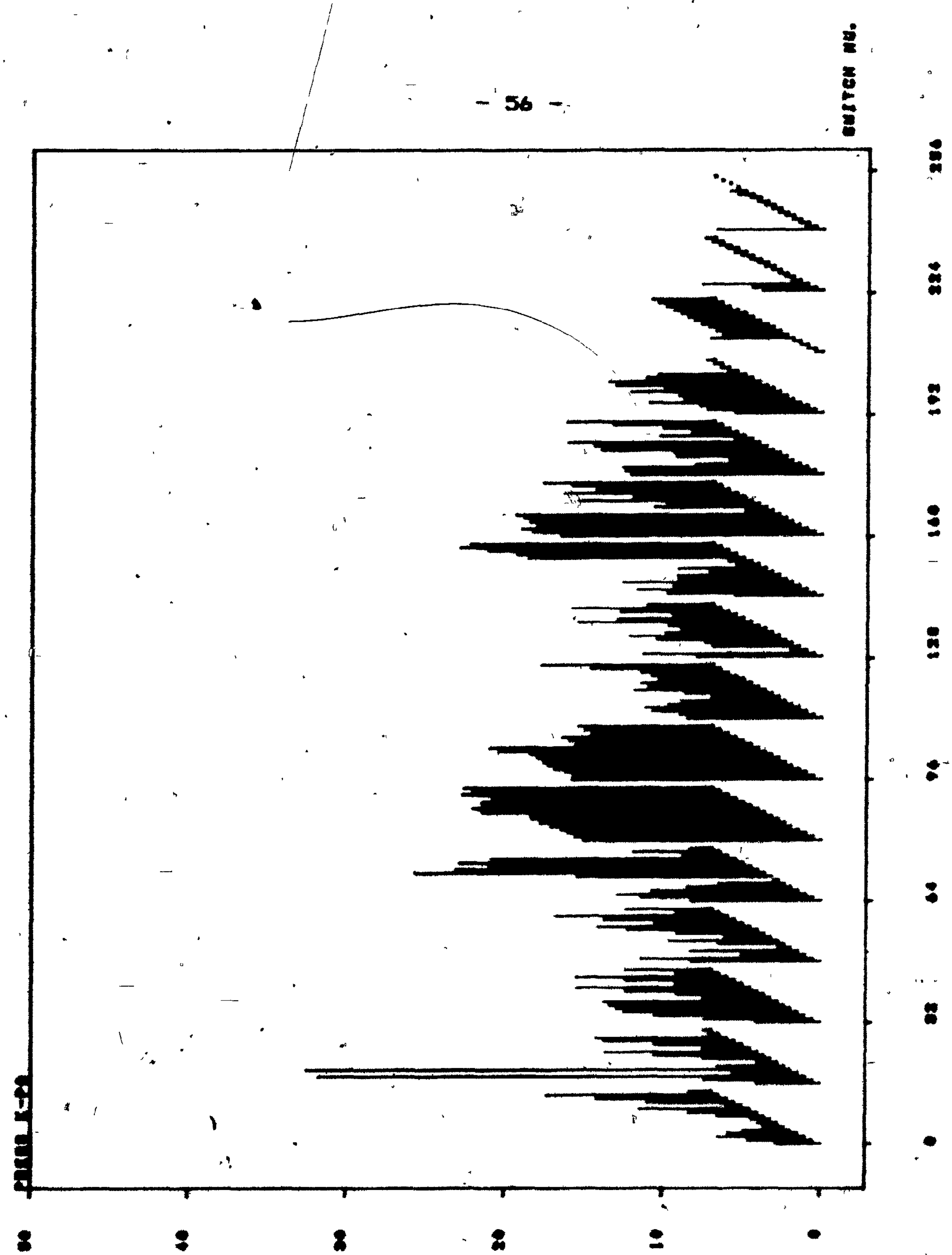

FIGURE 3-7 SIMPLE REPRESENTATION OF PRESSURE DISTRIBUTION

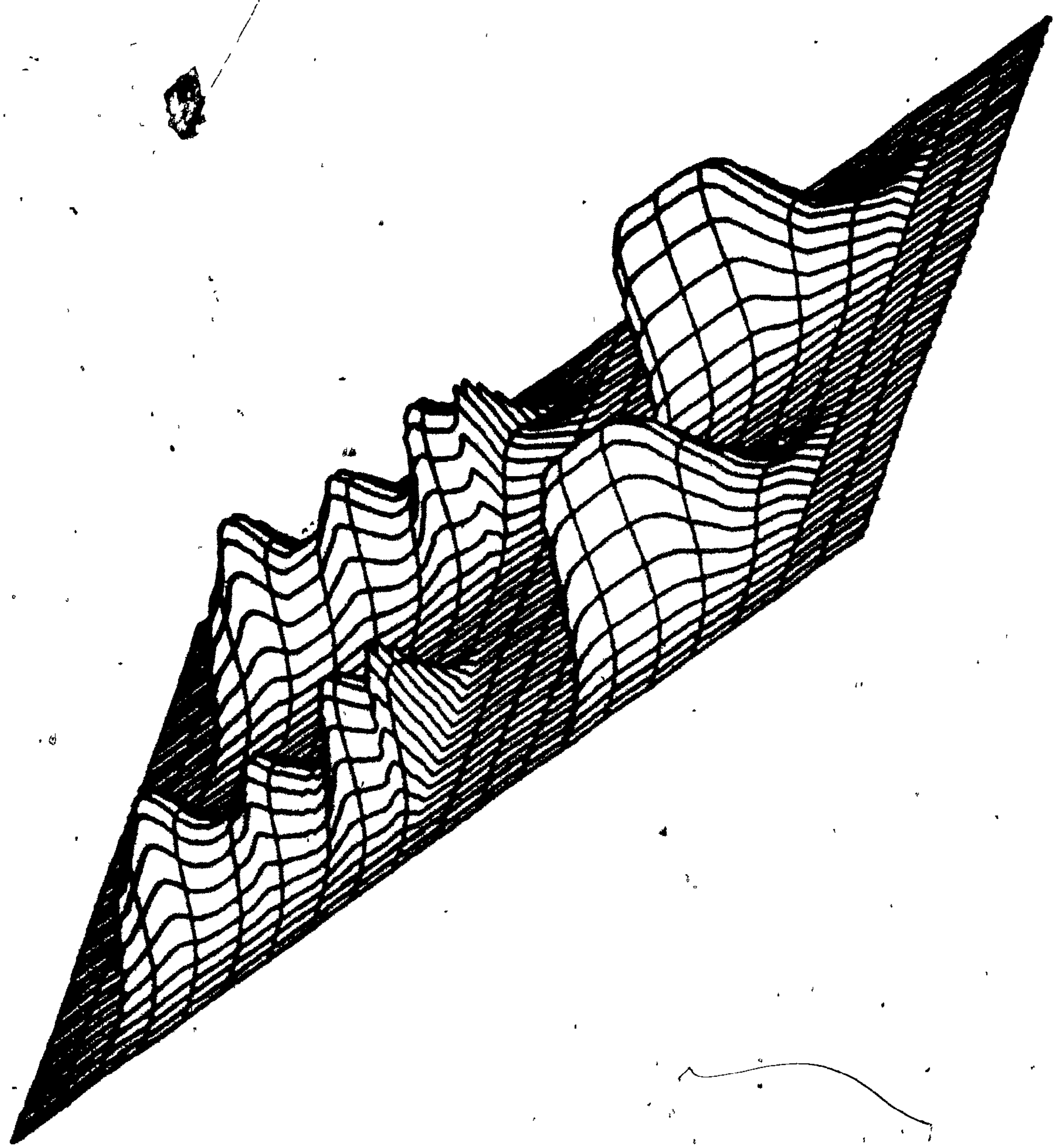

FIGURE 3-8 FIGURE 3-D REPRESENTATION OF PRESSURE DISTRIBUTION

CHAPTER 4 EXPERIMENTAL MODEL

4.1 Introduction.

In order to analyze the performance of the proposed system, a small prototype consisting of 16 cells was built. A Motorola 6809 microcomputer with CRT display, keyboard, diskette unit, printer and plotter was used. Interface circuits for the pressure sensor and the movement detector, as well as the manifold required to control plenum pressurization, were also built. Figure 4-1 shows a schematic diagram of the system. Air is supplied through an air regulator. Two solenoid valves control the air flow input and output. The output valve exhausts air to the atmosphere. A needle valve provides an adjustable restriction to vary the time constant of the pressure system. A pressure transducer measures the internal pressure of the sensor plenum. A pressure gauge is used as a reference instrument. The output signal of the pressure transducer and the output signal of the movement detectors are supplied via the interface circuit to the computer which provides the solenoid valve control signals. Figure 4-2 is a photograph of the prototype system.

4.2 Model Fabrication.

A machined aluminum prototype, Figure 4-3, measuring

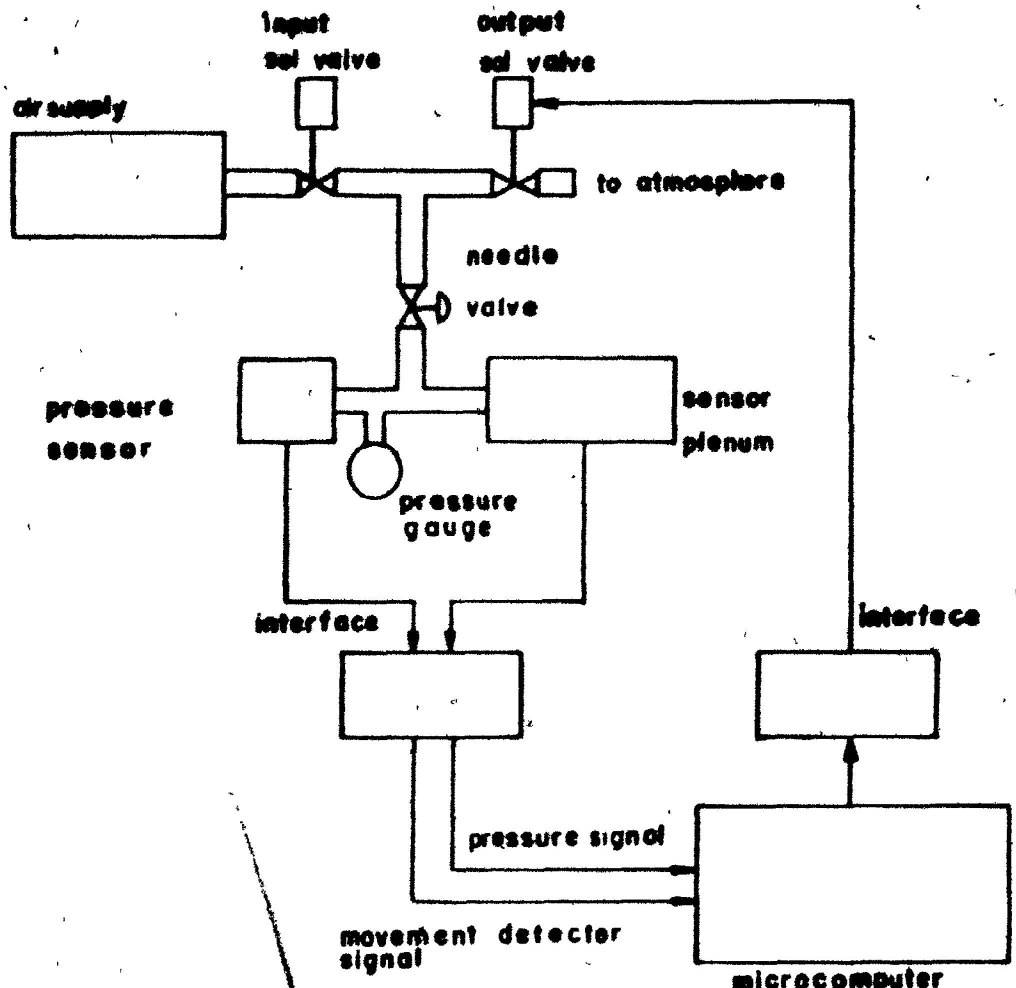

FIGURE 4-1 SCHEMATIC DIAGRAM OF THE SYSTEM

FIGURE 4-2 OVERVIEW OF PROTOTYPE SYSTEM

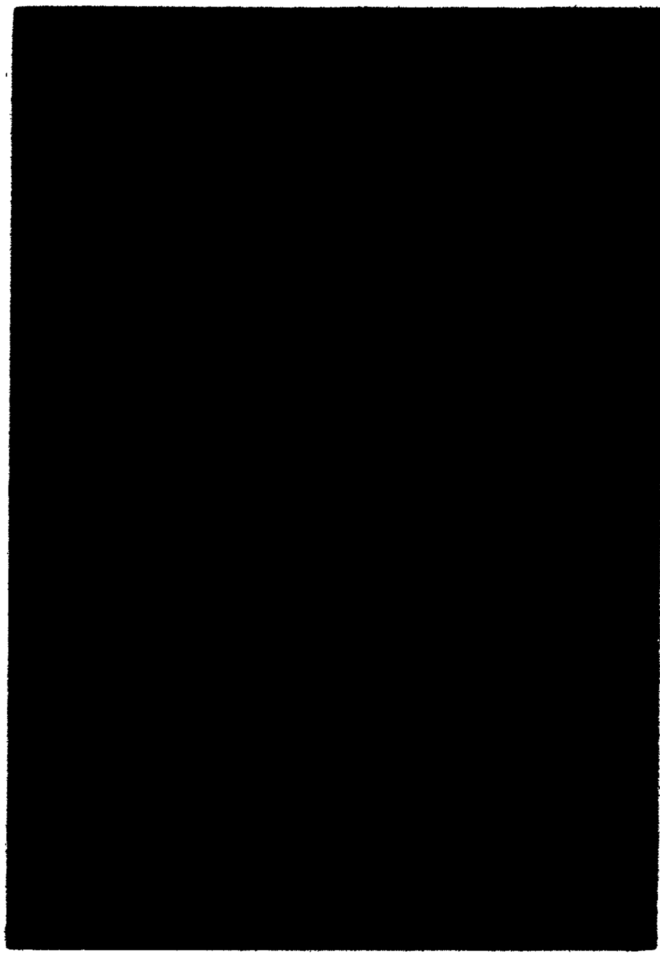

FIGURE 4-3 VIEW OF THE PROTOTYPE

100x100x50 mm, was fabricated. The weight of each sensor piston is 11.2 g and the whole sensor and plenum assembly weighs 800 g. The 12.5 mm diameter cell sensing faces lie on a 25 mm square pitch. A small rod, to activate the movement detector, protrudes 12.5 mm from the base of each sensor.

Two 0.2 mm thick polythylene films, with sealing compound applied to both surfaces, were used as gaskets to control air leakage from the cells. The sensor assembly is a three-layer sandwich, with adjacent layers separated by a gasket. Figure 4-4 is a section of this assembly while Figure 4-5 shows the corresponding three piston components. A loop of plastic film was maintained between the cells to allow free axial movement of the pistons.

4.3 Movement Detector.

Originally, microswitches were installed because these were considered to be a simple and obvious way to sense air pressure/contact force equilibrium. However, they required a minimum piston displacement of about 0.4 mm and, more seriously, an actuation force of 0.69 N.

Very sensitive microswitches are expensive. Optoelectronic sensors cost no more than inexpensive microswitches; about US $3 in 1983. As shown in Figure 4-6 a metal shim was attached to the protruding axial rod on each piston to occult and expose the optoelectronic sensor.

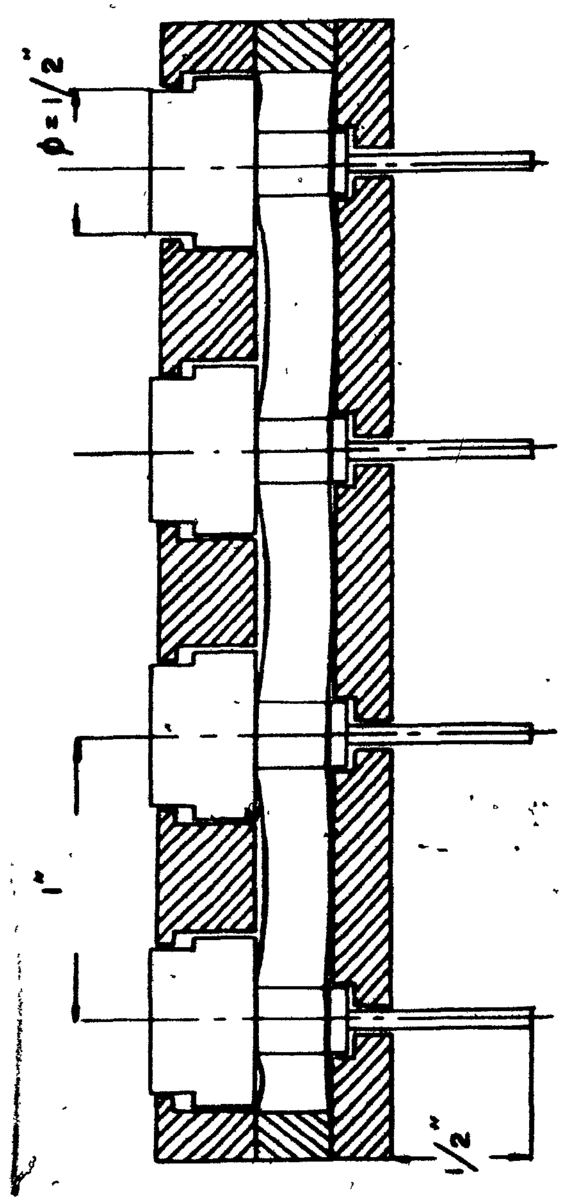

FIGURE 4-4 PROTOTYPE DIAGRAM SHOWING THE PLASTIC FILM POSITION

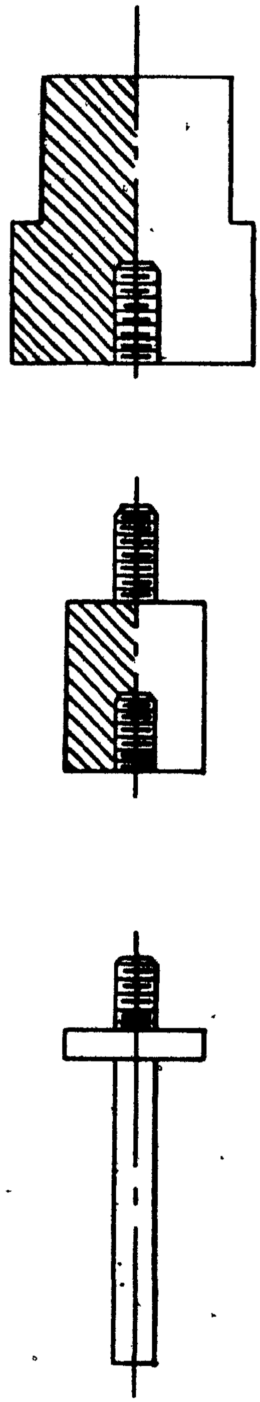

FIGURE 4-5 PARTITION OF A CELL TO ALLOW THE INSTALLATION OF THE SEALING FILM

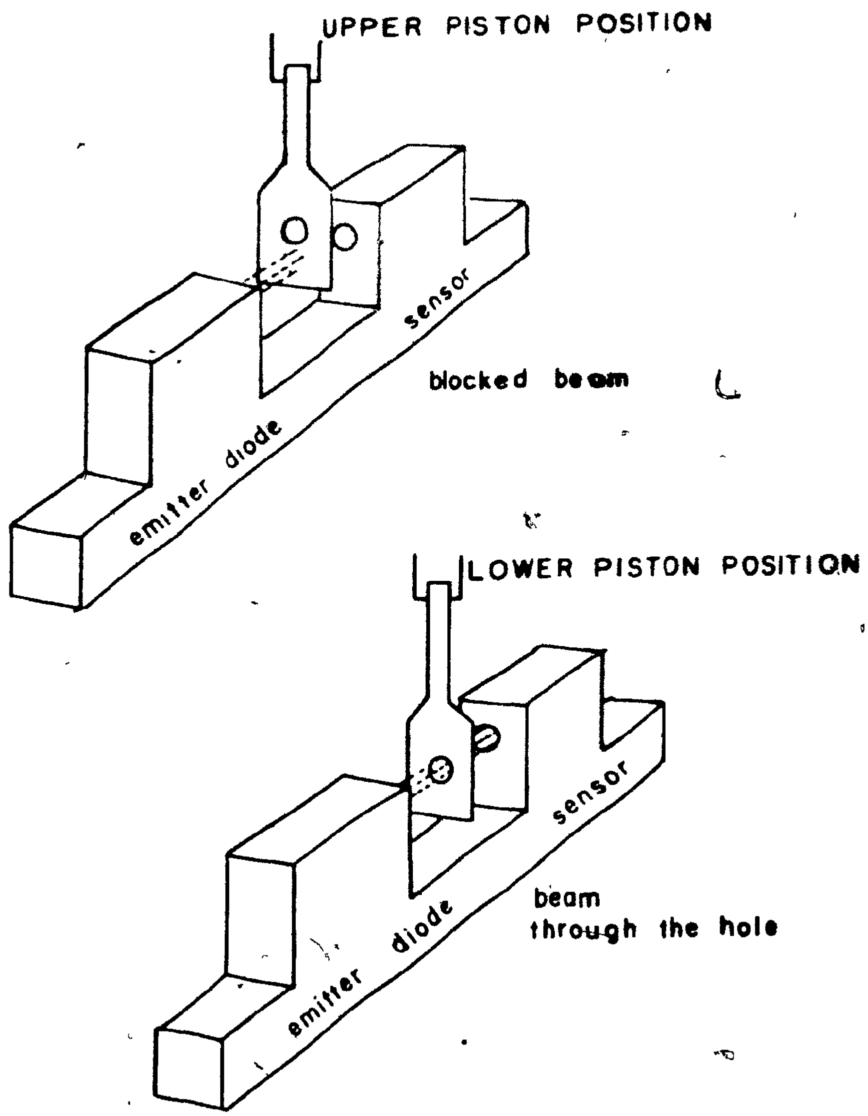

FIGURE 4-6 SCHEMATIC DIAGRAM OF THE MOTION DETECTOR OPERATION

When the cell is raised by air pressure, the thin plate blocks the beam from the light emitting diode. When the cell moves down, the hole in the plate allows the light beam to turn on the phototransistor.

The displacement required to produce maximum output voltage is similar to that required to activate the microswitches. However, the optoelectronic sensor introduces no resistance to motion and is effective over a smaller displacement.

In general, logic level ONE corresponds to a nominal voltage of 5 V and logic level ZERO corresponds to a nominal voltage of 0 V. However, the digital interface used to read the output of the movement detector, a PIA with input buffers, reproducibly distinguishes between ONES and ZEROS at a threshold of about 1.1 V as shown in Figure 4-7.

On the other hand, the output signal of the opto-electronic sensor is 10 mV at the upper position of the cell, level ZERO, and 180 mV at the lower position, level ONE.

Therefore, an amplifier with a gain of 100 was inserted between the sensor and the digital interface so that, in the upper position, the output voltage was 1.0 V. With this gain, the amplifier output voltage, for a photodetector output of 100 mV, saturates to the power supply voltage of 12 V. A zener diode was added to limit this output to 5 V. In this way, the displacement required to detect the movement of a cell is reduced by a factor of about four.

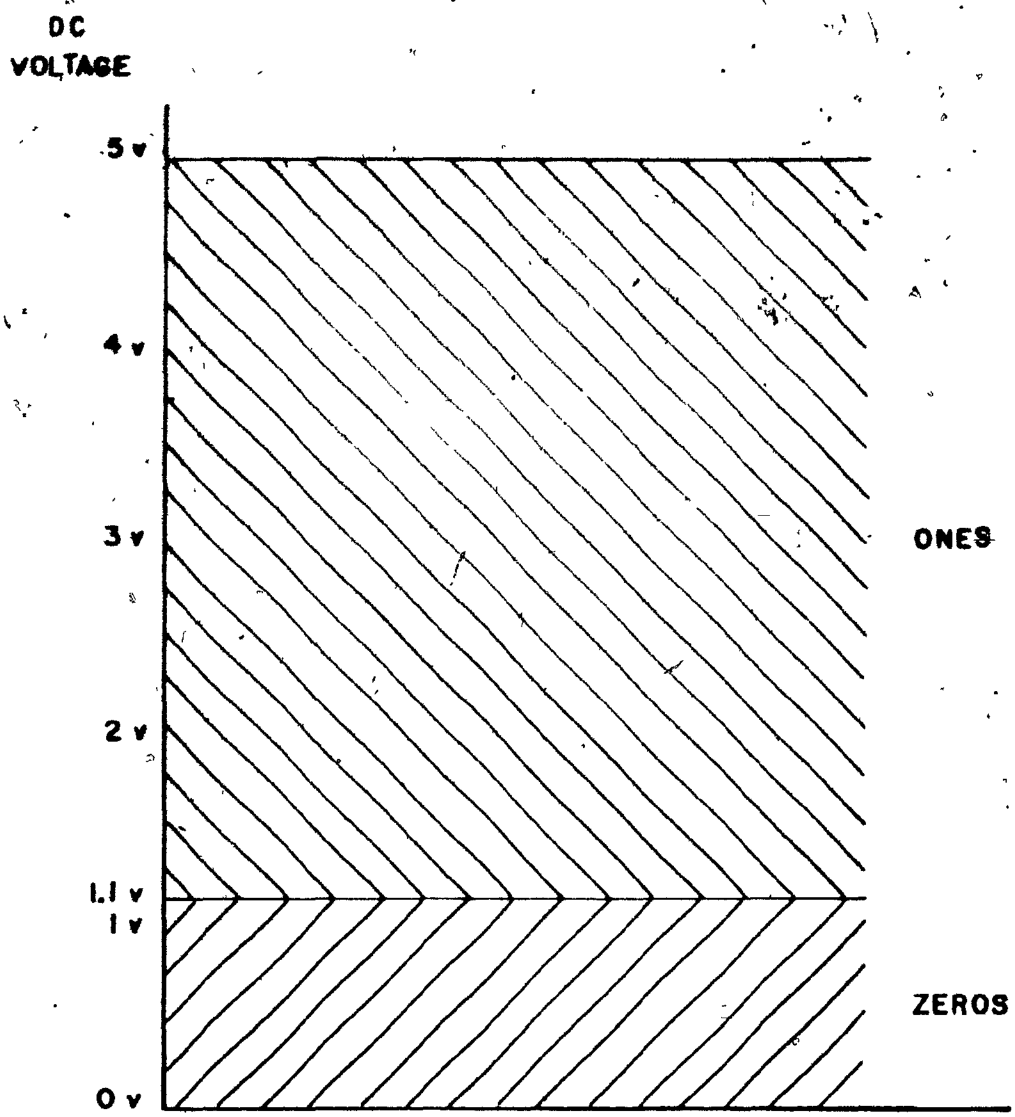

FIGURE 4-7 VOLTAGE RANGE FOR LOGIC ONES AND ZEROS

The displacement required to detect the movement of a cell may be adjusted, in order to strike the appropriate compromise between sensitivity and noise immunity, by changing the gain of the amplifier. Figure 4-8 is a diagram of the movement detector circuit.

4.4 Pressure Transducer.

A 20 PSI (138 kPa) full scale Statham, unbonded strain gage pressure transducer, was used to measure the air pressure applied to the sensor cells. The transducer sensitivity is 20 uV/kPa/V. An amplifier with a gain of 1080 was added to obtain the maximum voltage of 5.4 V at a cell pressure of 6 PSI (41 kPa). This was the maximum required to exploit the full scale range of the ADC. 5.4 V corresponds to a full scale 8-bit binary output of 255.

Figure 4-9 shows the pressure transducer circuit and the amplifier stage. Potentiometer #1 provides zero adjustment and potentiometer #2 provides offset adjustment.

4.5 Solenoid Valve Circuit.

A 110 V Skinner solenoid valve, controls air escape from the cell plenum. A Hamlin HE 721 co5-10 relay provides solenoid current to the valves. The relay coil is supplied through an open collector inverter gate, connected to a digital output line of the interface circuit. A

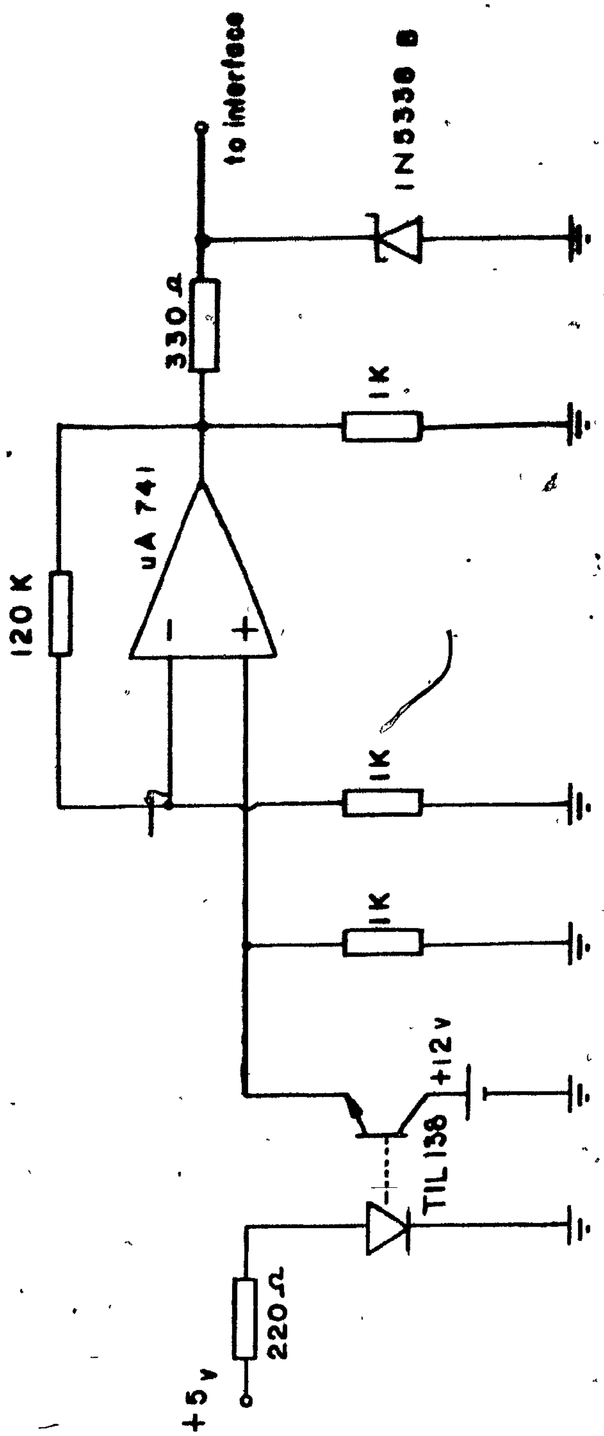

FIGURE 4-8 MOTION DETECTOR CIRCUIT

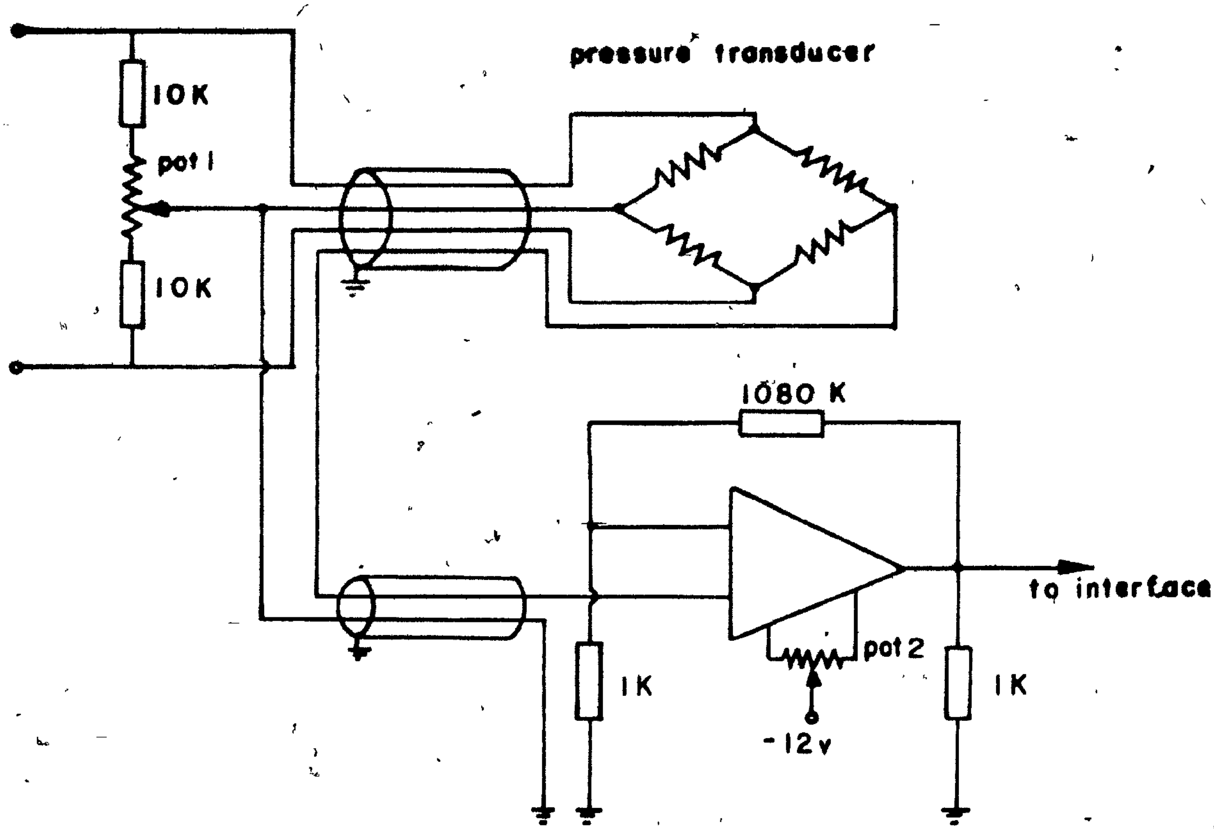

FIGURE 4-9 PRESSURE TRANSDUCER CIRCUIT

second, air input, solenoid valve is controlled manually. I.e., during experiments with the sensor assembly the plenum was charged manually and transition measurements were done while pressure was decreasing. The solenoid valve circuit is shown in Figure 4-10.

4.6 Interface Circuits.

The interface circuits connect the transducer, the movement detector and the solenoid valve circuit, to the microcomputer.

4.6.1 Pressure Transducer Interface.

The amplified pressure transducer output signal is an analog voltage of 0 - 5.4 V. A commercial 8-bit ADC system, JPC AD-16, converts this voltage into a binary number. The microcomputer takes ADC readings via a PIA. This small data acquisition system has a 16 channel multiplexer and programable gain selection. Figure 4-11 shows the interface schematic.

4.6.2 Movement Detector Interface.

The output of each movement detector is connected through a 256:16 multiplexer to the 16 inputs of a second PIA as shown in Figure 4-11. To scan the status of the movement detectors, a 4-bit counter selects multiplexer channels in blocks of 16. The counter is incremented by a

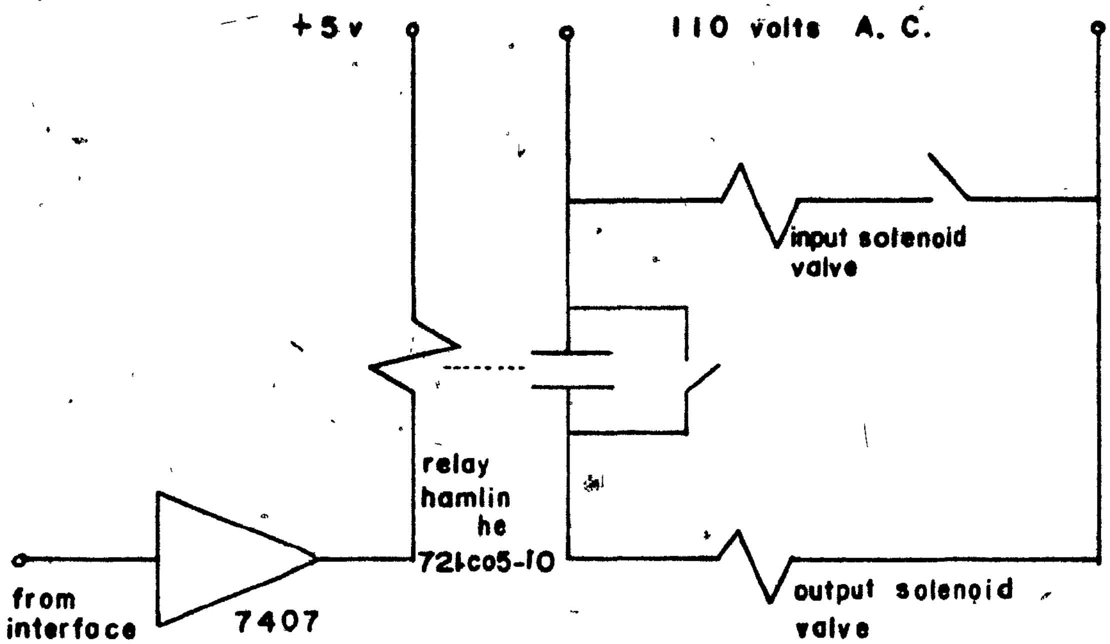

FIGURE 4-10 SOLENOID VALVE CIRCUIT

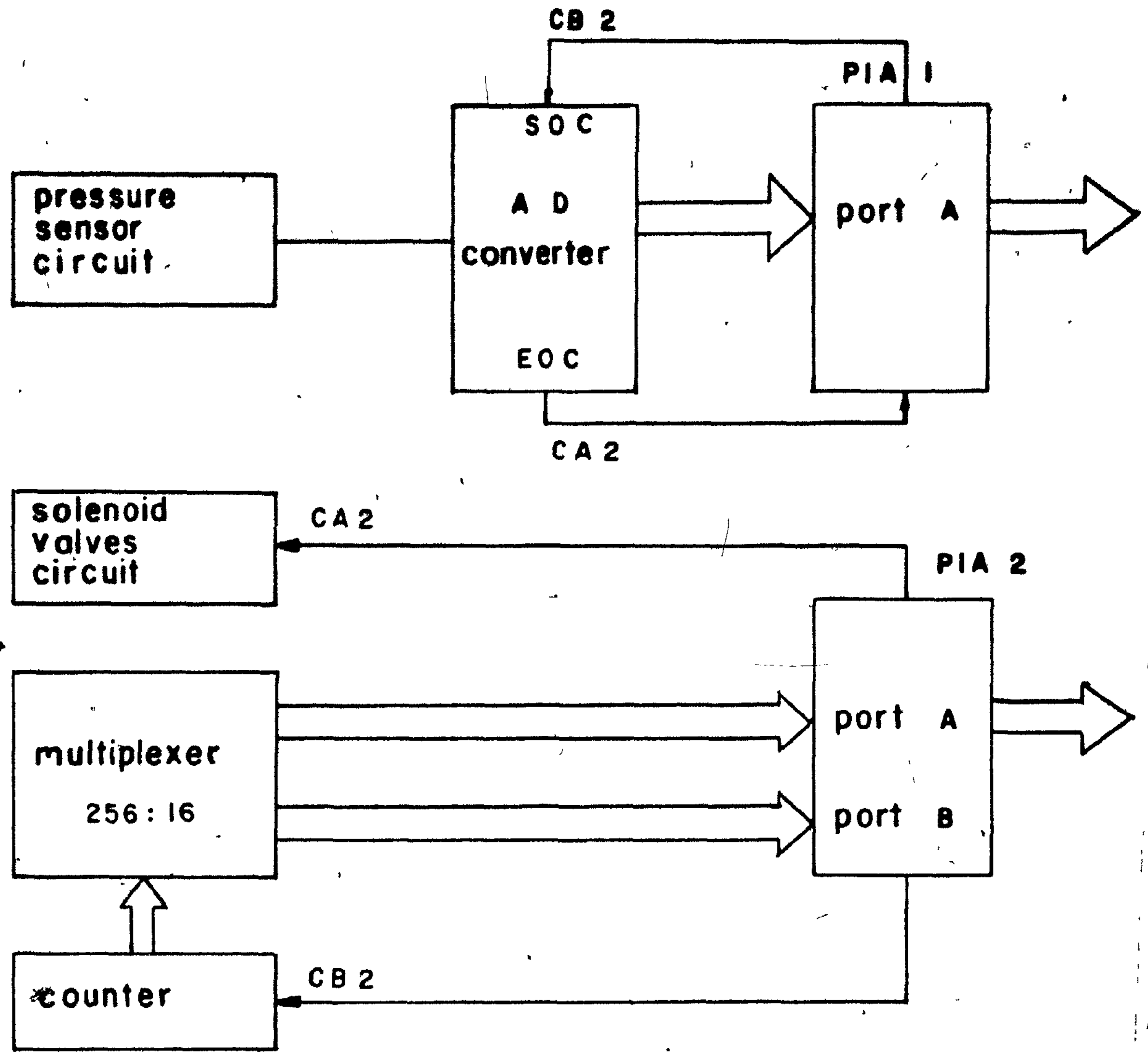

FIGURE 4-11 BLOCK DIAGRAM OF THE INTERFACE

PIA output control line which pulses whenever the PIA inputs are read by the microcomputer. At the beginning of a measurement cycle, the counter is cleared. To operate the 16 cell prototype, it was not necessary to use the multiplexer because the output lines of the 16 movement detectors were connected directly to the 16 parallel inputs of the second PIA.

4.6.3 Solenoid Interface.

Solenoid valve operation is controlled through another control line, CA2, of PIA 2. This is also shown in Figure 4-11.

4.7 Calibration.

The system was calibrated in stages. The first stage consisted of the calibration of the pressure transducer and its amplifier unit. The ADC was calibrated next. Afterwards, the movement detectors were adjusted so as to produce an output signal of 1 V when the cells are raised and a maximum voltage of 5 V when they are in the lower position. Finally, an overall system calibration was performed.

4.7.1 Pressure Transducer Calibration.

Air pressure was applied through a regulator in increments of 0.5 PSI (3.45 kPa) up to 6 PSI (41 kPa). The average of three voltage readings was plotted against the

AIR PRESSURE (PSI)	(kPa)	OUTPUT VOLTAGE (V)
0.0		0.0
0.5	3.45	0.45
1.0	6.90	0.95
1.5	10.34	1.40
2.0	13.79	1.80
2.5	17.24	2.25
3.0	20.68	2.65
3.5	24.13	3.10
4.0	27.58	3.60
4.5	31.03	4.05
5.0	34.47	4.55
5.5	37.92	4.95
6.0	41.37	5.40

$$\text{LINEARITY} = \frac{\text{MAX. DEVIATION}}{\text{MAX. VALUE}} \times 100 = \frac{0.05}{5.4} \times 100 = 0.93\%$$

TABLE 4.1 PRESSURE SENSOR CALIBRATION DATA.

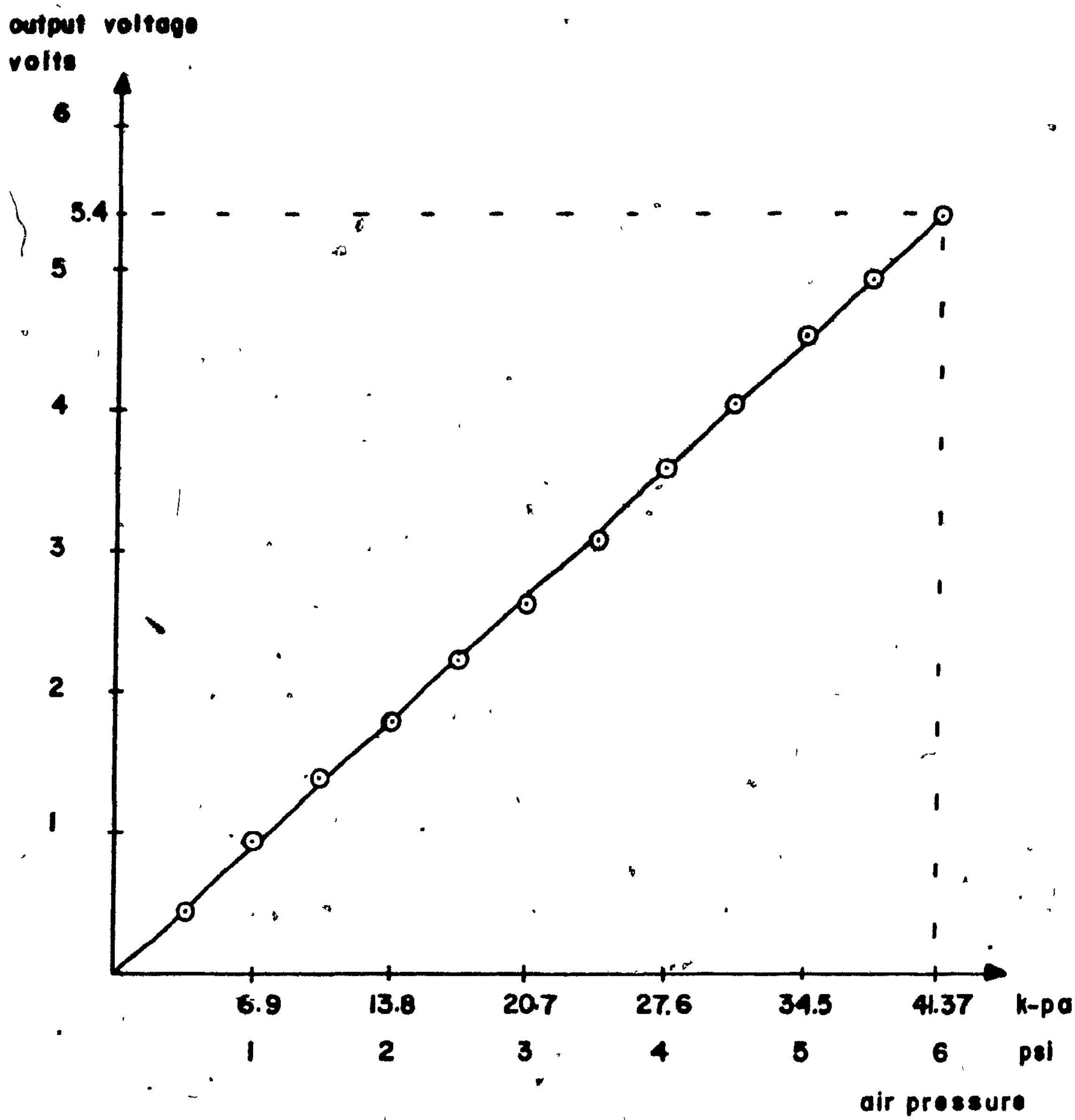

FIGURE 4-12 PRESSURE SENSOR CALIBRATION CURVE

applied pressure. Table 4.1 shows the calibration data and Figure 4-12 shows the calibration curve of the pressure sensor and amplifier. Its linearity deviation is 0.93 %.

4.7.2 ADC Calibration.

The ADC was calibrated with a variable DC power supply. The first step was to find the analog input voltage that produces the maximum binary output, i.e., 255. This value is 5.4 V. Then, increments of voltage of 1/16 of this value, starting from zero, were applied. Table 4.2 shows the calibration data and Figure 4-13 shows the calibration curve of the ADC. Linearity deviation is 0.78 %. Then, the pressure transducer and amplifier and the ADC were calibrated as a unit. Table 4.3 shows this calibration data which is plotted in Figure 4-14. In this case, linearity deviation is 1.18 %.

4.7.3 Movement Detector Calibration.

This calibration consisted of two stages. First, the position of the plates that block the light beam of the optoelectronic sensor was mechanically adjusted. Then, the small differences which still remained were corrected by adjusting the gain of the corresponding amplifier.

4.7.4 Overall Calibration.

A set of standard weights, from 50 g to 500 g, were used to calibrate the system. A problem arose during this

D.C. VOLTAGE (V)	A/D CONVERTER OUTPUT (Hexadecimal)
0.34	10
0.68	20
1.01	30
1.35	3F
1.69	4F
2.03	60
2.36	70
2.70	80
3.04	8F
3.38	A0
3.71	AF
4.05	BE
4.39	CF
4.73	DF
5.06	EF
5.40	FF

$$\text{LINEARITY} = \frac{2}{255} \times 100 = 0.78\ \%$$

TABLE 4.2 ANALOG TO DIGITAL CONVERTER CALIBRATION DATA.

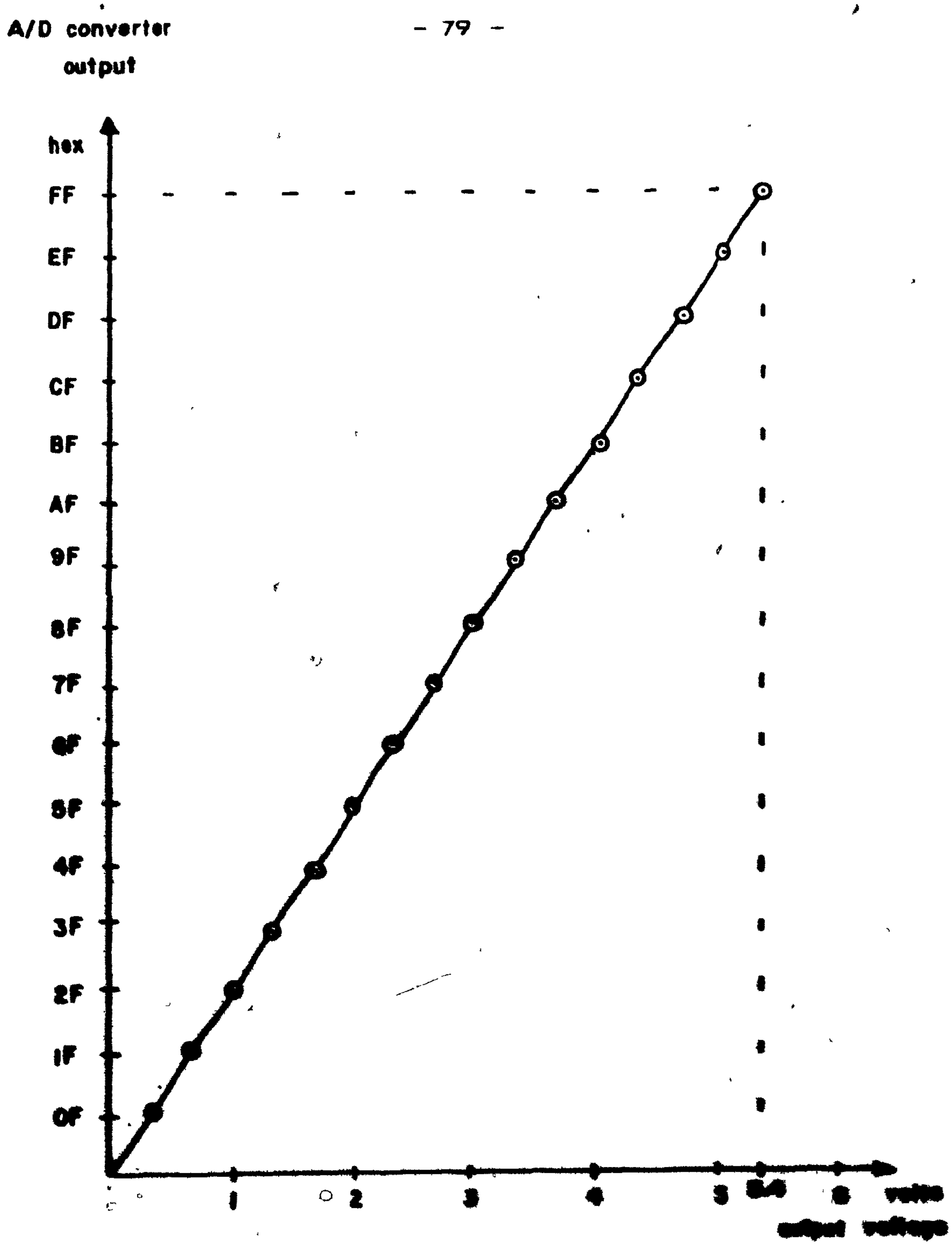

FIGURE 4-13 ANALOG TO DIGITAL CONVERTER CALIBRATION CURVE

AIR PRESSURE		ADC OUTPUT	
(PSI)	(kPa)	Hex	Decimal
0.5	3.45	16	21
1.0	6.90	2A	42
1.5	10.34	41	65
2.0	13.79	56	86
2.5	17.24	6A	106
3.0	20.68	80	128
3.5	24.13	94	148
4.0	27.58	A7	167
4.5	31.03	BF	191
5.0	34.47	D4	212
5.5	37.92	EA	234
6.0	41.37	FF	255

$$\text{LINEARITY} = \frac{3}{255} \times 100 = 1.18\ \%$$

TABLE 4.3 PRESSURE SENSOR - ADC CALIBRATION DATA.

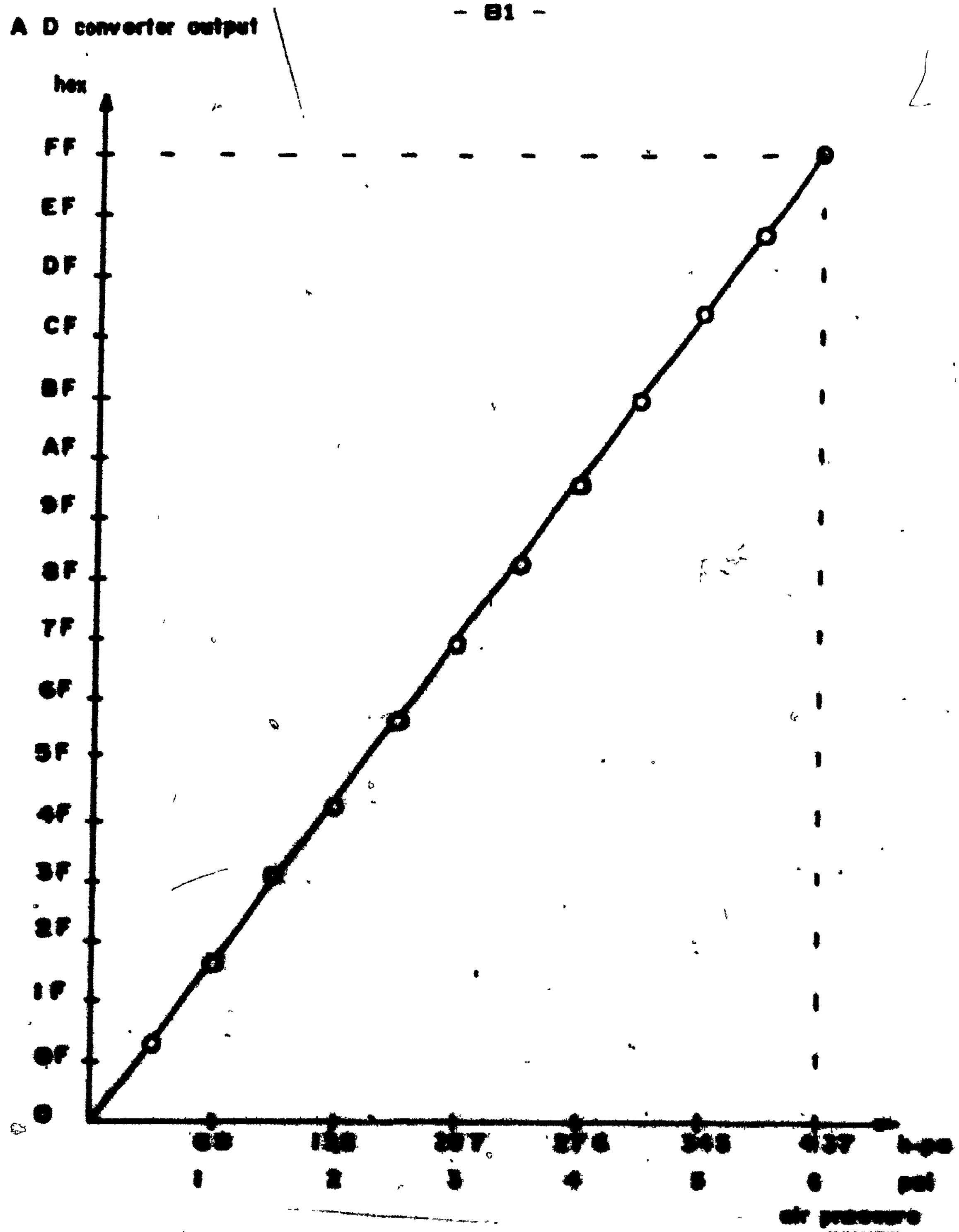

FIGURE 6-10 PRESSURE SENSOR-ADC CALIBRATION CURVE

calibration. The pressure transducer signal exibited excessive noise levels.

To solve this problem it was necessary to minimize the length of all analog signal conductors and to use shielded wire. A low noise power supply was required and a small capacitor was connected across the pressure transducer output. This filter capacitor reduces noise but it also reduces frequency response. Figure 4-15 shows the effect of a 220 uF filter capacitor on the output signal. A 47 uf capacitance provides an acceptable compromise. Figure 4-16 shows the effect of a 47 uF filter capacitor and the response of the movement detector to four different cell calibrating loads.

Table 4.4 shows the values of forces and pressures corresponding to the different standard loads. Table 4.5 shows results, the average of three hexadecimal readings, obtained when the needle valve exhaust restriction is adjusted to obtain a time constant R_1C_1=50 ms. Measured pressure versus applied pressure is shown in Figure 4-17.

To obtain the actual pressure value, it is necessary to apply a calibration factor. Figure 4-17 shows the curves for the ideal and real output pressure. The equation for ideal output pressure is :

$$y_1 = K_1 x \qquad (49)$$

where: x = applied pressure

FIGURE 4-15 EFFECT OF A FILTER CAPACITOR (220 uF) ON THE OUTPUT SIGNAL

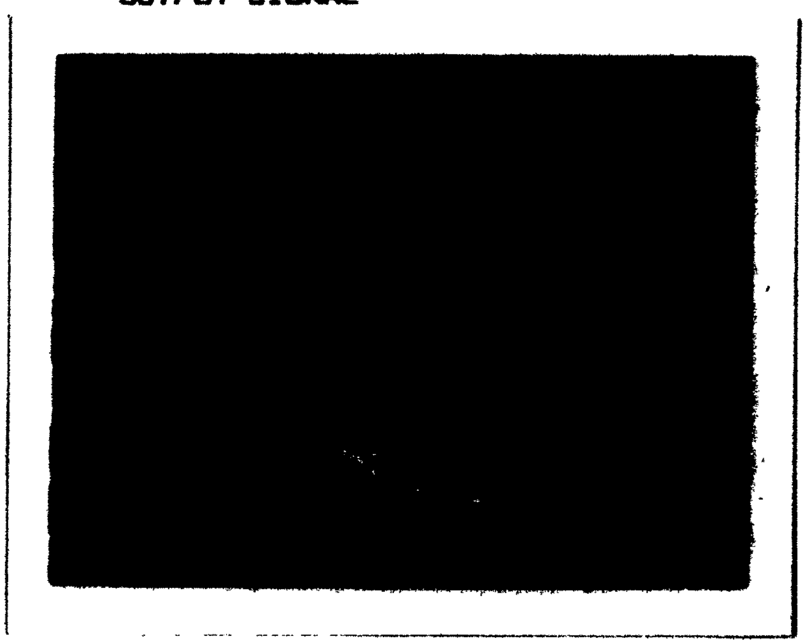

FIGURE 4-16 EFFECT OF A 47 uF CAPACITOR AND RESPONSE OF THE MOTION DETECTOR TO FOUR DIFFERENT EXTERNAL PRESSURES

APPLIED LOAD (standard weights)	EXTERNAL APPLIED PRESSURE		
g	kPa	PSI	mm Hg
50	3.86	0.56	28.96
100	7.72	1.12	57.92
150	11.58	1.68	86.88
200	15.44	2.24	115.84
250	19.31	2.80	144.80
300	23.17	3.36	173.76
350	27.10	3.93	203.24
400	30.89	4.48	231.68
450	34.82	5.05	261.16
500	38.68	5.61	290.12

Cell Area where pressure is applied = 127 mm^2

TABLE 4.4 PRESSURES EXERTED ON A CELL BY STANDARD WEIGHTS.

STANDARD WEIGHT		MEASURED PRESSURE		
g	kPa	Hex	Dec	kPa
50	3.86	0B	11	1.78
100	7.72	25	37	6.00
150	11.58	3D	61	9.90
200	15.44	55	85	13.79
250	19.31	6C	108	17.52
300	23.17	86	134	21.74
350	27.10	9E	158	25.63
400	30.89	B6	182	29.53
450	34.82	CF	207	33.58
500	38.68	E8	232	37.64

$$kPa = \frac{Dec \times 41.37}{255}$$

$$\text{LINEARITY DEVIATION} = \frac{0.3}{37.64} \times 100 = 0.80\ \%$$

TABLE 4.5 GENERAL CALIBRATION DATA.

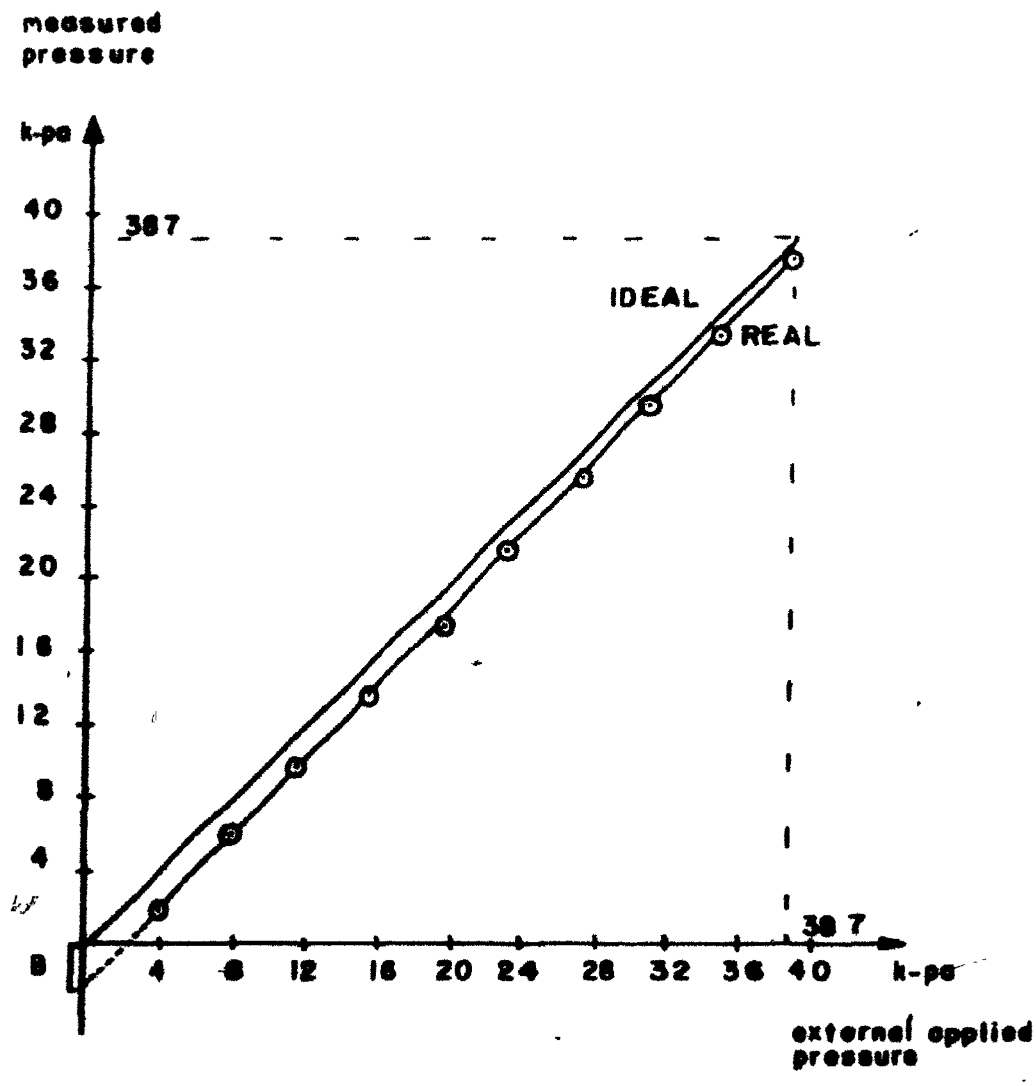

FIGURE 4-17 MEASURED PRESSURE VERSUS EXTERNAL APPLIED PRESSURE

y_1 = measured pressure

k_1 = ideal slope = 1

The equation for the actual output pressure is:

$$y_2 = -B+K_2x \tag{50}$$

where: K_2 = actual curve slope

B = difference between measured pressure corresponding to a minimum applied pressure and the theoretical value obtained from the actual curve when x = 0

Then,

$$y_1 = x$$

and

$$\frac{y_1}{y_2+B} = \frac{x}{K_2x} \tag{51}$$

Finally,

$$y_1 = \frac{1}{K_2}(y_2+B) \tag{52}$$

K_2 may be obtained with two extreme values:

$$K_m = \frac{P_{m2}-P_{m1}}{P_{ap2}-P_{ap1}} \qquad (53)$$

and B may be obtained as:

$$B = K_m\, P_{ap1} - P_{m1} \qquad (54)$$

where: P_{ap1} = minimum applied pressure

P_{ap2} = maximum applied pressure

P_{m1} = measured press. corresponding to P_{ap1}

P_{m2} = measured press. corresponding to P_{ap2}

Table 4.6 shows the pressure values corrected according to equation 52.

Accuracy is in the range ± 1%.

Resolution is $\frac{38.80}{255}$ = 0.152 kPa/bit

4.8 Tests.

The purpose of the following tests is to investigate the influence, on system response, of the time constant of the pressure system, the trigger distance required by the motion detector and the maximum air pressure. Equations 37 and 40 establish the mathematical relations between the measured pressure and these parameters. The friction coefficient which was assumed to be viscous for the purposes of analysis is not easy to modify and therefore no attempt was made to vary it.

APPLIED PRESSURE		MEASURED PRESSURE	CORRECTED PRESSURE	ERROR
g	kPa	kPa	kPa	%
50	3.86	1.78	3.86	0.0
100	7.72	6.00	7.95	0.59
150	11.58	9.90	11.73	0.39
200	15.44	13.79	15.49	0.13
250	19.31	17.88	19.12	-0.49
300	23.17	21.74	23.21	0.10
350	27.10	25.63	27.03	-0.18
400	30.89	29.53	30.75	-0.36
450	34.82	33.58	34.74	-0.21
500	38.60	37.64	38.60	0.0

$$\text{ERROR} = \frac{\text{CORRECTED PRESSURE} - \text{APPLIED PRESSURE}}{\text{MAXIMUM APPLIED PRESSURE}} \times 100$$

TABLE 4.4 CORRECTED PRESSURE VALUES.

4.8.1 Effect of the Variation of the Time Constant R_1C_1.

The time constant R_1C_1 may be easily modified by changing the resistance R_1. This resistance depends on the setting of the needle valve. Four different time constants were used and for each case, the set of standard loads from 50 g to 500 g was applied. Table 4.7 lists the results and Figure 4-18 shows the corresponding curves.

It was observed that measured pressure increases with increasing time constant. Equation 48 states that the time delay due to the mass acceleration effect increases when the time constant increases. However, as may be seen from equation 55 the measured pressure increases when the time constant increases. For pressure values close to the maximum, the effect of time constant variation is minimal.

From Eq. 38:

$$P_e(t) = P_{max} e^{-(t_o - t)/R_1C_1} =$$
$$P_{max} e^{-t_o/R_1C_1} \, e^{t/R_1C_1}$$

$$e^{-t_o/R_1C_1} = (F_1 + Mg)/AP_{max}$$

Thus

$$P_e(t) = \frac{(F_1 + Mg)}{A} e^{-t/R_1C_1}$$

APPLIED LOAD		MEASURED PRESSURE							
R_1C_1		40 ms		50 ms		100 ms		150 ms	
g	kPa	hex	kPa	hex	kPa	hex	kPa	hex	kPa
50	3.86	0A	1.62	0B	1.78	0F	2.43	10	2.60
100	7.72	24	5.80	25	6.00	29	6.65	2B	6.98
150	11.58	3B	9.57	3D	9.90	40	10.38	42	10.70
200	15.44	54	13.63	55	13.79	59	14.44	5A	14.60
250	19.31	6A	17.20	6C	17.52	70	18.17	72	18.50
300	23.17	7E	20.44	86	21.74	87	21.90	8A	22.29
350	27.10	98	24.66	9E	25.63	9F	25.80	A0	25.96
400	30.89	B3	29.04	B6	29.53	B7	29.68	B7	29.68
450	34.82	CE	33.42	CF	33.58	CF	33.58	D0	33.74
500	38.68	E4	36.98	E8	37.64	E9	37.80	E9	37.80

TABLE 4.7 PRESSURE VALUES FOR DIFFERENT TIME CONSTANTS.

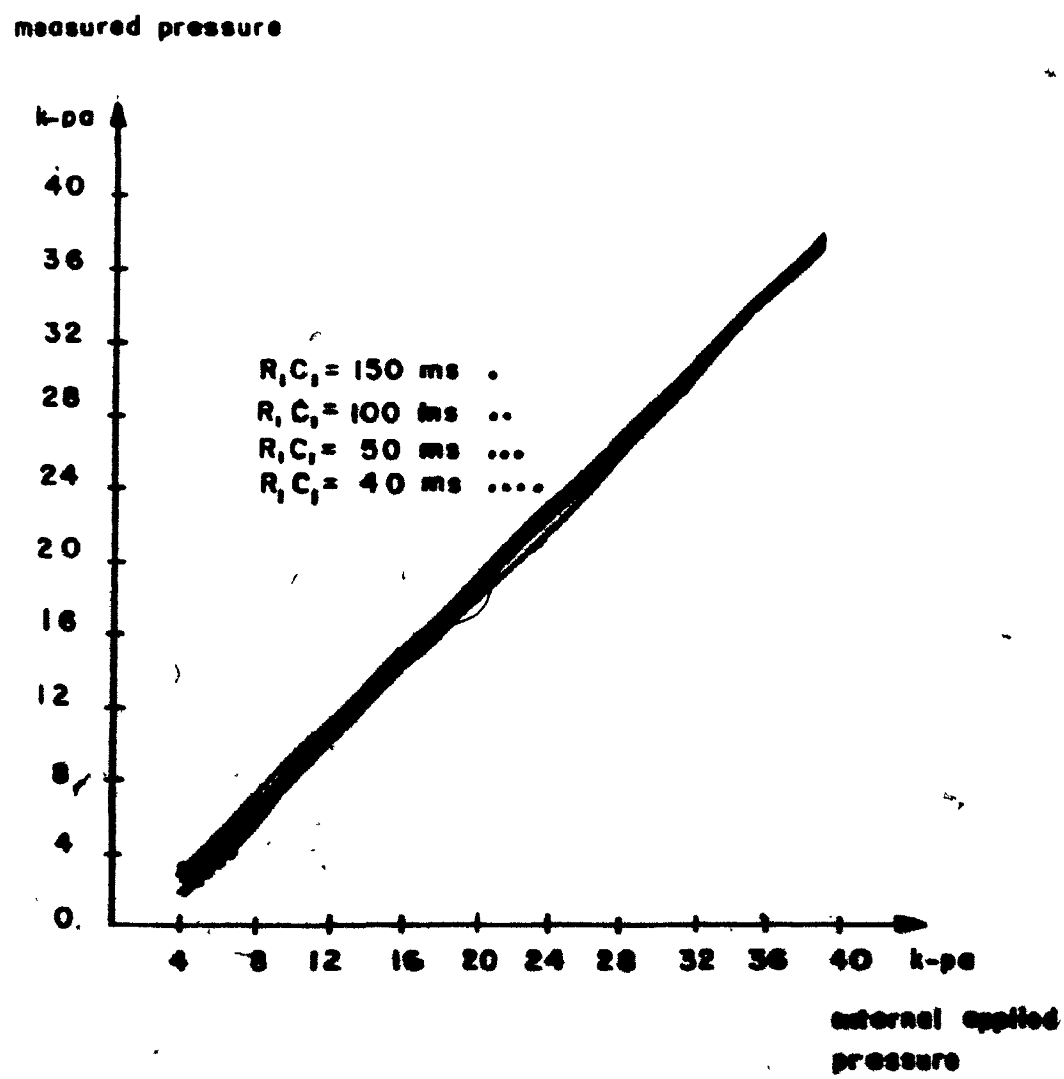

FIGURE 4-18 PRESSURE CELL CALIBRATION FOR DIFFERENT TIME CONSTANTS

and according to the simplified expression for t (Eq.48):

$$P_o(t) = \frac{(F_1 + M_2 g)}{A} e^{-\sqrt{\frac{X_1 K}{(F_1 + M_2 g)} R_1 C_1} / R_1 C_1}$$

and finally:

$$P_o(t) = \frac{(F_1 + M_2 g)}{A} e^{-\sqrt{\frac{X_1 K}{(F_1 + M_2 g) R_1 C_1}}} \quad (55)$$

4.8.2 Effect of the Variation of the Trigger Distance.

Each prototype piston can move a maximum distance of approximately 0.4 mm. The output voltage of the motion detector depends on the distance moved by the piston and the gain of the detector amplifier. When the piston is in its upper position, the output voltage has a minimum value. When the piston begins to move down, i.e., during a pressure decay test, the output voltage begins to increase. A zener diode limits the output voltage to 5 V. When this voltage reaches approximately 1.1 V the interface recognizes this value as a logic ONE. This instant is taken as the instant of the motion detector operation. Therefore, changing the gain of the detector amplifier is equivalent to changing the trigger distance. To analyze the effect of the trigger distance, four different detector amplifier gains were used. The set of standard loads was applied for each case. The time constant was maintained at 100 ms and a 6 PSI (41 kPa) maximum air pressure was applied. Results are listed in Table 4.8 and Figure 4-19 shows the corresponding curves.

It was observed that as the gain decreases, which means that the trigger distance increases, the obtained values of raw pressure data are decreased in agreement with equation 48.

4.8.3 Effect of the Variation of the Maximum Air Pressure.

The air pressure was increased by 16% to 7 psi (48 kPa) and the set of standard loads was applied while the pressure transducer amplifier gain was maintained constant and therefore, the output voltage increased to 6.3 V. Table 4.9 lists the values obtained for this condition. It was observed that these results are very similar to those obtained with 6 psi (41 kPa). The maximum difference is :

$$\frac{0.32}{37.64}\ 100 = 0.85\ \%$$

This means that the air pressure value at the beginning of a test is not critical. Equation 55 establishes that the measured pressure is independent of the value of the maximum air pressure in the case of decaying pressure.

APPLIED PRESSURE		MEASURED PRESSURE							
		gain=120		gain=82		gain=68		gain=47	
g	kPa	hex	kPa	hex	kPa	hex	kPa	hex	kPa
50	3.86	0B	1.78	0A	1.62	08	1.30	06	0.97
100	7.72	25	6.00	21	5.35	1F	5.03	1B	4.38
150	11.58	3D	9.90	38	9.09	33	8.27	32	8.11
200	15.44	55	13.79	52	13.30	49	11.84	49	11.84
250	19.31	6C	17.52	68	16.87	5E	15.25	5C	14.93
300	23.17	86	21.74	82	21.09	7A	19.79	79	18.63
350	27.10	9E	25.63	99	24.82	94	24.01	8F	23.20
400	30.89	B6	29.53	B5	29.36	AF	28.39	AE	28.23
450	34.82	CF	33.58	CB	32.93	CA	32.77	C9	32.61
500	38.68	E8	37.64	E6	37.31	E5	37.15	E4	36.99

TABLE 4.8 CALIBRATION DATA FOR DIFFERENT MOTION DETECTOR AMPLIFIER GAINS.

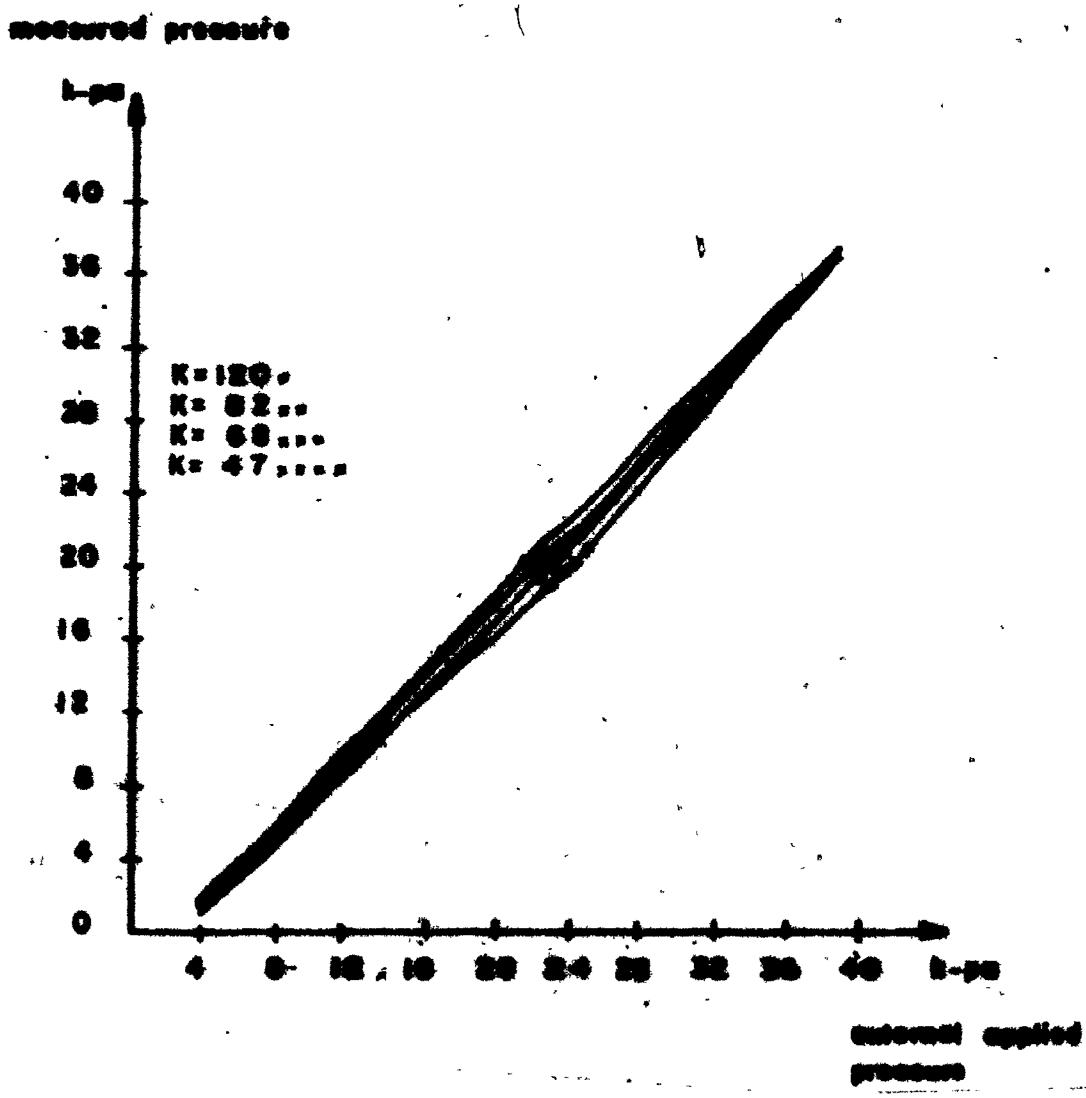

FIGURE 4-19 PRESSURE CELL CALIBRATION FOR DIFFERENT DETECTOR AMPLIFIER GAINS

APPLIED PRESSURE		MEASURED PRESSURE			
		MAX. AIR PRESS= 6 PSI (41 kPa)		MAX. AIR PRESS= 7 PSI (48 kPa)	
g	kPa	hex	kPa	hex	kPa
50	3.86	OB	1.78	OB	1.78
100	7.72	25	6.00	26	6.16
150	11.58	3D	9.90	3E	10.06
200	15.44	55	13.79	54	13.63
250	19.31	6C	17.52	6A	17.20
300	23.17	86	21.74	85	21.58
350	27.10	9E	25.63	9E	25.63
400	30.89	B6	29.53	B6	29.63
450	34.82	CF	33.58	CE	33.42
500	38.68	E8	37.64	E7	37.48

R_1C_1 = 50 ms.

TABLE 4.9 MEASURED PRESSURE FOR 6 AND 7 PSI (41 AND 48 kPa) MAXIMUM AIR PRESSURE.

4.9 Analysis of Results.

The system shows a threshold of approximately 1.5 kPa. For an applied pressure equal to or less than this value, the air pressure at the end of the time delay is zero.

The air pressure is read by an 8-bit ADC, therefore the resolution is:

$$\frac{38.68}{255} = 0.152 \text{ kPa/bit}$$

The static sensitivity corresponds to the slope of the calibration curve shown in Figure 4-17. This value may be calculated using Table 4.5 as:

$$\frac{E8 - 0B}{37.64 - 1.78} = \frac{232 - 11}{35.86} = 6 \text{ bits/kPa}$$

The system shows a proportional response when external pressures are applied to the cells, as observed in Figure 4-17. Deviation from linearity is 0.8%.

When the time constant of the pressure system is decreased lower values of pressure are obtained which is in agreement with the theoretical analysis. It is observed in Figure 4-18 that this decrement is smaller for higher pressures.

The effect of decreasing the detector amplifier gain, which is equivalent to increasing the distance required to trigger the detector, is to decrease the measured pressure. This is also in agreement with the theoretical analysis. It was observed that the decrements were larger for intermediate pressures.

A small increase of the initial air pressure (16 %) applied to the plenum had negligible effect upon the results.

In order to minimize the execution time of a test, measurements were done only with decreasing air pressure. Theoretically, similar results are obtained when the air pressure is increasing, i.e., if the plenum is charged rather then exhausted during a measurement cycle. External load was applied individually to each piston. As the prototype is small having only 16 pistons rather than the full complement of 256, it was not practical to make an actual test with a patient seated on the sensor plate.

4.10 Cost Estimation.

Cost for a 256-piston sensor has been estimated as follows :

Material	$
Aluminum 0.4 x 0.4 x 0.1 m	100.00
2 solenoid valves ($30.00 each)	60.00
1 air pump and accessories	800.00

1 pressure transducer..................................600.00

256 optoelectronic sensors............................650.00

Machining

100 hours at $ 30 per hour..........................3000.00

TOTAL $ 5210.00

If the sensor is built with commercial semiconductor strain gages, then the cost of 256 gages becomes :

Material

256 bridges $

4 gages each at $ 10 per gage......................10240.00

Cementing

50 man-hours at $ 30 each...........................1500.00

TOTAL $ 11740.00

Cost of the microcomputer system is similar for both solutions and it is estimated to be about $5000, including printer, plotter, CRT terminal, keyboard and dual diskette drive.

CHAPTER 5 CONCLUSIONS

5.1 Summary.

a) The proposed system exhibits a proportional response to the external pressure exerted on the pistons.

b) The system has good static characteristics. These characteristics are the following:

Linearity	:	0.8%
Accuracy	:	1%
Resolution	:	152 Pa/bit
Static Sensitivity	:	6 bits/kPa
Threshold	:	1.5 kPa

c) The data acquisition routine for a 256 piston sensor takes approximately 0.4 s and 12 kilobytes of RAM memory are required.

d) The data processing routine for the same number of cells takes approximately 1.6 s.

e) The results of the data processing routine are saved in a file on diskette by subprogram SAVE.

f) The data displaying program DISPLY uses the data in file created by SAVE to show the pressure distribution and list the local pressure exerted on each piston.

g) A 256-piston sensor has an estimated cost of $5200 which is much less than the cost of a system using commercial semiconductor strain-gages.

5.2 Limitations.

a) It is necessary that the contact pressure distribution remains fairly stationary during measurement. The data acquisition phase of a test lasts about 0.4 s.

b) Applied pressure must be normal to the surface of the sensor plate, otherwise the static friction will modify the results.

5.3 Recommendations for Future Work.

If the air pressure applied to the pistons could be individually controlled, then it would be possible to produce a relief of pressure at those points of excessive contact pressure. The system could work in two modes; as a measuring device and as an ulcer preventing device. In prophylactic mode, the pistons must have greater travel in order to effect the necessary reduction in local contact pressure. With selective pressure relief at proper intervals, high local pressures will be sustained only during short periods, in this way avoiding the generation of ulcers.

REFERENCES

1 Bush, Ch. — Study of Pressures on Skin Under Ischial Tuberosities and Thighs During Sitting. Arch. Phys. and Rehab., 50: 207-213, 1969.

2 Drummond, D.S., Narechania, R., Rosenthal, A., Breed, A., Lange, T., Drummond, D.K. — A Study of Pressure Distributions Measured During Balanced and Unbalanced Sitting. The Journal of Bone and Joint Surgery, 64-A: 1034-1039, 1982.

3 Frisina, W., Lehneis, H. — Pressure Mapping: A Preliminary Report. J. Biomech., 3: [illegible], 1970.

4 Garfin, S., Pye, S., Hargens, A., Akeson, W. — Surface Pressure Distribution of the Human Body in the Recumbent Position. Arch. Phys. and Rehab., 61: 409-413, [illegible].

5 Houle, R. — Evaluation of Seat Devices Designed to Prevent Ischemic Ulcers in Paraplegic Patients. Arch. Phys. and Rehab., 50: 587-594, 1969.

6 Knapp, R.
Bradley, P.
Medical Applications of an Electrical Pressure Gauge for External Use: Preliminary Report. Biomed. Eng., 5: 116-119, 1970.

7 Kosiak, M.
Kubicek, W.
Olson, M.
Danz, J.
Kottke, F.
Evaluation of Pressure as a Factor in the Production of Ischial Ulcers. Arch. Phys. and Rehab., 39: 623-629, 1958.

8 Kosiak, M.
Etiology and Pathology of Ischemic Ulcers. Arch. Phys. Med. and Rehab., 40: 62-69, 1959.

9 Kosiak, M.
A Mechanical Resting Surface: Its Effect on Pressure Distribution. Arch. Phys. and Rehab., 57: 481-484, 1976.

10 Lindan, O.
Greenway, R.
Piazza, J.
Pressure Distribution on the Surface of the Human Body: I. Evaluation in Lying and Sitting Positions Using a "Bed of Springs and Nails". Arch. Phys. Med. and Rehab., 46: 378-385, 1965.

11 Vroomen,L.C. SWTP 6809, 3-Dimensional Data Representation of Protein Analysis, DATAC Internal Report, 1983.

APPENDIX A

Dynamic Analysis of a Cell.

Case of Increasing Pressure.

Dynamic Analysis of a Cell. Case of Increasing Pressure.

The motion equation for the pressure increase case is given by expression 5:

$$AP_o(t) - F_1 - M_2 g = M\frac{d^2x(t)}{dt^2} + K\frac{dx(t)}{dt}$$

and the time of impending motion is:

$$P_{max}(1 - e^{-t_o/R_1C_1}) = \frac{F_1 + M_2 g}{A}$$

$$e^{-t_o/R_1C_1} = \frac{AP_{max} - F_2 - M_2 g}{AP_{max}}$$

$$-t_o/R_1C_1 = \log\left(\frac{AP_{max} - F_1 - M_2 g}{AP_{max}}\right)$$

and:

$$t_o = R_1C_1 \log\left(\frac{AP_{max}}{AP_{max} - F_1 - M_2 g}\right)$$

The distance moved by a particular cell may be obtained by:

$$AP_{max}(1-e^{-(t_o+t)/R_1C_1})-F_1-M_2g = M\frac{d^2x(t)}{dt^2} + K\frac{dx(t)}{dt}$$

Replacing the value of e^{-t_o/R_1C_1}:

$$(AP_{max}-F_1-M_2g)(1-e^{-t/R_1C_1}) = M\frac{d^2x(t)}{dt^2} + K\frac{dx(t)}{dt}$$

It is observed that the only difference between this expression and expression 23, which corresponds to the pressure decay case, is the constant factor $AP_{max}-F_1-M_2g$.

Therefore, the expression for the time to trigger the motion detector is:

$$t= \frac{X_1K}{AP_{max}-F_1-M_2g} + \sqrt{\frac{X_1^2K^2}{4(AP_{max}-F_1-M_2g)^2} + \frac{X_1K}{AP_{max}-F_1-M_2g}R_1C_1}$$

and the expression for the pressure is:

$$P_o(t) = P_{max} - \frac{AP_{max}-F_1-M_2g}{A}e^{-t/R_1C_1}$$

APPENDIX B

Listing of Program DATACQ

```
        NAM     DATACQ
        OPT     PAG
*
*********************************************************
*                                                       *
*     DATACQ    THIS PROGRAM IS SCANNING CONTINUOUSLY  *
*               THE STATUS OF THE SENSOR PLATE CELLS.  *
*               AFTER A WHOLE CELLS SCANNING, AN AIR   *
*               PRESSURE READING IS DONE.              *
*               DATA OF SENSOR CELLS IS STORED         *
*               SEQUENTIALLY FROM ADDRESS $0400        *
*               TO ADDRESS $3600                       *
*               DATA OF AIR PRESSURE IS STORED         *
*               SEQUENTIALLY FROM ADDRESS $0200        *
*               TO ADDRESS $ 0390                      *
*               AFTER 400 WHOLE SCANNING, ROUTINE      *
*               "DATA PROCESSING" DETECTS WHEN A CELL  *
*               HAS BEEN ACTIVATED AND ASSOCIATE TO    *
*               IT THE CORRESPONDING AIR PRESSURE.     *
*                                                       *
*********************************************************
*
*
*********************************************************
*               INITIALIZATION                          *
*********************************************************
*
        ORG     $0060       BOTTOM OF THE STACK
        RMB     30          ROOM FOR STACK
        FDB     START       INITIAL ADDRESS
*
*               VARIABLES, POINTERS AND REFERENCES
*
COUNT   RMB     1           # OF PAIR OF BYTES TO SCAN
POINT   RMB     1           # OF BITS TO SHIFT
*
BUFF    EQU     $18         ANALOG CHANNEL
REF     EQU     $FA         MIN PRESSURE TO START
FINAL   EQU     $3700       FINAL TABLE ADDRESS
*
*               PIA         ADDRESSES
*
DRA1    EQU     $E050       PIA 1
CRA1    EQU     $E051
DRB1    EQU     $E052
CRB1    EQU     $E053
DRA2    EQU     $E060       PIA 2
CRA2    EQU     $E061
DRB2    EQU     $E062
CRB2    EQU     $E063
*
*               CONFIGURE PIA 1
*

        CLR     CRA1
        CLR     CRB1
```

```
        CLR     DRA1
        LDA     #$06
        STA     CRA1
        LDA     #$FF
        STA     DRB1
        LDA     #$34
        STA     CRB1
        LDA     BUFF        SELECT ANALOG CHANNEL
        STA     DRB1
*
*               CONFIGURE PIA 2
*
        LDA     #$30
        STA     CRA2
        STA     CRB2
        CLR     DRA2
        CLR     DRB2
        LDA     #$34
        STA     CRA2
        STA     CRB2
*
        ORG     $0100
START   LDA     #$04
        STA     $E06E       SELECT BUFFER PIA 2
*
        LDA     #$00        CLEAN FINAL TABLE
        LDY     #FINAL
LOOP1   STA     ,Y+
        CMPY    #$3800      FINISH?
        BNE     LOOP1
*
        LDX     #$0200      AIR PRESSURE ADDRESS
        LDY     #$0000      DATA BYTES ADDRESS
*
*               FIRST START OF CONVERSION ORDER
*
CONV1   LDA     #$3C        CB2=1
        STA     CRB1
        LDA     #$34        CB2=0
        STA     CRB1
TEST    LDA     CRA1        CHECK IF CONV. IS READY
        BPL     TEST
        LDA     DRA1        READ AIR PRESSURE
        CMPA    REF         CHECK IF NORMAL
        BHS     OPENV       IF YES GO TO OPEN VALVE
*
*       SEND    ORDER  TO  OPEN  OUTPUT  VALVE
*
OPENV   LDA     #$3C
        STA     CRA2
*
*
*
*
*
*
```

```
***************************************************************
*                DATA      READING                            *
***************************************************************
*
REPRE   LDA     DRA1            READ AIR PRESSURE
        STA     ,X+             STORE IT
SOC     LDA     #$3C            NEW SOC ORDER
        STA     CRB1
        LDA     #$34
        STA     CRB1
        LDA     #$10            SET COUNT
        STA     COUNT
REDATA  LDA     DRA2            READ DATA BYTES
        STA     ,Y+             STORE IT
        LDB     DRB2
        STD     ,Y++
*
*       SEND PULSE TO COUNTER
*
        LDA     #$3C
        STA     CRB2            CB2=1 (PIA 2)
        LDA     #$34
        STA     CRB2            CB2=0
        DEC     COUNT           CHECK IF SCANNING IS COMPLETE
        BNE     REDATA          IF NO, READ NEXT DATA
        CMPX    #$0300          CHECK IF TEST IS COMPLETE
        BNE     REPRE           IF NO, DO A NEW SCANNING
*
*       SEND ORDER TO CLOSE OUTPUT VALVE
*
        LDA     #$34
        STA     CRA2
*
*
*
*
***************************************************************
***
*                               DATA                PROCESSING
*
***************************************************************
***
*
        LDX     #$0200          INITIA ADDRESS OF AIR PRESSURE
        LDY     #$0400          INITIAL ADDRESS OF DATA BYTES
        LDU     #$3700          INITIAL ADDRESS OF FINAL TABLE
DETECT  LDA     #$08            LOAD POINT WITH 8 BITS TO SHIFT
        STA     POINT
        LDA     ,Y+             ACC A WITH DATA BYTE
        EORA    31,Y            EXCLUSIVE OR WITH NEXT BYTE
        BEQ     ADDEIG          IF 0 GO TO UPDATE FINAL ADDR
SHIFT   LSRA                    IF NOT SHIFT RIGHT ONE BIT
        BCS     LOPR            IF 1 GO TO LOAD PRESSURE
        TST     ,U+             INCREMENT FINAL TABLE
DECR    DEC     POINT           CHECK IF ALL BITS ARE SHIFTED
        BNE     SHIFT           IF NOT GO TO SHIFT AGAIN
```

```
CHETAB   CMPU   #$37FF      CHECK IF LAST ADDR IS REACHED
         BNE    CHEDAT      IF NOT EQ GO TO CHECK DATA
         LDU    #$3700      IF EQ START FIRST ADDR AGAIN
         TST    ,X+         INCREMENT PRESSURE ADDR
CHEDAT   CMPY   #$3600      CHECK IF ALL DATA READY
         BNE    DETECT      IF NOT GO TO DETECT NEXT BYTE
         SWI
ADDEIS   TST    ,U++        UPDATE FINAL TABLE
         TST    ,U++
         TST    ,U++
         TST    ,U++
         BRA    CHETAB      GO TO CHECK LAST ADDRESS
LOPR     LDB    ,X          LOAD AIR PRESS IN FINAL TAB
         STB    ,U+
         BRA    DECR        GO TO CHECK BITS SHIFTED
 END START
```

APPENDIX C

Listing of Subprogram SAVE

```
                NAM     SAVE
 OPT PAG
*
*
*               SAVE - THIS PROGRAM CREATES A FILE
*                      TO STORE THE RESULT OF PROGRAM
*                      DATACQ.THIS FILE IS CALLED DA.DAT
*                      AND IT IS USED BY THE PROGRAM
*                      DISPLY TO SHOW THE CELLS PRESSURE
*
*
*
FMS             EQU     $D406
FMSCLS          EQU     $D403
RPTERR          EQU     $CD3F
SETEXT          EQU     $CD33
*
*
                ORG     $0700
*
*               INITIAL ADDRESS AND SETUP FCB
*
START           [illegible] #$3700
                [illegible] #FCB
                JSR     SETFCB
*
*               OPENING FCB FOR WRITE
*
                LDA     #2
                STA     0,X
                JSR     FMS
                BNE     SAV5
                LDX     #FCB
                LDA     #$FF
                STA     59,X
*
*               WRITING DATA
*
SAV3A           LDA     0,Y+
                JSR     FMS
                BNE     SAV5
                CMPY    #$37FF
                BNE     SAV3A
*
*               CLOSE   FCB   FOR WRITE
*
                LDX     #FCB
                LDA     #4
                STA     0,X
                JSR     FMS
                BNE     SAV5
                SWI
*
*               CHECKING ERRORS
*
SAV5            JSR     RPTERR
```

```
                LDD     #$3039
                STD     ERRNUM
                BRA     SAV7
*
SAV7            LDD     $D409
                BEQ     SAV8
                JSR     FMSCLS
                BNE     SAV8
                LDD     #$3132
                STD     ERRNUM
SAV8            SWI
*
*               SUBROUTINE SETFCB: THIS SUBROUTINE SET UP
*               THE FILE SPECIFICATION IN THE FILE CONTROL
*               BLOCK.
*
SETFCB          PSHS    X
                LDB     #64
SET1            CLR     0,X+
                DECB
                BNE     SET1
                PULS    X
*
*               DEFINE NAME AND EXTENSION OF THE FILE
*
                LDD     #$4141
                STD     4,X
                STD     6,X
                LDA     #01
*
                STA     3,X
                LDA     #07
                JSR     SETEXT
                RTS
FCB             RMB     320
ERRNUM          RMB     2
                END     START
```

APPENDIX D

Listing of Program DISPLY

```
10 REM*********************************************************
20 REM*  PROGRAM DISPLY                                         *
30 REM*  DISPLAY VALUES FROM FILE DA.DAT TO THE CRT.            *
40 REM*  THESE VALUES ARE CREATED BY PROGRAM DATACQ.            *
50 REM*  THE PROGRAM IS DIVIDED IN THE FOLLOWING PARTS :        *
60 REM*    - OPEN THE FILE, READ THE DATA AND PREPARE CRT       *
70 REM*      TO WORK IN GRAPHICAL MODE.                         *
80 REM*    - DRAW THE VALUES IN ( X,Y ) AXIS WHERE X IS THE*
90 REM*      SWITCH (S) DIVIDED IN 16 SETS OF 16 SWITCHES  *
100 REM*      EACH AND Y-AXIS WITH THE PRESSURE VALUE          *
110 REM*    - PRINT THE SWITCH NUMBER AND THE PRESSURE VALUE*
130 REM*    - RESTORE CRT ( OR PLOTTER IF USED ).               *
140 REM********************************************************
150 DIM J%(256),B$(18),C$(27),D$(3)
160 B$="  0 10 20 30 40 50"
170 C$="0   32 64  96 128160192224256"
180 REM*
190 REM*  OPEN FILE AND PREPARE VARIABLES TO SUBSEQUENT WORK*
200 REM*
210 OPEN OLD "DATOS.DAT" AS 1
220 KY=0.      : JJ=0.75   : I=0.78
230 INPUT "ENTER AMOUNT OF SWITCHES TO READ...:";N%
240 IF N%>256 THEN PRINT "RANGE IS BETWENN 1-256..." : GOTO 230
250 REM*
260 REM*  RECOVER DATA FROM FILE DA.DAT IN FLOPPY                *
270 REM*
280 FOR K%=1 TO N% STEP 1
290 INPUT #1,J%(K%)
300 NEXT K%
310 CLOSE 1
320 REM*
330 REM*  INITIALIZE CRT FOR GRAPHICS MODE ( OR PLOTTER )      *
340 REM*
350 MT%=1
360 CD%=0     : XV=0.      : YV=0.
370 GOSUB 1120
380 CD%=2      : REM MOVE TO (0.5,0.5)
390 XV=.5     : YV=.5      : GOSUB 1120
400 CD%=3     : REM DRAW LINE TO (7.5,0.5)
410 REM*
420 REM*  CREATE VIEWPORT WITH A SQUARE TO DRAW INSIDE          *
430 REM*
440 XV=7.5     : YV=0.5    : GOSUB 1120
450 CD%=2      : XV=7.6        : YV=0.62    : GOSUB 1120
460 LB$="SWITCH NU."  : LB%=1    : LR%=2    : GOSUB 1500
470 XV=7.5     : YV=0.5        : GOSUB 1120
480 CD%=3
490 XV=7.5   : YV=6.25    : GOSUB 1120
500 XV=.5    : YV=6.25    : GOSUB 1120
510 CD%=2       : XV=0.5    : YV=6.25   : GOSUB 1120
520 LB$="PRESS K-PA   "   : LB%=1    : LR%=2       : GOSUB 1500
530 XV=.5       : YV=6.25   : GOSUB 1120    : CD%=3
540 XV=0.5      : YV=0.5    : GOSUB 1120
```

```
550 C%=1
560 FOR KY=0.8 TO 6.25 STEP 1.09
570 CD%=2      : XV=0.01  : YV=KY    : GOSUB 1120
580 D$=MID$(B$,C%,3)
590 C%=C%+3
600 LS$=D$     : LS%=1    : LR%=2    : GOSUB 1500
610 CD%=2      : XV=0.45  : YV=KY    : GOSUB 1120
620 CD%=3      : XV=0.5   : YV=KY    : GOSUB 1120
630 NEXT KY
640 C%=1
650 FOR KX=0.8 TO 7.8 STEP 0.82
660 CD%=2      : XV=KX    : YV=0.01  : GOSUB 1120
670 D$=MID$(C$,C%,3)
680 C%=C%+3
690 LS$=D$     : LS%=1    : LR%=2    : GOSUB 1500
700 CD%=2      : XV=KX    : YV=0.46  : GOSUB 1120
710 CD%=3      : XV=KX    : YV=0.5   : GOSUB 1120
720 NEXT KX
730 REM*  DISPLAY VALUES WITH RANGE IN Y-AXIS FROM 0 TO 255 *
750 REM*
760 FOR K%=1 TO N% STEP 1
770 IF JJ<=1.50 THEN 790
780 JJ=0.75    : I=I+0.03
790 JJ=JJ+0.05
800 I=I+0.0238
810 KT=J%(K%)
820 KT=KT/75
830 JF=KT+JJ
840 CB%=2
850 XV=I
860 YV=JJ         :GOSUB 1120
870 CD%=3
880 XV=I
890 YV=JF         : GOSUB 1120
900 NEXT K%
910 CD%=4     : REM RESTORE CRT
920 XV=1.     : YV=.0     : GOSUB 1120
930 REM*
940 REM*  PRINT SWITCH NU. AND PRESSURE VALUE ON PRINTER *
950 REM*
960 OPEN "0.EPSON" AS 0
970 PRINT #0,TAB(10);"SWITCH NU.";TAB(30);"PRESSURE(K-PA)"
980 DIGITS 5.2
990 FOR K%=1 TO N% STEP 1
1000 KJ=J%(K%)*41.37*0.9847/255.+1.36
1010 PRINT #0,TAB(13);K%;TAB(34);KJ
1020 NEXT K%
1030 CLOSE 0
1040 REM*
1050 REM*  END OF THE PROGRAM
1060 REM*
1070 STOP
1080 REM*
```

```
1090 REM*  APPENDED SUBROUTINES WHICH ALLOWS TO USE GRAPHICS *
1100 REM*             IN CRT  ( OR PLOTTER )                  *
1110 REM*
1120 REM
1130 REM  *--< MAIN PLOTTING ROUTINE >--*
1140 REM
1150 IF CD%=0 OR CD%=16 THEN GOTO 1220
1160 IF (CD%<0 OR CD%>18) OR (CD%<>4 AND VF%<>1) THEN RETURN
1170 IF CD%>12 THEN ON CD%-12 GOTO 1360,1270,1270,1270
1180 IF CD%>8  THEN ON CD%-8  GOTO 1380,1390,1270,1270
1190 IF CD%>4  THEN ON CD%-4  GOTO 1270,1270,1210,1370
1200 ON CD% GOTO 1210,1270,1270,1360
1210 RETURN
1220 XF=1. : YF=1. : CA=1. : SA=0. : XG=0. : YG=0.
1230 XO=XV : YO=YV : XD=XV : YD=YV : VF%=1
1235 POKE HEX("C54F"),0
1240 POKE HEX("C550"),HT% : GOSUB 1430
1250 IF HT%=1 THEN CD%=0
1260 X%=0 : Y%=0 : GOTO 1340
1270 XD=XV : YD=YV
1280 XX=(XV-XO)/XF      : YY=(YV-YO)/YF
1290 XA=CA*XX-SA*YY+YG : YA=SA*XX+CA*YY+YG
1300 ON HT%+1 GOTO 1310,1320,1330,1330
1310 X%=XA/0.01      : Y%=YA/0.01      : GOTO 1340
1320 X%=XA/0.005     : Y%=YA/0.005     : GOTO 1340
1330 X%=XA*1024/10. : Y%=YA*780/7.0
1340 POKE CV,CD% : DPOKE VX,X% : DPOKE VY,Y%
1350 KV%=USR(1) : RETURN
1360 VF%=0 : X%=XV : GOTO 1340
1370 XF=XV : YF=YV : RETURN
1380 CA=XV : SA=YV : RETURN
1390 XO=XV : YO=YV : RETURN
1400 REM
1410 REM  *-< SET UP 'USR' INFO >-*
1420 REM
1430 CV=PEEK(HEX("CC2B"))*256+PEEK(HEX("CC2C"))
1440 POKE CV-2,HEX("E8") : POKE CV-1,HEX("OC")
1450 CV=HEX("C551") : VX=HEX("C552") : VY=HEX("C554")
1460 RETURN
1470 REM
1480 REM  *--< LETTERING ROUTINE >--*
1490 REM
1500 DPOKE CV,PTR(LS$)
1510 POKE HEX("C553"),LR% : POKE HEX("C554"),LS%
1520 KV%=USR(4)
1530 RETURN
1540 REM
1550 REM  *--< OBTAIN A POINT FROM THE 'BITPAQ' >--*
1560 REM
1570 KV%=USR(2)
1580 CV%=PEEK(HEX("C52F")) : XV%=DPEEK(HEX("C530")) : YV%=DPEEK(HEX("C532"))
1590 XV=XV%*0.005 : YV=YV%*0.005
1600 RETURN
```

```
1610 REM
1620 REM  *--< CHECK IF IN SYSTEM MENU >--*
1630 REM
1640 KV%=USR(3) : MV%=DPEEK(HEX("C534"))
1650 RETURN
```

APPENDIX E

Listing of Program 3-D

```
{
   Program to generate a three-dimensional surface
    plot of pressure data
}

program VIEW3D(MAINDATA,TEMPRUN);

{
   { System constants }
}
const
  MAXLEN=11.0; MAXPNTS=2000; LOWLIM=-10000; STEP=0.005; PI=3.14159;


{
   { System variables }
}
var
  MAINDATA,TEMPRUN:file of integer;

  VISIBLE: array[1..MAXPNTS] of integer;
  ZD:      array[1..500] of real;
  ZPLOT:   array[1..500] of integer;

  CUTFLG,DONE:boolean;

  XAXIS,YAXIS,YNEG,YPOS,XSCALE,ZSCALE,SKEW,
  ELEVATION,CSKEW,SSKEW,CELEV,SELEV,DX,DY,
  XSHIFT,ZSHIFT,PXINC,PZINC,X,Y,Z,
  YSPAN,XINX,YINX,XREAL,ZREAL,MAXSLICE,
  SIGMA,MU,TIME:real;

  FISIZE,NUMOFALL,I,IP,ITIME,LENGTH,LOCAT,
  INCR,FINAL,INCREMENT,XPNT,YPNT,IPOOP,IPASS,
  XSLICE,YSLICE,YP:integer;

  PLOTTINGDEVICE:byte;

  CROSS,ANSWER:char;
```

```
{
        **::::::::::::::::::::**
        *                      *
        :  GRAPHICS SECTION    :
        *                      *
        **::::::::::::::::::::**

   --< User functions >--
}
function PUSER(A,B,C,D:integer):integer;
 external $E81D;

{
   --< Initialize plotting device >--
}
procedure PLOTINIT(DEVICE:byte);
var
  {$ADDRESS=$C550 }
    PLOTTER:byte;
  {$STACK }

begin
PLOTTER:=DEVICE
end;

{
   --< Low-level plotting routine >--
}
procedure PLOT(IP,I,J:integer);
var
  K:integer;

begin
K:=PUSER(1,IP,I,J)
end;

{
   --< Procedure to plot the 'visible' line >--
}
procedure PLOTLINE(X:real;I:integer);
var
  IP:integer;

begin
IP:=2;

while I<>FINAL do
  begin
  I:=I+INCR;
  if YPLOT[I]<>-999 then
                                begin
                                XPNT:=trunc((X*XSCALE+XSHIFT)/STEP)+1;
                                PLOT(IP,XPNT,YPLOT[I]);
                                IP:=3
                                end
```

```
                              else
                                 IP:=2;
       X:=X+XINX;
       end;
    PLOT(2,XPNT,ZPLOT[I]);
    end;

    {
       --( Draw cross-hatching )--
    }
    procedure PLOTCROSS;
    var
      I,J:integer;

    begin
    reset(TEMPRUN);

    writeln('Number of passes=',IPASS:2);

    for I:=1 to NUMOFALL do
      read(TEMPRUN,VISIBLE[I]);

    I:=FISIZE div XSLICE;
    XINX:=XINX*I;
    X:=0.0;

    for ITIME:=1 to XSLICE+1 do
      begin
      IP:=2;  YP:=0;

      for I:=1 to YSLICE do
         begin
         LOCAT:=(I-1)*(XSLICE+1)+ITIME;
         if VISIBLE[LOCAT]<>-999 then
                                     begin
                                     IP:=3;
                                     YP:=VISIBLE[LOCAT];
                                     TIME:=I-1;
                                     XREAL:=X*XSCALE+TIME*PLINC;
                                     XPNT:=trunc(XREAL/STEP);
                                     PLOT(IP,XPNT,YP);
                                     end
                                 else
                                     begin
                                     IP:=2;
                                     TIME:=I-1;
                                     XREAL:=X*XSCALE+TIME*PLINC;
                                     XPNT:=trunc(XREAL/STEP);
                                     PLOT(IP,XPNT,YP);
                                     end;
      end;

      X:=X+XINX;
      end
    end;
```

```
{
        **::::::::::::::::::::::::::**
        *                            *
        :   MAIN GENERATION SECTION  :
        *                            *
        **::::::::::::::::::::::::::**


   --< Procedure to set up one line of Z data >--
}
procedure SETALINE(I:integer);

begin
repeat
  I:=I+INCR;

  ZREAL:=ZD[I]*ZSCALE+ZSHIFT;
  ZPLOT[I]:=trunc(ZREAL/STEP);

until I=FINAL;
end;

{
   --< Procedure to generate a 'visible' line >--
}
procedure LINE(X:real;I:integer);
const
  DELTA=10;

begin
while I<>FINAL do
  begin
  I:=I+INCR;
  XREAL:=X*XSCALE+XSHIFT;
  LOCAT:=trunc(XREAL/MAXSLICE)+1;
  XPNT:=trunc(XREAL/STEP);
  YPNT:=ZPLOT[I]+DELTA;

  if YPNT<VISIBLE[LOCAT] then ZPLOT[I]:=-999
                         else VISIBLE[LOCAT]:=YPNT;
  X:=X+XINX
  end
end;
```

```
{
        **:::::::::::::::**
        *               *
        :  DATA HANDLING :
        *               *
        **:::::::::::::::**

   --( Get raw data from disk )--
}
procedure RAWDATA;
var
  I,J,DATA,WHICHONE:integer;
  ANSWER:char;

begin

for I:=1 to FISIZE do
  begin
  read(MAINDATA,DATA);
  ZD[I]:=DATA*0.005;
  end;

end;

{
   --( Save cross-hatching data )--
}
procedure CROSSDATA;
var
  I,J,XSIZE:integer;

begin
I:=1; J:=1;
XSIZE:=FISIZE div XSLICE;

while I<=FISIZE do
  begin
  NUMOFALL:=NUMOFALL+1;
  write(TEMPRUN,ZPLOT[I]);
  J:=J+1;
  I:=(J-1)*XSIZE
  end;

end;
```

```
{
        **::::::::::::::::::::**
        *                      *
        :  INITIALIZE SECTION  :
        *                      *
        **::::::::::::::::::::**

   --< Get initial info >--
}
procedure WAKEUP;
var
  ANSWER:char;

begin

repeat
  write('Plotting device (S/P)? ');readln(ANSWER);writeln;
until (ANSWER='S') or (ANSWER='P');

if ANSWER='S' then PLOTTINGDEVICE:=2
              else PLOTTINGDEVICE:=1;


writeln;
repeat
   write('Length of X-axis? ');readln(XAXIS);writeln;
until (XAXIS>1.0E-4);

writeln;
repeat
   write('Length of Y-axis? ');readln(YAXIS);writeln;
until (YAXIS>1.0E-4);

writeln;
repeat
   write('Number of cuts in Y direction? ');
   readln(YSLICE);writeln;
until (YSLICE>0);

writeln;
repeat
  write('Cross-hatching? ');readln(CROSS);writeln;
until (CROSS='Y') or (CROSS='N');

if CROSS='Y' then
  begin
  repeat
   write('Number of cuts in the X direction? ');readln(XSLICE);writeln;
  until (XSLICE>0);
  end;

writeln;
repeat
   write('Skew and elevation angles? ');readln(SKEW,ELEVATION);writeln;
```

```
until (SKEW>1.0E-4) and (ELEVATION>1.0E-4);

writeln;
repeat
   write('Z scale? ');readln(ZSCALE);writeln;
until (ZSCALE>1.0E-4);

end;

{
   --< Set-up main variables >--
}
procedure SETVARIABLES;

begin
DY:=YAXIS/YSLICE;
if CROSS='Y' then DX:=XAXIS/XSLICE;
MAXSLICE:=XAXIS/FISIZE;

ZSCALE:=1.0;

SKEW:=SKEW*PI/180.0; ELEVATION:=ELEVATION*PI/180.0;
SKEW:=PI/2.0-SKEW;

CSKEW:=cos(SKEW); CELEV:=cos(ELEVATION);
SSKEW:=sin(SKEW); SELEV:=sin(ELEVATION);

YNEG:=-1.0; YPOS:=1.0;
YSPAN:=YPOS-YNEG;

XINC:=XAXIS/FISIZE;
YINC:=YSPAN/YSLICE;

XSHIFT:=0.0; ZSHIFT:=0.0;
PXINC:=SELEV*DY; PZINC:=CSKEW*DY;

NUMOFALL:=0; IPASS:=0;
LENGTH:=FISIZE;
FINAL:=LENGTH;
INCR:=1;
CUTFLG:=FALSE;

for I:=1 to MAXPNTS do VISIBLE[I]:=LOWLIN;

end;
```

```
{
        **::::::::::::::::**
        *                  *
        :  MAIN PROGRAM    :
        *                  *
        **::::::::::::::::**
}
begin
reset(MAINDATA);
read(MAINDATA,FISIZE);

MAKEUP;
SETVARIABLES;

if CROSS='Y' then rewrite(TEMPRUN);

PLOTINIT(PLOTTINGDEVICE);
PLOT(0,0,0);

Y:=0.0; IPASS:=0;
repeat
  INCREMENT:=1;
  RANDATA;

  CUTFLG:=not(CUTFLG);
  SETALINE(0);
  LINE(0.0,0);
  PLOTLINE(0.0,0);
  if CROSS='Y' then CROSSDATA;
  INCREMENT:=INCREMENT+1;
  XSHIFT:=XSHIFT+PXINC;
  ZSHIFT:=ZSHIFT+PZINC;
  Y:=Y+YINC;

  IPASS:=IPASS+1;
until (IPASS=YSLICE);

PLOT(2,0,0);
if CROSS='Y' then PLOTCROSS;
PLOT(2,0,0);

end.
```