

## **Middlesex University Research Repository:**

an open access repository of  
Middlesex University research

<http://eprints.mdx.ac.uk>

Kochekali, Hodayun, 1991.  
Force sensing enhancement of robot system.  
Available from Middlesex University's Research Repository.

---

### **Copyright:**

Middlesex University Research Repository makes the University's research available electronically.

Copyright and moral rights to this thesis/research project are retained by the author and/or other copyright owners. The work is supplied on the understanding that any use for commercial gain is strictly forbidden. A copy may be downloaded for personal, non-commercial, research or study without prior permission and without charge. Any use of the thesis/research project for private study or research must be properly acknowledged with reference to the work's full bibliographic details.

This thesis/research project may not be reproduced in any format or medium, or extensive quotations taken from it, or its content changed in any way, without first obtaining permission in writing from the copyright holder(s).

If you believe that any material held in the repository infringes copyright law, please contact the Repository Team at Middlesex University via the following email address:  
[eprints@mdx.ac.uk](mailto:eprints@mdx.ac.uk)

The item will be removed from the repository while any claim is being investigated.

**BEST COPY**

**AVAILABLE**

TEXT IN ORIGINAL IS  
CLOSE TO THE EDGE OF  
THE PAGE

**PAGE**  
**NUMBERING**  
**AS ORIGINAL**

# ***Force Sensing Enhancement of Robot System***

*by*

***Homayun KOCHEKALI BSc MSc***

*A thesis submitted to*

*the Council for National Academic Awards*

*in partial fulfilment of the requirements for the degree of*

*Doctor of Philosophy*

*The work was carried out*

*at the robotic centre of Middlesex Polytechnic in*

*collaboration with Welwyn Strain Measurements Ltd. (Basingstoke)*

*&*

*Keithley Instruments Ltd. (Reading)*

***November 1991***

**Dedicated to the living memory  
of Agha-Joon with R=1-Sin ☉.**

y9430541

Site <i>TP</i> <i>1841</i>	MIDDLESEX UNIVERSITY LIBRARY
Accession No.	71871152
Class No.	629.892 KOC
Special Collection <input checked="" type="checkbox"/>	Thesis Coll.

## **Acknowledgement**

There are many people to whom I wish to extend my most grateful thanks.

To Dr. Y.B. Kavina for his role as Director of Studies. His advice has been greatly appreciated.

To my supervisors, Dr. R.A. Whitaker of Middlesex Polytechnic, Professor T.M. Husband, Vice Chancellor of Salford University, and Dr. K.F. Gill of Leeds University, for their enthusiastic support and kindly guidance.

To my Industrial Collaborators Weiwyn Strain Measurements and in particular to Mr K. Copleston for his practical advice.

To the Polytechnic technical staff Mr N. Salam, who helped in manufacturing of hardware, and Mr J. Morgan for his support and kindly guidance in developing the electronic control unit.

To my colleagues at Middlesex Polytechnic, S. Shaheen, C.W. Wong, H. Feelib and in particular Dr. A. Nowrouzi my lifetime friend.

And last, but certainly not least, to my loving family, Ayda my wonderful wife and Manijeh my lovely mother-in-law without whose support I would not have had time to devote to this research.

## **Abstract**

At present there is a general industrial need to improve robot performance. Force feedback, which involves sensing and actuation, is one means of improving the relative position between the workpiece and the end-effector.

In this research work various causes of errors and poor robot performance are identified. Several methods of improving the performance of robotic systems are discussed.

As a result of this research, a system was developed which is interposed between the wrist and the gripper of the manipulator. This system integrates a force sensor with a micro-manipulator, via an electronic control unit, with a micro-computer to enhance a robot system. The force sensor, the micro-manipulator and the electronic control unit, were all designed and manufactured at the robotic centre of Middlesex Polytechnic.

The force feedback is provided by means of strain gauges and the associated bridge circuitry. Control algorithms which define the relationship between the force detected and the motion required are implemented in the software. The software is capable of performing two specific tasks in real time, these are:

- 1- Inserting a peg into a hole
- 2- Following an unknown geometric path

A rig was designed and manufactured to enable the robot to follow different geometric shapes and paths in which force control was achieved mainly by control of the micro-manipulator.

## *Table of Contents*

---

Research Aims .....	vii
Research Achievements .....	vii
Presentation of thesis .....	viii
System Operating Principle .....	x
<b>Chapter 1</b> .....	<b>1</b>
1.0 Historical Perspective and State of the Art in Robot Force Control .....	2
1.1 Passive Compliance .....	3
1.1.1 Inherent Robot Compliance .....	3
1.1.2 Additional Compliance .....	4
1.2 Active Compliance .....	5
1.2.1 Force Feedback Control .....	5
1.2.1.1 Classical Control .....	6
1.2.1.2 Adaptive Control .....	7
1.3 Special Actuators for Force Feedback Control .....	9
1.4 Review of Literature Survey .....	13
1.5 Re-Appraisal of Aims .....	14
<b>Chapter 2</b> .....	<b>15</b>
2.0 Factors Affecting Robot Performance Accuracy .....	16
2.1 Environmental Factors .....	16
2.2 Parametric Factors .....	18
2.2.1 Kinematic Parameters .....	18
2.2.2 Dynamic Parameters .....	20
2.3 Measurement Factors .....	23
2.4 Computation Factors .....	24
2.5 Application Factors .....	25
<b>Chapter 3</b> .....	<b>26</b>
3.0 Methods of Improving Robot Accuracy .....	27
3.1 Calibration Method .....	28
3.1.1 Robot Calibration .....	28
3.1.2 Workpiece Calibration .....	29
3.1.3 Sensor Frame Calibration .....	29
3.2 Open-Loop Methods .....	29



3.2.1	Measurement tasks	30
3.2.2	Modelling Tasks	30
3.2.3	Compensation Tasks	31
3.3	Closed-loop methods	31
<b>Chapter 4</b>		<b>33</b>
4.0	The Survey of Force Sensors Mounted on Wrist	34
4.1	Separation of Forces and Moments with Strain Gauges	38
4.2	Design of the Force Sensor	41
4.2.1	Manufacture of the Sensing Element	49
4.2.1	Sensing Element Material	52
4.3	Calibration of Force Sensor	52
4.4	Construction of A Geometric Path for the Robot to Follow	55
<b>Chapter 5</b>		<b>58</b>
5.0	Development of A Micro-Manipulation Device	59
5.1	Determination of PUMA 560 Robot Accuracy and Repeatability	61
5.2	The Fundamental Design Criteria	62
5.3	Design Considerations	63
5.3.1	Central support tube sub-assembly	66
5.3.2	Z-Translation Sliding Tube	66
5.3.3	Y-Translation Sliding Tube	67
5.3.4	X-Translation End-effector	68
5.3.5	Z-Translation Drive Mechanism	68
5.3.6	Y-Translation Drive Mechanism	70
5.3.7	X-Translation Drive Mechanism	70
<b>Chapter 6</b>		<b>71</b>
6.0	Data Acquisition	72
6.1	Errors Associated with Digitising a Signal	76
<b>Chapter 7</b>		<b>77</b>
7.0	Motor Control	78
7.1	Methods of Providing Positional Feedback Control	78
7.1.1	Pulse Width Modulation	78
7.1.1.1	Problems involved with the PWM control system	79
7.1.2	Pulse Position Modulation	79

7.1.3	Pulse Code Modulation	79
7.1.4	DC control	80
7.1.5	Binary Encoding	80
7.2	Selection of the control method	81
7.3	Implementation of Selected Method	82
7.3.1	Summary of Block Diagram	82
7.3.2	Role of PC-30A interface card within the control system	82
7.3.3	Detailed Explanation	84
7.3.3.1	The need for auto setup	96
<b>Chapter 8</b>		<b>99</b>
8.0	Software	100
8.1	Preliminary Program Operations	103
8.2	Peg-in-hole Strategy	114
8.2.1	Peg-in-hole Program Operation	116
8.2.1	Appreciation of the software	120
8.2.2	Example Screen Output	121
8.2.3	Programme 5: Peg-in-hole Insertion Algorithm	123
8.3	Path Following Strategy	130
8.3.1	Programme 6: Path Following Algorithm	133
8.3.2	Example Screen Output	136
<b>Chapter 9</b>		<b>137</b>
9.0	Discussions & Conclusions	138
9.1	Recommendation for Future Work	141
References		144
<b>Appendix I</b>	<b>Manufacturing of the force sensor</b>	<b>153</b>
	Description of programs for machining the sensing element and overload protection device.	154
	CNC programs for machining the sensing element and overload protection device.	157
	Detail Drawings	160
	Material selection and properties data	162
<b>Appendix II</b>	<b>Calibration of the Force Sensor</b>	<b>167</b>
	Tables of data due to loading and unloading of force sensor	168
	Bar charts to illustrate the effect of strain gauge hysteresis due to loading and	

unloading of the force sensor . . . . .	172
Calibration linear graphs where gradients were evaluated from to represent the individual constant 'K' for calibration matrix. . . . .	180
<b>Appendix III      Design of micro-manipulator . . . . .</b>	<b>186</b>
Accuracy and repeatability data plots of Puma 560 robot using Hewlett Packard laser measurement system type HP5526A. . . . .	187
Prototype Manufacture . . . . .	189
Prototype Assembly Instructions . . . . .	192
Layout Drawings . . . . .	195
Motors Specifications . . . . .	201
<b>Appendix IV      Motor Control Unit . . . . .</b>	<b>205</b>
Motor Control Unit Circuit Diagram . . . . .	206
General Description of PC-30A plug in board . . . . .	207
Data Sheets and general description of Monolithic 16-bit D-A converters DAC 703JP . . . . .	220
Data sheets and general description of Full Bridge Driver L6203 . . . . .	230
Data sheets and general description of the level shifter 14C89 . . . . .	246
Specifications of 2100 system, Strain Gauge Conditioner and Amplifier system . . . . .	253
<b>Appendix V      Publications . . . . .</b>	<b>254</b>
Six papers were published in National Conferences and International Journals during the course of this research work.	

## ***Research Aims***

---

The specific aims of this research programme are:

1. The development and implementation of hardware for force feedback techniques capable of reducing forces during peg in hole insertion and also in path following operations and thereby enhance the performance of the robot.
2. The design of a control algorithm to maximise the speed of operation of the robot whilst ensuring that contact forces remain within specified limits.

## ***Research Achievements***

---

To achieve these aims, the work programme had the following major objectives:

- 1-To make critical review of the state of the art in force feedback applied to "peg-in-hole" insertion and "path following".
- 2-To identify and categorise most of the factors (if not all) affecting robot performance and to suggest techniques for improvement.
- 3- Design and development of a hardware system capable of performing the required compensatory positional movements under force feedback control.

This included the design and manufacturing of:

I- a force sensor capable of measuring force and torque in six degrees of freedom (DOF).

II- a micro-manipulator with three DOF.

III- an electronic control unit for data acquisition and derive of the micro-manipulator through a 286AT computer.

- 4- Design and development of control algorithms for the developed hardware to perform two specific tasks. These are:
  - a) inserting a peg into hole, and

b) following an unknown geometric path.

5- The testing and proving of the final force feedback hardware and software.

## ***Presentation of thesis***

---

This thesis is divided into nine chapters.

Chapter 1 is a survey of the literature on force feedback techniques in robotic assembly and path following, followed by state of the art in robot force control: passive and active compliance are discussed, and active force feedback algorithms reviewed.

Chapter 2 investigates factors affecting robot performance.

Chapter 3 suggests methods of improving robot accuracy, followed by a proposed solution using a micro-manipulator.

Chapter 4 outlines the different types of wrist force sensor designed to date, followed by the new design of a force sensor incorporating an overload protection device capable of measuring force and torque in six degrees of freedom: design, manufacture and calibration of the force sensor is described in detail; the design and manufacture of a geometric path is described for the robot to follow as well as the testing of the final force feedback hardware and software.

Chapter 5 introduces the design and development of a micro-manipulator that can perform the high speed small scale movement required to implement fine motion force feedback.

Chapter 6 explores and discusses the problems in data acquisition: errors associated with digitising a signal are fully described, followed by a proposed solution for data acquisition.

Chapter 7 details the development of an electronic control unit, discussing several methods of control and the means of implementing a suitable control method: the whole system is then described, with the aid of a block diagram,

followed by a detailed analysis of each part of the designed circuit.

Chapter 8 introduces the development of control algorithms to perform four main operations; set up, sample, calculation of forces, and direction of movement, as preliminary software programs: the implementation of the developed system is then demonstrated by performing the two specific separate tasks, that of inserting a peg into a hole and of following an unknown geometric path, fully describing the program operations and strategies by itemising the algorithms.

Chapter 9 presents the conclusions of this work and some recommendations for future work are proposed.

## **System Operating Principle**

The principle of operation of the designed system is illustrated in Figure I. The robot end effector (paths b & c) consists of a micro-manipulator(block 4), and a gripper with force sensor (block 5).

The force sensor (block 5) sends eight analogue signals from the force sensor bridge (path d to block 6) to the PC-30 board's A-D converters mounted inside the computer (path e to block 1). The signals are first fed through a strain gauge conditioner and amplifier to enhance their properties (block 6).

The appropriate computer software then analyses the reaction forces calculated from the voltages representing the forces. This information is compared with the desired force level (path i). If the force level is outside the preset range the computer will send signals to the micro-manipulator (via path f), through a control circuit (block 7) to the three DC motors within the micro-manipulator (block 4) driven by a motor driver circuit (block 8). The position of the micro-manipulator is determined through the feedback loop (path h), and chosen by the computer software in such a way that the displaced force sensor experiences a force magnitude within a specified limit determined in the software. When this is achieved, the signal will be sent from the computer to the robot to make the next coarse movement if necessary. The same cycle is repeated, via a robot control connection (path a) and any additional control information, not part of the feedback loop, is indicated by path g.

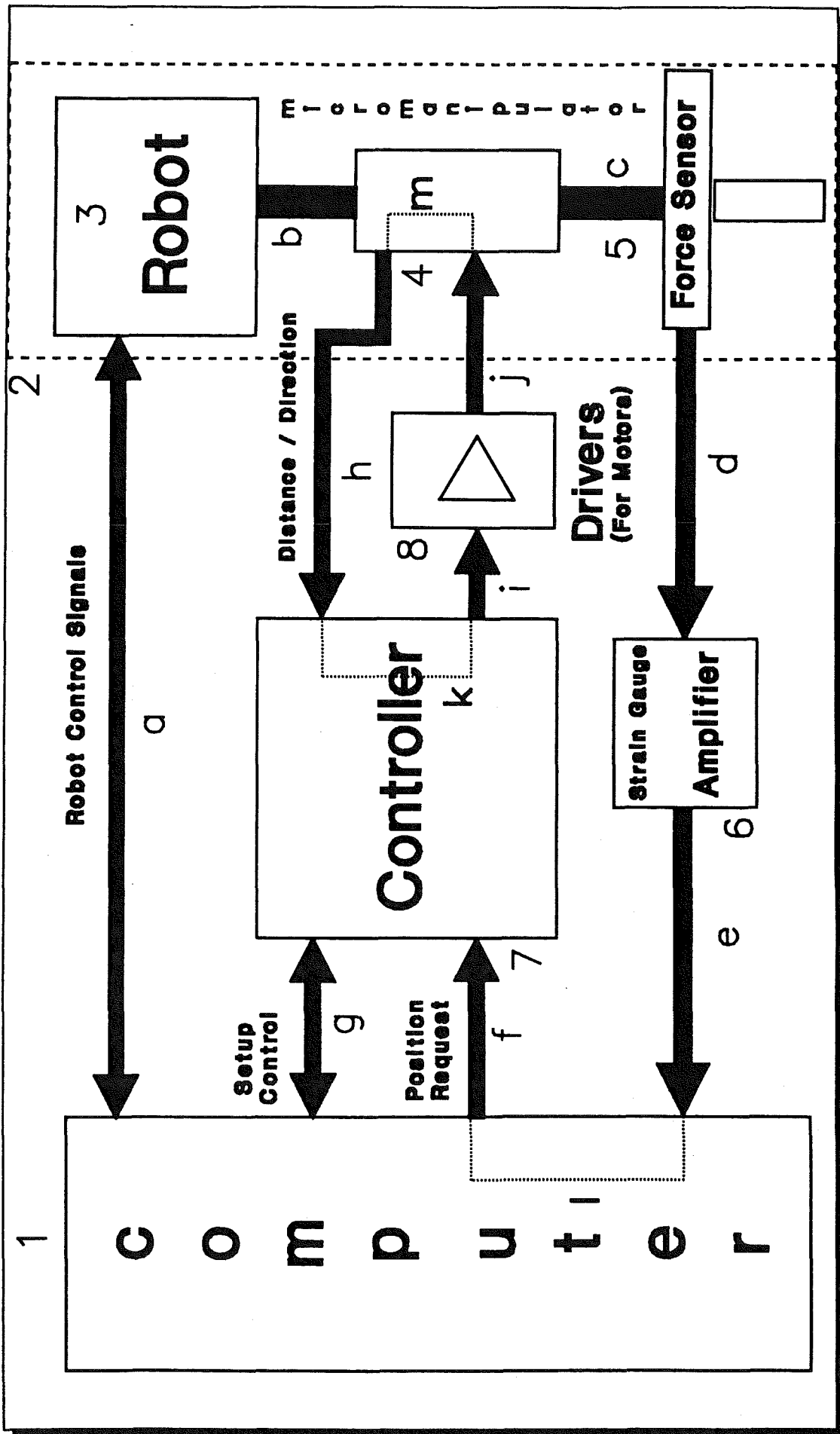


Figure 1 System operating principle

Figure 1 System operating principle



# ***CHAPTER 1***

## **State of the art & literature survey in robot force control**

## **1.0 Historical Perspective and State of the Art in Robot Force Control**

In robot environmental interactions, it is important to detect, locate and characterise the reaction forces so that this information can be used in force control.

Robot force control involves integration task goals, trajectory generation, force and position feedback, and modification of the trajectories. It requires understanding contact tasks so that effective strategies can be planned and feedback data can be understood.

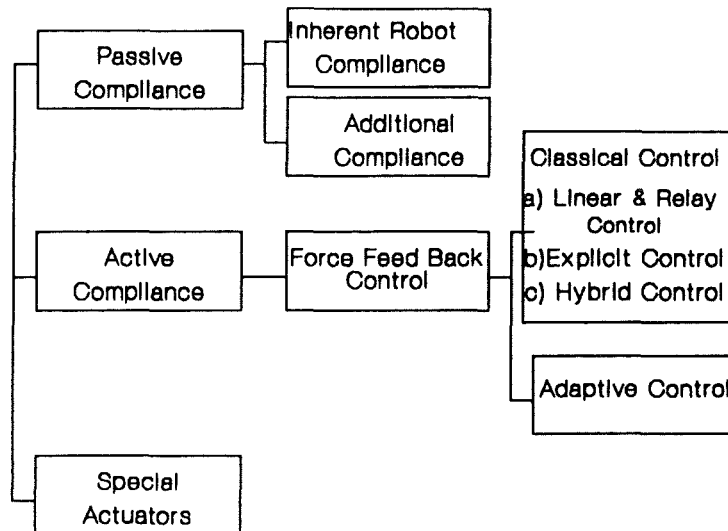
There are tasks in robotics which cannot be accurately performed using simple position control as achieved by programming a robot.

This is because of possible errors in the programmed path caused by eg, repeatability errors, dynamic factors, overshoot, undershoot, idling servoing and structural deflections, etc. One solution for a satisfactory performance of such tasks is to use the technique of force control. Robot force control with remote manipulator and artificial arm control was first used in the 1950s and 1960s [1].<sup>1</sup> In the late 1960s and 1970s, the first computer controls of force feedback were attempted. Continuous force control begins with work by Nevins, Whitney and Groome, and is based on multi axis force torque information obtained from 6 axis sensors, combined with multi-axis response motions. Continuous force control was applied to assembly and edge following tasks. Following such early efforts, several other methods have been developed.

A categorised sampling of these methods[1] is given in the following page :

---

<sup>1</sup> The sign [...] refer to a specific reference in reference section, throughout the thesis.



The brief description of each group is as follow:

## 1.1 Passive Compliance

### 1.1.1 Inherent Robot Compliance

Most robots exhibit some compliance as the joints move slightly and the links deflect elastically under load. This compliance allows the robot to compensate for small positioning errors, which becomes apparent when the mating parts touch for the first time. The inherent compliance in a robot is deemed to be advantageous, since a totally rigid robot would be unable to perform peg-in-hole assemblies without frequent jamming. Some robots have been designed with intentional compliance e.g. the SCARA(selective compliance Assembly Robot Arm) type robots [2].

Despite the possibility of parts damage, robotic assembly and assembly by hard automation is successful for large numbers of components by virtue of this inherent compliance. Sometimes additional compliance may be required, and there are various methods of supplying this.

### 1.1.2 Additional Compliance

Perhaps the best known of all passive accommodation devices is the Remote Centre Compliance, or RCC [2]. This is an entirely mechanical device which has the designed capabilities to act as a multi-axis float to provide lateral and angular alignment of components being assembled. One of the most important features of the RCC is that it can absorb lateral and angular positioning errors independently. i.e. it will move laterally in response to a laterally applied force but will not move angularly. This means that lateral misalignments do not develop into angular errors as insertion proceeds. It is commercially available and proven in industrial assembly tasks [3], but it does have some disadvantages. It is limited to applications in which the positional misalignment is less than the compliant travel range of the RCC (generally  $\pm 3$  mm). Another limitation is the need for lead-in chamfers. When the parts have no chamfer or a chamfer of less than the width of the positional misalignment of the peg, the peg is likely to "stub" on the edge of the hole and fail to enter [4]. In practice, part chamfers may be imperfect, or badly machined, thus rendering the RCC less effective. The RCC also suffers from a lack of versatility, in that the remote centre must always be near the tip of the part to be inserted. Unless all of the parts to be mated in an assembly are of a similar length different RCC's would be required for each part. There are other passive accommodation techniques which are not as widely exploited as the RCC. Karelin [5], as early as 1967, discussed the importance of compliance and proposed both elastic and electromechanical devices to allow small displacements of mating parts.

Stepourjine [6] presents the Automatic Insertion Module which achieves chamferless insertion using a pre-inclination strategy, where the energy stored in elastomeric pads pushes the parts together.

Yakimovich [7] discusses parts alignment by the air jet method while Drazan et al [8,9,10,11,12,13] present a pneumatic unit which acts as switchable low stiffness device. Torr and Cartwright [14] proposes another pneumatic device which aligns a hole in the work-piece with a tube through which the parts to be mated are fed. Although passive accommodation is used successfully in a number of assembly applications, its main limitation is that it cannot provide

precise control of the assembly forces. Although it can reduce assembly forces in a given environment, there is no guarantee of successful assembly, and no scope for recovery should jamming or any other unexpected event occur. To be able to control the forces a force measurement capability is required.

### **1.2 Active Compliance**

With active accommodation, the position of the manipulation device is updated according to sensory information. The two names, active accommodation and force feedback have become synonymous in the literature.

Active accommodation is used in robotic assembly applications because it can allow greater flexibility than passive accommodation. The knowledge of the forces prevailing during assembly can be used to modify the path of the robot from that originally programmed leading to a significant increase in effective positional accuracy [15].

Force feedback applications can be divided into two types [16]:

1-applications where the contact force is an essential part of the process itself e.g. grinding, polishing, deburring, etc.

2-applications in which the contact forces can be used as a source of information on the current position and orientation of the end effector, relative to its environment.

Most operations in assembly belong to this second category and the application of force feedback can enlarge the allowed region of uncertainty in parts alignment.

#### **1.2.1 Force Feedback Control**

Force feedback techniques use measured forces to generate a new movement trajectory, so that it is the relative positioning between the robot and its environment which is being controlled. In some implementations, the compliance of the end effector is controlled [16,17,18,19], but ultimately this affects the relative positioning. Force feedback techniques have been implemented using the basic robot as the positional compensation device

[21,24,25,27]. Such a typical application is detailed by Smith [27].

There are two types of force feedback control, known as classical and adaptive control.

Classical control is continuous and uses force information to control both force and position. This type of control includes methods like hybrid control, explicit control, and linear/ relay-linear control.

In adaptive control, force is used as a source of information about the process, and the state of the operation. Adaptive control may take various forms but generally relies on a series of logical decisions to influence the movement of the robot. This type of control has also been called the logic branching method [54].

#### 1.2.1.1 Classical Control

The application of force feedback can be considered as classical servo loop in the same way as the more familiar velocity and position servo loops. There are several different types of classical control.

Hybrid control [20,21,22,23] is where there is simultaneous control of position and force. The force experienced is subtracted from the required force and the position of the actuator updated. This control loop works in a similar manner to a normal position control loop, and the movement made is usually proportional to the force experienced. As forces arise only during contact with the environment, a basic error driven force feedback system can allow large forces to arise. Feed forward compensation can be applied to reduce this problem [22,24], by generating trajectory control laws based on a kinematic modelling of the process.

Another similar control technique is to use an explicit force control law which feeds the sensed forces back to a position or velocity controller [54,25]. This technique is suited in applications where the actuator has to adopt to restrictions in its workspace.

Other classical control techniques include linear and relay linear control [26].

Linear control can ultimately make the system oscillatory or unstable, and is similar to the hybrid and explicit force control schemes in this respect. Relay-linear control stabilises the assembly system by introducing a "dead zone of forces in which no corrective movement is made.

Although there are some examples [54,23] where peg in a hole assembly has successfully been performed using classical control, they are usually slow. Problems often arise in the coordinate transformation between sensor, gripper and workpiece reference systems, and the solution of this results in complex feedback matrices which require lengthy evaluation. Classical control is usually used in situations where the application of a nominal force is a requirement and there is a unique and time independent relationship between the force vector and corresponding corrective motion [17]. This includes applications like grinding and polishing which are essentially contour following operations.

### 1.2.1.2 Adaptive Control

Adaptive control may take many different forms, depending on the type of operation being performed. Adaptive control is usually performed via a series of logical control decisions, and lends itself to easy implementation on micro-computers.

Adaptive control algorithms can take several different forms. The simplest of these is based on binary contact sensing, where robot motion is stopped as quickly as possible should contact be made [9,10,54,23,27].

Using several sensors, motion need only be stopped in the direction in which contact has been made. More force information is needed before complete control can be exercised.

There are numerous applications for adaptive control, one of which, of course, is the peg-in-hole insertion task. A typical example of peg-in-hole insertion under adaptive control using a binary search technique, is given by Inoue [28]. The problem here is to put a spacer onto a vertical shaft. Three possible landing situations, are shown in Figure 1.1.

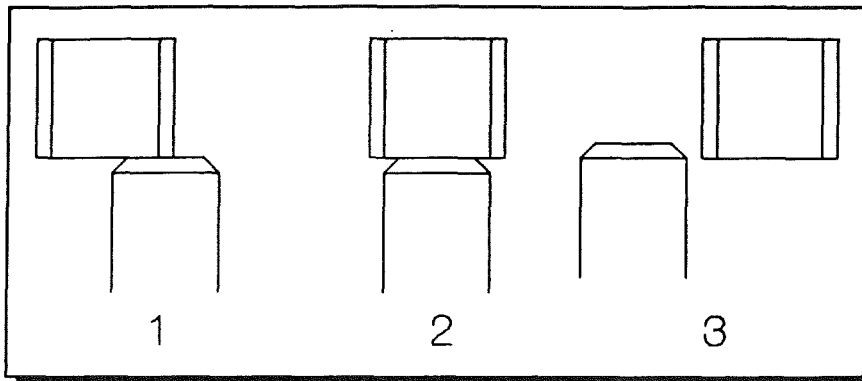


Figure 1.1, Three possible first contact situations when putting a spacer onto a peg

ie. the spacer is partially offset(1), nearly centred(2), or totally offset relative to the peg(3). A different search strategy must be used for each situation. To avoid the need for three positioning strategies, a deliberate positioning error is introduced so that the spacer is always offset to the right. Thus only a simple search to the left is required. This is shown schematically in Figure 1.2.

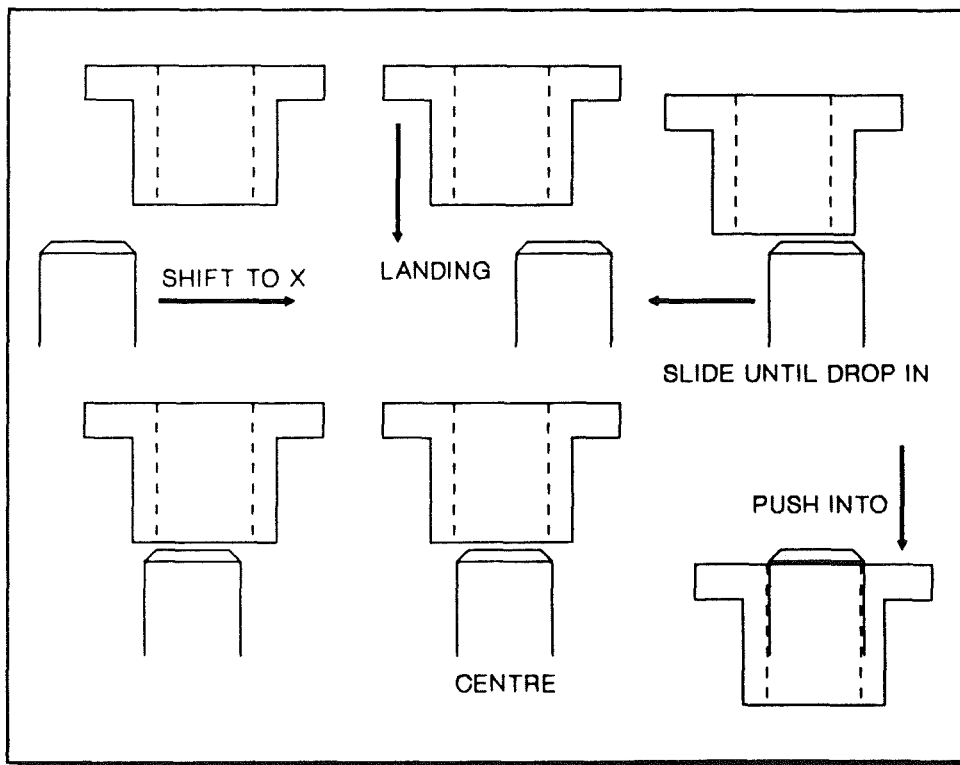


Figure 1.2, Schematic of an adaptive solution

Following the deliberate offset, the two parts are brought into contact. This is detected by a change in the force level. The next step is to find the centre of the spacer relative to that of the shaft. The shaft is moved along to the left until it partially falls into the hole. This can be detected by a sudden drop in the



insertion force. The movement is continued until a rise in side force is detected, signifying that the far side of the spacer is in contact with the shaft. The shaft can then be moved midway between these two contact points, where the insertion phase can begin.

This approach works for loosely fitting parts only. Smaller clearances require a different strategy with the spacer tilted one way for the initial alignment phase, and then tilted back to normal again once the insertion is under way [28].

This strategy is typical of adaptive control. The force information here is not being used to its full extent, and for this application, a simple thresholding sensor is used to signal when the force exceeds a pre-set level.

Similar strategies to this have been proposed by other workers [10,19,27,29].

### 1.3 Special Actuators for Force Feedback Control

A common problem in the area of force feedback control is that the robot is sometimes incapable of the fine control actions required of it by the control strategy. Although a six degrees of freedom robot should have the articulation to cope with most required movements, it often incapable of small corrective movements or adjustment. The relative positioning of the parts may be improved by the application of force feedback, but the resolution of the robot is not.

In addition, the large inertia of industrial robots make them unable to perform small high speed positional movements. The calculation of joint movement commands is lengthy, causing time delays. Lastly, most robot controllers used in industry today lack the communication and processing capability to be able to cope with sensory information [22]. Even when this information is processed by external computer, it is often not possible to communicate movement commands to the robot controllers, due to its limited communication facilities.

Some researchers have devoted their attention to the development of more accurate and more finely controlled robots. However, these tend to be expensive research tools and are generally not commercially available. It is argued by some workers [16,18] that this approach adversely affects the

universality of a robot.

Other avenues of research use some separate device to provide controllability and high resolution. Usually the robot makes the coarser, more approximate positioning moves, while this device makes small, fine corrective motions, using between two and six degrees of freedom. This coarse/fine system allows the use of less accurate, and hence cheaper, robots. This becomes more accurate, if more expensive, with the addition of a special actuator. This sort of actuator can also be used to improve the relative positioning between parts in a hard automation system.

The actuator can be mounted [8,9,16,18,19,25,31,32,33,34] on the robot end effector or it can be positioned on a special worktable with one of the parts fixed to it [9,10,25]. One major advantage of using a wrist mounted actuator is that it can easily be applied to different makes and models of robots and does not limit the convenience of the working region of a robot: this is the method of improvement chosen for this reported research at Middlesex Polytechnic. It can also be applied to the two types of force feedback applications mentioned earlier. The advantage of mounting the actuator on the worktable is that some of the robot payload is not sacrificed and it is also very suitable for assembly. However, it does limit the effective area where feedback control can be applied.

One of the earliest of these special devices was the Hitachi Hi-T-Hand which was developed as early as 1974 [34]. This robot insertion system incorporates a positioning mechanism, a tactile force sensor, and a flexible wrist. This arrangement is shown in Figure 1.3. The robot system shown comprises a main robot, an auxiliary robot, and parts supply devices. The auxiliary robot picks up the object with the hole from the parts supply and places it securely in the working position. The main robot picks up the shaft and performs the insertion operation under force feedback control.

Making use of sensors in the wrist, the relative positions of the shaft and hole are detected, and these are then controlled by moving an appropriate robot axis. There are several Hi-T-hands for use in different applications.

Many other micro-manipulation devices have been developed mainly in

assembly operation.

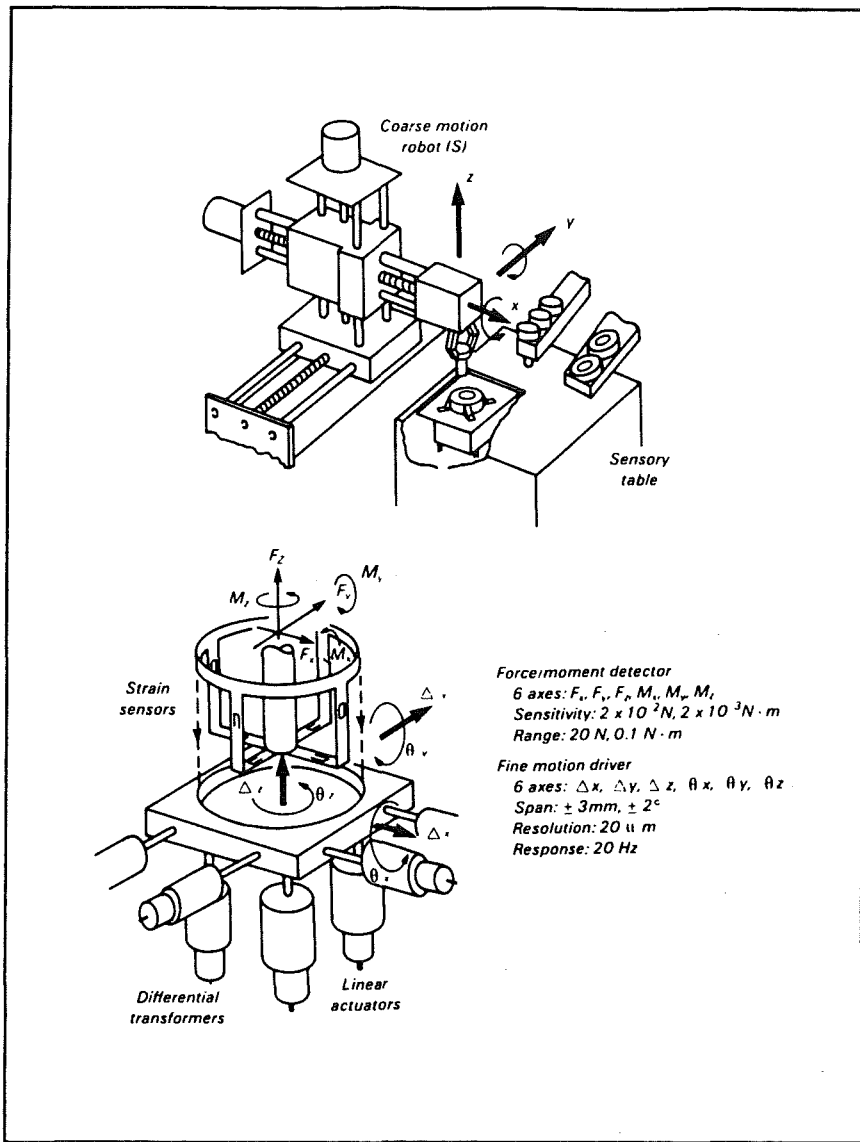


Figure 1.3, Hitachi experimental learning system consists of a simple robot with multi-directional force sensor on the table.

Stepourjine [6] proposed an automatic Insertion Module for light robotics. This device is attached to the robot wrist, and achieves insertion using a pre-inclination technique. Once the peg has been centred in the hole using force feedback, axis alignment is restored. This device has a remote centre through which its compliance effectively acts, and can also act as a passive device with force feedback turned off.

Vachtsevanos et al [35] proposed the novel use of a micro-manipulator which incorporates a spherical motor at the end effector of the robot. This provides

possibilities for combining roll, pitch and yaw motions in a single joint. Motion in the other three axes appears to be provided by the manipulator itself. Force sensing is achieved by monitoring the motor load, and the use of an artificial intelligence based heuristic fuzzy logic control algorithm is proposed, based on the force/position control and constrained space first presented by Raibert and Craig [22].

Asakawa [30,31] describes VACCOM a variable compliance device which is able to absorb positioning errors. This device also incorporates force sensing, removing the need for a separate sensor.

Drazan, Hopkins and Bland [9,10,11,12,13,29] describe several different sensing/micro-manipulation stages with control strategies for assembling different components.

Another special device for insertion tasks is the Active Adaptable Compliant Wrist (AACW), developed at University of Leuven, Belgium [16,17,18,19,36]. This device is mounted at the end of the robot arm and has three translational and two rotational axes.

Purnell et al applied the force feedback control in robotic meat cutting and deboning, using database information. A robot drives a powered knife through the carcass utilising force feedback control to follow bone profiles thus removing meat from a beef forequarter[55].

#### **1.4 Review of Literature Survey**

Positioning errors in path-following and assembly operation can cause large contact forces to arise between mating parts. These may deform mating parts, and wedging or jamming can occur in assembly operation causing damage to the end effector and possibly also damaging the robot, or workstation.

The RCC device is a means of providing extra compliance in the assembly system, whilst reducing the possibility of large contact forces and the likelihood of parts wedging and jamming. As with any passive device, the RCC can only work within certain limitations, and hence there is limit to its versatility.

Active force feedback provides similar force reduction abilities over a wider working range. This method varies widely in implementation and usage.

Force feedback can be implemented directly on a robot. This results generally in a slow moving, relatively inaccurate, system suited for applications such as contour following, where accuracy may not be paramount.

For assembly tasks requiring fine motion force feedback, a highly accurate micro-manipulator is necessary to increase the basic accuracy of the robot system. Such a micro-manipulator should be of low mass, thus allowing high speed corrective motions to the path of the end effector.

Although this literature search has reviewed numerous methods for implementing force feedback, all the control algorithms used so far are quite simple ones and lack the benefit of any controlling "intelligence" of the task in hand. This comment has been made by several workers in the field, notably Witney [54]. Many insertion algorithms use, at times, a trial and error control policy in the area where the relationship between forces and positional errors becomes ambiguous.

In a situation where forces are to be reduced there exists the problem that, if the forces become zero, there will be insufficient information to guide the movements of the robot end effector. In addition to this there can be times during the peg-in-hole assembly task when the direction of the force vector becomes impossible to ascertain using standard techniques so that good control of the system is difficult.

These factors suggest that it is not possible to rely solely on current force levels to generate the control strategy and it is here that more advanced control techniques are necessary.

### 1.5 Re-Appraisal of Aims

This literature survey has revealed numerous implementations of force feedback techniques, and it would appear that some specialised micro-manipulation device is usually employed to effect force reduction in peg-in-hole insertion tasks. Most of the control algorithms used seem simple and are often

error driven and yet, judging by the large amount of research in this field, a simple human task like peg-in-hole assembly is a highly complicated task in automated assembly. More complex control algorithms are needed to eliminate the trial and error approach described in the literature for use in the region where the force/position relationship becomes ambiguous. This justifies the first aim of this work, which is to develop algorithms capable of reducing contact forces during insertion and in the path following.

An area of research that has not been addressed is that of developing a control algorithm capable of doing peg-in-hole assembly and also of following a specified path. As far as it is known, no attempts have been made to consider robotic force control in general in order to discover common elements that may exist in a variety of such tasks. This justifies the second aim of this work.

Thirdly, an attempt was made to identify and categorise most of the factors (if not all) affecting robot performance and suggesting improvement techniques. The significance of this analysis is that robot users can now determine the combination of error sources that have the most influence on the performance of a robot being characterised.

This analysis lead to a proposed solution using a micro-manipulator.

# ***CHAPTER 2***

## **Factors affecting robot performance accuracy**

## **2.0 Factors Affecting Robot Performance Accuracy**

Factors affecting robot accuracy are identified in this section. Several methods for the improvement of robot accuracy are discussed. It is important that users of robots understand the factors affecting accuracy and where present technology stands in making accuracy improvements.

Robot accuracy can be influenced by a number of factors. They are classified into the following categories:

- (i) "Environmental" factors, such as temperature, humidity and electrical noise;
- (ii) "Parametric" factors, including kinematic parameters such as robot link lengths, joint zero-reference angles, dynamic parameters such as structural compliance, drive-train compliance, friction parameters, varying inertia, and other non-linearities such as hysteresis, backlash etc.;
- (iii) "Measurement" factors, such as resolution and non-linearity of encoders, resolvers etc.;
- (iv) "Computational" factors, such as robot path computation errors from digital computer round-off, steady-state control error and etc.;
- (v) "Application" factor, such as installation errors, part presentation errors, errors in defining workpiece coordinate frame etc.;

Each category of inaccuracy may not be totally independent from the others. For instance, temperature change affects link length, friction coefficients, drifts in control electronics and measurement sensitivity, etc.

### **2.1 Environmental Factors**

Temperature variations can cause link length changes in structural components and drive-train components. For instance, a change of 10 degrees Fahrenheit (typical from day to night in plant environment) will cause a 1000 mm aluminium structure to change by  $(1000\text{mm} \times 0.13\text{E-}4 \times 10 =)$  0.13mm. With the same temperature change a 600 mm steel ball screw can change length by (600



$\text{mm} \times 0.08 \times 10^{-4} = 0.048$  mm and if the distance of the end-effector is 8 times that of the moment arm of the ball screw, the resulting actual change of end-effector position can be 0.384 mm. In some applications this change is acceptable but in others, requiring precision, this may not be tolerable. Temperature affects long-term repeatability and accuracy and needs to be carefully considered in robotic applications.

Temperature and relative humidity affect characteristics of lubricants used in drive trains. Common lubricants used in robots are non-Newtonian and their rheology behaviour depends on how they have been used. The amount of lubricants on the drive elements varies depending on design and service conditions. It is normally difficult to achieve a reliable measurement of frictional characteristics in working industrial robots. Temperature variations affect dimensions of rotating parts of rotary drives and these in turn cause changes in friction, Coulomb friction and viscous friction values. Clearance in drive train components and hysteresis characteristics are similarly affected. Such friction related factors affect the accuracy of robots and cause difficulties in control.

Fibre-reinforced composite materials are sometimes used in robot structures. By combining fibres of different coefficients of thermal expansion, an arm can be designed so that there is little effective change of arm length within a range of temperature. This is one possible solution in designing extremely accurate robots. However, resins used in fibre-reinforced composite material are also subject to changes in chemical composition within the working environment of the robot, e.g. in an extremely acidic or caustic environment, the structural rigidity of a fibre-reinforced composite arm can suffer.

Typically, a robot controller contains a number of analogue parts such as resistors, capacitors and operational amplifiers. There can be slight drifts in characteristics of these electronic components that nominally do no harm at all. However, they may contribute several "counts" of offset values in each joint of the robot when ambient temperature varies or electrical noise is randomly injected. For a typical resolution of 0.001 mm per encoder count, a drift of 10 counts can mean 0.01 mm at end effector, and the cumulative drift from several joint axes can be several times greater than 0.01 mm.

## 2.2 Parametric Factors

Robot parametric factors include kinematic parameters and dynamic parameters. Inaccurate representations of robot parameters contribute to robot theoretically predicted inaccuracy. Generally, kinematic parameters affect "pose" accuracy and dynamic parameters affect "trajectory" accuracy.

### 2.2.1 Kinematic Parameters

The most noticeable kinematic parameters are robot link length parameters. Link lengths are normally specified by design. They can vary due to environmental factors and due to manufacturing and assembly errors. A link length can be different from its normal value due to machining tolerance and tolerance stack-ups from a number of sub-assemblies. For each machined piece of a structure, machining tolerance is generally around  $\pm 0.05$  mm. If a robot link were to be assembled from three pieces fastened together, overall length of the link due to tolerance stack-ups could vary by  $\pm 0.05 \text{ mm} \times \text{SQRT } 3$  i.e. 0.085mm RMS. For a three-jointed robot, end-effector position could then vary by  $\pm 0.085 \text{ mm} \times 3$  i.e.  $\pm 0.255$  mm which would be a relatively significant amount of error for many robot applications. Quality inspection and retrofit at final stage of machining will help reduce such error. It would add cost to the robot components if accuracy were so desired.

Kinematic parameters include offset distances between each pair of adjacent joint axes. Offset distances can be different from nominally designed values by a similar amount as indicated in the last paragraph. When assembling robot manipulators, some link lengths or drive shaft lengths are slightly modified to make all pieces fit together. With careful design and precision machining many of the "fitting" tasks can be reduced in sub-assemblies but not totally eliminated in final assembly of the entire machine.

Angularity between each pair of adjacent axes also contributes to inaccuracy of the robot. Angularity can result from assembly error when fitting sub-assemblies together, or error in the seating of bearings supporting the axes. Just an error of 0.1 degree in angularity could cause a link offset error of 1.745 mm when the two adjacent axes were at 1000 mm apart.

Many industrial robots in use today adopt the design of a so called "ideal" robot [37,38]. "Ideal" robots have closed-form inverse solutions such that desired Cartesian motions of robot end-effector can be converted to joint angle commands directly. This allows fast computation in the robot controller so that robots can follow desired programmed paths in real-time. Typical "ideal" robots contain three adjacent axes of rotation which intersect at a common point or at infinity (in this case, the three adjacent axes are parallel). Realistically, errors caused by kinematic factors due to manufacturing and environmental factors often make "ideal" robots non-ideal, thus affecting accuracy.

The other kinematic parameters that contain inaccuracy are the zero-angle position of robot joints. The zero-angle position of each joint can contain errors due to imprecise seating of encoder/resolver, or improper procedure in determining the zero-angle position. As stated before, an error of 0.1 degree in defining zero-reference angle can cause an inaccuracy of 1.745 mm at 1000mm away from the centre of that particular axis. Whereas all the kinematic parameters contribute to errors in accuracy, they generally do not contribute error in repeatability.

Drive-train errors include non-uniformity of belts, chains, gears and lever mechanisms in their range of motion. They cause inaccuracy in the workspace of robot. Belts and chains by their vary nature are elastic and there will be inaccuracies associated with the extension of the belt and chain when power is transmitted. Gears introduce backlash into the transmission system which will cause inaccuracies once the direction of rotation is reversed. The error is a cumulative one and will increase as the gears wear. Lever mechanisms may introduce large backlashes into the system. This is a result of the necessity for clearance fits for freedom of movement. In order to improve the performance of the robot, more accurate, finer and more selective power transmission system can be employed to minimise the effects of joint clearances. These, including hardware techniques for improving the system anti-backlash mechanism and so greatly improving the performance characteristics. However, this significantly increases the cost of the robot. Also, by careful measurements of gear train motions, gear train error can be mapped in robot software for accuracy improvement [39].

### 2.2.2 Dynamic Parameters

Structurally, robots are designed to be more flexible than machine tools, otherwise robots can become heavy and inefficient. Being flexible, robots will deflect under load and under their own weight. There are several components of robots that contribute to flexibility, i.e. 1) bearings, 2) drive train components and 3) link structures. Each one of these three components (structure, bearing, drive-train) can contribute as much significant flexibility as that of the other two, depending on the particular design of robot mechanism. There is no fixed rule in determining flexibility values for industrial robots. All the above components contribute to robot inaccuracy when payload changes at the robot end-effector. Bearings are usually designed not only to carry loads at desired speeds with given life, but also to have small angular deflection under moment loading. Drive-train components are designed with overall stiffness to be within a reasonable value. Links are usually designed with cross section either box-shaped or I-shaped to maximise their sectional moduli of elasticity. links are subject to bending under dynamic and static loads. Thus the end-effector will deviate from its expected position an amount proportional to the sum of the dynamic and static factors. The forces will vary as the robot accelerates and decelerates the payload and there will be a degree of dynamic oscillatory error. To reduce deflections caused as a result of dynamic and static loads the effective link stiffness must be increased. This may be achieved simply by increasing the cross-sectional second moment of area. Unfortunately as the cross-section of area is increased the mass of the link will increase, resulting in a loss of performance in terms of speed and increased inertia effects and possible overshoot problems. The link stiffness can be increased using materials possessing the following characteristics for use in robot operations :-

- 1) High strength to mass ratio,
- 2) High stiffness to mass ratio,
- 3) Low inertia,
- 4) High impact resistance,

## 5) Good hostile environment resistance.

Composite materials prepared appropriately possess the above properties and are ideal for robot structures. It should be noted that composite fibre material linkages will reduce beam bending but will not eliminate it entirely. Thus some positioning error due to beam bending as well as amplification of encoder measurement errors will still exist.

One way to reduce bending of the robot links is through the concept of "local support" [56]. Since robots are designed for large ranges of motion, significant bending moments are created, especially when the manipulator is fully extended. In order to reduce this large moment arm, the robot can attach itself to a local support in the vicinity of the task location. After completing the operation, the robot would detach itself and move on to the next operation where it would again attach itself to another local support. Thus, there would be several local supports in the work space, located near every critical operation. Large bending moments on robot linkages can be eliminated by attaching the robot to local supports within the working zone. There are many advantages to be gained using this idea; bending is virtually eliminated and since the position of the support will be accurately defined in relation to the workpiece the absolute position of the end point will be known. Unfortunately the major requirement for the use of robots is the flexibility they afford and the use of local supports would significantly restrict robot flexibility.

The use of composite materials and/or local support are passive solutions to the beam bending problem. A possible active solution is that of "Active Stiffness Control" for each link [33,40]. If the amount of bending was known for each link then it would be possible to compensate for it, using links comprising of an outer and inner beam with perpendicular actuators. Bending of the outer beam may be detected by strain gauges suitably placed on each beam or link of the robot. An appropriate actuator could then generate a force between the outer and inner beams, necessary to eliminate beam bending errors. The end point displacement would in effect be eliminated by a closed loop active stiffness control system.

The system described is a valuable solution but has a number of drawbacks.

firstly measuring errors and errors associated with the drive, such as backlash in gears, etc., are still present. Secondly if a fast control system is to be implemented for real time compensation control then the dynamic characteristics of each link must be well known. Thirdly there is increased complexity and thereby cost considerations. Finally the inertia of links would be significantly increased making accurate control of the end-effector difficult.

Inertial parameters of robots play a role in trajectory accuracy and velocity accuracy of robots. Since the power drives of robots, such as electric motors, do not provide instantaneously infinite power and are torque limited, inertial parameters contribute to centripetal, tangential and Coriolis forces that make robot controls challenging. Inertial parameters are usually not precisely measured in robot manipulators. Castings of structural parts could vary in thickness. Machining of casting could be offset if the starting reference dimension is offset. Heat from welding could cause slight distortion of parts. All such factors cause variations of inertial parameters from their originally designed values. Inertial parameters could vary by 10% or more from robot to robot of the same model. Usually measurements of inertial parameters are difficult and costly. Off-diagonal terms of the inertial matrix are especially difficult to obtain. Inaccurate inertial parameters can cause control errors.

With inertia  $J$  and flexibility (second moment of area  $I$  / radius of gyration  $k$ ) in a dynamic system, resonance or natural vibration is a problem to be overcome. Every machine has frequency resonances and every robot shakes at or near its natural frequencies. Typical industrial robots have their first resonance at about 20 Hz, and a large majority of industrial robots have their first harmonic at less than 5 Hz [57]. Depending on the speed and direction of drives, resonances may cause problems in accuracy and in life of components.

Friction parameters of robots are difficult to quantify accurately. In a typical non-direct drive gear train, only 50% to 90% power is delivered from the output of the motor shaft to the final driven joint: the rest is lost through friction as heat. Proper lubrication reduces friction, improves efficiency, and increases life of drive train components. Viscous friction is one of the many forms of friction to be overcome. In robot drive systems, friction is normally evident, especially in

large drives carrying a high payload. Hysteresis shows up when the drive is cycled back and forth, thus contributing errors in multi-directional repeatability and accuracy. Coulomb friction is a function of normal force exerted between two contacting surfaces. All friction parameters dissipate energy and contribute to robot inaccuracy.

### 2.3 Measurement Factors

Encoders and resolvers are the most common position feedback devices for robots. Other position feedback devices include inductosyn, LVDT, etc.. They provide resolutions from several hundred counts to several hundred thousand counts per revolution. Encoders and resolvers can be mounted directly onto the motor shaft or the joint.

Encoders mounted directly to the motor provide ease of control of the motor since motor communication is controlled directly. However, flexibility, backlash and other parametric factors affecting robot accuracy are not directly fed back to the robot controller. Thus end effector accuracy is not controlled because the absolute and point position is derived from the various joint displacements. If there is a slight measurement error associated with the encoders, it is greatly amplified through the length of each link, resulting in a large end effector position error. Encoders mounted directly to joint axes will require high resolution to detect small increments of joint positions. Compliance of drive train between motor and rotated joint can affect robot performance and accuracy, especially at the high loads.

Sensors such as strain gauge devices can be mounted at appropriate places on robot links to measure strain changes so that robot flexibility can be accurately controlled. End effector inaccuracy could be reduced by using such methods. Load sensing helps monitor machine tool conditions and can be similarly used in monitoring robot conditions [41]. It detects loads which can be used for flexibility compensations. However, all sensors are resolution-limited and if accuracy were to be improved, resolution of measurement devices would need to be better than the required accuracy.

### 2.4 Computation Factors

The desired path of a robot end effector is calculated in a robot controller. Precise commands to individual joints are then issued. This path computation takes time. In today's industrial robots it can range from several milliseconds to several hundred milliseconds. Computations of paths contain several types of error. The computer controller uses a finite number of bits to represent numbers. For most computations, 16 bits will be enough to retain precision in robot path computations. For computations that contain an accumulation of small increments of motions, there can be inaccuracy due to round-off errors.

For instance [42], a motor shaft attached to a ball screw may have a resolution of 1000 counts per revolution. This ball screw is used to rotate an arm hinged at one end. If the pitch of the ball screw is 10 mm per revolution, finite resolution of the ball screw position is then 0.01 mm per count. Since the length of this ball screw is used to control angular motion of the arm, actual resolution of the angle spanned by the ball screw is about  $(0.01 \text{ mm}/L_a)$  radian, where  $L_a$ =length of moment arm. If the length is about 250 mm, actual resolution of angle is about 0.00004 radian per count. For a total range of travel of 1.2 radian, a total of 30,000 counts is required, enough for a 16 bit representation of angles. However, actual angle computed from the ball screw travel comes from the cosine formula. From computations of square roots and arcosine, it is likely to lose one count of theta due to round-off. After several cycles of the ball screw, actual position of the robot arm can be "off" by several counts. This example is meant to illustrate the general round-off or truncation error. Increasing the number of bits to represent numbers will help to reduce or eliminate such round-off error.

Another factor contributing to inaccuracy of robots is the case of the "non-ideal" arm. Typical computations of robots use direct formulations which assume ideal robot kinematics. In reality arms are seldom built with ideal kinematics even though they are so designed. This error is in the range of manufacturing tolerance which can be in the order of a tenth of a millimetre. In order to reduce error due to a non-ideal robot arm, robot kinematic computation needs to be done using non-ideal formulations. Non-ideal formulations require much more



computation than today's robot controllers can handle. Algorithms using numerical iterative approaches to handle non-ideal robot configurations can be developed [43], but real-time general solutions are needed to overcome error due to "non-ideal" robots. Robot positions at or near robot singularities present challenges in computations. There are many ways to handle singular positions in robot manipulators. Very often, accuracy is slightly compromised near such singular positions in order to achieve stable motions.

### **2.5 Application Factors**

Besides the above-mentioned factors affecting robot accuracy, actual applications of robots also contribute to inaccuracy of the system. The main reason for demanding accuracy in the first place is the need to be able to perform the specific job or application. Besides accuracy of robots, factors related to the application need to be looked into.

The first application factor concerns installation. Some robots are installed on rather soft surfaces. Some are installed by grouting on the cement floor. It is essential that a robot to have a firm foundation.

The next factor relates to dimensional integrity of the workpiece. Due to tolerance stack-ups and other manufacturing errors of the workpiece, the demand to perform is often unduly placed on the robot. It may just need a simple correction of the workpiece design and manufacturing procedure to allow the entire robot system to perform accurately.

End-of-Arm tooling inaccuracy is another factor contributing to overall robot inaccuracy. Like manufacturing errors in robot parameters, tooling design/manufacturing/installation also contain errors. Sometimes it may be less costly and more effective to improve the accuracy of end-of-arm tooling rather than the robot.

# ***CHAPTER 3***

## **Methods of improving robot accuracy**

### **3.0 Methods of Improving Robot Accuracy**

After looking over factors affecting robot accuracy, it may seem alarming that robots can be quite inaccurate. It is generally true that inaccuracy exists, but there have been many methods developed by users, manufacturers and researchers to make improvements for intended applications. Quality inspection with particular attention to accuracy will reduce errors in manufacturing and assembly and there is a general need to provide an environment for consistent robot accuracy.

Obviously, robot users and manufacturers will constantly strive to make improvements in robot accuracy. Some of the methods which may be used for improving robot accuracy are discussed in the following three categories:

(i) Calibration methods, including many of the existing methods of calibration, with coordinate frame reference matching techniques. They generally improve robot accuracy at only one particular point in the robot workspace. They may result in an improvement termed a "first-order" improvement in robot accuracy.

(ii) Open-loop methods, including many of the robot parameter identification methods in use or in development. Many of the identification techniques deal with kinematic parameter identification. Drive train errors and compliance errors have also been investigated with good results. Open-loop methods depend on off-line measurements to identify the necessary parameter corrections. After identification of error, the corrected parameters are used by robot controllers in open-loop computation of robot paths. These methods do provide another order of improvement over calibration methods as accuracy for the robot workspace can be enhanced. Dynamic identification and control require computations that are not yet economically attainable in most industrial robots.

(iii) Closed-loop methods, including methods that incorporate external sensors that detect end-effector positions, integrated on-line with robot controllers, also provide another order of improvement: the pose of the end effector is fed back to the robot controller on-line and an additional control loop

modifies robot pose.

Besides the above mentioned compensation methods, some special end-of-arm tooling development have worked successfully in robot assembly applications. The remote centre compliance devices and force sensors can greatly improve robot accuracy, they work reasonably well in dealing with inaccuracy of robots and associated parts [44].

### **3.1 Calibration Method**

#### **3.1.1 Robot Calibration**

Robot calibration is a procedure to establish zero reference position of robot joints. It determines the relationship between encoder values and actual robot joint angles. Calibration must be done in order to command accurate robot moves in Cartesian frames. There are several methods by which robot calibration can be accomplished. As a common rule, a predetermined robot pose is selected. This pose can be established using external sensors or more frequently by physical marking on the robot structure. Some robots use hair-line scribe marks to line up the axes and some robots use micro-switches tripped by cams/dogs to determine the particular pose. After the pose is attained, a record command is issued to establish a common reference between encoder values and actual joint values.

Some robots utilise absolute encoders/resolvers as feedback of joint positions. Absolute encoders eliminate the need for users to go through the robot calibration process. It then becomes the manufacturer's responsibility to ensure the initial determination of zero reference positions.

Using scribe marks or dogs may not be consistent from robot to robot, therefore a fixture is sometimes used to establish a common position of the dogs/cams. Sometimes the fixture is used directly as part of the calibration procedure. This fixture is commonly referred to as a calibration fixture or a mastering fixture. Depending on the design of robot mechanisms, the precision mastering fixture can also determine the adjustment needed on robot lengths in addition to providing reference for the joint zero angles. Calibration/mastering procedure

helps establish accuracy at one particular position in the robot workspace: it is one of first-order correction to improve robot accuracy.

### **3.1.2 Workpiece Calibration**

There are several reference frames used in describing a robot system. In order to perform the intended task accurately, the workpiece or station frame must be accurately established. The workpiece frame can be measured using external sensors with respect to a world frame. Since robots perform tasks directly in the workpiece frame, direct transformation from robot frame to workpiece frame can be sufficient.

One common method to establish relative accuracy between a workpiece and robot is to command the robot to move to three calibrated position on the workpiece. A 4x3 transformation matrix is then calculated to transform from robot coordinate frame to the needed workpiece frame. This transformation calculation is direct and fast, even though robot parameter errors still remain uncorrected.

### **3.1.3 Sensor Frame Calibration**

In case of sensor-guided robots, the sensor reference frame needs to be established. Sensors, such as vision cameras can be mounted on the robot or at a fixed position away from the robot. The sensor frame needs to be established in order to perform accurate robot tasks. There are many methods in which sensor frames can be established with respect to the robot or with respect to the workpiece. This is one of the challenging areas in sensor-guided robot applications.

## **3.2 Open-Loop Methods**

Based on some predetermined information about the robot and factors contributing to inaccuracy, the open-loop methods make compensated commands to the robot so that robot end-effector accuracy can be improved. Such accuracy improvement is intended to work in most of the robot workspace rather than for one particular calibrated point.

Several methods have been proposed to make improvements. They are classified here as open-loop methods.

Three categories of tasks are involved in the open-loop methods, i.e., 1) measurement task, 2) modelling task, and 3) compensation task. Measurement and modelling are done off-line to identify the needed compensations. Actual compensations are done on-line while commanding robot moves.

### **3.2.1 Measurement tasks**

Robot accuracy measurement is the first step in understanding what errors need to be corrected. Two sets of measurements are needed. One set contains robot and effector poses measured by an external sensor. The other set contains the corresponding robot positions as commanded by the robot controller. With proper transformation between the two reference frames, actual errors in robot accuracy can be measured.

Automated methods of measuring robot accuracy have been widely applied. Measurements on robot accuracy required several man-months of effort in data-taking. Even though part of the time was spent in checking the integrity of data, it was generally recognised as a rather tedious task.

### **3.2.2 Modelling Tasks**

After obtaining robot accuracy data, analysis is needed to determine the type of corrective measures to be made. Several algorithms have been proposed by researchers, from purely kinematic parameter corrections [37,45,46] to full model including gear train and backlash corrections [39,43,47].

Generally, error modelling assumes small parameter errors so that correction parameters become linear in the equations. Parameters to be corrected are then calculated by using least-square fit or by minimising a global error functional. With correction of robot parameters, robot inaccuracy can be reduced 5-10 times to about 1 mm in the workspace [47]. It becomes increasingly difficult to achieve better than 1 mm since many parameters such as load-deflection models are needed. With better understanding of robot mechanisms and more parameters identified, accuracy improvement can

theoretically be made more easily. Small robots can be made more accurate by using a good set of identified parameters, but it is extremely difficult to achieve anything better than 0.5 mm accuracy [47].

### 3.2.3 Compensation Tasks

There is a limit to how many parameter compensations the robot controller can actually handle. From a knowledge of the mechanical design of the robot, critical parameters to be corrected can be isolated out of a set of general parameters, thus reducing the identification task and compensation task.

In open-loop methods, the corrected parameters are stored in the robot controller. Robot motions are then commanded based on the corrected parameters without additional feedback of robot accuracy. Compensation tasks can be easily done with ideal robot configurations. If there were parameters that would render the robot non-ideal, computations would become increasingly complex. Parameters such as offsets may require computations of robot jacobians as part of path planning. In most industrial robots today, it is still not economically practical to compute jacobians on-line. In the near future when computation power of robot controller improves, this may not be a problem.

### 3.3 Closed-loop methods

Using parameter identification and then compensating robot parameters does not provide exact positioning information of the robot. As discussed in the previous section, robot parameters are identified using a "best fit" approach to the data taken at various locations. There is still error left at each measured location and likely more so at locations elsewhere in the workspace. The next improvement to achieve positioning accuracy in the overall robot workspace is to use a feedback method where robot positions are constantly monitored using an external sensor[48]. The robot end-effector pose is controlled closed-loop with the externally fed-back position.

If the absolute position of the end effector is measured, the accuracy of a robot would theoretically be limited only by the measurement device and/or actuator servo resolution. Thus, if the position of the end effector is fed back to the

controller, the various actuators can react accordingly to eliminate the error. Besides reducing static positioning errors dramatically, closed loop control about the end-point also provides greater application flexibility since the information obtained from the endpoint can be used for active stiffness control. The desired end effector pose can be maintained in the presence of external loads, appearing to be an "infinitely" stiff system, or extremely compliant to permit mating part assembly.

The end-effector is sometimes shielded from the view of the base receiver because of structures in the workspace, creating more problems. Nevertheless, with the proper resources or application flexibility compromises, it appears that this problem can be overcome.

Even with closed loop control utilising a pose measurement device, there is still another problem. In most commercial robots the actuators are positioned in series and correction of the end-effector error would still require movement of the whole robot. Thus there could still be cumulative errors associated with each link in the robot structure including measuring errors. Thus each actuator must provide high speed and good response for large motion, while at the same time providing very accurate positioning for fine motion.



# ***CHAPTER 4***

**Design of Force  
Sensor**

**&**

**Construction of  
Geometric Path**

#### 4.0 The Survey of Force Sensors Mounted on Wrist

There have been numerous designs of force sensor for many different applications. The

earlier ones go back to 1977 designed by Nitzan et al, built at SRI[54]. The SRI sensor, as well as many other force sensors, uses strain gauges cemented to elastic beams. Strain gauges were used because of their low cost, high resolution, and reliability. The elastic beams in the SRI sensor were designed to minimise hysteresis and to separate the force and torque components. As shown in Fig. 4.1, one end of each

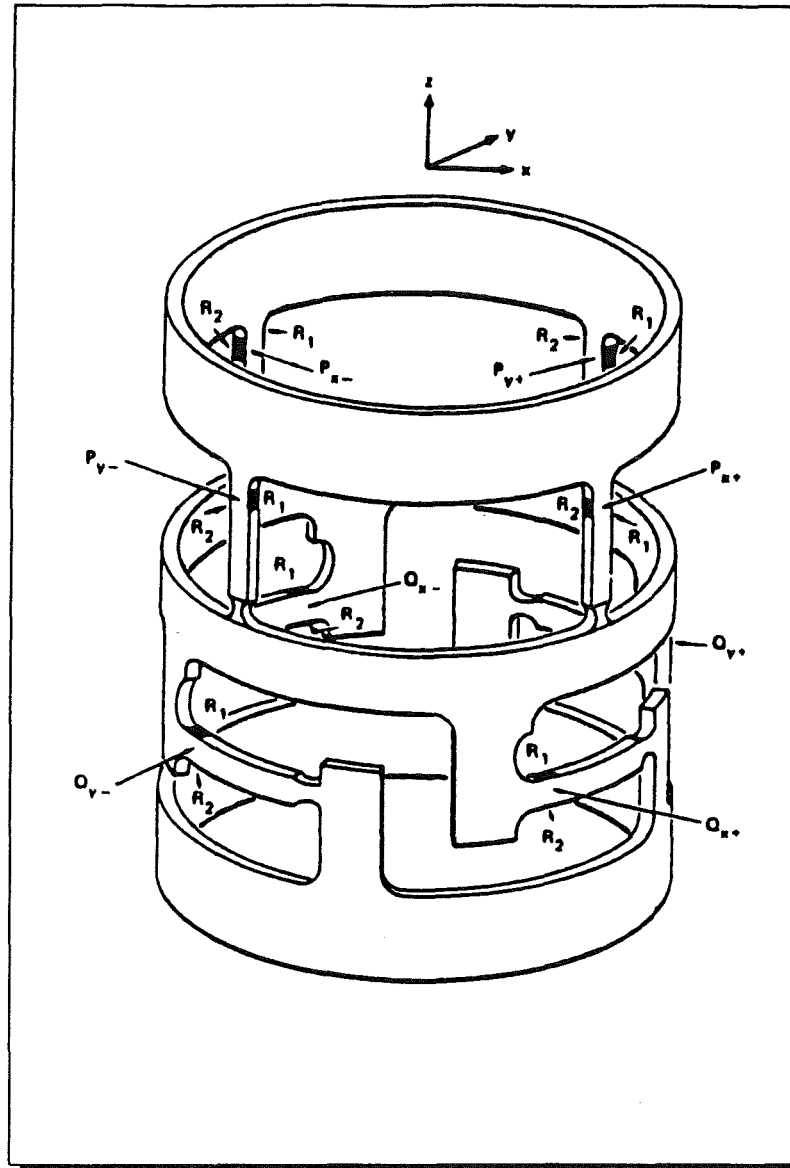


Figure 4.1, Six-components wrist sensor based on strain gauge elements. (SRI International)

beam has a neck profile. This neck transmits negligible bending moment and thus increases the strain at the end at which the gauges are located.

Although the design is very light, it is bulky and thus the distribution of mass produces large inertia and undesirable moments when it is used in conjunction with a micro-manipulator at the wrist.

A three component force sensor was designed for the PUMA-600 robot in the

Robotics Laboratory of the Technion in Israel [58], as shown in Figure 4.2.

The sensor is 70mm in length, 51mm in diameter and weighs 200g. It consists of four parts:

thin walled cylinder(A), inner protective cylinder(B), outer protective cylinder (C), and torsional protection pin (D).

Eight strain gauges are mounted on the thin walled region of (A).

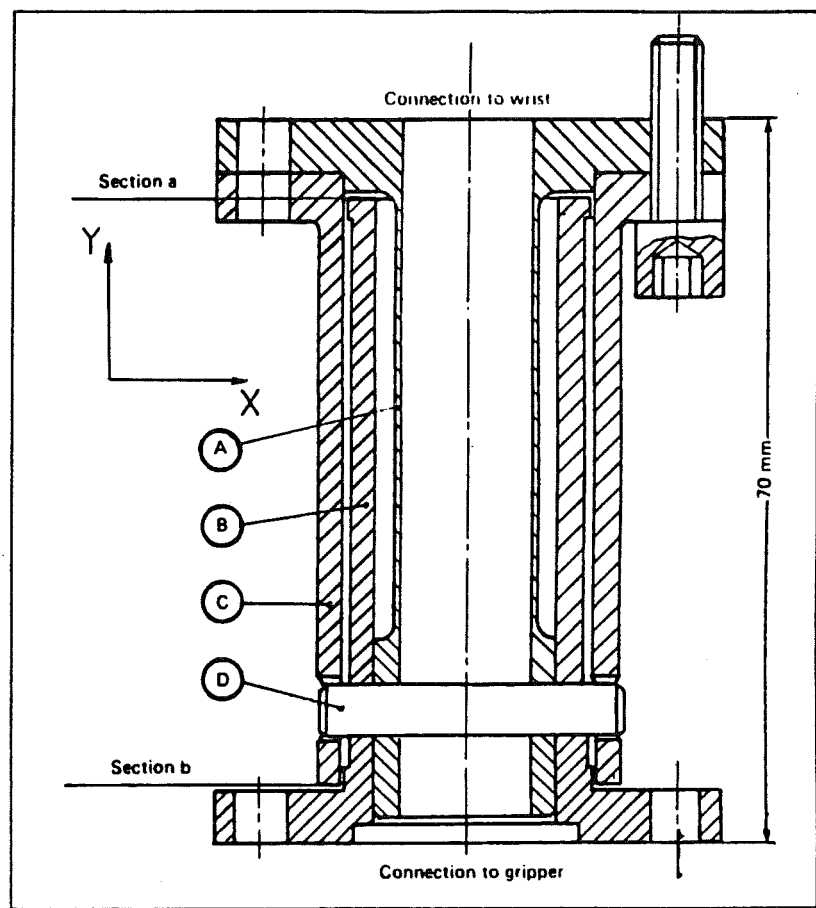


Figure 4.2, Structure of the Technion's three component force sensor.

The main feature of this sensor is its overload protection. This is an important factor in the design of a force sensor which has been ignored in many other designs. The thin walled cylinder is protected from overloads in the following manner.

Force acting on the centre of the gripper in the X or Y direction causes an internal force and moment at section b (Figure 4.2).

This will cause a deflection, and a deflecting angle, at section b. For any force exceeding 50 N, these deflections will cause contact between the protective cylinders (B) and (C) which will then bear any load above 50 N to 1000 N.

The limitations of a three component force sensor is its inability to perform so as to facilitate insertion operations in any direction.

H. Van Brussel et al [36] designed two different types of force sensor.

1- The sensor consists of two rigid rings connected by four flexural strips. The construction is monolithic, obtained by machining a solid block of special Al-alloy, to avoid the non-linearities normally present in bolted joint or glued interfaces. This is illustrated in Figure 4.3.

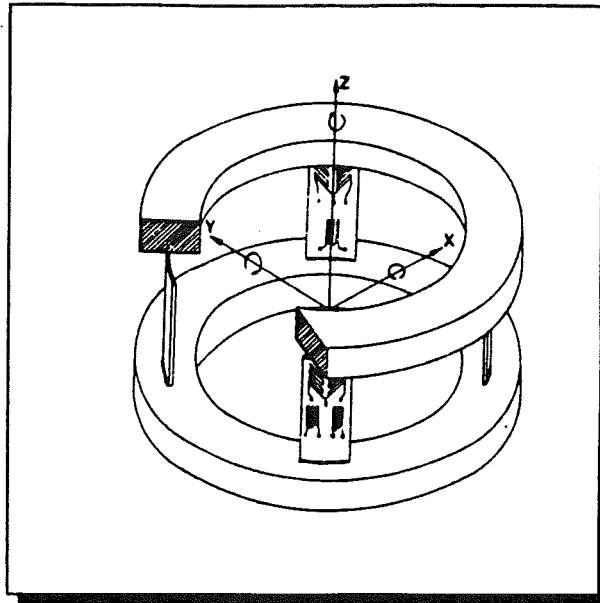


Figure 4.3, Six component force-torque wrist sensor for deburring application

This large diameter (170 mm), small height (60 mm) configuration gives rise to a high resistance to bending torque (except for the Z-axis) but it has a very good sensitivity to lateral forces. The total sensor mass is 2 kg, which is too heavy to be used in assembly or path following operation. Even if it is scaled down to a reasonable weight and dimension, it would still have a low sensitivity of the Z-component due to the fact that compression strains are produced to represent forces in Z-direction.

2- In this design the sensor consists of a central square block and four cantilevered radial spokes with square cross section. The outer end of the spokes are freely supported by means of thin flexures as shown in Figure 4.4.

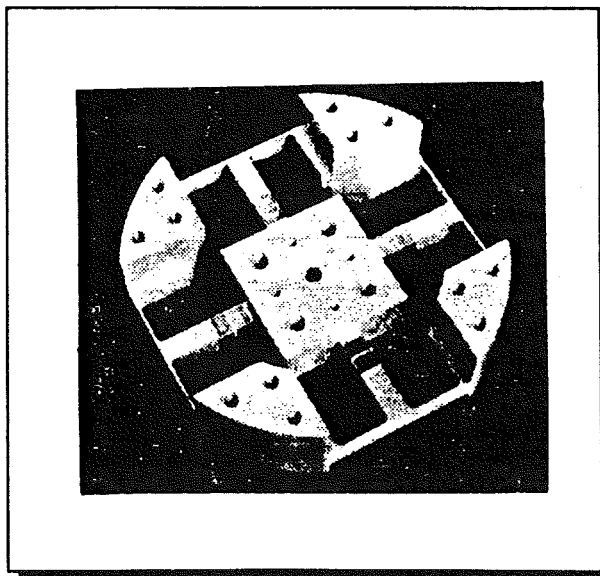


Figure 4.4, Improved design of six component force-torque wrist sensor

The force sensor designed for this work at Middlesex polytechnic has similar features to the above force sensor. Unlike the above design, however, the outer ends of the spokes are fixed and the central part is free to move. This gives it much more flexibility to

simulate a ball hinge behaviour (see Figure 4.10 & 4.20).

A sensor with similar features was also developed at UWIST[10]

This sensor comprised a flat plate in the shape of a cross shown in Figure 4.5.

A strain gauge bridge was mounted on each arm of the cross. The outputs of the strain gauges are processed by a computer which

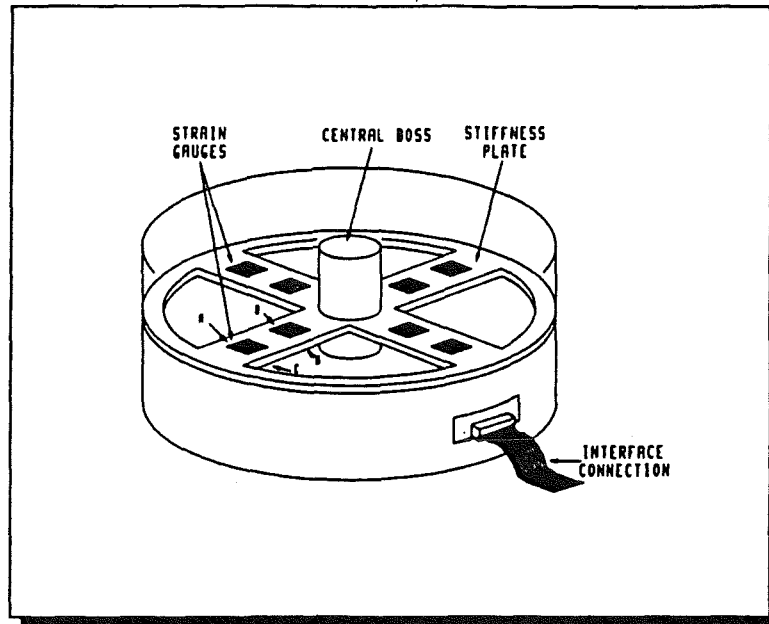


Figure 4.5, Strain gauge sensor developed at UWIST.

monitors the assembly. The X and Y outputs are evaluated from the values of the opposite strain gauge bridges. The Z-component is obtained by averaging the four values. The major shortfall of this design is the lack of any torque measurement and also the lack of an overload protection device. A high quality connector should be used to transmit the signals.

At the MIT Draper Laboratory, a sensing wrist with three connecting columns is used Figure 4.6.

Three extensional strain gauges and three strain gauge shear bridges are cemented on the connecting columns, which provide six differential signals for computing the forces and moments.

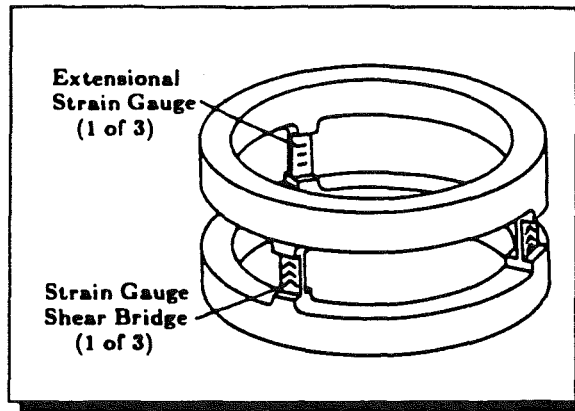


Figure 4.6, Draper Laboratory force sensing wrist

The design is very similar to that of Van Brussel et al, described earlier, and thus also suffers from the low sensitivity of the Z-component.

#### 4.1 Separation of Forces and Moments with Strain Gauges

Figure 4.7, shows a structural member which is subjected to both vertical and horizontal loads.

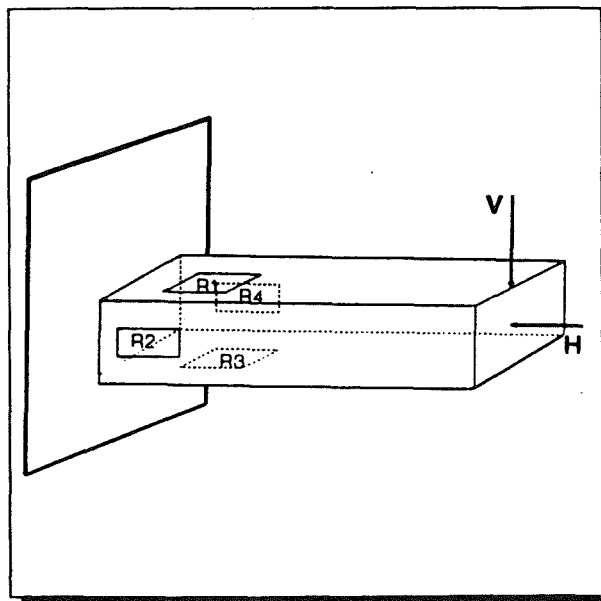


Figure 4.7, Cantilever beam instrumented with strain gauges for sensitivity to horizontal forces and insensitivity to vertical forces

It will be noticed that (Figure 4.7) strain gauges R1 and R3 are mounted on the top and bottom of the beam, and located at their usual positions in the Wheatstone bridge circuit. With this arrangement the electrical outputs of the gauges are additive. If the member is subjected to a bending moment, the electrical output of the gauges cancel because of the difference in signs of their strains.

A horizontal or axial load results in equal strains of like sign and a net electrical output from the bridge circuit. Gauges R2 and R4 must either be used as inactive dummies on a separate unstrained piece of metal or be mounted on the member in the poisson arrangement.

From this principle, when vertical and horizontal forces are applied on the four sided beam, these forces can be separated by appropriate wiring and connection of strain gauges to form a Wheatstone bridge. Figure 4.8, shows part of the sensing element in the form of a beam which is subjected to vertical and horizontal forces. These forces cause the element to bend.

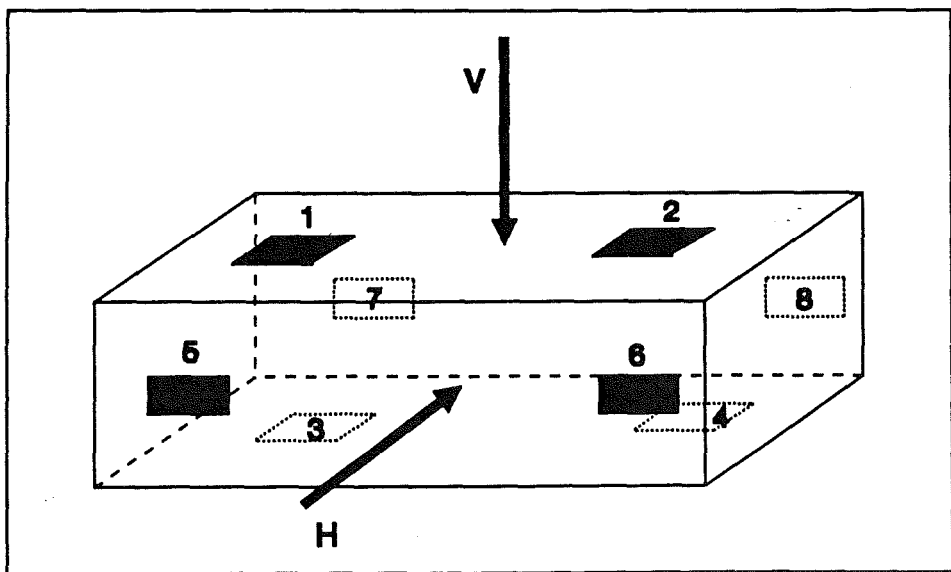
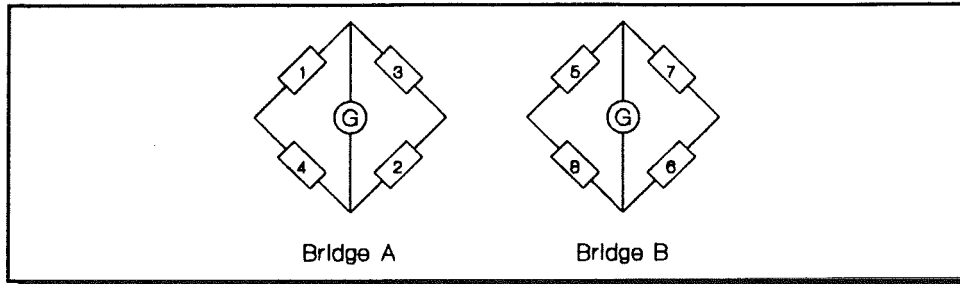


Figure 4.8, Sensing element subjected to vertical and horizontal forces

Due to bending caused by the vertical force only, the strain gauges 1 and 2 are in compression and 3 and 4 at the bottom are in tension. As a result the bridge made up of these strain gauges is insensitive to the vertical force. To the

contrary the gauges 5, 6 and 7,8 on either side of the beam are all in tension and the electrical output from Wheatstone bridge made up of 5,6,7,8 indicates the vertical force. The principle is similar when horizontal forces are applied to the bar. In this case bridge A made up of gauges 1,2,3,4 gives a significant output which indicates horizontal force while bridge B shows virtually zero response, Figure 4.9.



**Figure 4.9, Bridges A & B sensitive to vertical & horizontal forces respectively**



## 4.2 Design of the Force Sensor

Three force sensors were made during the course of this research, all with similar features.

The original one shown in plate 1, Figure 4.11, was made by machining the sensing element shown in plate 2, Figure 4.12. The sensing element consists of 16 strain gauges bounded to the four spokes of the wheel. This force sensor was capable of measuring force and torque in six degrees of freedom by eight Wheatstone bridges.

This force sensor which was designed specifically for the purpose of this research was also then wrongly used in another project when measuring the forces of a cutter in machining plastics. It worked perfectly well for a time but, unfortunately, it then failed from a mechanical force overload. While being used for analysing the pattern of forces in cutting a thin slice of Aluminium it started vibrating violently (as evidenced the marks shown on the protective plate in plate 3, Figure 4.13) and the central part fractured under the excessive dynamic load, as shown in plate 4, Figure 4.14.

Soon after this incident another force sensor was made. However, this time it was considered adequate to limit the number of bridges to four recognising the practical limitations in hardware, in data acquisition and signal processing. A four channel Unilab A/D was used to digitise the analogue electrical signals from the strain gauges with a BBC micro-computer. The hardware is shown in plate 5, Figure 4.15. Unlike the original eight bridge sensor, this four bridge sensor was capable of measuring forces in only two directions.

In this design it was decided to use connectors (see plate 6, Figure 4.16) in order to ease the mounting and dismounting of the force sensor on the robot whenever required. The idea was however, found to be unsuitable in practice as the connectors were responsible for much noise into the system. There was also a problem of loss of signal due to the cracking and breaking of the solder at the connecting point under dynamic loading.

As a result the third force sensor, Figure 4.20, Plate number 7, was made in

a manner very similar to the original one but with some further improvements. This time the sensing element, Figure 4.12, consists of 32 strain gauges bonded to the four spokes of a wheel. Each spoke acts as a deflection bar, Figure 4.10. The use of double strain gauges improved the sensitivity of the signal output from the gauges: enlarged drawings of the force sensor are in Appendix {1}<sup>2</sup>.

A separate research paper was also published [53] which covers the design features and its application in path following.

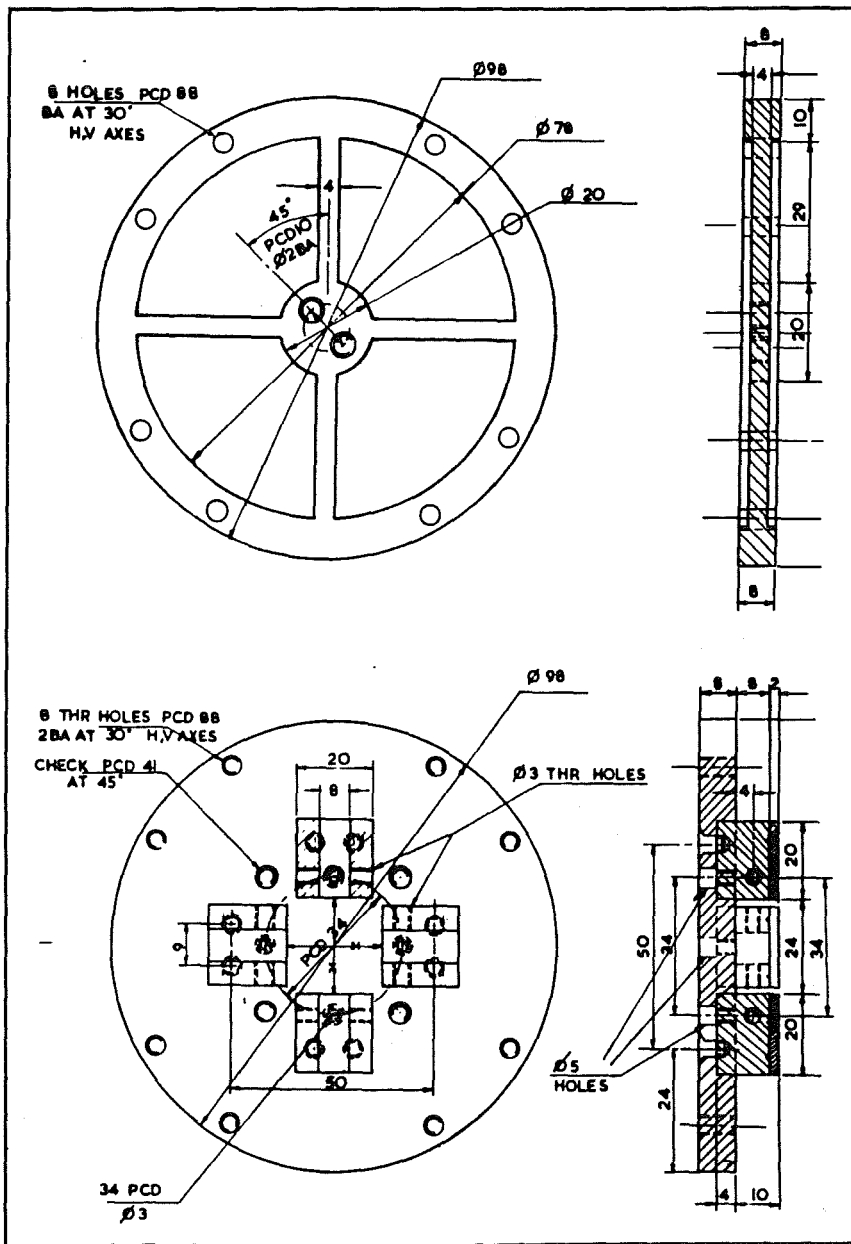


Figure 4.10, Sensing element and overload protection device

<sup>2</sup> The sign {..} refer to a specific appendix in the appendix section, throughout the thesis.

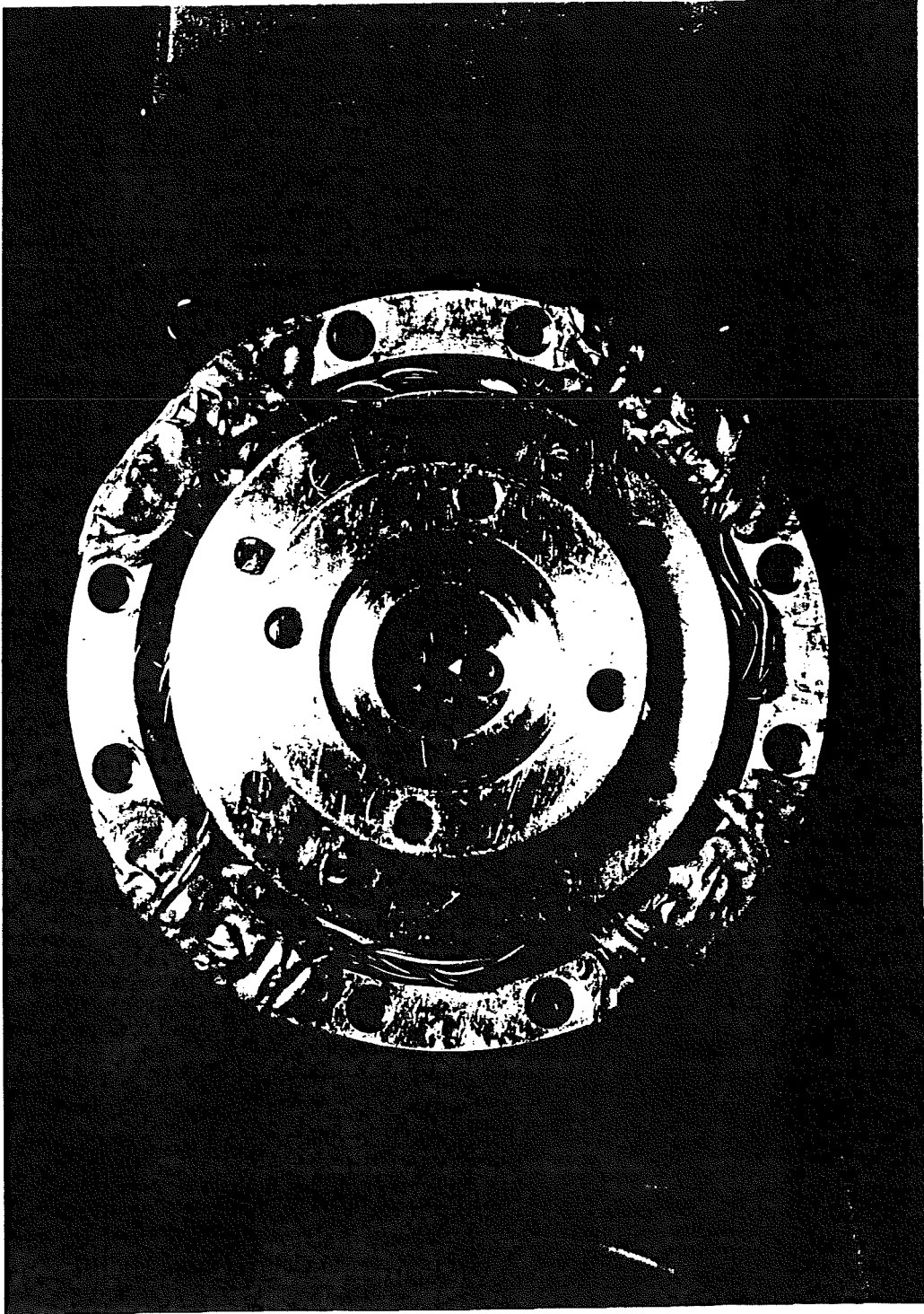
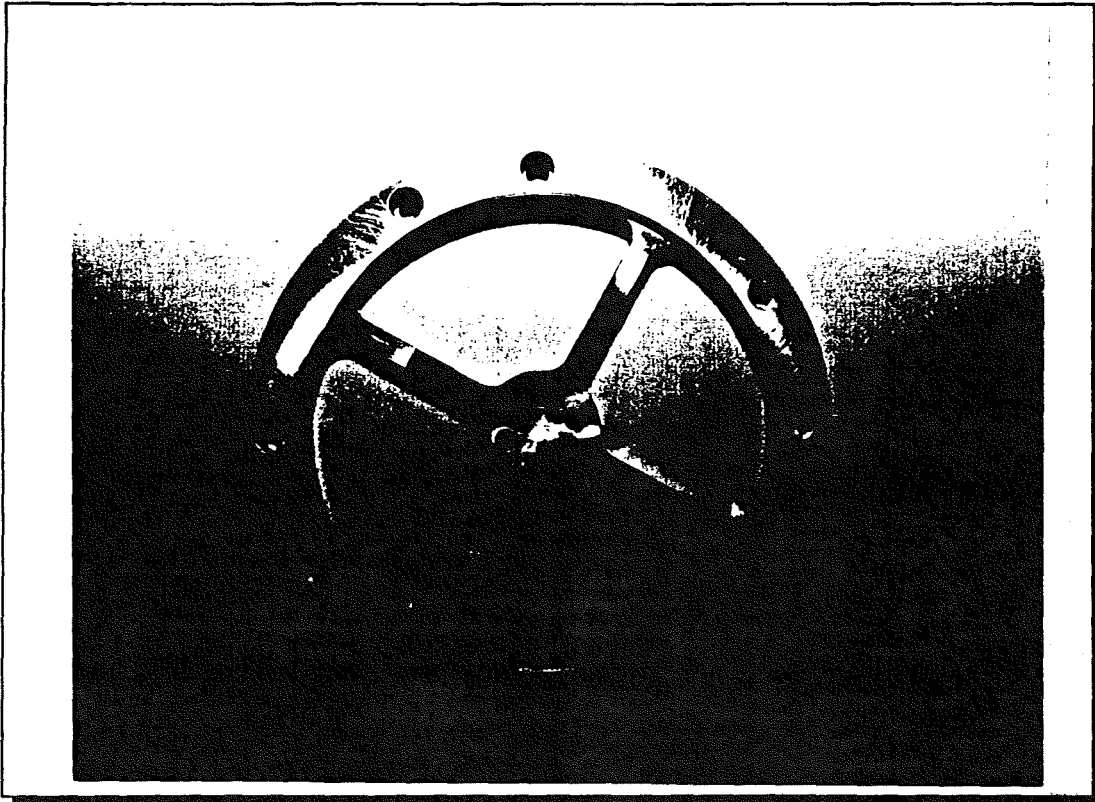


Figure 4.11, Plate number 1: Force sensor with overload protection device



**Figure 4.12, Plate number 2: The monolithic sensing element (machined on Wadkin CNC Milling Machine)**

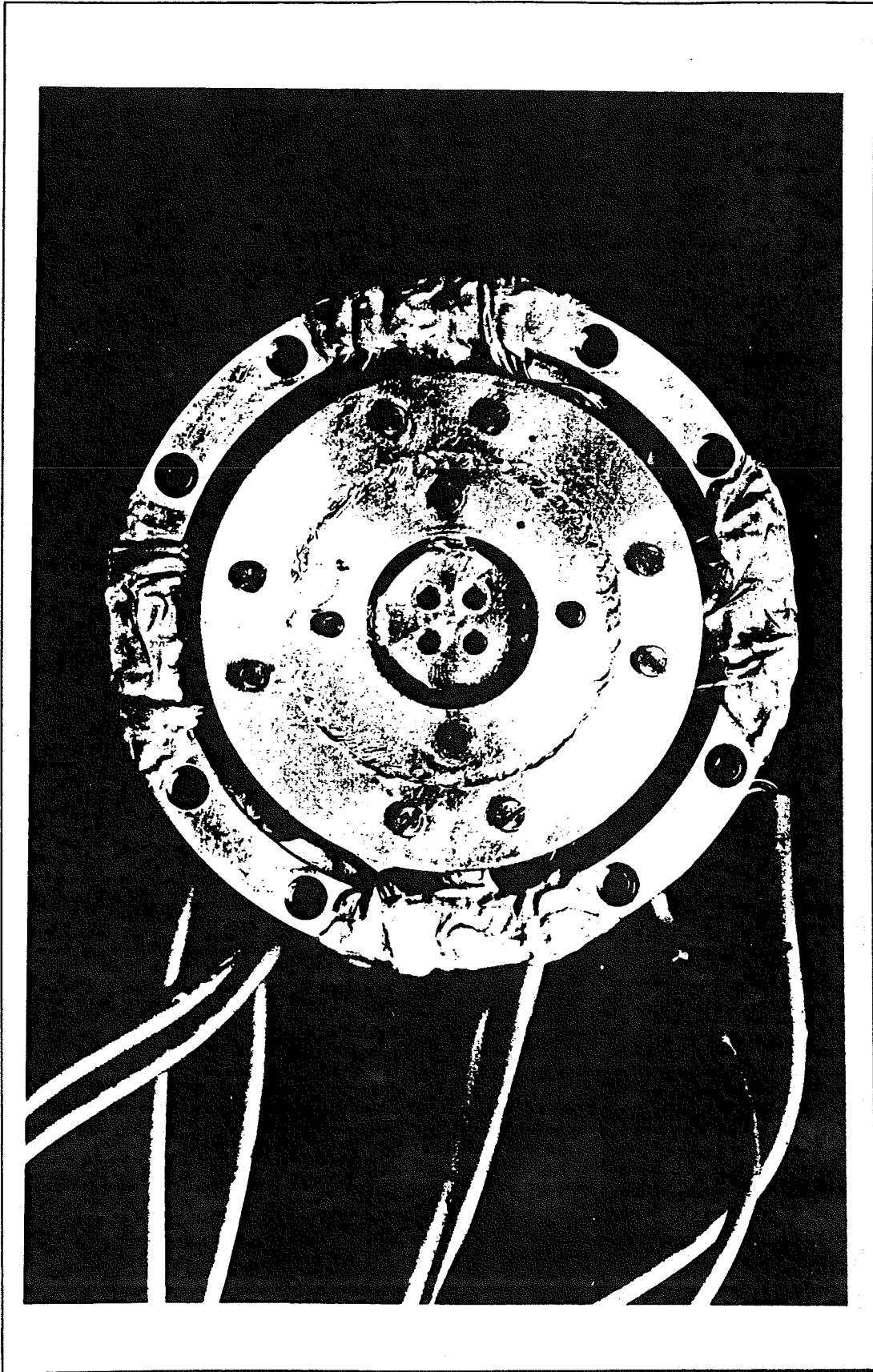


Figure 4.13, Plate number 3: Marks of mechanical failure resulted in measuring forces of a cutter in machining Aluminium alloy

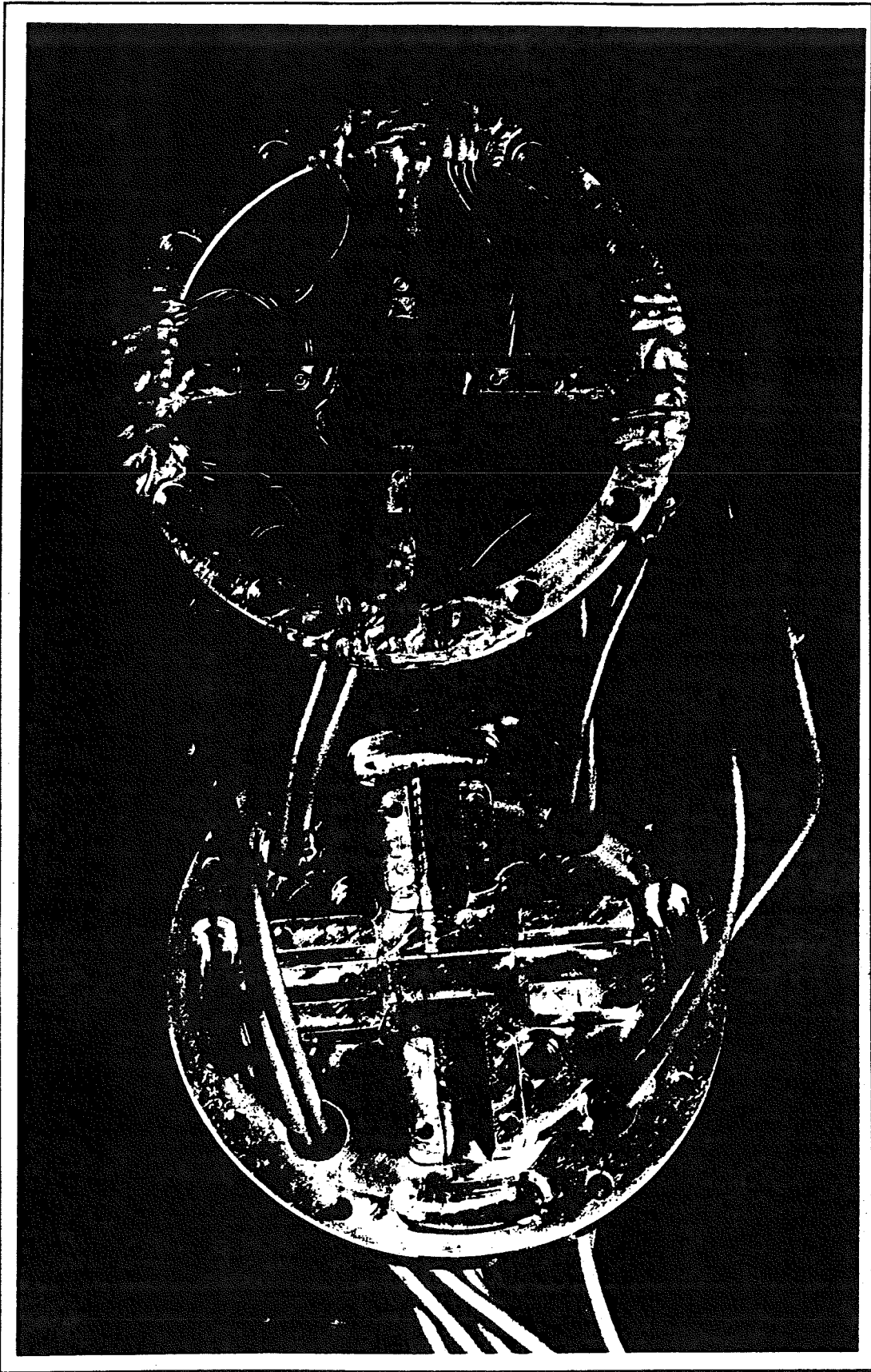


Figure 4.14, Plate number 4: The dismantled force sensor with fractured central part under excessive dynamic load

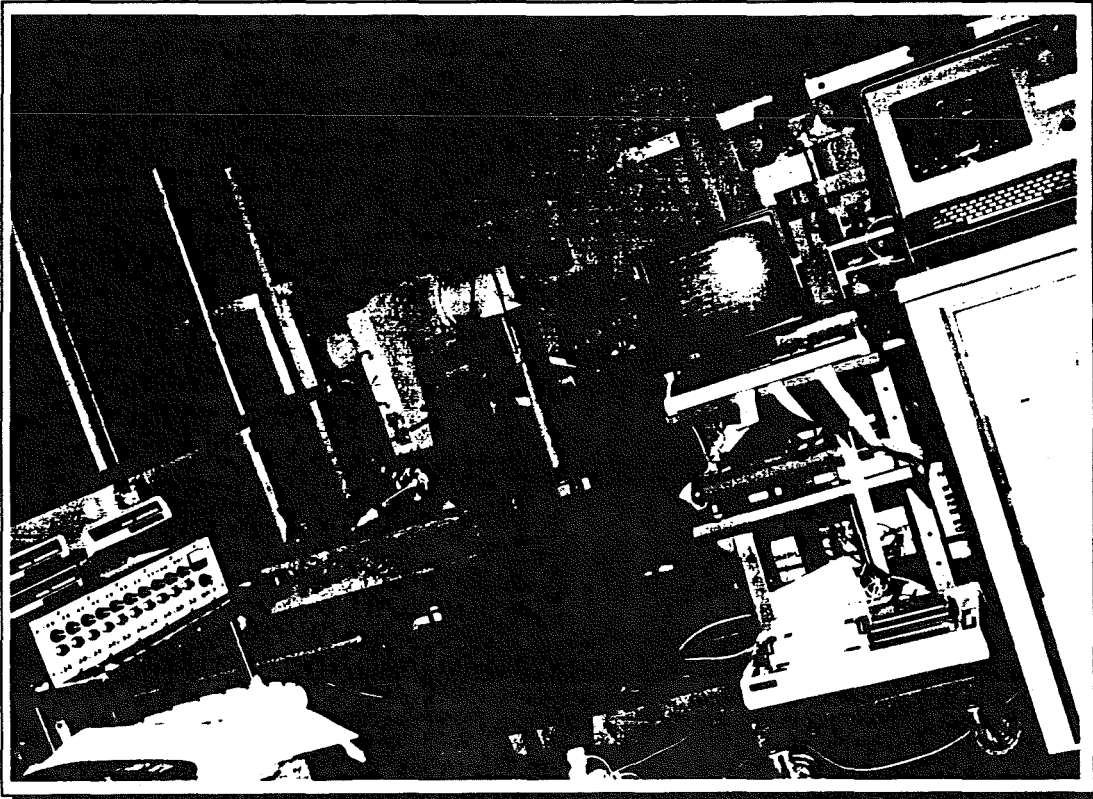


Figure 4.15, Plate number 5: The earlier hardware system used in data acquisition and signal processing

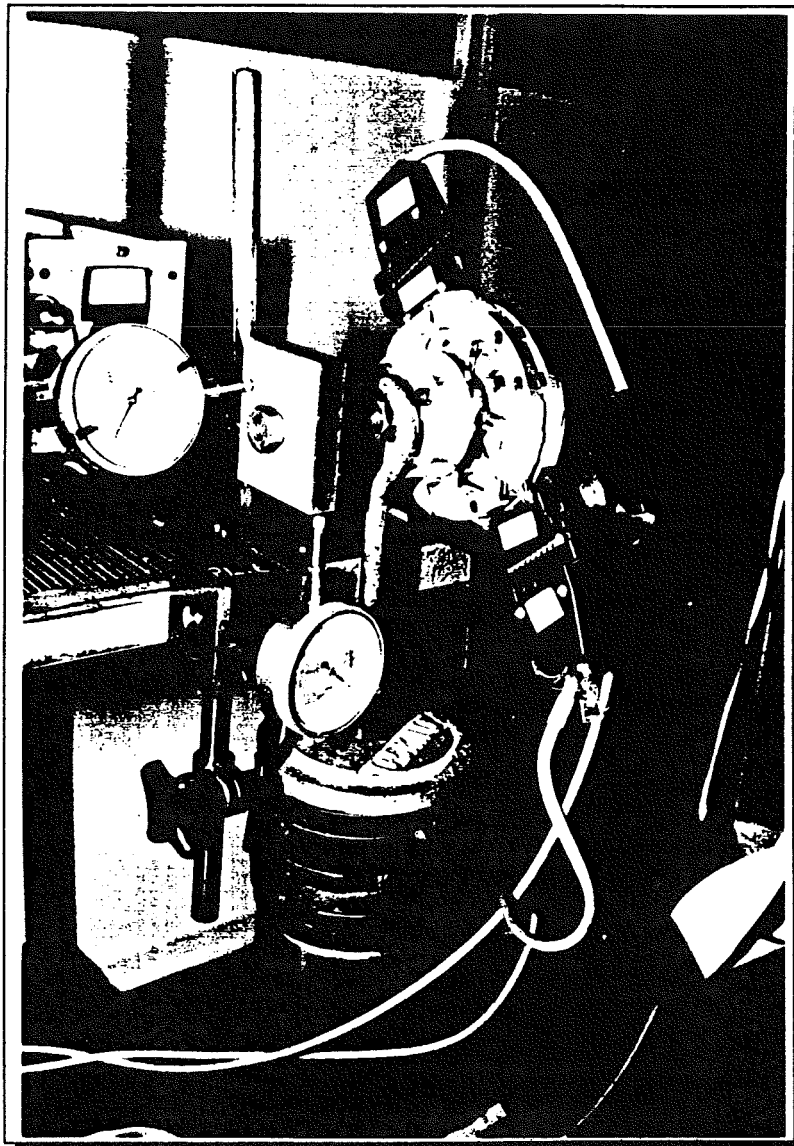


Figure 4.16, Plate number 6: The second force sensor designed with new features is shown in calibration stage. The dial gauges were used to measure deflections in two dimensions under static loading.



### 4.2.1 Manufacture of the Sensing Element

The sensor wheel was made by first turning a solid disc to the required shape. Once in this stage, the four centre segments were cut out using the Wadkin CNC Milling Machine, this left the four spokes of the wheel forming the cross, Figure4.12, plate number 2.

The positions of the central holes, and the eight outer holes were also centre drilled on the Wadkin machine. Once centre drilled, all the holes were drilled and tapped as 2BA.

The housing plate was also turned on a lathe to a blank disc, and then slots were cut for the positioning of the overload protection devices using a normal milling machine. The overload protection device were machined on a milling machine as one length of bar, then it was cut into four separate pieces.

Details of the CNC programme for use on the Wadkin Milling Machine are given in appendix {1}.

To relieve any stresses built up in the material due to machining, the sensor was heat treated for stress relief. This involved heating it to 250°C and then cooling very slowly in the oven. Once the stress relieving was completed, the sensor was ready for the strain gauges to be fitted.

The procedure for attaching the strain gauges to the sensing element is described in detail by the Welwyn Strain Measurements Bulletins [59]. This company acted as one of the collaborating establishment to support this research programme. The installation was carried out and supervised according to instructions issued by the company. The type of gauge which was selected for the force sensor was EA06060 PB 350 with extended fatigue life recommended for dynamic measurements [60].

In the design of sensing element, the outer rim is fixed and the central part is free to move. This gives it much more flexibility in such a way as to simulate a ball hinge behaviour. The gauges are bonded to the deflection bars near the outer rim: there is a double gauge on each side of the four deflection bars, resulting in a total of 32 gauges.

Two pairs of two gauges on opposite sides of a deflection bar form a full bridge as shown in Figures 4.17 & 4.18.

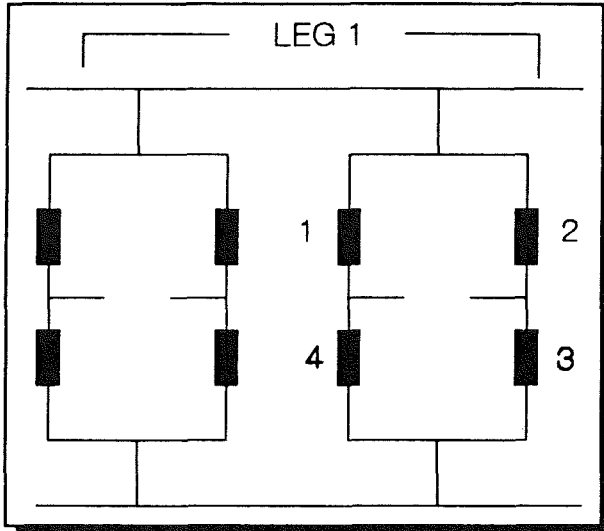


Figure 4.17, Circuit diagram for strain gauge wiring on a deflection bar

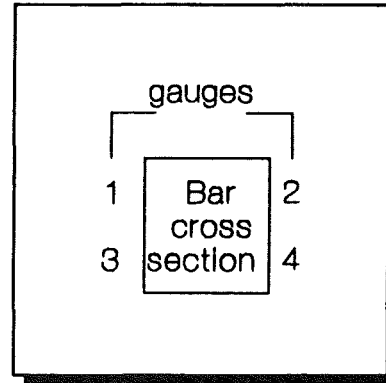


Figure 4.18, Bar cross section

Note that, from Figures 4.17 & 4.18, the gauges which are placed diagonally opposite in the bridge (i.e. the bridge pairs 1,3 and 2,4) are on the opposite sides of a deflection bar. Each full bridge provides a single reading which reflects the differences in the strain levels on opposite sides of a bar. This arrangement of strain gauges reduces the thermal drift problem since all sides of deflection bar will contract or expand by approximately the same amount for a given temperature change. However the strain gauges selected were those suitable for good dynamic performance and of temperature compensated quality.

The full sensor provides eight output readings B1,.....,B8, each representing a force in the direction shown in Figure 4.19.

Using simple force and torque balance considerations, the eight measurement components can be resolved into three orthogonal force and torque components referenced to a sensor based coordinate frame, Figure 4.19, through a 6 by 8 matrix.

$$\begin{bmatrix} F_x \\ F_y \\ F_z \\ T_x \\ T_y \\ T_z \end{bmatrix} = \begin{bmatrix} 0 & K_{12} & 0 & 0 & 0 & 0 & K_{16} & 0 & 0 \\ 0 & 0 & 0 & K_{24} & 0 & 0 & 0 & 0 & K_{28} \\ K_{31} & 0 & K_{33} & 0 & K_{35} & 0 & K_{37} & 0 & 0 \\ K_{41} & 0 & 0 & 0 & K_{45} & 0 & 0 & 0 & 0 \\ 0 & 0 & K_{53} & 0 & 0 & 0 & K_{57} & 0 & 0 \\ 0 & K_{62} & 0 & K_{64} & 0 & K_{66} & 0 & K_{68} & 0 \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ B_3 \\ B_4 \\ B_5 \\ B_6 \\ B_7 \\ B_8 \end{bmatrix}$$

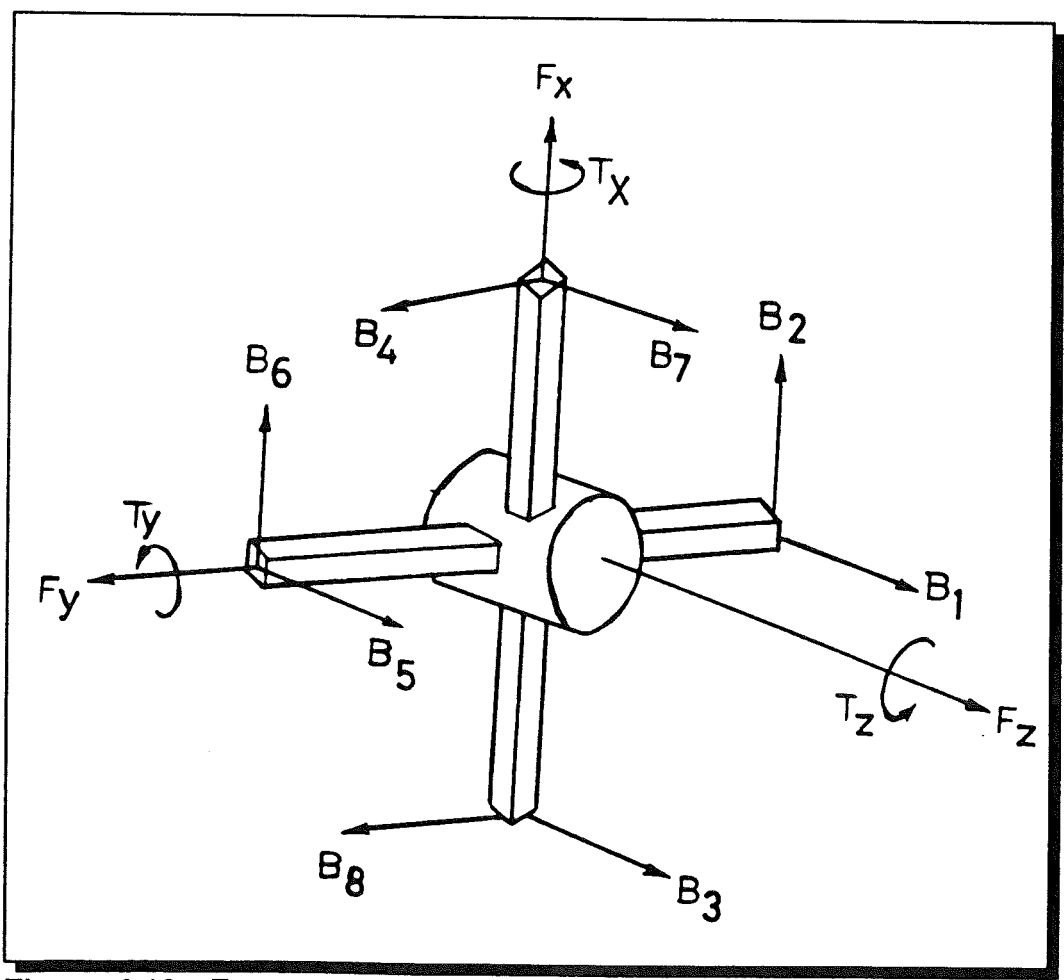
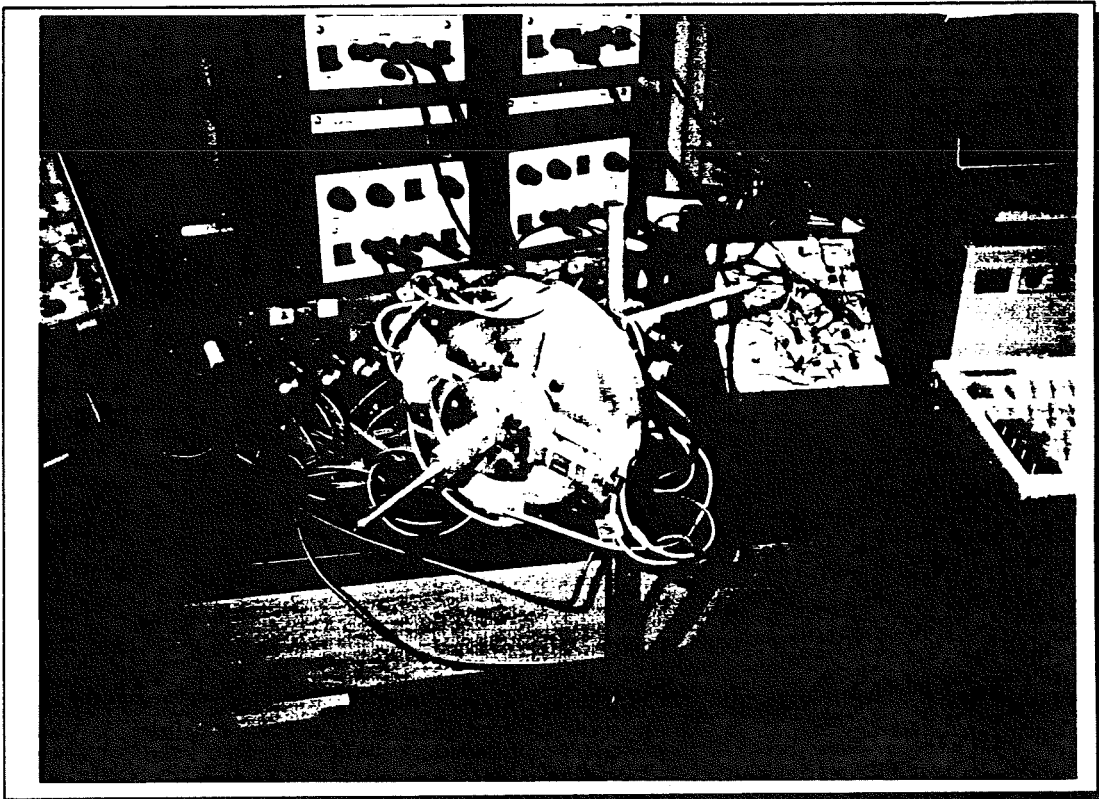


Figure 4.19, Force sensor reference frame



**Figure 4.20, Plate number 7:**The third force sensor designed for measuring force & torque in three orthogonal directions. The two torque arms with notches at specified interval from the centre facilitates the testing of force-torque sensor.

following. The deviation of output voltages in loading and unloading due to the effect of hysteresis is also shown: probably due to the high quality of the selected strain gauges and also of the great care taken in mounting them hysteresis effects were not very significant. The effect of hysteresis due to loading and unloading is better shown with the aid of bar charts, as in appendix {2}.

The linear graphs were plotted by loading the force sensor to maximum load of 15 kgf and torque of 8 N-m. This data was gathered over 10 repetitions of loading, and the average voltage reading were then plotted against load. The gradient of each individual channel was calculated and multiplied by  $g=9.8$  N/kg for three orthogonal forces. In the case of torque, the variable factor of arm's length was also taken into account. The following analysis shows the typical formulae which were used in calculating K values.

Force related values of  $K=\text{gradient of line} \times g$

Torque related values of  $K=\text{gradient of line} \times g \times \text{arm's length}$

(where arm's length is 0.065 m)

$$\begin{bmatrix} F_x \\ F_y \\ F_z \\ T_x \\ T_y \\ T_z \end{bmatrix} = \begin{bmatrix} 0 & -0.148 & 0 & 0 & 0 & 0.151 & 0 & 0 \\ 0 & 0 & 0 & -0.160 & 0 & 0 & 0 & 0.147 \\ 0.294 & 0 & 0.281 & 0 & 0.255 & 0 & 0.257 & 0 \\ 0.00950 & 0 & 0 & -0.0087 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.0094 & 0 & 0 & 0 & -0.0088 & 0 \\ 0 & 0.023 & 0 & 0.0157 & 0 & 0.015 & 0 & 0.014 \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ B_3 \\ B_4 \\ B_5 \\ B_6 \\ B_7 \\ B_8 \end{bmatrix}$$

The sample and hold operation is performed by a monolithic integrated circuit.

Continuous voltages from the 8 bridges mounted on the force sensor is sampled in order of millisecond intervals to obtain new sets of values of  $B_1$  to  $B_8$ , for continuous force and torque calculation.

The signs (+/-) relates to the direction of the force/torque applied to the sensor. These directions can be either taken from the reference sensor frame, Figure 4.19, or from the LEDs mounted on the front panel of each channel of strain gauge conditioner and amplifier.

#### 4.4 Construction of A Geometric Path for the Robot to Follow

To study the nature of contact forces in the path following, some geometrical panels were made. Figure 4.22, shows the front view of this path and its dimensions. Figure 4.23, plate 8, shows the picture of this geometric path. The red marks on the path are the positions recorded by the PUMA Robot with different spacing. To implement the force control technique it was essential to detect the forces and the deflections exerted on the end effector as the robot travelled between the points. This information about the pattern of forces in following the path was used to write the software with an efficient strategy to correct for tool setting errors and dynamic position errors.

Normally when the robot is instructed to move from one point to the next, the path would be a straight line between the two points. But here, since the panel is circular, there is an arc between any two points. The probe moves along the arc experiencing forces as it tries to travel to the next point in a straight line, as illustrated in Figure 4.21. It was found that greater the height of the arc greater the force of contact between the probe and path. In some cases forces of up to 78 N were experienced by the force sensor.

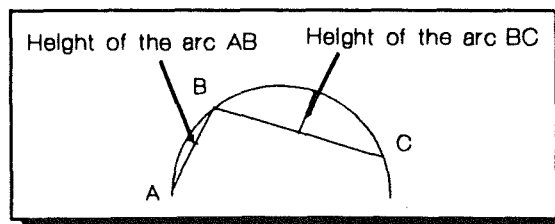


Figure 4.21, Three adjacent points on a curved path

The robot was programmed to move along the complete panel through 53 points, (these points are marked by red colour as seen on Figure 4.23), while the micro-manipulator conducts the end-effector along the path, keeping the contact force within a limit specified in the software.

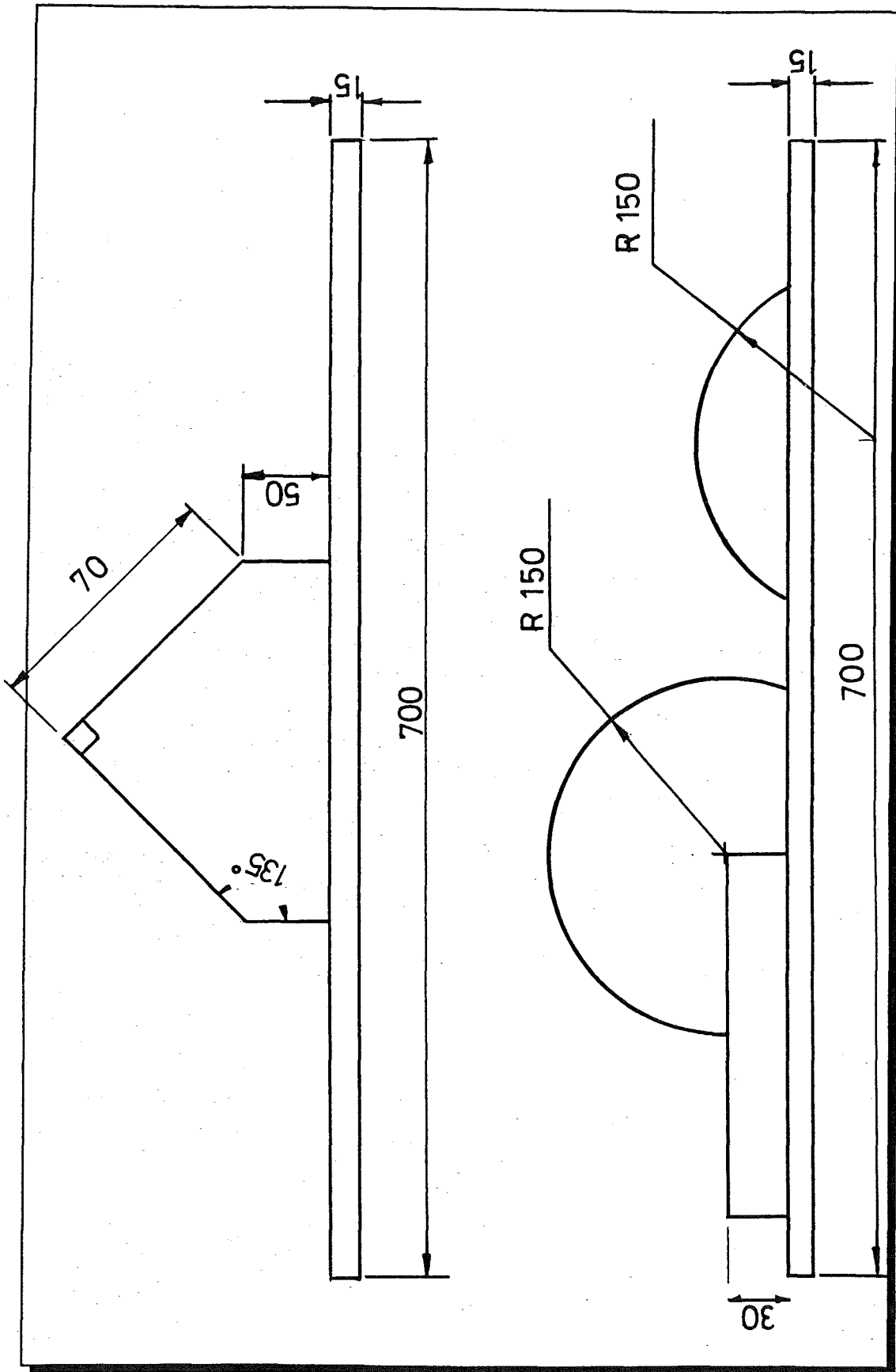


Figure 4.22, Front view of geometric path

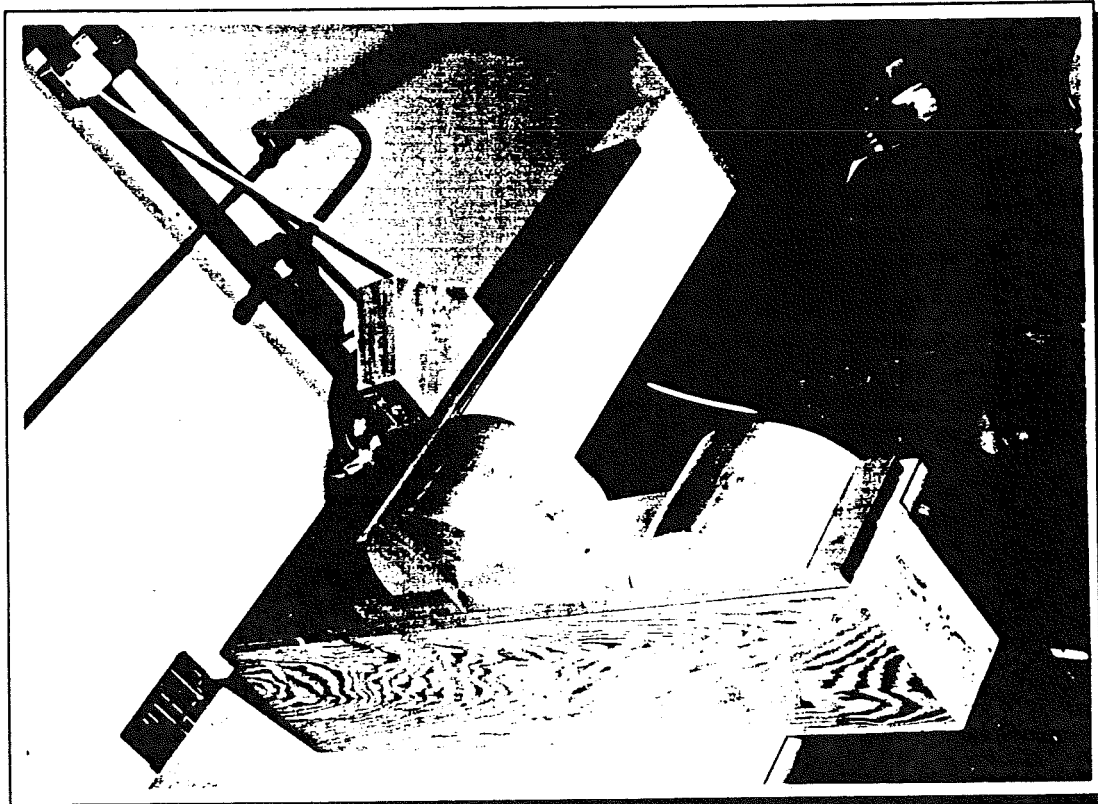


Figure 4.23, Plate number 8: The designed rig for the robot to follow different geometric paths using force control technique



# ***CHAPTER 5***

## **Design of Micro Manipulator**

### 5.0 Development of A Micro-Manipulation Device

Industrial robots are generally subject to limitations in dynamic performance. In summary, an industrial robot

- has a large inertia which results in slow acceleration and deceleration,
- has limited communication facilities,
- has relatively poor positional resolution,
- does not allow sufficient control over individual axes,

The above limits can be improved in various ways. For instance, a basic redesign of robot structures, using light and very stiff composite materials, could contribute to improvements in dynamic behaviour. Also development such as VAL II, by Staubli/Unimation, would contribute to ease of communication with Puma Unimate industrial robots. The major benefits of VAL II over VAL I:-

- (i) Structured programming Background/2nd task program functions such as PC operation.
- (ii) Enhanced communication, such as real time path control.
- (iii) Program compatibility with the many VAL II machines in existence.

Problems in using the basic robot to effect fine motion force feedback have been noted by other workers in the field, notably Van Brussel[17], who proposes closing the feedback loop via an end-effector at the robot wrist with particular application assembly. This end-effector gives the robot several additional degrees of freedom, high resolution, and controllable high speed movement. This sort of arrangement generally requires a less complicated robot, which takes care of the coarse pre-programmed positioning. The small, high speed movements under force feedback are then performed by the micro-manipulator. The micro-manipulator is an attachment to the macro-manipulator wrist end and has the benefit of accurate position control as shown in Figure 5.1. The use of such a device has none of the disadvantages attributed earlier to the basic robot.

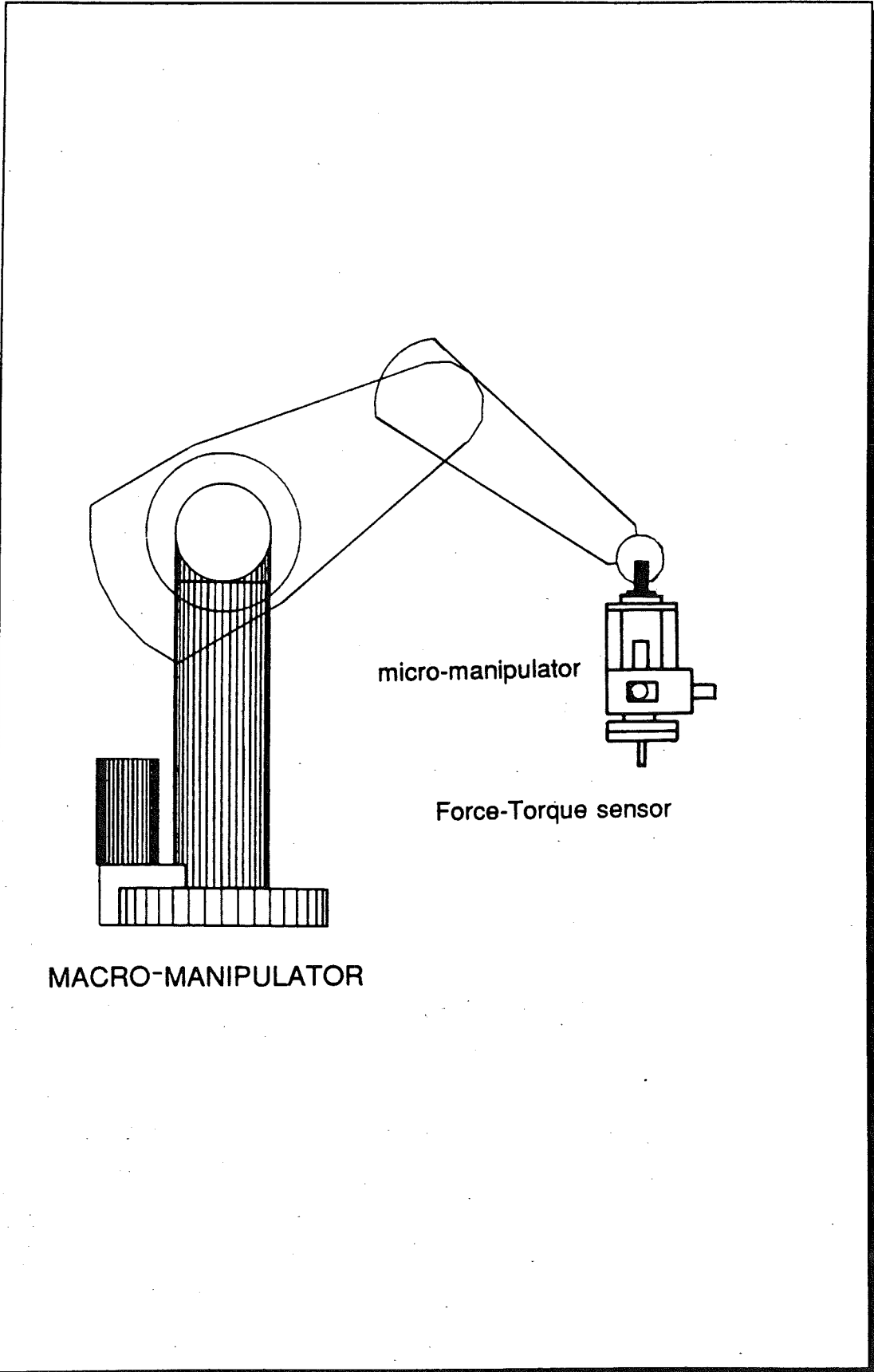


Figure 5.1, Macro/Micro manipulator

**5.1 Determination of PUMA 560 Robot Accuracy and Repeatability**

Prior to design of the micro-manipulator, it was necessary to measure the accuracy and repeatability of the PUMA 560 robot. Since this robot will form the basis of testing of the micro-manipulator in situ.

The measurement of positioning errors in terms of accuracy and repeatability were achieved using Hewlett Packard laser interferometer system type HP5526A.

The laser interferometer was set up in accordance with the application notes manual [61]. Before the test could be performed it was necessary to compensate for variance in the velocity of light associated with variances in humidity, air pressure and temperature. The compensation is achieved by calculating or reading off from CIBS Psychrometric chart the humidity once the wet and dry bulb temperatures of the air have been determined. The air dry bulb temperature, air pressure and humidity are entered into the Hewlett Packard computer and a compensation factor is calculated which is then entered manually to the laser system. Since air temperature, pressure and therefore humidity vary over time it was important to conduct the test over the shortest possible time. Ideally an automatic compensator would have been used but this was not available on the system. The compensation data at the time of the experiment were measured as follows:

Wet-bulb Temperature	18°C
Dry-bulb Temperature	21°C
Air Pressure	742mm Hg
Humidity	46%

These tests were performed only on linear positioning of the robot and not on the three possible rotations of yaw, pitch and roll.

The graphical representation of the results from the Hewlett Packard measurement system for two identical test runs are shown in Appendix{3}. The

accuracy was found to be approximately  $\pm 3.3$  mm with a repeatability of approximately 0.6 mm.

### 5.2 The Fundamental Design Criteria

In order to establish a successful design certain fundamental criteria were established which would have to be satisfied by any respective design. The detailed design and manufacture have been carried out based upon the following criteria:

#### 1- High accuracy and repeatability

High accuracy and repeatability are the fundamental requirements for a micro-manipulator.

#### 2- Simplicity of Design and Manufacture

If a design is to have commercial application it must be simple to manufacture and to assemble. Additionally a simple design should aid maintenance and increase product reliability.

#### 3- Low Cost

For the design to be economically viable it is essential for it to be relatively low cost compared with any competing device.

#### 4- Low Mass & High Stiffness to Inertia Ratio

It is important that the addition of a micro-manipulator to the macro-robot should not significantly affect the payload capacity. In order to achieve this and maintain a high degree of accuracy, the design must be of low mass but possess a high degree of stiffness.

#### 5- Feedback Control

In order to get true positional accuracy, real time positional feedback of the end-effector is a fundamental requirement.

#### 6- Multi-axis

In macro-robot the wrist motions can be controlled quite accurately since there is no cumulative measuring error and the effective length of a link is comparatively small at wrist. It would be possible to utilise macro-manipulator actuators within the wrist in conjunction with micro-manipulator actuators. However this would require a real time processor language such as VAL II which is currently not available on the Puma

560. Since both the micro-manipulator and the macro-manipulator have to be controlled separately in order to get true flexibility and good accuracy, the micro-manipulator would require six degrees of freedom ideally.

#### 7- Selective Compliance

For true robot flexibility in assembly operations selective compliance is required: a degree of dexterity would, in effect, be introduced. This would assist in assembly operations such as the fitting of a peg into a hole. Because compliance introduces some inaccuracies, assembly operations requiring high degrees of accuracy necessitate the elimination of compliance so that a "selective compliance" system is required.

#### 8- Adequate Range

A reasonable stroke length of 2 cm was considered here so that corrective operations can take place within adequate range.

### 5.3 Design Considerations

Following the above design criteria, some initial ideas based upon a Cartesian framework were put forward. The first was a Gantry Crane idea. This was rejected because the bulk size as well as the bulk mass was considered to be too great. A novel idea, that of intersecting telescopic tubing, was then considered (Figure 5.2).

This idea was borne out of the necessity for a high stiffness to inertia ratio: aluminium tubing could thus be specified with a minimum wall thickness. The strength, and also the stiffness, would primarily be the result of a design with closed circular sections. An important design consideration was the need for extremely good accuracy. The "tube" design was seen to achieve these accuracies without the need to specify the very close tolerances which would have increased the cost of the device. This was achieved by having large bearing surfaces so that the angle error resulting from "play" is significantly reduced.

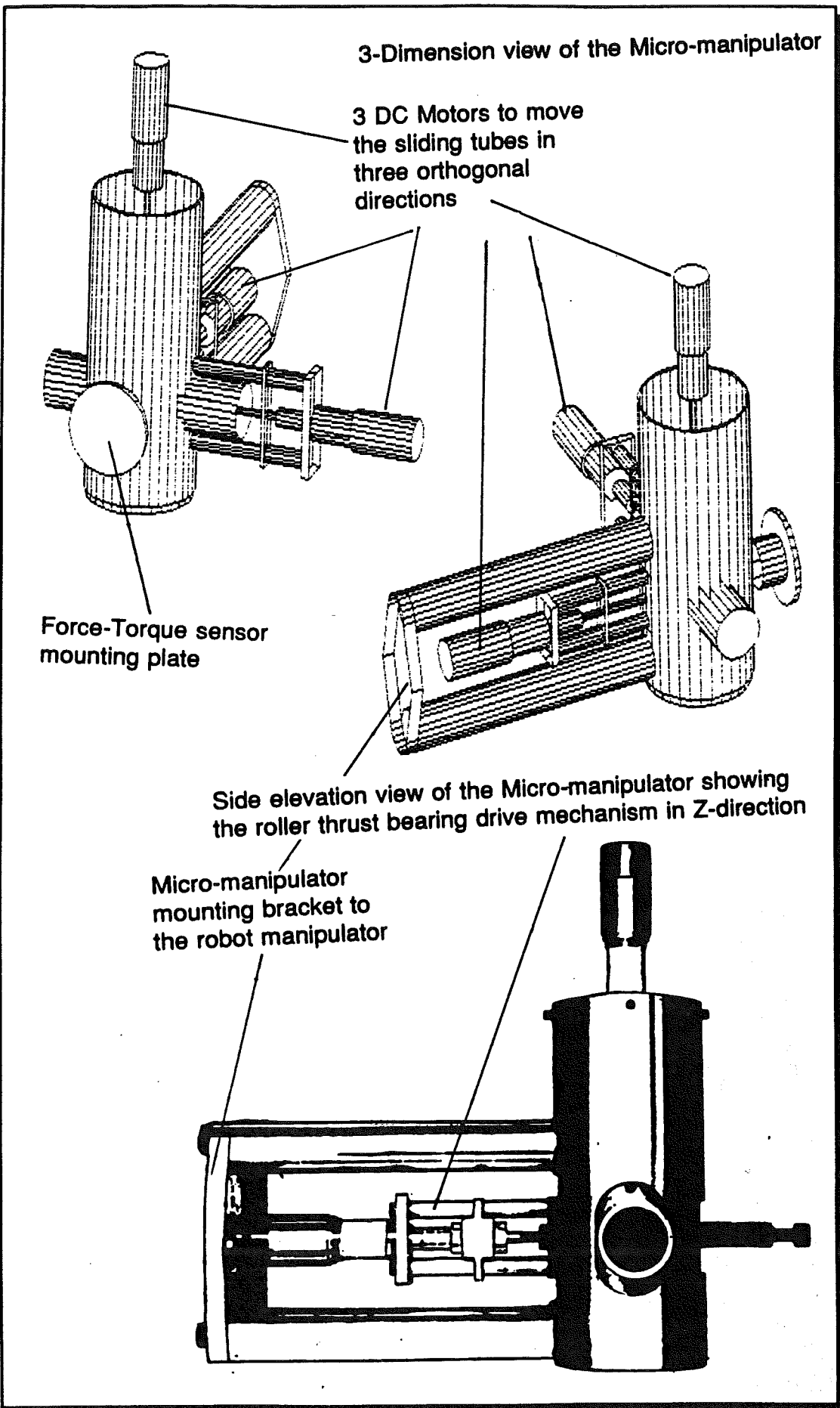


Figure 5.2, Sliding Tube Micro-manipulator

In order to achieve the required 20 mm range, a minimum central support tube diameter of 75 mm had to be specified. Lay out drawings as shown in Appendix {3} were used to determine these minimum tube diameter requirements. The tubes used in the prototype manufacture were subject to material availability and not, therefore, the ideal. The specific design issues will be dealt with separately for each sub-assembly as shown in Figure 5.3.

The detailed prototype manufacture and prototype assembly instructions are fully described in Appendix {3}.

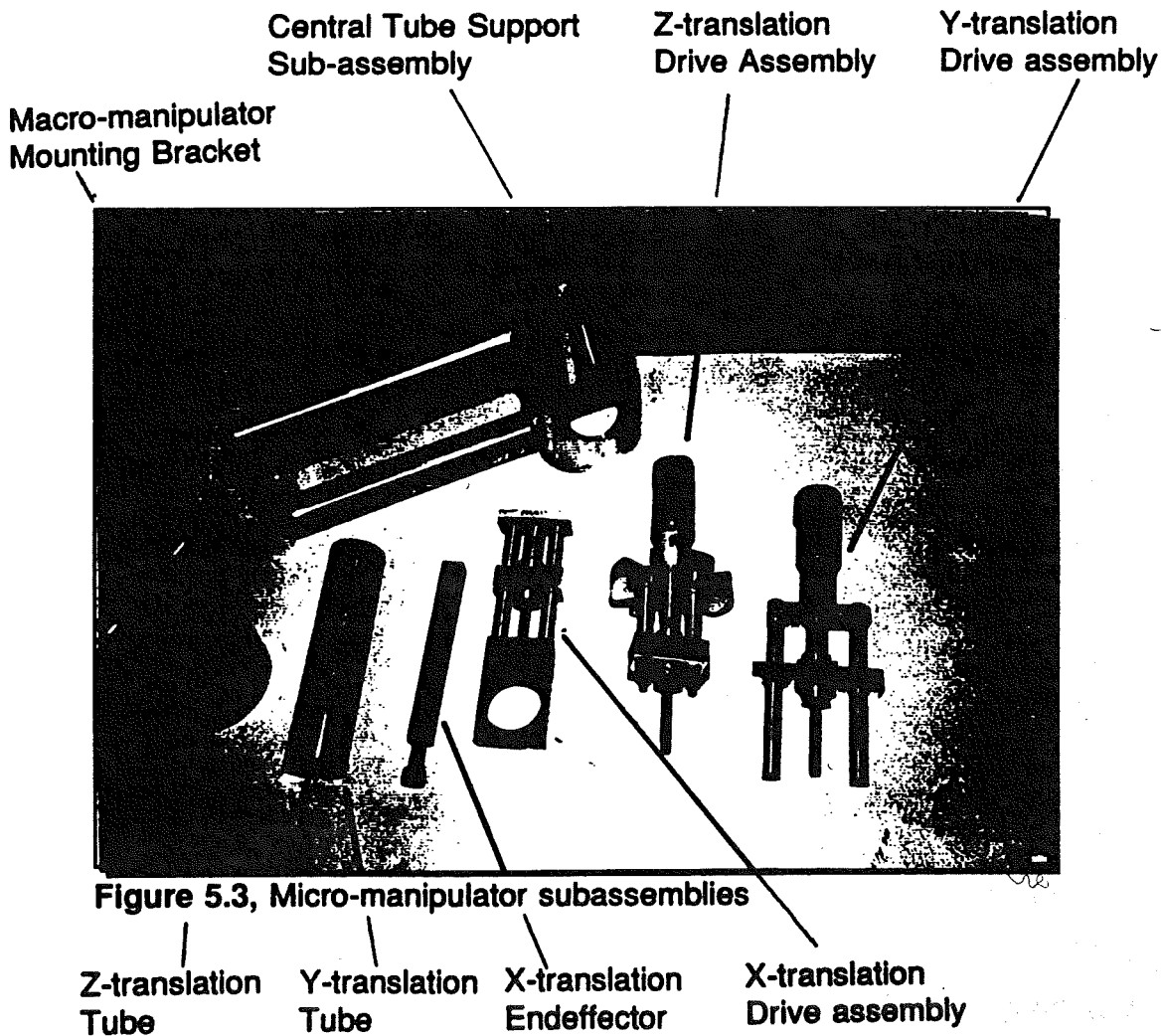


Figure 5.3, Micro-manipulator subassemblies



### 5.3.1 Central support tube sub-assembly

The specific drawing detail is shown in Appendix {3}. The central support tube sub-assembly is attached to the macro-manipulator by a mounting plate and two mounting arms. These mounting arms are fixed to the central tube via four screws, two screws per mounting arm. The mounting arms are themselves tubes to reduced overall weight but, in order to get a good location to these arms, mild steel inserts were located in the ends of the mounting arms by means of interference fits. Holes could thus be drilled and tapped for the mounting screws, both to the central mounting tube and to the macro-manipulator. Two screws were specified for each mounting arm in order to spread the shear loading when the micro-manipulator mass is rotated by the macro-manipulator and thus restrict any tendencies for rotational errors due to poor fixing of the mounting arms. Additionally, bending loads are also reduced in each screw.

To assist in the location of the central tube, the ends of the mounting arms are curved. The actual fixing of the arms is achieved through access holes in the front of the mounting tube as shown in Figure 5.4. These holes enable the clearance holes for the screws to be countersunk. Since this central tube internal surface will form a bearing surface the screws had to be countersunk below the internal surface of the tube.

The elongated slots enable the intersecting tubes to slide for Z, X and Y translations. Finally the mounting bracket itself was profiled to reduce weight still further.

### 5.3.2 Z-Translation Sliding Tube

The sliding tube is illustrated in Figure 5.4. An important design consideration was for the drive mechanism to pull the sliding tubes from the centre of the tube. This was important because any binding due to a couple being produced as a result of off-axis loads, had to be eliminated. It was also important for adjustment of the location of the threaded hole through which the drive screw would locate. Binding will result if there are any small misalignments, increasing the torque requirement for the motor, leading to a reduced motor mechanism

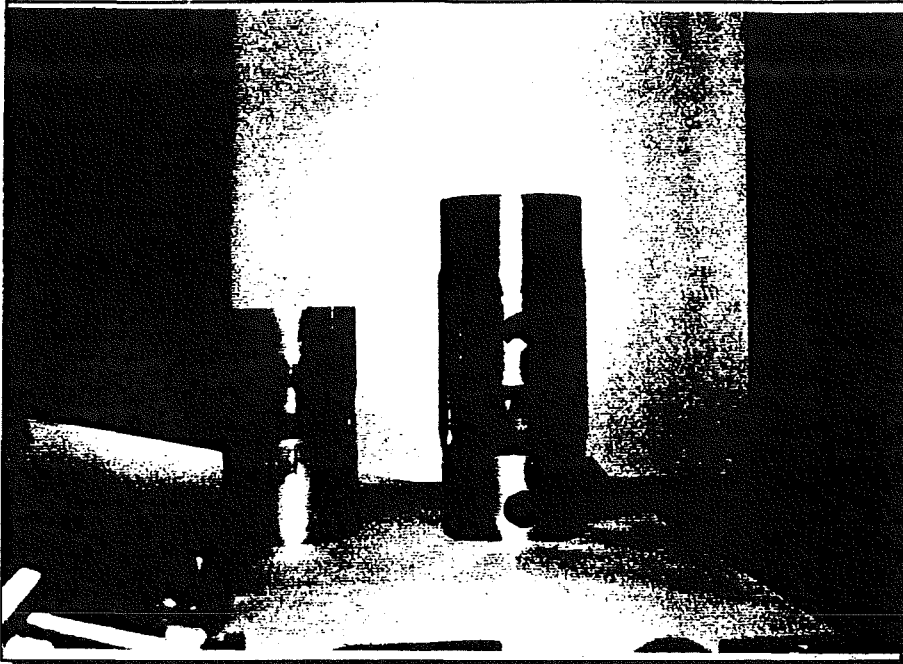


Figure 5.4, Front elevation of the central support tube and the Z-translation tube.

life. To achieve this adjustment the threaded bar was located within the tube via two thrust collars which can be fixed to the bar by the grub screws.

The horizontal elongated slot shown in Figure 5.4 enables the end-effector to move horizontally in the Y-direction and forward in the X-direction. But the vertical movement of the end-effector in the Z-direction can only be achieved by movement of the tube itself sliding within the central tube. The circular holes passing through the tube enable the Y-translation tube to slide only in the Y-direction.

### 5.3.3 Y-Translation Sliding Tube

Similar to the Z-translation tube a threaded bar is located centrally via two collars (see Figure 5.3). The tube itself has a square hole passing through it. This enables the X-translation end-effector to slide through it but restricts the tendency to rotate. The end-effector can only slide in the X-direction and translations in the Y-direction can only be achieved via the Y-translation sliding tube sliding through the Z-translation tube which is supported within the central tube.

### 5.3.4 X-Translation End-effector

This was a simple square bar made of mild steel, see Figure 5.3, to prevent rotation of the end-effector in the Y-translation tube. This design eliminated the need for keyways. There are weight penalties here but the weight was reduced by slotting the bar. Due to the requirement for a threaded hole steel was preferred over aluminium as thread wear would have been a problem producing greater backlash errors.

### 5.3.5 Z-Translation Drive Mechanism

Referring to Figure 5.5, the Z-translation drive mechanism consisted of a top bracket which was profiled to remove material so as to reduce weight. The thickness in fact was found to be more than that required and there is the possibility here for further weight reductions. The bracket has a central clearance hole to suit the motor face and spindle, together with appropriate holes for fixing of the motor to the bracket.

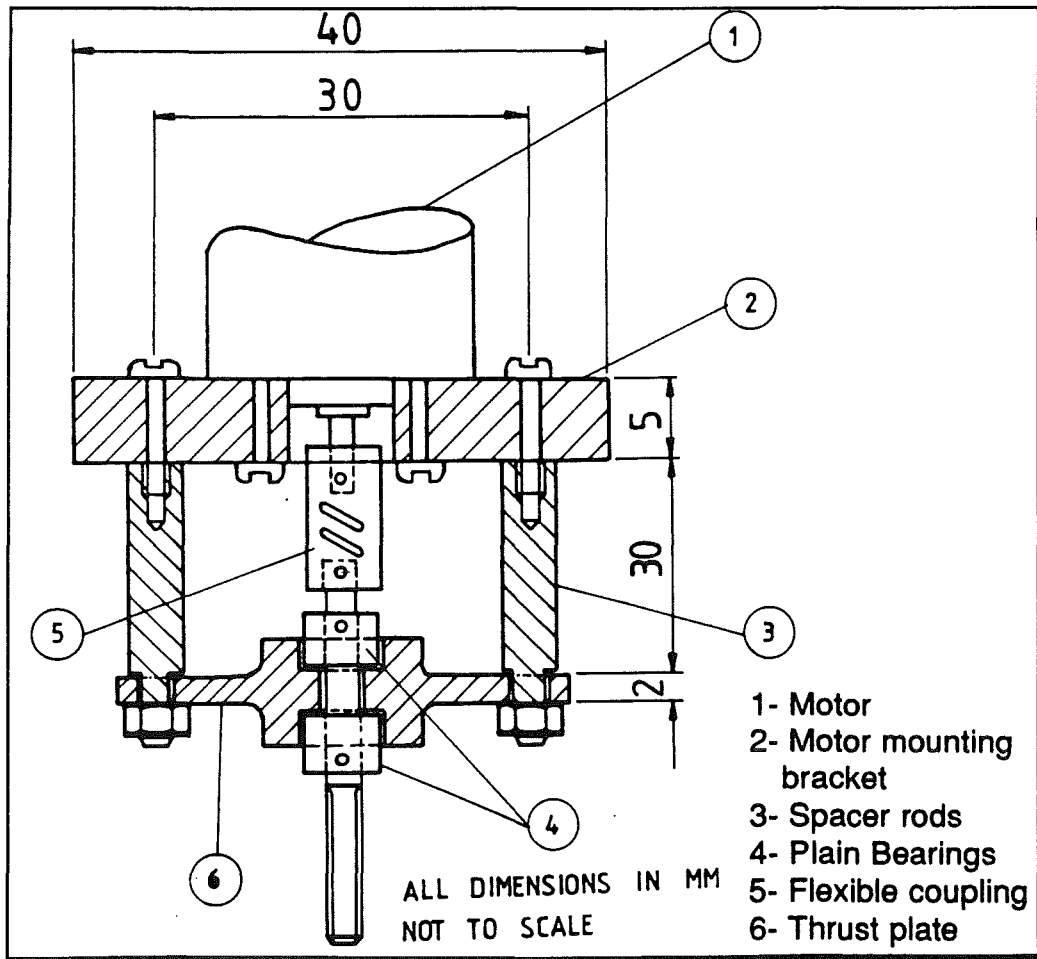


Figure 5.5, Z-translation drive assembly

Thrust plate spacers separate the thrust plate from the mounting bracket and give adequate clearance between the mounting bracket and the thrust plate for the flexible coupling to be fixed in place. The thrust plate initially consisted of recesses on either side to suit the thrust bearings. As specified by Auchterlonie Bearings based at London [64], the bearings should consist of two thrust washers: one should be an interference fit into the recess and the other an interference fit onto the shaft. In fact this design was used here and the recess washer was located by an adhesive (such as Loctite 601). The other thrust washer was located by a shoulder which was fixed to the shaft by a grub screw. The thrust bearings specified were 1/4 inch flat thrust bearings. Ideally, for the mechanism specified, 6mm thrust bearings should have been used but these are non standard and the cost therefore prohibitive (£20 as opposed to £4). The manufacturing costs were also taken into account in specifying the bearings, since additional machining would have been required to produce a shaft to suit a 1/4 inch bearing and a 6mm thread.

Since the thrust washers were made of a gauge steel, which in service will be in contact with an Aluminium surface, there is the possibility for a problem with future bi-metallic corrosion.

The performance of the drive mechanism was poor at first as a result of the bearings binding within their housings. This was significantly improved by implementing a different drive mechanism as shown in Figure 5.6.

This idea required less machining without the necessity for tighter tolerances. A 6mm bar stock was used and threaded, not necessitating additional turning operations. Two standard 6mm nuts were used which required holes to be drilled and tapped on the flats. This enable grub screws to be located which were used to fix the nuts in place on the screw thread. The threaded bar was then passed through the thrust plate and the nuts were fed onto the screws and screwed up to the plate.

Molybdenum-disulphide grease was applied to the plate and the nuts were then fixed in place by their grub screws. Effectively the result was a plain bearing using Molybdenum-disulphide grease as a lubricant. No perceptible end play could be detected and the reduced machining as well as the elimination of the

two thrust bearings represented a considerable cost and saving of time. This new mechanism is shown in Figure 5.6.

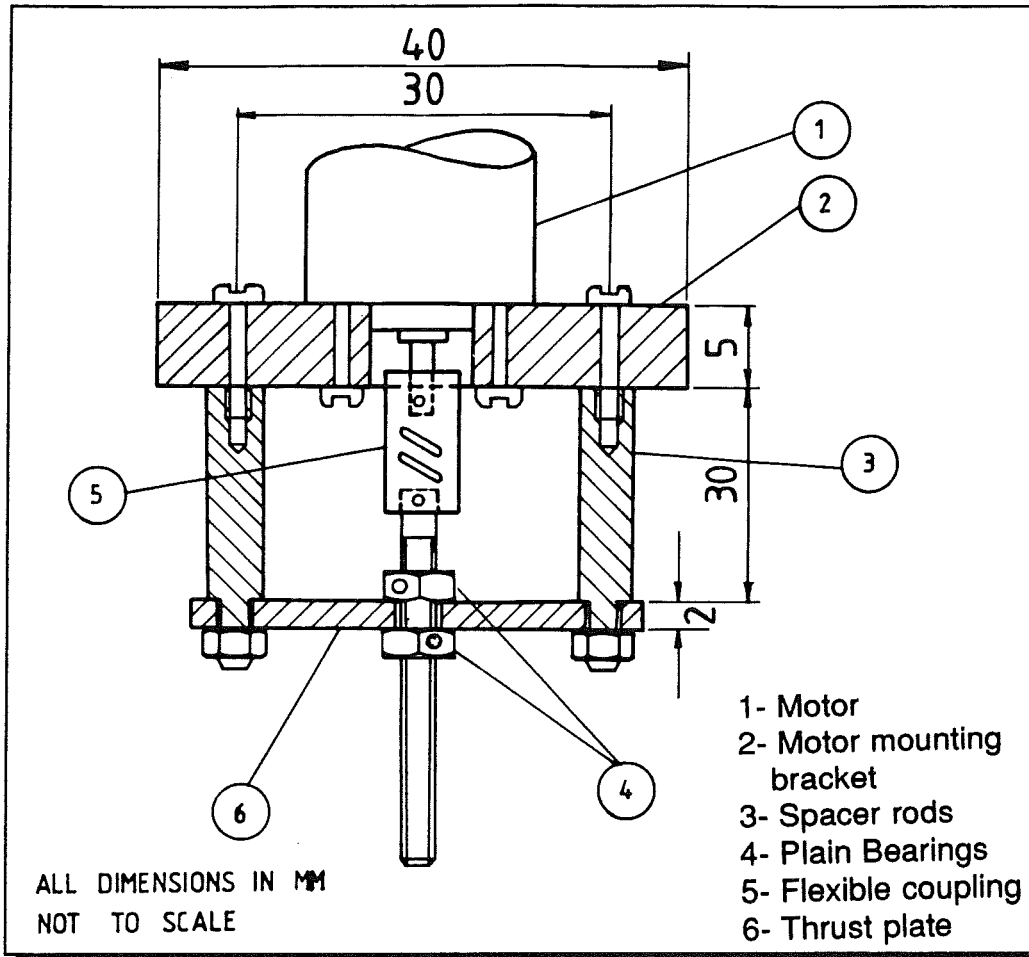


Figure 5.6, Plain bearing drive mechanism

### 5.3.6 Y-Translation Drive Mechanism

The mechanism is similar to the Z-translation drive mechanism except that two additional spacers are required (see Figure 5.3). The drive mechanism is fixed to the Z-translation tube and, to assist in this, the ends of the additional spacers are curved with tapped holes which allow fixing by screws through the Z-translation tube.

### 5.3.7 X-Translation Drive Mechanism

This mechanism is similar to the Y-translation drive mechanism except that the fixing is to the Y-translation support bracket as shown in Figure 5.3. This allows the whole drive mechanism to move in the Y-direction thus allowing translation in the X-direction.

# ***CHAPTER 6***

## **Data Acquisition**

### 6.0 Data Acquisition

Information from the force sensor array attached to the micro-manipulator has to be interpreted by a computer to enable it to perform the task of calculating the direction and magnitude of any force applied to the robots end effector.

This force information is obtained from an array of eight bridges of the force sensor (see chapter 4, page 51) and must be converted into a form in which the computer program is able to perform the calculations.

Since the computer can only perform calculations on digital information, and the information from the force sensor was in a very weak analogue form, an interface was provided to perform the conversion between these two. Part of this interface was an A-D converter, to quantise the analogue signal from the sensor into the digital form that the computer read through its internal I/O bus.

The properties of A-D converters vary greatly, but basically the differences between them are due to the following variables:

- 1: Accuracy.
- 2: Speed.
- 3: Cost.
- 4: External circuitry.

The third variable is usually a function of the first two, ie. the faster, and more accurate the device, the more it will cost. It is of course unnecessary to purchase an expensive, fast, and accurate A-D converter if it provides no better function than a much cheaper one. Conversely, a very cheap and simple device may not provide an accurate enough expression of the analogue signal being quantised, or be able to sample it fast enough to fully describe the rate that the signal is changing.

Common A-D converter integrated circuits can be obtained with between 4 and 16-bit accuracy, where the number of bits determines the number of voltage levels that can be described by the device between its reference voltages.

eg. If a 16-bit A-D converter has a lower reference of 5V, and an upper reference of 15V, then it will have  $2^{16}$  possible values describing the 10V difference between the references, or a resolution of  $10/65536=152.5 \mu\text{V}$ , this being the smallest change in voltage it can discriminate.

The time in which the converter completes the analogue to digital conversion is governed by the method used to perform the quantisation, such methods are; successive approximation, flash converters, dual slope, integrating, quad slope, ratiometric converters, subranging converters.

The most expensive, and the fastest are the flash converters, as these do not need a clock to perform the conversion, only to control the internal latches. A flash converter can perform the conversion within one clock cycle at many tens of megahertz, but this is achieved at the expense of internal complexity and hence at high cost. All the other types need many more clock cycles and can be much slower, down to a few kilohertz.

It was decided to choose a ready made PC add-on card to deal with the A-D conversion during design and experimentation. The PC-30A by Amplicon Liveline was selected and used with moderately fast 12-bit successive approximation converters with a 16-1 multiplexer, and a sample and hold circuit (general description of the board and detailed information are in Appendix {4}). With a total conversion time of only  $35 \mu\text{s}$  or up to 28,000 samples per second, this is many times faster than the rate at which the signal from the force sensor will be expected to change, taking into account the Nyquist sampling theorem which states that the maximum frequency that can be completely digitised is half of the frequency at which the samples are taken.

The multiplexer allows many analogue signals to be monitored by one A-D converter IC, and the "sample and hold" is used to hold the analogue input voltage during conversion to prevent it from changing, producing a meaningless result.

The PC-30A also has D-A converters and an I/O port that can be used in later parts of the design. A description of the role of PC-30A in the designed control system is on section 7.3.2, page 82. For full details refer to Appendix{4}.



Figure 6.1 shows the block diagram which interface the force sensor to the computer and illustrates the role of PC-30A. The output of the force sensors (block 1 & path a) is a very weak signal, and prone to electrical noise like radio and power supply interference. This signal must be amplified to increase its strength, filtered and screened to remove radio and electrical interference: the power supply to the force sensor must be very stable to prevent misleading signals.

While the force sensor follows a geometrical path the useful signals from the sensors were found to vary between DC and about 200 Hz. However, since higher frequencies still provide information about the path, such as surface finish, it was decided that a filter, removing signals above a few kilohertz, would allow the computer to obtain all the information about the force it needs yet limit radio and electrical interference.

Power supply interference can be present at all frequencies, but it is prevalent at up to about 100Hz due to the AC nature of the national grid power supply. This was within the range of the signals being measured, so a good power supply was seen to be of the utmost importance and the electrical screening of all the signal wires was therefore essential.

To enable experimentation a commercial strain gauge amplifier, the 2100 system from Welwyn Strain Measurements( Blocks 2&3) was used (for specifications refer to Appendix{4}, having a good power supply and amplifier with no more than 10  $\mu$ V/day drift, 10  $\mu$ V peak to peak noise and a filter with a dc - 15 kHz bandwidth.

The wires used to connect the strain gauges to the amplifier and the amplifier to the A-D interface (paths a & c) are an important part of the system, and an improvement in the accuracy of the signals being sent to the computer was made by using a high quality twisted pair of screened cables, minimising electrical noise, radio interference, and resistive effects.

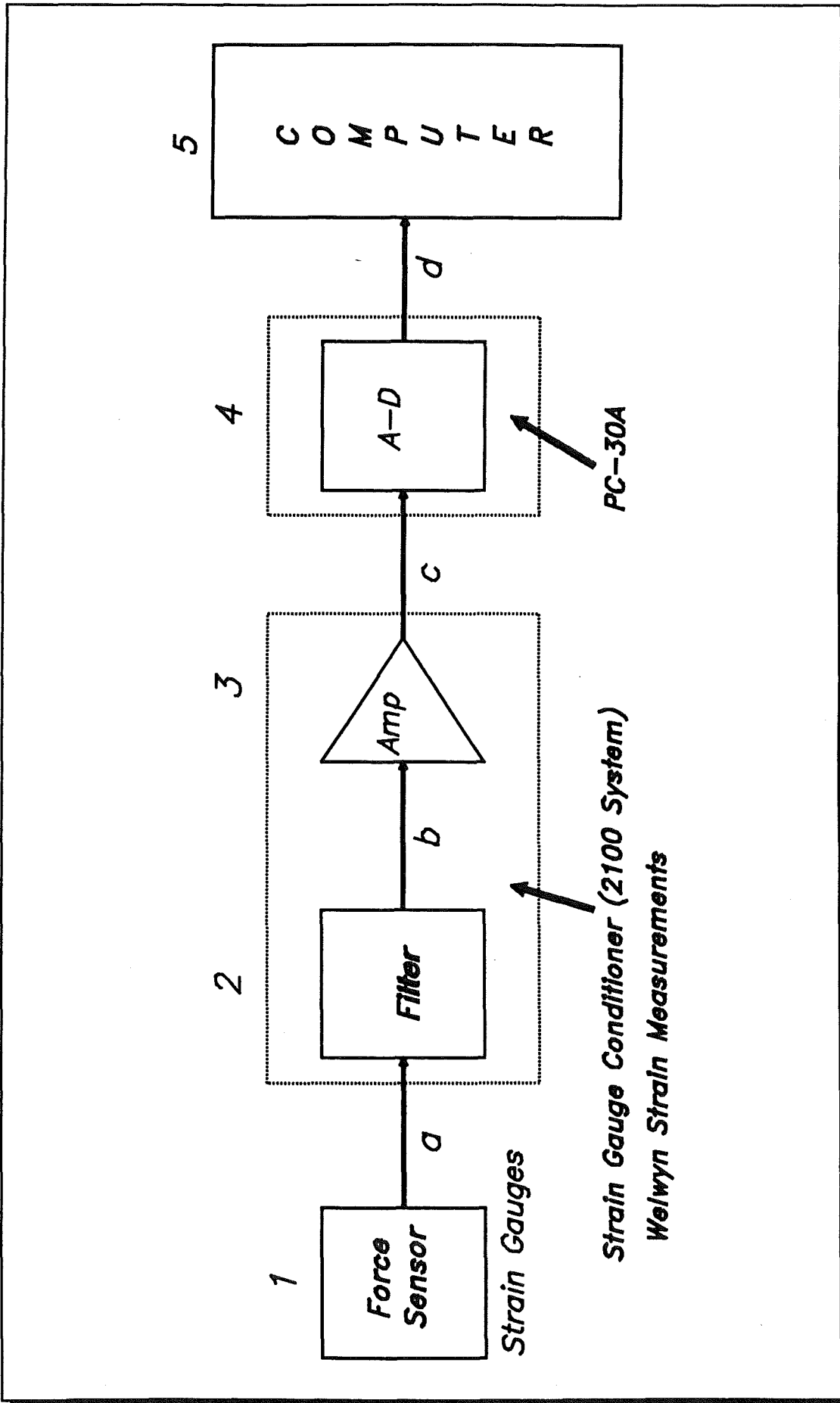


Figure 6.1, Data Acquisition Block Diagram

### 6.1 Errors Associated with Digitising a Signal

A discussion of the types of errors and inaccuracies that can be generated by digitising a signal are as follow:-

Quantising Uncertainty: The analogue continuum is partitioned into  $2^n$  discrete ranges for n-bit conversion. All analogue values within a given range are represented by the same digital code, usually assigned to the mid-range value. There is therefore an inherent quantisation uncertainty of  $\pm\frac{1}{2}$  LSB, in addition to any other errors.

Bandwidth: The bandwidth of the converter is half of the rate at which the samples are taken. If the signal changes faster than this then it will seem that the signals are actually changing at a much lower rate, and the voltages read will also be meaningless.

Non-Linearity: This is a number, usually a percentage given by the manufacturer to say how accurate the digital value is a representation of the analogue input, it is governed by the accuracy of the construction of the circuit within the IC.

Temperature Drift: This again is given by the manufacturer of the device, and is also an indication of how inaccurate the digital value is, it is expressed as parts per million per degree celsius (PPM/°C). For the AD574 used in the PC-30A, the linearity is better than eleven bits, and the drift is less than half a PPM/°C.

Offset Error: This occurs when an input voltage of 0V produces a binary output of other than zero.

# ***CHAPTER 7***

## **Motor Control**

## **7.0 Motor Control**

The micro-manipulator has three degrees of freedom, each axis driven by one motor. The position of the end-effector in space has to be controlled by some supervisory system, either a dedicated computer, or a control circuit interfaced to a computer.

The controlling system (e.g. a computer) must be able to control each of these motors independently and concurrently. It must also be accurate enough to sustain the accuracy required of the micro-manipulator, the controller should have a response time short enough to prevent unwanted anomalies such as hunting or overshoot.

To provide the first criteria, that of each motor able to work independently and concurrently, each motor should be controlled by an identical controller. The following description applies only to one motor controller which is identical to the other two.

### **7.1 Methods of Providing Positional Feedback Control**

There are five basic methods of providing positional feedback control of a DC motor, these are *Pulse Width Modulation (PWM)*, *Pulse Position Modulation (PPM)*, *Pulse Code Modulation (PCM)*, *DC Control*, *Binary Encoding*.

#### **7.1.1 Pulse Width Modulation**

This method involves supplying the controller with a stream of pulses, the frequency of which determines the position of the motor shaft, and the rate of change of frequency determines the speed at which the motor should move in the required direction.

This stream of pulses is compared with a similar stream from a second pulse generator this rate is determined by a positional sensor mounted on the motor shaft driving the micro-manipulator along each axis. Such sensors are potentiometers, Linear Voltage Differential Transformers LVDT's, or shaft encoders.

Any difference noticed by the controller between the two pulse trains is an error

signal, and is used to adjust the position of the motor.

#### **7.1.1.1 Problems involved with the PWM control system**

Due to the pulse nature of the system the motor is continually switched on and off, this makes the motor draw excessive current, and introduces a lot of electrical and RF (Radio Frequency) noise that will have to be filtered from the rest of the control logic, computer and force sensor amplifier.

Due to the construction there will always be a small error signal present on the motor, causing the motor to hunt or oscillate continuously. This can be removed with more control circuitry but at more expense and complexity. Solving this problem involves 'cutting' supply current to the motor when it is near enough to its required position. This adds yet another problem of a slight inaccuracy which may be undesirable in some applications.

PWM in its very nature needs continuous control by the computer, this adds extra processor overhead and would cause problems if used in this application. This can be solved by using a free running pulse generator separate from the computer, but again at more cost and complexity, with no obvious benefit.

#### **7.1.2 Pulse Position Modulation**

PPM is very similar to PWM, and differs only in that several position signals can occupy the same wire. Again the position information is a function of time, but instead of a change in frequency meaning a change in position, the time separation of each of the pulses indicates the position. This method is only suitable to small applications where only one channel of information is available, e.g. Radio Control. Due to the difficulty of accurately timing the separation of the pulses to any great resolution this system is only any use when accuracy is not a major factor.

#### **7.1.3 Pulse Code Modulation**

PCM is again used in remote control, again where only one channel of information is available. It is simply a stream of binary numbers, each indicating both the position signal number, and the position being encoded. TV remote

handsets, radio control, and digital storage mediums use this system.

As this is a strictly digital system only a finite number of positions can be encoded, secondly the binary information still has to be converted back into analogue or PCM on reception to be able to control the motor, so the extra complex hardware needed to implement it would provide no useful service.

#### 7.1.4 DC control

This method of encoding position information involves sending a DC analogue voltage to a controller, this voltage is directly proportional to the position required. A sensor mounted on the motor, similar to that used in the PWM system provides the position information for a controller to turn into another DC voltage, these two voltages are compared to produce an error, and this is used to adjust the motor's position.

The position information within the controller can be stored in a binary counter which is incremented/decremented by a sensor mounted on the motor. The binary value held by the counters is converted into analogue by a D-A converter to be compared with the position request from the computer also containing a D-A converter.

This system is similar to the PWM system in that it uses mounted sensor but the signals are DC, so less noise is present. It can be affected by noise, but cheap screening and a simple decoupling capacitor should prevent noise becoming too much of a problem. Lastly this method is easier to control, as a D-A converter inside a controlling computer can be programmed with the desired position to be sent to the controller and left, with no processor intervention, freeing the computer to perform the extensive calculation task given it by the force sensor.

#### 7.1.5 Binary Encoding

This is basically similar to the DC system with the two D-A converters missing, the position request from the computer is simply in the form of binary numbers, these are compared with the value held in a counter and any difference used for the error signal to a motor driver.

This method is most suited to an application where the computer is mounted on the control board, as each axis would require at least 16 wires, ie. 2 computer output ports. Another disadvantage is that without a local processor the comparison of the request and position numbers would have to be carried out with a large number of logic devices, which would be too complex. Its advantage is that less error can occur due to less data conversion taking place.

### 7.2 Selection of the control method

PCM, and PPM can be at once ignored as they are used more for limited applications having the limitation of only one channel of information. Both these methods still have to have the information received by the controller turned into PWM or DC so as to be in a form to control the motor.

With respect to accuracy PWM will have no limitation, the problem comes more with implementation of the system. To produce an effective PWM controller a large amount of programming and hardware is needed to overcome the problems mentioned in section 6.1, page 76. Both adding cost and more importantly complexity.

Therefore the last method, the DC control system seems to give a simple and effective solution to the problem. It should be adequate for the application in hand. A more in depth description of a proposed solution now follows.



### **7.3 Implementation of Selected Method**

#### **7.3.1 Summary of Block Diagram**

The block diagram (Figure 7.2, page 85) details a closed loop motor control system, it is considered by starting at block 1, path j where the desired position enters the system from the computer. This desired position is based on the information from the force sensor and the specified force levels in the software. For example, if force in direction X exceeds the specified limit in the software, the motor is retreated by some increment specified in the control algorithm. It is this "apparent" position which is compared with the actual position of the motor that generates an error signal. The error signal has logic applied, and the resulting signal amplified so as to drive the motor. The direction and distance the motor has moved is then measured by a counter, thus producing a new actual position and hence a new error signal. This cycle is repeated until the 'apparent position' is found to be matching the actual position. This will indicate that the force experienced by the force sensor is within the force level specified in the software.

#### **7.3.2 Role of PC-30A interface card within the control system**

The A-D and D-A converters connected to the computer shown in the block diagram as blocks 10 & 1 are part of the PC-30A interface system used to connect the controller and the force sensor to the computer.

The PC-30A contains a 16 channel 12-bit A-D converter, a two channel 8-bit D-A converter and two 12-bit D-A converters, it is accessed through standard IBM-PC I/O channels. For further details on the hardware and information on programming the PC-30A see appendix {4}.

The multi channel A-D was used to interface the force sensor (see chapter 5 on data acquisition) and also the path i tap to allow the computer to monitor the operation of the controller. The two 12-bit D-A converters are used to send the desired position information to the controller and are configured in bipolar mode with a  $\pm 10\text{v}$  range. Three controller boards are used, one for each axis, so three D-A converters within the computer are needed. Hence, a second PC-30A

card was employed to provide the other channel in order to control the third motor.

It would be desirable to use 16-bit D-A converters rather than 12-bit, giving 16 times improvement in resolution, but this would mean building a custom interface card with three 16-bit D-A converters and a 16 channel A-D converter as such an interface card was not available. This extra resolution was not deemed necessary for a prototype though a production model would benefit from such an interface.

### 7.3.3 Detailed Explanation

To ease the description of the design, the system has been split up into a block diagram (Fig 7.2). The final circuit diagram which can be referred to is in Appendix IV, page 206.

The required position calculated by the computer (Block 11) is converted into analogue form by the Digital to Analogue Converter (Block 1) of PC30-A board mounted inside computer. The signal is sent via a single wire (j) to the comparator on the controller board (sum junction 2) this finds any difference in the signals between the desired and actual positions, and provides an error signal (a). The logic (Block 3) decides which way the motor should be turned to achieve an error signal of 0V. The combination of blocks 2 & 3 of the block diagram shown in Figure 7.2 can be replaced by a single circuit shown in Figure 7.1.

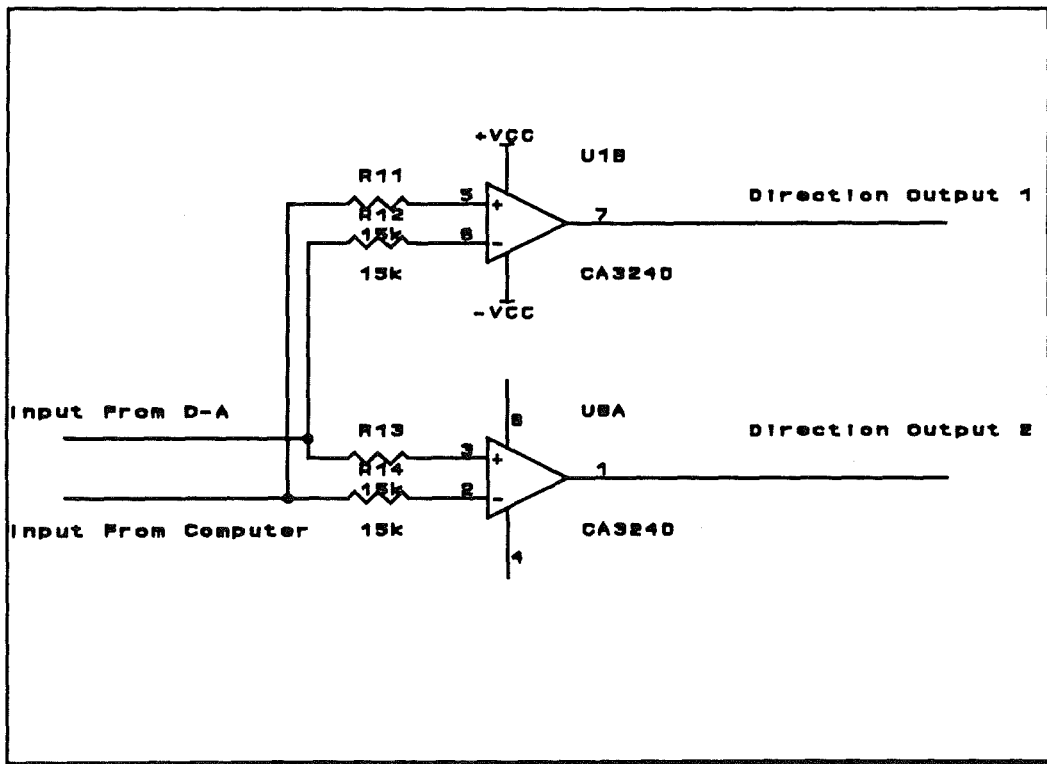
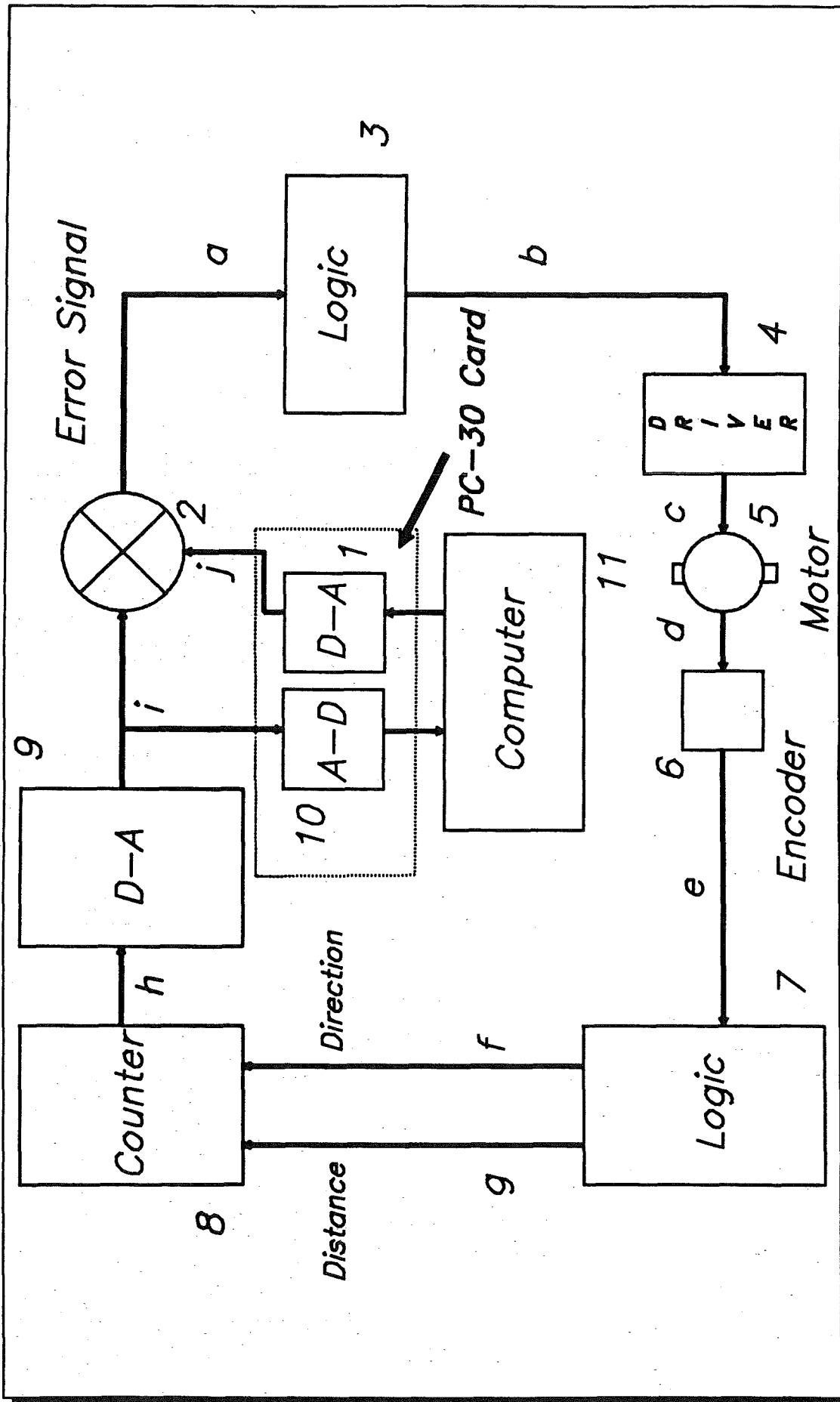


Figure 7.1, The Direction Decoder Circuit

This is split in the block diagram simply to aid understanding. It provides the logic needed to encode the three operations the motor must perform, *ie.* STOP, LEFT and RIGHT.



Block Diagram showing a closed loop control system

Figure 7.2, Block Diagram showing a closed loop motor control system

To illustrate how the comparator performs the logic, the table below considers the possible inputs that it may get from the Computer (signal A) and the feedback signal (B). It works by simultaneously adding and subtracting the bipolar signals (both with a range of -10V to 10V).

ie.  $X=A-B$

$$Y=B-A$$

Note: The op-amp cannot produce an output greater than its supply of  $\pm 10V$  even if the calculation should generate a greater answer.

(For all  $A>B$  signals)

$$A= 10V$$

$$B=-10V$$

$$A-B= 20V \text{ (10V) converted into logic 1}$$

$$B-A=-20V \text{ (-10V) converted into logic 0}$$

(For all  $B>A$  signals)

$$A=-10V$$

$$B= 10V$$

$$A-B=-20V \text{ (-10V) converted into logic 0}$$

$$B-A= 20V \text{ (10V) converted into logic 1}$$

(For all  $A=B$  and are both -ve)

$$A=-5V$$

$$B=-5V$$

$$A-B=0V \text{ converted into logic 0}$$

$$B-A=0V \text{ converted into logic 0}$$

(For all  $A=B$  and are both +ve)

$A=5V$

$B=5V$

$A-B=0V$  converted into logic 0

$B-A=0V$  converted into logic 0

The gain of the op-amp is in the region of  $2 \times 10^5$ , so less than  $5 \times 10^{-6}V$  difference is needed between the inputs to produce 0V at the output. This difference is what the system will try to sustain.

The above logic states the word 'converted' at the end of each of the results, this conversion is performed by a level shifter Figure 7.3 (i.e. Block 3 of the Block Diagram), as the bipolar result from the op-amps will not directly interface to the motor driver explained below. The 14c89 in Figure 7.3 expresses any voltage greater than 0.7V as logic 1, and any voltage less than this as logic 0.

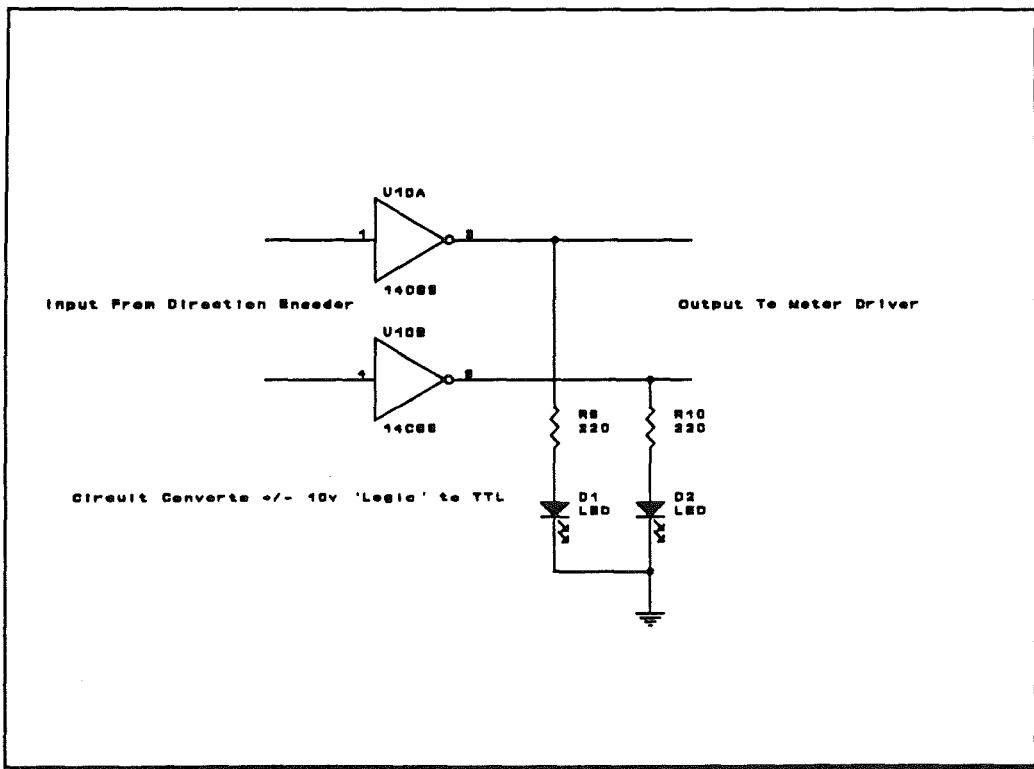


Figure 7.3, The Level Shifter Circuit

The motor driver (block 4) is used to supply the motor with the relatively large current it requires (a load test gave up to 1A under load or 100mA no load). The driver is known as a full bridge driver, this name indicates its internal design enabling it to move the motor in both directions. A full bridge driver is fairly simple to apply using transistors but, to eliminate calibration problems and to reduce circuit complexity, an "off the shelf" driver was chosen. Manufactured by Burr-Brown, the device chosen for the design was the L6203 (See Appendix {4} for Data Sheets). This was used partly because of the full bridge construction and partly because its logic compares perfectly with that generated by the comparator/logic in blocks 2 & 3. Figure 7.4 shows the motor driver circuit, and is followed by the logic tables 2 & 3 detailing the signals from the comparator to the motor (paths b & c).

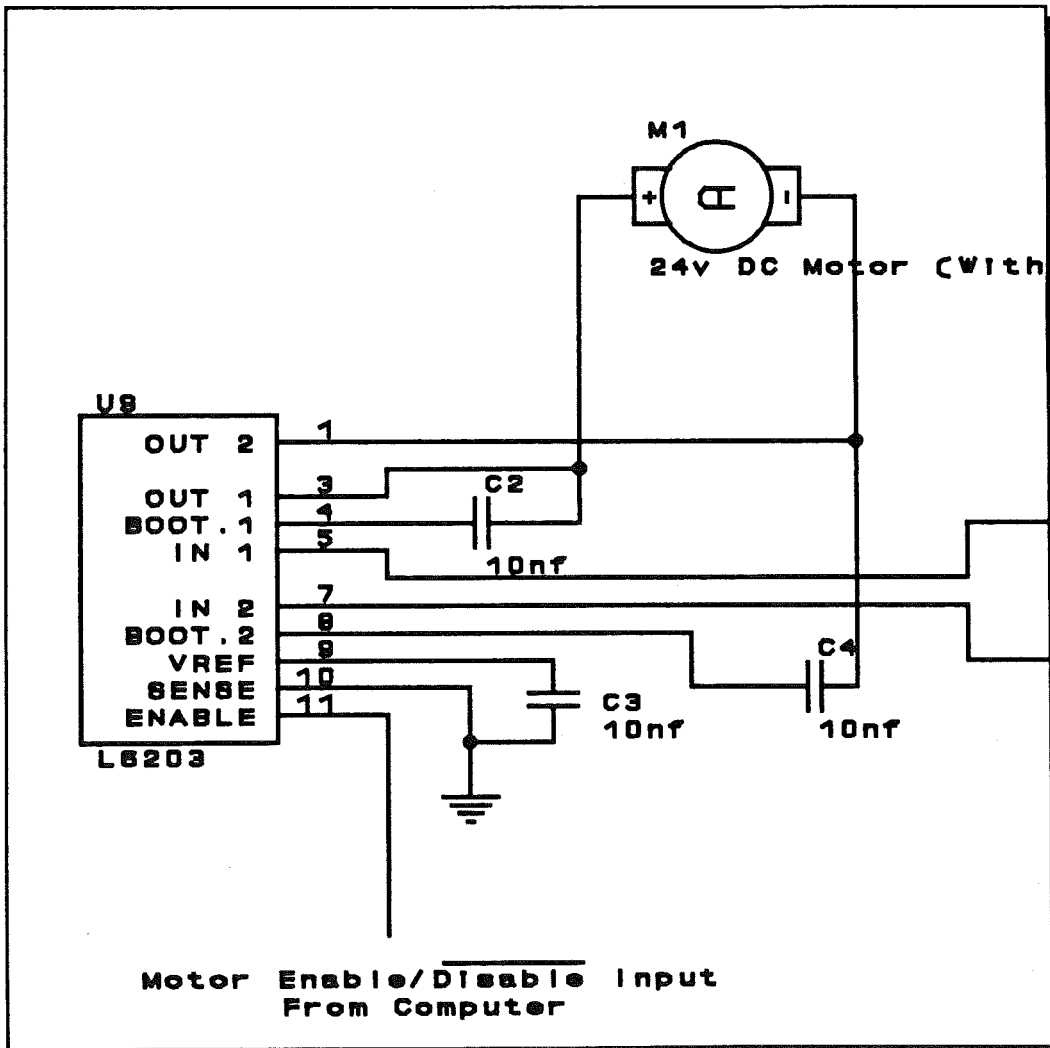


Figure 7.4, showing the Motor Driver

Comparison	Binary Output		Direction
	X	Y	
$+A > +B$	0	1	Forward
$+A > -B$	0	1	Forward
$-A > +B$	Impossible		Never Happens
$-A > -B$	0	1	Forward
$+A < +B$	1	0	Reverse
$+A < -B$	Impossible		Never Happens
$-A < +B$	1	0	Reverse
$-A < -B$	1	0	Reverse
$+A = +B$	Both Equal		Stop
$+A = -B$	Impossible		Never Happens
$-A = +B$	Impossible		Never Happens
$-A = -B$	Both Equal		Stop

Table 2, Shows the summary of the comparator outputs



The table for the L6203 is listed now so the similarity can be seen.

A	B	Enable	Output
0	0	0	Motor Free
0	1	0	Motor Free
1	0	0	Motor Free
1	1	0	Motor Free
0	0	1	Stop
0	1	1	Anti-clockwise
1	0	1	Clockwise
1	1	1	Stop

Table 3, Truth table for the Bridge Driver L6203

The L6203 makes use of the two possible methods of encoding the STOP signal that the comparator will produce.

A full bridge driver uses regenerative braking whereby the motor contacts are effectively shorted (due to the back emf), stopping the motor much faster than just simply removing the power.

The phase distance information from the encoder integrated to the motor (block 6, path de) must be turned into direction-distance codes. The encoder produces a pair of sine waves (path e) of about 1 volt peak to peak, with a small bias. Each wave is  $\pm 90^\circ$  out of phase with the other: the leading waveform depends on the direction of the motor shaft.

The waves are first clipped, using a pair of op-amps. (see block 7 on block diagram, Figure 7.2). This is to interface the analogue sine wave to the digital circuit. The level at which the waves are clipped is set using a preset. The 1k Ohms resistor on the input is to provide an emitter current for the opto-transistor inside the encoder.

A diode clamping circuit shown in Figure 7.6 was used to clip the sine wave as

shown in Fig.7.5, but the op-amp also buffers the relatively high impedance sources in the encoders hence giving the signal more strength.

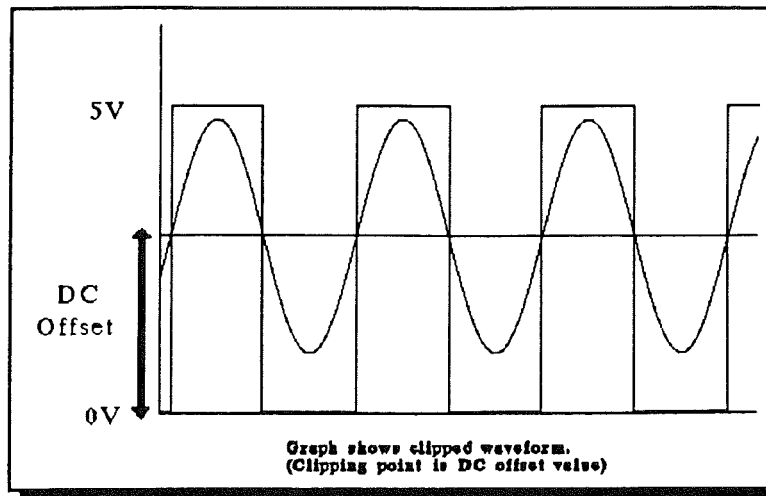


Figure 7.5, Graph shows clipped waveform. (Clipping point is DC offset value)

The signals could have the DC component removed,

using a coupling capacitor and the resulting waveform then compared with 0V. This would combat the problem of DC drift changing the clipping point. The clipper circuit shown in Figure 7.6 is used due to the frequency range needed by this part of the system (about DC to 1 kHz). An op-amp had to be chosen that allowed a single rail supply and the input to swing below the negative supply by up to 1V (if AC coupling became necessary) the CA3240 was used as it contains two op-amps on one chip.

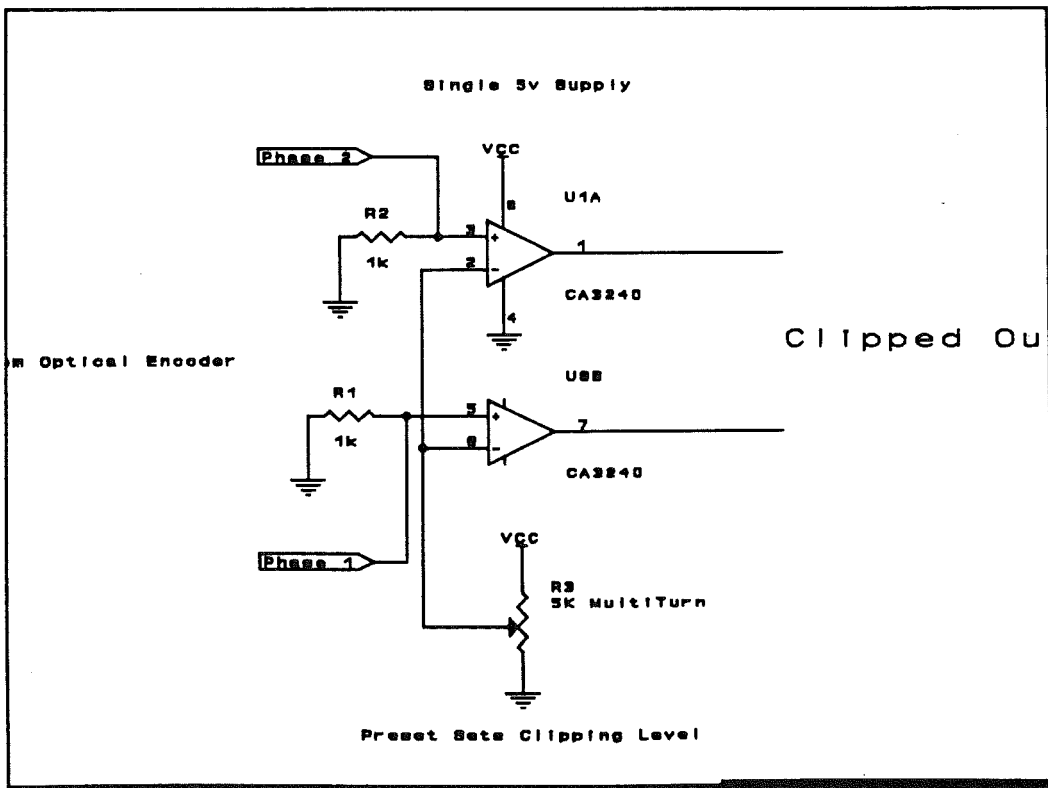


Figure 7.6, Clipper Circuit

If later DC drift became a problem, the other method, utilising a DC filter could be used, but it could require a pre-amp with automatic gain control to stabilise the input amplitude over a range of shaft speeds.

Block 7 also separates the distance information from the direction information using a D-type flip flop, this produces a boolean signal for the direction (path f), and a series of pulses to indicate the distance moved in a particular direction (path g). Figure 7.7, shows two TTL waveforms as being the two inputs to a D-type flip flop illustrated in Figure 7.8.

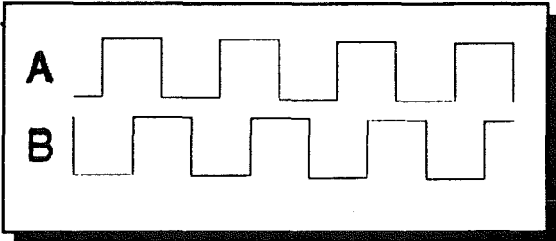


Figure 7.7 can be used to show how the D-type flip flop is used to decide the direction. The information on the D input is transmitted to the Q output only on the rising edge of the clock signal, so if the clock rises before the D input then Q will be low, and if the clock rises after D then the output Q will be high. Referring to Figure 7.7, if 'A' is the clock input and 'B' is the D input, then the output will always be logic 1 as the information in 'B' is clocked into the Q output on the falling edge of 'A'.

Figure 7.7, shows the input to the D-type Flip-Flop from the clipper

Referring to Figure 7.7, if 'A' is the clock input and 'B' is the D input, then the output will always be logic 1 as the information in 'B' is clocked into the Q output on the falling edge of 'A'.

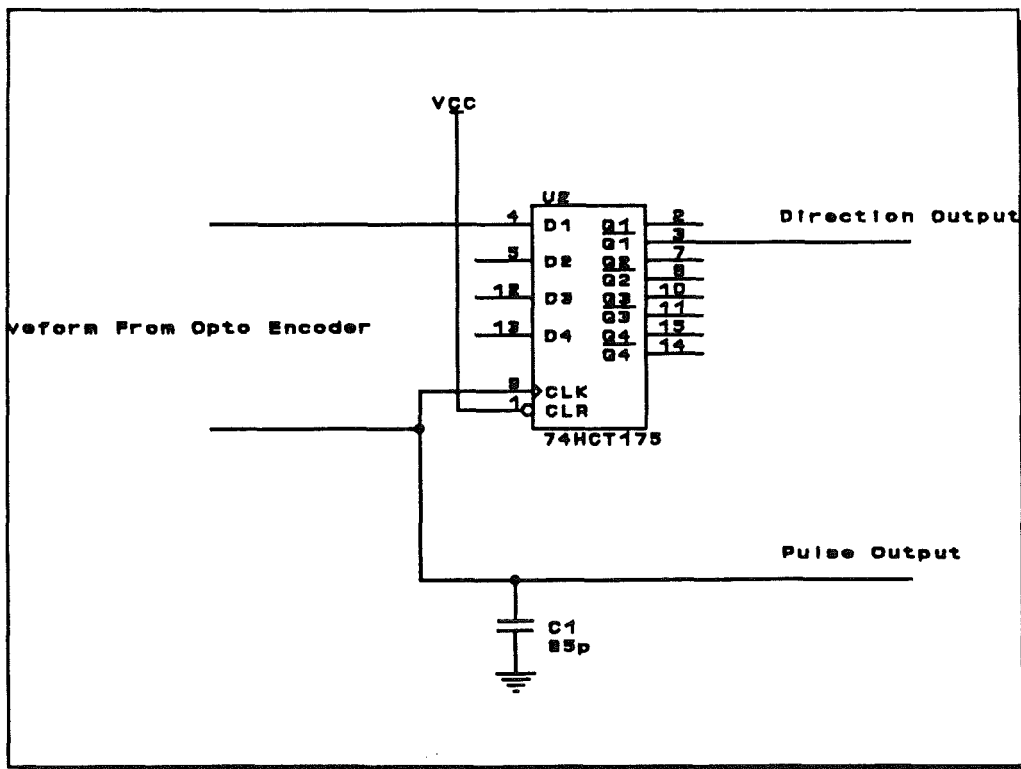


Figure 7.8, showing the DFF Direction decoder

In the contrary, if 'A' is the D input and 'B' is the clock, the output will always be logic 0. The Q output, Figure 7.8, is used to indicate the direction. Therefore by applying the inputs to the device, either the clock leading by 90°(case 1) or the data leading by 90°(case 2).

The distance/direction signals (paths g & f) are passed onto the counter, this is a 16-bit up/down counter. This could allow the control circuit to encode 65536 points along the manipulators travel of 2cm. This is nearly 0.3  $\mu\text{m}$  minimum possible resolution, but it is limited by the gear reduction ratio (16.2) of the motor and the 1 mm pitch of the screwfeed drive mechanism. If the D-A converter (Block 9) was given a  $\pm 10\text{V}$  supply then it will have an analogue resolution of  $20/65526\text{V}$  or  $304 \mu\text{V}$ . So noise within the analogue section should be limited to a level smaller than this.

Binary up/down counters are usually available in either 4 or 8 bits, 16-bit counters are rare and do not often come with all 16 outputs accessible, so one has to be constructed by cascading several together. The 74LS169 is a widely available device, and is easy to cascade each device is one four bit counter, so to make 16bits four counters are needed.

One problem with cascading counters is that it can only be done asynchronously, i.e. the change in MSB (most significant bit) triggering the clock of the next counter in the chain. This introduces a delay between the triggering of the first counter, and the value on the data lines becoming valid. This would cause invalid data values to enter the D-A converter (block 9), thus introducing troughs and spikes in the output analogue value. A way to improve this situation is to latch<sup>3</sup> all the data lines after the clock has been triggered so that the data entering A-D converter is valid, but this just adds more complexity to the circuit. The LS169 allows for a better solution by ensuring that the carry to the next device is synchronous: the previous choice in the chain enables the clock to the next device, the clock itself is common to the whole chain of counters. Figure 7.9 shows the counter circuit.

---

<sup>3</sup>latch: In electronics, a circuit that maintains an assumed position or condition until it is reset to its former state by external means.

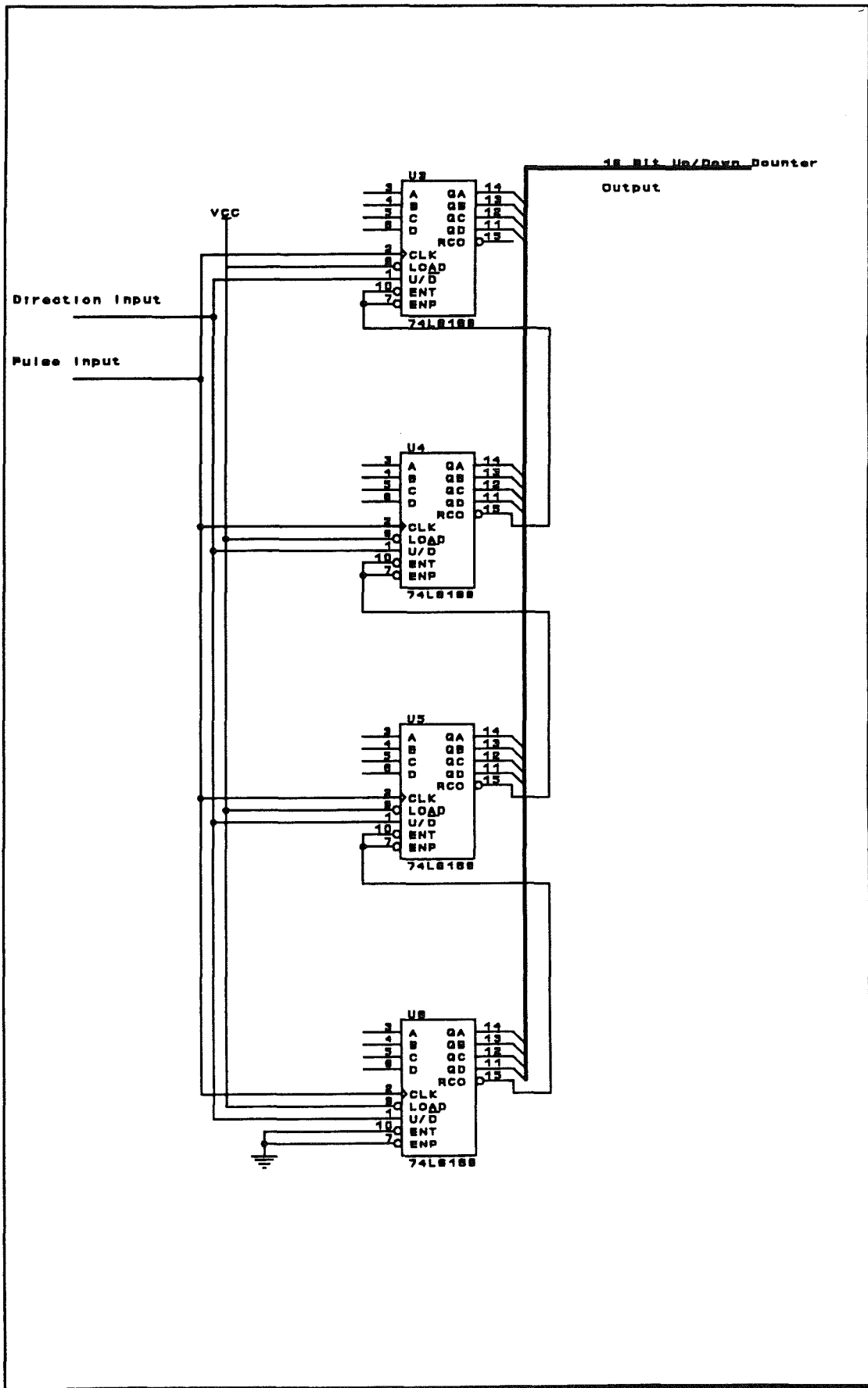


Figure 7.9, The 16-bit Up-Down Counter

The output from the counter (16 bits wide) is sent via path h to the D-A converter discussed next.

The digital to analogue converter (Block 9) was chosen to fit three criteria:

- 1: Price
- 2: Resolution
- 3: Circuit Complexity

The DAC703JP fitted the above criteria perfectly (see Appendix {4}, for Data Sheets). It is fast enough, relatively cheap and has a 16-bit resolution. The circuit used is the standard manufacturer's suggested configuration. This is shown in the accompanying diagram Figure 7.10.

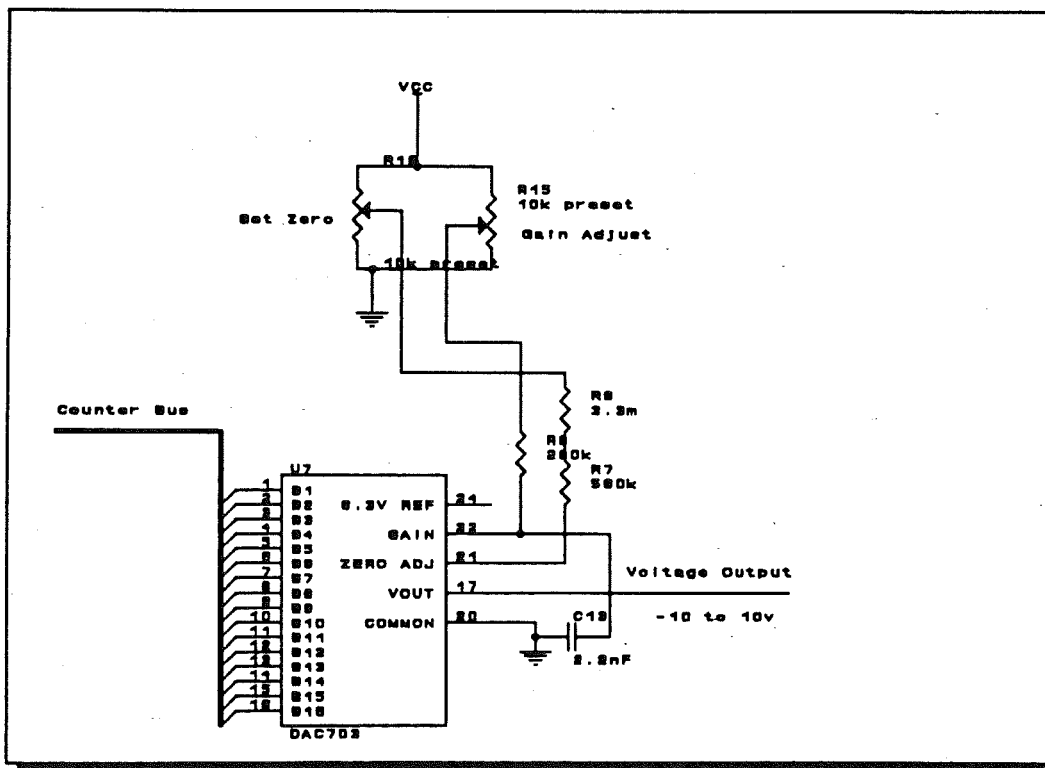


Figure 7.10, The D/A converter

Lastly the output of the D-A is sent to the comparator to complete the feedback loop (along path i). A tap is taken off path i to enable the software to check the position of the control system, but this plays no part in the overall feedback or

control system, it was only used for software debugging, and hardware testing.

The complete control system is set up as shown in Figure 7.11, this simply highlights the fact that three identical controller circuit is used to achieve a working system, and that the computer and controlling program connect the whole four component system together.

(Force Sensor & 3 motor axes system).

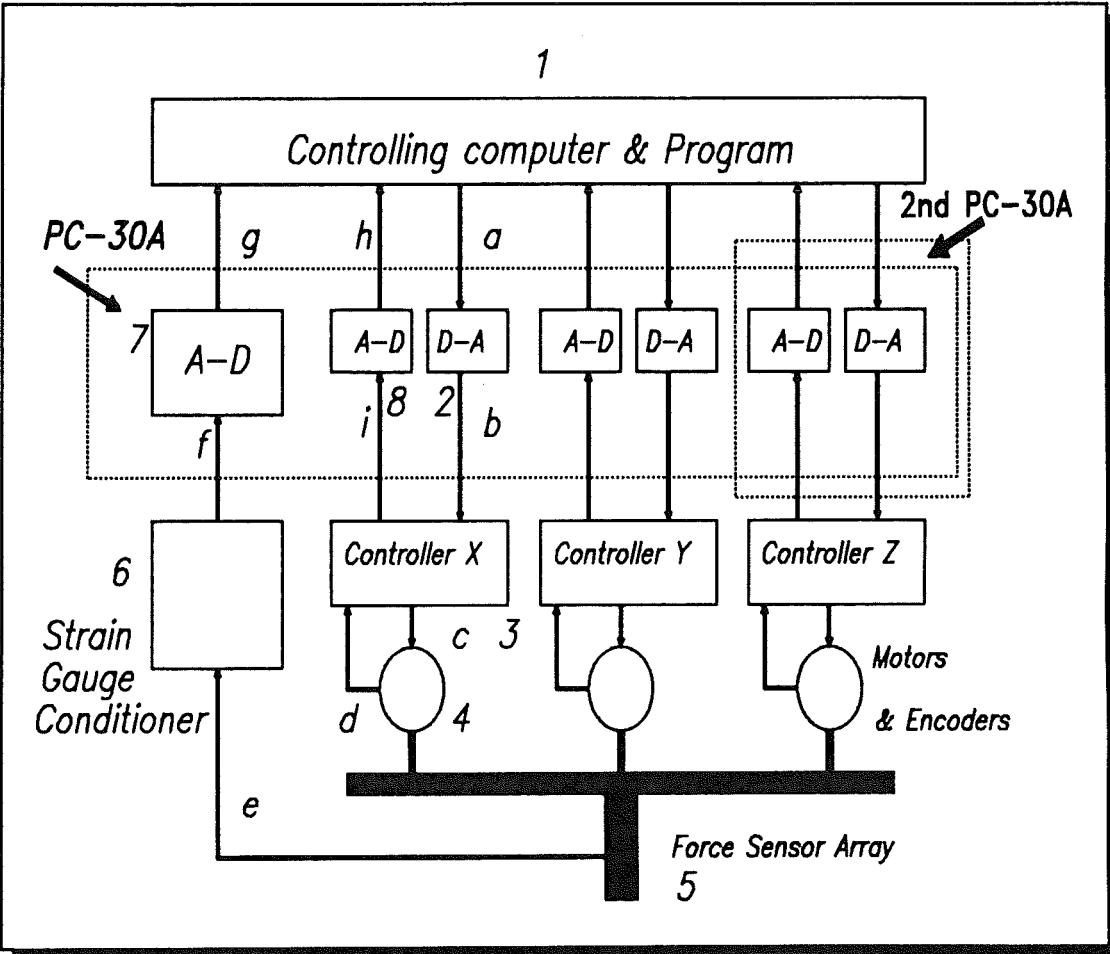


Figure 7.11, The complete control system for the micro-manipulator

Note that again the paths h/i and hence block 8 are *not* part of the feedback system, they are only to allow for experimentation with the software.

**7.3.3.1 The Need For Auto Setup**

When the system is powered up the controller does not know where the micro-manipulator is along its travel, also the computer does not know the position of

either the controller or the manipulator. This is the main problem with the chosen system which is one of strictly relative movement. All that is needed though is a method of determining for sure where each of the components are.

An automatic set up method was devised to cure this problem. It involves a slight modification to the counter system and three opto-sensors mounted at each end of the micro-manipulator axes. The idea is that, on starting, the computer requests the axis to move to the end where the sensor is mounted; on reaching this point the sensor is activated and the counter is loaded with the binary value of this position. The computer has only to wait long enough for the motor to move through the whole axis to be sure of its position.

The computer monitors the force values during this operation to be sure that the micro-manipulator does not hit any obstacles and, if it does, stop the motors and warn the operator.

The algorithm below summarises the above method of removing the uncertainties at power up.

- 1- Turn off the motor
- 2- Enable auto reset (Also disable counters)
- 3- Set computer to FSD (Full scale deflection)
- 4- Enable Motor

As the counter is disabled the counters will always be at a value greater than FSD, so the motor will always move towards the negative end of the travel.

- 5- Wait for motor to reach destination (FSD)
- 6- On reaching -FSD an Opto Slot Sensor is triggered.

The opto sensor enables the counter briefly and loads the counters with the counter value of FSD.

- 7- As the motor and the computer both show FSD the motor stops.
- 8- Disable motor
- 9- Disable auto reset (Enable counters)
- 10- Set computer to move the motor to mid position.
- 11- Enable motor

The motor now moves to the mid position, and the computer can be sure it



knows where it is. The alteration to the design of the counter circuit is shown in Figure 7.12.

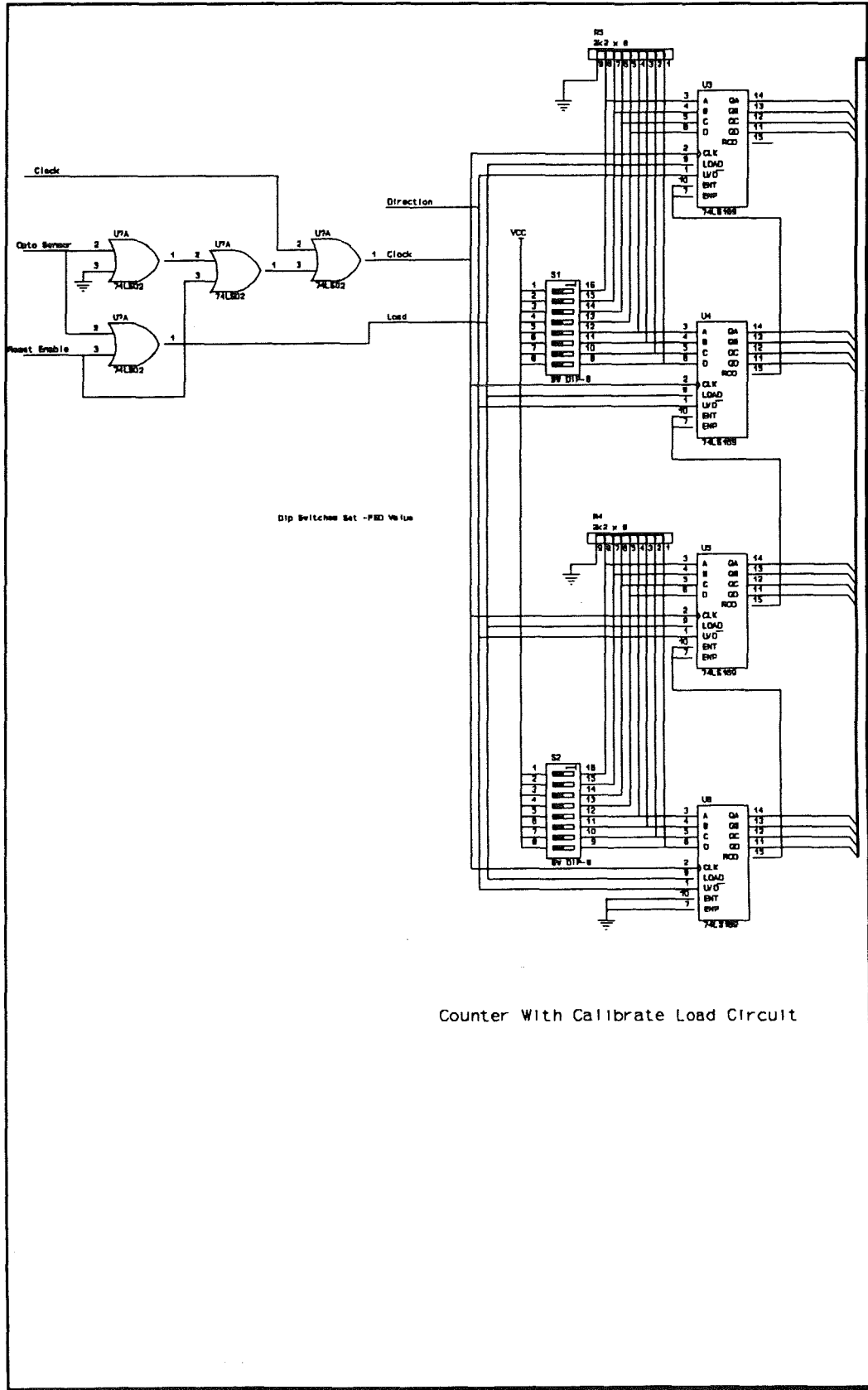


Figure 7.12, Modified Counter Circuit

# ***CHAPTER 8***

## **Software**

## 8.0 Software

The implementation of the developed system as a result of this research is demonstrated by performing two specific separate tasks, these are:

- 1- Inserting a peg into a hole.
- 2- Following an unknown geometric path.

In both situations, the robot arm will introduce the micro-manipulator to the workpiece, and either on an instruction from the robot controller, or from the computer, the micro-manipulator must complete the task, supervised by an application specific computer program.

The computer program will also monitor the forces on, and the position of the micro-manipulator, to ensure that damaging excessive forces are not applied, and sending commands to the robot controller to stop it in a dangerous situation.

Of course the computer should also try and ascertain the failure or completion of the task.

The implementation of the above tasks required here the development of software specially written for the system.

Most of the software written to control the system was written in the C language, this was to make use of the fast compact object code<sup>4</sup> that can be generated due to the brief and cryptic nature of this language. During development of the programs, the code was tested on an Intel 286 PC compatible with a maths co-processor. The co-processor is able to do complex calculations in a fraction of the time of the equivalent machine code program and was needed because of the large number of calculations that had to be performed for this work.

The computer was programmed to perform all the calculations needed to read the voltages from the strain gauges, convert them into forces, decide the

---

<sup>4</sup> Object code is the machine code produced by the high level language compiler.

correct direction in which to move the motors according to the current strategy, and inform the user of the status of the system.

Within the working versions of the software (ie those not designed to be used for experimentation, testing and calibration of the system), the programs were designed to run as fast as possible. This meant removing unnecessary screen output and unused calculations to give the computer more time for actually controlling the motors.

Flow chart in Figure 8.1, showing the basic program operation.

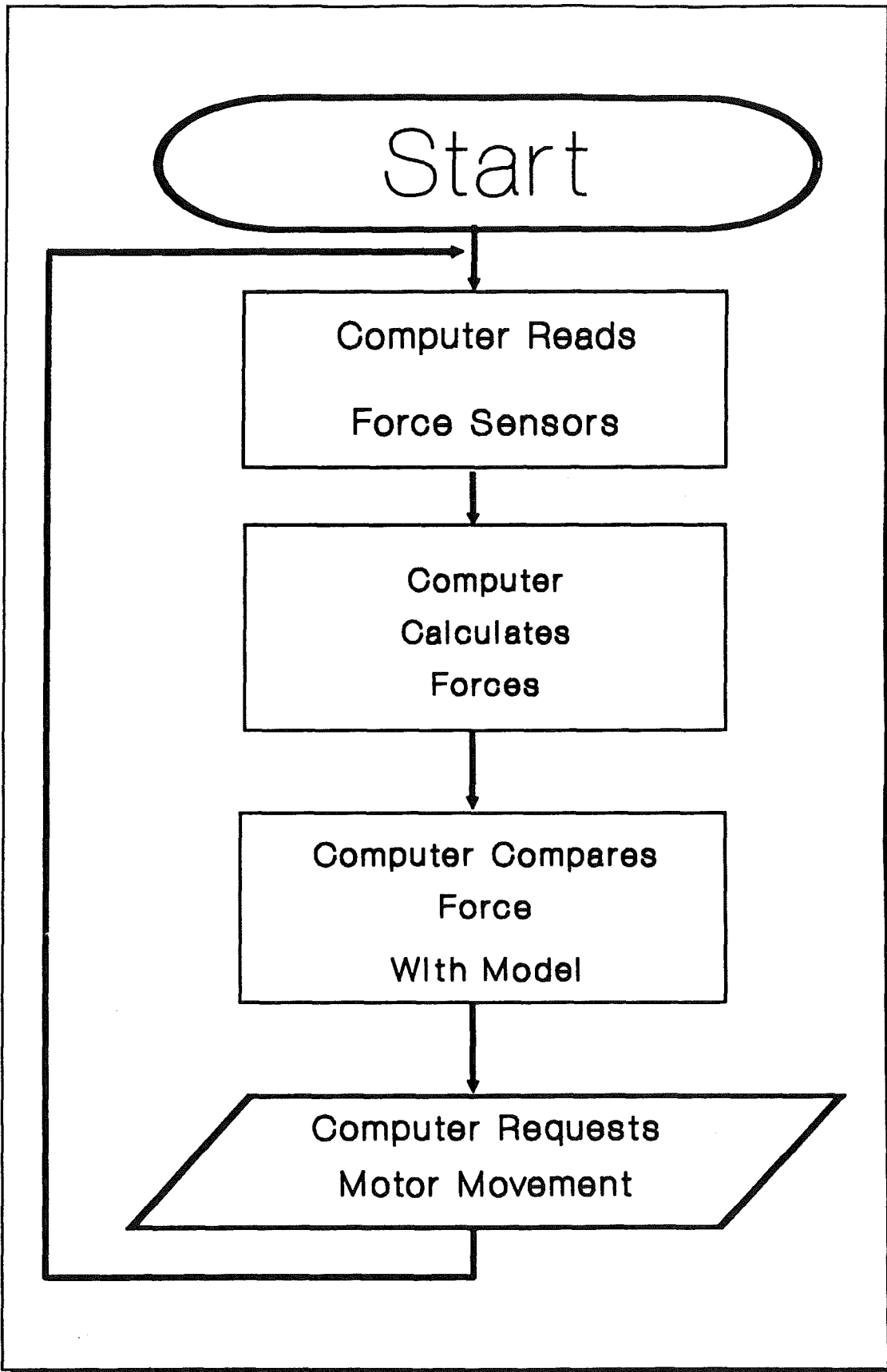


Figure 8.1, Flow chart, showing basic program operation

### 8.1 Preliminary Program Operations

The programs perform four main operations, these are;

(i) "Setup", this includes determining the absolute position of the micro-manipulator on power up.

(ii) "Sample", this is the process of gathering the information from the strain gauges.

(iii) "Calculation", this transforms the voltages into forces, torques and direction information.

(iv) "Movement", this final stage includes deciding which motor(s) to move, and in what direction, and finally moving the motors, checking the calculated position for an over range error (i.e. attempting to move the micro-manipulator to a position outside the mechanical range).

Other less important functions of the program are to display "force/direction", "actual position" (as read back from the controller), and desired position (as calculated by the computer).

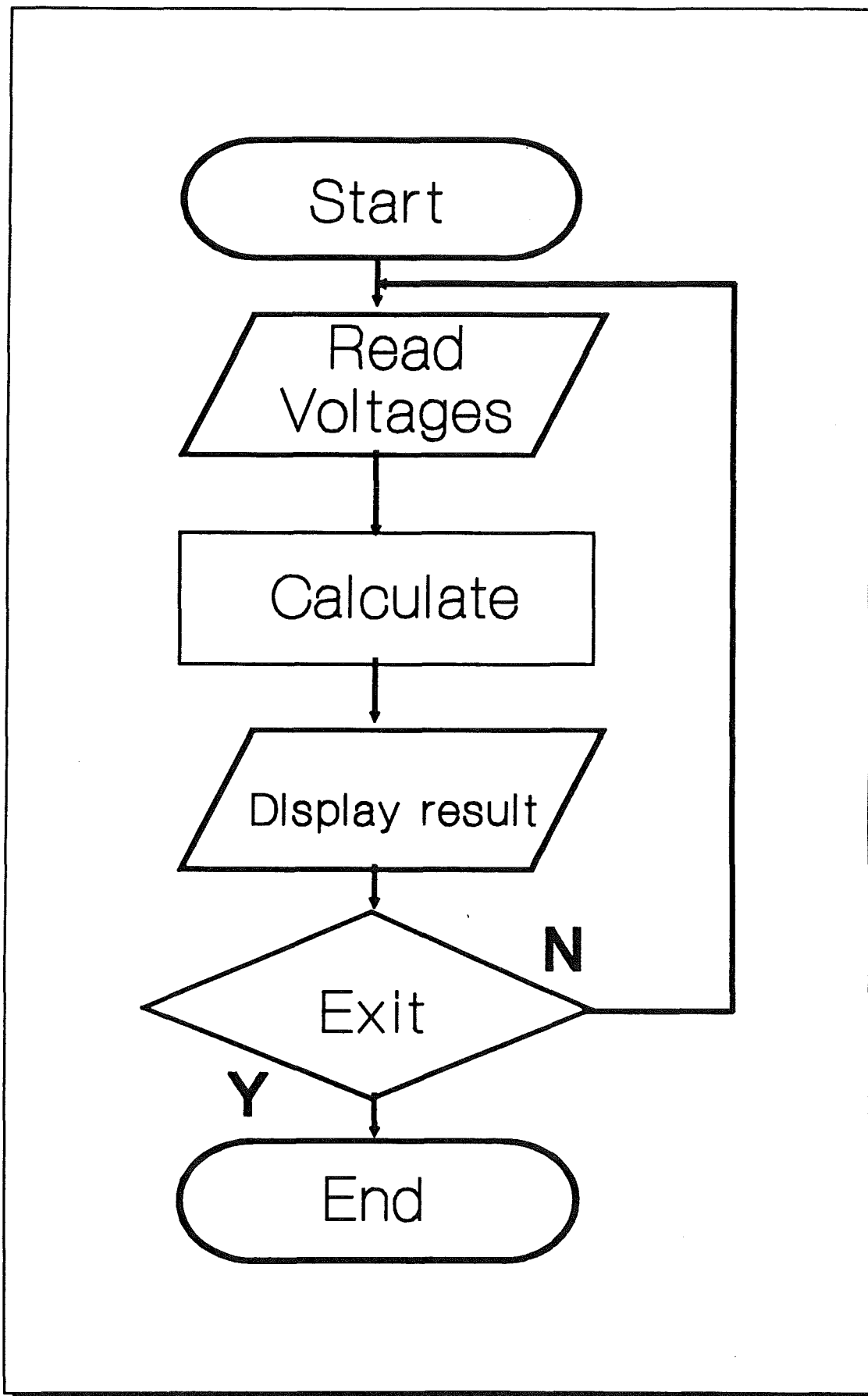
The programs written are accompanied by flow charts to illustrate the flow of information. Descriptive notes are added to each software to remark important points.

Program 1, was used during testing to help check the interface between the force sensor and the computer(PC-Board 30A). It simply reads the voltages and displays their values on the screen until a key is pressed, thus allowing notes to be taken for the purpose of calibration of the analogue to digital conversion. As a result the voltage calibration factors were calculated for use in further programs. The program can also be used to find the optimum value for the delay after the conversion allowing the A/D converter to settle.

Program 2, is a test program that allows the micro-manipulator to be moved and tested using cursor keys in order to optimise the performance of the control board.

Program 3, digitises the output of the strain gauge conditioner unit and calculates the force and torque in three orthogonal directions. The program was later modified into its current form to compare these forces with the model determining its tasks ie. that of peg-in-hole and path-following. This third program was then modified for the purpose of following the path in the designed rig.

The scale factor of 4.8828, used in the programs that make use of the force sensor, was found by placing a DVM (Digital Volt Meter) on the input, and calculating the number needed to make the voltage read by the computer the same as that displayed by the DVM. For example placing +10v on the input (checked by the meter) produced a binary value of (1)1111111111 (+2040, the bit in brackets is the sign bit). Thus  $10000\text{mV}/2048=4.8828$ , this was used as a scale factor to convert the binary value into the 'analogue' representation.



Flow chart of program 1, Showing the operation of the routine to display voltage readings from the force sensor.



```

/*Program 1*/
/* Test utility for force sensor */

#include <stdio.h>
#include <conio.h>
#include <dos.h>
#include <math.h>

#define base1 0x0700

void sample(void);

double data[10];

main()
{
    int i;
    long int d;

    while(kbhit()==0)          /* Keep going until key is pressed*/
    {                          /* Get data from strain gauge */
                              /* amplifier and clear the screen*/

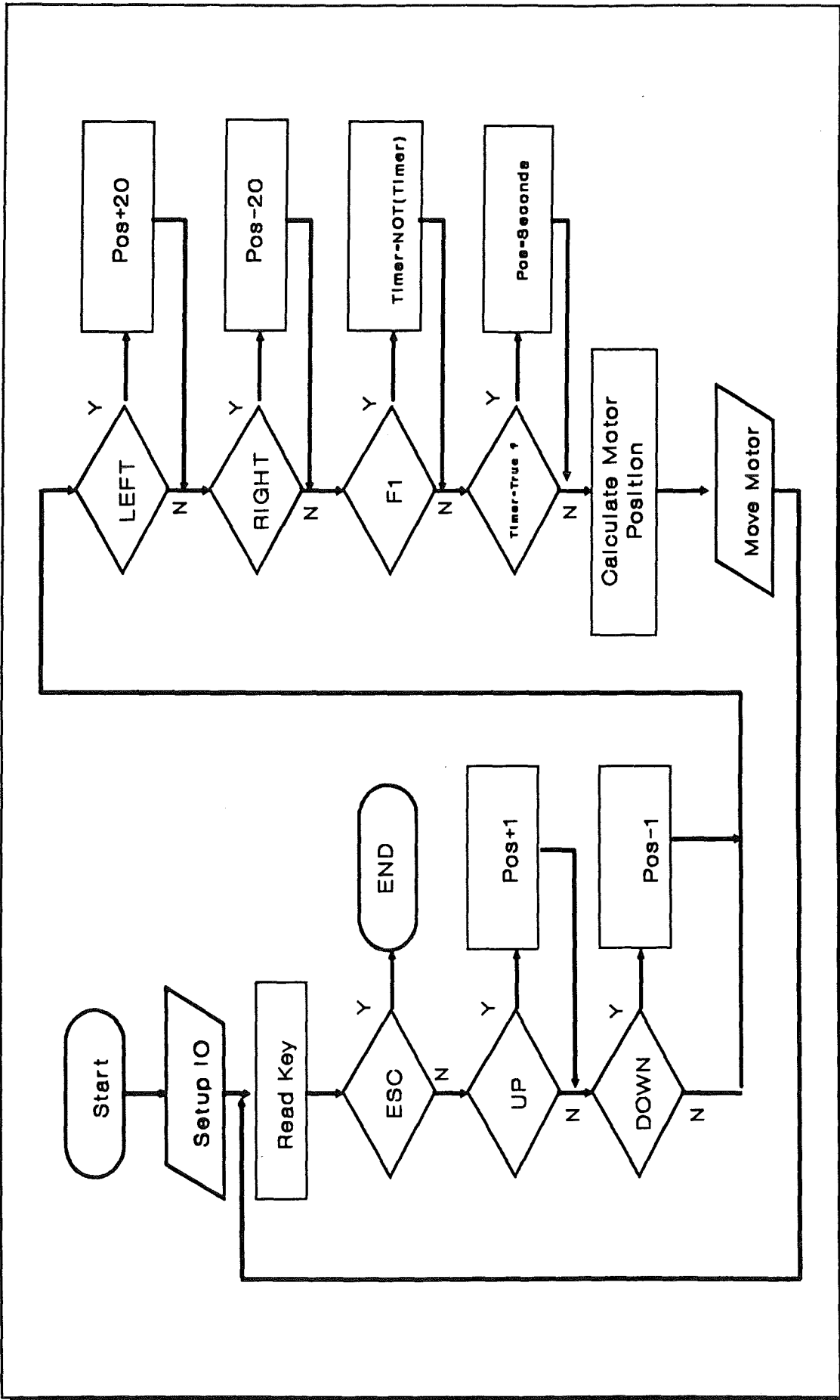
        sample();
        clrscr();
        for(i=1;i<=8;i++)
        {
            data[i]*=4.90015;    /* turn readings into millivolts */
            printf("Channel %+d Value %+03.5f mV\n",i,data[i]);
        }                      /* print voltages on the screen */
        do{}while(kbhit()==0);  /* Wait for key to press*/
        getch();
    }
}

void sample()                  /* This bit does the actual sampling */
{
    int n,d;

    for(n=1;n<=8;n++)
    {
        outportb(base1+0x02,(n*16)+2); /* Select analogue channel */
        outportb(base1+0x02,(n*16)+3); /* Trigger software clock */
        for(d=0;d<2000;d++);          /* Delay to allow settling */
        data[n]=((inportb(base1+0x01) & 0x0F)<<8)+inportb(base1+0x00);

        data[n]-=2048.0;              /* Read value */
    }                                  /* Subtract +FSD/2 to turn binary */
}                                      /* value into bipolar voltage*/

```



Flow chart for motor controller test program

{Program 2}

{ Test program for motor controller to test DC motor control of this simple program allows the motor to be moved in small or large steps, and move at a preset step rate.

The program was used during testing to check that the same position can be reach successively, with no error.}

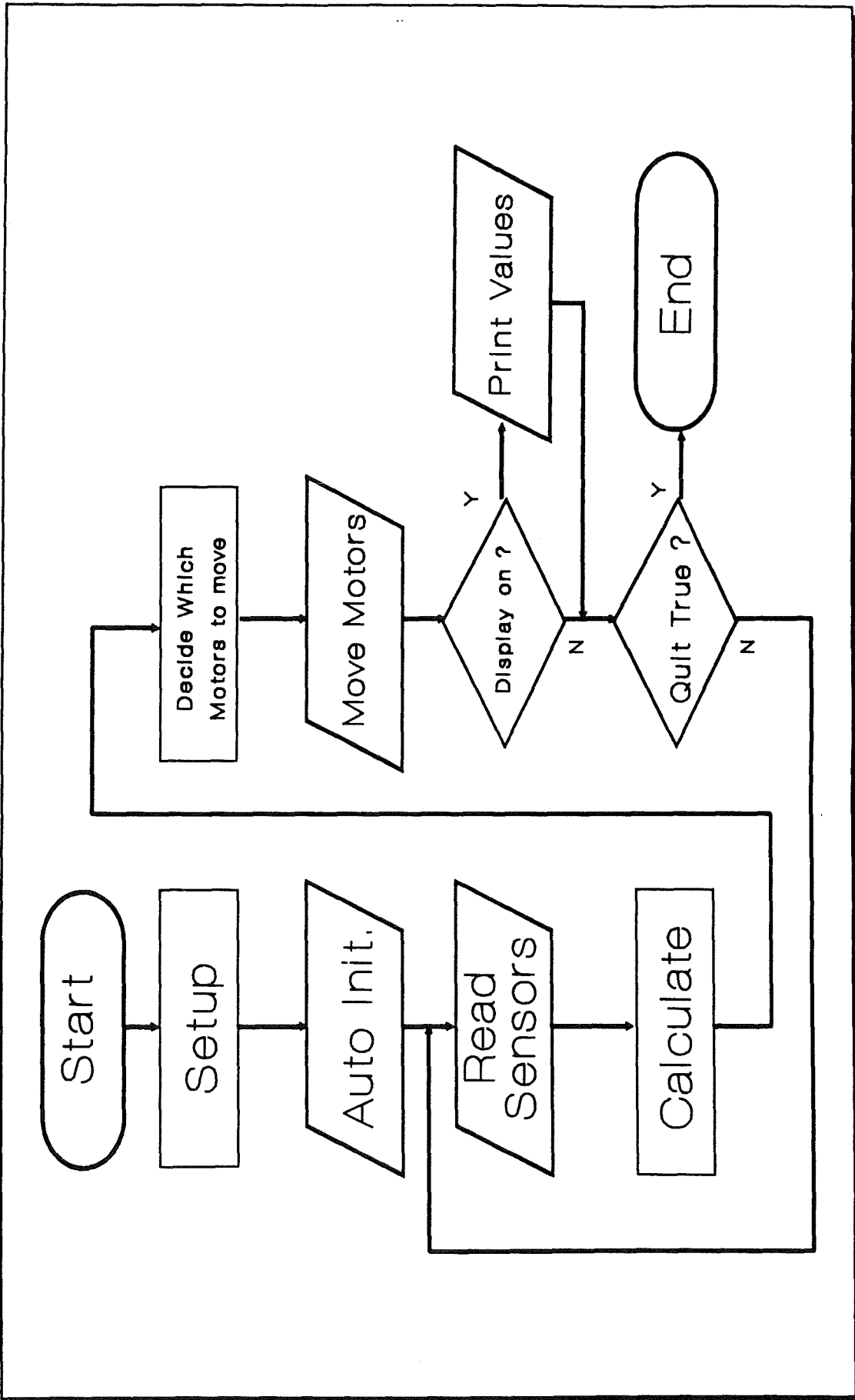
```
program test1;
uses dos,crt;           { Program constants and definitions }
const BA=$700;
      UP=72;
      DOWN=80;
      ESC=59;
      LEFT=75;
      RIGHT=77;
      F2=60;
```

```
var   adval,oldadval : integer;      { Variable definitions }
      inadval : integer;
      key : integer;
      lsb : integer;
      msb : integer;
      time : boolean;
      h,m,s,hs : word;
```

```
begin
  port[BA+$03]:=$92;      { Set up the variables }
  adval:=2000;
  key:=0;
  time:=false;
  while(true) do
  begin
    if (keypressed=true){ If a key is pressed, find out what it is,
                          if it is one of the keys used in the program,
                          ie. UP/DOWN/LEFT/RIGHT Arrow keys
                          for small and large steps,
                          ESC to quit, and F2 to toggle
                          timed steps.
                          }
    then begin
      key:=ord(readkey);
      if key=0 then begin
        key:=ord(readkey);
        case key of
          UP : adval:=adval+1;   {Small Steps}
          DOWN : adval:=adval-1;
          ESC : halt(1);
          LEFT : adval:=adval-20; {Large Steps}
          RIGHT : adval:=adval+20;
          F2 : time:=not time;   {Timer}
        end;
```

```
    end;
    if adval=-1 then adval:=0; {Limit the output values }
    if adval=2249 then adval:=2248;
end;
gettime(h,m,s,hs);
if time=true then adval:=(s*10)+2000;
lsb := (adval SHL 4) and $F0;    {Calculate output values for PC-30A
                                board}
msb := adval SHR 4;
if oldadval<>adval then begin
    writeln('Adval is ',adval,' : Low byte is ',lsb,' :High byte is ',msb);
    if time=true then writeln('Timer is on');
    end;                    {Display the output values
                            on the screen}
oldadval:=adval;
Port[BA+$0D]:=msb;        {Actually send the values
                          to PC-30, and hence the
                          motor controller}

Port[BA+$0C]:=lsb;
end;
end.
```



Flow chart showing implementation of Force/Direction feedback control software

```

/*****/
/*  Program 3 : To demonstrate motor control system reaction  */
/*              to external force.                          */
/*              */
/*  Program is cut down to a minimum to show maximum speed at */
/*  which such a program can run.                             */
/*  Functions removed are : Torque calculation.              */
/*                      Data Display.                      */
/*                      Calculation of magnitude/direction */
/*                      of force and torque vectors.       */
/*  Although of course it does calculate the individual     */
/*  components.                                             */
/*****/
#include <stdio.h>
#include <dos.h>
#include <math.h>

#define base1 0x0700
#define base2 0x0300
#define RD 180/3.1415926

/* Calibration matrix */
double s_matrix[6][9]=

    { { 0.0000,-0.1470, 0.0000, 0.0000, 0.0000, 0.1470, 0.0000, 0.0000,2},
      { 0.0000, 0.0000, 0.0000,-0.1490, 0.0000, 0.0000, 0.0000, 0.1500,2},
      { 0.2940, 0.0000, 0.2810, 0.0000, 0.2550, 0.0000, 0.2570, 0.0000,4},
      { 0.0095, 0.0000, 0.0000, 0.0000,-0.0087, 0.0000, 0.0000, 0.0000,2},
      { 0.0000, 0.0000, 0.0088, 0.0000, 0.0000, 0.0000,-0.0090, 0.0000,2},
      { 0.0000, 0.0220, 0.0000, 0.0146, 0.0000, 0.0150, 0.0000, 0.0140,4} };

double data[8];          /* Function, variable and array definitions */
double force;
int pos[3],incr[3];
double limit[3];
long double s_data[8];

void sample( void );
void calculate( void );
void motor( void );
void initialise( void );

main()
{
    initialise();          /* Set up subroutine */

    while(kbhit()==0)    /* Keep running until keypress */
    {
        sample();        /* read sensors */
    }
}

```

```

calculate();          /* turn voltages into forces */
motor();             /* move manipulator */
}

printf("\n\n          Controller Program Ended.");
}

void initialise()     /* This sets up the motors, and reads the
                    /* quiescent force */
{
int n;
const char direction[3]="XYZ";

clrscr();
printf("\n\n\n          Setting up....\n");
outportb(base1+0x03,0x92);      /* Set up PC-30 */
for(n=0;n<=2;n++)pos[n]=2048;  /* set manipulator to mid position */
sample();                      /* Read quiescent voltages */
calculate();                   /* Calculate force */
printf("          Force limits are :");
for(n=0;n<=2;n++)
{
limit[n]=s_data[n];
printf("          %c %04lf",direction[n],limit[n]);
}

printf("\n\n          Controller Program Running.\n");
}

void sample()
{
int n,d;

for(n=1;n<=8;n++)             /* Usual PC-30 voltage sampling program */
                             /* Reads eight channels */
                             /* and converts them into a bipolar ±10v*/
{
outportb(base1+0x02,(n*16)+2);
outportb(base1+0x02,(n*16)+3);
for(d=0;d<500;d++);
data[n-1]=(double)((inportb(base1+0x01) & 0x0F)<<8)+inportb(base1+0x00);
data[n-1]-=2046;
data[n-1]*=4.8828;
}
}

void calculate()
{
/* This performs a matrix multiplication */
/* between the input data and the calibration */
}

```

```

        /* matrix, the output is the forces in */
        /* each direction */

int row,col;

for(row=0;row<=2;row++)
    s_data[row]=0.0;

for(row=0;row<=2;row++)
    for(col=0;col<=7;col++)
        s_data[row]+=s_matrix[row][col]*data[col]
}

void motor()          /* The motor subroutine actually moves the */
                    /* three motors, depending on the current */
                    /* mode of operation, this particular */
                    /* subroutine waits for the force to change */
                    /* from the quiescent value by about 0.1N */
                    /* and moves the motor a distance proportional */
                    /* to the force. */

{
    int n;
    int msb[3],lsb[3];
    for(n=0;n<=2;n++)
    {
        if((s_data[n]-limit[n]<-0.1)
            ||(s_data[n]>=limit[n]+0.1))incr[n]=(int)(s_data[n]/2)+2048;
        pos[n]=incr[n];
        if(pos[n]>4095)pos[n]=4096; /* Here the comparison with the */
                                /* trigger level (0.1N) note how the */
                                /* quiescent value is subtracted */

        if(pos[n]<1)pos[n]=0;
        msb[n]=(pos[n]>>4)&0xFF;
        lsb[n]=(pos[n]&0x0F)<<4;
    }
    outportb(base1+0xC,(unsigned char)lsb[2]);
                                /* Here the new voltage is sent */
                                /* to the PC-30 */

    outportb(base1+0xD,(unsigned char)msb[2]);
    outportb(base2+0xC,(unsigned char)lsb[1]);
                                /* Note : For an application */
                                /* making use of all three */
                                /* axis a second PC-30 board */
                                /* is used for the third motor*/
                                /* as one PC-30 only has two */
                                /* 12bit D-A channels */

    outportb(base2+0xD,(unsigned char)msb[1]);
    outportb(base1+0x10,(unsigned char)lsb[0]);
    outportb(base1+0x11,(unsigned char)msb[0]);
}

```



**8.2 Peg-in-hole Strategy**

A series of tests were conducted, using the following sizes:

Peg Diameter	4.5mm rounded tip
Hole Diameter	5.525mm 1mm chamfer at 45 degrees
Hole Depth	25mm
Lateral Error	0-1.0mm
Angular Error	0

A metal workpiece with a drilled hole was placed within the robot's working envelope while the robot had the micro-manipulator/force sensor/ gripper and a peg arrangement attached to it as its end effector. On running the program the micro-manipulator was first reset to make sure that the computer knew its location. During this procedure the computer disabled the robot arm. Whilst setting up the computer checks were made for any excessive force on the end-effector, just in case it found itself in a position to collide with its surroundings.

When the set up operation was completed the computer instructed the robot controller to move the end-effector into position. At an appropriate time the micro-manipulator came into action to complete the task.

The way that the micro-manipulator completes the task is known as "the strategy", and several possibilities for this are considered as follows:

**1- Random poking:** When the arm is in the vicinity of the workpiece the manipulator makes random jabbing movements at the surface, in various (x,y) locations within its envelope around the workpiece. If the micro-manipulator succeeds in finding the hole, the Z-axis will be able to extend with no force in the Z-direction, (ie. parallel to the hole axis).

**2- Sweeping:** The micro-manipulator is moved to one corner of the workpiece, and moved in a zig-zag fashion across the piece. The hole is again

found when the force in the Z-direction is zero while the micro-manipulator is able to move in the Z direction.

3- The Spiral: On contact with the work surface the micro-manipulator moves in a spiral of increasing diameter until the hole is found, in a similar manner to method 2.

Strategy 1 will be literally hit and miss, as the random element will determine whether the task is completed, for this reason it was not used.

Strategy 2 assumes the micro-manipulator is not near the hole, in fact the robot should be within a short distance so the micro-manipulator will spend a lot of time searching needlessly. For this reason this strategy was also rejected.

Strategy 3 still searches, but in a more sensible fashion, the circular movement should also aid the insertion. This strategy was selected as the basis for the design of software for performing the peg in hole task.

A method had also to be found of telling the micro-manipulator to start its task, and sensing when the task is complete. The most favourable situation would be that the micro-manipulator remains motionless unless the force in the Z-direction rises to an unacceptable level, indicating that the hole has been missed: at this point the micro-manipulator begins its search for the hole which, once found, it should then try and complete the operation of insertion, still finely adjusting its position to make an accurate fit. The task will be determined complete when the manipulator is able to extend in the Z-direction with no retarding force.

The above method of enabling the micro-manipulator is a very useful one due to the fact that no communication is needed between the robot arm controller and the micro-manipulator controller's computer, the manipulator simply senses the need for adjustment.

The software (listed on page 123), designed to perform the task of inserting a peg into a hole is similar to the previous test software for the micro-manipulator system; basically all that has been changed is the MOTOR subroutine, it now contains the method as described by strategy 3, above. However, an auto-

initialise program, and an error handler were also added here to inform the operator of the progress of the system.

### **8.2.1 Peg-in-hole Program Operation**

The program listing is divided up into 'paragraphs', e.g. the paragraph number 12.2.3 indicates procedure 12, part 2, line 3. This will aid in referencing parts of the program within the program description which follows.

The flow chart on page 122 outlines the program described below.

The first ten parts of the program are the various constants, variables, and functions used to control the program operation.

Part:

- 1- Lists the #include files that will be needed to compile the program.
- 2- Contains the constants that control the PC-30A card within the computer, this includes the IO base addresses, and the offsets of the devices used to control the system, see Appendix {4}, General Description of pc-30A, section 2.10.
- 3- Definitions specific to the configuration of the micro-manipulator, including the way the ports are used to interface the manipulator to the PC-30.
- 4- The value of  $\pi$  (PI) used in the calculation of the spiral path that the micro-manipulator will follow to find the hole.
- 5- Maximum force to exert on the force sensor.
- 6- Depth moved in negative Z-direction at which peg insertion seems complete.
- 8- Force sensor calibration matrix, used to turn the voltages read from the force sensor into forces and torques in the X,Y & Z planes.
- 9- Various global variables used within the program.

- 10- Function prototypes.
- 11- Main program entry point.
  - 11.1- Call to Initialise procedure, this sets up the force sensor, the P-30A interface, and the micro-manipulator.
  - 11.2- The main program loop which continuously calls the three subroutines used to sense the voltages, calculates the forces, and moves the motor within the criteria set up by the strategy. This loop can be exit three ways:
    - i) by pressing a key which shuts down the robot and the manipulator,
    - ii) by excessive force or other error condition, and
    - iii) by completing the task in hand.
  - 11.3- Normal end of the program, only reached by pressing a key.
- 12- The initialise procedure.
  - 12.2.1- The screen is cleared, and a welcome message printed.
  - 12.2.4- The IO port is set up that controls the A-D converters.
  - 12.2.6- Quiescent<sup>5</sup> forces are read to prevent having to spend time re-calibrating the software each time the program is run.
  - 12.3- If the quiescent forces are too large the program stops with an error condition.
  - 12.4.3- The auto-setup program is run and this sets up the micro-manipulator into a known condition.

---

<sup>5</sup> The quiescent force is defined as the force apparent when no physical contact is effecting the force sensor. The major factor causing this is gravity, and the mass of the components attached to the force sensor. The other minor factor is drift within the strain gauges and amplifiers.

12.4.6- A control signal is sent to the robot and the micro-manipulator controller to allow the manipulator to operate and to allow the robot to begin its sequence.

13- The sample routine reads the voltages sent from the strain gauge amplifier: the use of faster *register* variables (13.1) should be noted to help increase the speed of this relatively slow procedure.

13.2- Data is read from each of the 8 A-D channels.

13.2.3- The channel is selected here and the software clock is triggered. Note that the maximum rate at which samples can be taken is 28kHz and that is controlled by the delay set by the SETTLETIME constant; on a 12MHz Intel 80286 computer the constant is about 5.

13.2.6 to 13.2.8 Here the data read is turned into a bipolar voltage, first by subtracting an offset and then by multiplying by a scale factor.

14- The "calculate" subroutine converts the voltages read by the sample subroutine into forces by multiplying them by the calibration matrix. 14.2.6 is where the quiescent forces are subtracted to eliminate any major error.

15- Section 15 controls the motor according to the strategy. Every time it is called it also checks the forces to make sure that they remain within the limit set up within the definitions (15.2).

15.3.1- Tries to keep the Z-axis in the middle.

15.3.3- Tries to keep the force in the Z-direction within a set of limits. If the force is greater than 2N the manipulator is moved from the surface, if the force is less than 1N the micro-manipulator is pushed back onto the surface.

15.4- This part of the program makes the peg move in a spiral of increasing diameter. This is only done if the force at any time has been greater than 2N. If the manipulator reaches the edge of its envelope in the X,Y plane then the operation is deemed to have failed, and the whole system is stopped (15.5.5).

15.4.3- Here the operator is informed of the attempt to correct the robot's

positioning by the micro-manipulator's controlling computer.

15.6- A sudden drop in force to 0, and the existence of a searching operation create the conditions for the micro-manipulator to attempt inserting the peg into the hole itself: the operator is warned of this decision.

16.7- When the peg has been inserted a pre-determined distance, the program deems the operation a success.

15.8- The next three lines of the program send commands to the three axes of the manipulator to move to the new desired position.

16- The auto-setup routine is used to put the micro-manipulator into a known situation. It can happen that the micro-manipulator may still be within the workpiece when the controlling program is run. If this happens then the force setup routine will detect the existence of an undesirable force, and stop the program.

16.2- The Auto-reset is turned on and the motors are requested to move to the full scale position: the robot arm is also switched off.

16.3- While the motors are moving the forces are monitored. At this point the system could be programmed to detect a hardware feedback signal, indicating the micro-manipulator reaching the end of its travel. But it was decided to simplify the program by simply waiting long enough for the operation to be completed. The micro-manipulator controller deals with the rest of the setting-up.

16.4- Once the end position has been reached, the motors are requested to move to their mid positions, the auto setup status is cancelled and, after another delay, the robot arm is put into operation.

17- This procedure simply stops the motors of the micro-manipulator and the robot arm. It is mainly used within error routines in an emergency.

18- During program execution it becomes necessary to provide the operator with information about the status of the program, and inform what may have caused an error situation, this procedure deals with this.

The errors are split up into various categories to allow for expansion. Some, such as the error reports, cause the system to shut down. The error handler is also used to report the program's progress but these errors of course do not cause the program to terminate.

18.1- Explains the severity of the error.

18.2- Prints the actual error message.

18.3- Determines whether the error should cause the program to exit.

19- It is used to send control signals to the robot arm and the micro-manipulator controller, this is done via PIO built into the P-30A interface card.

20- This procedure sends the position request voltage to the controller: as two PC-30A cards are used to control all three axes of the micro-manipulator, an array is set up containing all the addresses needed to send the position to the D-A converters within the PC-30A card.

20.2- This part of the procedure checks that the position request is valid, ie within the working envelope.

21- The subroutine that actually keeps the position requests within the envelope and warns of any request outside the envelope.

### 8.2.1 Appreciation of the software

Operation: The program was run on a 80286 PC with a maths co-processor it was found that, after some optimisation with the delays, a motor update rate of 30 cycles/second could be obtained. This was considered fast enough to cope with the problem.

**8.2.2 Example Screen Output**

The following is a typical computer screen output from the controlling program.

*Program strategy to complete the peg in hole task.*

*Initialising ...*

*Quiescent Forces are :*

*X 0.000000 N*

*Y 0.303000 N*

*Z -1.002300 N*

*Performing Auto-Setup, Please Wait*

*Controller Program Running.*

*Starting Robot...*

*Status Report, Contact assumed with surface*

*Status Report, Inserting peg into hole*

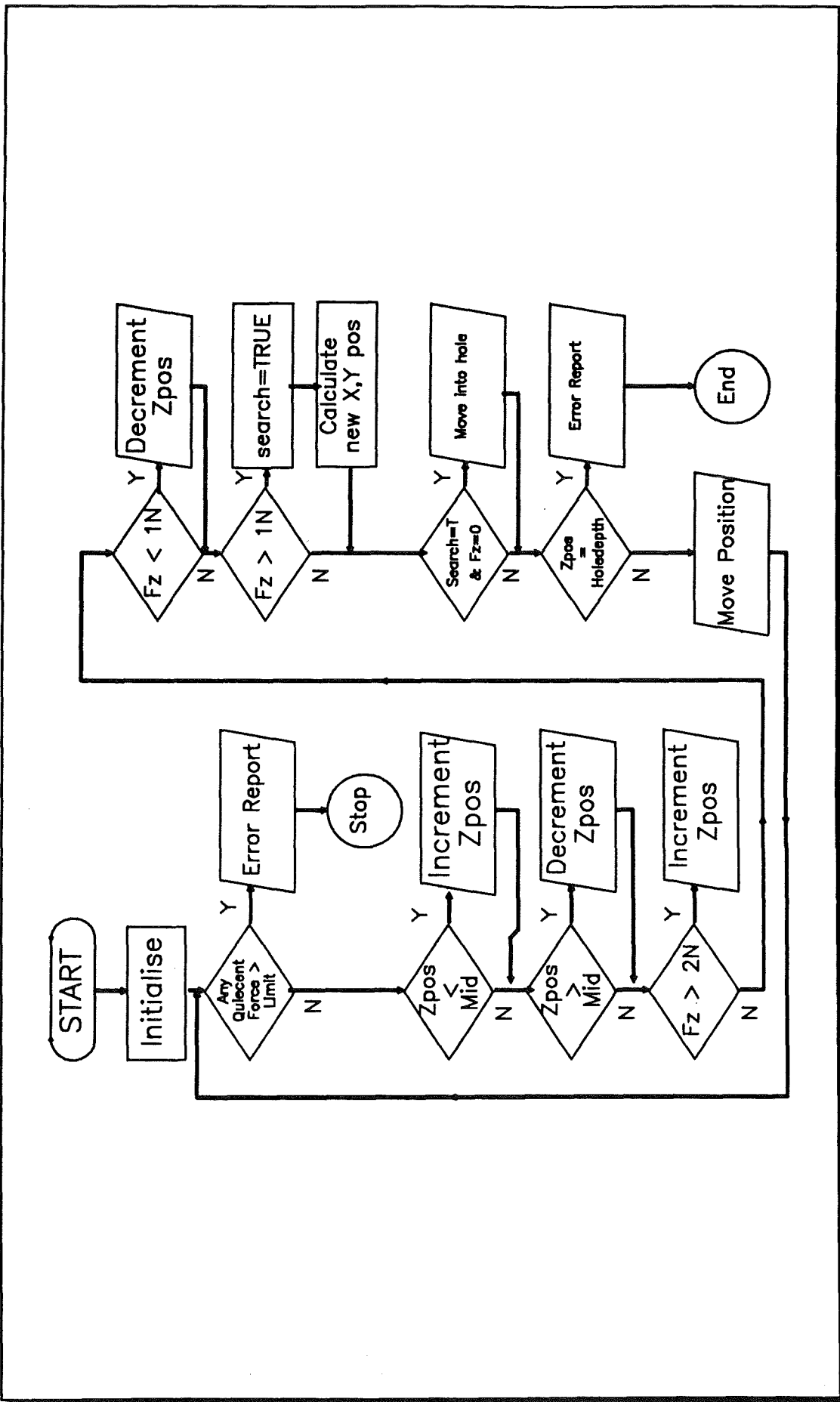
*Success!*

*The Strategy Program Has*

*Decided That The Operation Is Complete*

*Normal Program Termination.*





Peg-in-hole Program Operation Flow Chart.

Peg-in-hole Program Operation Flow Chart.

```

/*****
/*   Program 5 : Peg in hole strategy   */
/*                                     */
/*****
//1
#include <stdio.h>
#include <stdlib.h>
#include <dos.h>
#include <math.h>
#include <conio.h>
//2
// PC-30A Definitions
#define BASE1 0x0700
#define BASE2 0x0300
#define DA12_1L 0x0C
#define DA12_1H 0x0D
#define DA12_2L 0x10
#define DA12_2H 0x11
#define Q11    0x03
#define ADC_CON 0x02
#define AD12L  0x00
#define AD12H  0x01
#define PORTA  0x08
//3
// Micro-Manipulator Controller Definitions
#define FSD 4095
#define MAXAXIS 3
#define MSBX BASE1+DA12_1H
#define LSBX BASE1+DA12_1L
#define MSBY BASE2+DA12_1H
#define LSBY BASE2+DA12_1L
#define MSBZ BASE1+DA12_2H
#define LSBZ BASE2+DA12_2L
#define MON 0x07 // Code for all motors on
#define RON 0x00 // Code for all reset enables on
#define MOFF 0x00 // Code for all motors off
#define ROFF 0x38 // Code for all reset enables off
#define AOFF 0x00 // Code for Robot arm power off
#define AON 0x40 // Code for Robot arm power on
#define WAITTIME 500
#define SETTLETIME 5
//4
// Math Definitions
#define PI 3.1415926

// Force Sensor Definitions
#define MAXFORCE 20
#define SCALEFACTOR 4.8828
//5
// Environment Definitions
#define HOLEDEPTH mid-500

```

```

#define MAXSEARCH mid*2*PI

// Type Definitions
typedef enum {FALSE,TRUE} boolean;

Calibration matrix */
//6
double s_matrix[6][9]=
{ { 0.0000,-0.1470, 0.0000, 0.0000, 0.0000, 0.1470, 0.0000, 0.0000,2},
  { 0.0000, 0.0000, 0.0000,-0.1490, 0.0000, 0.0000, 0.0000, 0.1500,2},
  { 0.2940, 0.0000, 0.2810, 0.0000, 0.2550, 0.0000, 0.2570, 0.0000,4},
  { 0.0095, 0.0000, 0.0000, 0.0000,-0.0087, 0.0000, 0.0000, 0.0000,2},
  { 0.0000, 0.0000, 0.0088, 0.0000, 0.0000, 0.0000,-0.0090, 0.0000,2},
  { 0.0000, 0.0220, 0.0000, 0.0146, 0.0000, 0.0150, 0.0000, 0.0140,4} };
//9
double data[8];
double force;
int pos[MAXAXIS];
double limit[MAXAXIS]={50,50,50};
double quies[MAXAXIS];
long double s_data[8];
enum axis {x,y,z};
int mid = (int)FSD/2;
int thita = 0;
boolean search=FALSE;
float searcharea=1/(2*PI);
//10
void initialise( void );
void auto_setup( void );
void sample( void );
void calculate( void );
void motor( void );
void error( int );
void stopmotor( void );
void control( unsigned char);
void conmotor( int, int);
int checkmax( int );
//11
void main()
{
//11.1
initialise();
//11.2
while(!kbhit())
{
    sample();
    calculate();
    motor();
}
//11.3

```

```

stopmotor();
printf("\n\n          Controller Program Ended Normally\n\n");
}

//12
void initialise(void)
{
//12.1
int n;
const char direction[MAXAXIS]="XYZ";
//12.2
clrscr();
printf("\n\n          Program strategy to complete the peg in hole task.\n");
printf("\n\n\n\n          Initialising ...\n\n");
outportb(BASE1+Q11,0x92);
for(n=x;n<=z;n++)pos[n]=mid;
sample();
calculate();
printf("          Quiescent Forces are :\n\n");
for(n=x;n<=z;n++)
{
    quies[n]=s_data[n];
    printf("          %c %04lf N\n",direction[n],limit[n]);
}

//12.3
for(n=x;n<=z;n++)if(abs(quies[n])>MAXFORCE)
{
    stopmotor();
    error(12);
}
12.4
printf("\n");
printf("          Performing Auto-Setup, Please Wait \n\n");
auto_setup();
printf("\n\n          Controller Program Running.\n\n");
printf("          Starting Robot ..... \n");
control(ROFF|MON|AON);
}

//13
void sample()
{
//13.1
register int n,d;
//13.2
for(n=1;n<=8;n++)
{
    outportb(BASE1+ADC_CON,(n*16)+2);
    outportb(BASE1+ADC_CON,(n*16)+3);
}

```

```

        for(d=0;d<SETTLETIME;d++);
        data[n-1]=(double)((inportb(BASE1+AD12H) &
0x0F)<<8)+inportb(BASE1+AD12L);
        data[n-1]-=2046;
        data[n-1]*=SCALEFACTOR;
    }
}

//14
void calculate()
{
    register int row,col;

    for(row=x;row<=z;row++)
    {
        s_data[row]=0.0;
        for(col=0;col<=7;col++)
            s_data[row]+=s_matrix[row][col]*data[col];
        s_data[row]=s_data[row]-quies[row];
    }
}

//15
void motor()
{
//15.1
int n,r;
//15.2
    for(n=x;n<=z;n++)if(abs(s_data[n])>MAXFORCE)
    {
        stopmotor();
        error(11);
    }
//15.3
    if(pos[z]>mid)pos[z]--;
    if(pos[z]<mid)pos[z]++;
    if(s_data[z]>2)pos[z]+=10;
    if(s_data[z]<1)pos[z]-=10;
//15.4
    if((s_data[z]>1)||((search==TRUE))
    {
        if(search==FALSE)error(40);
        search=TRUE;
        r=searcharea*thita++;
//15.5
        pos[x]=mid+(r*cos(thita));
        pos[y]=mid+(r*sin(thita));
        if(thita > MAXSEARCH)
        {
            stopmotor();
            error(0);

```

```

    }
}
//15.6
if((s_data[z]==0)&&(search==TRUE))
{
    error(41);
    pos[z]+=50;          /* Extend into hole */
}
//15.7
if(pos[z]<HOLEDEPTH)
{
    stopmotor();
    search=FALSE;
    error(30);
}
//15.8
conmotor(x,pos[x]);
conmotor(y,pos[y]);
conmotor(z,pos[z]);
}
//16
void auto_setup()
{
//16.1
int s,m;
//16.2
control(ROFF|MOFF|AOFF);
for(m=x;m<=z;m++)conmotor(m,FSD); /* move motor n to full scale position*/
control(ROFF|MON|AOFF);
//16.3
for(m=0;m<WAITTIME;m++)          /* Wait while monitoring forces */
{
    sample();
    calculate();
    for(s=x;s<=z;s++)if(abs(s_data[s])>1)
    {
        stopmotor();
        error(10);
    }
}
stopmotor();
//16.4
for(m=x;m<=z;m++)conmotor(m,mid);
control(ROFF|AOFF|MON);
sleep(10);
}

//17
void stopmotor()
{
control(MOFF|ROFF|AOFF);

```

```

}
//18
void error(int errno)
{
//18.1
printf("\n\n");
if(errno<10)printf("      Warning \n\n");
else if(errno<20)printf("      Possible Fatal Error \n\n");
else if(errno<30)printf("      Unknown Error \n\n");
else if(errno<40)printf("      Success !\n\n");
else if(errno<50)printf("      Status Report, ");
//18.2
switch(errno)
{
    case 0 :
        printf("      Hole Not Found \n");
        printf("      Search Algorithm Failure\n\n");
        break;
    case 10 :
        printf("      Collision During Auto-Setup\n");
        printf("      Stop Robot \n");
        printf("      And Check system.\n\n");
        break;
    case 11 :
        printf("      Force Limit Exceeded\n");
        printf("      Stop Robot \n");
        break;
    case 12 :
        printf("      Force Limit Exceeded\n");
        printf("      Quiescent Force Too Large\n");
        printf("      Check Full System\n");
        break;
    case 30 :
        printf("      The Strategy Program Has \n");
        printf("      Decided That The Operation Is Complete \n");
        break;
    case 40 :
        printf("\n      Contact assumed with surface \n");
        break;
    case 41 :
        printf("      Inserting peg into hole\n");
        break;
    case 42 :
        printf("      Axis limit requested\n");
        break;
    default : errno=30;
        break;
}
//18.3
if(errno<10)printf("      Program Failed \n");
else if(errno<20)printf("      Abnormal Program Termination\n");

```

```
else if(erno<30)printf(" Something Strange is Happening ???\n");
else if(erno<40)printf(" Normal Program Termination\n");
if(erno<40)exit(erno);
}
//19
void control(unsigned char cbyte)
{
    outportb(BASE1+PORTA,cbyte);
}
//20
void conmotor( int motor, int pos )
{
//20.1
    static int marray[2][MAXAXIS]={ {MSBX,MSBY,MSBZ},
                                    {LSBX,LSBY,LSBZ}};

    char msb,lsb;
//20.2
    pos=checkmax(motor);
    msb=(pos>>4)&0xFF;
    lsb=(pos&0x0F)<<4;
    outportb(marray[0][motor],msb);
    outportb(marray[1][motor],lsb);
}
//21
checkmax(int n)
{
    pos[n]=(pos[n]<0)?0:pos[n];
    pos[n]=(pos[n]>FSD)?FSD:pos[n];
    if(pos[n]==0||pos[n]==FSD)error(42);
    return pos[n];
}
```



### 8.3 Path Following Strategy

The robot was programmed to move along the geometric path (see Figures 4.22 & 4.23), through the complete panel by 53 points (these points are marked by red colour as seen on Figure 4.23). When the robot is instructed to move from one point to the next, the actual path would be a straight line due to the point-to-point movement nature of robot. It is the job of the micro-manipulator to 'fill in' the gaps between these points. This means that it is the micro-manipulator which conducts the end-effector to follow a curved path while the robot attempts to cross the chords between points (see Figure 4.21).

Since the setup procedure is the same as that already described for the peg in hole strategy it will not be repeated again here.

As the micro-manipulator must "sense" its way around the workpiece, the number of solutions are limited because the micro-manipulator must stay in contact with it all the time. It was therefore decided that the only way of doing this was to keep the force in the Y-direction constant (ie perpendicular to horizontal), hence forcing the end effector to keep contact. The Z-direction was not used, so it is simply kept in the middle, the X-direction was used to pass information to the Y-direction about the shape of the surface ahead. Thus if the X-axis has to move too far then the Y-axis is also moved, this enables the micro-manipulator to deal with vertical surfaces.

The computer determines that the operation should start by an increase in force in the Y-direction, and it should end when the force in all directions is 0; i.e. the micro-manipulator is no longer in contact.

Since effectively all that has been changed from the peg-in-hole program is the motor subroutine, it was not necessary to include the whole listing again, nor produce again here a description of the whole program.

The modified listing for the motor subroutine is presented here only, in the same way as the peg-in-hole program, the numbers meaning *subroutine.section.line*.

15- The Path following motor subroutine.

15.2- Check that the forces are within the limits.

15.3- Try and keep the X,Y,and Z-axes in the middle.

Note: The Z-axis is not used in the path following strategy, the

Z-axis is only kept in the middle to counteract external force.

Y-axis is only kept in the middle if not in search mode, else Y motor is tended to the negative Y-direction.

15.4- The X-axis is kept within a 1N force range.

15.5- If the force in the Y-direction is greater than 1N, then the search mode is entered, the manipulator will try and keep the Y-axis to the surface with a 1N force.

15.6- If the X-axis has to move too far, the Y-axis is moved instead, this is to make the micro-manipulator move over objects, whilst keeping contact with the surface.

15.7- The system reports the end of the process when the axes have no external force acting on them, and the Y-axis has moved as far as it can towards the workpiece, i.e. the end effector has moved off the other end of the workpiece.

16.8- Commands are sent to the motor at this point.

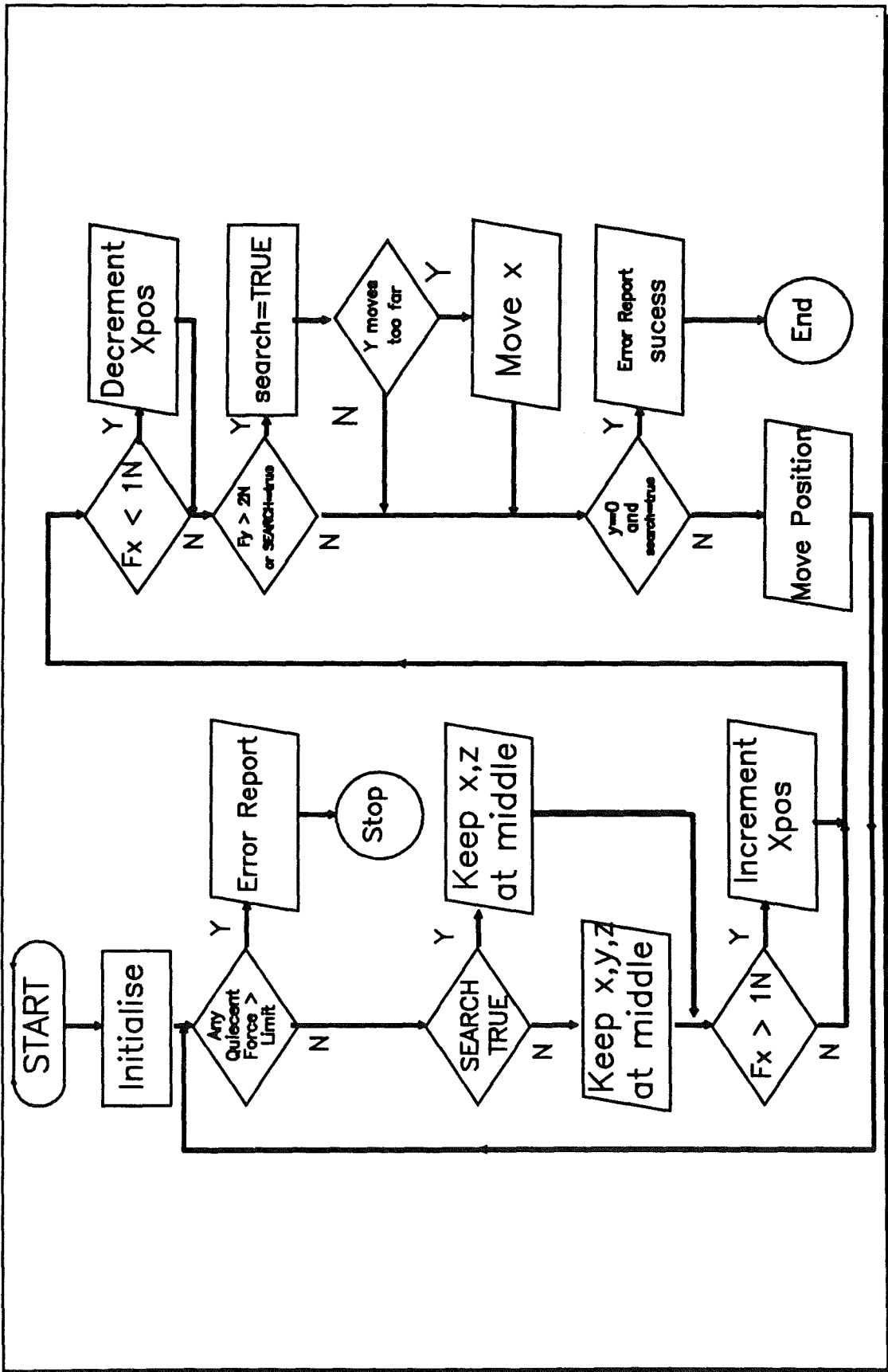


Figure 8.2, Flow chart, outlining the operation of the path following strategy.

Figure 8.2, Flow chart, outlining the operation of the path following strategy.

```

/*****/
/*  Program 6 : Path Following strategy          */
/*                                          */
/*****/
/*The following program segment contains the changes to the peg in*/
/*hole program, this mainly concerns the motor() subroutine that */
/*performs the required task.          */
/*Revised initialise() and error() handler are included, though */
/*all that has changed in the latter two are the messages.      */
//12
void initialise(void)
{
//12.1
int n;
const char direction[MAXAXIS]="XYZ";
//12.2
clrscr();
printf("\n\n      Program strategy to complete path following task.\n");
printf("\n\n\n\n      Initialising ...\n\n");
outportb(BASE1+Q11,0x92);
for(n=x;n<=z;n++)pos[n]=mid;
sample();
calculate();
printf("          Quiescent Forces are :\n\n");
for(n=x;n<=z;n++)
{
    quies[n]=s_data[n];
    printf("          %c %04lf N\n",direction[n],limit[n]);
}

//12.3
for(n=x;n<=z;n++)if(abs(quies[n])>MAXFORCE)
{
    stopmotor();
    error(12);
}
//12.4
printf("\n");
printf("          Performing Auto-Setup, Please Wait \n\n");
auto_setup();
printf("\n\n      Controller Program Running.\n\n");
printf("          Starting Robot .....");
control(ROFF|MON|AON);
}

//15
void motor()
{
//15.1

```

```

    int n;
    double sum=0;
//15.2
    for(n=x;n<=z;n++)if(abs(s_data[n])>MAXFORCE)
    {
        stopmotor();
        error(11);
    }

    /* x,y,z axis tend to mid position */
//15.3
    for(n=x;n<=z;n++)
    {
        /* only affect y if not in search mode */
        if(!(search&& n==y))
        {
            if(pos[n]>mid)pos[n]--;
            if(pos[n]<mid)pos[n]++;
        }
    }

    /* 1N limit in x direction */
//15.4
    if(s_data[x]>1)pos[x]+=10;
    if(s_data[x]<-1)pos[x]-=10;
//15.5

    /* A net 1N force is kept on y axis */
    if((s_data[y]>1)|| (search==TRUE))
    {
        if(search==FALSE)error(40);
        search=TRUE;
        /* y tends to -FSD in y axis.*/
        if(pos[y]>0)pos[y]--;
        if(abs(s_data[y])>2)pos[y]+=10;
        if((s_data[y]<1)&&(s_data[y]>=0))pos[y]-=10;

        /* If x direction moves to far, try moving in the
           y direction
//15.6
        */
        if(abs(mid-pos[x])>20)pos[y]++;

    }

    /* End is assumed when no force is on any axis
       and y is at -FSD
    */
//15.7
    for(n=x;n<=z;n++)sum+=s_data[n];
    if(sum==0&&pos[y]==0)

```

```

    {
        stopmotor();
        search=FALSE;
        error(30);
    }
//15.8
    conmotor(x,pos[x]);
    conmotor(y,pos[y]);
    conmotor(z,pos[z]);
}

//18
void error(int erno)
{
//18.1
    printf("\n\n");
    if(erno<10)printf("        Warning \n\n");
    else if(erno<20)printf("        Possible Fatal Error \n\n");
    else if(erno<30)printf("        Unknown Error \n\n");
    else if(erno<40)printf("        Success !\n\n");
    else if(erno<50)printf("        Status Report, ");
//18.2
    switch(erno)
    {
        case 10 :
            printf("        Collision During Auto-Setup\n");
            printf("        Stop Robot \n");
            printf("        And Check system.\n\n");
            break;
        case 11 :
            printf("        Force Limit Exceeded\n");
            printf("        Stop Robot \n");
            break;
        case 12 :
            printf("        Force Limit Exceeded\n");
            printf("        Quiescent Force Too Large\n");
            printf("        Check Full System\n");
            break;
        case 30 :
            printf("        The Strategy Program Has \n");
            printf("        Decided That The Operation Is Complete \n");
            break;
        case 40 :
            printf("\n        Contact assumed with surface \n");
            break;
        case 42 :
            printf("        Axis limit requested\n");
            break;
        default : erno=30;
            break;
    }
}

```

```
//18.3
if(erno<10)printf("      Program Failed \n");
else if(erno<20)printf("  Abnormal Program Termination\n");
else if(erno<30)printf(" Something Strange is Happening ???\n");
else if(erno<40)printf("  Normal Program Termination\n");
if(erno<40)exit(erno);
}
```

### 8.3.2 Example Screen Output

Example output of the motor control program performing the path following strategy.

*Program strategy to complete path following task.*

*Initialising ...*

*Quiescent Forces are :*

*X 0.002300 N*

*Y -1.08200 N*

*Z 0.002300 N*

*Performing Auto-Setup, Please Wait*

*Controller Program Running.*

*Starting Robot...*

*Status Report, Contact assumed with surface*

*Status Report, Axis limit requested*

*Status Report, Axis limit requested*

*Success!!*

*The Strategy Program Has*

*Decided That The Operation Is Complete*

*Normal Program Termination.*

# ***CHAPTER 9***

## **Discussions & Conclusions**



### 9.0 Discussions & Conclusions.

The current design of the motor control circuit and apparatus was proved, and seen to work well in both the applications of path following and of peg in hole.

With reference to the work detailed in this document these are the immediate improvements that could now be made to the controlling system:

1: In the design the movement is relative, that is to say the computer can be sure that the robot has moved a certain distance in a particular direction, but it cannot be completely sure that it knows where the micro-manipulator mechanism was to start with. This required a "setup" routine, and the necessary hardware to operate it. The method used here was to put the micro-manipulator mechanism through a routine that made sure it was in a particular known place to start with: a DC control design which resulted in a complicated initial system design. An alternative method would be to use an absolute position sensor instead so that no set-up is needed, a reference being taken from the absolute position sensor. A suitable device for this would be a LDVT (Linear Differential Voltage Transformer) but using an absolute position encoding with LDVTs would not make the system any less complicated.

2: The time needed for the mechanism to cover a particular distance could be further improved, in both designs, but the only way to solve this problem is to use even more powerful motors with smaller gear ratios, and more accurate shaft encoders, but this could add more mass to the end effector.

3: The calculation time, to convert the voltages from the force sensors into the parameters of force and direction, can only be improved by using a more powerful computer.

This investigative work concluded that the above problems, (1) to (3), could easily be solved, at a price, when producing a future commercial system. The amount of software that can be written for such a device is limitless, as each new application would benefit from dedicated software, streamlined to fit its purpose.

The speed of an insertion operation will depend on the size of the increment in the downwards direction, and this in turn will depend on the characteristic sizes of that assembly, and the levels of force that can be safely tolerated. Using force feedback, it is just as easy to safely insert a needle sized part, as it is to insert a pen sized part; it is only the acceptable side forces before deformation occurs that will alter the speed of the insertion operation.

The latter could be achieved within the software, simply by changing the lines performing the position calculation within the program to read similar to this:

...

```
if(s_data[n]>2)pos[n]-=(int)abs(s_data[n]*Q[n]);
```

...

Where Q is some scale factor particular to the axis and even maybe, the application. The other method of improving speed is by making the speed of the motors proportional to the size of the error signal, this can be done by controlling the current available to the motors with the error signal magnitude.

Of course the overall speed of the system is set by the controlling computer, and it is no good moving the axes faster if the computer can not calculate and correct for the changes in force that can happen between each sample taken.

In turn, the speed of the system must be such that it can keep up with the expected changes caused by the robot environment in which the experimental rig is being used.

The speed of the system could possibly be improved by studying the relation between the size of the position increments and the magnitude of the force using Hook's law principle. This means that the magnitude of the force is the result of the amount of strain energy stored in the material. A small reverse displacement is enough to release some of the strain energy and hence reduce the contact force significantly. This depends on the stiffness of material on which the force is applied. It is clear that a large force suggests a large displacement, so a large force should cause a large step to be taken as "apparent" position (see page 82, section 7.3.1).

signal can also improve the transient behaviour of the motors, ie reducing oscillation and overshoot, both being undesirable effects. Although after a point this slows down the system, especially if it has to perform a lot of small movements.

The design of the micro-manipulator, utilising sliding telescopic tubes, in theory allows acceptable engineering tolerances of  $\pm 0.05$  mm. But in practice, the manufacture of the prototype resulted in unallowable tolerances between parts. However, this limitation was greatly improved by using shims between moving parts to minimise tolerances and the undesirable backlash effect in the system. An excessive use of the micro-manipulator will result in wear (as it is made of an Aluminium alloy material) which will introduce a further inaccuracy into the system.

The original design mass limitation of 1 kg for the micro-manipulator was exceeded by 1.3 kg due to the weight of 3 DC motors. In addition, the mass of the force sensor (0.35 kg), including the overload protection device resulted in a dead load of 1.65 kg. This is approximately 70% of the payload of the PUMA 560 robot.

By conducting a laser measurement test experiment on the PUMA 560 robot the accuracy was found to be approximately  $\pm 3.3$  mm with a repeatability of approximately 0.6 mm.

The reasons for the limitations in accuracy and in repeatability for the macro-robot were found to be mainly twofold:-

- (i) Beam deflections and,
- (ii) Measurement factor errors (for more details please refer to page 23)

Composite carbon fibre links are probably the best means of restricting the beam deflections. It is also possible to increase the second moment of area. ie the bigger and the thicker sections of the links the greater the rigidity but this will restrict the dynamic performance of the macro-robot in terms of speed and acceleration.

The measurement factor errors can be improved by the use of more accurate encoders and drive mechanisms manufactured to closer tolerances but this would result in significant cost penalties.

The techniques developed here in this work have applications which can be used in industry, particularly where currently no satisfactory robotic solutions are available for certain process. e.g. the robotic deboning and cutting of meat or in the deburring of metal products.

The hardware developed can be used to correct for any deflection errors caused by the excessive forces exerted between the tool and the workpiece, hence maintaining a specified range of force within a limit.

### 9.1 Recommendation for Future Work

The specifications of tolerable force limits have been quite arbitrary in this work and were selected to demonstrate specific features of the control algorithm. The actual force levels would be determined by such factors as the shape and variation of the path and in assembly as the part sizes vary. One area in which this work could be extended would be to devise a method of determining just what the tolerable force levels should be.

The speed of the auto-setup system could be vastly improved by including some feedback in the form of a wire sending the signal from the auto-setup circuit within the counter to the computer, this would allow the computer to detect when the axis has reached the full scale position, rather than waiting a preset time.

In a situation where the micro-manipulator actually loses contact with the work surface (the time for which this occurs depends on the speed of the system and the intelligence of the software) a significant gap could be a problem. This effect may involve any distance depending on the shape of the contours of the workpiece and ways of preventing this effect should be sought.

Another effect seen with this system is when the robot completes the path without making contact with the work at all. The only two solutions which may solve this problem: one is to employ robot vision of some kind and the other,

far more simply and cheaply, is to open better communication between the robot and the computer controlling the micro-manipulator. These solutions should be further investigated.

Another area of possible improvement in future work would be in the area of extensions to the hardware and software.

The control loop can be speeded up by faster sampling and computation, or by the use of a lighter micro-manipulator and a force sensor with less inertia and faster acceleration. Nevertheless, data transfer between the microprocessor and robot environment could be made more rapid by adopting Direct Access Memory (DAM). This approach involves a programmable device, such as the Intel 8237, taking over control of the address, data and control buses, to effect direct transfer of data into memory. As with the other methods discussed, the main processor is thereby freed of workload.

Other methods of speeding up data transfer may involve using faster processors. Sticking with the Intel family, such a progression would be made by employing 80386 or 80486.

However, the modularity of the application at hand may lend itself to a parallel system approach, with a number of different input signals being monitored and a number of output signals being supplied and compared with data in the real world. The system could be split up into a number of tasks, all to be run in parallel. If transputers were used, these tasks could be shared out between a number of transputers within a network. The tasks would then be run independently, with each transputer receiving and supplying information to its neighbours, via communication links, as and when required.

The considerable computing power that such an approach would bring would result in a fast and sophisticated system. The nature of application would need to justify the cost incurred in designing and implementing such a system, with its estimated performance being compared to the approaches discussed above.

Artificial intelligence can be incorporated in the system to achieve economy of time.

An area for which the system can further be developed is in simulated medical application e.g. in the field of physiotherapy.

Technology is also improving regarding magnets, and hence motors, in the near future this may mean lighter, more powerful motors, resulting less mass of the end effector. With current technology, computer intelligence is merely a buzz word. Artificial intelligence is slow, and requires large complex computers, but in the future there may be computers and languages that can be artificially intelligent.

Hence, looking even further into the future, improvements are likely to result from creating intelligent controlling software which would make the system more effective. Not only would the end effector feel its surroundings, it would make predictions based on the knowledge of the sort of environment it is working in.

## References

1. Whitney Daniel E, "*Historical Perspectives & State of the Art in Robot Force Control*", Proceedings of the IEEE Comp Soc Press, 1985, pp 262-268.
2. H. Makino, N. Furuya, "*Selective Compliance Assembly Robot Arm*", Procs 1st Conf Ass. Automation, pp 77-86, 1980.
3. J. D. Lane, "*Applications of the Remote Centre Compliance to Assembly Operations*", SME Technical paper 1979.
4. D. S. Seltzer, "*Tactile Sensory Feedback For Difficult Robot Tasks*", Robotics 6th Conference Procs. March 1982
5. M. M. Karelin, "*The Automatic Alignment of Parts for Automatic Assembly*", Russian Engineering Journal, Vol XLVII, N° 9, pp 73-76, 1967.
6. Stepourjine & Rouget "*Automatic Insertion Module for Light Robotics*", from "developments in Robotics 1983", Ed. B. Rooks
7. Y. A. Yakimovich, etal, "*Automatic Assembly of components by the Jet Method*", Russian Engineering Journal, Vol L, No 6, pp 68-63, 1970.
8. C. J. Bland, "*Peg-Hole Assembly: Literature Survey*", UWIST, int. report, Cardiff Tech note ARC10, 1986.
9. P. J. Drazan, S. H. Hopkins, C. J. Bland, "*Autonomous assembly Devices and related control strategies*", IEE Control Engineering series, no 28, pp 121-8, 1985.
10. C. J. Bland, P.J. Drazan, S. H. Hopkins, "Autonomous Devices for Automated Assembly", I Mech E Pub C 361/86, pp 111-115, 1986.
11. P. J. Drazan, S. H. Hopkins, C. J. Bland, "*Insertion & Sensing*", I Prod E Automated assembly/Robotics working party seminar on "Techniques in Automated Assembling", June 85.
12. P. J. Drazan, S. H. Hopkins, "*Semi-Autonomous System for Automatic Assembly*", Annals of CIRP, 33/1, 1984.

13. P. J. Drazan, S. H. Hopkins, *"Adaptable Handling for Automatic assembly Colloquium in Robot Sensors"*, IEEE London, Feb 1984.
14. B. Torr, A. G. Cartwright, G. A. Parker, *"A Hole to Hole Alignment Sensor for assembly Tasks"*, Procs I Mech E, paper No C459/84, pp 35-41, 1984.
15. V. M. Stokic, D. Hristic, *"Implementation of Force Feedback in Manipulation Robots"*, Int Journal Robotics Research, Vol 5, No 1, pp 66-76.
16. H. Van Brussel, J. Simons, *"The Adaptive Compliance Concept and its Use for Active Force Feedback"*, Proc 9th Int Sys Ind Robots, 1979.
17. H. Van Brussel, J. Simons, *"Adaptive Assembly"*, Procs 4th British Robot Conf, pp 95-106, 1981.
18. H. Van Brussel, J. Simons, J. Peters *"Robot Assembly by Active Force Feedback Accommodation"*, CIRP, Vol 28/1, pp 397-401, 1979.
19. J. Simons, *"Force Feedback in Robotic Assembly Using an Active Wrist with Adaptable Compliance"*, PhD Thesis, Leuven University, Belgium, 1980.
20. M. T. Mason, *"Compliant Motion"*, In *"Robot Motion, Planning And Control"*, ed Brady, Hollerbach, Johnson, Loranzo-Perez, MIT Press, pp 305-322, 1983.
21. C. S. Lee, R. H. Smith, *"Force Feedback Control Pattern Analysis Technique"*, Proc American Control Conf, Vol 1, pp 39-44, 1984.
22. M. C. Raibert, J. J. Craig, *"Hybrid Position/ Force Control of Manipulators"*, Journal Dyn Sys, Meas & Control, Vol 102, 1981.
23. D. E. Whitney, *"Force Feedback Control of Manipulator Fine Motion"*, Trans ASME, Ser. G, Journal dynamic Sys, Meas, & Control, Vol 99, No 2, pp 91-97, 1977.
24. A. A. Goldenberg, A. Bazerghi, *"A Preview Approach to Force Control of Robot Manipulators"*, Mechanism & Machine Theory, Vol 20, No 5, pp 449-464, 1985.
25. J. Merlet, *"Trends in Force Feedback Command For Robotic Assembly"*, Proc 8th BRA, ed Collins, pp 153-162, 1985.



26. V. Gurfinkel, E. Devyanin, A. Lenskii, "*Force Feedback in manipulator Control System*", *Mechanics of Solids*, Vol 19, No 6, pp 52-56.
27. G. P. Smith, "*Implementation of a Simple, Universal, Low Cost, Sensory Feedback System for Industrial Robots*", *Conf Intelligent Autonomous Systems*, Amsterdam, 1986.
28. H. Inoue, "*Force Feedback in Precise Assembly Tasks*", from "*Artificial Intelligence-An MIT Perspective*", ed Winston & Brown, pp 223-243.
29. P. J. Drazan, S. H. Hopkins, C. J. Bland, "*Autonomous Assembly Devices and Related Control Strategies*", *IEE Control Eng Series No 28, Robots and Automated Manufacture*, Paper 9, 1985.
30. K. Asakawa, F. Akiya, F. Tabata, "*A Variable Compliance Device and its Application for Automatic Assembly*", *SME Tech Paper MS83-790*.
31. K. Asakawa, "*Robot Assembly of Precision Parts using Tactile sensors*", *Advanced Robotics*, No 1, Vol 1, pp 56-69, 1986.
32. R. H. Taylor, R. Hollis, M. A. Lavin, "*Precise Manipulation with End-Point Sensing*", *IBM J. Res Devel.*, Vol 29, No 4, July 85.
33. A. Sharon, D. Haardt, "*Enhancement of Robot Accuracy Using End-Point Feedback, and a macro-micro-manipulator system*", *American Conf Control Proc*, San Diego, Calif, pp 1836-1842, 1984.
34. T. Goto, K. Takesayu, T. Inoyama, "*Control Algorithm for Precision Insert Robot*", *IEEE Trans Syst Man & Cybernetics*, Vol SMC-10, pp 19-25, Jan 1980.
35. G. J. Vachtsevanos, K. Davey, K. M. Lee, "*Development of a Novel Robotic Manipulator*", *Control Sys Mag*, Part 3, Vol 7, pp 9-15, 1987.
36. H. Van Brussel, H. Belien, H. Thielemans, "*Force Sensing for Advanced Robot Control*", *Proceeding of the 5th International Conference on Robot Vision & Sensory Controls*, pp 55-68, 1985.
37. B. K. P. Horn, "*Identification and Composition of Robot Kinematic Errors*", *Technical Notes*, 1983.

38. R. Paul, "Robot Manipulators: Mathematics, Programming And Control", MIT Press, 1981.
39. D. E. Whitney, C. A. Lozinski And J. M. Rourke, "*Industrial Robot: Calibration Method And Results*", Proceedings of the 1984 Int Comp in Eng conf Las Vegas, pp 92-100, August 1984.
40. A. Zalucky, D. Hardt, "*Active Control of Robot Structure Deflections*", ASME Journal of Dynamic System Measurement and Control, Vol 106, No 1, pp 63-69, March 1984.
41. S. K. Birla, "*Sensor for Adaptive Control and Machine Diagnostic*", Machine Tool Task Force Study-Machine Tool Controls, Vol 4, ed G Sutton, SME, Dearborn, MI, pp 7.12.1-7.12.70, October 1980.
42. J. Craig, "*Introduction to Robotics: Mechanics and Control*", Addison Wesley, Chapter 3, 1986.
43. Z. C. Lai, C. P. Day, C. H. Menq, "*A Solution Algorithm of the Inverse Kinematic Problem for Robots With Axial Offset Wrist*", IEEE Robotics and Automation Conference, Raleigh, NC, March 1987.
44. S. Drake, "*Using Compliance Instead of Sensory Feedback for High Speed Robot Assembly*", 6th North American Metal Working Research Conference, April 1987.
45. C. H. Wu, "*The Kinematic Error Model for the Design of Robot Manipulator*", Proceeding of the 1983 American Control Conference, San Francisco, pp 497-502, September 1983.
46. H. W. Stone, A. C. Sanderson, C. P. Neuman, "*Arm Signature Identification*", Third IEEE International Conference on Robotics, pp 41-48, March 1986.
47. J. Chen, L. M. Chao, "*Position Error Analysis for Robot Manipulators with all Rotary Joints*", Third IEEE International Conference on Robotics, San Francisco, pp 1011-1015, March 1986.
48. M. Tucker, N. D. Perreira, "Motion Planning and Control for Improved Robot

Performance", International Industrial Controls Conference, Long Beach, pp 1-17, Sept 1986.

49. Y. B. Kavina, H. Koçekali, et al, *"Macro/micro Manipulator part 1. A justification for the enhancement of industrial robot capability and performance, for tasks such as assembly and position adjustment by the attachment of a suitable micro-manipulator to the industrial robot or macro-manipulator"*, Proceedings of the Fourth National Conference on Production Research, Sheffield, UK, pp 250-253, September 1988.

50. A. Nowrouzi, H. Koçekali, et al, *"Pose error determination of robot end-effector"*, Proceedings of the Forth National Conference on Production Research, Sheffield, UK, pp 268-272, Sept 1988.

51. D. Nitzan, A. E. Brain, R. O. Duda, *"The Measurement and Use of Registered Reflectance and Range Data in scene Analysis"*, Procs IEEE, Vol 65, pp 206-220, Feb 1977.

52. J. Borenstein, *"Development of 3D Force sensor"*, MSc Thesis, Technion, Haifa, Israil, 1983.

53. Y. B. Kavina, H. Koçekali, R. A. Whitaker, *"An analytical and modular approach to robotic force control, using a wrist based force sensor"*, Proceeding of the Third National Conference on Production Research, Nottingham, pp 175-179, Sept 1987.

54. Whitney D E, *"Historical Perspectives & State of the Art in Robot Force Control"*, Int J. Rob Res, Part 1, Vol 6, 1987, pp 3-14.

55. G.Purnell, N.A. Maddock, K.Khodabandehloo, *"Robot Deboning for Beef Forequarters"*, Robotica, Vol 8, pp 303-310, 1990.

56. H.West, H.Asada, *"Kinematic Analysis and Design of Two Hold Mechanisms for Grinding Robot"*, Proceeding of Computer Integrated Manufacturing and Robotics, ASME, November 1984.

57. C.P.Day, *"Robot Accuracy Issues and Methods of Improvement"*, Proceedings of 17th International Robots, pp5.1-5.23, 1987.

58. J. Borenstein, "*Development of 3D Force Sensor*", Msc Thesis, Technion, Haifa, Israel, 1983.
59. Welwyn Strain Measurements Instruction Bulletins, B-137, B-129 and Tech Tip TT-605, TT-610.
60. Welwyn Strain Measurements, Catalogue 400, 1987.
61. Hewlett Packard Laser Interferometer System Type HP5526A, Application Notes 156-4.
62. Strain Gauge Conditioner & Amplifier System, Instruction Manual, Measurement Group Inc, Instrument Division.
63. PC-30A, 16 channel A-D & D-A 12-bit converter manual, Amplicon Liveline Ltd, Brighton, East Sussex, UK.
64. Lubrication & Bearings, Auchterlonie, New Southgate, London.

**List of-All Figure Boxes and Captions**

Figure I System operating principle ..... xi

Figure 1.1, Three possible first contact situations when putting a  
spacer onto a peg ..... 8

Figure 1.2, Schematic of an adaptive solution ..... 8

Figure 1.3, Hitachi experimental learning system consists of a simple  
robot with multi-directional force sensor on the table. .... 11

Figure 4.1, Six-components wrist sensor based on strain gauge  
elements. (SRI International) ..... 34

Figure 4.2, Structure of the Technion's three component force  
sensor. .... 35

Figure 4.3, Six component force-torque wrist sensor for deburring  
application ..... 36

Figure 4.4, Improved design of six component force-torque wrist  
sensor ..... 36

Figure 4.5, Strain gauge sensor developed at UWIST. .... 37

Figure 4.6, Draper Laboratory force sensing wrist ..... 38

Figure 4.7, Cantilever beam instrumented with strain gauges for  
sensitivity to horizontal forces and insensitivity to vertical forces .. 38

Figure 4.8, Sensing element subjected to vertical and horizontal  
forces ..... 39

Figure 4.9, Bridges A & B sensitive to vertical & horizontal forces  
respectively ..... 40

Figure 4.10, Sensing element and overload protection device ..... 42

Figure 4.11, Plate number 1: Force sensor with overload protection  
device ..... 43

Figure 4.12, Plate number 2: The monolithic sensing element  
(machined on Wadkin CNC Milling Machine) ..... 44

Figure 4.13, Plate number 3: Marks of mechanical failure resulted in  
measuring forces of a cutter in machining Aluminium alloy ..... 45

Figure 4.14, Plate number 4: The dismantled force sensor with  
fractured central part under excessive dynamic load ..... 46

Figure 4.15, Plate number 5: The earlier hardware system used in data  
acquisition and signal processing ..... 47

Figure 4.16, Plate number 6: The second force sensor designed with

new features is shown in calibration stage. The dial gauges were used to measure deflections in two dimensions under static loading. ....	48
Figure 4.17, Circuit diagram for strain gauge wiring on a deflection bar .....	50
Figure 4.18, Bar cross section .....	50
Figure 4.19, Force sensor reference frame .....	51
Figure 4.20, Plate number 7: The third force sensor designed for measuring force & torque in three orthogonal directions. The two torque arms with notches at specified interval from the centre facilitates the testing of force-torque sensor. ....	53
Figure 4.21, Three adjacent points on a curved path .....	55
Figure 4.22, Front view of geometric path .....	56
Figure 4.23, Plate number 8: The designed rig for the robot to follow different geometric paths using force control technique .....	57
Figure 5.1, Macro/Micro manipulator .....	60
Figure 5.2, Sliding Tube Micro-manipulator .....	64
Figure 5.3, Micro-manipulator subassemblies .....	65
Figure 5.4, Front elevation of the central support tube and the Z-translation tube. ....	67
Figure 5.5, Z-translation drive assembly .....	68
Figure 5.6, Plain bearing drive mechanism .....	70
Figure 6.1, Data Acquisition Block Diagram .....	75
Figure 7.1, The Direction Decoder Circuit .....	84
Figure 7.2, Block Diagram showing a closed loop motor control system .....	85
Figure 7.3, The Level Shifter Circuit .....	87
Figure 7.4, showing the Motor Driver .....	88
Figure 7.5, Graph shows clipped waveform. (Clipping point is DC offset value) .....	91
Figure 7.6, Clipper Circuit .....	91
Figure 7.7, shows the input to the D-type Flip-Flop from the clipper ....	92
Figure 7.8, showing the DFF Direction decoder .....	92
Figure 7.9, The 16-bit Up-Down Counter .....	94
Figure 7.10, The D/A converter .....	95

Figure 7.11, The complete control system for the micro-manipulator . . . . 96

Figure 7.12, Modified Counter Circuit . . . . . 98

Figure 8.1, Flow chart, showing basic program operation . . . . . 102

Figure 8.2, Flow chart, outlining the operation of the path following  
strategy.  
..... 132

Figure A1, Side elevation, showing plain bearing drive mechanism. . . . 193

Figure A2, End-elevation showing roller thrust bearing drive  
mechanism. . . . . 194

Figure A3, Side-elevation. . . . . 194

# ***Appendix I***



1 Program for machining the sensing element and overload protection device

PROGRAM FOR WADKIN MILLING MACHINE

(ID, PROG, 3434)	Title
E0 D0	zero coordinates
T23 M6	call up tool 23(7 mm slot drill)
S900 M13	set spindle speed at 900 rpm and start
G0 X0 Y0 E4	move quickly to centre point
workpiece	
G0 Z0 D23	Z0 plane is set at coordinates
in D23	
X30 G0 Y30	move quickly to point above
centre of	segment to be
machined out	
G1 Z-18 F100 G42	move down 18 mm into workpiece
at feed	100, set offset
G1 X9.967 Y14.308 F250	set feed rate for cutting at 250
G2 X3.5 Y17.457 I-2.467 J3.149	numerical
coordinates for G1 X3.5 Y50.446	
G2 X8.088 Y54.402 I4 J0	moving the cutting
tool G2 X54.402 Y8.088 I-8.088 J-54.402	around the
profile to	
G2 X50.446 Y3.5 I-3.957 J-0.588	be machined out
G1 X17.457 Y3.5	
G2 X13.782 Y5.921 I0 J4	
G3 X5.921 Y13.782 I-13.782 J-5.921	
G2 X6.118 Y21.211 I1.579 J3.675	
G1 X30 Y30 G40	cutter returns to centre of
segment,	offset off
G0 Z0	cutter lifts up clear of
workpiece	
G0 X-30 Y30	move to centre of second segment
ie	between 270 to 360
G1 Z-18 F100 G42	slow feed at tool drops down
cutting	through the metal,
offset on	
G1 X-14.308 Y9.967 F250	cutting feed rate of 250
G2 X-17.457 Y3.5 I-2.467 J-3.149	
G1 X-50.446 Y3.5	
G2 X-54.402 Y8.088 I0 J4	numerical
G2 X-8.088 Y54.402 I54.402 J-8.088	coordinates
G2 X-3.5 Y50.446 I0.588 J-3.957	for profile
G1 X-3.5 Y17.457	of second
G2 X-5.921 Y13.782 I-4 J0	segment
G3 X-13.782 Y5.921 I5.921 J-13.782	
G2 X-21.211 Y6.118 I-3.675 J1.579	

2 Program for machining the sensing element and overload protection device

```
G3 X-5.921 Y-13.782 I13.782 J5.921
G2 X-6.118 Y-21.211 I 1.579 J-3.675

G1 X-30 Y-30 G40      move cutting tool back to centre
                      of segment, offset off
G0 Z0                raise tool clear of workpiece
G0 X30 Y-30          move tool above centre of 4th
                      final segment ie 90 to 180 degrees

G1 Z-18 F100 G42      slowly lower tool, offset on
G1 X14.308 Y-9.967 F250 T2 set cutting tool rate,
                      call up empty tool space

G2 X17.457 Y-3.5 I3.149 J2.467
G1 X50.446 Y-3.5
G2 X54.402 Y-8.088 I0 J-4
G2 X8.088 Y-54.402 I-54.402 J8.088      numerical
coordinates
G2 X3.5 Y-50.446 I-0.588 J3.957      for profile of 4th
                      segment
G1 X3.5 Y-17.457
G2 X5.921 Y-13.782 I4 J0
G3 X13.782 Y-5.921 I-5.921 J13.782
G2 X21.211 Y-6.118 I3.675 J-1.579

G1 X30 Y-30 G40      return to centre of
segment
G0 Z0 M6             move cutter upwards and clear of work,
                      activate empty tool space, to put tool
away

E0 D0               return machine to zero
M30
(PROG,END)          end of program.
```

3 Program for machining the sensing element and overload protection device

SUBROUTINE FOR CENTRE DRILLING HOLES

```
(ID,PROG,3435)      Title of program
T4 M6              call up tool 4 and load(centre
drill)
S1000 M13          set spindle speed at 1000 rpm
and start
G0 X0 d4           set zero at centre of workpiece
F100 G81 G0 X0 Y0 Z-13 R0  feed 100 downwards, call
                           subroutine
G80 T2            return subroutine and call up
empty             space tool 2
F100 G81 G0 X0 Y0 Z-16 R0  feed 100 downwards,
call             subroutine,
lower drill 16 mm
from Z0 plane

(GSUB,3435)        centre holes subroutine
```

```
(ID,GSUB,3434)    Title of subroutine
```

```
X31.25 Y54.127
X54.127 Y31.25
X-31.25 Y54.127
X-54.127 Y31.25   coordinates of
X-31.25 Y-54.127 eight outer
X-54.127 Y-31.25 holes
X31.25 Y-54.127
X54.127 Y-31.25
```

```
(END,GSUB)        end of subroutine
```

```
(ID,GSUB,3435)    Title off subroutine
```

```
X5.657 Y5.657
X-5.657 Y5.657   coordinates of four
X-5.657 Y-5.657 centre holes
X5.657 Y-5.657
```

```
(END,GSUB)        end of subroutine
```

# CNC Program for Machining of the Sensing Element &

## the Overload Protection Device

22 \*00f2

N0010G1X0Y0

N0020X0Y18

N0030G3X-4Y22I-4J0

N0040G1X-6Y22

N0050G2X-10Y26I0J4

N0060G1X-10Y29

N0070G2X-7Y32I3J0

N0080G1X7Y32

N0090G2X10Y29I0J-3

N0100G1X10Y15

N0110G2X7Y12I-3J0

N0120G1X-7Y12

N0130G2X-10Y15I0J3

N0140G1X-10Y29

N0150G2X-7Y32I3J0

N0160G1X-4Y32

N0170G2X0Y28I0J-4

N0180G1X0Y0

N0190X-18Y0

N0200G3X-22Y-4I0J-4

N0210G1X-22Y-6

N0220G2X-26Y-10I-4J0

N0230G1X-29Y-10

N0240G2X-32Y-7I0J3

N0250G1X-32Y7

N0260G2X-29Y10I3J0

N0270G1X-15Y10

N0280G2X-12Y7I0J-3

N0290G1X-12Y-7

N0300G2X-15Y-10I-3J0

N0310G1X-29Y-10

N0320G2X-32Y-7I0J3

N0330G1X-32Y-4

N0340G2X-28Y0I4J0

N0350G1X0Y0

N0360X0Y-18

N0370G3X4Y-22I4J0

N0380G1X6Y-22

N0390G2X10Y-26I0J-4

N0400G1X10Y-29

N0410G2X7Y-32I-3J0

N0420G1X-7Y-32

N0430G2X-10Y-29I0J3

N0440G1X-10Y-15

N0450G2X-7Y-12I3J0

N0460G1X7Y-12

N0470G2X10Y-15I0J-3

N0480G1X10Y-29

N0490G2X7Y-32I-3J0

N0500G1X4Y-32

N0520G1X0Y0

N0530X18Y0

N0540G3X22Y4I0J4

N0550G1X22Y6

N0560G2X26Y10I4J0

N0570G1X29Y10

N0580G2X32Y7I0J-3

N0590G1X32Y-7

N0600G2X29Y-10I-3J0

N0610G1X15Y-10

N0620G2X12Y-7I0J3

N0630G1X12Y7

N0640G2X15Y10I3J0

N0650G1X29Y10

N0660G2X32Y7I0J-3

N0670G1X32Y4

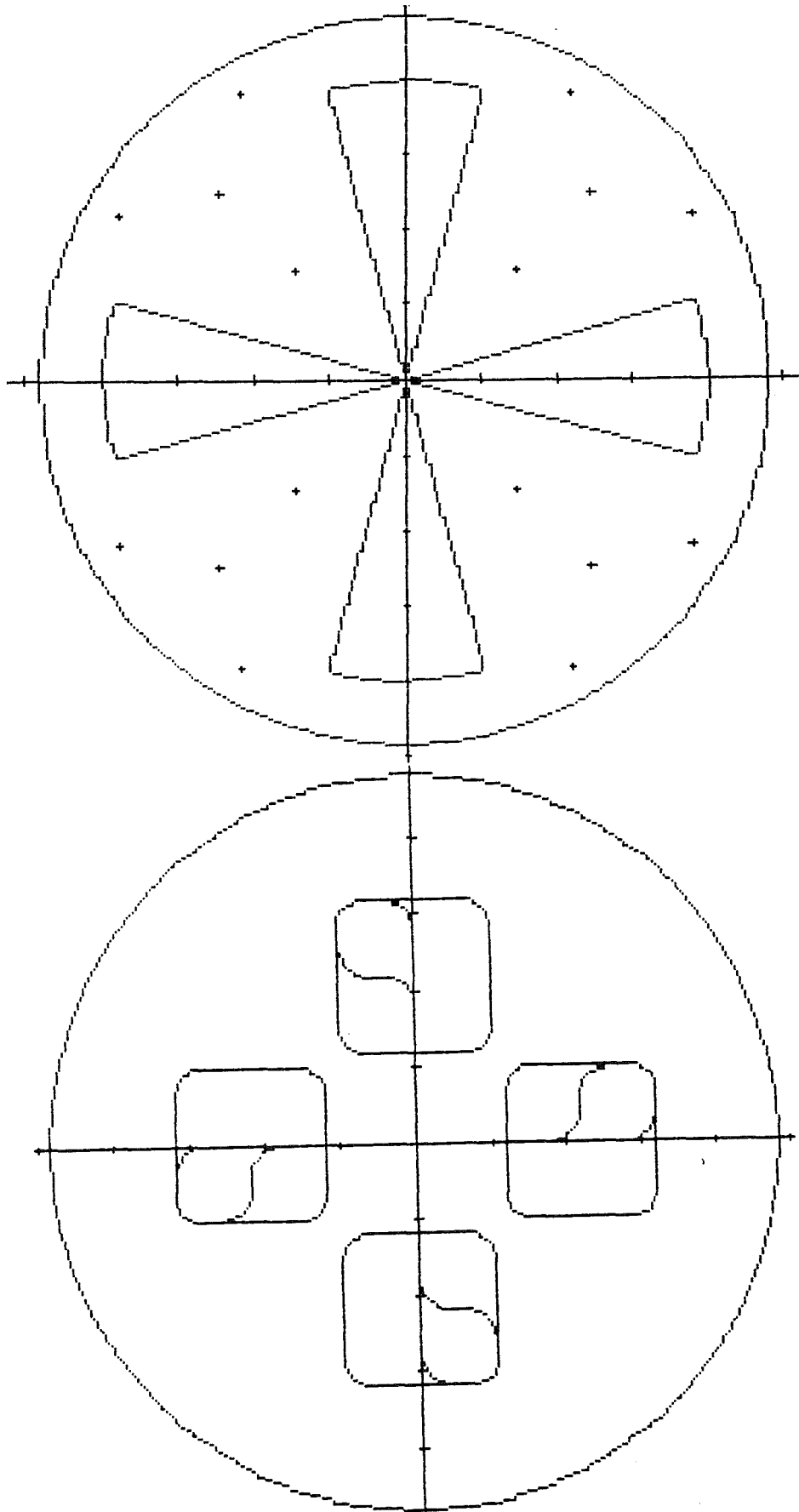
N0680G2X28Y0I-4J0

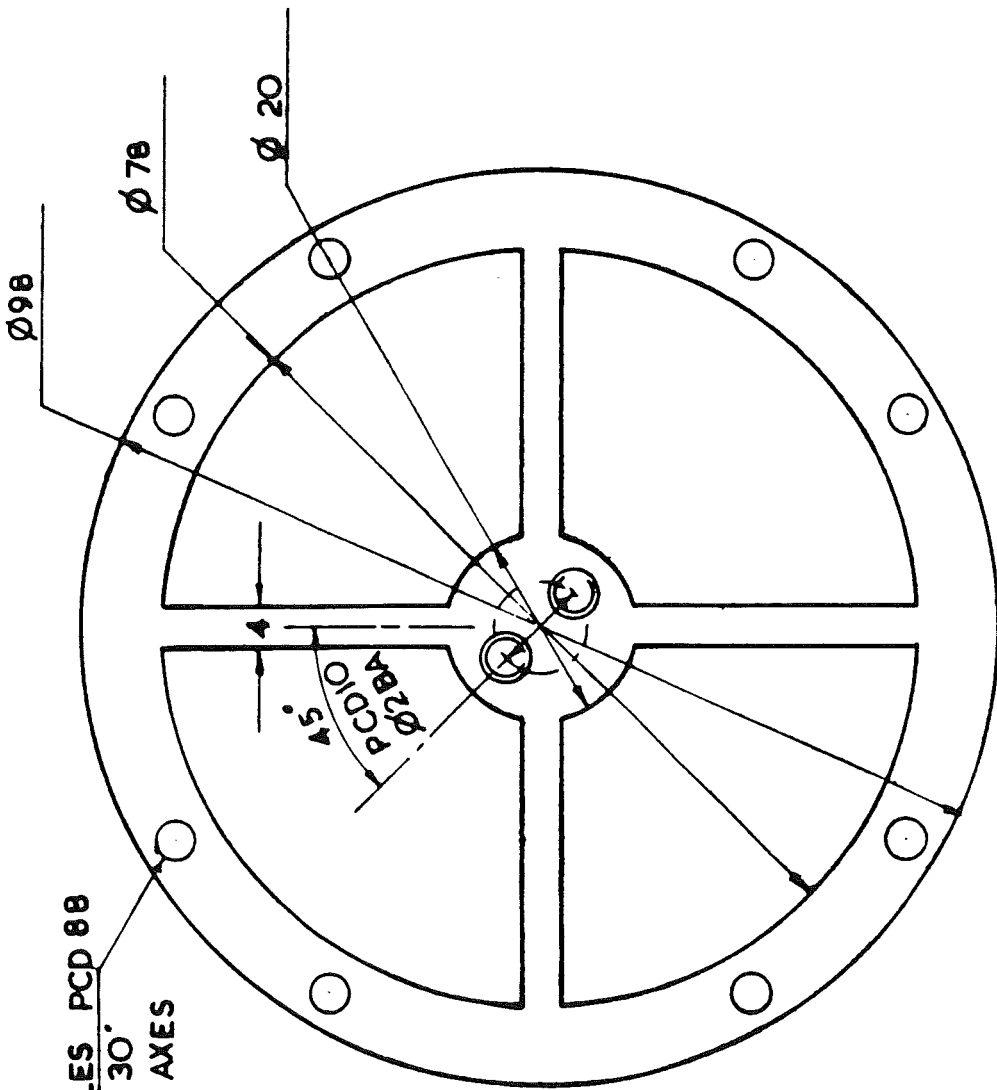
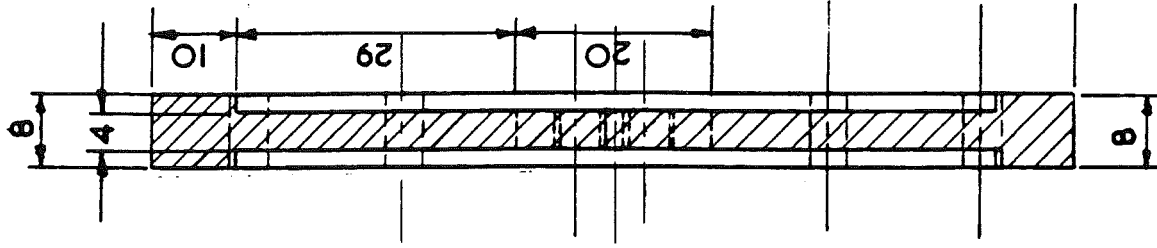
N0690G1X0Y0

## CNC Program for Machining of the Sensing Element

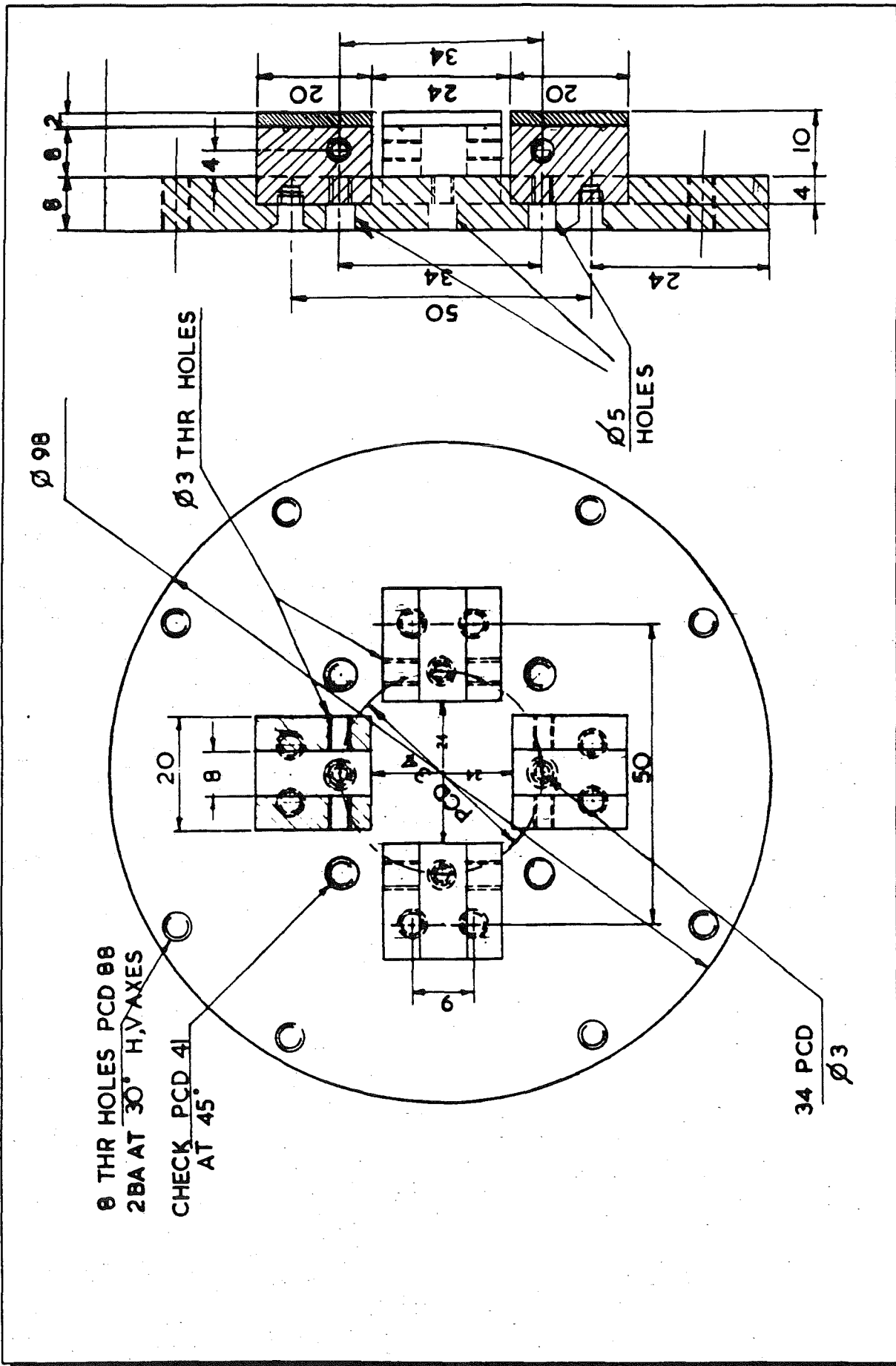
```
N0700X0Y0
N0710X-10.0906Y37.6587
N0720G2X-9.3587Y38.8898I0.966J0.259
N0730X9.3587Y38.8898I9.359J-38.89
N0740X10.0906Y37.6587I-0.234J-0.972
N0750G1X0Y0
N0760X-37.6587Y-10.0906
N0770G2X-38.8898Y-9.3587I-0.259J0.966
N0780X-38.8898Y9.3587I38.89J9.359
N0790X-37.6587Y10.0906I0.972J-0.234
N0800G1X0Y0
N0810X10.0906Y-37.6587
N0820G2X9.3587Y-38.8898I-0.966J-0.259
N0830X-9.3587Y-38.8898I-9.359J38.89
N0840X-10.0906Y-37.6587I0.234J0.972
N0850G1X0Y0
N0860X37.6587Y10.0906
N0870G2X38.8898Y9.3587I0.259J-0.966
N0880X38.8898Y-9.3587I-38.89J-9.359
N0890X37.6587Y-10.0906I-0.972J0.234
N0900G1X0Y0
  36 set1=cir4,x0,y0,a45,d90,r35
      X   24.7487 Y   24.7487
      X  -24.7487 Y   24.7487
      X  -24.7487 Y  -24.7487
      X   24.7487 Y  -24.7487
  37 set2=cir4,x0,y0,a45,d90,r20.6375
      X   14.5929 Y   14.5929
      X  -14.5929 Y   14.5929
      X  -14.5929 Y  -14.5929
      X   14.5929 Y  -14.5929
  38 set3=cir12,x0,y0,a30,d30,r44,out1,4,7,10
      * STAT: SYNTAX ERROR
  39 set3=cir12,x0,y0,a30,d30,r44,out,3,6,9,12
      X   38.1051 Y   22.0000
      X   22.0000 Y   38.1051
      X  -22.0000 Y   38.1051
      X  -38.1051 Y   22.0000
      X  -38.1051 Y  -22.0000
      X  -22.0000 Y  -38.1051
      X   22.0000 Y  -38.1051
      X   38.1051 Y  -22.0000
  39 draw
```

*Profile & Path of Cutting Tool in Machining the parts*





8 HOLES PCD 88  
 BA AT 30°  
 H.V AXES





# Materials Selection Guide for Transducer Spring Elements

MATERIAL	PROPERTIES (in "as-normally-used" condition) *										RELATIVE FIGURE OF MERIT †								REMARKS
	Modulus of Elasticity, $\text{psi} \times 10^{-6}$	Yield Strength (0.2% offset) $\text{psi} \times 10^{-3}$	Thermal Expansion Coeff., $\text{ppm}/^\circ\text{F}$	Density, $\text{lb}/\text{in}^3$	Max. Recommended Temperature, $^\circ\text{F}$	Thermal Conductivity $\text{Btu}/\text{hr ft}^2 (^\circ\text{F}/\text{in})$	Elongation, %	Hardness (Rc, unless noted)	Thermal Conductivity	Linearity	Hysteresis	Creep/Relaxation	Material Cost	Machinability	Hardening Distortion	Corrosion Resistance	Welding or Brazing Suitability	Lot-to-lot Consistency	
<b>High-Modulus Alloys (AISI Steels)</b>																			
4140	30	200	6.5	0.283	+400	270	12	42-45	3	8	7	9	5	3	1	5	6	6	very good tool steel
E4340	30	210	6.3	0.283	+500	260	12	43-46	3	8	8	8	5	3	1	5	8	5	excellent tool steel
18 Ni (250), maraging	27	245	5.6	0.289	+600	140	7-10	48-49	2	8	8	7	4	5	2	8	5	8	good, but seldom used
410 stainless	29	140	6.0	0.280	+400	170	15	39	2	7	7	6	4	2	6	2	6	6	seldom used
630 stainless (17-4 PH)	28.5	185	6.0	0.280	+400	135	14	41-44	2	7	7	6	6	8	7	7	7	3-4	widely used
631 stainless (17-7 PH)	29	220	5.7	0.276	+400	130	6-7	46-48	2	8	8	7	6	6	8	7	7	5	widely used
632 stainless (PH 15-7 Mo)	29	220	5.0	0.277	+500	110	6	47	2	8	8	8	4	6	8	7	7	5	excellent, but seldom used
S15500 stainless (15-5 PH)	28.5	185	6.0	0.280	+400	135	14	41-44	2	7	7	6	6	8	7	7	7	5-6	improved version of 630
<b>Low-Modulus Alloys</b>																			
2014-T6 alum.	10.6	60	12.4	0.101	+200	1075	10	B 135	8	6	7	6	5	8	9	3	3	6	good
2024-T6/T351 alum.	10.6	46	12.6	0.101	+200	840	18	B 120	7	6	7	6	5	8	9	3	3	6	good, widely used
2024-T81 alum.	10.6	65	12.6	0.101	+250	1040	8	B 128	8	7	8	7	5	8	9	3	3	6	best of aluminum alloys
6061-T6 alum.	10.0	40	12.9	0.098	+150	1160	14	B 95	8	5	4	4	6	7	8	4	5	5	fair performance
7075-T6 alum.	10.4	70	12.9	0.101	+100	840	10	B 150	7	7	6	6	5	8	8	3	2	5	poor at elevated temperatures
Beryllium copper 25	17	170	9.3	0.298	+250	750	2-3	40-42	7	8	8	8	2	5	9	3	4	7	excellent, but costly
6Al 4V titanium	16.5	165	4.9	0.160	+300	50	12	40	1	7	7	7	2	3	2	8	4	5	used in special applications only
<b>Miscellaneous Materials</b>																			
AISI 304 stainless steel	28	150	9.6	0.290	+250	110	4-5	39	2	5	4	4	7	3	8	5	6	6	poor spring material
Ni-Span-C alloy	28	180	4.3	0.294	+250	90	9	37-38	1	8	8	8	3	3	4	3	5	7	very good, but costly
AZ31B magnesium	6.5	32	14.5	0.064	+100	650	15	B 73	6	3	2	2	6	8	1	7	5	7	very poor spring material
Ceramic, $\text{Al}_2\text{O}_3$ (99+%)	55	60-70	3.3	0.141	+500	260	—	Mohs: 9	3	7	8	8	1-3	1	10	—	—	7	special applications only
Epoxy-glass laminate (high-pressure, low resin content)	5-7	70-100	5.8	0.06-	100-	2	—	—	0	5	4	3	5	3	4	—	—	3	special, low-accuracy applications only

\* The mechanical and physical properties given in this table have been carefully selected from a wide range of sources as representative for typical transducer applications and environments.

† Numbers tabulated for FIGURE OF MERIT represent relative ratings on a scale of 0 to 10, with higher numbers indicative of more desirable qualities. The ratings are based on experience, judgment, and current practices in the transducer field.

1) for GPa, multiply by 6.9  
 2) for MPa, multiply by 6.9  
 3) from 75°F to 212°F; for ppm/°C, multiply by 1.8  
 4) for  $\text{g}/\text{cm}^3$ , multiply by 27.7  
 5) transducer applications: for °C, subtract 32 and multiply by 5/9  
 6) for  $\text{watt}/\text{m}^2\text{ }^\circ\text{C}$ , multiply by 0.144  
 7) in 2 in. or 51 mm  
 8) no ductile range  
 9) not hardenable by heat treatment

Table 1. Typical Mechanical Properties of 2024 Alloy (a)

Temp <sup>r</sup>	Tensile strength, psi (b)	Yield strength, psi (c)	Elongation, % (d)	Hardness		Shear strength, psi (f)	Fatigue limit, psi (g)	Modulus of elasticity, psi (h)
				Rockwell (e)	Bhn (f)			
O.....	27,000	11,000	20	T 25 to 25	47	18,000	13,000	10,000,000
T3.....	70,000	50,000	18	F 94 to 103	130	41,000	20,000	10,000,000
T4.....	62,000(1)	47,000(1)	20	F 94 to 103	130	41,000	20,000	10,000,000
T34.....	72,000	57,000	13	T 83 to 87	130	42,000	18,000	10,000,000
Bare Products								
O.....	28,000	11,000	20	E 90 to 100	...	18,000	...	10,000,000(m)
T3.....	63,000(k)	43,000(k)	18	E 90 to 100	...	40,000	...	10,000,000(m)
T4.....	64,000(k)	42,000(k)	18	T 77 to 84	...	40,000	...	10,000,000(m)
T34.....	67,000(k)	53,000(k)	11	E 90 to 100	...	41,000	...	10,000,000(m)
T81.....	65,000(k)	50,000(k)	6	...	...	40,000	...	10,000,000(m)
T84.....	70,000(k)	56,000(k)	6	...	...	42,000	...	10,000,000(m)

(a) These typical properties are average for various forms, sizes, and methods of manufacture, and may not exactly describe any one particular product. (b) Strengths and elongations are unchanged or improved at low temperatures. (c) The compressive yield strength is approximately the same as the tensile yield strength. (d) Sheet specimens, 1/16 in. thick. (e) Round specimens, 1/2 in. in diameter. (f) 200-kg load, 10-mm ball. (g) Based on 500,000 cycles using R. R. Moore type rotating-beam machine. (h) Average of tension and compression moduli. Compression modulus is about 2% greater than tension modulus. (i) Extruded products more than 1/4 in. thick will have strengths 15 to 20% higher than these values. (j) Rockwell hardness for clad sheet applies to 0.032 to 0.061-in. thicknesses. The T36 values are equivalent to those of T8. (k) Sheet more than 0.062 in. thick will have strength slightly higher than these values. (m) Secondary modulus is approximately 10,100,000 psi for sheet 0.064 in. and thicker, and 9,800,000 psi for sheet less than 0.064 in.

Table 2. Typical Tensile Properties of 2024 Alloy at Various Temperatures (Bare Products)

Testing temp., F	Tensile strength, psi	Yield strength, psi	Elongation, %	Testing temp., F	Tensile strength, psi	Yield strength, psi	Elongation, %
70	70,000	50,000	18	73	62,000	47,000	18
250	55,000	50,000	11	300	45,000	36,000	17
400	32,000	22,000	23	400	27,000	20,000	27
500	12,000	8,000	33	500	12,000	9,000	35
600	2,000	6,000	73	600	8,000	6,000	75
700	3,500	4,000	100	700	5,500	4,900	100

Table 3. Heat Treatment for 2024 Alloy

Purpose	Temperature		Time, hr	Cooling
	C	F		
Complete annealing, O ..	413	775	2 to 3	Furnace cool to 500 F (50 °F per hr max)
Removal of cold work ...	300	500	No holding	Not critical
Solution heat treatment, 420 to 420	500 to 500	900 to 900	10 min to 1 hr in salt bath (a); longer time in air	Cold water quench
(followed by)				
Precipitation, T4 .....	Room temperature		48	
Precipitation, T81 .....	120 to 120	250 to 250	11 to 13(b)	Not critical
(Start with T3 material)				
Precipitation, T84 .....	120 to 120	250 to 250	7 to 9	Not critical
(Start with T36 material)				
Precipitation, T8 .....	120 to 120	250 to 250	11 to 13	Not critical

(a) Time at temperature depends on product and thickness. Longer times may be required for heavy sections and forgings. (b) For extrusions, age for 6 hr.

strength, 0.55 times the tensile yield strength.

L 2 Radiography. Same as for 2024 alloy

M 1 Chemical composition limits. Core

2.5 to 4.8 Cu, 1.2 to 1.8 Mg, 0.20 to

0.9 Mn, 0.50 max Si, 0.20 max Fe,

0.10 max Cr, 0.25 max Zn, 0.05 max

each of other elements, total of which

shall be 0.15 max, rem Al

Cladding: 99.20 min Al, 0.10 max Cu,

0.05 max Mn, 0.7 max Fe, 0.10 max

max Zn, 0.05 max each of others

N 7 Hot working temperature range 300

to 900 F (260 to 480 C)

15 Heat treatment. See Table 3

2218 Alloy

(4% Cu, 2% Ni, 1.5% Mg)

A 3 ASTM number. Forgings. E347

4 Government number. Alloy E218

Forgings: QQ-A-357.

5 Other designations. SAE 203; AMS

413.

Table 1. Typical Tensile Properties of 2218-F30 Alloy at Various Temperatures When Heated for 10,000 Hr

Testing temp., F	Tensile strength, psi	Yield strength, psi	Elongation, %
73	50,000	44,000	13
200	41,000	33,000	17
300	32,000	18,000	30
400	10,000	6,000	70
500	5,500	3,000	83
700	4,000	2,500	100

Table 2. Heat Treatment for 2218 Alloy

Purpose	Temperature		Time, hr	Cooling
	C	F		
Complete annealing, O .....	413	775	2 to 3	Furnace cool to 500 F (50 °F per hr max)
Solution heat treatment	504 to 516	900 to 900	4	Cold water quench. Heavy or complicated forgings, water 150 to 212 F
(followed by)				
Precipitation, T72 .....	232 to 243	400 to 470	5 to 8	Not critical

B 1 Typical uses. Applications involving elevated temperatures. Forged cylinder heads and pistons are typical.

C 1 Density at 68 F (20 C). 2.81 g per cu cm (0.102 lb per cu in.).

D 1 Liquidus temperature, 1175 F (633 C).

2 Solidus temperature, 940 F (504 C).

3 Thermal expansion, micro-in./in./°C.

-48 to + 68 F (-20 to + 20 C).....21.1

68 to 212 F (20 to 100 C).....22.3

68 to 302 F (20 to 150 C).....22.4

68 to 572 F (20 to 300 C).....24.3

11 Specific heat at 212 F (100 C). 0.23 cal/g

10 Thermal conductivity at 77 F (25 C). cal/cm/cm/°C/sec. 0.37, T72 temper

E 1 Electrical volume conductivity, 40% IACS, T72 temper

2 Electrical resistivity at 68 F (20 C). 4.21 microhm-cm, T72 temper

K 1 Mechanical properties. See Table 1.

The following typical mechanical properties of 2218-F30 alloy are averages for various sizes of forgings and may not exactly describe any one particular size. Tensile strength, 44,000 psi (strengths and elongations are unchanged or improved at low temperatures); yield strength, 37,000 psi (compressive yield strength is approximately the same as the tensile yield strength); shear yield strength is approximately 0.55 times the tensile yield strength; elongation, 11% on 1/2-in. diam specimens; hardness, 95 Bhn (500-kg load, 10-mm ball); shear strength, 30,000 psi; modulus of elasticity (average of tension and compression moduli), 10,000,000 psi. The compression modulus is about 2% greater than tension modulus. Modulus of rigidity, 4,000,000 psi; Poisson's ratio, 0.33.

2 Directional properties (forgings). Transverse tensile strength approximately same as longitudinal. Transverse elongation approximately 70% of longitudinal.

L 2 Radiography. Same as for 2024 alloy

M 1 Chemical composition limits. 3.5 to 4.5 Cu, 1.7 to 2.3 Ni, 1.2 to 1.8 Mg,

0.9 max Si, 1.0 max Fe, 0.20 max Mn,

0.10 max Cr, 0.25 max Zn, 0.05 max

each of other elements, total of which shall be 0.15 max, rem Al

N 7 Hot working temperature range, 500 to 900 F (260 to 480 C)

15 Heat treatment. See Table 3 for details of the various processes.

3003 Alloy

(1.2% Mn)

A 3 ASTM numbers. E209, E210, E211,

E221, E224, E225, E241, E247, E307,

E313, E318 and E345.

4 Government numbers. Alloy 3003. QQ-A-156, QQ-A-357, QQ-A-359, and WW-T-728.

5 Other designations. SAE 29 and 245; AMS 4006, 4008, 4043, and 4067

B 1 Typical uses. Applications where good formability or very good resistance to corrosion or good weldability, or all three, are required, and where more strength is desired than is provided by commercially pure aluminum. Cooking utensils, food and chemical handling and storage equipment, gasoline and oil tanks, trim in transportation equipment, pressure vessels and piping are typical.

C 1 Density at 68 F (20 C). 2.73 g per cu cm (0.099 lb per cu in.)

D 1 Liquidus temperature, 1216 F (654 C)

2 Solidus temperature, 1180 F (643 C)

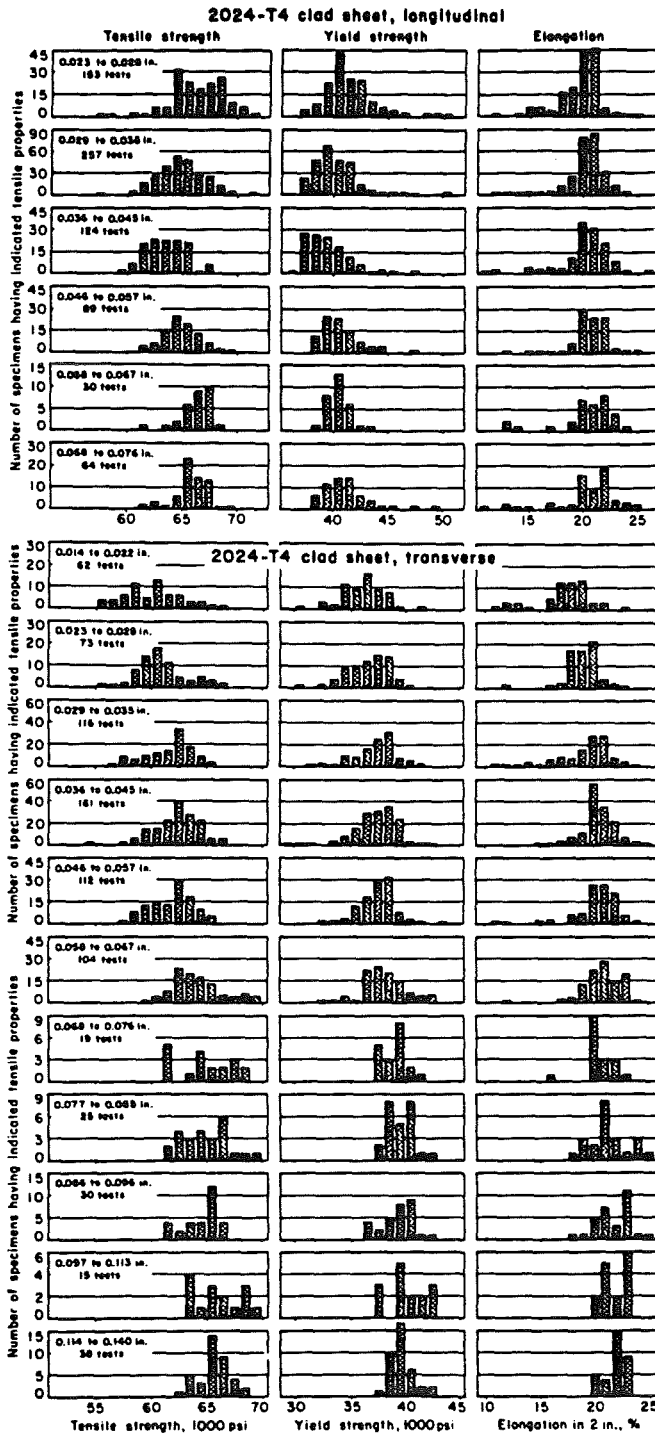


Table 2. Typical Tensile Properties of 2014 Alloy at Various Temperatures When Heat-Treated for 16,000 Hr (Bare Products)

Testing temp, F	Tensile strength, psi	Yield strength, psi	Elongation, %
<b>T4 Temper</b>			
75	42,800	42,800	30
300	44,800	33,800	..
400	18,800	12,800	..
500	11,800	8,300	..
600	6,500	5,900	..
700	4,500	3,500	..
<b>T6 Temper</b>			
75	79,800	69,800	13
300	87,800	49,800	15
400	18,800	12,800	25
500	11,800	8,500	45
600	6,500	5,900	65
700	4,500	3,500	70

Table 3. Heat Treatment of 2014 Alloy

Purpose	Temp, F	Time, hr
Complete annealing, O	775	2 to 3(a)
Removal of cold work	650	No holding time required(b)
Solution heat treatment, T4	825 to 845	Up to 1 hr in salt bath(c); longer time in air(d)
Precipitation, T6.. (start with T4)	335 to 345	6 to 12(b)

(a) Furnace cool to 500 F (50 °F per hour max). (b) Cooling rate not critical. (c) Time depends on product thickness. Longer times for heavy sections. (d) Cool by cold water quench. Quench heavy sections in water at 150 to 212 F.

Fe, 0.35 max; Cr, 0.20 max; Zn, 0.10 max; Ti, 0.05 max each of other elements, total of which shall be 0.15; rem Al  
 N 7 Hot working temperature, 800 to 900 F (380 to 480 C)  
 15 Heat treatment. See Table 3.

**2024 Alloy**

(4.5% Cu, 1.5% Mg, 0.6% Mn)

- A 3 ASTM numbers, B200, B210, B211, B212, B235, and B316.
- 4 Government numbers, Alloy 2024, QQ-A-287, QQ-A-288, QQ-A-355, and WW-T-785, Alclad 2024, QQ-A-382.
- 5 Other designations, SAE 34 and 240; AMS 4033, 4034, 4035, 4037, 4040, 4041, 4042, 4086, 4087, 4088, 4119, 4120, 4152 and 4164.
- B 1 Typical uses, Aircraft structures, rivets, hardware, truck wheels, screw-machine products, and other miscellaneous structural applications.
- C 1 Density at 68 F (20 C), 2.71 g per cu cm (0.100 lb per cu in.)
- D 1 Liquidus temperature, 1180 F (638 C)  
 2 Solidus temperature, 825 F (452 C)  
 6 Thermal expansion, micro-in./in./°C.  
 -76 to +48 F (-60 to +10 C)..... 21.4  
 68 to 212 F (20 to 100 C)..... 22.8  
 68 to 352 F (20 to 200 C)..... 23.9  
 68 to 572 F (20 to 300 C)..... 24.7
- 11 Specific heat at 212 F (100 C), 0.23 cal/g
- 16 Thermal conductivity at 77 F (25 C), cal/mq cm/°C/sec. 0.45, O temper; 0.29, T3, T4, and T36 tempers
- E 1 Electrical conductivity, 50% IACS, O temper; 30% IACS, T3, T4, T36  
 2 Electrical resistivity at 68 F (20 C), 3.45 microhm-cm, O temper; 5.75 microhm-cm, T3, T4, and T36 tempers.
- 11 Electrolytic solution potential versus N/10 calomel electrode is 0.59 volt, T3, T4, and T36 tempers (5% NaCl + 3 g H<sub>2</sub>O solution).
- K 1 Mechanical properties. See Tables 1 and 2 and graphs. Modulus of rigidity (bare products), 4,600,000 psi; Poisson's ratio (all tempers, bare and clad products), 0.33; shear yield

# CHOOSING THE CORRECT ALLOY FOR PLATE, SHEET & STRIP

... is determined by what the plate, sheet or strip will be used for, how severely it will be formed and what the welding requirements are. Then you can quickly match your application needs with the properties of the commonly available aluminium alloys. You will find below useful descriptions of what each alloy can do, and on the following pages are quick comparison charts of strength properties, formability, weldability, corrosion resistance etc.

## NON HEAT TREATABLE ALLOYS

**1050/1080/1200** - Commercially pure aluminium, highly resistant to chemical attack and weathering. Easily worked and welded, but the lowest strength aluminium. Excellent for chemical processing equipment and other uses where product purity is important, and for metal pressings of all types where ductility is critical.

**1350** - The electrical quality alloy not often found in the rolled form, normally rolled to special requirements.

**3003/3103** - Stronger than 1200 with same good formability. Fine corrosion resistance and weldability. Used for storage tanks, chemical equipment, brazing applications, van bodies and cryogenic uses. For higher strength, consider 5251 or 5052.

**5005** - Specified for anodised applications to match with 6063 extrusions. Anodised coating is clearer and lighter than for 3003/3103. Otherwise similar although it does work harden faster than 1200 or 3003/3103 in severe drawing applications. Uses are decorative, architectural and consumer products.

**5251/5052** - Far stronger than either of the above alloys. Forms readily in the intermediate tempers. Corrosion resistance is very good as is weldability. Better salt water corrosion resistance than 1200. Used for pressure vessels, tanks, fittings or applications involving forming or welding where strength higher than 3003 is needed. Also houseboat hulls, pontoon boats and van bodies.

**5083** - Stronger than 5251. For high strength welded applications with outstanding joint strength - rail wagons, marine components, bridges, stationary cryogenic vessels and overhead cranes. Excellent weldability and corrosion resistance. Static welded strength is second only to 5754 and 5083 is superior to 5754 in energy absorption. Restricted to temperatures below 150°F = 65.5°C. The widest range of plate from stock.

**5086** - Excellent for welded structures - ship hulls and superstructures, road tankers, pressure vessels, movable and stationary cryogenic vessels and structures requiring

a high rate of energy absorption. Restricted to temperatures below 150°F = 65.5°C. More commonly used in the U.S.A.

**5154** - Similar to 5083 which has largely superseded it. Mainly used on MOD and naval applications.

**5454** - For all types of welded structures. ASME Code approved for temperatures up to 400°F = 204.4°C. Used for hot asphalt road tankers, dump bodies, pressure vessels and marine structures.

**5754** - Similar to 5454 which has largely superseded it because of superior stress corrosion characteristics.

## HEAT TREATABLE ALLOYS

**2014/2024** - High strength alloys with excellent machinability widely used in aircraft. Have limited formability and only fair corrosion resistance in the heat treated condition. Not recommended for fusion welding. Used for high strength parts in aircraft and machinery, including gears and bolts and for security vans where strength is critical.

**6082/6061** - Widely used structural alloys for light to medium strength applications. Combine good formability, weldability, corrosion resistance and strength after heat treatment. Since they lose appreciable strength when welded, the 5000 series alloys replace them in some marine applications. Availability of bars, shapes, tubing and pipe in the same alloy helps make this a popular selection. 6082 is the most common heat treated plate from stock. 6061 is more commonly used in the USA.

**7020** - A good machining quality alloy mainly used on the continent instead of 6082.

**7075** - A very high strength aircraft alloy. Good machinability and hardness. Not for welding or corrosion resistance.

## TOOLING PLATE

**CAST** - A continuous cast plate machined both sides to about 25 microinch finish and thermally stress relieved. Has best thickness tolerance of any aluminium plate plus maximum stability. Excellent for precision applications such as electronic base or mounting plates, photo vacuum plates, checking fixtures and jigs. Good welding, machining, anodizing and corrosion resistance properties. No forming.

**7022** - Rolled tooling plate. This alloy is generally made to rolling mills house specifications. It is not as stable as cast tooling plate but gives much better life in highly machined components. Its porosity free characteristics make it ideal for plastic injection moulds.

## AEROSPACE MANUFACTURERS HOUSE SPECIFICATIONS (continued)

Issuing Authority	Specification	Alloy	Temper	Related National Specification
Rolls Royce	MSRR 8037	7019	M 16	
	MSRR 8039	5083	M O	L44
	MSRR 8058	5251	M 16	L113
	MSRR 8065	6082	M 16	L83 L168
	MSRR 8071	2031	M 16	L65 L168
		2031A	M 16	
	MSRR 8074	2618A	M 14 16	DTD 5014
		2618B	M M	
	MSRR 8075	2618A	M M O	
	MSRR 8082	2618A	M M O	
	MSRR 8086	7017	M O	

↑ Relate to IGC 04 32 213 Will become 4053  
 † Relate to IGC 04 32 263 Will become 4055  
 ‡ Will become 4071 \* These alloys do not have a direct international designation equivalent

## TYPICAL LONGITUDINAL MECHANICAL PROPERTIES OF 75-100mm SECTION EXTRUSIONS

Alloy	Temper	0.2% Proof Stress		Tensile Stress		Elongation
		MPa	ksi	MPa	ksi	
1200	O	60	87	116	20	
2014A	M	120	174	348	12	
	O	110	160	290	15	
	T4	320	464	638	14	
	T4511	440	638	814	14	
2024	T4	315	457	638	14	
	T6	470	682	914	9	
	T6510	470	682	747	9	
	T6511	470	682	747	9	
2071	T4	390	562	814	14	
	T6	470	682	914	9	
	T6510	470	682	747	9	
	T6511	470	682	747	9	
2017A	T4	310	450	638	14	
	T6	350	508	725	16	
	T6510	350	508	624	16	
	T6511	350	508	624	16	
2618A	T6	350	508	624	16	
	T8	370	537	624	16	
	O	120	174	320	15	
	T3	380	551	725	10	
2031	T4	380	551	725	10	
	T6	380	551	725	10	
	T6510	380	551	725	10	
	T6511	380	551	725	10	
5251	T4	360	522	711	10	
	T6	410	587	814	7	
	T8	410	587	814	7	
	T8510	410	587	814	7	
5454	T4	190	275	390	14	
	T6	230	335	480	14	
	O	190	275	390	14	
	T4	160	232	320	14	
5083	T4	160	232	320	14	
	T6	180	261	360	14	
	H111	180	261	360	14	
	H111	180	261	360	14	
5086	M	110	160	290	14	
	M	120	174	320	14	
	M	120	174	320	14	
	M	120	174	320	14	
6061	O	100	145	203	20	
	T4	130	189	269	18	
	T4510	130	189	269	18	
	T4511	130	189	269	18	
T6	T6	260	377	530	10	
	T6	260	377	530	10	
	T6510	260	377	530	10	
	T6511	260	377	530	10	

## TYPICAL LONGITUDINAL MECHANICAL PROPERTIES OF 75-100mm SECTION EXTRUSIONS

Alloy	Temper	0.2% Proof Stress		Tensile Stress		Elongation
		MPa	ksi	MPa	ksi	
6082	M	120	174	320	46.4	14
	O	170(max)	247(max)	320	46.4	18
	T4	140	203	330	47.5	18
	T4510	140	203	330	47.5	18
	T4511	140	203	330	47.5	18
	T6	285	413	415	59.7	10
	T6510	285	413	415	59.7	10
	T6511	285	413	415	59.7	10
	T6	510	740	560	81.2	9
	T6510	510	740	560	81.2	9
	T6511	510	740	560	81.2	9
	7014	T6	500	72.5	560	81.2
T6510		500	72.5	560	81.2	8
T6511		500	72.5	560	81.2	8
T6		500	72.5	560	81.2	8
7017	T6	370	53.7	420	60.9	11
	T6510	370	53.7	420	60.9	11
	T6511	370	53.7	420	60.9	11
	T6	370	53.7	420	60.9	11
7019	T4	280	40.6	450	65.3	12
	T4510	280	40.6	450	65.3	12
	T4511	280	40.6	450	65.3	12
	T6	450	65.3	480	71.1	11
7020	T4	330	47.5	480	71.1	11
	T4510	330	47.5	480	71.1	11
	T4511	330	47.5	480	71.1	11
	T6	320	46.4	410	59.5	11
7050	T6	320	46.4	410	59.5	11
	T6510	320	46.4	410	59.5	11
	T6511	320	46.4	410	59.5	11
	T6	500	72.5	550	79.8	11
7150	T6	500	72.5	550	79.8	11
	T6510	500	72.5	550	79.8	11
	T6511	500	72.5	550	79.8	11
	T6	500	72.5	550	79.8	11
7075	O	165(max)	239(max)	275(max)	39.9(max)	14
	T6	525	761	590	85.6	7
	T6510	525	761	590	85.6	7
	T6511	525	761	590	85.6	7
	T6	460	667	530	76.9	9
	T6510	460	667	530	76.9	9
	T6511	460	667	530	76.9	9
	T6	440	638	510	74.0	9
	T6510	440	638	510	74.0	9
	T6511	440	638	510	74.0	9
	T6	440	638	510	74.0	9
	7175	O	165(max)	239(max)	275(max)	39.9(max)
T6		525	761	590	85.6	7
T6510		525	761	590	85.6	7
T6511		525	761	590	85.6	7
T6		460	667	530	76.9	9
T6510		460	667	530	76.9	9
T6511		460	667	530	76.9	9
T6		440	638	510	74.0	9
T6510		440	638	510	74.0	9
T6511		440	638	510	74.0	9
T6		440	638	510	74.0	9
L84*		T4	210	30.5	330	47.9
	T6	380	55.1	420	60.9	10
	T6	340	49.3	430	62.4	4
	T6	340	49.3	430	62.4	6
DTD324**	T6	340	49.3	430	62.4	6
	T6	340	49.3	430	62.4	6
	T6	340	49.3	430	62.4	6
	T6	340	49.3	430	62.4	6
DTD5004*	T6	530	76.9	590	84.1	7
	T6	530	76.9	590	84.1	7
	T6	530	76.9	590	84.1	7
	T6	530	76.9	590	84.1	7
DTD5114*	T6	530	76.9	590	84.1	7
	T6	530	76.9	590	84.1	7
	T6	530	76.9	590	84.1	7
	T6	530	76.9	590	84.1	7

\* These alloys do not have a direct international designation equivalent

# ***Appendix II***

**Table 1 Force in X-direction**

Calibration of force sensor by loading and unloading for force in X-direction. Activating channels 2 and 6

LOAD /kg	Channel 2 /mV			Channel 6 /mV		
	Loading	Unloading	Deviation	Loading	Unloading	Deviation
0	0	0.5	0.5	0	0.4	0.4
0.5	31.2	34.6	3.4	32.6	32.4	-0.2
1	66.5	69.7	3.2	63.5	66.5	3
1.5	97.2	101.4	4.2	94.9	98.4	3.5
2	133.7	137.2	3.5	128.1	131.9	3.8
2.5	163.9	165.8	1.9	157.4	162.1	4.7
3	197.3	199.7	2.4	195.1	198.5	3.4
3.5	233.1	237.4	4.3	226.3	230.3	4
4	265.1	265.1	0	259.5	263.3	3.8

**Table 2 Force in Y-direction**

Calibration of force sensor by loading and unloading for force in Y-direction. Activating channels 4 and 8

LOAD /kg	Channel 4 /mV			Channel 8 /mV		
	Loading	Unloading	Deviation	Loading	Unloading	Deviation
0	0	0.6	0.6	0	0.3	0.3
0.5	30.7	33.6	2.9	33.2	34.5	1.3
1	61.5	65.4	3.9	66.5	68.1	1.6
1.5	92.2	96	3.8	100.1	101.2	1.1
2	124.1	130.5	6.4	135.4	137.7	2.3
2.5	154.1	158.4	4.3	167.1	167.6	0.5
3	184.5	190.2	5.7	199.5	200.5	1
3.5	214.9	218.3	3.4	232.8	233.8	1
4	245.6	245.6	0	266.3	266.3	0

**Table 3a Force in Z-direction**

Calibration of force sensor by loading and unloading for force in Z-direction. Activating channels 1 and 3

LOAD /kg	Channel 1 /mV			Channel 3 /mV		
	Loading	Unloading	Deviation	Loading	Unloading	Deviation
0	0	0.2	0.2	0	-1.1	1.1
0.5	16.9	17	0.1	18.3	17.9	0.4
1	33.1	33.3	0.2	38.9	39.4	-0.5
1.5	49.5	49.5	0	56	55.5	0.5
2	66.1	66	-0.1	72.4	71.9	0.5
2.5	82.5	82.3	-0.2	88.7	88.5	0.2
3	98.1	98	-0.1	107.6	107.6	0
3.5	114.7	114.7	0	123.6	123.5	0.1
4	133.3	133.3	0	139.4	139.4	0

**Table 3b Force in Z-direction**

Calibration of force sensor by loading and unloading for force in Z-direction. Activating channels 5 and 7

LOAD /kg	Channel 5 /mV			Channel 7 /mV		
	Loading	Unloading	Deviation	Loading	Unloading	Deviation
0	0	0.3	-0.3	0	3.9	3.9
0.5	18.5	18.4	0.1	17	19.9	2.9
1	38.4	39.3	-0.9	33.3	35.8	2.5
1.5	58.5	58.3	0.2	53.3	55.2	1.9
2	77.9	77.8	0.1	73.5	75.2	1.7
2.5	97.2	97	0.2	93.2	94.7	1.5
3	117.2	117	0.2	110.6	111.7	1.1
3.5	136.8	136.7	0.1	131.8	132.9	1.1
4	154	154	0	152.5	152.5	0



**Table 4 Torque in X-direction**

Calibration of force sensor by loading and unloading for torque in X-direction. Activating channels 5 and 1

LOAD /kg	Channel 5 /mV			Channel 1 /mV		
	Loading	Unloading	Deviation	Loading	Unloading	Deviation
0	0	0.2	-0.2	0	0.3	-0.3
0.5	36.5	36.2	0.3	34.5	33.5	1
1	73.1	72.7	0.4	68.1	66.9	1.2
1.5	109.2	108.9	0.3	101.2	100	1.2
2	145.2	145	0.2	134.7	133.3	1.4
2.5	181.1	180.5	0.6	167.6	169.4	-1.8
3	216.7	216.2	0.5	200.5	201.2	-0.7
3.5	252	251.6	0.4	233.8	234.6	-0.8
4	291.7	291.7	0	268.7	268.7	0

**Table 5 Torque in Y-direction**

Calibration of force sensor by loading and unloading for force in Z-direction. Activating channels 3 and 7

LOAD /kg	Channel 3 /mV			Channel 7 /mV		
	Loading	Unloading	Deviation	Loading	Unloading	Deviation
0	0	2.3	-2.3	0	0.4	-0.4
0.5	31.4	29.5	1.9	36.5	36	-0.5
1	66.1	64.4	1.7	73.2	72.6	0.6
1.5	100.5	99.3	1.2	109.4	109	0.4
2	134.8	133.6	1.2	145.8	145.3	0.5
2.5	168.7	168	0.7	181.8	181.3	0.5
3	202.6	201.9	0.7	217.4	217.1	0.3
3.5	236.7	235.8	0.3	252.9	252.7	0.2
4	270.7	270.7	0	289.3	289.3	0

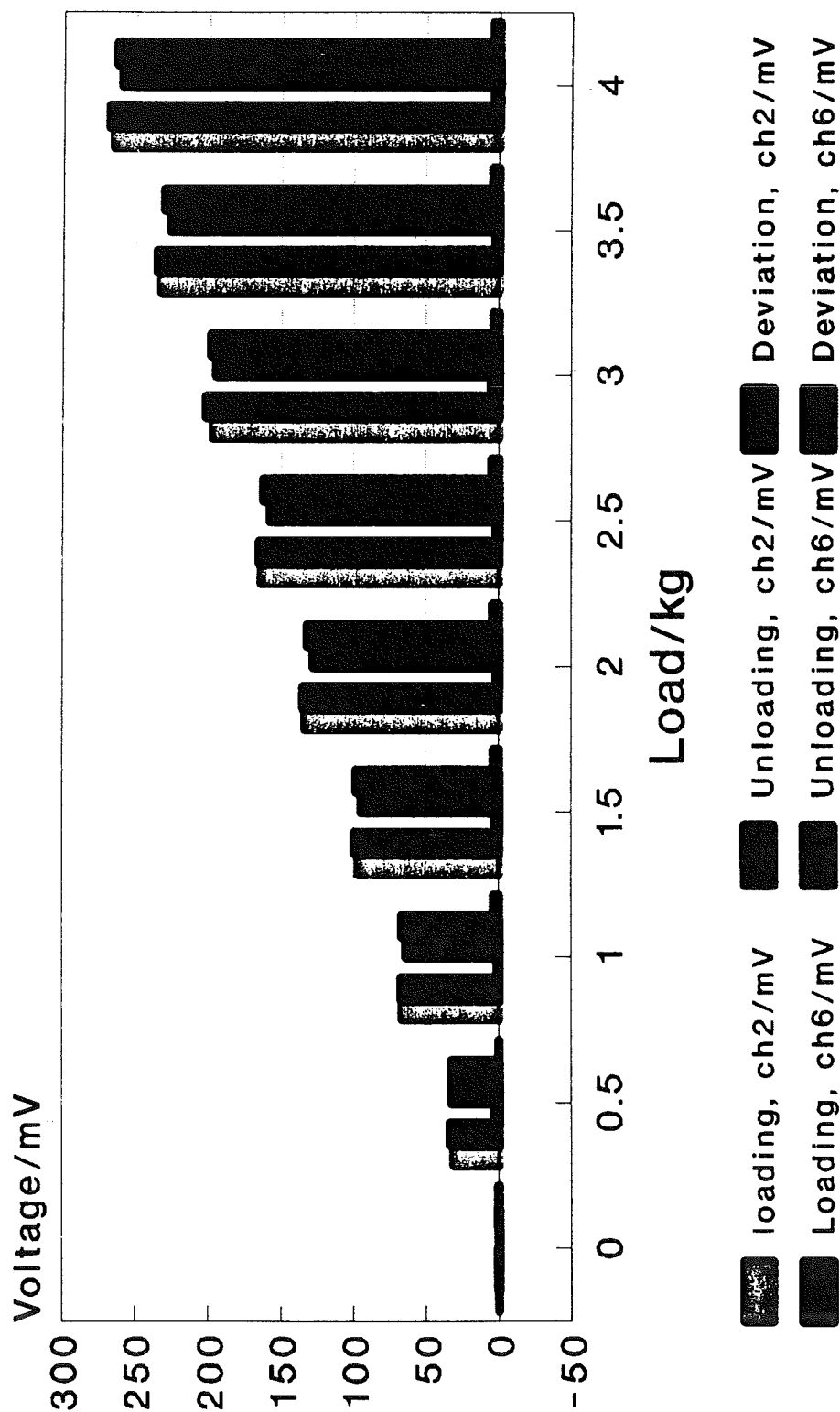
**Table 6a Torque in Z-direction**  
 Calibration of force sensor by loading and unloading for torque in Z-direction.  
 Activating channels 2 and 4

LOAD /kg	Channel 2 /mV			Channel 4 /mV		
	Loading	Unloading	Deviation	Loading	Unloading	Deviation
0	0	0.3	0.3	0	0.2	0.2
0.5	12.5	13.6	1.1	20.6	20.2	-0.4
1	26.8	28	1.2	41.2	41.3	0.1
1.5	39.8	41.6	1.8	61.6	61.8	0.2
2	53	54.7	1.7	82	82.2	0.2
2.5	67.2	69.5	2.3	102.5	102.4	-0.1
3	81.5	82.8	1.3	122.5	122.7	0.2
3.5	94.6	96.5	1.9	141.8	142.5	0.7
4	108.4	108.4	0	162.1	162.1	0

**Table 6b Torque in Z-direction**  
 Calibration of force sensor by loading and unloading for torque in Z-direction.  
 Activating channels 6 and 8

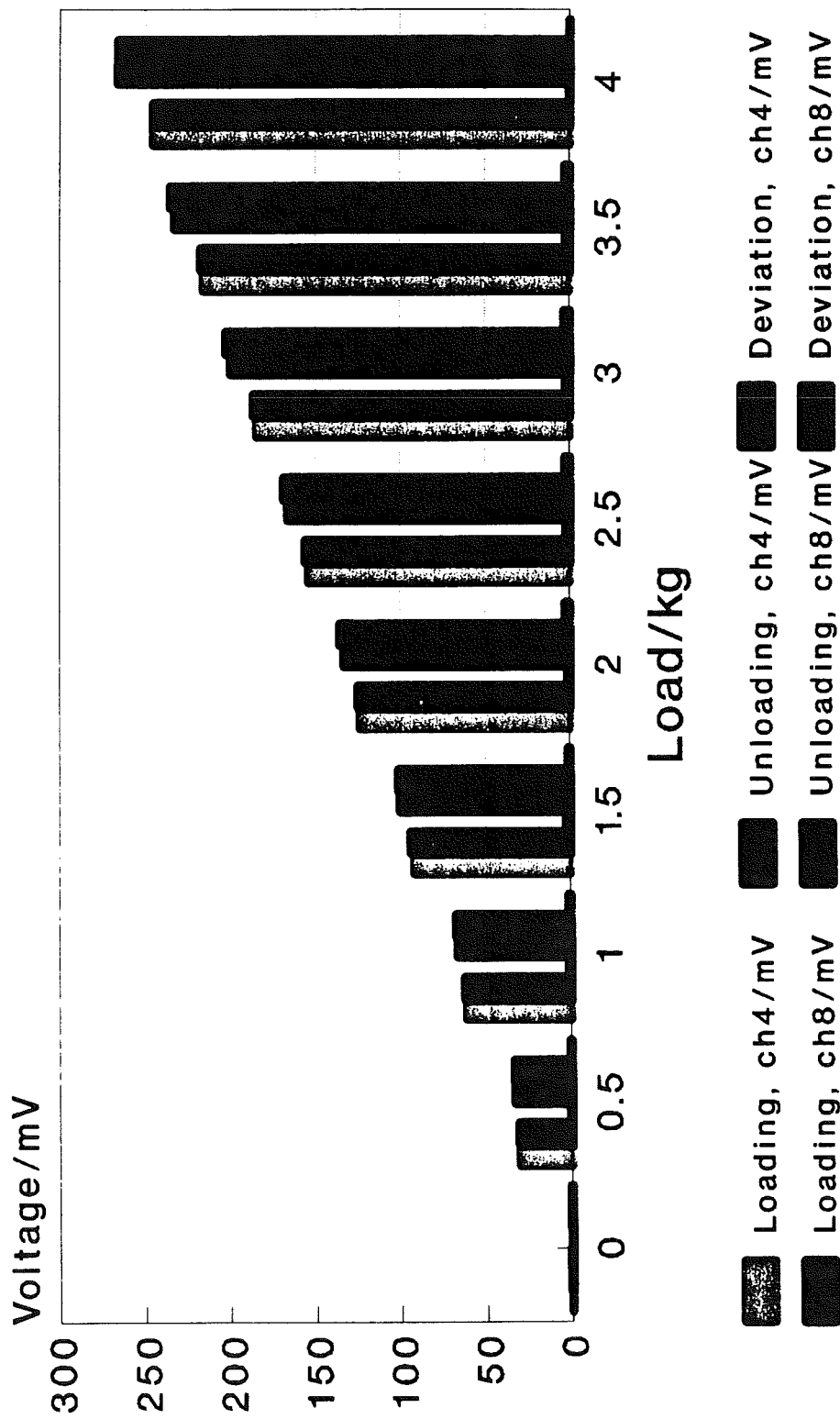
LOAD /kg	Channel 6 /mV			Channel 8 /mV		
	Loading	Unloading	Deviation	Loading	Unloading	Deviation
0	0	0.4	0.4	0	0.4	0.4
0.5	19.4	20.3	0.9	22.4	22.7	0.3
1	39.3	40.4	1.1	44.9	45.4	0.5
1.5	59.2	60.2	1	66.9	68.4	1.5
2	79	80	1	89.1	89.6	0.5
2.5	99.1	99.5	0.4	114.4	114.1	2.7
3	118.6	119.1	0.5	132.8	133.3	0.5
3.5	137.3	138.3	1	154.3	155.2	0.9
4	156.9	156.9	0	176.2	176.2	0

# Hysteresis effect, force in X-direction bridges/channels 2,6



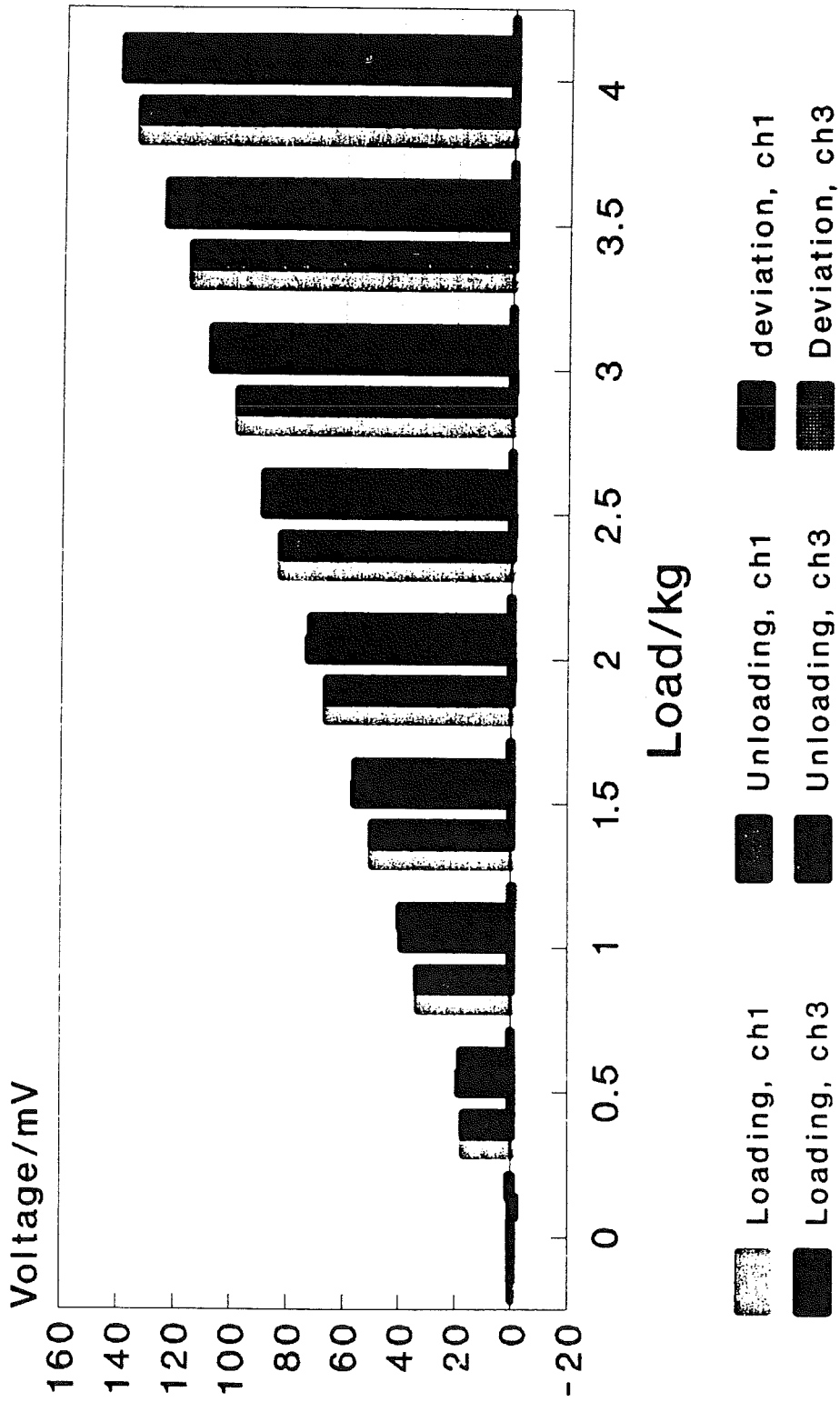
Loading increment 0.5

# Hysteresis effect, force in Y-direction Bridges/channels 4,8



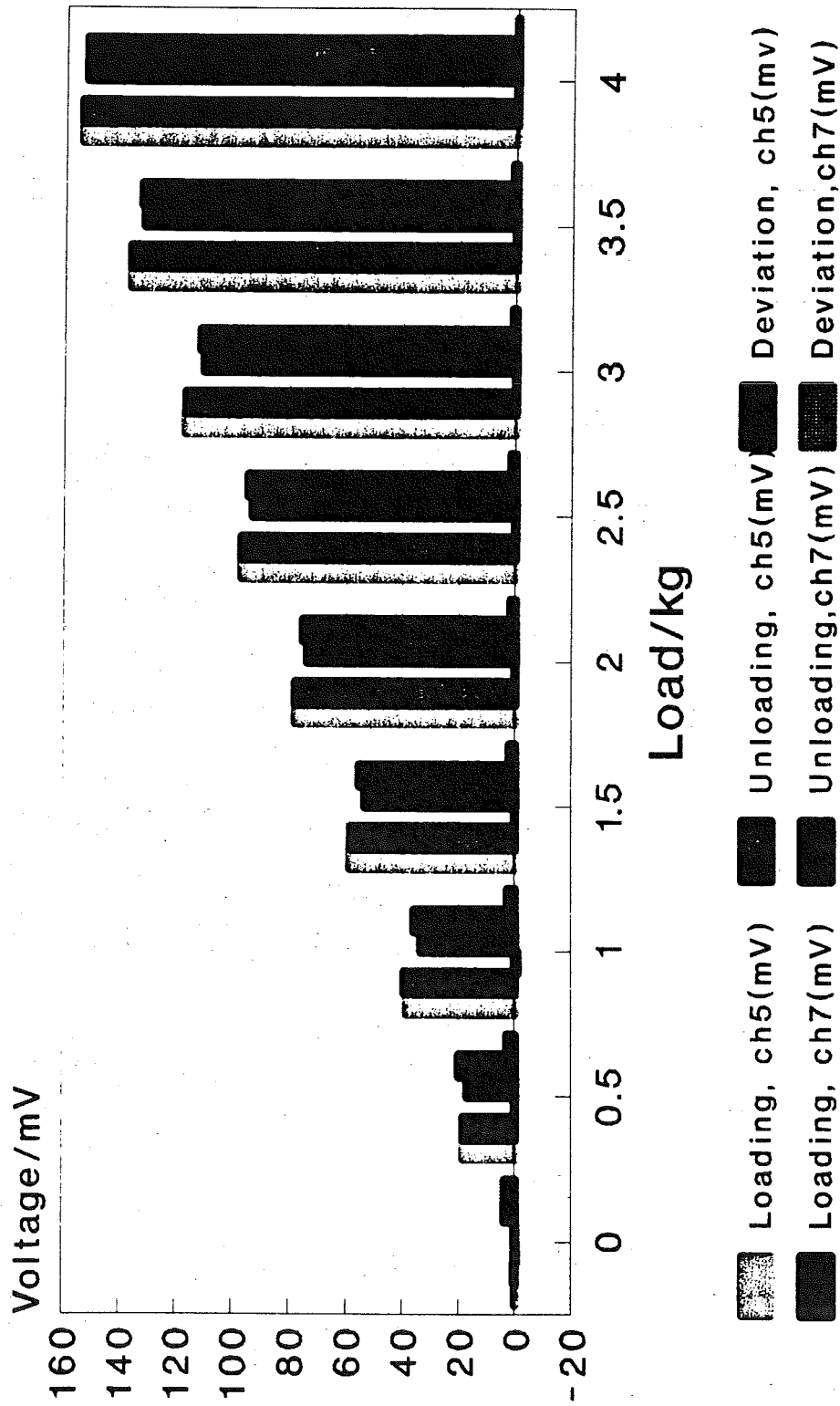
Loading increment 0.5 kg

# Hysteresis effect, Force in Z-direction bridges 1 & 3



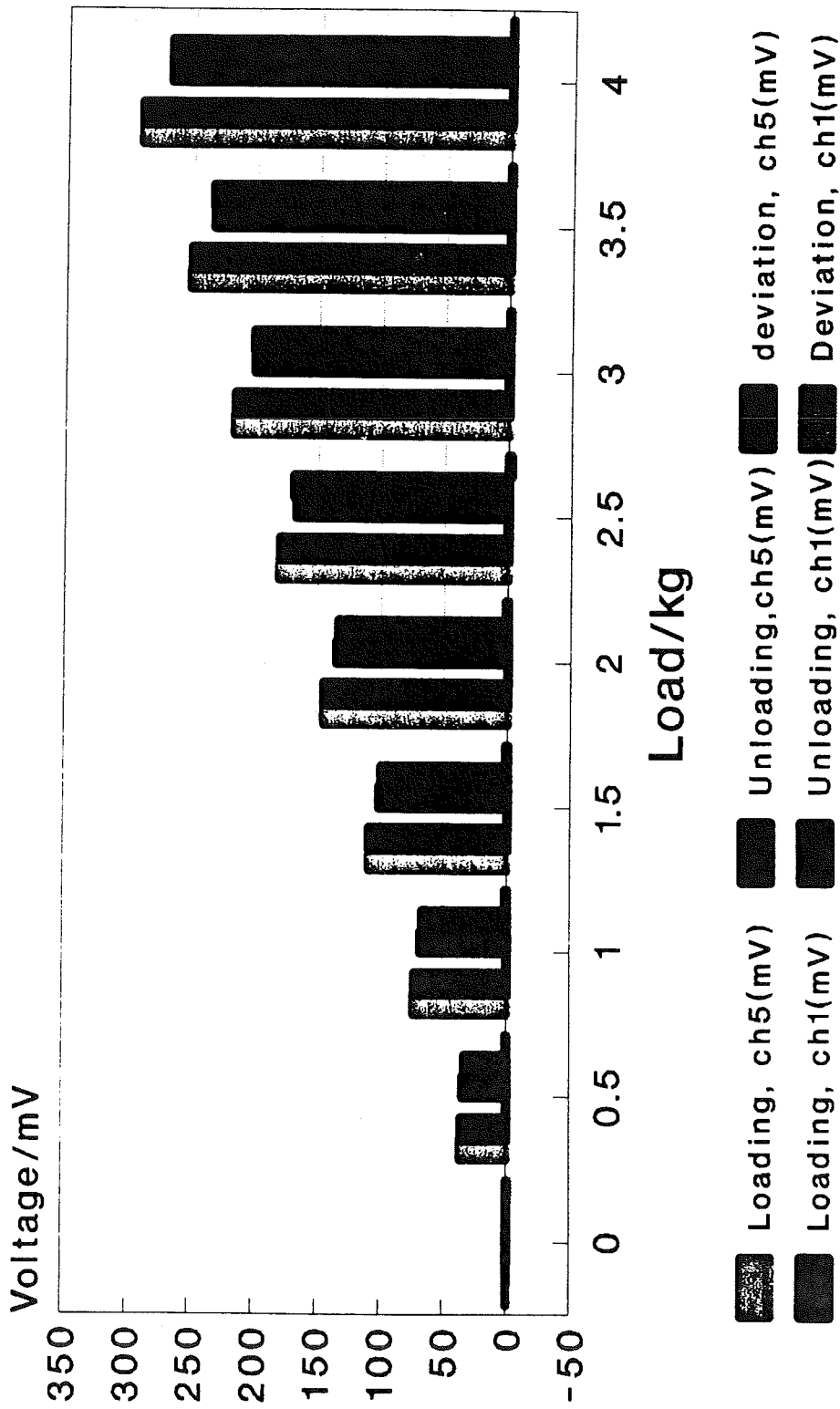
Loading increment 0.5/kg

# Hysteresis effect, Force in Z-direction Bridges/channels 5,7



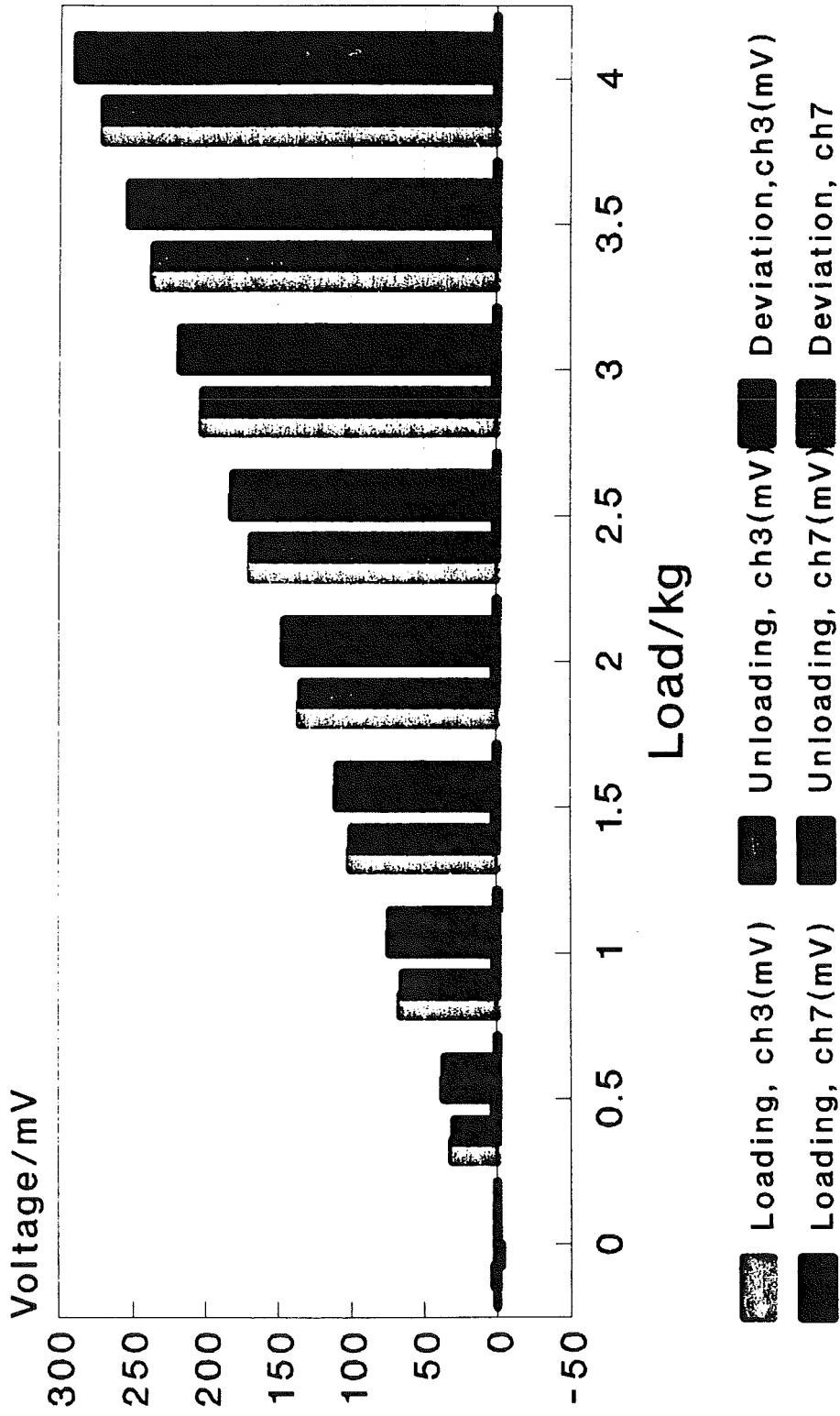
Loading increment 0.5

# Hysteresis effect, Torque in X-direction bridges 5 & 1



Load increment 0.5 kg

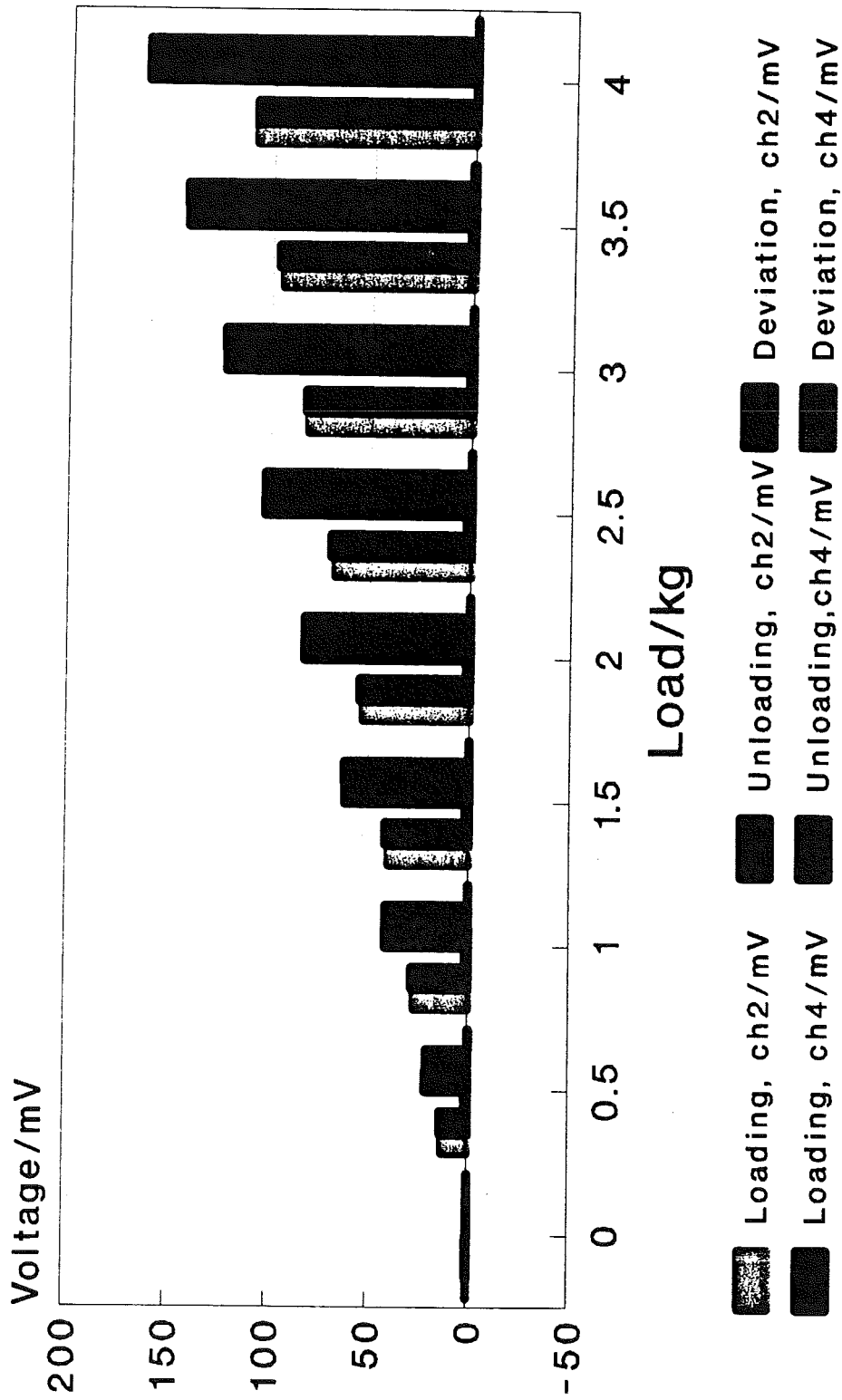
# Hysteresis effect, Torque in Y-direction Bridges/channels 3 & 7



Loading increment 0.5 kg

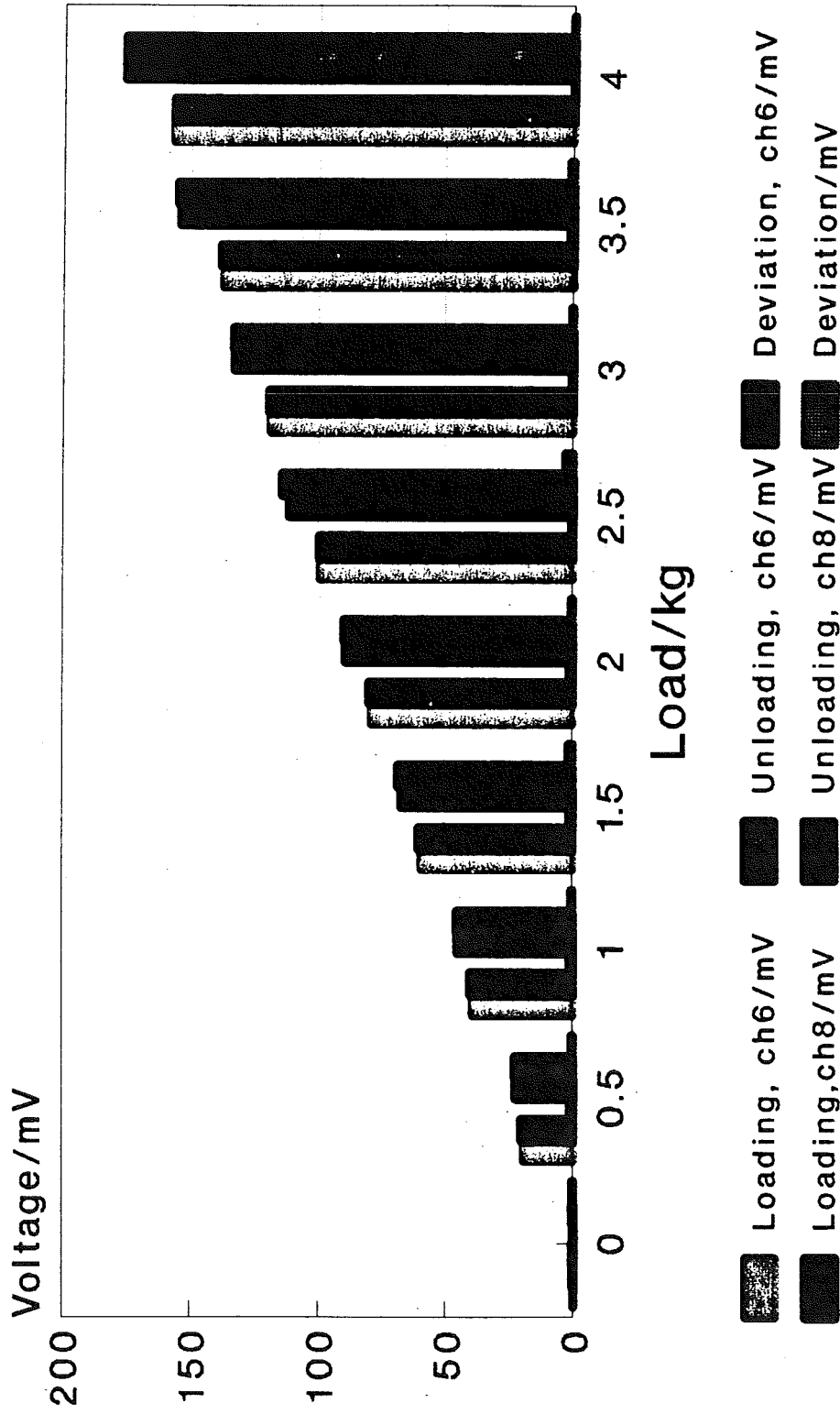


# Hysteresis effect, torque in Z-direction Bridges/channels 2,4



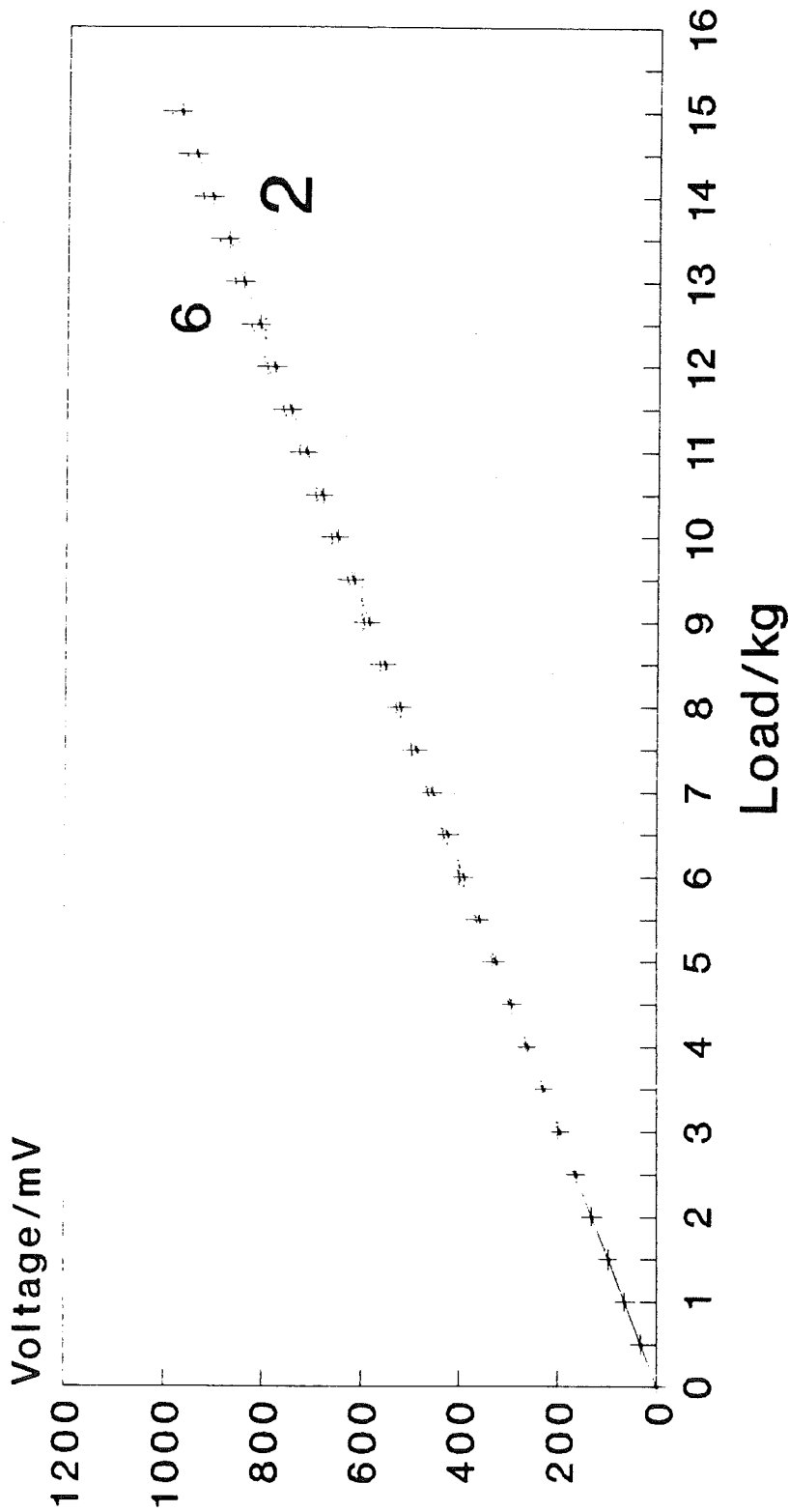
Loading increment 0.5 kg

# Hysteresis effect, torque in Z-direction Bridges/channels 6,8



Loading increment 0.5 kg

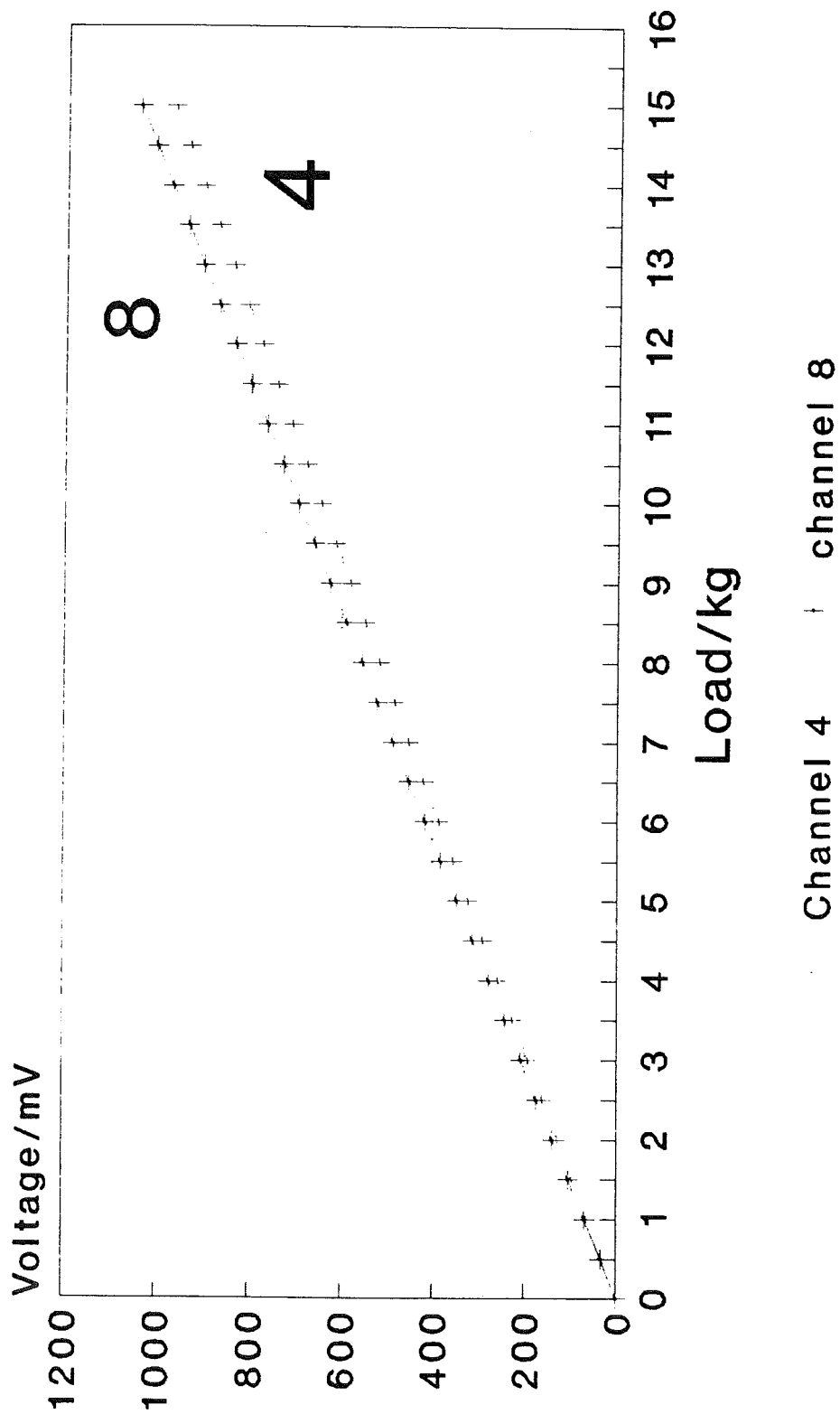
# Force in X-direction bridges/channels 2,6



Channel 2    +    channel 6

Loading increment 0.5/kg

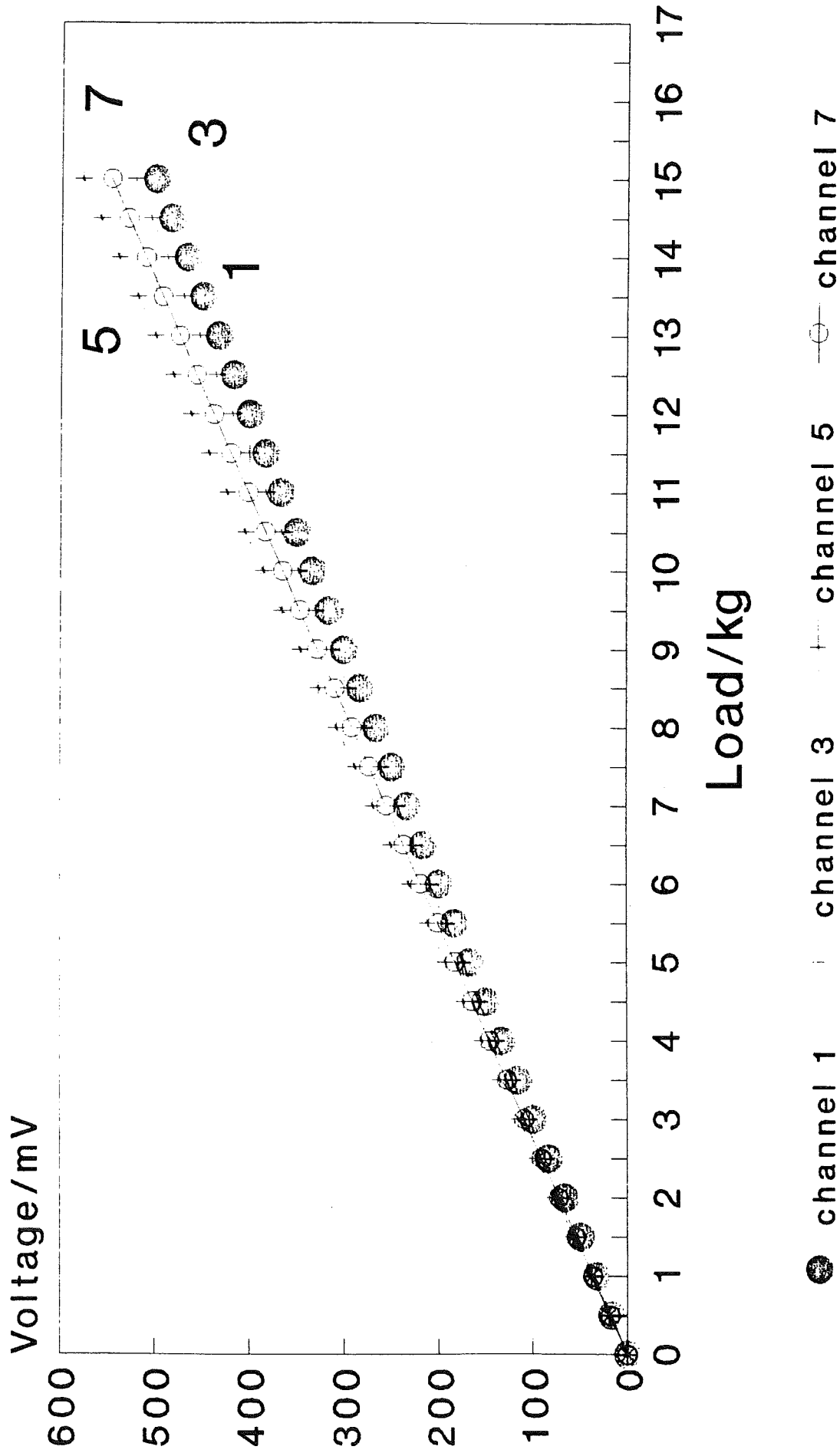
# Force in Y-direction bridges/channels 4,8



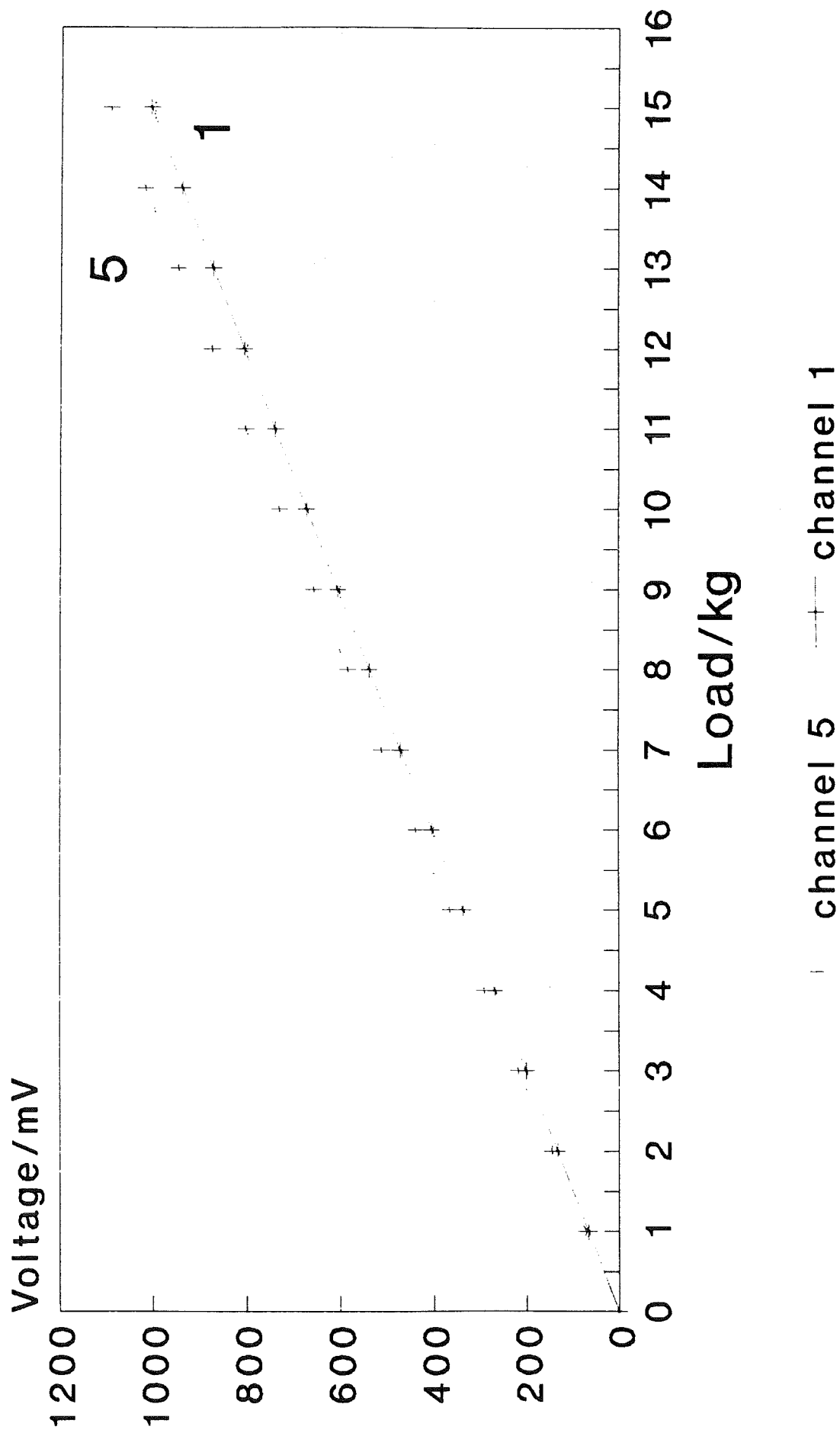
Loading increment 0.5/kg

# Force in Z-direction

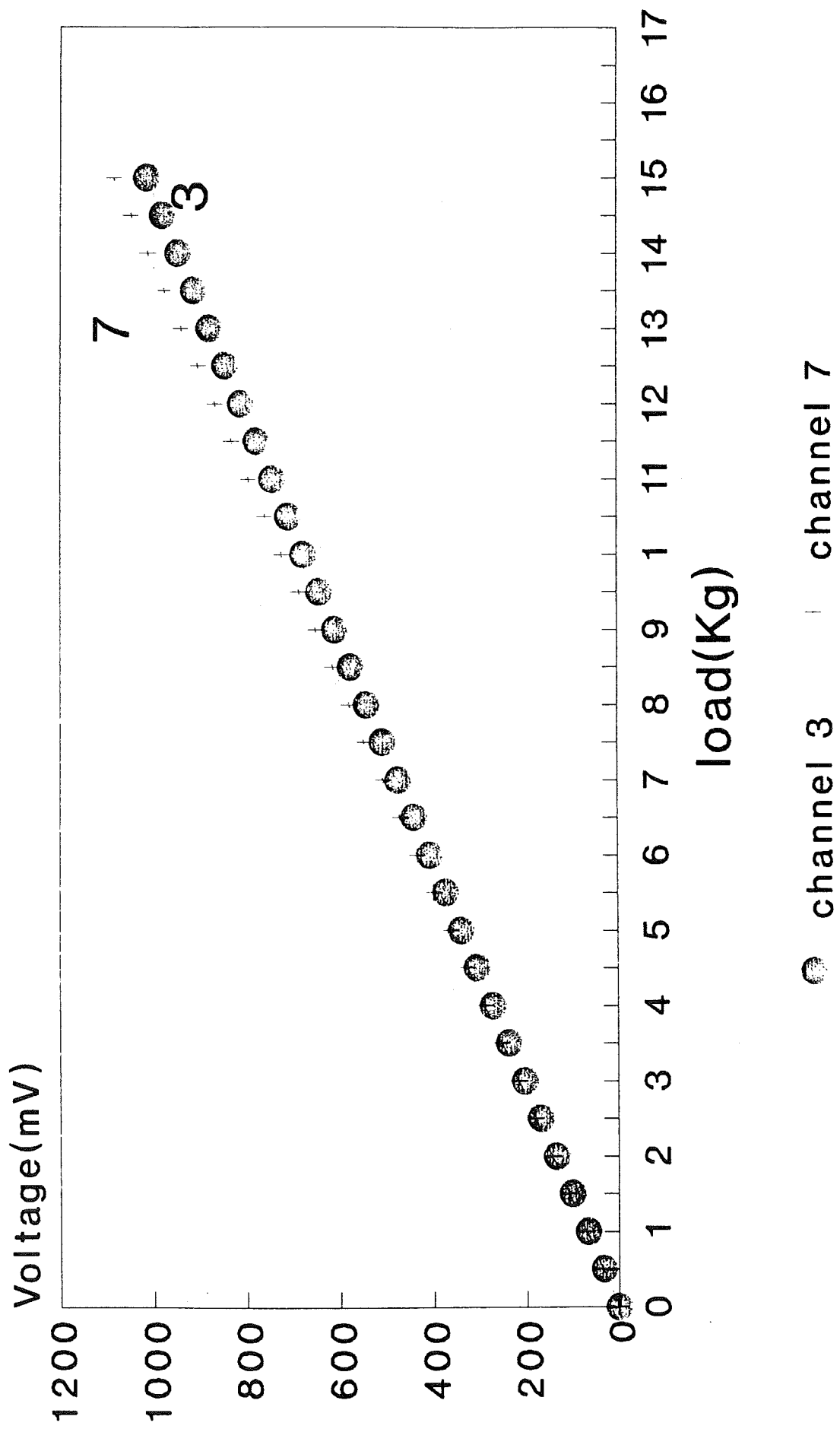
## channels 1,3,5,7



# Torque in X-direction channels 5 & 1

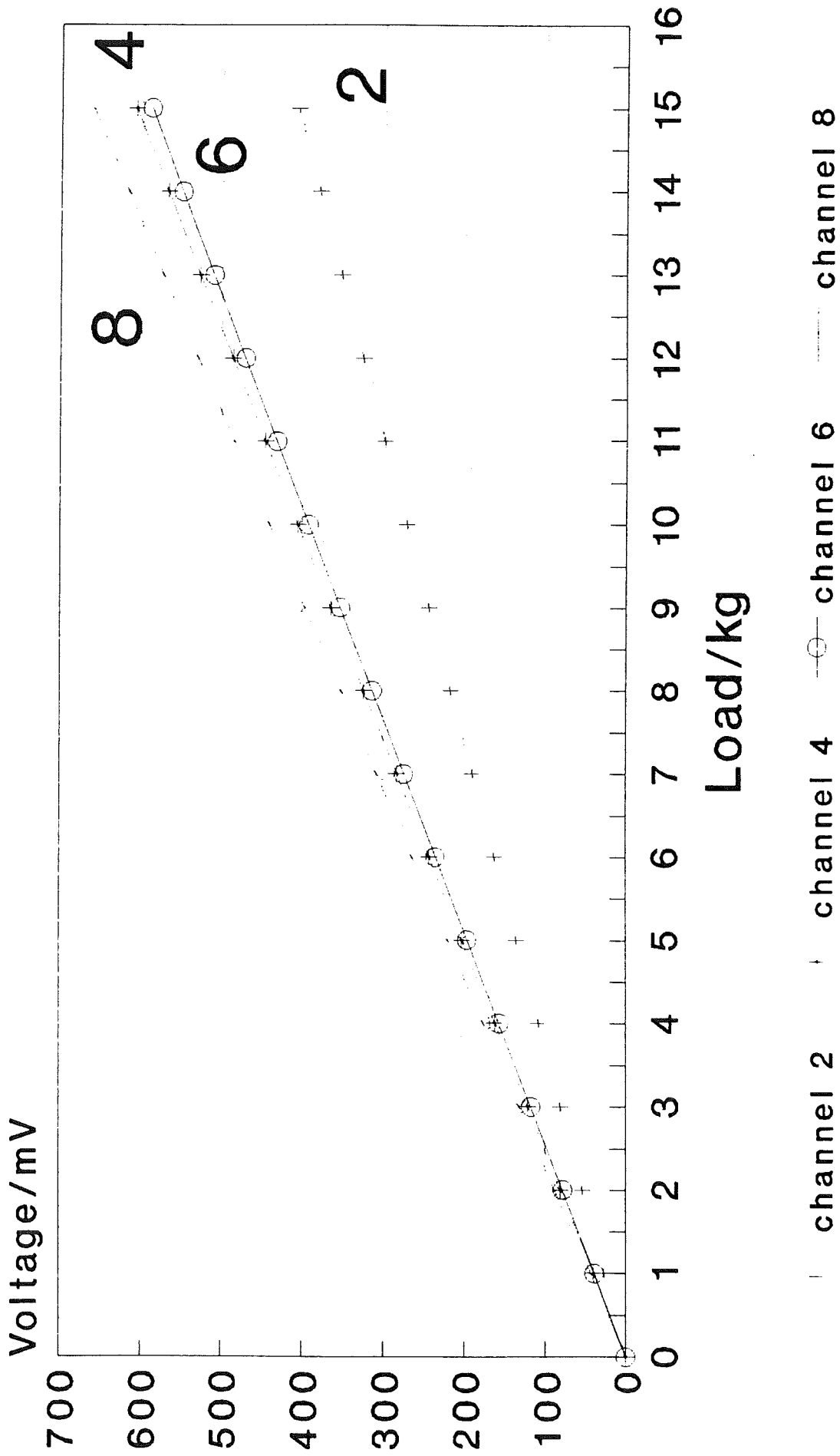


# Torque in Y-direction channels 3 & 7



# Torque in $\angle$ -direction

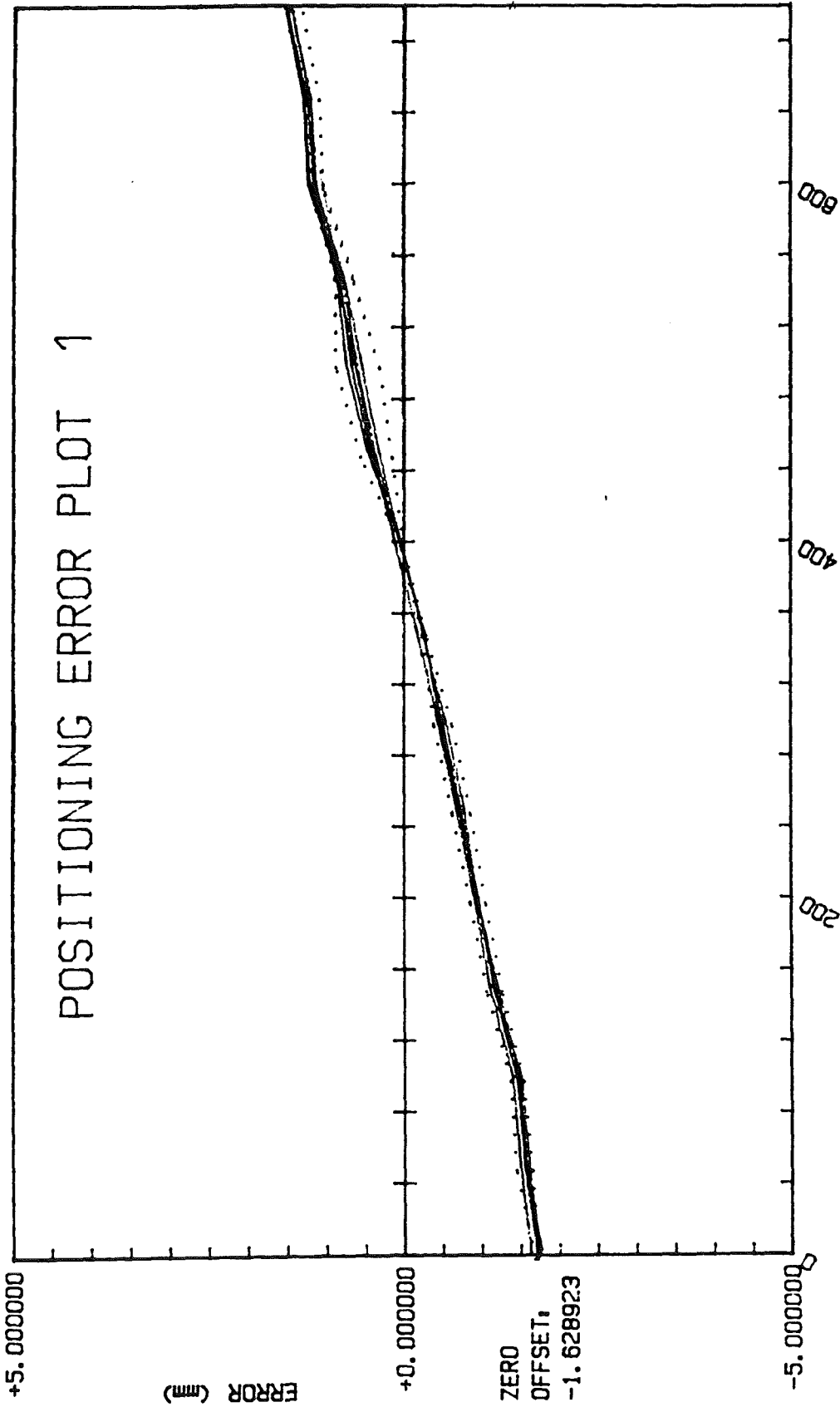
## channels 2,4,6,8





# ***Appendix III***

# POSITIONING ERROR PLOT 1

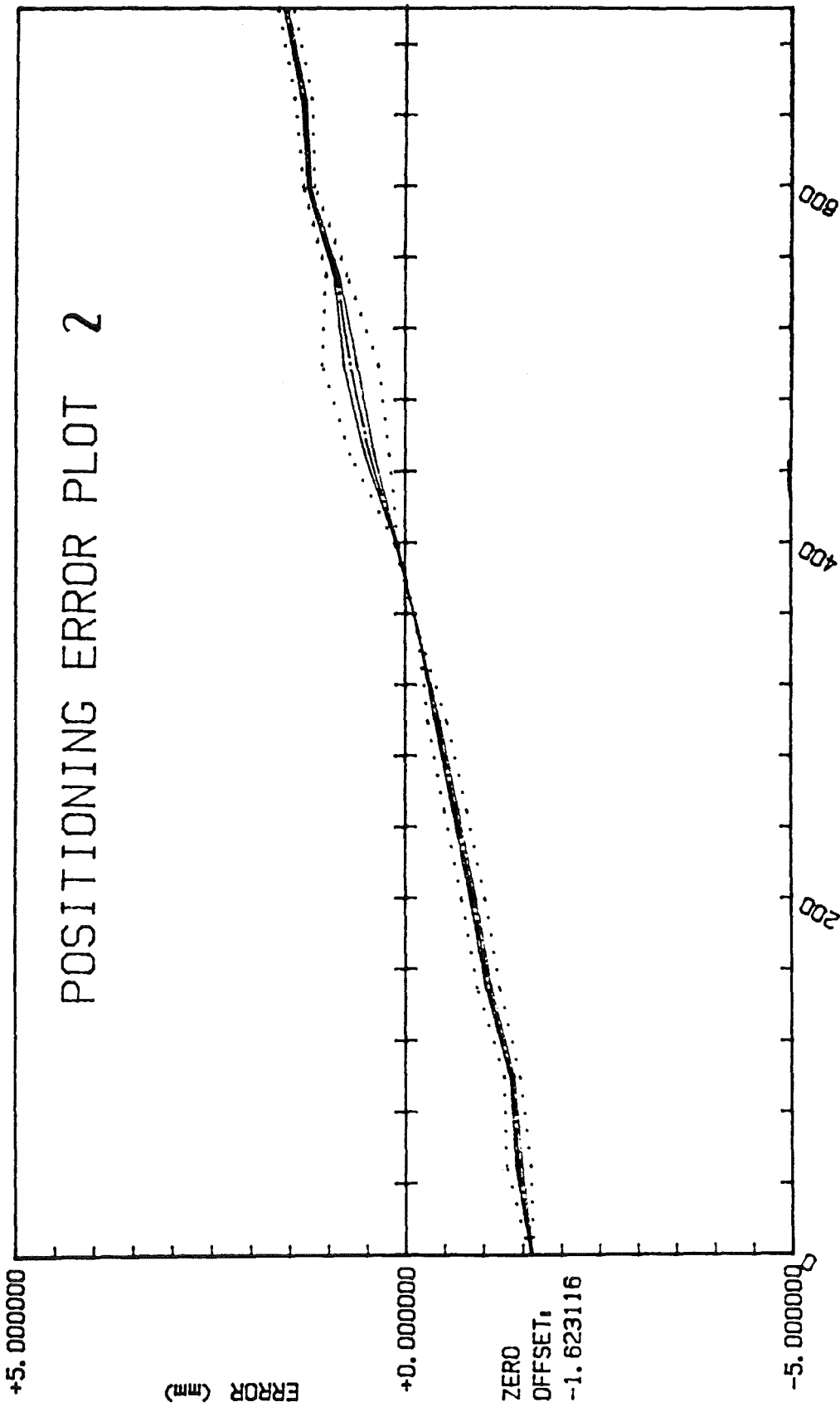


MACHINE: PUMA  
 NUMBER: 560  
 DATE: 8/10/87  
 AXIS: X

LOCATION: DRS

+/- 3 SIGMA  
 ERROR BAND: 3.296073  
 NON-REPEAT: .560472

# POSITIONING ERROR PLOT 2



MACHINE: PUMA NUMBER: 560 DATE: 8/10/87 AXIS: X	LOCATION: DRS	+/- 3 SIGMA ERROR BAND: 3.284172 NON-REPEAT: .730583
--	---------------	--

## **Prototype Manufacture**

### **Central Support Tube**

Since the internal surface was to form a bearing surface for the Z-translation tube it had to be made concentric. To achieve this the tube was mounted in a four jaw chuck on a centre lathe and 'clocked' with a dial gauge indicator firstly around the diameter close to the chuck, and secondly along the tube length parallel to the bedway. Before the tube was mounted in the lathe, however, a PVC plug was made. This prevented the tube being distorted by the jaws of the chuck.

A very fine cut was taken on the diameter of the tube at the end of the tube furthest away from the chuck itself. This was to insure the outside diameter of the tube was concentric. A steady could then be located on this concentric diameter which restricted side deflections as a result of the boring operation as well as reducing the tendency to chatter or for the tube to 'sing'.

The slot profiles were then produced on the Cincinnati Cintimatic CNC milling machine. A five-axis milling machine was used to enable profiles to be cut in the tube at 90° and 180° apart. This was achieved by mounting a dividing head on the bed. It was important to insure that the dividing head was square in the horizontal and vertical planes. Additionally the CNC machine offsets had to be set and this was achieved by clocking the diameter of the tube with the dial gauge mounted in the CNC machine chuck. To eliminate deflections of the tube when cutting commenced, 'V' blocks were positioned under the free end of the tube.

### **Z-Translation Sliding Tube**

The Z-translation tube was produced in a similar fashion to that of the central tube except that the outside diameter was made concentric with the central tube support. During manufacture both ends of the tube were plugged to reduce vibration effects.

### **Y-Translation Tube**

This was produced as for the Z-translation tube but for this the square hole was produced using a CNC wire eroder.

### **Y-Translation Support Bracket**

The support bracket hole for the Y-translation tube was bored on a centre lathe. An Aluminium block was mounted in the lathe and successive diameters were drilled to within a few millimetres of the required diameter and the final cut was

completed by boring. Working from this hole it was possible to mill the required the required dimensions. In order to achieve maximum accuracy the block should be roughly milled to shape and finished by grinding.

The holes for the X-translation drive were spotted on a milling machine, drilled and tapped. The square hole was produced on the wire eroder as part of a sub-assembly as follows.

#### **Y-Translation Mounting Bracket Assembly**

With the respective parts manufactured, as described above, it was then possible to produce the square hole. This was achieved by locating the Y-translation tube and mounting bracket together. Holes were spotted through the mounting bracket into the tube and screws were used to fasten the two together. It was then necessary to produce a hole in the centre of the bracket and tube so that the wire of the wire eroder could be passed through. The hole had to be central to the tube and this was produced on the Cincinnati CNC machine by clocking off the tube diameter.

The assembly was then mounted in the wire eroding machine and, in order to make sure the square hole was square to the tube, the tube was clocked in both the horizontal plane and in the vertical plane.

#### **Z-Translation Motor Mounting Bracket**

This was made from a solid bar Aluminium alloy stock. The bar, with faced off ends from a centre lathe, was mounted in the Cincinnati CNC milling machine. The profile was then cut and the motor mounting holes and the thrust plate spacer holes drilled. The bracket was then sawn off from the stock material and faced up in a lathe.

#### **Thrust Plates, X and Y Motor Mounting Brackets**

These were made from rectangular bar stock. The appropriate strip lengths were cut from the material and two sides were milled square to each other for each plate. This was to enable the plates to be secured in the vice of the Cincinnati machine. The holes were then drilled in accordance with the drawings.

#### **Spacers**

It was important to insure the spacers were of consistent lengths. To achieve this, the spacers were made on the Antares CNC lathe in accordance to the drawings specified in appendix {3}.

#### **Macro-manipulator Mounting Bracket**

The opposite edges of the two plates were milled together so that the plates

could be secured in the vice of the Cincinnati CNC machine. Fixing holes were then spotted through and the plates separated. Clearance holes were then drilled in the top plate and the holes in the bottom plate were tapped. The two plates were then secured together by the screws. This enable the profile to be cut and the holes produced in accordance to the drawings in Appendix {3}.

#### **Macro-manipulator mounting Arms**

These were manufactured from Aluminium alloy tubing. The appropriate lengths were cut and the tubes were then bored to suit a mild steel insert at each end. The inserts were interference fits and, once these were in place, the arm was replaced in the lathe and the mounting bracket hole(M8) was drilled and tapped. The arms were then mounted in the Cincinnati CNC machine and the central tube fixing holes were drilled and tapped(4BA). The arms were placed in a vertical milling machine, centred with respect to the cutter, and the radius was produced in accordance to the drawings in Appendix {3}.

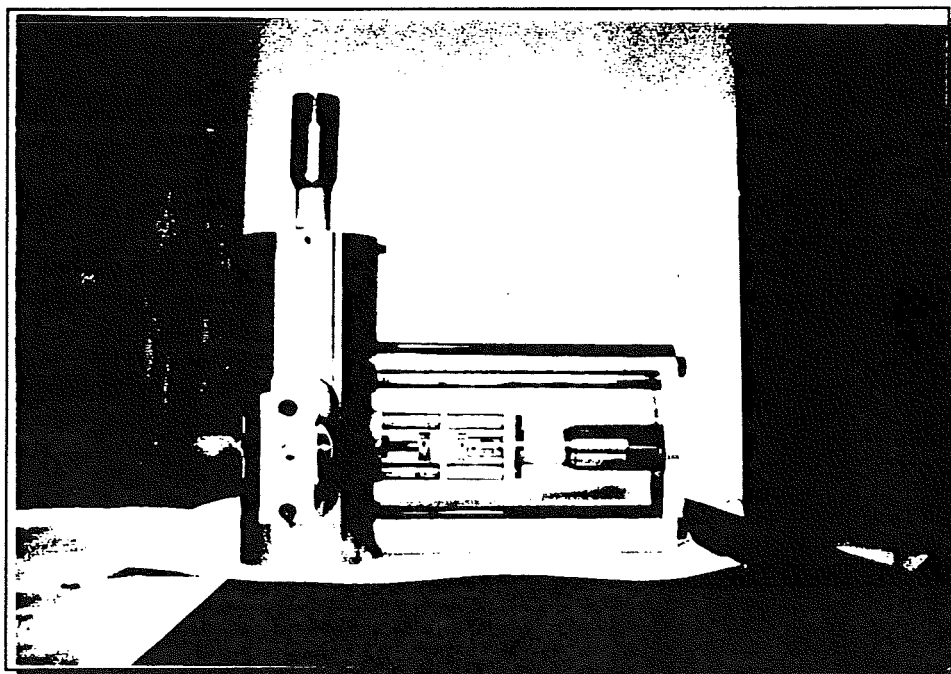
### **Prototype Assembly Instructions**

- 1- Secure macro-manipulator mounting arms to the central tube using the access holes, shown in Figure 5.4, to screw the 4BA screws into the mounting arm steel inserts.
- 2- Secure the threaded bar into the Z-translation tube using the collars. Lubricate the Z-translation tube with Rocol anti-scuffing paste and slide the tube into the central mounting tube. Ensure the narrow horizontal slot of the inner tube is facing forward and aligns with the slot in the central mounting tube.
- 3- Feed the Y-translation mounting bracket (Figure 5.3) through the slot at the back of the central tube aligning with the holes for the Y-translation tube in the Z-translation tube.
- 4- Slide the Y-translation tube, Figure 5.3, through the Z-translation tube and the Y-translation mounting bracket. Align the holes in the centre of the bracket and secure the mounting bracket and Y-translation tube together with the M5 bolts and nuts from above and below. An allen key can be located through the centre of the threaded bar in the Z-translation tube to screw the bracket and tube together whilst the M5 nut is held by long nose pliers.
- 5- Secure the threaded bar in the Y-translation tube using the collars.
- 6- Attach the Y-translation spacers to the Z-translation tube. Secure the thrust plate to the spacer using other spacers. Locate a plain bearing nut onto the threaded drive shaft and pass the threaded drive through the plate. Secure the other plain bearing nut to the drive shaft and having greased the thrust plate secure the plain bearings in place by their grub screws. Rotate the drive shaft into the threaded bar in the Y-translation tube. Attach the flexible coupling to the drive shaft and secure the motor to the motor mounting bracket via the three M2 bolts. Locate the motor spindle into the flexible coupling and secure the motor mounting bracket to the thrust plate spacers using the M5 bolts.
- 7- The X-translation drive mechanism is assembled as above.
- 8- The Z-translation drive mechanism is assembled as above and as illustrated in Figure 5.3. To secure the mechanism to the Z-translation tube either power up the motor as to rotate the drive shaft into the threaded bar or rotate the whole assembly until the top mounting bracket holes align with the holes in the central tube.
- 9- Feed the end effector through the slots in front of the micro-manipulator and rotate the X-translation drive shaft until the end-effector is attached to the

drive mechanism.

10- Secure the macro-manipulator mounting bracket to the mounting arms via the M8 bolts.

Figures A1, A2, A3 illustrate the assembled micro-manipulator at different angles.



**Figure A1** Side elevation, showing plain bearing drive mechanism.



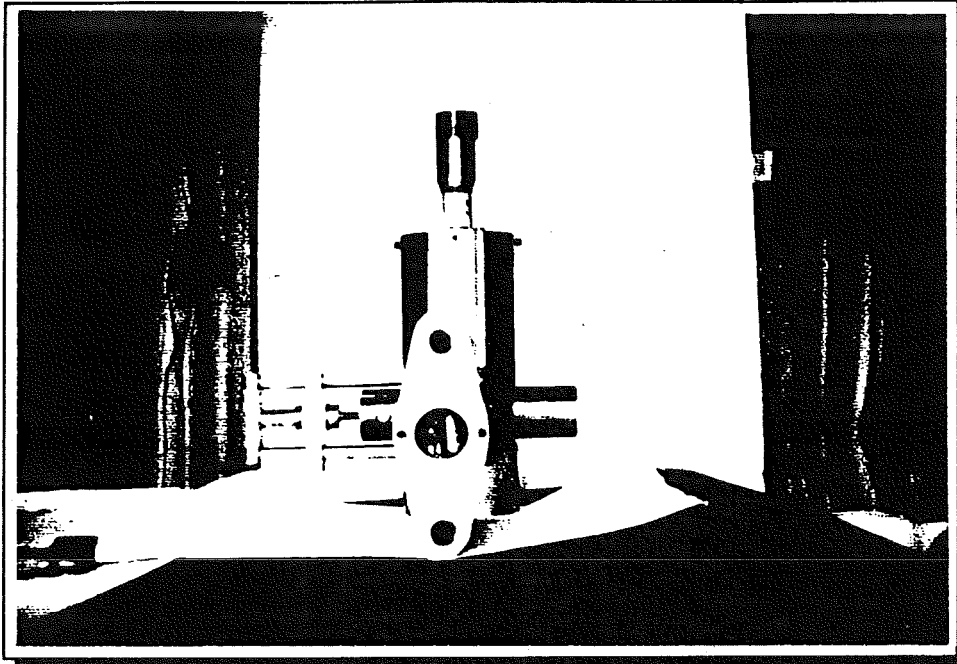


Figure A2 End-elevation showing roller thrust bearing drive mechanism.

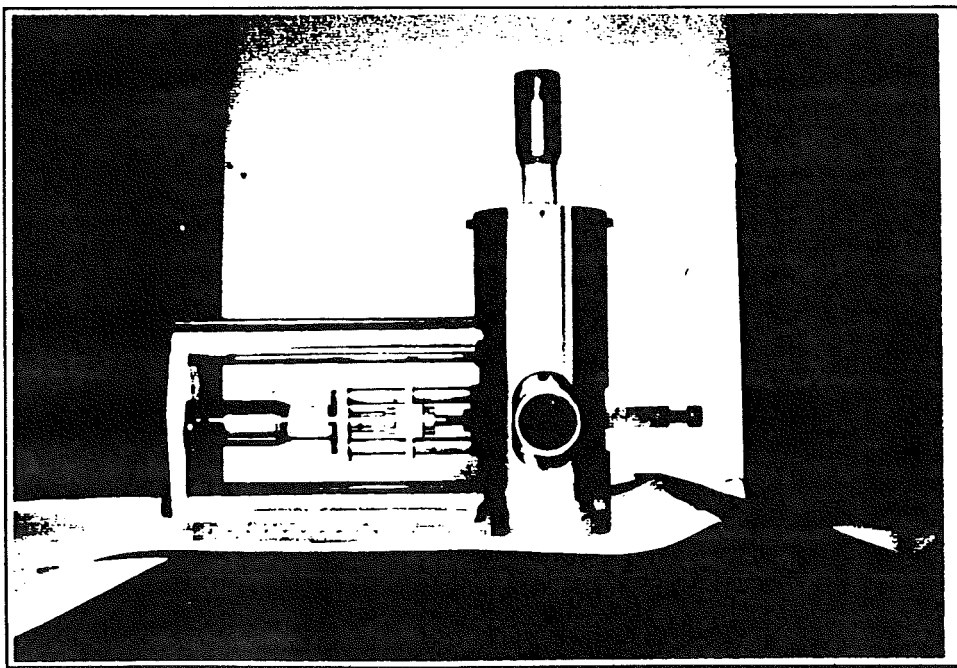
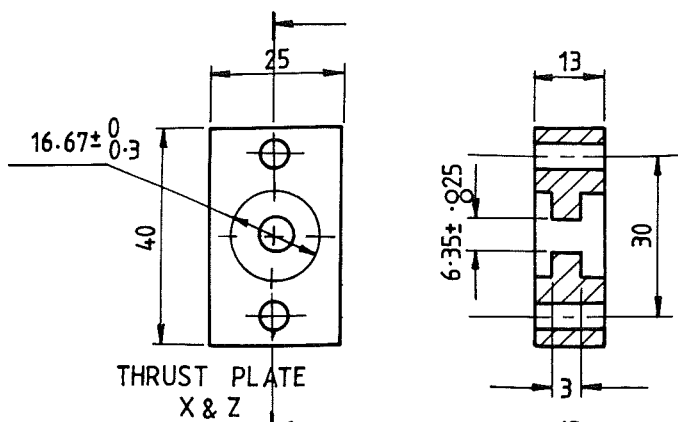
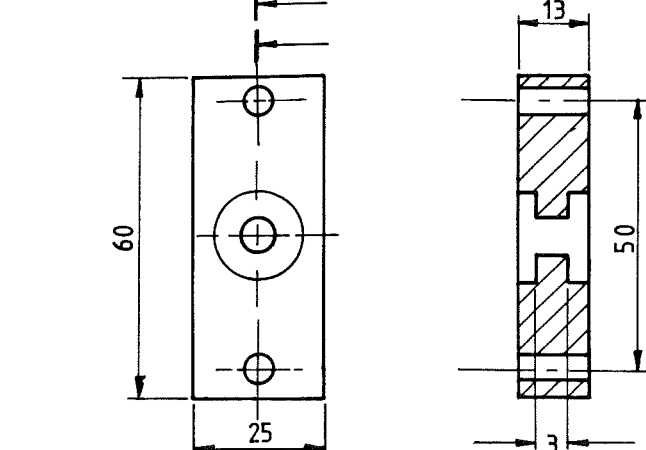


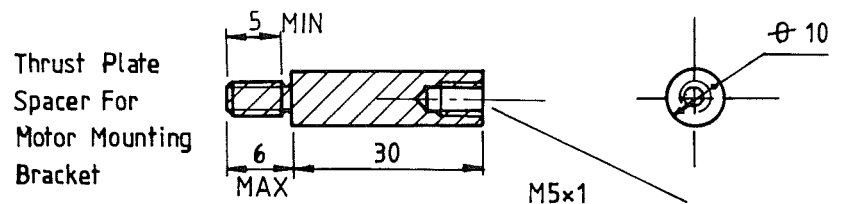
Figure A3 Side-elevation.



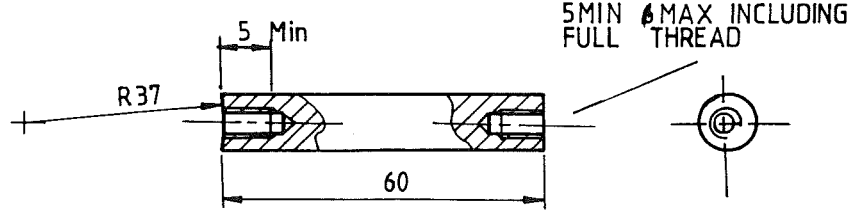
THRUST PLATE  
X & Z



Y Thrust Plate

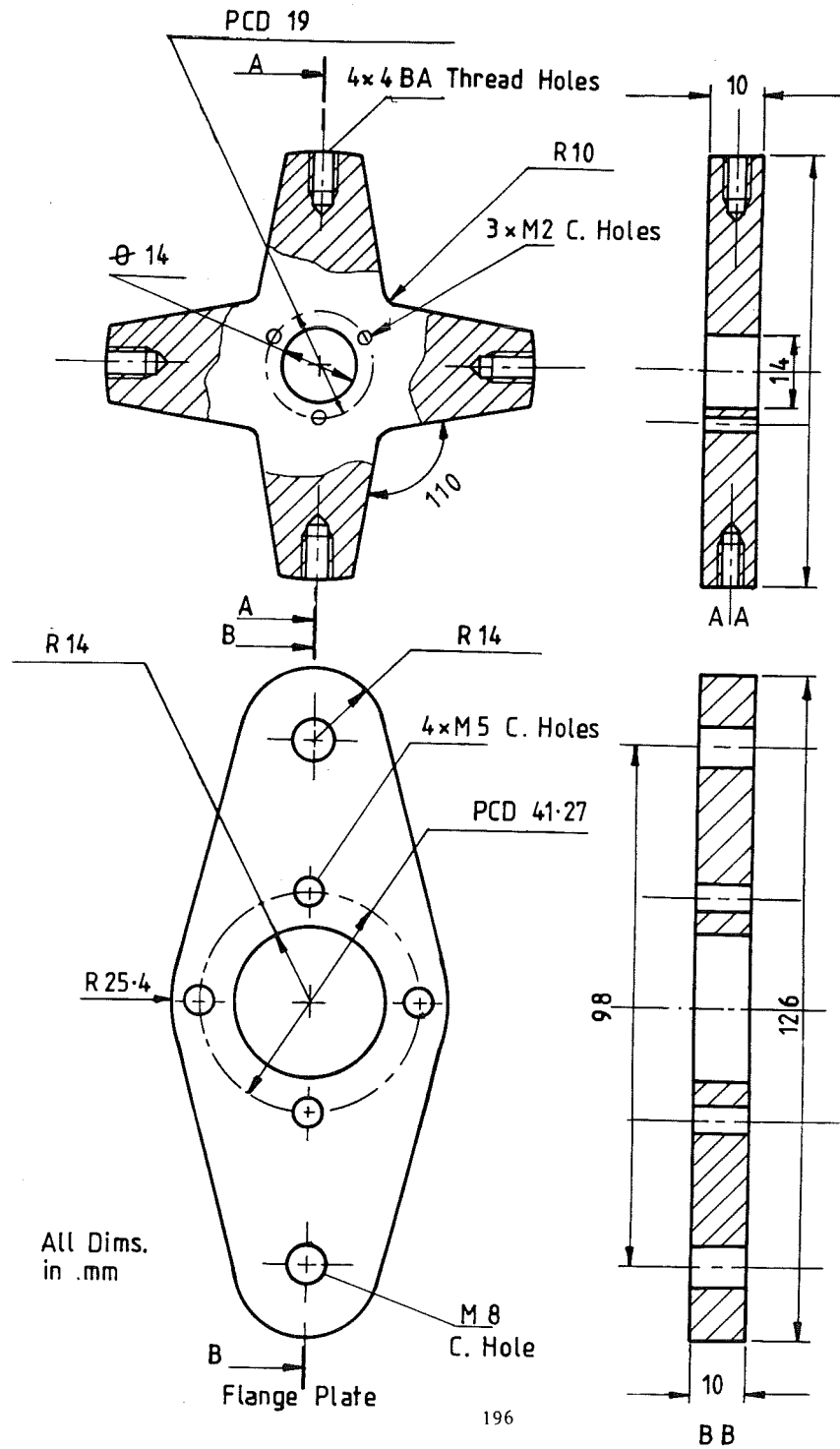


Thrust Plate  
Spacer For  
Motor Mounting  
Bracket

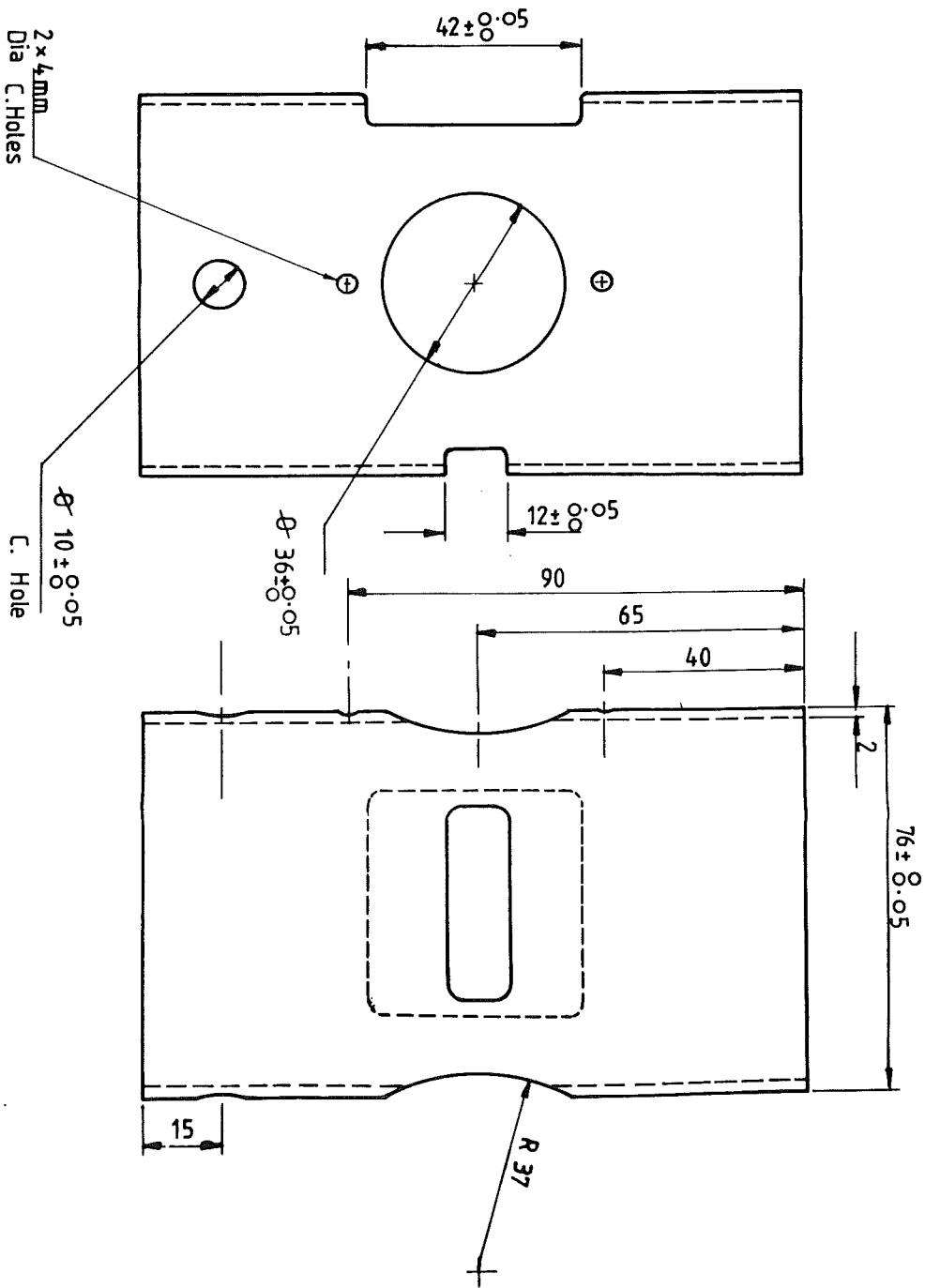
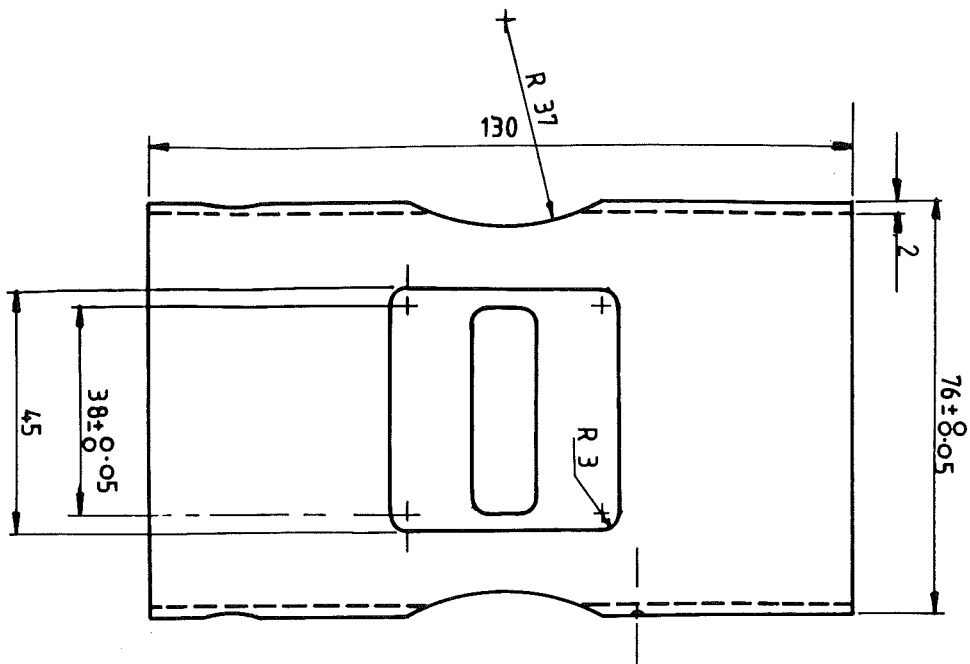


Y Thrust Plate Spacer

### Top Z-translation Mounting Bracket





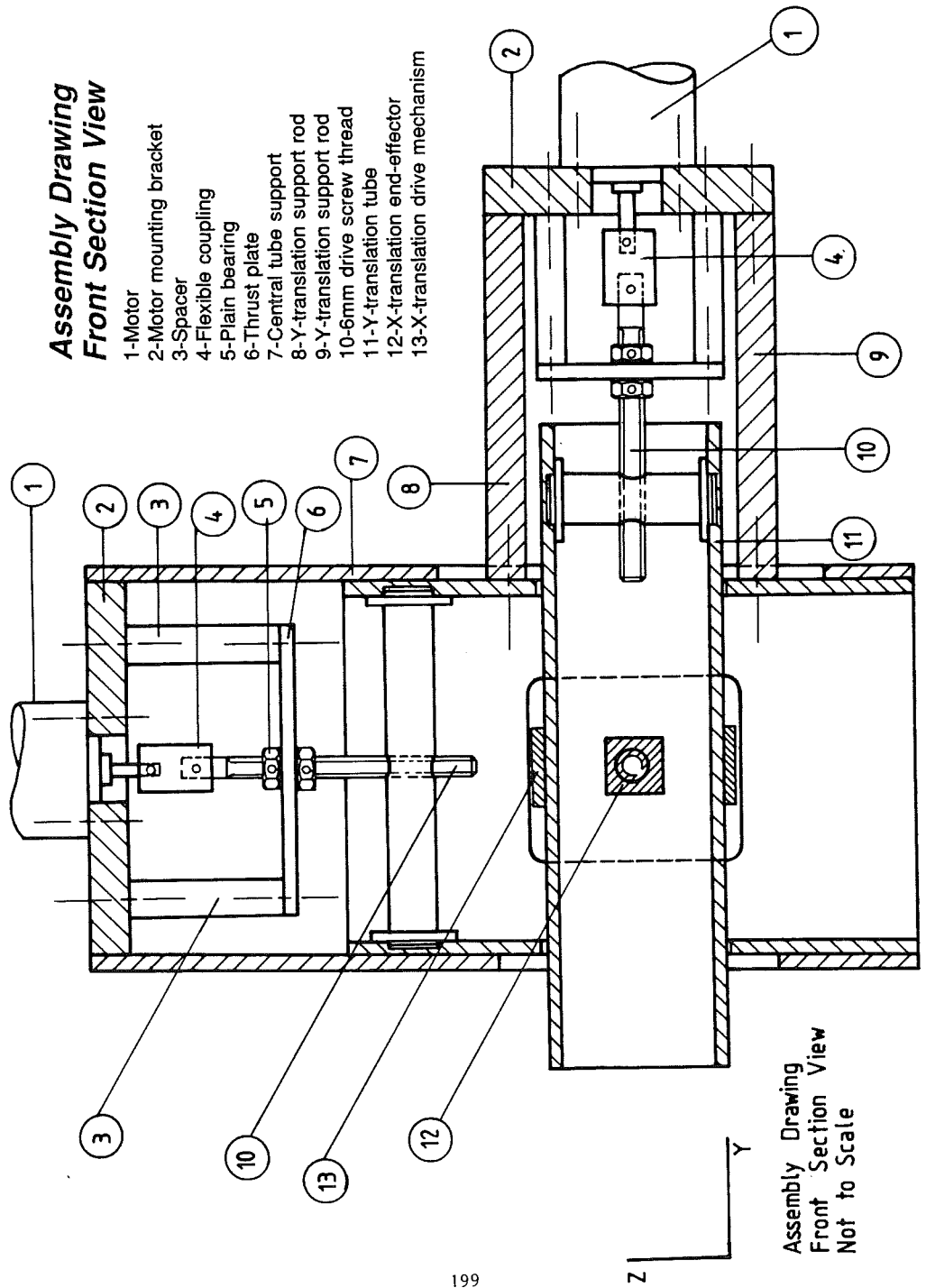


All Dimensions in mm

Micro-manipulator Inner Tube Z-Translation

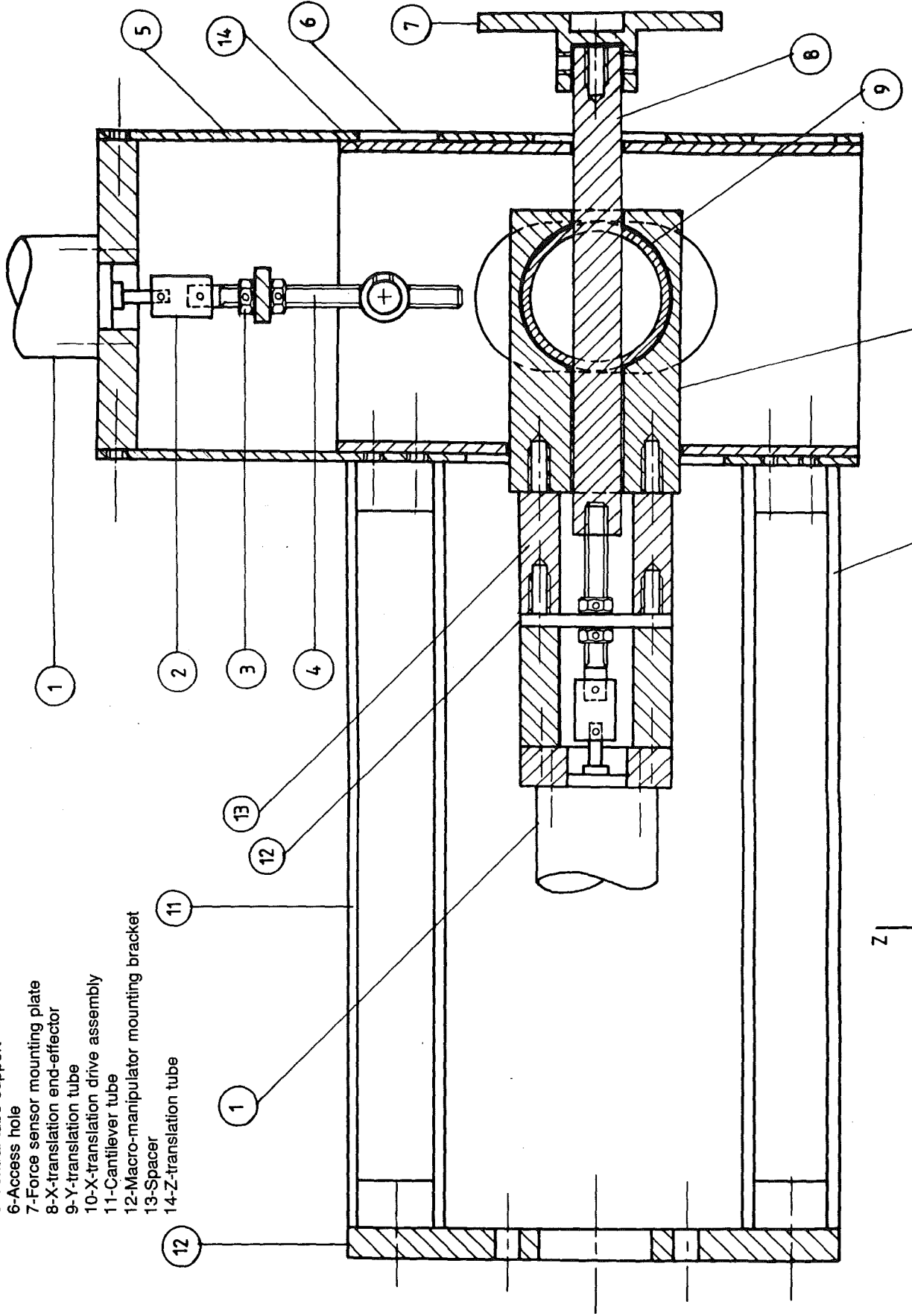
**Assembly Drawing  
Front Section View**

- 1-Motor
- 2-Motor mounting bracket
- 3-Spacer
- 4-Flexible coupling
- 5-Plain bearing
- 6-Thrust plate
- 7-Central tube support
- 8-Y-translation support rod
- 9-Y-translation support rod
- 10-6mm drive screw thread
- 11-Y-translation tube
- 12-X-translation end-effector
- 13-X-translation drive mechanism



**Assembly Drawing  
Side Section View**

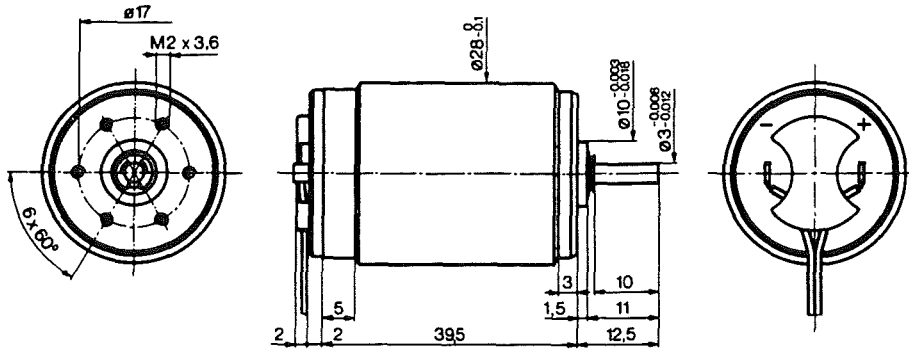
- 1-Motor
- 2-Flexible coupling
- 3-Plain bearing
- 4-Z-translation drive screw thread
- 5-Central tube support
- 6-Access hole
- 7-Force sensor mounting plate
- 8-X-translation end-effector
- 9-Y-translation tube
- 10-X-translation drive assembly
- 11-Cantilever tube
- 12-Macro-manipulator mounting bracket
- 13-Spacer
- 14-Z-translation tube



**Assembly Drawing  
Side Section View**

# D.C. motor

## escap® 28L28



scale: 1:1  
dimensions in mm  
mass 125 g

28L28-... .49

### Standard types available from stock

		-219P	-219	-416E	-413E	-410E
Measuring voltage	V	6	12	24	28	36
No-load speed	rpm	5200	5200	5300	5300	4900
Stall torque	mNm	46	46	55	42	35
	oz-in	6.5	6.5	7.8	5.9	5
Power output	W	6.2	6.2	7.6	5.9	4.5
Av. no-load current	mA	35	18	9	8	6
Back-EMF constant	V/1000 rpm	1.2	2.3	4.5	5.2	7.3
Rotor inductance	mH	0.1	0.5	2.4	3.2	5.2
Typical starting voltage	V	0.05	0.1	0.15	0.2	0.4
Motor regulation R/k <sup>2</sup>	10 <sup>3</sup> /Nms	12	12	10	14	14
Terminal resistance	ohm	1.45	5.8	18.8	33	71
Torque constant	mNm/A	11	22	43	50	70
	oz-in/A	1.56	3.12	6.09	7.08	9.92
Rotor inertia	kgm <sup>2</sup> · 10 <sup>-7</sup>	10.4	10.4	17.5	13.5	11
Mechanical time constant	ms	13	13	18	18	16
Thermal time constant	rotor	s	17	17	27	15
	stator	s	760	760	760	760
Thermal resistance	rotor-body	°C/W	5	5	5	5
	body-ambient	°C/W	12	12	12	12

Precious metal alloy commutator with 9 segments

Standard version fitted with sleeve bearings

Viscous damping constant  $0.5 \times 10^{-6}$  Nms

Max. permissible coil temperature 100°C (210°F)

Max. axial static force for press-fit 250 N

End play  $\leq 100 \mu\text{m}$

Radial play  $\leq 18 \mu\text{m}$

Shaft runout  $\leq 10 \mu\text{m}$

Max. side load at 5 mm from mounting face:

sleeve bearings 6 N ball bearings 8 N

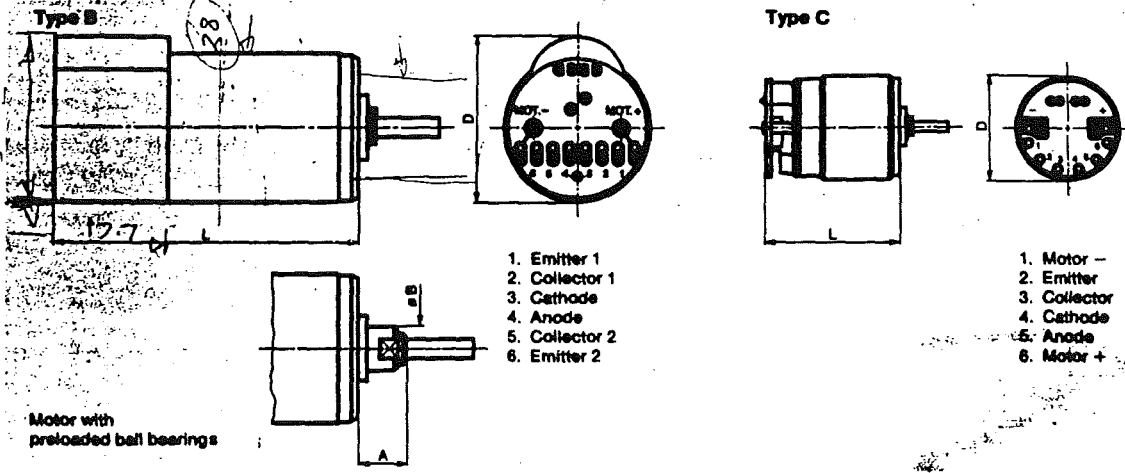
### This motor series is available in the following variants:

- with integrated B-type encoder, see page 42, 43
- with D.C. tachometer (motor-tachometer unit 28HL 18), see page 44, 45
- with reduction gearboxes R22 and A42, see page 47, 48.



# Integrated optical encoders

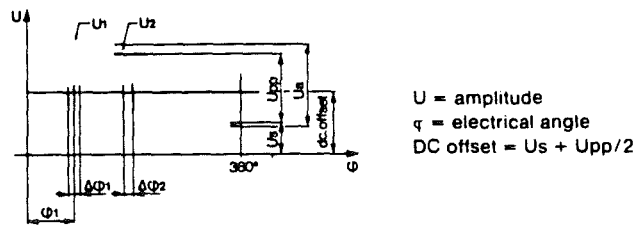
## escap<sup>®</sup> Type B and C



Characteristics at 22°C

			Type B	Type C
Supply voltage	V		5	5
Peak to peak output voltage	$U_{pp}$	V	0.2 to 2	0.2 to 2
Max. ripple over 1 revolution	$(U_a - U_{pp})/U_{pp}$	%	20	30
Signal/DC offset ratio	$U_{pp}/(U_s + U_{pp}/2)$	-	0.5 to 2	0.5 to 2
Electrical phase error within 1 increment	$\Delta\varphi_2$	°	$\pm 20^\circ$	$\pm 25^\circ$
Electrical phase shift between U1 and U2	$\varphi_1 \Delta\varphi_1$	°	$90^\circ \pm 23^\circ$	—
Amplitude ratio between U1 and U2	$U_1/U_2$	-	1 to 2	—
Photodiode type (or equivalent)	-	-	TIL 31	TIL 32
Phototransistor type (or equivalent)	-	-	TIL 78	TIL 78
Max. recom. diode current at 50°C	mA		20	20
Max. recom. transistor current at 50°C	mA		5	5
Max. recom. frequency response	kHz		20	15
Operating temperature range	°C		-20 to +60	-20 to +60
Number of lines available			27, 40, 60, 64, 80 96, 100, 120, 144, 160, 192, 200, 252, 256, 360	1, 4, 16, 32

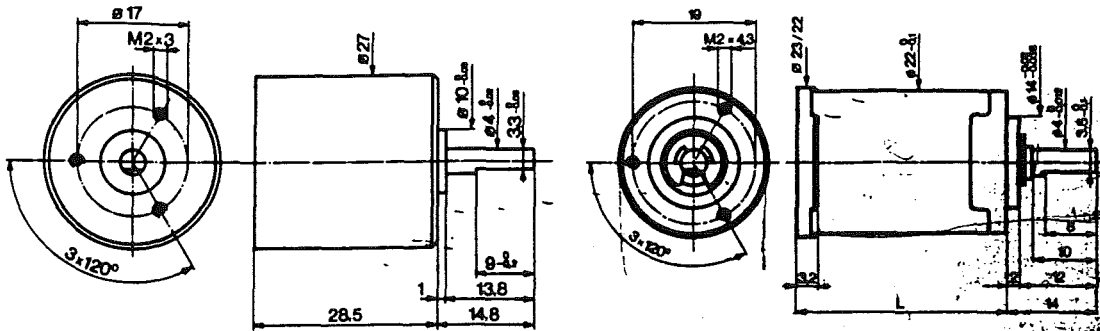
### Typical output signal of encoder



The Band C type encoders are integrated into the motor giving a very compact unit. All the external connections are plated through holes, and on the B type are spaced 2.54 mm apart and aligned. Due to the many possible combinations of motors, gearmotors and type B and C encoders, these products are not available from stock.

# Reduction gearbox

## escap<sup>®</sup> K 27 and R22



scale: 1:1  
dimensions in mm

K 27.0

R 22.0

Standard types available from stock		K 27			R 22		
Max. recom. dynamic output torque	Nm	0.15 at 100 rpm			0.4 (-33:1); 0.6 (-1090:1)		
	oz-in	21.2 at 100 rpm			56 (-33:1); 85 (-1090:1)		
Max. recom. static output torque	Nm	0.3			1.5		
	oz-in	42.5			212		
Max. recommended input speed	rpm	3000			5000		
Average backlash		2° at no-load			1.5° at no-load		
		3° at 0.2 Nm			3° at 0.3 Nm		
Radial shaft play (typical)	µm	35 at 2 mm from mounting face			25 at 5 mm from mounting face		
End play (typical)	µm	150			100		
Max. recommended side load	N (lb)	20 (4.5) at 8 mm from mounting face			10 (2.2) at 8 mm from mounting face		
Max. recommended axial load	N (lb)	8 (1.8)			10 (2.2)		
Max. axial static force for press-fit	N (lb)	300 (67)			300 (67)		
Available reduction ratios		6.2	99.1	2970	16.2 <sup>2)</sup>	65.5 <sup>2)</sup>	376 <sup>3)</sup>
		18.6	198		19.4 <sup>2)</sup>	111 <sup>1)</sup>	641 <sup>3)</sup>
		27.9	501		33.1 <sup>1)</sup>	190 <sup>1)</sup>	1090 <sup>3)</sup>
		55.7	979				
Average efficiency		0.65	0.55	0.4	0.7	0.6	0.5
Nr. of geartrains/direction of rotation		4/ =	6/ =	9/ ≠	2/ =	3/ =	4/ =
Length L	mm	28.5	28.5	28.5	32.5	40	40
Mass	g	40	42	48	25	30	33

Reduction gearbox with spur gears for K27 and planetary gears for R22

Standard version fitted with sleeve bearings

Recommended temperature range -30 to +65°C (-22 to +150°F)

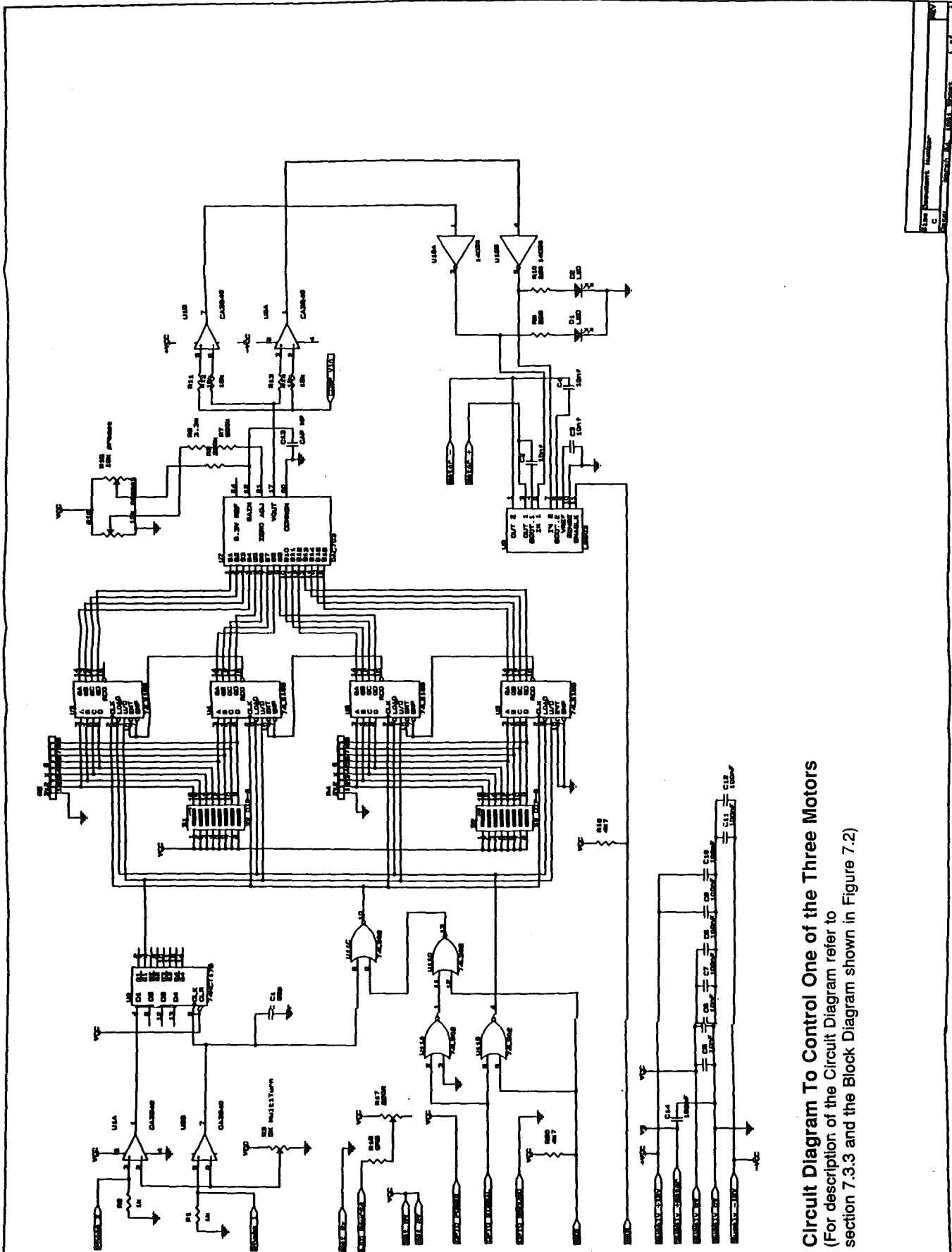
At +65°C the R22 dynamic torque is 2/3 of the value indicated.

Standard motor types available from stock with gearbox series:

- K27: 22C11-216E/210E/207/205 execution .5.  
23L21-216E/213E/208E and 23HL11-213E/204 execution .5.  
26L28-216/210 execution .100.  
P310.158.005/170 execution .15.
- R22: 22C11-216E/210E/207 executions .39<sup>1)</sup>, .40<sup>2)</sup> and .55<sup>3)</sup> (length L is reduced by 2 mm and adaptor flange Ø 22 mm).  
23L21-216E/213E/208E executions .19<sup>1)</sup>, .20<sup>2)</sup> and .48<sup>3)</sup>.  
26L28-216/210 and 28L28-416E executions .176<sup>1)</sup> and 177<sup>2)</sup>.  
P310.158.005/170 executions .16<sup>2)</sup>, .17<sup>1)</sup> and .18<sup>3)</sup>.



# ***Appendix IV***



**Circuit Diagram To Control One of the Three Motors**  
 (For description of the Circuit Diagram refer to section 7.3.3 and the Block Diagram shown in Figure 7.2)

### **General description of PC-30A plug in board**

The following notes are extracted from the manual of PC-30A board supplied by Amplicon Liveline [ref 63].

Note that the indicated section numbers are the same as the ones used in the manual. The relevant information extracted from the manual for controlling the micro-manipulator are listed in below:

<b><u>PARAGRAPH</u></b>	<b><u>SUBJECT</u></b>
<b>1</b>	<b>GENERAL INFORMATION</b>
1.1	General Description
1.2	Features
<b>2</b>	<b>SPECIFICATIONS</b>
2.1	A-D Converter AD574 AJD
2.2	Sample and Hold LF398
2.3	Multiplexer DG506ACJ
2.4	12-bit D-A Converters AD7548
2.5	8-bit D-A Converters AD7528
2.6	Base Address
2.7	Interrupt Request Levels
2.8	Analog Signal Pre-conditioning
2.9	Analog Signals up to 100 volts
2.10	I/O Address Space used by PC-30A
2.11	Jumper Selection
<b>4</b>	<b>ELECTRICAL CONNECTIONS</b>
4.2	50 way D type Connector
<b>5</b>	<b>CALIBRATION</b>
5.1	A-D Converter AD574 ADJ
5.1.2	Bipolar Mode Calibration
5.2	12-bit D-A converters QA15 and QA16
5.2.2	Bipolar Mode Calibration
<b>6</b>	<b>PROGRAMMING</b>
6.2	A-D Conversion
6.2.1	software clock
6.3	D-A Conversion
6.3.1	12-bit D/A Conversion

- 7 **PC-30A ADAPTATIONS**
- 7.1 DAC output noise
- 7.2 Improving the performance  
of the A/D and the D/As

## 1 GENERAL INFORMATION

### 1.1 General Description

The PC-30A is a full sized plug-in board which provides both analog to digital A-D and digital to analog D-A conversion facilities as well as parallel digital input/output and a timer/counter.

The board has jumper selected flexible addressing to allow it to be used with the IBM PC and any of the compatible PC's.

A 16 channel multiplexer connects the selected analog signal via a sample and hold device to a 12 bit AD574 ADC. The overall conversion time from analog to digital is typically 35  $\mu$ sec. The demonstration software supplied with the board enables sampling rates of up to 6 kHz to be used and the additional software package.

PC-28 enables higher sample rates to be achieved. Very slow timing signals down to 1 per hour can be obtained by appropriate selection and programming of the on-board 8253 timer chip.

The board has four D-A converters. Two of these are 12-bit and the other two are 8-bit. Their output is either bipolar (-10 volts to +10 volts) or unipolar(0 to 10 volts).

An on-board 8255 ppi provides three 8-bit input/output ports which can be interconnected in different ways to provide parallel TTL input or output with or without handshake control.

Six levels of interrupt can be selected by means of an on-board jumper. Other jumpers enable unipolar or bipolar modes to be selected for the A/D to be 10 volt or 20 volt and a signal pre-conditioning mode to be selected for the A/D.

## 1.2 Features

Flexible addressing.

Six levels of interrupt.

16 channel, multiplexer, 12-bit A-D converter.

Very fast and very slow sampling rates achievable.

Three 8-bit parallel I/O programmable ports.

Two 8-bit and two 12-bit D-A converters.

## 2.0 SPECIFICATIONS

### 2.1 A-D converter AD574 AJD

Resolution	12 bit
Conversion Time	25 $\mu$ sec 35 $\mu$ sec (MUX + S/H + ADC)
Input Voltage	Bipolar -5 volt to +5 volt -10 volt to +10 volt Unipolar 0 volt to 10 volt
Linearity	11 bits
offset	Adjustable to zero
Linearity drift	0.5 ppm/ $^{\circ}$ C
Gain drift	50 ppm/ $^{\circ}$ C

### 2.2 Sample and hold LF 398

Settling time	< 8 $\mu$ sec
Sample and Hold error	< 0.001%

### 2.3 Multiplexer DG506ACJ

Settling time	<1 $\mu$ sec
Input impedance	100 kOhm nominal

### 2.4 12-bit Digital to Analog Converters AD7548

Resolution	12 bits
Linearity	13 bits
Settling Time	<1 $\mu$ sec
Gain error	Adjustable

### 2.6 Base address

The base address is selected by means of jumpers to be within the range 00000 to 0FFE0. The fourth decade can only have even values selected.



### **2.7 Interrupt Request levels**

The interrupt request level is jumper selected to be IRQ 2,3,4,5,6,or7.

### **2.8 Analog Signal Pre-conditioning**

An external analog signal pre-conditioning unit which will act on all analog channels can be plug connected to the board.

### **2.9 Analog Input Signal up to 200 volt Peak to Peak**

The analog input signal channel tracks may be cut and attenuation resistors fitted to enable A/D conversions on signals up to a peak voltage of  $\pm 100$  volts.

### **2.10 Input/Output Address Space used by PC-30A**

The base address (BA) for the PC-30A board is selected by means of jumpers as described in PC-30A Manual[Ref 63].

Port Address	Function
BA + 00 BA + 01 BA + 02 BA + 03	Input 8 LSB's from QA6/AD574 Input 4 MSB's from QA6/AD574 Bit 0 to Bit 3 inclusive: control bits for ADC Bit 4 to Bit 7 inclusive: Multiplexer channel selection. Control word for QA11/8255 ppi
BA + 04 BA + 05 BA + 06 BA + 07	Counter 0 of the QA18/8253 timer/counter. Counter 1 of the QA18/8253 timer/counter. Counter 2 of the QA18/8253 timer/counter. Control word for the QA1/8253 timer/counter.
BA + 08 BA + 09 BA + 0A BA + 0B	Port A QA19/8255 ppi. Port B QA19/8255 ppi. Port C QA19/8255 ppi. Control word for the QA19/8255 ppi.
BA + 0C BA + 0D	D-A 12-bit No 1 4 LSB's. D-A 12-bit No 1 8 MSB's.
BA + 10 BA + 11	D-A 12-bit No 2 4 LSB's. D-A 12 bit No 2 8 MSB's.
BA + 14 BA + 15	D-A 8 bit No 1. D-A 8 bit No 2.

Table 1, Address space functions.

## 2.11 Jumper Selection

Jumper	Selection	Device	Function Selected
J1	UNI BIP	D/A - 12.2	0 to +10 volts -10 to +10 volts
J2	UNI BIP	D/A - 12.1	0 to +10 volts -10 to +10 volts
J3	UNI BIP	D/A - 8.1	0 to +10 volts -10 to +10 volts
J4			Pre-conditioning
J5	10 v 20 v	ADC 574A	10 volt selection 20 volt selection
J6	UNI BIP	ADC 574A	Unipolar selection Bipolar selection
J7	UNI BIP	D/A - 8.2	0 to +10 volts -10 to +10 volts
J8	TMR CLK	8253	2 timer cascaded with 1&0 2 timer connected to system clock
J9		Base Address	4th Hex Digit(nnnXn) 0,2,4,6,8,A,C or E
J10		Base Address	3rd Hex Digit(nnXnn)
J11	Lower Upper	Base Address	3rd Decade Even digits Odd digits
J12		Base Address	2nd Hex Digit(nXnnn)
J13	Lower Upper	Base Address	2nd Decade Even digits odd digits
J14	IRQ	Interrupt Request Level	IRQ2,IRQ3,IRQ4,IRQ5,IRQ6,IRQ7

Table 2, Jumper selections

## 4 ELECTRICAL CONNECTIONS

### 4.2 50 way D type Connector

Connections from the PC30-A to the outside world are made via a 50 way D type connector. The pin assignments for this connector are shown in below.

Function	Pin	Pin	Function
Output D/A 12/1	1	26	Port B6
+12 volts	2	27	Port B3
-12 volts	3	28	Port A7
channel 8	4	29	Port A5
channel 7	5	30	Port A3
channel 6	6	31	Port A1
channel 3	7	32	Port C7
channel 1	8	33	Port C5
Port B7	9	34	Output D/A 8/2
Port B5	10	35	Output D/A 8/1
Port B4	11	36	channel 10
Port A6	12	37	channel 11
Port A4	13	38	channel 12
Port A2	14	39	channel 13
Port A0	15	40	channel 14
Port C6	16	41	channel 15
+5 volts	17	42	Ground
Output D/A 12/2	18	43	Port B0
channel 9	19	44	Port B1
channel 0	20	45	Port B2
Output timer 2	21	46	Port C3
channel 6	22	47	Port C2
channel 4	23	48	Port C1
channel 2	24	49	Port C0
Trigger Port B4	25	50	Port C4

- Notes: 1) Channel means Analog channel.  
 2) Port refers to the I/O ports of QA19/8255

## 5.0 CALIBRATIONS

### 5.1 A-D Converter AD574 ADJ

#### 5.1.2 Bipolar Mode Calibration

- a) Remove the PC-30A board from the computer.
- b) Select jumper J6 to the BIP position.
- c) Select jumper J5 to the 10 v position.
- d) With the computer switched off fit the PC-30A board to one of the spare slots.
- e) Switch on the computer, using the demonstration disc as the boot program.
- f) Follow the instructions of the demonstration program, entering the board address and the system clock frequency when requested.
- g) Connect a reference voltage source capable of providing 10 volts, stable to <0.1 mV to pin 20 (analog channel 0). Use a digital voltmeter to check this input voltage.
- h) Select '3' on the demonstration disc menu and enter 0 when a channel number is requested, to obtain a reading for channel 0 on the

screen.

Use fig 5 to find RV10 and RV11.

j) Apply a voltage of -4.9988 volts to channel 0 and adjust RV11 to obtain the first transition from 0 to 1.

k) Apply a voltage of +4.9963 volts to channel 0 and adjust RV 12 to obtain the last transition from 4094 to 4095.

## **5.2 12-bit D-A converters QA15 and QA16**

### **5.2.2 Bipolar Mode Calibration**

a) Remove the PC-30A board from the computer.

b) Select jumper

J2 to the UNI position for Number 1 (QA16)

J1 to the UNI position for Number 2 (QA15)

c) With the computer switched off fit the PC-30A board to one of the spare slots.

d) Switch on the computer, using the demonstration disc as the boot program.

e) Follow the instruction of the demonstration program, entering the board address and the system clock frequency when requested.

f) Select '6' (Set 12-bit and 8-bit D-A converters) on the demonstration disc menu and then '2' to obtain Unipolar mode.

g) When the Bipolar Mode screen is displayed input +10 as the Output-Voltage.

h) Use Figure 5 to find RV4 and RV16 is being calibrated or RV1 and RV3 when QA15 is being calibrated.

i) Adjust RV6 to obtain a +10 volt reading at pin 1 or RV3 to obtain a +10 volt reading at pin 18.

j) Select the Output-voltage as -10 volts.

k) Adjust RV4 to obtain -10 volts at pin 1 or RV1 to obtain -10 volts at pin 18.

l) Repeat the sequence given by steps e to 1 inclusive as the settings of the  $\pm$  potentiometers influence each other.

## **6.2 A/D Conversion**

The PC-30A provides two methods for determining the sampling frequency. The first of these uses a software clock where the sampling frequency depends

upon the execution speed of the program which, in turn, is a function of the system clock frequency.

The second method makes use of the 8253 counter/timers to provide the sampling frequency. When an A-D conversion has been completed the interrupt system is triggered in order to plot the reading.

### 6.2.1 Software Clock

When Using the software clock, sampling conversion is initiated by the program so the frequency is a function of the system clock. Thus it is difficult to ascertain the exact sampling frequency when using the software clock.

The software clock is selected by loading the 8255 Port C bit 1 with '1'. It is triggered when Port C bit 0 is set to 1. Bits 4 to 7 of

port C are loaded with the analog channel number that the system is required to sample. The address of the Port C is base address + 02.

The function ADSAMPLE in the demonstration program shows how this information is used to sample the selected analog channel 'channel'. This function provides a maximum sample frequency of about 1500 Hz.

The procedure High Speed Sample on the demonstration disc increases the sampling frequency to 6000 Hz when used with a PC having a system clock frequency of 4.77 MHz. This is achieved by storing the values read from Port A and Port B in an array, carrying out the calculations and plotting routines after the sampling has been completed.

```
FUNCTION ADSAMPLE (channel: INTEGER) : INTEGER;
    VAR I:INTEGER;
    BEGIN
        PORT [ba+$02] :=(channel SHL 4) + 2; {channel selection and clear
software cloak                                     triggering bit}
        PORT [ba+$02]:= (channel SHL 4) + 3; {channel selection and sets
software clock                                     trigger bit}
        FOR I = 1 to 6 DO
            BEGIN                                     {loop until end of conversion}
            END;
        ADSAMPLE := ((PORT[ba+$01] AND $0F) SHL 8) + PORT [ba+00];
    END;
```

NOTES: 1) The statement PORT[ba+\$02]:= (channel SHL 4)+2 loads the

value of the selected analog channel into Port C bit 4 to 7, sets Port C bit 0 to 0 so that the clock is not triggered.

2) The statement `PORT[ba+$02]:=(channel SHL 4)+3` sets Port C bit 0 to 1 in order to trigger the software clock.

3) The A-D conversion process requires about 40  $\mu$ sec. The short FOR - DO loop in ADSAMPLE provides this delay.

4) The statement `ADSAMPLE:=((PORT[ba+$01] AND $0F)SHL8) 6+ PORT[ba + 00]`; reads the port B bits, masks out bits 4 to 7 and then shifts bits 3 to 0 to the four most significant bits of the 12 bit A-D output. The eight bit values from Port A are added to these four most significant bits to provide the 12 bit value ADSAMPLE.

### 6.3 D-A Conversion

The D-A output voltages are established using the procedure SetDAs.

#### 6.3.1 12-bit D-A Conversion

In the unipolar mode the 12-bit value D to set the D-A is calculated using the Turbo Pascal expression  $D = V_{out} * 4095/10$  volts where  $V_{out}$  is the desired output voltage.

In the bipolar mode the 12 bit value D set D-A is calculated by  $D = V_{out} * (2084/-10) + 2048$ .

FOR  $V_{out} = +10$  volts  $D=0$

FOR  $V_{out} = 0$  volts  $D= 2048$

FOR  $V_{out} = -9.9976$   $D=4095$ .

The full scale error of the DAC is -10 volts +1 LSB. This equals -10 +0.00244=-9.9976 volts. The error can be adjusted to be zero so that when  $D=4095$  the output voltage is -10 volts.

The DAC accept a left justified data word in which the 8 most significant bits are loaded first followed by the least significant bits.

In the Procedure 'SetDAs' this is achieved by the following Turbo Pascal routine.

```
temp:=ROUND((output_voltage*(2048/-10))+2048);
```

```
IF temp>4095 then temp=4095;
```

```
da_msb := temp SHL 4; {temp shifted 4 bits right}
```

```
da_lsb :=(temp SHL 4) AND $F0; {temp shifted 4 bits left, the 4 LSB's masked out}
```

Port (ba + \$OD) := da\_msb {set msb of D/A-12 1}

Port (ba + \$0C) := da\_lsb {set lsb of D/A-12 1}

## **7 PC-30A ADAPTATION**

If you want to configure the PC-30A to meet your own particular requirements a number of adaptations of the PCB are possible. Details of these are included for your information.

### **7.1 DAC output noise**

The DAC circuits have been designed for optimum performance, providing maximum bandwidth and frequency response commensurate with an acceptably low DAC output noise level.

The output noise amplitude of a particular DAC can be further reduced by connecting a capacitor across particular op-amps. However the improvement in noise level will be accompanied by a reduction in bandwidth and frequency response. The following paragraphs give guidance on fitting the capacitors. The choice and frequency response. The following paragraphs give guidance on fitting the capacitors. The choice of the capacitor value is left to the user.

### **7.2 Improving the performance of the A-D and D-As**

When the PC-30A is to be used in a hostile environment its performance may be improved by replacing the socketed AD574AJD with a higher performance version such as a military standard AD574.

Similarly the performance of the D-As may be improved by replacing the socketed AD7548JN's and AD7528JN with higher performance versions.



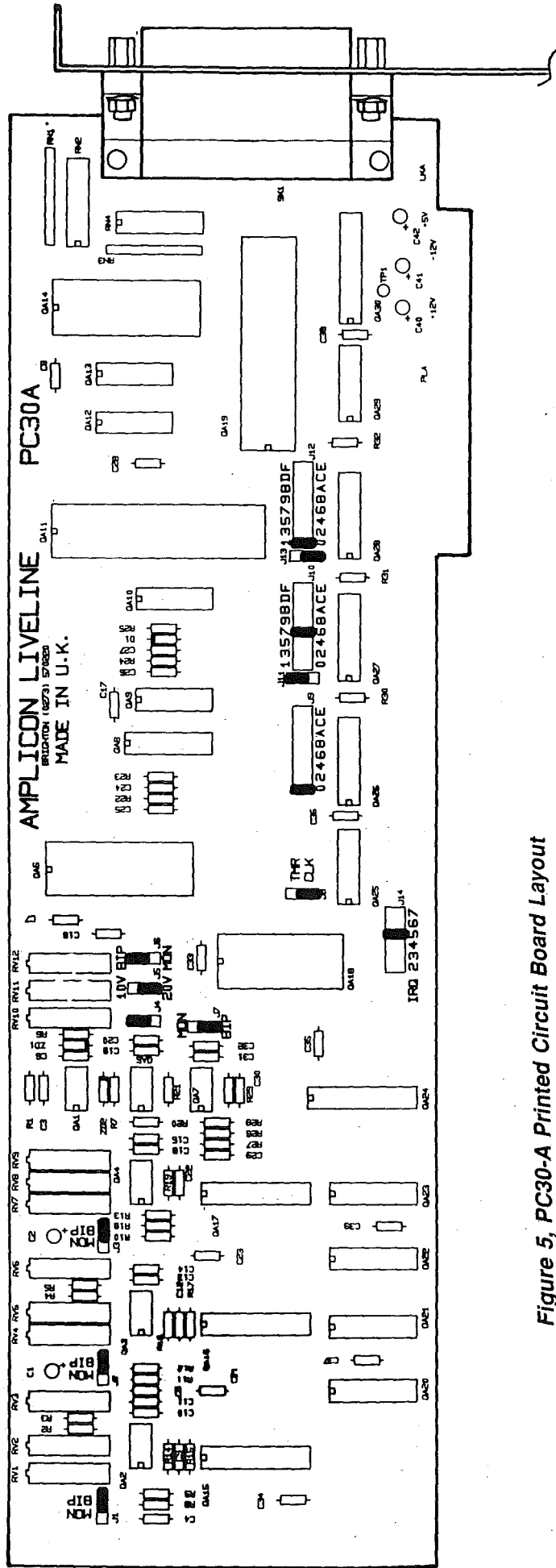
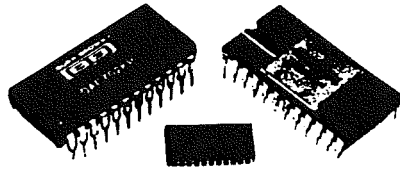


Figure 5, PC30-A Printed Circuit Board Layout





**DAC700/702**  
**DAC701/703**

## Monolithic 16-Bit DIGITAL-TO-ANALOG CONVERTERS

### FEATURES

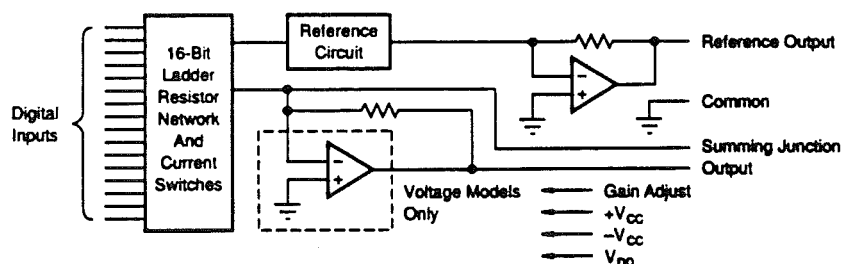
- MONOLITHIC CONSTRUCTION
- $V_{OUT}$  AND  $I_{OUT}$  MODELS
- HIGH ACCURACY:  
Linearity Error  $\pm 0.0015\%$  of FSR max  
Differential Linearity Error  $\pm 0.003\%$  of FSR max
- MONOTONIC (at 15 bits) OVER FULL SPECIFICATION TEMPERATURE RANGE
- PIN-COMPATIBLE WITH DAC70, DAC71, DAC72
- LOW COST
- DUAL-IN-LINE PLASTIC AND HERMETIC CERAMIC
- /QM ENVIRONMENTAL SCREENING AVAILABLE
- BURN-IN PROGRAM AVAILABLE (-BI)

### DESCRIPTION

This is another industry first from Burr-Brown—a complete 16-bit digital-to-analog converter that includes a precision buried-zener voltage reference and a low-noise, fast-settling output operational amplifier (voltage output models), all on one small monolithic chip. A combination of current-switch design techniques accomplishes not only 15-bit monotonicity over the entire specified temperature range, but also a maximum end-point linearity error of  $\pm 0.0015\%$  of full-scale range. Total full-scale gain drift is limited to  $\pm 10\text{ppm}/^\circ\text{C}$  maximum (LH and CH grades).

Digital inputs are complementary binary coded and are TTL-, LSTTL-, 54/74C- and 54/74HC-compatible over the entire temperature range. Outputs of 0 to +10V,  $\pm 10\text{V}$ , 0 to  $-2\text{mA}$ , and  $\pm 1\text{mA}$  are available.

These D/A converters are packaged in hermetic 24-pin ceramic side-brazed or molded plastic. The DIP-packaged parts are pin-compatible with the voltage and current output DAC71 and DAC72 model families. The DAC700 and DAC702 are also pin-compatible with the DAC70 model family. In addition, the DAC703 is offered in a 24-pin SOIC package for surface mount applications.



International Airport Industrial Park • Mailing Address: PO Box 11400 • Tucson, AZ 85734 • Street Address: 6730 S. Tucson Blvd. • Tucson, AZ 85706  
Tel: (602) 746-1111 • Twx: 910-852-1111 • Cable: BBRCORP • Telex: 066-6401 • FAX: (602) 890-1510 • Immediate Product Info: (800) 548-6122

# SPECIFICATIONS

## ELECTRICAL

At +25°C and rated power supplies unless otherwise noted.

MODEL	DAC702/703J			DAC700/701/702/703K			DAC700/701/702/703B, S			DAC700/701/702/703L, C			UNITS
PARAMETER	MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
<b>INPUT</b>													
DIGITAL INPUT													
Resolution			16										Bits
Digital Inputs <sup>(1)</sup>													
V <sub>H</sub>	+2.4		+V <sub>CC</sub>										V
V <sub>L</sub>	-1.0		+0.8										V
I <sub>H</sub> , V <sub>I</sub> = +2.7V			+40										µA
I <sub>L</sub> , V <sub>I</sub> = +0.4V		-0.35	-0.5										mA
<b>TRANSFER CHARACTERISTICS</b>													
ACCURACY <sup>(2)</sup>													
Linearity Error <sup>(4)</sup>		±0.0015	±0.006			±0.003					±0.00075	±0.0015	% of FSR <sup>(6)</sup>
Differential Linearity Error <sup>(4)</sup>		±0.003	±0.012			±0.006					±0.0015	±0.003	% of FSR
Differential Linearity Error at Bipolar Zero (DAC702/703) <sup>(4)</sup>					±0.003	±0.006		±0.0015	±0.003				% of FSR
Gain Error <sup>(5)</sup>		±0.07	±0.30			±0.15		±0.05	±0.10				%
Zero Error <sup>(5, 6)</sup>		±0.05	±0.10										% of FSR
Monotonicity Over Spec. Temp Range	13			14						15			Bits
DRIFT (over specification temperature range)													
Total Error Over Temperature Range (all models) <sup>(7)</sup>		±0.08				±0.15		±0.05	±0.10				% of FSR
Total Full Scale Drift:													
DAC700/701		±10				±30		±8.5	±18		±6	±13	ppm of FSR/°C
DAC702/703		±10				±25		±7	±15		±5	±10	ppm of FSR/°C
Gain Drift (all models)		±10	±30			±25		±7	±15		±5	±10	ppm/°C
Zero Drift:													
DAC700/701					±2.5	±5		±1.5	±3				ppm of FSR/°C
DAC702/703		±5	±15			±12		±4	±10		±2.5	±5	ppm of FSR/°C
Differential Linearity Over Temp. <sup>(4)</sup>			±0.012			+0.009, -0.006							% of FSR
Linearity Error Over Temp. <sup>(4)</sup>			±0.012			±0.006							% of FSR
SETTLING TIME (to ±0.003% of FSR) <sup>(8)</sup>													
DAC701/703 (V <sub>OUT</sub> Models)													
Full Scale Step, 2kΩ Load		4				8							µs
1LSB Step at Worst-Case Code <sup>(9)</sup>		2.5											µs
Slew Rate		10											V/µs
DAC700/702 (I <sub>OUT</sub> Models)													
Full Scale Step (2mA), 10 to 100Ω Load		350				1000							ns
1kΩ Load		1				3							µs
<b>OUTPUT</b>													
VOLTAGE OUTPUT MODELS													
DAC701 (CSB Code)					0 to +10								V
DAC703 (COB Code)		±10											V
Output Current	±5												mA
Output Impedance		0.15											Ω
Short Circuit to Common Duration		Indefinite											
CURRENT OUTPUT MODELS													
DAC700 (CSB Code) <sup>(10)</sup>					0 to -2								mA
Output Impedance <sup>(10)</sup>					4								kΩ
DAC702 (COB Code) <sup>(10)</sup>		±1											mA
Output Impedance <sup>(10)</sup>		2.45											kΩ
Compliance Voltage		±2.5											V

## ELECTRICAL (CONT)

MODEL	DAC702/703J			DAC700/701/702/703K			DAC700/701/702/703B, S			DAC700/701/702/703L, C			UNITS
	MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
<b>REFERENCE VOLTAGE</b>													
Voltage		+6.3		+6.0	+6.3	+6.6	+6.24	+6.3	+6.36	*	*	*	V
Source Current Available for External Loads		+2.5		+1.5	*	*	*	*	*	*	*	*	mA
Temperature Coefficient		±10		*	*	±25	*	*	±15	*	*	*	ppm/°C
Short Circuit to Common Duration		Indefinite		*	*	*	*	*	*	*	*	*	
<b>POWER SUPPLY REQUIREMENTS</b>													
Voltage: +V <sub>cc</sub>	13.5	15	16.5	*	*	*	*	*	*	*	*	*	V
-V <sub>cc</sub>	13.5	15	16.5	*	*	*	*	*	*	*	*	*	V
V <sub>oo</sub>	+4.5	+5	+16.5	*	*	*	*	*	*	*	*	*	V
Current (No Load):													
DAC700/702													
(I <sub>out</sub> Models)													
+V <sub>cc</sub>		+10	+25	*	*	*	*	*	*	*	*	*	mA
-V <sub>cc</sub>		-13	-25	*	*	*	*	*	*	*	*	*	mA
V <sub>oo</sub>		+4	+8	*	*	*	*	*	*	*	*	*	mA
DAC701/703													
(V <sub>out</sub> Models)													
+V <sub>cc</sub>		+16	+30	*	*	*	*	*	*	*	*	*	mA
-V <sub>cc</sub>		-18	-30	*	*	*	*	*	*	*	*	*	mA
V <sub>oo</sub>		+4	+8	*	*	*	*	*	*	*	*	*	mA
Power Dissipation:													
(V <sub>oo</sub> = +5.0V) <sup>(11)</sup>													
DAC700/702		365		*	*	790	*	*	630	*	*	*	mW
DAC701/703		530		*	*	940	*	*	780	*	*	*	mW
Power Supply Rejection:													
+V <sub>cc</sub>		±0.0015	±0.006	*	*	*	*	*	±0.003	*	*	*	% of FSR/%V <sub>cc</sub>
-V <sub>cc</sub>		±0.0015	±0.006	*	*	*	*	*	±0.003	*	*	*	% of FSR/%V <sub>cc</sub>
V <sub>oo</sub>		±0.0001	±0.001	*	*	*	*	*	*	*	*	*	% of FSR/%V <sub>oo</sub>
<b>TEMPERATURE RANGE</b>													
Specification:													
B, C Grades							-25		+85	*	*	*	°C
S Grades							-55		+125	*	*	*	°C
J, K, L Grades	0		+70	*	*	*	*	*	*	0		+70	°C
Storage: Ceramic				-60		+150	*	*	*	*	*	*	°C
Plastic, SOIC	-60		+100	*	*	*	*	*	*	*	*	*	°C

\* Specification same as model to the left.

NOTES: (1) Digital inputs are TTL, LSTTL, 54/74C, 54/74HC, and 54/74HTC compatible over the operating voltage range of V<sub>cc</sub> = +5V to +15V and over the specified temperature range. The input switching threshold remains at the TTL threshold of 1.4V over the supply range of V<sub>cc</sub> = +5V to +15V. As logic "0" and logic "1" inputs vary over 0V to +0.8V and +2.4V to +10V respectively, the change in the D/A converter output voltage will not exceed ±0.0015% of FSR for the LH and CH grades, ±0.003% of FSR for the BH grade and ±0.006% of FSR for the KG grade. (2) DAC700 and DAC702 (current-output models) are specified and tested with an external output operational amplifier connected using the internal feedback resistor in all parameters except settling time. (3) FSR means full-scale range and is 20V for the ±10V range (DAC703), 10V for the 0 to +10V range (DAC701), FSR is 2mA for the ±1mA range (DAC702) and the 0 to +2mA range (DAC700). (4) ±0.0015% of full-scale range is equivalent to 1LSB in 15-bit resolution, ±0.003% of full-scale range is equivalent to 1LSB in 14-bit resolution, ±0.006% of full-scale range is equivalent to 1LSB in 13-bit resolution. (5) Adjustable to zero with external trim potentiometer. Adjusting the gain potentiometer rotates the transfer function around the zero point. (6) Error at input code FFFF<sub>n</sub> for DAC700 and DAC701, 7FFF<sub>n</sub> for DAC702 and DAC703. (7) With gain and zero errors adjusted to zero at +25°C. (8) Maximum represents the 3σ limit. Not 100% tested for this parameter. (9) At the major carry, 7FFF<sub>n</sub> to 8000<sub>n</sub> and 8000<sub>n</sub> to 7FFF<sub>n</sub>. (10) Tolerance on output impedance and output current is ±30%. (11) Power dissipation is an additional 40mW when V<sub>oo</sub> is operated at +15V.

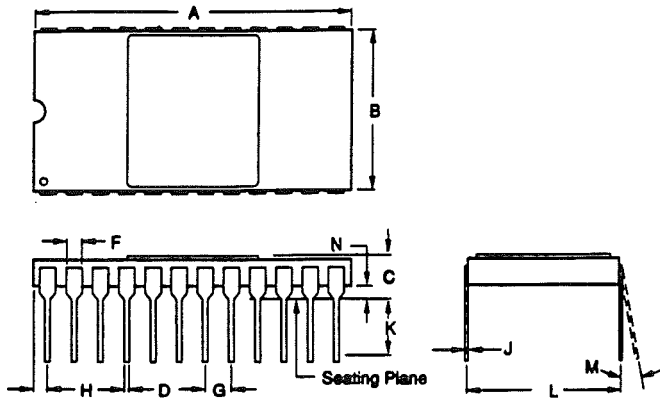
## ABSOLUTE MAXIMUM RATINGS

+V <sub>cc</sub> to Common	0V, +18V	V <sub>out</sub> (DAC701/703)	Indefinite Short to Common
-V <sub>cc</sub> to Common	0V, -18V	Power Dissipation	1W
V <sub>oo</sub> to Common	0V, +18V	Storage Temperature	-60°C to +150°C
Digital Data Inputs to Common	-1V, +18V	Lead Temperature (soldering, 10s)	300°C
Reference Out to Common	Indefinite Short to Common	NOTE: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. Exposure to absolute maximum conditions for extended periods may affect device reliability.	
External Voltage Applied to R <sub>p</sub> (DAC700/702)	±18V		
External Voltage Applied to D/A Output (DAC701/703)	-5V to +5V		

The information provided herein is believed to be reliable; however, BURR-BROWN assumes no responsibility for inaccuracies or omissions. BURR-BROWN assumes no responsibility for the use of this information, and all use of such information shall be entirely at the user's own risk. Prices and specifications are subject to change without notice. No patent rights or licenses to any of the circuits described herein are implied or granted to any third party. BURR-BROWN does not authorize or warrant any BURR-BROWN product for use in life support devices and/or systems.

## MECHANICAL

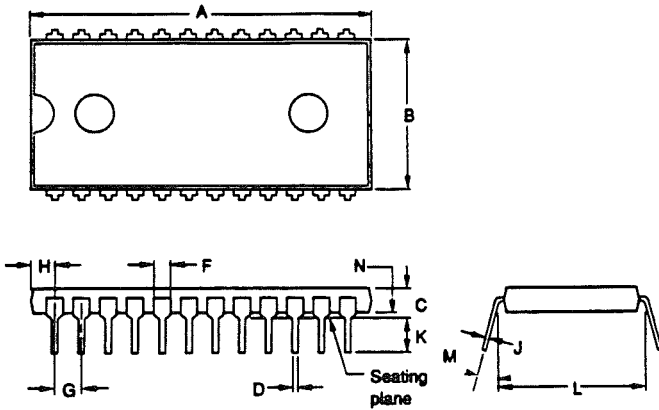
### H Package—24-Pin Hermetic DIP



DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	1.185	1.215	30.10	30.86
B	.600	.620	15.24	15.75
C	.125	.171	3.18	4.34
D	.015	.021	0.38	0.53
F	.035	.060	0.89	1.52
G	.100 BASIC		2.54 BASIC	
H	.030	.070	0.76	1.78
J	.008	.012	0.20	0.30
K	.120	.240	3.05	6.10
L	.600 BASIC		15.24 BASIC	
M	—	10°	—	10°
N	.025	.060	0.64	1.52

NOTE: Leads in true position within 0.01" (0.25mm) R at MMC at seating plane.

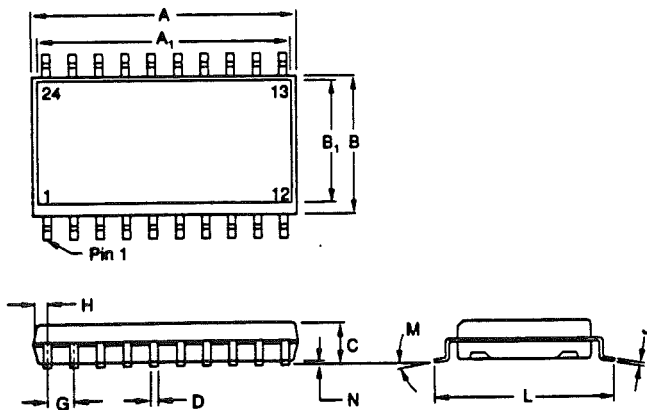
### P Package—24-Pin Plastic DIP



DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	1.233	1.283	31.32	32.59
B	.538	.575	13.87	14.61
C	.169	.224	4.29	5.70
D	.015	.023	0.38	0.58
F	.043	.062	1.09	1.57
G	.100 BASIC		2.54 BASIC	
H	.030	.090	0.76	2.29
J	.008	.015	0.20	0.38
K	.100	.132	2.54	3.35
L	.600 BASIC		15.24 BASIC	
M	0°	10°	0°	10°
N	.018	.022	0.46	0.56

NOTE: Leads in true position within 0.01" (0.25mm) R at MMC at seating plane.

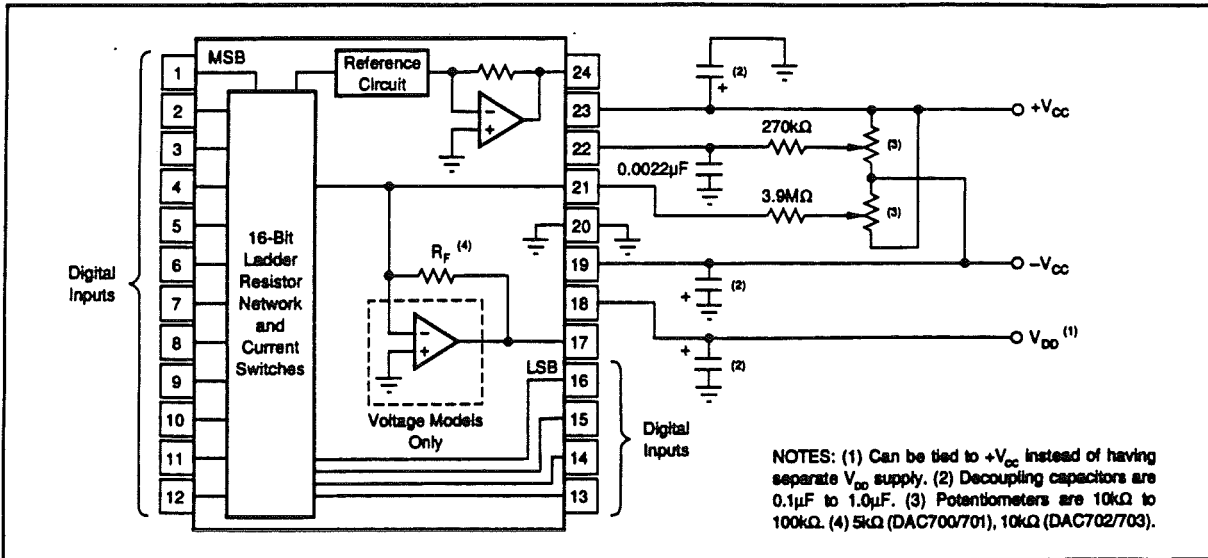
### U Package (DAC703 Only)—24-Pin SOIC



DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	.602	.618	15.29	15.70
A <sub>1</sub>	.595	.618	15.11	15.70
B	.286	.302	7.26	7.67
B <sub>1</sub>	.270	.285	6.86	7.24
C	.093	.108	2.36	2.74
D	.015	.019	0.38	0.48
G	.050 BASIC		1.27 BASIC	
H	0.26	.034	0.66	0.86
J	.008	.012	0.20	0.30
L	.390	.422	9.91	10.72
M	0°	10°	0°	10°
N	.000	.012	0.00	0.30

NOTE: Leads in true position within 0.01" (0.25mm) R at MMC at seating plane. Pin numbers shown for reference only. Numbers may not be marked on package.

## CONNECTION DIAGRAM



## PIN ASSIGNMENTS

PIN #	ALL PACKAGES	
	DAC700/702	DAC701/703
1	Bit 1 (MSB)	Bit 1 (MSB)
2	Bit 2	Bit 2
3	Bit 3	Bit 3
4	Bit 4	Bit 4
5	Bit 5	Bit 5
6	Bit 6	Bit 6
7	Bit 7	Bit 7
8	Bit 8	Bit 8
9	Bit 9	Bit 9
10	Bit 10	Bit 10
11	Bit 11	Bit 11
12	Bit 12	Bit 12
13	Bit 13	Bit 13
14	Bit 14	Bit 14
15	Bit 15	Bit 15
16	Bit 16 (LSB)	Bit 16 (LSB)
17	$R_{FEEDBACK}$	$V_{OUT}$
18	$V_{DD}$	$V_{DD}$
19	$-V_{CC}$	$-V_{CC}$
20	Common	Common
21	$I_{OUT}$	Summing Junction (Zero Adjust)
22	Gain Adjust	Gain Adjust
23	$+V_{CC}$	$+V_{CC}$
24	+6.3V Reference Output	+6.3V Reference Output

DIGITAL INPUT CODES	ANALOG OUTPUT		
	DAC700/701 Complementary Straight Binary (CSB)	DAC702/703 Complementary Offset Binary (COB)	DAC702/703 Complementary Two's Complement (CTC)*
0000 <sub>H</sub>	+ Full Scale	+ Full Scale	-1LSB
7FFF <sub>H</sub>	+1/2 Full Scale	Bipolar Zero	- Full Scale
8000 <sub>H</sub>	+1/2 Full Scale	-1LSB	+ Full Scale
FFFF <sub>H</sub>	-1LSB	- Full Scale	Bipolar Zero

\* Invert the MSB of the COB code with an external inverter to obtain CTC code.

TABLE I. Digital Input Codes.

## DISCUSSION OF SPECIFICATIONS

### DIGITAL INPUT CODES

The DAC700/701/702/703 accept complementary digital input codes in either binary format (CSB, unipolar or COB, bipolar). The COB models DAC702/703 may be connected by the user for either complementary offset binary (COB) or complementary two's complement (CTC) codes (see Table I).

### ACCURACY

#### Linearity

This specification describes one of the most important measures of performance of a D/A converter. Linearity error is the deviation of the analog output from a straight line drawn through the end points (all bits ON point and all bits OFF point).

#### Differential Linearity Error

Differential linearity error (DLE) of a D/A converter is the deviation from an ideal 1LSB change in the output from one adjacent output state to the next. A differential linearity error specification of  $\pm 1/2$ LSB means that the output step sizes can be between  $1/2$ LSB and  $3/2$ LSB when the input changes from one adjacent input state to the next. A negative DLE specification of no more than -1LSB (-0.006% for 14-bit resolution) insures monotonicity.

#### Monotonicity

Monotonicity assures that the analog output will increase or remain the same for increasing input digital codes. The DAC700/701/702/703 are specified to be monotonic to 14 bits over the entire specification temperature range.

#### DRIFT

##### Gain Drift

Gain drift is a measure of the change in the full-scale range output over temperature expressed in parts per million per

## ORDERING INFORMATION

MODEL	PACKAGE	OUTPUT CONFIGURATION	TEMPERATURE RANGE	LINEARITY ERROR, MAX AT +25°C (% of FSR)	GAIN DRIFT MAX (ppm/°C)
DAC702JP, DAC703JP	Plastic DIP	±1mA, ±10V	0°C to +70°C	±0.006	±30
DAC702KP, DAC703KP	Plastic DIP	±1mA, ±10V	0°C to +70°C	±0.003	±25
DAC700KH, DAC701KH	Ceramic DIP	0 to -2mA, 0 to +10V	0°C to +70°C	±0.003	±25
DAC702KH, DAC703KH	Ceramic DIP	±1mA, ±10V	0°C to +70°C	±0.003	±25
DAC700BH, DAC701BH	Ceramic DIP	0 to -2mA, 0 to +10V	-25°C to +85°C	±0.003	±15
DAC702BH, DAC703BH	Ceramic DIP	±1mA, ±10V	-25°C to +85°C	±0.003	±15
DAC700BH/QM, DAC701BH/QM	Ceramic DIP	0 to -2mA, 0 to +10V	/QM Screening	±0.003	±15
DAC702BH/QM, DAC703BH/QM	Ceramic DIP	±1mA, ±10V	/QM Screening	±0.003	±15
DAC700LH, DAC701LH	Ceramic DIP	0 to -2mA, 0 to +10V	0°C to +70°C	±0.0015	±10
DAC700CH, DAC701CH	Ceramic DIP	0 to -2mA, 0 to +10V	-25°C to +85°C	±0.0015	±10
DAC700SH, DAC701SH	Ceramic DIP	0 to -2mA, 0 to +10V	-55°C to +125°C	±0.003	±15
DAC702LH, DAC703LH	Ceramic DIP	±1mA, ±10V	0°C to +70°C	±0.0015	±10
DAC702CH, DAC703CH	Ceramic DIP	±1mA, ±10V	-25°C to +85°C	±0.0015	±10
DAC702SH, DAC703SH	Ceramic DIP	±1mA, ±10V	-55°C to +125°C	±0.003	±15

BURN-IN SCREENING OPTION—See Text for Details.

MODEL	PACKAGE	OUTPUT CONFIGURATION	TEMPERATURE RANGE	LINEARITY ERROR, MAX AT +25°C (% of FSR)	BURN-IN TEMP. (160h) <sup>(1)</sup>
DAC702JP-BI	Plastic DIP	±1mA	0°C to +70°C	±0.006	85°C
DAC703JP-BI	Plastic DIP	±10V	0°C to +70°C	±0.006	85°C
DAC702KP-BI	Plastic DIP	±1mA	0°C to +70°C	±0.003	85°C
DAC703KP-BI	Plastic DIP	±10V	0°C to +70°C	±0.003	85°C
DAC703JU	Plastic SOIC	±10V	0°C to +70°C	±0.006	85°C
DAC703KU	Plastic SOIC	±10V	0°C to +70°C	±0.003	85°C
DAC700KH-BI	Ceramic DIP	0 to -2mA	0°C to +70°C	±0.003	85°C
DAC700LH-BI	Ceramic DIP	0 to -2mA	0°C to +70°C	±0.0015	85°C
DAC700BH-BI	Ceramic DIP	0 to -2mA	-25°C to +85°C	±0.003	85°C
DAC700CH-BI	Ceramic DIP	0 to -2mA	-25°C to +85°C	±0.0015	85°C
DAC700SH-BI	Ceramic DIP	0 to -2mA	-55°C to +125°C	±0.003	125°C
DAC701KH-BI	Ceramic DIP	0 to +10V	0°C to +70°C	±0.003	85°C
DAC701LH-BI	Ceramic DIP	0 to +10V	0°C to +70°C	±0.0015	85°C
DAC701BH-BI	Ceramic DIP	0 to +10V	-25°C to +85°C	±0.003	85°C
DAC701CH-BI	Ceramic DIP	0 to +10V	-25°C to +85°C	±0.0015	85°C
DAC701SH-BI	Ceramic DIP	0 to +10V	-55°C to +125°C	±0.003	125°C
DAC702KH-BI	Ceramic DIP	±1mA	0°C to +70°C	±0.003	85°C
DAC702LH-BI	Ceramic DIP	±1mA	0°C to +70°C	±0.0015	85°C
DAC702BH-BI	Ceramic DIP	±1mA	-25°C to +85°C	±0.003	85°C
DAC702CH-BI	Ceramic DIP	±1mA	-25°C to +85°C	±0.0015	85°C
DAC702SH-BI	Ceramic DIP	±1mA	-55°C to +125°C	±0.003	125°C
DAC703KH-BI	Ceramic DIP	±10V	0°C to +70°C	±0.003	85°C
DAC703LH-BI	Ceramic DIP	±10V	0°C to +70°C	±0.0015	85°C
DAC703BH-BI	Ceramic DIP	±10V	-25°C to +85°C	±0.003	85°C
DAC703CH-BI	Ceramic DIP	±10V	-25°C to +85°C	±0.0015	85°C
DAC703SH-BI	Ceramic DIP	±10V	-55°C to +125°C	±0.003	125°C

NOTE: (1) Or equivalent combination of time and temperature.

degree centigrade (ppm/°C). Gain drift is established by: (1) testing the end point differences for each D/A at  $t_{MIN}$ , +25°C and  $t_{MAX}$ ; (2) calculating the gain error with respect to the +25°C value; and (3) dividing by the temperature change.

### Zero Drift

Zero drift is a measure of the change in the output with  $FFFF_H$  (DAC700 and DAC701) applied to the digital inputs over the specified temperature range. For the bipolar models, zero is measured at  $7FFF_H$  (bipolar zero) applied to the digital inputs. This code corresponds to zero volts (DAC703) or zero milliamps (DAC702) at the analog output. The maximum change in offset at  $t_{MIN}$  or  $t_{MAX}$  is referenced to the zero error at +25°C and is divided by the temperature change.

This drift is expressed in parts per million of full scale range per degree centigrade (ppm of FSR/°C).

### SETTLING TIME

Settling time of the D/A is the total time required for the analog output to settle within an error band around its final value after a change in digital input. Refer to Figure 1 for typical values for this family of products.

### Voltage Output

Settling times are specified to ±0.003% of FSR (±1/2LSB for 14 bits) for two input conditions: a full-scale range change of 20V (DAC703) or 10V (DAC701) and a 1LSB change at the "major carry," the point at which the worst-



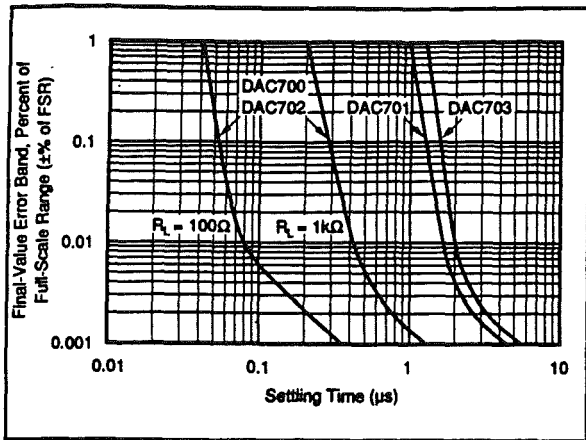


FIGURE 1. Final-Value Error Band vs Full-Scale Range Settling Time.

case settling time occurs. (This is the worst-case point since all of the input bits change when going from one code to the next).

#### Current Output

Settling times are specified to  $\pm 0.003\%$  of FSR for a full-scale range change for two output load conditions: one for  $10\Omega$  to  $100\Omega$  and one for  $1000\Omega$ . It is specified this way because the output RC time constant becomes the dominant factor in determining settling time for large resistive loads.

#### COMPLIANCE VOLTAGE

Compliance voltage applies only to current output models. It is the maximum voltage swing allowed on the output current pin while still being able to maintain specified accuracy.

#### POWER SUPPLY SENSITIVITY

Power supply sensitivity is a measure of the effect of a change in a power supply voltage on the D/A converter output. It is defined as a percent of FSR change in the output per percent of change in either the positive supply ( $+V_{CC}$ ), negative supply ( $-V_{CC}$ ) or logic supply ( $V_{DD}$ ) about the nominal power supply voltages (see Figure 2).

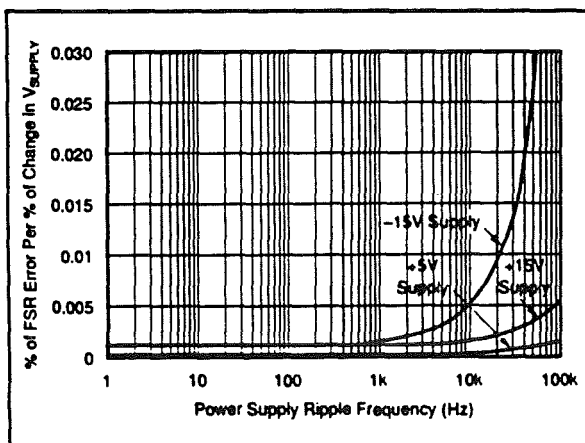


FIGURE 2. Power Supply Rejection vs Power Supply Ripple Frequency.

It is specified for DC or low frequency changes. The typical performance curve in Figure 2 shows the effect of high frequency changes in power supply voltages.

#### REFERENCE SUPPLY

All models have an internal low-noise  $+6.3V$  reference voltage derived from an on-chip buried zener diode. This reference voltage, available to the user, has a tolerance of  $\pm 5\%$  (KH models) and  $\pm 1\%$  (BH models). A minimum of  $1.5mA$  is available for external loads. Since the output impedance of the reference output is typically  $1W$ , the external load should remain constant.

If a varying load is to be driven by the reference supply, an external buffer amplifier is recommended to drive the load in order to isolate the bipolar offset (connected internally to the reference) from load variations.

#### BURN-IN SCREENING

Burn-in screening is an option available for selected models of the DAC700 family of products. Burn-in duration is 160 hours at the temperature shown below (or equivalent combination of time and temperature).

MODEL	TEMP. RANGE	BURN-IN SCREENING
DAC700BH-BI	$-25^{\circ}C$ to $+85^{\circ}C$	160 hours at $85^{\circ}C$
DAC702SH-BI	$-55^{\circ}C$ to $+125^{\circ}C$	160 hours at $125^{\circ}C$

All units are tested after burn-in to ensure that grade specifications are met. To order burn-in add "-BI" to the base model number.

#### ENVIRONMENTAL SCREENING

##### /QM Screening

Burr-Brown /QM models are environmentally screened versions of our standard industrial products designed to provide enhanced reliability. The screening, tabulated below, is performed to selected methods of MIL-STD-883. Reference to these methods provides a convenient method of communicating the screening levels and basic procedures employed; it does not imply conformance to any other military standards or to any methods of MIL-STD-883 other than those specified below. Burr-Brown's detailed procedures may vary slightly, model-to-model, from those in MIL-STD-883.

##### Screening Flow for /QM Models

SCREEN	MIL-STD-883 METHOD	CONDITION	COMMENTS
Internal Visual	2010	B	
High Temperature Storage (Stabilization Bake)	1008	C	$+150^{\circ}C$ 24h
Temperature Cycling	1010	C	$-65^{\circ}C$ to $+150^{\circ}C$ 10 cycles
Burn-in	1015	B	$-125^{\circ}C$ 160h
Constant Acceleration	2001	E	30.000G
Hermeticity Fine Leak	1014	A1 or A2	$5 \times 10^{-4}$ atm cc/s
Gross Leak	1014	C	60psig, 2h
External Visual	2009		

# OPERATING INSTRUCTIONS

## POWER SUPPLY CONNECTIONS

For optimum performance and noise rejection, power supply decoupling capacitors should be added as shown in the Connection Diagram. 1 $\mu$ F tantalum capacitors should be located close to the D/A converter.

## EXTERNAL ZERO AND GAIN ADJUSTMENT

Zero and gain may be trimmed by installing external zero and gain potentiometers. Connect these potentiometers as shown in the Connection Diagram and adjust as described below. TCR of the potentiometers should be 100ppm/ $^{\circ}$ C or less. The 3.9M $\Omega$  and 270k $\Omega$  resistors ( $\pm$ 20% carbon or better) should be located close to the D/A converter to prevent noise pickup. If it is not convenient to use these high-value resistors, an equivalent "T" network, as shown in Figure 3, may be substituted in place of the 3.9M $\Omega$  part. A 0.001 $\mu$ F to 0.01 $\mu$ F ceramic capacitor should be connected from Gain Adjust to Common to prevent noise pickup. Refer to Figures 4 and 5 for the relationship of zero and gain adjustments to unipolar and bipolar D/A converters.

### Zero Adjustment

For unipolar (CSB) configurations, apply the digital input code that produces zero voltage or zero current output and adjust the zero potentiometer for zero output.

For bipolar (COB, CTC) configurations, apply the digital input code that produces zero output voltage or current. See Table II for corresponding codes and the Connection Diagram for zero adjustment circuit connections. Zero calibration should be made before gain calibration.

### Gain Adjustment

Apply the digital input that gives the maximum positive output voltage. Adjust the gain potentiometer for this positive full scale voltage. See Table II for positive full scale voltages and the Connection Diagram for gain adjustment circuit connections.

## INSTALLATION CONSIDERATIONS

This D/A converter family is laser-trimmed to 14-bit linearity. The design of the device makes the 16-bit resolution available. If 16-bit resolution is not required, bit 15 and bit 16 should be connected to  $V_{DD}$  through a single 1k $\Omega$  resistor.

Due to the extremely high resolution and linearity of the D/A converter, system design problems such as grounding and contact resistance become very important. For a 16-bit converter with a 10V full-scale range, 1LSB is 153 $\mu$ V. With a load current of 5mA, series wiring and connector resistance of only 30m $\Omega$  will cause the output to be in error by 1LSB. To understand what this means in terms of a system layout, the resistance of #23 wire is about 0.021 $\Omega$ /ft. Neglecting contact resistance, less than 18 inches of wire will produce a 1LSB error in the analog output voltage!

In Figures 6, 7, and 8, lead and contact resistances are represented by  $R_1$  through  $R_4$ . As long as the load resistance  $R_L$  is constant,  $R_2$  simply introduces a gain error and can be removed during initial calibration.  $R_3$  is part of  $R_L$ , if the output voltage is sensed at Common, and therefore introduces no error. If  $R_1$  is variable, then  $R_2$  should be less than  $R_{L\text{MIN}}/2^{16}$  to reduce voltage drops due to wiring to less than 1LSB. For example, if  $R_{L\text{MIN}}$  is 5k $\Omega$ , then  $R_2$  should be less than 0.08 $\Omega$ .  $R_L$  should be located as close as possible to the D/A converter for optimum performance. The effect of  $R_4$  is negligible.

In many applications it is impractical to sense the output voltage at the output pin. Sensing the output voltage at the system ground point is permissible with the DAC700 family

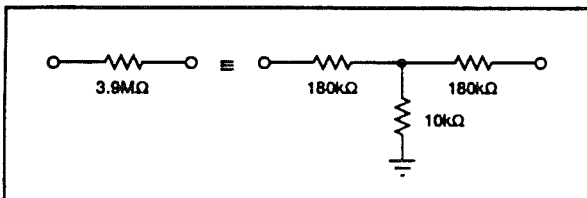


FIGURE 3. Equivalent Resistances.

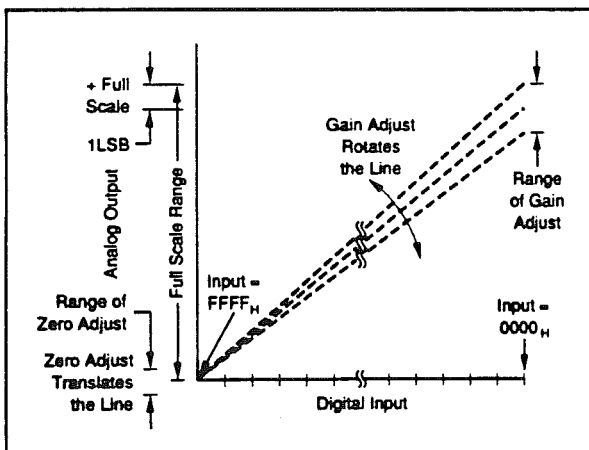


FIGURE 4. Relationship of Zero and Gain Adjustments for Unipolar D/A Converters, DAC700 and DAC701.

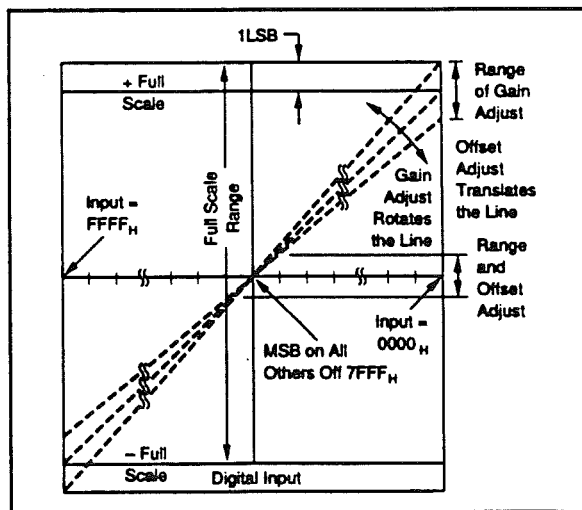


FIGURE 5. Relationship of Zero and Gain Adjustments for Bipolar D/A Converters, DAC702 and DAC703.

VOLTAGE OUTPUT MODELS						
DIGITAL INPUT CODE	ANALOG OUTPUT					
	DAC701 UNIPOLAR			DAC703 BIPOLAR		
	16-BIT	15-BIT	14-BIT	16-BIT	15-BIT	14-BIT
1LSB ( $\mu\text{V}$ )	153	305	610	305	610	1224
0000 <sub>H</sub> (V)	+9.99985	+9.99969	+9.99939	+9.99960	+9.99939	+9.99878
FFFF <sub>H</sub> (V)	0	0	0	-10.0000	-10.0000	-10.0000

ANALOG OUTPUT MODELS						
DIGITAL INPUT CODE	ANALOG OUTPUT					
	DAC700 UNIPOLAR			DAC702 BIPOLAR		
	16-BIT	15-BIT	14-BIT	16-BIT	15-BIT	14-BIT
1LSB ( $\mu\text{A}$ )	0.031	0.061	0.122	0.031	0.061	0.122
0000 <sub>H</sub> (mA)	-1.99997	-1.99994	-1.99988	-0.99997	-0.99994	-0.99988
FFFF <sub>H</sub> (mA)	0	0	0	+1.00000	+1.00000	+1.00000

TABLE II. Digital Input and Analog Output Relationships.

because the D/A converter is designed to have a constant return current of approximately 2mA flowing from Common. The variation in this current is under 20 $\mu\text{A}$  (with changing input codes), therefore  $R_4$  can be as large as 3 $\Omega$  without adversely affecting the linearity of the D/A converter. The voltage drop across  $R_4$  ( $R_4 \times 2\text{mA}$ ) appears as a zero error and can be removed with the zero calibration adjustment. This alternate sensing point (the system ground point) is shown in Figures 6, 7, and 8.

Figures 7 and 8 show two methods of connecting the current output models (DAC700 or DAC702) with external precision output op amps. By sensing the output voltage at the load resistor (ie, by connecting  $R_F$  to the output of  $A_1$  at  $R_L$ ), the effect of  $R_1$  and  $R_2$  is greatly reduced.  $R_1$  will cause a gain error but is independent of the value of  $R_L$  and can be eliminated by initial calibration adjustments. The effect of  $R_2$  is negligible because it is inside the feedback loop of the output op amp and is therefore greatly reduced by the loop gain.

If the output cannot be sensed at Common or the system ground point as mentioned above, the differential output circuit shown in Figure 8 is recommended. In this circuit the output voltage is sensed at the load common and not at the D/A converter common as in the previous circuits. The value of  $R_6$  and  $R_7$  must be adjusted for maximum common-mode rejection at  $R_L$ . Note that if  $R_3$  is negligible, the circuit of Figure 8 can be reduced to the one shown in Figure 7. Again the effect of  $R_4$  is negligible.

The D/A converter and the wiring to its connectors should be located to provide optimum isolation from sources of RFI and EMI. The key concept in elimination of RF radiation or

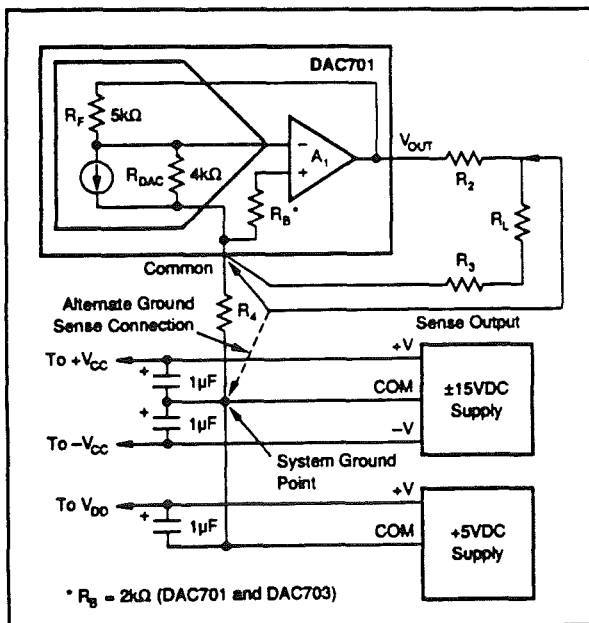


FIGURE 6. Output Circuit for Voltage Models.

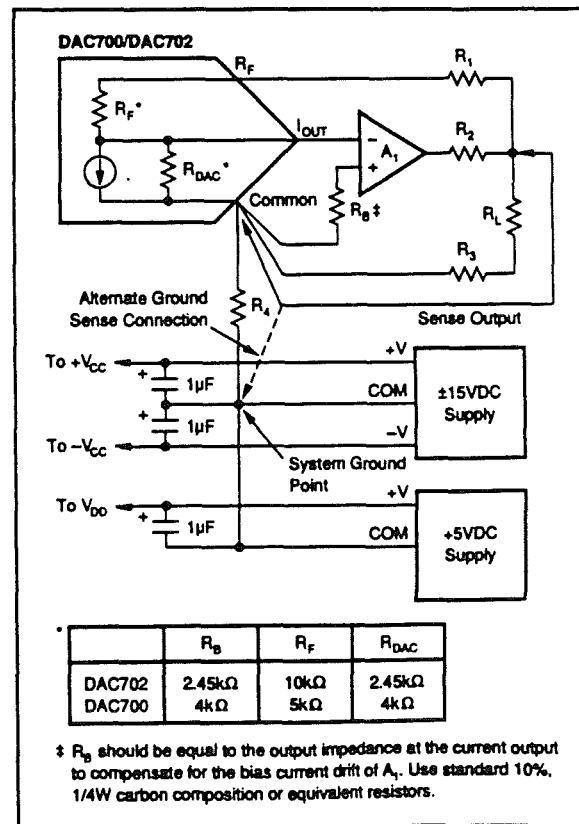


FIGURE 7. Preferred External Op Amp Configuration.

pickup is loop area; therefore, signal leads and their return conductors should be kept close together. This reduces the external magnetic field along with any radiation. Also, if a single lead and its return conductor are wired close together, they present a small flux-capture cross section for any external field. This reduces radiation pickup in the circuit.

## APPLICATIONS

### DRIVING AN EXTERNAL OP AMP WITH CURRENT OUTPUT D/AS

DAC700 and DAC702 are current output devices and will drive the summing junction of an op amp to produce an output voltage as shown in Figure 9. Use of the internal feedback resistor is required to obtain specified gain accuracy and low gain drift.

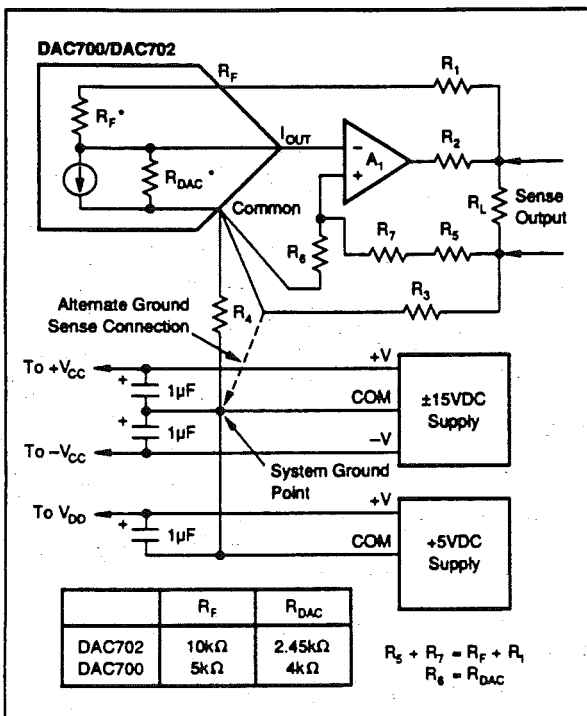


FIGURE 8. Differential Sensing Output Op Amp Configuration.

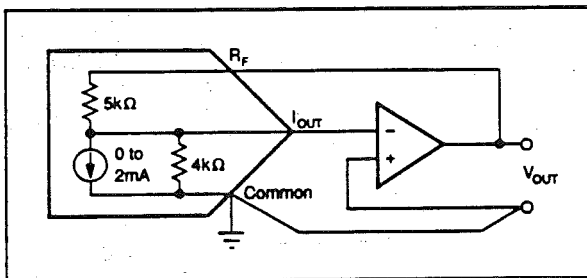


FIGURE 9. External Op Amp Using Internal Feedback Resistors.

DAC700 or DAC702 can be scaled for any desired voltage range with an external feedback resistor, but at the expense of increased drifts of up to  $\pm 50\text{ppm}/^\circ\text{C}$ . The resistors in the DAC700 and DAC702 ratio track to  $\pm 1\text{ppm}/^\circ\text{C}$  but their absolute TCR may be as high as  $\pm 50\text{ppm}/^\circ\text{C}$ .

An alternative method of scaling the output voltage of the D/A converter and preserving the low gain drift is shown in Figure 10.

### OUTPUTS LARGER THAN 20V RANGE

For output voltage ranges larger than  $\pm 10\text{V}$ , a high voltage op amp may be employed with an external feedback resistor. Use  $I_{\text{OUT}}$  values of  $\pm 1\text{mA}$  for bipolar voltage ranges and  $-2\text{mA}$  for unipolar voltage ranges (see Figure 11). Use protection diodes as shown when a high voltage op amp is used.

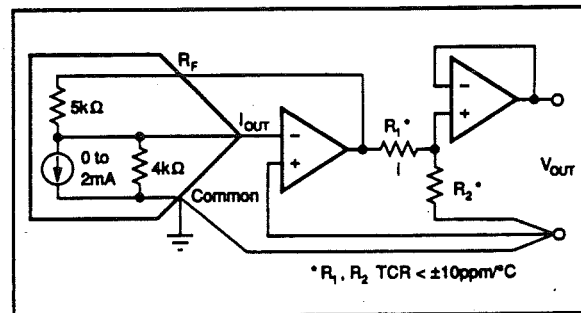


FIGURE 10. External Op Amp Using Internal and External Feedback Resistors to Maintain Low Gain Drift.

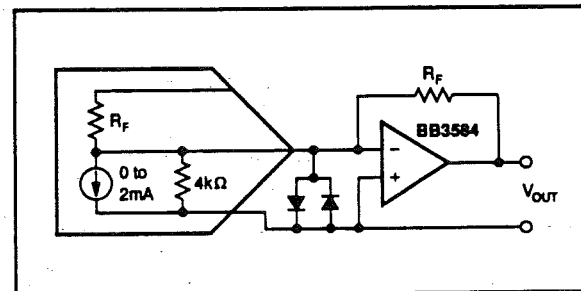


FIGURE 11. External Op Amp Using External Feedback Resistors.



## L6203

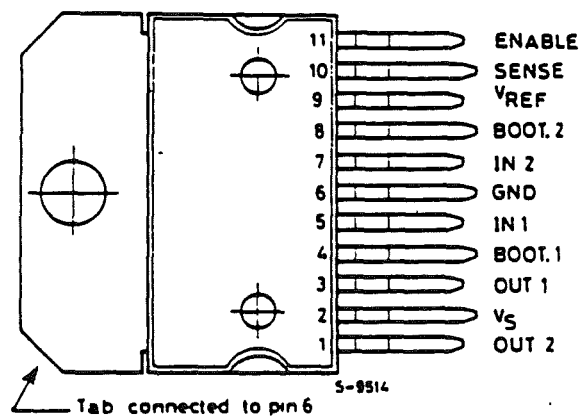
### ABSOLUTE MAXIMUM RATINGS

$V_s$	Power supply	52	V
$V_{OD}$	Differential output voltage (Between pins 1 and 3)	60	V
$V_{IN}, V_{EN}$	Input or Enable voltage	-0.3 to 7	V
$I_o$	Pulsed output current (note 1)	5	A
	- non repetitive (< 1ms)	10	A
$V_{sense}$	Sensing voltage	-1 to 4	V
$V_b$	Bootstrap peak voltage	60	V
$P_{tot}$	Total power dissipation ( $T_{case} = 90^\circ\text{C}$ )	20	W
	( $T_{amb} = 70^\circ\text{C}$ free air)	2.3	W
$T_{stg}, T_j$	Storage and junction temperature	-40 to 150	$^\circ\text{C}$

Note 1 : Pulse width limited only by junction temperature and the transient thermal impedance.

### CONNECTION DIAGRAM

(Top view)



### THERMAL DATA

$R_{th\ j-case}$	Thermal resistance junction-case	max	3	$^\circ\text{C/W}$
$R_{th\ j-amb}$	Thermal resistance junction-ambient	max	35	$^\circ\text{C/W}$

2/16

**ST** SCS-THOMSON  
MICROELECTRONICS

220

## V FUNCTIONS

V	NAME	FUNCTION
	OUT2	Output of the half bridge.
	V <sub>s</sub>	Supply voltage.
	OUT1	Output of the half bridge.
	BOOT1	A bootstrap capacitor connected to this pin ensures efficient driving of the upper POWER DMOS transistor at high switching frequencies.
5	IN1	Digital input from the motor controller.
6	GND	Common ground terminal.
7	IN2	Digital input from the motor controller.
8	BOOT2	A bootstrap capacitor connected to this pin ensures efficient driving of the upper POWER DMOS transistor at high switching frequencies.
9	V <sub>ref</sub>	Internal voltage reference, a capacitor from this pin to GND increases stability of the POWER DMOS logic drive circuit.
10	SENSE	A resistance R <sub>SENSE</sub> connected to this pin provides feedback for motor current control.
11	ENABLE	When a logic high is present on this pin the DMOS POWER transistors are enabled to be selectively driven by IN1 and IN2.

## L5203

**ELECTRICAL CHARACTERISTICS** (Refer to the test circuits  $T_j = 25^\circ\text{C}$ ,  $V_s = 42\text{V}$ , unless otherwise stated)

Parameter	Test Conditions	Min.	Typ.	Max.	Unit
$V_s$	Supply voltage	12	36	48	V
$V_{ref}$	Reference voltage		13.5		V
$I_{REF}$	Output current			2	mA
$I_s$	Quiescent supply current	EN = H $V_{IN} = L$ EN = H $V_{IN} = H$ EN = L Fig. 10	$I_L = 0$ 10 10 8		mA mA mA
$f_c$	Commutation frequency		30	100	KHz
$T_j$	Thermal shutdown		150		$^\circ\text{C}$
$T_d$	Dead time protection		100		ns

### TRANSISTORS

OFF					
$I_{DSS}$	Leakage current	Fig. 11 $V_s = 52\text{V}$		1	mA
ON					
$R_{DS}$	On resistance		0.3		$\Omega$
$V_{DS(ON)}$	Drain source voltage	$I_{DS} = 3\text{A}$	0.9		V
$V_{sens}$	Sensing voltage		-1	4	V

### SOURCE DRAIN DIODE

$V_{sd}$	Forward ON voltage	$I_{SD} = 3\text{A}$ EN = L	1.35		V
$t_{rr}$	Reverse recovery time	$I_F = 3\text{A}$ $\frac{df}{dt} = 25\text{A}/\mu\text{s}$	300		ns
$t_{fr}$	Forward recovery time		200		ns

### LOGIC LEVELS

$V_{INL}, V_{ENL}$	Input Low voltage		-0.3	0.8	V
$V_{INH}, V_{ENH}$	Input High voltage		2	7	V
$I_{INL}, I_{ENL}$	Input Low current	$V_{IN}, V_{EN} = L$		-10	$\mu\text{A}$
$I_{INH}, I_{ENH}$	Input High current	$V_{IN}, V_{EN} = H$	30		$\mu\text{A}$

### LOGIC CONTROL TO POWER DRIVE TIMING

$t_1 (V_i)$	Source current turn-off delay	Fig. 12	300		ns
$t_2 (V_i)$	Source current fall time	Fig. 12	200		ns
$t_3 (V_i)$	Source current turn-on delay	Fig. 12	400		ns
$t_4 (V_i)$	Source current rise time	Fig. 12	200		ns
$t_5 (V_i)$	Sink current turn-off delay	Fig. 13	300		ns
$t_6 (V_i)$	Sink current fall time	Fig. 13	200		ns
$t_7 (V_i)$	Sink current turn-on delay	Fig. 13	400		ns
$t_8 (V_i)$	Sink current rise time	Fig. 13	200		ns



Fig. 1 - Typical  $I_S$  normalized vs.  $T_J$

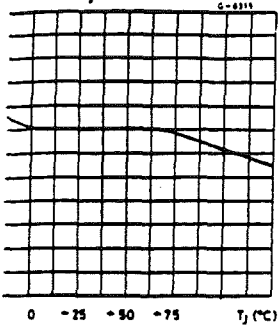


Fig. 2 - Quiescent current vs. frequency

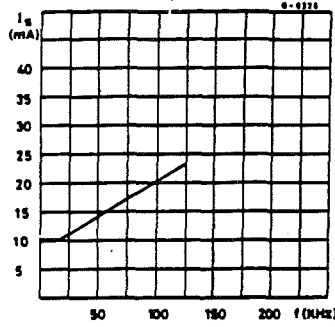


Fig. 3 - Typical  $I_S$  normalized vs.  $V_S$

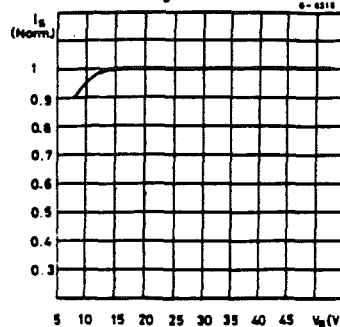


Fig. 4 - Typical diode behaviour in synchronous rectification

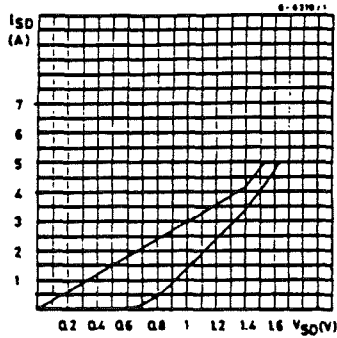


Fig. 5 - Typical  $R_{DS(ON)}$  vs.  $V_S \cong V_{REF}$

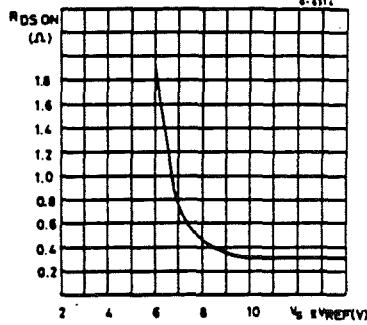


Fig. 6 -  $R_{DS(ON)}$  normalized at 25°C vs. temperature typical values

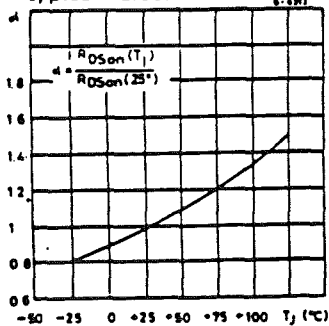
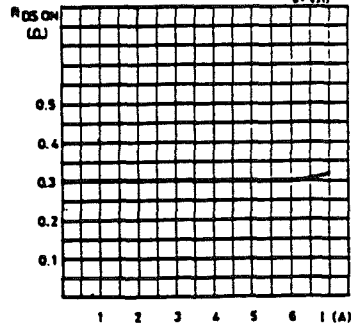


Fig. 7 -  $R_{DS(ON)}$  vs. DMOS transistor current



L6203

Fig. 8 - Typical power dissipation vs.  $I_L$

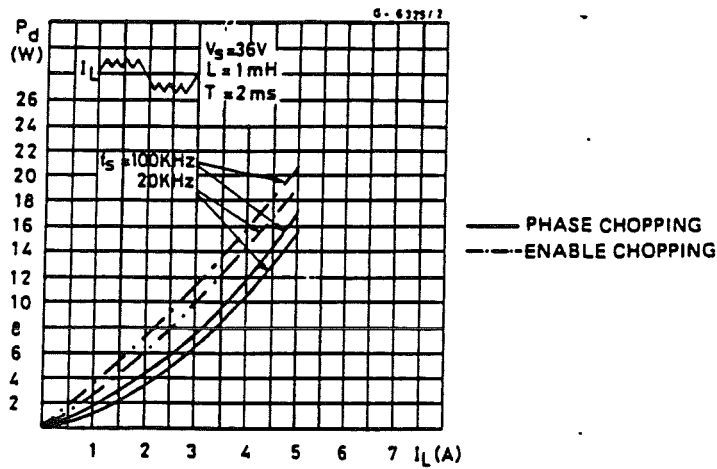


Fig. 8a - Two phase chopping

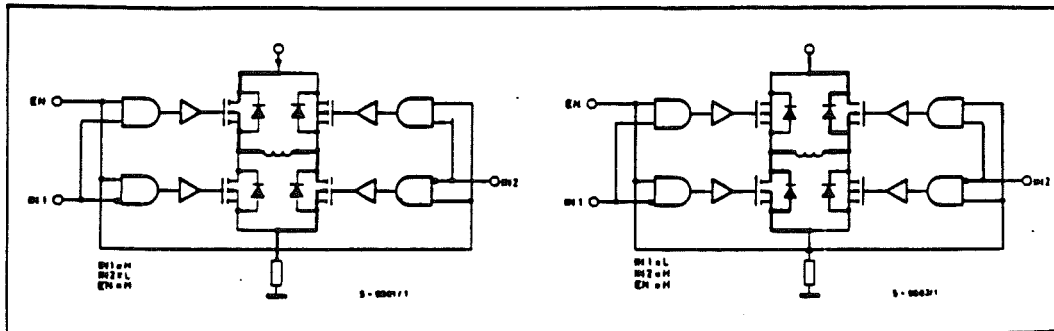


Fig. 8b - One phase chopping

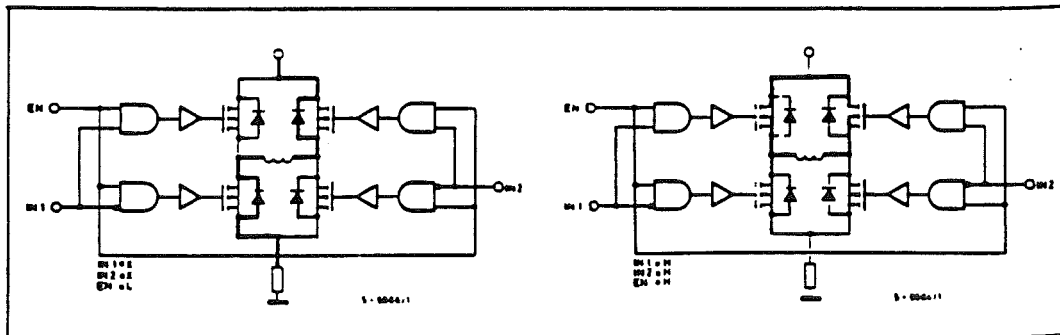
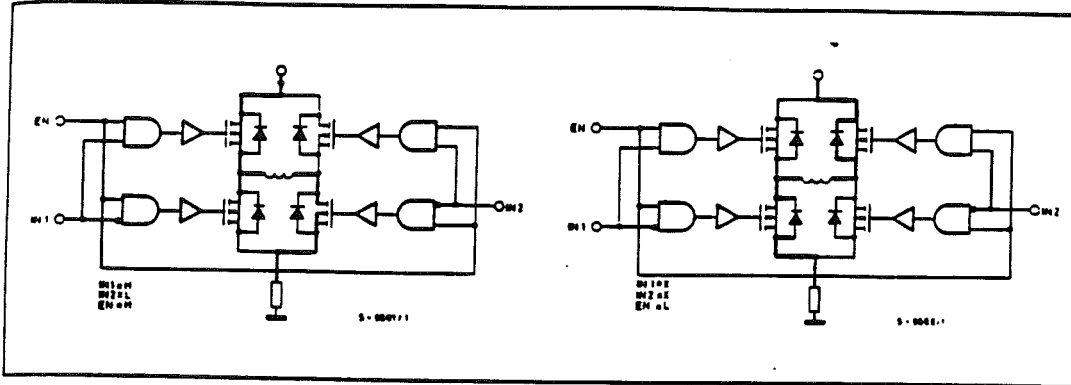


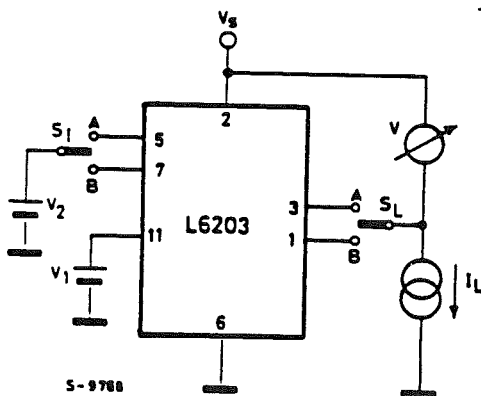
Fig. 8c - Enable chopping



TEST CIRCUITS

Fig. 9 - Saturation voltage

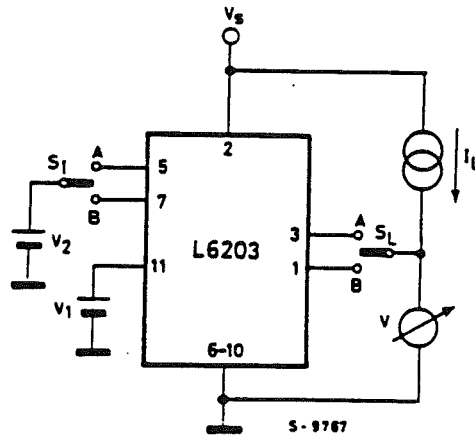
a) Source outputs



For IN1 source output saturation :  $V_1 = \text{"H"}$   
 $S_1 = A$   
 $S_L = A$  }  $V_2 = \text{"H"}$

For IN2 source output saturation :  $V_1 = \text{"H"}$   
 $S_1 = B$   
 $S_L = B$  }  $V_2 = \text{"H"}$

b) Sink outputs



For IN1 sink output saturation :  $V_1 = \text{"H"}$   
 $S_1 = A$   
 $S_L = A$  }  $V_2 = \text{"L"}$

For IN2 sink output saturation :  $V_1 = \text{"H"}$   
 $S_1 = B$   
 $S_L = B$  }  $V_2 = \text{"L"}$

# L6203

## TEST CIRCUITS (continued)

Fig. 10 - Quiescent current

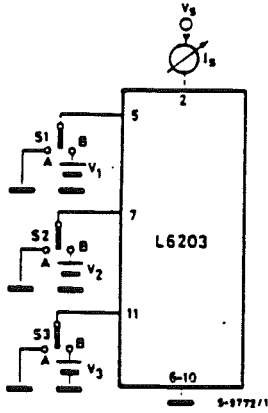
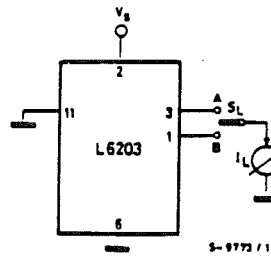
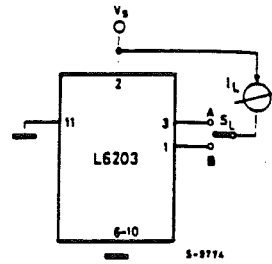


Fig. 11 - Leakage current

a) Source outputs



b) Sink outputs



## SWITCHING TIMES

Fig. 12 - Source current delay times vs. input chopper

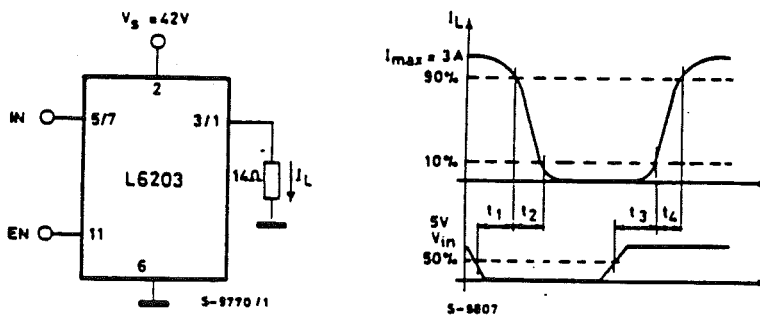
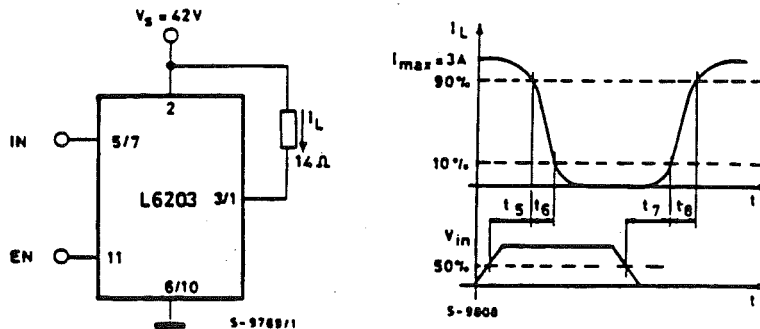


Fig. 13 - Sink current delay times vs. input chopper



**CIRCUIT DESCRIPTION**

The L6203 is a monolithic full bridge switching motor driver realized in the new Multipower-BCD technology which allows the integration of multiple, isolated DMOS power transistors plus mixed CMOS/bipolar control circuits. In this way it has been possible to make all the control inputs TTL, CMOS and  $\mu$ C compatible and eliminate the necessity of external MOS drive components.

**LOGIC DRIVE**

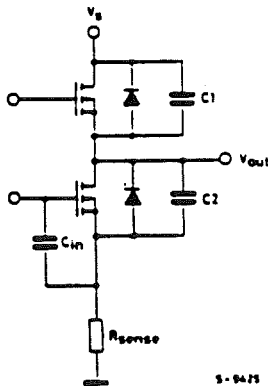
	INPUTS		OUTPUT MOSFETS (*)
	IN1	IN2	
$V_{EN} = H$	L	L	Sink 1, Sink 2
	L	H	Sink 1, Source 2
	H	L	Source 1, Sink 2
	H	H	Source 1, Source 2
$V_{EN} = L$	X	X	All transistors turned OFF

L = Low      H = High      X = Don't care  
 (\*) Members referred to INPUT 1 or INPUT2 controlled outputs stages

**CROSS CONDUCTION**

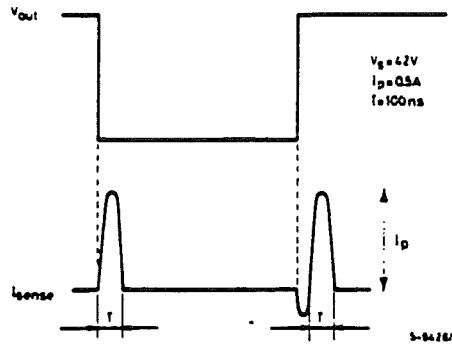
Although the device guarantees the absence of cross-conduction, the presence of the intrinsic diodes in the POWER DMOS structure causes the generation of current spikes on the sensing terminals. This is due to charge-discharge phenomena in the capacitors C1 & C2 associated with the drain source junctions (Fig. 14). When the output switches from high to low, a current spike is generated associated with the capacitor C1. On

Fig. 14 - Intrinsic structures in the POWER MOS transistors



the low-to-high transition a spike of the same polarity is generated by C2, preceded by a spike of the opposite polarity due to the charging of the input capacity of the lower POWER DMOS transistor. (Fig. 15)

Fig. 15 - Current typical spikes on the sensing pin



**TRANSISTOR OPERATION**

**ON STATE**

When one of the POWER DMOS transistor is ON it can be considered as a resistor  $R_{DS(ON)}$  ( $= 0.3\Omega$ ) throughout the recommended operating range. In this condition the dissipated power is given by:

$$P_{ON} = R_{DS(ON)} \cdot I_{DS}^2$$

The low  $R_{DS(ON)}$  of the Multipower-BCD process can provide high currents with low power dissipation.

**OFF STATE**

When one of the POWER DMOS transistor is OFF the  $V_{DS}$  voltage is equal to the supply voltage and only the leakage current  $I_{DSS}$  flows. The power dissipation during this period is given by:

$$P_{OFF} = V_s \cdot I_{DSS}$$

The power dissipation is in the order of pW and is negligible in comparison to that dissipated in the ON STATE.

**TRANSITIONS**

Like all MOS power transistors the DMOS POWER transistors have as intrinsic diode between their source and drain that can operate as a fast freewheeling diode in switched mode

## L6203

applications. During recirculation with the ENABLE input high, the voltage drop across the transistor is  $R_{DS(ON)} \cdot I_D$  and when the voltage reaches the diode voltage it is clamped to its characteristic. When the ENABLE input is low, the POWER MOS is OFF and the diode carries all of the recirculation current. The power dissipated in the transitional times in the cycle depends upon the voltage and current waveforms in the application.

$$P_{trans.} = I_{DS}(t) \cdot V_{DS}(t)$$

### BOOSTRAP CAPACITORS

To ensure that the POWER DMOS transistors are driven correctly gate source voltage of about 10V must be guaranteed for all of the N-channel DMOS transistors. This is no problem for the lower POWER DMOS transistors as their sources are referred to ground but a gate voltage greater than the supply voltage is necessary to drive the upper transistors. In the L6203 this is achieved by an internal charge pump circuit that guarantees correct DC drive in combination with the bootstrap circuit charges the external  $C_B$  capacitors when the source transistor is OFF and the sink transistor is ON giving lower commutation losses in switched mode operation. For efficient charging the value of the bootstrap capacitor should be greater than the input capacitance of the power transistor which is around 1nF. It is recommended that a capacitance of at least 10nF is used for the bootstrap. If a smaller capacitor is used there is a risk that the POWER transistors will not be fully turned on and they will show a higher  $R_{DS(ON)}$ . On the other hand if a elevated value is used it is possible that a current spike may be produced in the sense resistor.

### REFERENCE

To stabilise the internal drive circuit it is recommended that a capacitor be placed between this pin and ground. A value of 0.22 $\mu$ F should be sufficient for most applications.

This pin is also protected against a short circuit to ground.

### DEAD TIME

To protect the device against simultaneous conduction in both arms of the bridge and the

resulting rail to rail short, the logic control circuit provides a dead time greater than 40ns.

### THERMAL PROTECTION

A thermal protection circuit has been included that will disable the device if the junction temperature reaches 150°C. When the temperature has fallen to a safe level the device restarts under the control of the input and enable signals.

## APPLICATION INFORMATION

### RECIRCULATION

During recirculation with the ENABLE input high, the voltage drop across the transistor is  $R_{DS(ON)} \cdot I_L$  for voltages less than 0.7V and is clamped at a voltage depending on the characteristics of the source-drain diode for greater voltages. Although the device is protected against cross conduction, current spikes can appear on the current sense pin due to charge/discharge phenomena in the intrinsic source drain capacitances. In the application this does not cause any problems because the voltage created across the sense resistor is usually much less than the peak value, although a small RC filter can be added if necessary.

### POWER DISSIPATION

In order to achieve the high performance provided by the L6203 some attention must be paid to ensure that it has an adequate PCB area to dissipate the heat. The first stage of any thermal design is to calculate the dissipated power in the application, for this example the half step operation shown in figure 16 is considered.

### RISE TIME $T_r$

When an arm of the half bridge is turned on current begins to flow in the inductive load until the maximum current  $I_L$  is reached after a time  $T_r$ . The dissipated energy  $E_{OFF/ON}$  is in this case:

$$E_{OFF/ON} = [R_{DS(ON)} \cdot I_L^2 \cdot T_r] \cdot 2/3$$

**ON TIME  $T_{ON}$**

During this time the energy dissipated is due to the ON resistance of the transistors  $E_{ON}$  and the commutation  $E_{COM}$ . As two of the POWER DMOS transistors are ON  $E_{ON}$  is given by:

$$E_{ON} = I_L^2 \cdot R_{DS(ON)} \cdot 2 \cdot T_{ON}$$

In the commutation the energy dissipated is:

$$E_{COM} = V_s \cdot I_L \cdot T_{COM} \cdot f_{SWITCH} \cdot T_{ON}$$

Where:

$T_{COM}$  = Commutation Time and it is assumed that;

$T_{COM} = T_{TURN-ON} = T_{TURN-OFF} = 100ns$   
 $f_{SWITCH}$  = Chopper frequency

**FALL TIME  $T_f$**

For this example it is assumed that the energy dissipated in this part of the cycle takes the same form as that shown for the rise time:

$$E_{ON/OFF} = [R_{DS(ON)} \cdot I_L^2 \cdot T_f] \cdot 2/3$$

**QUIESCENT ENERGY**

The last contribution to the energy dissipation is due to the quiescent supply current and is given by:

$$E_{QUIESCENT} = I_{QUIESCENT} \cdot V_s \cdot T$$

**TOTAL ENERGY PER CYCLE**

$$E_{TOT} = E_{OFF/ON} + E_{ON} + E_{COM} + E_{ON/OFF} + E_{QUIESCENT}$$

The Total Power Dissipation  $P_{DIS}$  is simply:

$$P_{DIS} = E_{TOT}/T$$

- $T_r$  = Rise time
- $T_{ON}$  = ON time
- $T_f$  = Fall time
- $T_d$  = Dead time
- $T$  = Period

$$T = T_r + T_{ON} + T_f + T_d$$

Fig. 16



**DC MOTOR SPEED CONTROL**

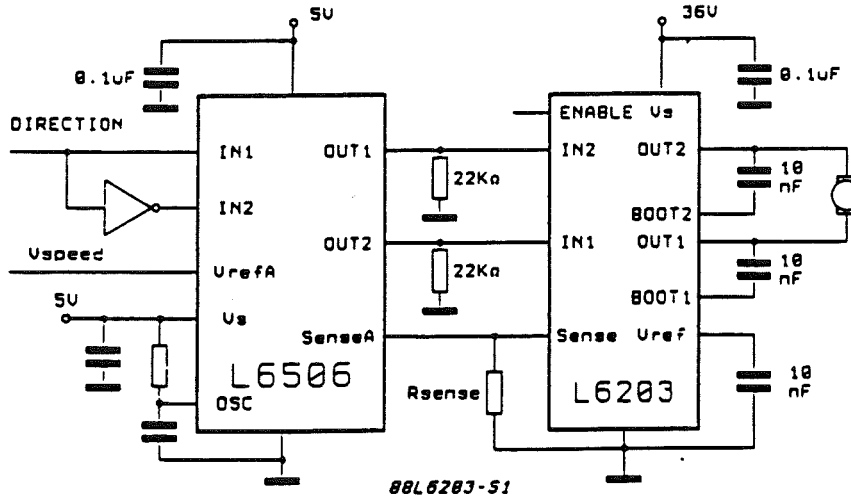
Since the L6203 integrates a full H-Bridge in a single package it is ideally suited for controlling small DC motors. When used for DC motor control the L6203 provides the power stage required for both speed and direction control. The L6203 can be combined with a current regulator like the L6506 to implement a transconductance amplifier for speed control, as shown in

figure 17. In this particular configuration only half of the L6506 is used and the other half of the device may be used to control a second motor.

In this configuration the L6506 sense the voltage across the sense resistor,  $R_{SENSE}$ , to monitor the motor current. The L6506 then compares the sensed voltage against the current control input and chops the input signals to the L6203 to control the motor current.

## L6203

Fig. 17 - Bidirectional DC motor control



### BIPOLAR STEPPER MOTORS APPLICATIONS

Bipolar stepper motors can be driven with an L297 or L6506, two L6203 bridge BCD drivers and very few external components. Together these three chips form a complete microprocessor-to-stepper motor interface.

As shown in Fig. 18 and Fig. 19, the controller connect directly to the two bridge BCD drivers. External component requirements are minimal: an RC network to set the chopper frequency and a resistive divider to establish the comparator reference voltage (pin 15 for L297, pin 10 and 15 for L6506). These solutions have a very high efficiency because of low power dissipation.

When the voltage drop across the  $R_{sense}$  is more negative than  $-0.4V$ , diodes must be used between each schottky sense output and ground.

Depending on the PCB configuration, a snubber network would be connected between pins 1 and 3 of each IC (Generally  $0.1\mu F$  in series to  $10\Omega$ ).

### HIGH CURRENT MICROSTEP DRIVE FOR STEPPER MOTORS

The L6203 can be used in conjunction with the L6217 to (figure 20) implement a high current microstepping controller for stepper motors. In this application the L6217 is used as a control

circuit and its outputs are used only to drive the inputs of the L6203. The application allows easy interface to a microprocessor since the L6217 may be connected directly to the microprocessor bus.

In the circuit shown in Figure 20, the L6217 senses the motor current by monitoring the voltage across the sense resistors,  $R_{sense}$ , and compares this value to the output of a 6 bit (7 bit if the L6217A is used) D to A Converter. The L6217 controls the current using a frequency modulated, constant off time, switching controller. The off time of each coil may be set using an external resistor and capacitor connected to PTA and PTB.

In this configuration the microprocessor simply loads the appropriate value for the direction of current flow through the coil and the data for the DAC into the L6217. The L6217 and L6203 then forms the complete interface between the micro and the motor.

When the pins 3 and 4 of the L6217 (Test A and B) are low, the bridges must be in tri-state condition.

For this reason two LM339 comparators must be used. The outputs of the comparators act on the enable inputs of the L6203 ICs.

A bilevel operation can be used for decreasing the minimum controllable load current. The mi-



minimum current that can be controlled is given by the following expression :

$$I_L \text{ (avg.)} = \frac{V_s}{R_{\text{sense}} + (2R_{\text{Dson}} + R_{\text{LOAD}})/DC}$$

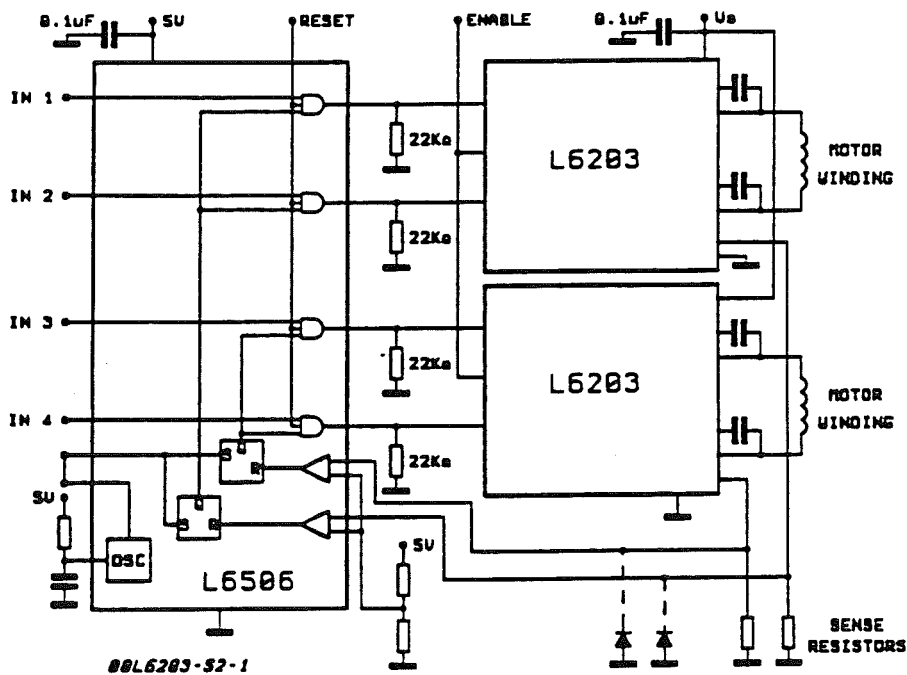
where  $R_{\text{LOAD}}$  is the equivalent resistance of the load DC is the duty cycle given by

$$\frac{T_{\text{on}}}{T_{\text{on}} + T_{\text{off}}}$$

If 12V is forced on pin (Reference voltage) and the supply voltage  $V_s$  is reduced below 12V the on resistance tends to increase above the normal guaranteed 0.3ohm.

Consequently the minimum current will also be reduced, as given in the above expression. When a minimum current operation is required, a high signal at point (A) can disable the pnp transistors in fig. 20. So it's possible to operate at a  $V_s$  of  $(7V - V_{\text{BE}})$ .

Fig. 18 - Two phase Bipolar stepper motor control circuit with chopper current control



# L6203

Fig. 19 - Two phase Bipolar stepper motor control circuit with chopper current control and translator

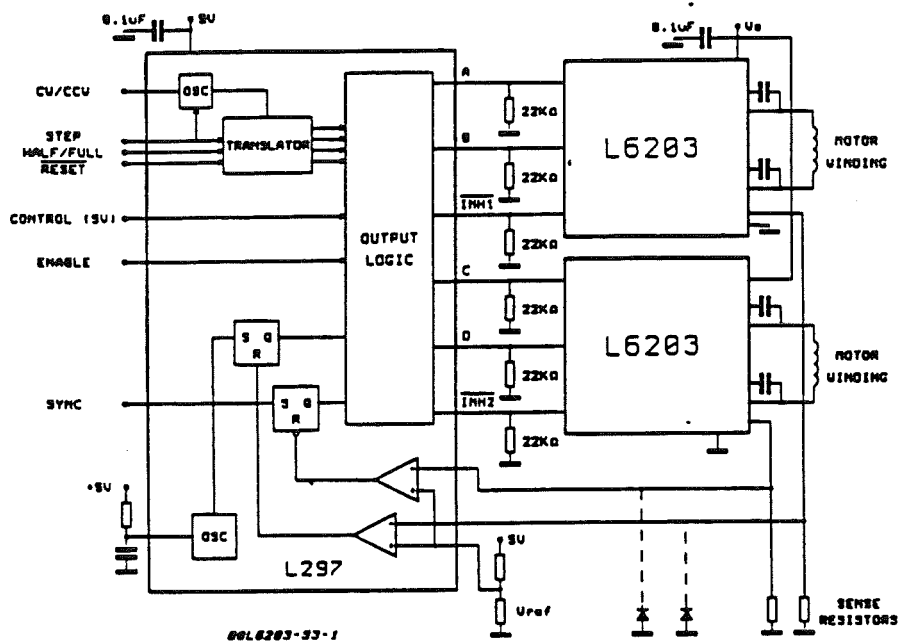
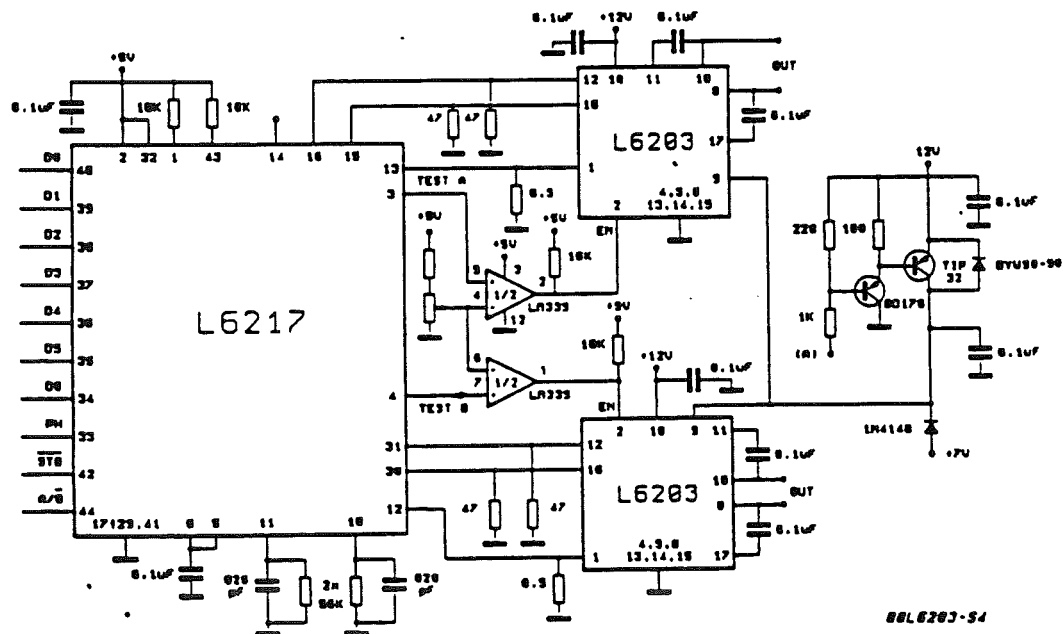


Fig. 20 - High current microstepping controller for stepper motors



THERMAL CHARACTERISTICS

Fig. 21 -  $R_{th(j-amb)}$  of Multiwatt package vs. dissipated power

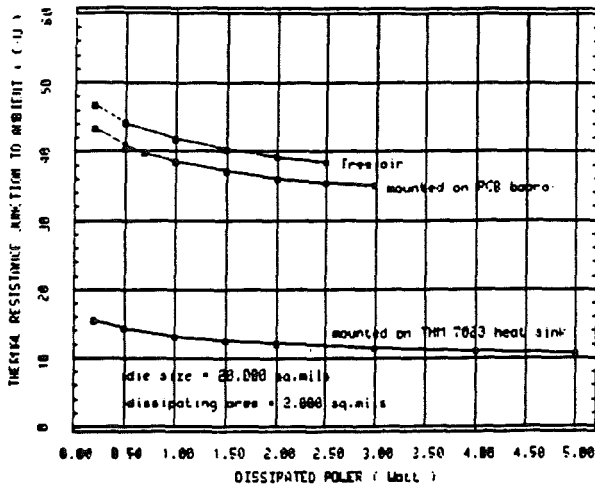
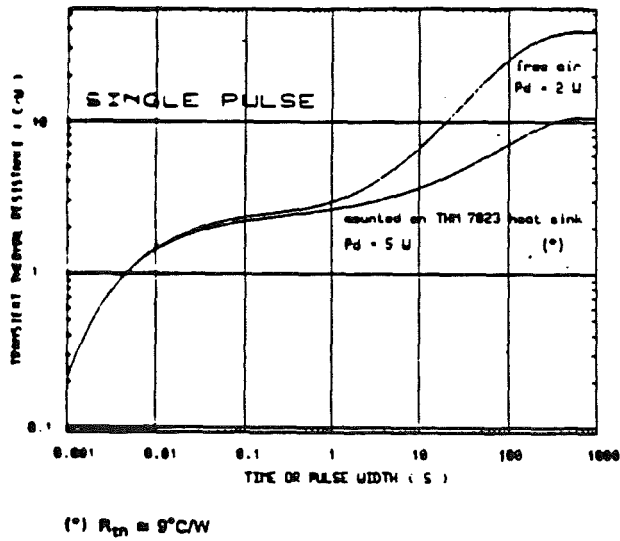
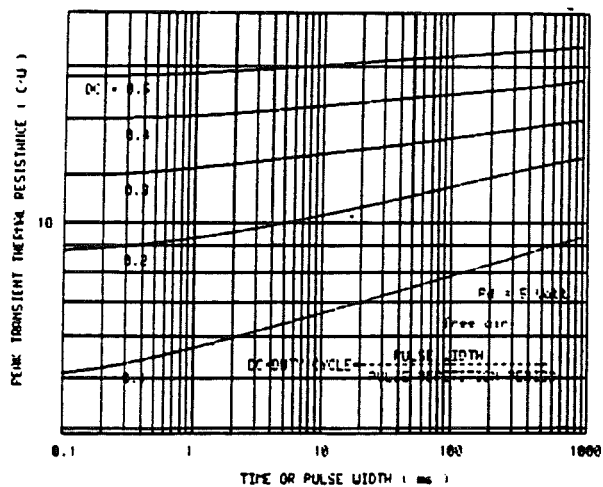


Fig. 22 - Comparison of transient  $R_{th}$  for single pulses with and without heatsink



L6203

Fig. 23 - Peak transient  $R_{th}$  vs. pulse width and duty cycle





## DS14C88/DS14C89A Quad CMOS Line Driver/Receiver

### General Description

The DS14C88 and DS14C89A, pin-for-pin replacements for the DS1488/MC1488 and the DS1489/MC1489, are line drivers/receivers designed to interface data terminal equipment (DTE) with data communications equipment (DCE). These devices translate standard TTL or CMOS logic levels to/from levels conforming to RS-232-C or CCITT V.24 standards.

Both devices are fabricated in low threshold CMOS metal gate technology. They provide very low power consumption in comparison to their bipolar equivalents; 900  $\mu$ A versus 26 mA for the receiver and 500  $\mu$ A versus 25 mA for the driver.

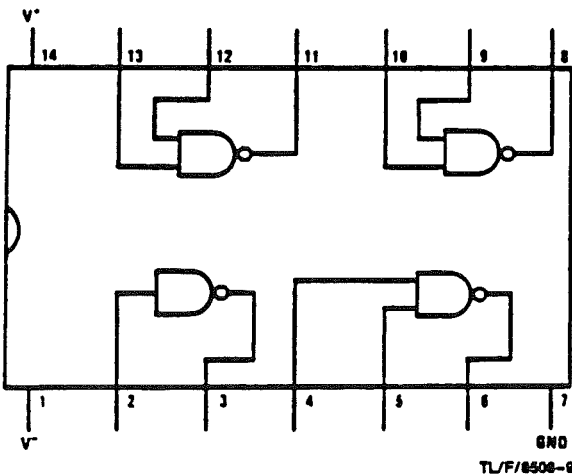
The DS14C88/DS14C89A simplify designs by eliminating the need for external capacitors. For the DS14C88, slew rate control in accordance with RS-232-C is provided on chip, eliminating the output capacitors. For the DS14C89A, noise pulse rejection circuitry eliminates the need for response control filter capacitors. When replacing the DS1489 with DS14C89A, the response control filter pins can be tied high, low or not connected.

### Features

- Meets EIA RS-232-C or CCITT V.24 standard
- Low power consumption
- Pin-for-pin equivalent to DS1488/MC1488 and DS1489/MC1489
- Low Delay Slew
- DS14C88 Driver
  - Power-off source impedance 300 $\Omega$  min.
  - Wide operating voltage range: 4.5V–12.6V
  - TTL/LSTTL compatible
- DS14C89A
  - Internal noise filter
  - Inputs withstand  $\pm$ 30V
  - Fail-safe operating mode
  - Internal input threshold with hysteresis

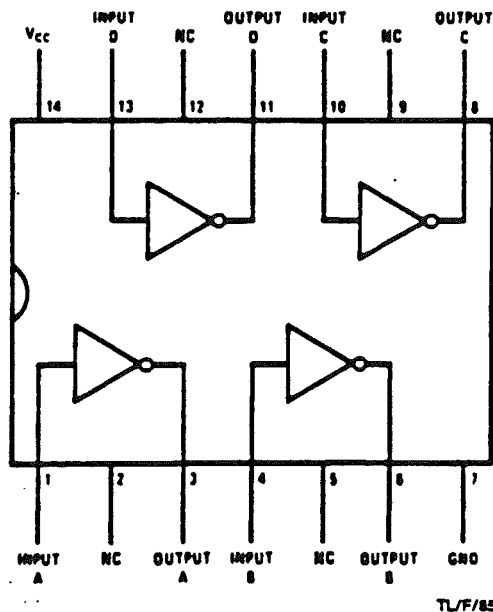
### Connection Diagrams

DS14C88 Dual-In-Line Package



Order Number DS14C88J, DS14C88N and DS14C88M  
 See NS Package Number J14A, M14A or N14A

DS14C89A Dual-In-Line Package



Order Number DS14C89AJ, DS14C89AM or DS14C89AN  
 See NS Package Number J14A, M14A or N14A

## DS14C88 Quad CMOS Line Driver

### Absolute Maximum Ratings (Note 1)

Specifications for Military/Aerospace products are not contained in this datasheet. Refer to the associated reliability electrical test specifications document.

Voltage at Any Input Pin	(V <sup>+</sup> ) + 0.3V to GND - 0.3V
Voltage at Any Output Pin	-25V to +25V
Storage Temp.	-65°C to +150°C
Power Dissipation	(See Note 2)
Junction Temperature	+150°C
Lead Temp. (Soldering 10 seconds)	+260°C

### Operating Conditions

	Min	Max
Supply Voltage V <sup>+</sup> (GND = 0V)	+4.5V	+12.6V
Supply Voltage V <sup>-</sup> (GND = 0V)	-4.5V	-12.6V
Temperature Range	0°C	+70°C

### DC Electrical Characteristics

T<sub>A</sub> = 0°C to +70°C, V<sup>+</sup> = 4.5V to 12V, GND = 0V, V<sup>-</sup> = -4.5V to -12V unless otherwise specified

Symbol	Parameter	Conditions	Min	Max	Units
I <sub>IL</sub> , I <sub>IH</sub>	Maximum Input Current	V <sub>IN</sub> = GND or V <sup>+</sup>	-10	10	μA
V <sub>IH</sub>	High Level Input Voltage		2.0	V <sub>DD</sub>	V
V <sub>IL</sub>	Low Level Input Voltage	V <sup>+</sup> ≥ 7V, V <sup>-</sup> ≤ -7V	GND	0.8	V
		V <sup>+</sup> ≤ 7V, V <sup>-</sup> ≥ -7V	GND	0.6	V
V <sub>OH</sub>	High Level Output Voltage	V <sub>IN</sub> = V <sub>IL</sub> R <sub>L</sub> = 3kΩ or 7kΩ V <sup>+</sup> = +4.5V, V <sup>-</sup> = -4.5V		3.0	V
		V <sup>+</sup> = +9V, V <sup>-</sup> = -9V		6.5	V
		V <sup>+</sup> = +12V, V <sup>-</sup> = -12V		9.0	V
V <sub>OL</sub>	Low Level Output Voltage	V <sub>IN</sub> = V <sub>IH</sub> R <sub>L</sub> = 3kΩ or 7kΩ V <sup>+</sup> = +4.5V, V <sup>-</sup> = -4.5V		-3.0	V
		V <sup>+</sup> = +9V, V <sup>-</sup> = -9V		-6.5	V
		V <sup>+</sup> = +12V, V <sup>-</sup> = -12V		-9.0	V
I <sub>OS+</sub>	High Level Output Short Circuit Current (Note 3)	V <sub>IN</sub> = V <sub>IL</sub> V <sub>OUT</sub> = GND V <sup>+</sup> = +12V, V <sup>-</sup> = -12V		+45	mA
I <sub>OS-</sub>	Low Level Output Short Circuit Current (Note 3)	V <sub>IN</sub> = V <sub>IH</sub> V <sub>OUT</sub> = GND V <sup>+</sup> = +12V, V <sup>-</sup> = 12V		-45	mA
R <sub>OUT</sub>	Output Resistance	V <sup>+</sup> = V <sup>-</sup> = 0V -2V ≤ V <sub>OUT</sub> ≤ 2V	300		Ω
I <sub>CC+</sub>	Positive Supply Current (per package)	V <sub>IN</sub> = V <sub>IL</sub> , R <sub>L</sub> = open V <sup>+</sup> = +4.5V, V <sup>-</sup> = -4.5V		10	μA
		V <sup>+</sup> = +9V, V <sup>-</sup> = -9V		30	μA
		V <sup>+</sup> = +12V, V <sup>-</sup> = -12V		60	μA
		V <sub>IN</sub> = V <sub>IH</sub> , R <sub>L</sub> = open V <sup>+</sup> = +4.5V, V <sup>-</sup> = -4.5V		30	μA
		V <sup>+</sup> = +9V, V <sup>-</sup> = -9V		190	μA
		V <sup>+</sup> = +12V, V <sup>-</sup> = -12V		425	μA
I <sub>CC-</sub>	Negative Supply Current (per package)	V <sub>IN</sub> = V <sub>IL</sub> , R <sub>L</sub> = open V <sup>+</sup> = +4.5V, V <sup>-</sup> = -4.5V		-10	μA
		V <sup>+</sup> = +9V, V <sup>-</sup> = -9V		-10	μA
		V <sup>+</sup> = +12V, V <sup>-</sup> = -12V		-10	μA
		V <sub>IN</sub> = V <sub>IH</sub> , R <sub>L</sub> = open V <sup>+</sup> = +4.5V, V <sup>-</sup> = -4.5V		-30	μA
		V <sup>+</sup> = +9V, V <sup>-</sup> = -9V		-30	μA
		V <sup>+</sup> = +12V, V <sup>-</sup> = -12V		-60	μA

Note 1: "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. Except for "Operating Temperature Range" they are not meant to imply that the devices should be operated at these limits. The table of "Electrical Characteristics" provides conditions for actual device operation.

Note 2: Power Dissipation N-Package: 1300 mW at 25°C, J-Package: 1000 mW at 25°C.

Note 3: I<sub>OS+</sub> and I<sub>OS-</sub> values are for one output at a time. If more than one output is shorted simultaneously, the device dissipation may be exceeded.

## DS14C88 Quad CMOS Line Driver

**AC Electrical Characteristics**  $T_A = 0^\circ\text{C to } +70^\circ\text{C}$ ,  $V^+ = +4.5\text{V to } 12\text{V}$ ,  $\text{GND} = 0\text{V}$ ,  
 $V^- = -4.5\text{V to } -12\text{V}$ ,  $R_L = 3\text{ k}\Omega$ ,  $C_L = 50\text{ pF}$  unless otherwise specified (Notes 4 and 5).

Symbol	Parameter	Conditions	Min	Typ	Max	Units
$t_{pd1}$	Propagation Delay to a Logic "1"	$V^+ = +4.5\text{V}, V^- = -4.5\text{V}$			6.0	$\mu\text{s}$
		$V^+ = +9\text{V}, V^- = -9\text{V}$			5.0	$\mu\text{s}$
		$V^+ = +12\text{V}, V^- = -12\text{V}$			4.0	$\mu\text{s}$
$t_{pd0}$	Propagation Delay to a Logic "0"	$V^+ = +4.5\text{V}, V^- = -4.5\text{V}$			6.0	$\mu\text{s}$
		$V^+ = +9\text{V}, V^- = -9\text{V}$			5.0	$\mu\text{s}$
		$V^+ = +12\text{V}, V^- = -12\text{V}$			4.0	$\mu\text{s}$
$t_r, t_f$	Output Rise and Fall Time (Note 6)		0.2			$\mu\text{s}$
$t_{SK}$	Typical Propagation Delay Skew	$V^+ = 12\text{V}, V^- = -12\text{V}$		400		ns
$S_R$	Output Slew Rate (Note 6)	$R_L = 3\text{ k}\Omega \text{ to } 7\text{ k}\Omega$ $15\text{ pF} \leq C_L \leq 2.5\text{ nF}$			30	$\text{V}/\mu\text{s}$

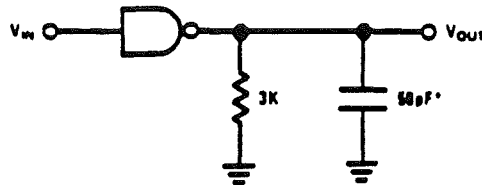
Note 4: AC input waveforms for test purposes:

$$t_r = t_f \leq 20\text{ ns}, V_{IH} = 2.0\text{V}, V_{IL} = 0.8\text{V (0.6V at } V^+ = 4.5, V^- = -4.5\text{V)}$$

Note 5: Input rise and fall times must not exceed  $5\text{ }\mu\text{s}$ .

Note 6: The output slew rate, rise time, and fall time are measured by measuring the time from  $+3.0\text{V}$  to  $-3.0\text{V}$  on the output waveforms.

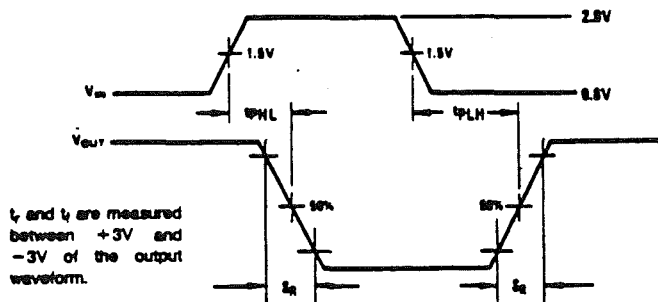
### AC Load Circuit



\* $C_L$  includes probe and jig capacitance.

TL/F/8508-7

### Switching Time Waveforms



TL/F/8508-8

# DS14C89A Quad CMOS Line Receiver

## Absolute Maximum Ratings

Specifications for Military/Aerospace products are not contained in this datasheet. Refer to the associated reliability electrical test specifications document.

Voltage at Any Input Pin	-30V to +30V
Voltage at Any Output Pin	(V <sub>OL</sub> ) = 0.3V to 0.4V, (V <sub>OH</sub> ) = 0.3V to 0.4V
Storage Temperature	-65°C to +150°C
Power Dissipation	500 mW at T <sub>A</sub> = 25°C
Junction Temperature	+150°C
Lead Temp. (Soldering 10 sec)	+260°C

## Operating Conditions

Supply Voltage (V <sub>CC</sub> )	+4.5V to +5.5V
Operating Temperature	-55°C to +125°C

## DC Electrical Characteristics

T<sub>A</sub> = 0°C to +70°C, +4.5 ≤ V<sub>CC</sub> ≤ 5.5V, GND = 0V, unless otherwise specified.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
V <sub>TH</sub>	Input High Threshold Voltage		1.5		2.0	V
V <sub>TL</sub>	Input Low Threshold Voltage		0.5		1.0	V
V <sub>H</sub>	Typical Input Hysteresis			1		V
I <sub>IN</sub>	Input Current	V <sub>IN</sub> = +2.5V V <sub>IN</sub> = -2.5V V <sub>IN</sub> = +3V V <sub>IN</sub> = -3V	3.0 2.0 -0.40 -0.40		0.1 0.1 -0.1 -0.1	mA
V <sub>OH</sub>	Output High Voltage	V <sub>IN</sub> = V <sub>TL</sub> (min) I <sub>OUT</sub> = -3.2 mA	2.0			V
V <sub>OL</sub>	Output Low Voltage	V <sub>IN</sub> = V <sub>TH</sub> (max) I <sub>OUT</sub> = +3.2 mA			1.0	V
I <sub>CC</sub>	Supply Current	R <sub>L</sub> = open V <sub>IN</sub> = V <sub>TH</sub> (max) or V <sub>TL</sub> (min)			0.001	mA

## AC Electrical Characteristics

T<sub>A</sub> = 0°C to +70°C, +4.5V ≤ V<sub>CC</sub> ≤ +5.5V, GND = 0V, CL = 50 pF, unless otherwise specified (Note 2).

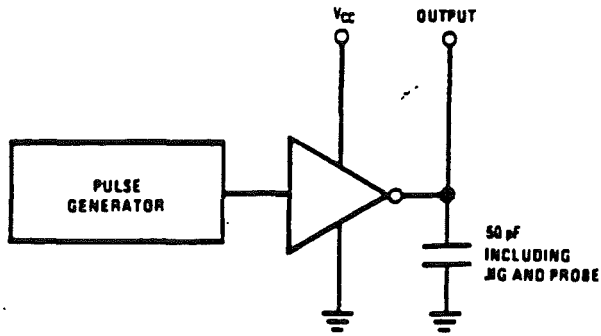
Symbol	Parameter	Conditions	Min	Typ	Max	Units
t <sub>PLH</sub>	Propagation Delay to a Logic "1"	Input pulse width ≥ 10 μs			6.1	n-s
t <sub>PHL</sub>	Propagation Delay to a Logic "0"	Input pulse width ≥ 10 μs			6.1	n-s
t <sub>SK</sub>	Typical Propagation Delay Skew			400		ns
t <sub>r</sub>	Output Rise Time				300	ns
t <sub>f</sub>	Output Fall Time				300	ns
t <sub>rw</sub>	Pulse Width Assumed to be Noise				1.0	n-s

Note 1: "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. Except for "Soldering Temperature Range" they are not meant to imply that the devices should be operated at these limits. The table of "Electrical Characteristics" provides conditions for actual device operation.

Note 2: AC input waveform for test purposes: t<sub>r</sub> = t<sub>f</sub> = 200 ns, V<sub>H</sub> = +3V, V<sub>L</sub> = -3V, f = 20 kHz.

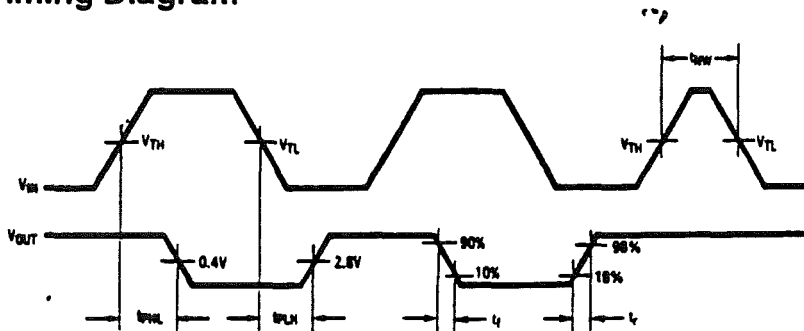


## DS14C89A AC Test Circuit



TL/F/8508-4

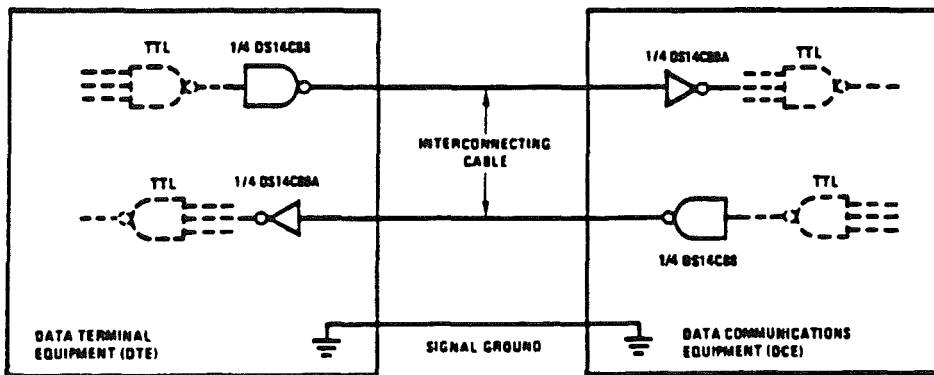
## DS14C89A Timing Diagram



TL/F/8508-5

## Typical Applications for DS14C88 and DS14C89A

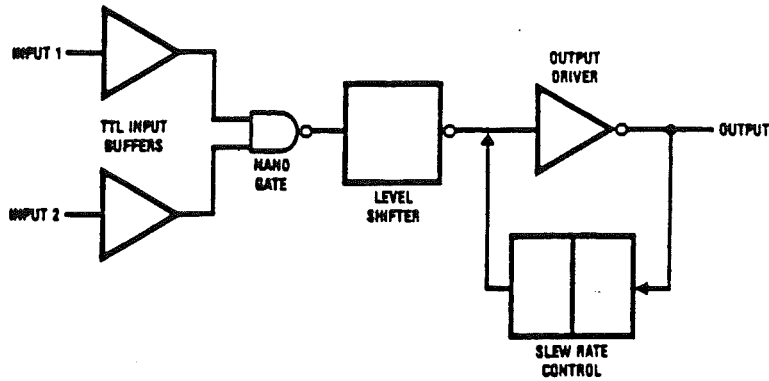
### RS232C Data Transmission



TL/F/8508-3

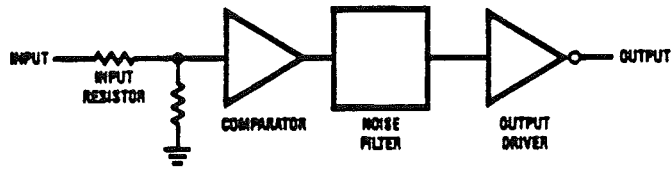
## Block Diagrams

DS14C88  
( $\frac{1}{4}$  circuit shown)

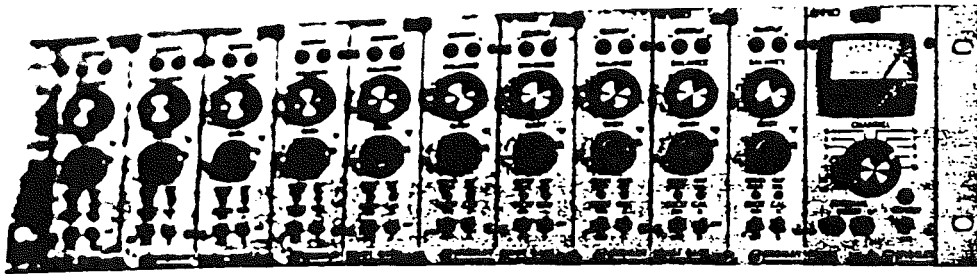


TL/F/8508-6

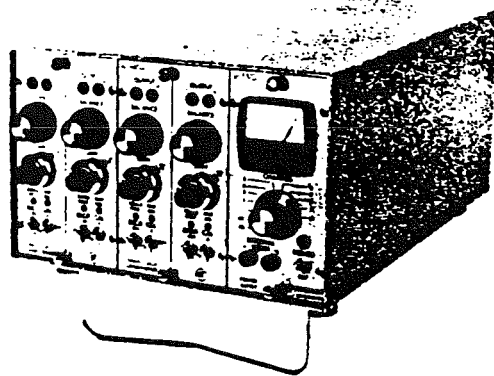
DS14C89A  
( $\frac{1}{4}$  circuit shown)



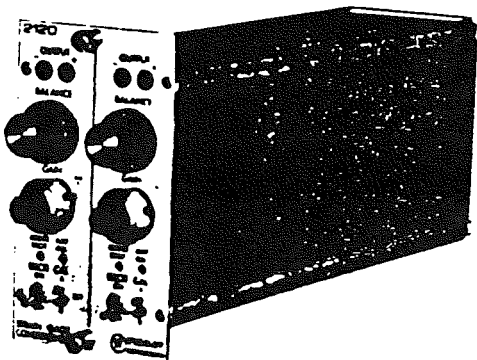
TL/F/8508-1



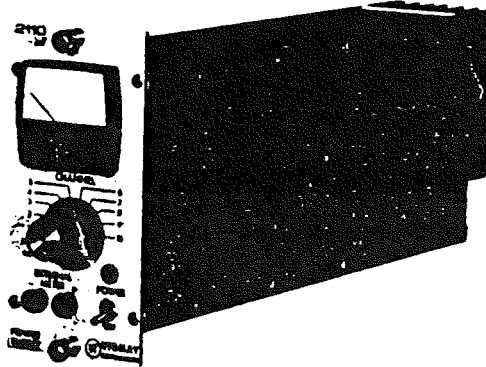
COMPLETE 10-CHANNEL 2100 SYSTEM



4-CHANNEL SYSTEM IN 2160 PORTABLE ENCLOSURE



2120 CONDITIONER MODULE



2110 POWER SUPPLY MODULE



2160 10-CHANNEL RACK ADAPTER

## DESCRIPTION

### GENERAL

1.1 The Series 2100 modules comprise a multichannel system for generating conditioned high-level signals from strain gage inputs for display or recording on external equipment. A system would be comprised of:

- a) One or more 2-channel 2120 Strain Gage Conditioners;
- b) One or more 2110 Power Supplies (each Power Supply will handle up to 10 channels; i.e., five 2120 Conditioner/Amplifiers); and,
- c) One or more rack adapters or cabinets, complete with wiring, to accept the above modules.

1.2 The principal features of the system include:

- ... Independently variable and regulated excitation for each channel (1 to 12 Vdc).
- ... Independently continuously variable gain (100 to 2100) for each channel.
- ... Bridge-completion components to accept quarter- (120Ω and 350Ω), half- and full-bridge inputs to each channel.
- ... LED null indicators on each channel - always active.
- ... 100 mA output.
- ... All supplies and outputs short-circuit proof with current limiting.
- ... Compact packaging - 10 channels in 5-¼ x 19 in (133 x 483 mm) rack space.

### SPECIFICATIONS

All specifications nominal or typical at +23°C unless noted.

#### 1.3 2120 STRAIN GAGE CONDITIONER

Note: These specifications apply for each of two independent channels per module.

INPUTS	Quarter (120Ω and 350Ω), half and full bridge (50-1000Ω). Quarter-bridge dummy resistors provided.
BRIDGE EXCITATION	1 to 12 Vdc (adjustable for each channel) with 120Ω full bridge load. Short-circuit current: less than 40 mA. Ripple, noise, and 10% line change: 2 mV max. Load regulation: ±0.2% no-load to 120Ω load.
BRIDGE BALANCE	±2000 με (quarter, half, or 350Ω full bridge); range can be changed by internal resistance change.
CALIBRATION	Two-position (center off) toggle switch. Standard factory-installed resistors (±0.1%) simulate ±1000 με at GF=2.
AMP GAIN	100 to 2100 continuously adjustable. Gain equal to twice dial reading (±2%).
BANDPASS	dc to 5 kHz (min): ±0.5 dB (±5%) dc to 15 kHz: ±3 dB
AMP INPUT	Temperature coefficient: ±2 μV RTI/°C (max) [Referred to Input]. Warm-up: 25 μV RTI over 30 minutes. Noise and drift (pk-pk): dc - 10 μV RTI/day;

0.01 to 100 Hz - 3 μV RTI (max);  
1 Hz up - 10 μV RTI (max).

Input impedance: >25 MΩ differential or common mode (balance limit resistor removed).

CMR: dc - 120 dB (min);  
0 to 500 Hz - 100 dB

Input bias: +0.1 μA each input.

### OUTPUT

±10 V (min) at 30 mA;  
±50 mA (min) into 150Ω load;  
±100 mA (min) into 15Ω load.  
Current limit: 120 mA.  
Linearity: ±0.05% at dc.

### SIZE & WEIGHT

5.25 in H x 2.94 in W x 10.97 in D  
(133 mm x 75 mm x 279 mm)  
2.2 lb (1.0 kg)

#### 1.4 2110 POWER SUPPLY

##### OUTPUTS

±15 V at 1.2 A and +17.5 V at 1.1 A; all regulators current-limited against overload.

##### INPUT

107, 115, 214, 230 Vac ±10% (selected internally); 50-60 Hz.

Power: 40 W typical, 100 W max.

##### METER

0 to 12 Vdc (with switch) to read bridge excitation.

Also ac input and dc output go/no-go monitor.

### SIZE & WEIGHT

5.25 in H x 2.44 in W x 12.34 in D  
(133 mm x 62 mm x 313 mm)  
6.7 lb (3.1 kg)

#### 1.5 2150 RACK ADAPTER

##### APPLICATION

Fits standard 19-in (483-mm) electronic equipment rack.

Accepts one 2110 Power Supply and one to five 2120 Strain Gage Conditioners.

Completely wired.

##### POWER

2-ft (0.6-m) 3-wire line cord; 10-ft (3-m) extension available.

Fuse: 1 A size 3 AG (32 mm x 6.4 mm dia.).

Receptacle to accept line cord from adjacent 2150 Rack Adapter.

### SIZE & WEIGHT

5.25 in H x 19 in W x 12.82 in D overall (133 mm x 483 mm x 325 mm)  
6.6 lb (3.0 kg)

#### 1.6 2160 PORTABLE ENCLOSURE

##### DESCRIPTION

Completely self-contained adapter and cabinet with all wiring for two or four channels.

Accepts one 2110 Power Supply and one or two 2120 Strain Gage Conditioners.

##### POWER

8-ft (2.4-m) detachable 3-wire cord.

Fuse: 1 A size 3 AG (32 mm x 6.4 mm dia.).

### SIZE & WEIGHT

5.50 in H x 8.75 in W x 13.13 in D  
(140 mm x 222 mm x 333 mm)  
5.2 lb (2.4 kg).