PAGE NUMBERING AS ORIGINAL

PAGE

MISSING

IN

ORIGINAL

HYBRID FORCE AND POSITION CONTROL IN ROBOTIC SURFACE PROCESSING

Andrew STEVEN

Thesis submitted for the Degree of Doctor of Philosophy

March 1989

Department of Mechanical, Materials
and Manufacturing Engineering
University of Newcastle upon Tyne
Newcastle upon Tyne NE1 7RU
England UK

To my wife Sara and my parents,

Frank and Maria Steven, for their love,

support and encouragement.

ABSTRACT

This programme of research was supported by NEI Parsons Ltd. who sought a robotic means of polishing mechanical components.

A study of the problems associated with robot controlled surface processing is presented. From this evolved an approach consistent with the formalisation of the demands of workpiece manipulation which included the adoption of the Hybrid robot control scheme capable of simultaneous force and position control.

A unique 3 axis planar experimental manipulator was designed which utilized combined parallel and serial drives. A force sensing wrist was used to measure contact force. A variant of the Hybrid control scheme was successfully implemented on a twin computer control system. A number of manipulator control programs are presented.

The force control aspect is shown both experimentally and analytically to present control problems and the research has concentrated on this aspect.

A general analysis of the dynamics of force control is given which shows force response to be dependent on a number of important parameters including force sensor, environment and manipulator dynamics. The need for a robust or adaptable force controller is discussed.

A series of force controlled manipulator experiments is described and the results discussed in the context of general analyses and specific single degree of freedom simulations. Improvements to manipulator force control are suggested and some were implemented. These are discussed together with their immediate application to the improvement of robot controlled surface processing.

This work also lays important foundations for long term related research. In particular the new techniques for actively controlled assembly and force control under 'fast' operation.

ACKNOWLEDGEMENTS

The author would like to thank Professor J. R. Hewit for the initiation of this research and for his advice and assistance as supervisor. He would also like to express gratitude for the help and support of Professor L. Maunder, who took over as supervisor upon the departure of Professor Hewit to The University of Technology, Loughborough.

Thanks go to the technical and clerical staff of the Department of Mechanical Engineering for their valuable assistance with this work.

Acknowledgements are also due to NEI Parsons Ltd. for their sponsorship of this work and to the Science and Engineering Research Council for the industrial studentship award and subsequent equipment grant.

Sincere thanks go to my wife Sara for her constant encouragement and for her assistance in the preparation of this thesis for publication and to my parents for their continued moral support.

CONTENTS

NOMENCLATURE

SYMBOL	REPRESENTATION
F_N	Normal polishing force
F_T	Tangential friction force
$\underline{T}$	Link torque vector
$\underline{I}_L$	Link inertia matrix (non-diagonal)
$\underline{I}_A$	Actuator inertia matrix (diagonal)
$\underline{B}$	Matrix of velocity dependent losses including Coriolis and centrifugal effects (non-diagonal)
$\underline{f}$	Vector of dynamic friction at the joints
$\underline{\mu}$	Vector of static friction at the joints
$\underline{g}$	Gravity load vector
$\underline{K}_T$	Diagonal transmission stiffness matrix
$\underline{N}$	Diagonal gearbox reduction matrix
$\underline{T}_L'$	External load vector acting on the links
$\underline{\theta}_A$	Vector of incremental actuator positions
$\underline{\theta}_L$	Vector of incremental link positions
$\underline{\theta}_D$	Vector of desired incremental actuator positions
$\underline{K}_C$	Diagonal matrix of overall servo-actuator gains
$\underline{I}_S$	Nom - diagonal sensor mass matrix
$\underline{K}_S$	Non - diagonal sensor support stiffness matrix
$^T\underline{X}_S$	Cartesian sensor incremental movement (tool frame)
$^T\underline{X}_L$	Link incremental movement transformed to Cartesian tool frame
$^T\underline{F}_{DI}$	Sensor disturbance force vector in Cartesian tool frame
$\underline{J}_M$	Jacobian transformation matrix between joints and tool frame
$\underline{J}_S$	Jacobian transformation matrix between tool and sensor frame

$\underline{J}=\underline{J}_S\ \underline{J}_M$	Jacobian transformation matrix between joints and sensor frame
$\underline{C}$	Compliance selection matrix
$\underline{K}_P$	Diagonal position controller matrix, tool coordinates (position control)
$\underline{K}_V$	Diagonal velocity feedback controller tool coordinates
$T_{\underline{X}_D}$	Vector of desired Cartesian tool incremental movements
$\Theta_{\underline{T}_X}$	Position transformation matrix from joint to local coordinates
$T_{\underline{F}_D}$	Vector of desired Cartesian force demands
$\underline{I}_E$	Non-diagonal environment inertia matrix
$\underline{K}_E$	Non-diagonal environment stiffness matrix
$\underline{I}_{E,S}$	Non-diagonal combined sensor and environment inertia matrix
$\underline{K}_F$	Diagonal force controller matrix
$T_{\underline{X}_E}$	Vector of environment support movements
$\underline{K}_B$	Diagonal matrix of empirical gains for sensor reaction elimination
$\underline{K}_P$	Diagonal matrix of proportional gains (force control)
$\underline{K}_D$	Diagonal matrix of derivative action gains
$\underline{T}_{FRIC}$	Vector of total joint friction torques
$\underline{T}_{STAT}$	Vector of static joint friction torques
$\underline{C}_L$	Diagonal matrix of equivalent viscous damping terms acting at the joints

Single degree of freedom case

M_1	Combined actuator and link mass
M_2	Combined sensor and environment mass
K_S	Sensor stiffness
K_E	Environment stiffness
C_R	Robot actuator viscous damping
C_S	Force sensor viscous damping
F	Controlled actuator force
K_F	Combined force controller gains (derivative + proportional)
K_P	Proportional controller gain
K_D	Derivative controller gain
X_1	Robot mass M_1 displacement
X_2	Sensor mass M_2 displacement
F_M	Measured sensor force
F_S	True sensor force
F_D	Desired force demand

CHAPTER 1

INTRODUCTION

1.1 Background

To date applications of industrial robots have mainly been in tasks requiring basic position control of the end effector. A prominent example is the automotive industry where robots are commonly employed to spot-weld, arc-weld, spray paint, apply adhesive and other coatings, and to perform simple locations. These tasks require accurate end point location of the end effector together with continuous path control. Reactive forces generated between the workpiece and end effector are uncontrollable and in most cases are negligible. For example, in arc-welding the torch must be controlled accurately in position and orientation but no significant forces exist between the torch and the workpiece.

Recently much interest has been focused on the next class of tasks to which robots might profitably be applied. These are tasks in which the robot control system must be capable of accurately positioning and orienting the tool, while at the same time using it to apply a given force/torque pattern to the workpiece. Examples of this type of task are polishing and grinding, fettling and assembly operations.

When human operators perform these tasks they make use of various sensing feedback facilities, notably vision and force sensing. The visual facility is generally used to get the tool to the workpiece or to bring the assembly components together in a gross sense. The force sensing facility is subsequently used to complete the task when the tool

and workpiece or the assembly components have been brought into appropriate contact. Then force feedback is used to control the movement of the contacting objects relative to one another. This simultaneous control of robot movement and force poses a very difficult problem to the system designer and for this reason these jobs have traditionally been done by human operators. However these jobs can also be rather repetitive and unrewarding so there is a good reason for seeking robotic means of doing them.

One important area where the foregoing simultaneous control of position and force in a robot would find immediate application is in polishing.

For about 4 years the Department of Mechanical Engineering has been working with NEI Parsons on the manipulative problems associated with turbine blade polishing.

NEI Parsons is a major manufacturer of steam turbine generators, each of which contains many thousands of turbine blades. Although the various blades are manufactured by a number of different processes, each eventually must be polished in order that its surface finish be acceptable. At present this polishing is performed manually but is fraught with problems. From an operator's view point, polishing is repetitive, strenuous and boring. NEI Parsons required for economic reasons and reasons of quality control, the possibility of automatic polishing (whether by robot or special purpose machine) to be investigated. They are not alone amongst turbine manufacturers in having this need.

1.2 Layout of Thesis

Initial work was concerned with understanding the basic mechanics of the polishing process and analysing the control functions required to be incorporated in an automatic system, this is discussed in chapter 2 and 3. From this evolved the detailed design of the experimental rig described in chapter 6. After a further injection of capital from both NEI Parsons and the SERC the experimental rig was manufactured and commissioned.

Stability of force control was found to be an immediate problem and as a consequence the remainder of this work was devoted to its study. This begins with general analyses in chapters 4 and 5, followed by experimental work in chapter 7 and simulation in chapter 8. Each chapter contains discussion where appropriate, but general discussions are given in chapter 9. Conclusions are given in chapter 10, and recommendations for future work are discussed in chapter 11.

Simplifications and justifications within the force controlled analyses are made with respect to the robot controlled polishing process, encompassing the effects of tool induced disturbances. The tangential polishing friction force is assumed to play no part in the force control analyses, it is totally resisted by the structure of the robot. This force could have an unquantifiable effect on the manipulator's joints frictional components but is unlikely to have a major effect on force control stability. Certainly early polishing results (not quoted) indicate this to be the case.

CHAPTER 2

LITERATURE REVIEW

2.1 Polishing and Related Work

Previous attempts to control tool interactions with the environment have generally been in applications covered by Mortensen in polishing [2], by Spur in brushing [4], by Stute in grinding [1,5] and Kramer and Stepien et al in deburring [3,6]. The approaches can be categorised into two main areas concerning contact force generation. Techniques [1,4,6] used compliance and controlled position errors to generate the appropriate contact forces, whilst [2,3] use servo-controlled forces, although the approach adopted in [3] is somewhat unconventional.

With the compliant approach of [1,4,6] position error demands are generated according to the measured forces; the method does not operate at servo cycle speed and can be classed as only a path correction routine. A limitation with such a technique is that combined compliances of the end effector and robot must obviously be sufficient to generate the required force tolerance within the positioning capability of the robot. Approach [6] differs in that the force generating path corrections are calculated independently of the existing position controller and finally added to the existing position demands at joint level. However, in such compliant situations, tool vibration may be a problem [3,4,7] and also, for the same reasons, loss of position accuracy. In an application of robotic grinding to machining by Asada et al [7], rather than to polishing, the stability of a rotating grinding tool is considered. The industrial manipulator is considered as a passive compliant support to the tool. The analysis shows the desirability in adding passive damping to the tool's compliant

direction.

A better approach is found in servo force control, which does not rely on compliance or the positioning accuracy capability of the robots to achieve the desired contact force. Since this control methodology is radically different from that of compliance in achieving force control, none of the approaches [2,3] uses a conventional industrial robot or controller to generate such contact forces. This has limited such applications to purpose built, geometrically simple manipulator configurations.

The approach in [3] uses an industrial robot to hold the deburring tool, equipped with a two degree of freedom wrist force sensor. However, the forces are actually generated by reacting the workpiece, held on a twin axis force servo-controlled platform, against the tool. Also, since the wrist and robot are compliant, tool induced forces are obviously limited and kept small.

Polishing of plough blades and the grinding of rough castings is described in [2]. Different machines are used in each application. Plough blades are held and orientated by a seemingly converted CNC machine, with the vertical Z axis implementing force control and attached to the belt polishing tool. Hydraulic pressure is used for servo force control. Rigid drives and construction are used to overcome disturbances, but this restricts dexterity and polishing to simple geometries.

The grinding application also described in [2] uses a robot but again with Cartesian axis geometry for simplicity. Force control is implemented on one axis only, the reach axis. Again this limits the application to less complicated geometries.

All of these approaches seem to have limitations whether they be geometric, load limitations or in force control. The work which follows in this thesis attempts to overcome some of these problems by adopting a complete approach to the polishing problem. This has lead to the design and build of a new type of manipulator plus controller and the implementation of new force control scheme.

2.2 Disturbance Control

The polishing process requires both the control of force and the control of disturbances caused by unwanted forces. There are two fundamental methods of disturbance control, active and passive. Active disturbance control methods [9,10,11,12,13,14,15,16,17,18] involve either sensing the disturbance forces or their effects and compensating in the control algorithm or by some external means, to hopefully eliminate the effects. Passive disturbance control presents high resistance or stiffness to the disturbance forces. The passive approach cannot eliminate the effects of disturbance entirely, and its adoption depends on the required degree of immunity to disturbance. Machine tools [2] present a high degree of resistance, careful manipulator design is also effective [19,25] and to a lesser extent impedance control techniques [36-38].

2.2.2 Active Disturbance Control

There are a number of disturbance invariance techniques which use a knowledge of the disturbance as a means to eliminate it. These are notably discussed by Johnson [9], and Davidson [10]. Unfortunately the unpredictable nature of polishing disturbances prevent their adoption.

Burdess/Metcalf [11] proposed a technique for the active control of unknown disturbances. This technique employed a disturbance observer and assumed the system to be linear. Experimentally implemented on a single degree of freedom rig, the system was shown to give good disturbance elimination. Practically however, most manipulative systems are neither dynamically simple or linear.

Hewit/Burdess proposed the technique of active force control (AFC) [12] for disturbance elimination in robot manipulation. This technique requires neither a knowledge of the nature of the disturbance or an exact dynamic model of the manipulator. Basically the actual manipulator system is forced to behave as the decoupled approximate model. Later work, by Hewit et al, describes promising implementations of the technique in both simulation [13,14] and experimentally [15].

R.P. Paul/Wu proposed a later technique, Resolved Motion Force Control, RMFC [15] which appears to be similar in approach to that proposed by Hewit et al. A sensor mass is attached via a force sensor to the terminal link of the manipulator. Firstly, theoretical actuator torques are calculated to drive the sensor mass with the desired accelerations, assuming the manipulator to be massless and with no losses. Actual sensor mass forces are measured and transformed back to the equivalent actuator torques. The differences between the theoretical and transformed actuator torques are assumed to be due to disturbances acting on the manipulator. The theoretical torques are then augmented by the difference. Increasing the theoretical torques is achieved using an incremental technique, force convergent control, FCC. The chief advantage of RMFC technique is that a dynamic model is not required, but as a consequence the technique can be slow to converge.

Techniques by F.W. Paul et al [17] and Hardt/Zalucky [18] differ from

those above by not interfering with the control strategy of the manipulator which remains unchanged and kinematic. Instead external electro-mechanical devices are added to eliminate disturbance effects. F.W. Paul et al attempted to remove the random forces of a chipping hammer and prevent shock from being transmitted to the robot. This was achieved by isolating the chipping hammer from the robot with a device to counteract the chipping hammer forces. The device monitored the disturbance of the chipping hammer forces. Opposing forces were generated internally, created from inertial means by a controlled acceleration of a mass. These inertial forces were reacted against the chipping hammer disturbance. Since the isolation system could be situated near to the disturbance, inertias were low and the bandwidth of the system high. In the experimental single degree of freedom case, most of the shock load from the chipping hammer was removed, leaving only the steady state loading to be taken by the robot structure. Arguably the mechanical life of the robot may be extended. The chief disadvantages of the system are the mechanical complexity and weight. Such a system employed for multi-degree of freedom invariance is clearly impractical.

Zalucky and Hardt considered the effects of external disturbance forces on a manipulator structure to deflect in bending. It was proposed to have within each limb of a manipulator an optical deflection sensor and an actuator assembly able to react internally. Thus the system restores the limb to its undeflected shape. A single beam system was implemented and produced good deflection compensation with a 10Hz bandwidth. The disadvantages of such a system are the increased mechanical and instrumentation complexities and weight. Arguably it may be more economical to use the extra weight incurred in the design of stiffer linkages and avoid the added complexity.

All these techniques of invariance control above have limitations in the form of bandwidth, complexity, economics or practicality. A passive approach to disturbance rejection would certainly go some way towards fulfilling these objectives.

2.2.3 Passive Disturbance Rejection

Passive disturbance rejection is the simplest of all techniques and is achieved by new manipulator designs presenting high stiffness to the disturbance. These high stiffness manipulators use parallel structures as opposed to the more usual open chain link construction. A particular adaptation of parallel structures for support has been proposed by Asada/West [19] where local task, supports have been used to improve the stiffness of open link chain manipulators. In applications where high disturbance forces are experienced with a particular task and manipulator configuration then, by some mechanical means and careful design, it may be possible to remove these troublesome degrees of freedom by parallel support to rigid earth. Fundamental disadvantages with this approach are, that the support is of fixed design for a particular task, and that for manipulator configurations and outside these bounds of support over constraining may in fact impede control.

The parallel structured manipulator easily overcomes these limitations by being completely adaptable. Earliest origins of parallel drive manipulators are found in the Stewart Platform [20] 1965 conceived to enable an aircraft simulator to operate uninhibited in 6 degrees of freedom. The terminal device, in this case a massive platform, has its motion completely controlled by attachment to six independent actuators, each reacting against a rigid foundation. This idea was developed further by GEC Marconi Research 1983, in a robot named the Gadfly [21]

which is basically an inverted Stewart Platform. The hydraulic actuators of the Stewart Platform were replaced by electrically driven screws, each with individual microprocessor control. High stiffnesses were achieved by the Gadfly, but the nature of the platform produced poor articulation, and at the extremes of angular articulation, the stiffness suffered greatly. This condition of poor angular stiffness was recognized and examined by Stewart in his original work. Further kinematic analyses of platform type manipulators was analysed in depth by Hunt [22] in 1983 and by Yand/Lee [23] in 1984, who also assessed the feasibility of platform type manipulators for general use. They conclude that the inverse kinematics is more solvable than for an equivalent link type robot, but that workspace and manoeuvrability are comparatively restricted.

More recently Fichter [24] 1986, in comprehensive work, considered the kinematics of the general Stewart Platform. Later he considers the kinematics design and implementation of a six degree of freedom Gadfly type platform manipulator. Rathbun/Dunlop [28] 1987, attempted to implement a Stewart platform as a cheap 6 axis N.C. milling machine. Using stepper motor drives on each axis, they found the working envelope comparatively restricted, and free play in the transmissions was explained by the shortcomings in the machining accuracy of the device.

Encouraged by the potential of the parallel drive construction for high loads and disturbance rejection, the Author in 1984, combined the benefits of serial link articulation with a parallel drive scheme [25] 1987. The resulting manipulator is planar with three degrees of freedom, twin parallel drives and a serial wrist, described comprehensively within this thesis. Its significant features are its high load capacity combined with articulation and the elimination of the kinematic singularity experienced by the Stewart platform.

In parallel with this work, other researches, namely GEC Research [26] 1986 and Rooke/Lewis [27] 1986 have also created similar devices of combined serial and parallel transmissions. GEC's implementation, called the Tetrabot, uses three parallel axes connected at an apex and responsible for gross motion. Articulation is achieved by the addition of three serial links at the common apex. The machine is reputed to have a high accuracy capability. Rooke/Lewis's implementation involved using twin ballscrew drives, in a similar triangular arrangement to that of the author but within the mechanism of a five axis robot, known as the Lamb robot. The parallel drives are arranged to provide the reach and lift of the robot, thus a high load and accuracy capability are displayed.

2.3 Force Control

2.3.1. Introduction

Force control is a relatively recent area to be addressed by robot control researchers. Prior to 1980 little work, with the few notable exceptions, had been performed in this area. These early works were the first attempts to control robot movement and force simultaneously. Inoue [57], in 1974, investigated the force feedback to monitor and control assembly tasks by using the data as a 'go no-go' signal. In 1976 Paul and Shimano [29] proposed a relatively crude means of robot force control, while Whitney [30], in 1977, devised a scheme based on velocity demands using his resolved motion rate control scheme (RMRC) [31]. The approach proposed by Paul and Shimano can be can be described as being logic based. The control switches certain joints to force control dependent on cartesian demands, while the remaining joints continue to be position controlled. Owing to the joint based nature of this

control, there were severe coupling problems. Whitney avoided these problems by arranging his existing RMRC system to include additional demands of force control. Force measurement was performed, in this case, by a wrist mounted force sensor. The technique was significant in two respects: force demands could be specified in Cartesian coordinates, and that force control could be achieved as an outer loop on top of an essentially velocity controlled inner loop.

Mason [32,33] in 1979, proposed a formalisation of general manipulation tasks, and can be considered to be the turning point of robot force control theory. He proposed the `C' coordinate system, or the `local' coordinate system, aligned to the task. For successful environment manipulation, without jamming, the appropriate degree of freedom within the local coordinate system should be switched in and out of compliant control.

Immediately following this formalisation two implementations for decoupled compliant control, in local coordinates, were proposed: the Hybrid position/force control scheme by Raibert and Craig [34,35] and impedance control techniques [36,37], including active stiffness control proposed by Salisbury [38]. Since this initial work there has been a vast increase in activity within this research area. Clearly force control is to be an essential requirement of the next generation of industrial robot.

Several reviews of practices within robot force control have been written, notably by Asada [39], Whitney [40], and Maples and Becker [41].

The following two sections are devoted to these two methods of force control, impedance and Hybrid control.

2.3.2 Impedance Control

Impedance control, as the name suggests, does not specify force at the end effector but rather the relationship between force and position, that is the `impedance' or mechanical stiffness of the manipulator's end point.

The assembly of closely mating parts by robotic means is an extremely difficult task. Small errors in tool/part mating, whether caused by positional inaccuracies of the manipulator or component machining errors, can result in high local contact forces, which inevitably leads to the jamming of the component. Various techniques have been developed in an attempt to control, or limit, theses unwanted jamming forces. Initially these techniques took the form of devices added to the manipulator's end effector.

Such schemes by Takeyasu et al [42], and Drake [43] add a compliant element to the manipulator end. These devices can accommodate the mismatches, by simple passive movement, and thus avoiding the unwanted jamming forces. In [42] successful assembly could be guaranteed by the manipulator performing blind, a set sequence of fine movements. A progression from this work was to the passive device, the Remote Centre Compliance (RCC) wrist, patented by Watson in 1978 [44] and investigated further by Whitney and Nevins [45,46] and McCallion [47]. It attempted to exploit the principle that when part mating, `pull' is better than `push'. The device decouples, by mechanical means, the compliances of the various degrees of freedom, at a point remote from the device. This point is usually arranged to be at the surface of a grasped component. The device was particularly effective in limiting the jamming torques of

the peg in hole operation. It thus avoids the jamming forces and torques. The RCC device has two major limitations, that of presenting constant stiffness and fixed RCC point. In efforts to overcome these disadvantages, various actively compliant wrists have been devised. One device by Van Brussel [48] has five Cartesian servoed axes and attaches to the end effector. Another device by Cutkosky and Wright [49], an instrumented RCC wrist (IRCC), is less bulky than that in [48]. Both devices have the ability to vary their compliance and remote centres of compliance under software control.

Combining the benefits of active compliance and set movement patterns to assist in assembly a device by Hollis [50] has been built. This device is held by the manipulator and has two axes controlled in movement and active compliance.

All the devices noted use the manipulator as a rather crude positioning platform which plays little part in the force and position control of the assembly task. The active stiffness approach, first proposed by Salisbury [38], attempts to combine both the compliance control and robot movement into a single robot controller eliminating the need for such add-on devices.

Active Stiffness Control

In keeping with Mason's requirements [23] for successful task manipulation, Salisbury proposed a control scheme which would present controlled compliances to the work piece in the local Cartesian coordinate system. Those coordinates required for positioning are controlled to have high impedance, while the remaining force controlled coordinates are controlled to have high compliance. The technique operates by transforming the desired local Cartesian stiffness matrix to

their servo joint components, hence the servo gains are determined. Force control is also implemented on the compliant coordinates. This is achieved by transforming to joint coordinates, the forces measured by the end-effector force sensor. Compensation error torques are generated in joint coordinates and added into the servo position demands. There are a number of serious limitations with the active stiffness. Since the cartesian stiffness matrix is transformed using the Jacobian, control can be lost at singularities [51]. Also the effects of friction in the transmissions can have cogging effects [41]. This latter limitation is investigated further by Salisbury [52]. Later work by Hogan [53] proposes a methodical approach for selecting manipulator end-point impedance, while taking into account the demands of the impending manipulation task. Hogan continues this work in [56]. In relatively recent work by Sheridan et al [54], the stability of active impedance control is investigated, chiefly in the frequency domain. The work covers the conditions, with and without dynamic force control.

2.3.3 Hybrid Force/Position Control

Hybrid control, initially proposed by Raibert and Craig [34], is similar in result to Salisbury's active stiffness approach. Both techniques should theoretically be able to control force and position in local Cartesian coordinates. Unlike the stiffness selection matrix of active stiffness control which operates in joint coordinates, the compliance selection matrix of Hybrid control operates in local Cartesian coordinates. The compliance selection matrix is used to switch each coordinate to force or position control depending on task. There are no intermediate states, unlike the approach adopted in Salisbury's work. In the active stiffness scheme, control of the switching is less rigorous and it is possible to specify both a positional stiffness on a coordinate as well as force demand. For large compliant movements this

could result in the position control fighting the demands of the force control loop. The effect would be rather like a weight hanging on a spring, an equilibrium position would be established.

In the original Hybrid scheme [33] force and position error demands are computed in cartesian coordinates before the compliance selection matrix and transformation to joint coordinates. With active stiffness control, error demands are computed in joint coordinates. A number of variants of the original Hybrid scheme have been developed, most to reduce computational burden [25, 51, 55].

Hybrid control is increasingly being proposed for use in a greater diversity of application, for example: West and Asada [19] propose it for a multiple contact manipulation problem, Tsai and Orin [55] have installed it to control the stepping force of a leg for a legged vehicle. Uchiyama et al [60] is developing a scheme for the force/position coordination of a two arm robot, and ourselves [25] in this work.

The original Hybrid scheme [34] was not without its problems. A number of these problems have still not been surmounted to date. Typically these problems are in: surface detection, switching between modes, friction effects, and force control stability and response. These and other problems are discussed by Paul in [59]. But by far the greatest problem is that of force control and as a consequence, force response. This problem is considered in detail within this thesis.

Force Control

Raibert, after his initial work, did not pursue further the problem he experienced with force control. In 1985, Roberts, Paul and Hillberry

[62] attempted to investigate the effects of sensor stiffness on force control and response. Single degree of freedom force control was experimentally investigated, with the conclusion that a soft force sensor allows for higher gains and thus improved response. Environment dynamics were not considered. Fukuda et al [63], in 1986, in an experiment involving a force controlled gripper, discovered that the mechanical properties of the component being gripped affected force response of the grip. Khatib and Burdick [64] also recognized that a force controller should adapt to changing environment dynamics and be able to overcome the static friction components in the manipulator joints. However, their model of the sensor has been shown, by work within this thesis, to be inadequate. At the other extreme, Seering and Eppinger [65] have performed a range of simulations modelling all possible elements in the dynamic chain, including environment, sensor and grip. The analysis is superficial and consequently has errors, particularly with respect to the roll of the sensor damping, as is later shown in this thesis.

Chae and Hallerbach [66] have shown that, by implementing identical force controllers on various manipulators of differing geometry, force control stability is also kinematically dependent.

Recent Force Control Techniques

In efforts to improve force response, joint friction disturbance rejection and force control stability, Maples and Becker [41], proposed a new force control technique. The system uses a position controlled manipulator, taking its demands from an outer force controlled loop. A compliant force sensor provides the measured force to generate force error. From a model of the force sensor, the force error is converted into position demand increments which are passed to the position

controller. The system is claimed to work well with good disturbance rejection. However the problem with such schemes is that, since force response is position measurement dependent, if a stiff sensor is used, movement ceases, and force control is lost.

Another scheme proposed in 1987 by Chae and Hollerbach [67] uses an open loop model of the manipulator, and local torque control loops [61] around each joint. Low-pass filtered force feedback is provided from a wrist mounted sensor to improve steady state error. The system works, but as a consequence of the low-pass filtering has an extremely low bandwidth of less than 1Hz.

CHAPTER 3

POLISHING PROBLEM ANALYSIS

3.1 Polishing Task

The polishing operation is distinct from grinding as a machining process, in that only surface finish is altered. Dimensions, assumed initially correct within tolerance, remain unchanged. An important area complementing polishing is that of automatic surface texture inspection, which is, as yet, unaddressed. Currently this is carried out manually but eventually an automatic means must be sought.

At present high stock removal rates are achieved using belt grinding machines which are difficult for manual operators to use over large surface areas due to the large forces involved. Therefore such large highly efficient belt machines are used with automatic polishing machines of restricted freedom; they can polish only with a single grade of belt for each blade setting, and are restricted to only simple geometries.

Manual operations are presently confined to the polishing of smaller blades and blades with more intricate geometry, using compliant 'bobs'. These bobs have a much lower stock removal rate and a shorter useful life than the belt equivalents. Since the capacity of abrasive belts is well documented, it should be possible for any proposed robot carrying out the blade polishing to use belt polishing techniques. A variety of belt types is available from coarse to fine and from narrow to wide, so maximum possible benefit should be obtained from robot usage.

To make full use of belt polishing, a production facility would involve the robot holding the blade and offering it to any of a number of polishing machines, depending on the current stage of the polishing task. However in this research, the problem is simplified from a computational point of view by the following approaches.

(i) Reversing the problem, mounting the blade stationary and reducing the abrading medium to a rotating polishing wheel held by a robot and applied to the blade.

(ii) Restricting the manipulator to polishing a two dimensional contour, such as a turbine blade of constant section.

A relatively simple planar robot is therefore required which has a minimum of 3 degrees of freedom.

For the purpose of developing a control strategy which enables robots to polish surfaces of varying contour, the analogy between the control strategy developed for the 3 degree of freedom rig and that for the production prototype of 6 degree of freedom, is exact. The only major difference between the approaches is the increase in computational burden owing to the extra complexity.

3.2 Defining the Problem

3.2.1 Polishing

The polishing task involves the relative movement of a spinning abraiding wheel over a workpiece surface. Given that all other parameters remain constant, including grit, wheel composition, spin speed, work piece surface properties and compliance, then the surface finish can be controlled completely by velocity of tangential travel V_T and normal contact force F_N, see figure 3.1.

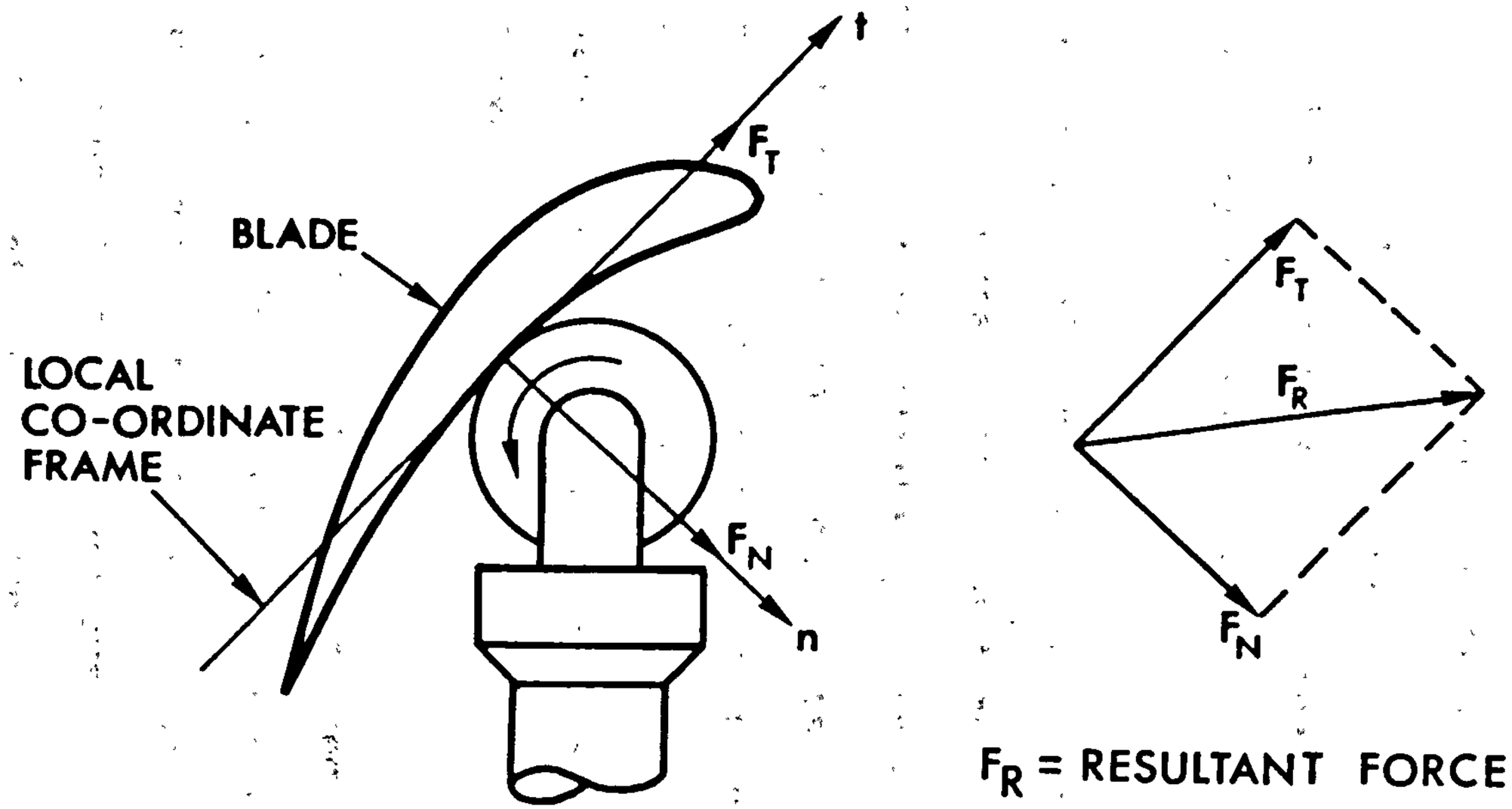


Figure 3.1. Shows the polishing tool forces of the planar polishing arrangement.

The formalisation proposed by Mason [32] lends itself directly to the polishing task. The working coordinate system of the polishing process is aligned to the local coordinates of the blade profile at the point of contact. The natural position constraint is normal to the surface but freedom to move exists in all other five degrees of freedom. Conversely, force control is unconstrained along the normal but constrained in all other degrees of freedom where friction can be neglected. However, in a real case, surfaces deflect as forces are applied and frictional forces exist in tangential directions which must be overcome before movement can begin.

These two types of natural constraints on force and position partition the task into two orthogonal control systems, one dealing with position the other force. The control strategy, dependent on the polishing task, operates within the natural constraints and imposes artificial constraints on the normal force coordinate and on the position coordinates of the remaining degrees of freedom. Noting that, artificial constraints can only exist on their respectively naturally free orthogonal axes.

Having established the fundamental control requirement of a manipulative task, that of independently controlling force and position on orthogonal local coordinates, consideration can now be given to the merits of different approaches likely to achieve this goal. In addition, complications such as tool induced friction forces should be encompassed in these approaches.

3.2.2 Disturbance Forces

The principal problem of simultaneously controlling the polishing force normal to the surface while attempting to control tangential movement is grossly aggravated by large unwanted friction forces. These are caused by the moving abraiding medium travelling along the surface tangent. Rather than attempt to control deflections by superimposing sophisticated invariance techniques, as discussed in chapter 2, on top of the force/position control problem, a new design of robot has been developed, and is extensively described in chapter 6. The design provides high force and high rigidity of the positioning structure and the necessary orientation of the end effector.

3.3 Force sensor compliance

A force sensing wrist; rather than joint torque sensing, is used to measure end effector forces. In this way the measurement problems caused by the inertial and dynamic effects of the robot structure itself are minimised.

In common with the Hybrid control [33] and Impedance control [36-38] approaches, a compliant wrist force sensor is used. The compliance effectively reduces peak forces caused by unforeseen irregularities or contacts which tend to be high frequency in nature and well above the bandwidth of the force controller . The obvious disadvantage with this approach is the loss of positional accuracy under disturbing forces. Hence there have been moves to avoid passive compliance and introduce active compliance into the robot control itself. This allows the compliance or the impedance of the robot to be programmable so enabling it to be varied to suit the task at hand. Major drawbacks with this approach are seen by consideration of the manipulators output

impedance [36] to a disturbance force, F_D.

$$\underline{F}_D = \underline{I}.\underline{\ddot{x}} + \underline{B}.\underline{\dot{x}} + \underline{K}.d\underline{x} + \underline{f} + \underline{v}$$

Where: $\underline{\ddot{x}},\underline{\dot{x}}$ are the absolute acceleration and velocity vectors,

$d\underline{x}$ small movements about fixed linearized position,

$\underline{I},\underline{B},\underline{K}$ output inertia, viscous, stiffness matrices,

$\underline{f},\underline{v}$ dynamic and static friction components.

This shows there are uncontrollable impedance components of inertia and friction in addition to controllable viscous and stiffness components. When dealing with low amplitude forces the static friction in the transmissions becomes significant, restricting end effector compliance. Whereas high frequency disturbances tend to be resisted by the inertial components of the manipulator. These two factors indicate why a compliant wrist is desirable in manipulative tasks, especially when unforeseen contact forces arise. A compliant wrist presents a constant stiffness to the disturbance, which is independent of manipulator configuration or control.

3.4 Possible Problem Approaches

There are three feasible approaches to the problem, as indicated in figure 3.2. Compliant force sensors are used in all approaches. The use of strain measurement makes steady state force measurement possible. In all cases (a), (b) and (c) of figure 3.2 the force sensor is integral with the wrist structure.

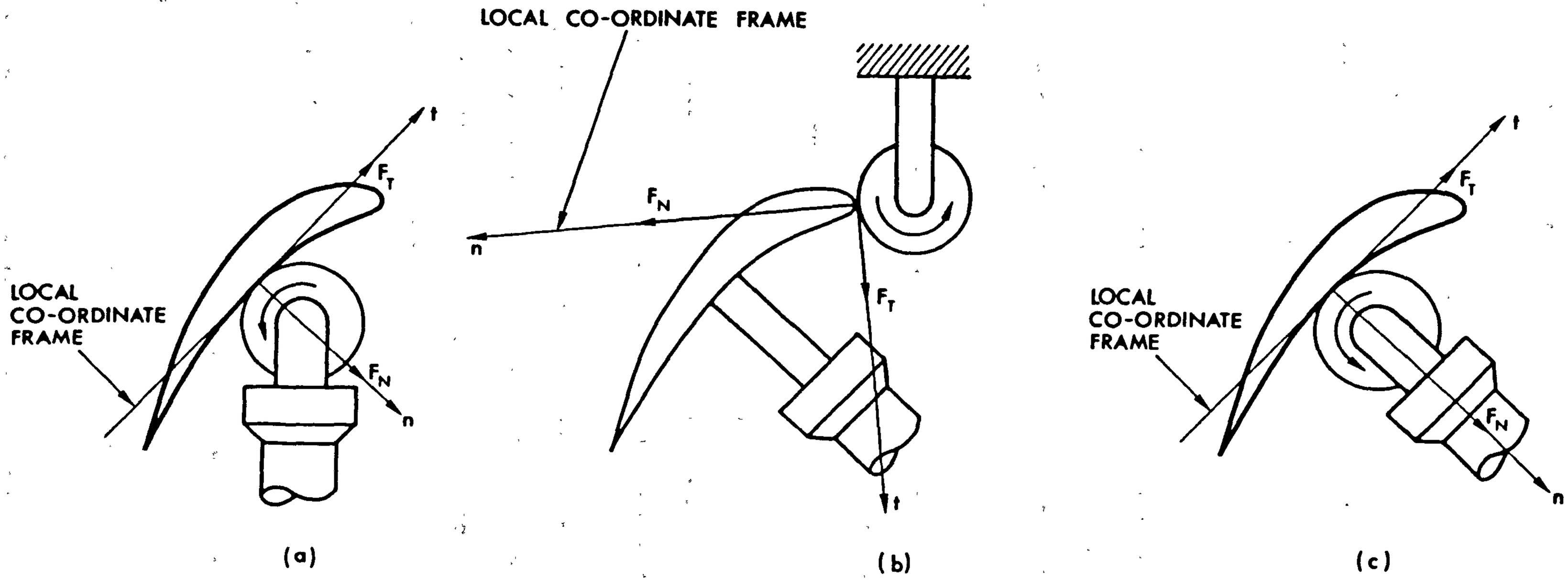


Figure 3.3. Three possible methods for robot force controlled polishing.

3.4.1 Wrist Orientated for Access

This approach is illustrated in figure 3.2(a). Unfortunately the random nature of the tangential force F_T has an unwanted effect on the wrist force sensor compliance. This causes a loss of both positional accuracy and knowledge of current task geometry.

The approaches only virtue is the technique of orientating the tool for easy access to the blade's surface. But the increases in computational and mechanical complexities, together with unwanted force sensor deflections, have precluded its adoption. In summary its disadvantages are as follows.

(i) A compliant multi-degree of freedom force sensor is required, with the associated computational burden of transforming force between wrist and local coordinates.

(ii) The force sensor measures the largely dynamic random tangential forces. This would require some form of dynamic modelling to eliminate F_T and obtain F_N.

(iii) As the contour is polished so the point of contact moves around the circumference of the polishing wheel. This necessitates the continual updating of the end effector to tool transformation.

3.4.2 Robot held blade

The approach illustrated in figure 3.2(b) considers the originally perceived idea, taken from the manual operator, of holding the blade and applying it to a stationary abraiding machine.

This technique has the greatest potential for utilizing belt polishing

machines, and hence achieving the highest polishing efficiency, providing some fundamental problems can be initially overcome. The problems are similar to those quoted in the example above, section 3.4.1, but are computationally more complex. In section 3.4.1 above, where the polishing wheel is held, the continual re-calculation of the tool to force sensor transformations is reasonably straight forward since the geometry of the wheel is analytically simple to handle. Conversely, performing the same calculations with an aerofoil shape whose geometry is numerically based would require a great deal of interpolation and file handling capacity. This argument is relatively academic since the afore mentioned disadvantages prevent its adoption at this stage.

3.4.3 Wrist Orientated Normal to Surface: The Adopted Approach

Figure 3.2(c) illustrates this approach. The force sensor, integral with the robot wrist, has a single axial degree of freedom, giving wrist compliance, and a force sensing capability gained from displacement measurement via a strain gauge. The remaining degrees of freedom are constrained by an axial bearing displaying low friction characteristics. Details of the sensor design are given in section 6.5.5.

Using this simplified form of sensor imposes the additional positional constraint of maintaining the sensor normal to the blade surface for measuring the normal contact force. The advantages of such a simple approach are given as follows.

a) Aligning the sensor to the local coordinate frame of the surface eliminates the need to transform force between wrist and local coordinates, so reducing computational burden.

(b) The frictional force F_T is decoupled mechanically at source from

the normal force F_N. F_T is restrained by the rigid force sensor and robot structure.

(c) Constraining the point of wheel contact, to remain in line with the force sensing axis, reduces the problems associated with predicting contact geometry.

With this latter approach finally adopted, the experimental manipulator was designed and constructed.

CHAPTER 4

DYNAMIC EQUATIONS OF COMPLIANCE BASED FORCE CONTROL

4.1 Introduction

Force control strategies are in general based on kinematic rather than dynamic considerations. These control techniques, encompassed by Hybrid [34] and Impedance control [38], deal with conventional types of robot which incorporate gear transmission drives. To use direct drive arms, involving rigid links without flexible transmissions, would tend to simplify the control problem, but the robots would be weak with limited application.

Dynamic complications increase with the increasing system order. Two orders are added for each additional flexibility. For example in a single degree of freedom case involving a dynamic train composed of - a flexible actuator transmission, a compliant force sensor and/or a compliant but massive environment, would result in a sixth order dynamic system. The problem is compounded in the multi-degree of freedom situation where there could also be severe dynamic coupling between the manipulator's degrees of freedom.

The Hybrid control technique involves the simultaneous control of selected degrees of freedom under force control, while the rest remain under positional control. Naturally if the motion is high speed there will be unwanted dynamic effects coupling with the force controlled degrees of freedom, this will affect force control accuracy. However it is reasonable to assume that, providing the force control causes minimal accelerations in the manipulator, the dynamic coupling between all the

degrees of freedom will be minimal and may be neglected. The exceptions to this are the large accelerations caused by the manipulator attempting to: seek a surface under high force demand, or apply large forces with an extremely compliant environment or wrist force sensor.

Hybrid control strategy is essentially static and kinematic, but the criteria for determining stability of both the force and position control loops are governed by their dynamic properties. Provided the speed of the manipulator remains slow, dynamic coupling between the degrees of freedom may be ignored, this allows independent analysis of position, and particularly the force controlled loops, to be carried out. The purpose of this chapter is to establish the dynamic equations for the manipulator and to use them to study particular properties of compliance based force control.

4.2 The Dynamic Equations of a General Manipulator

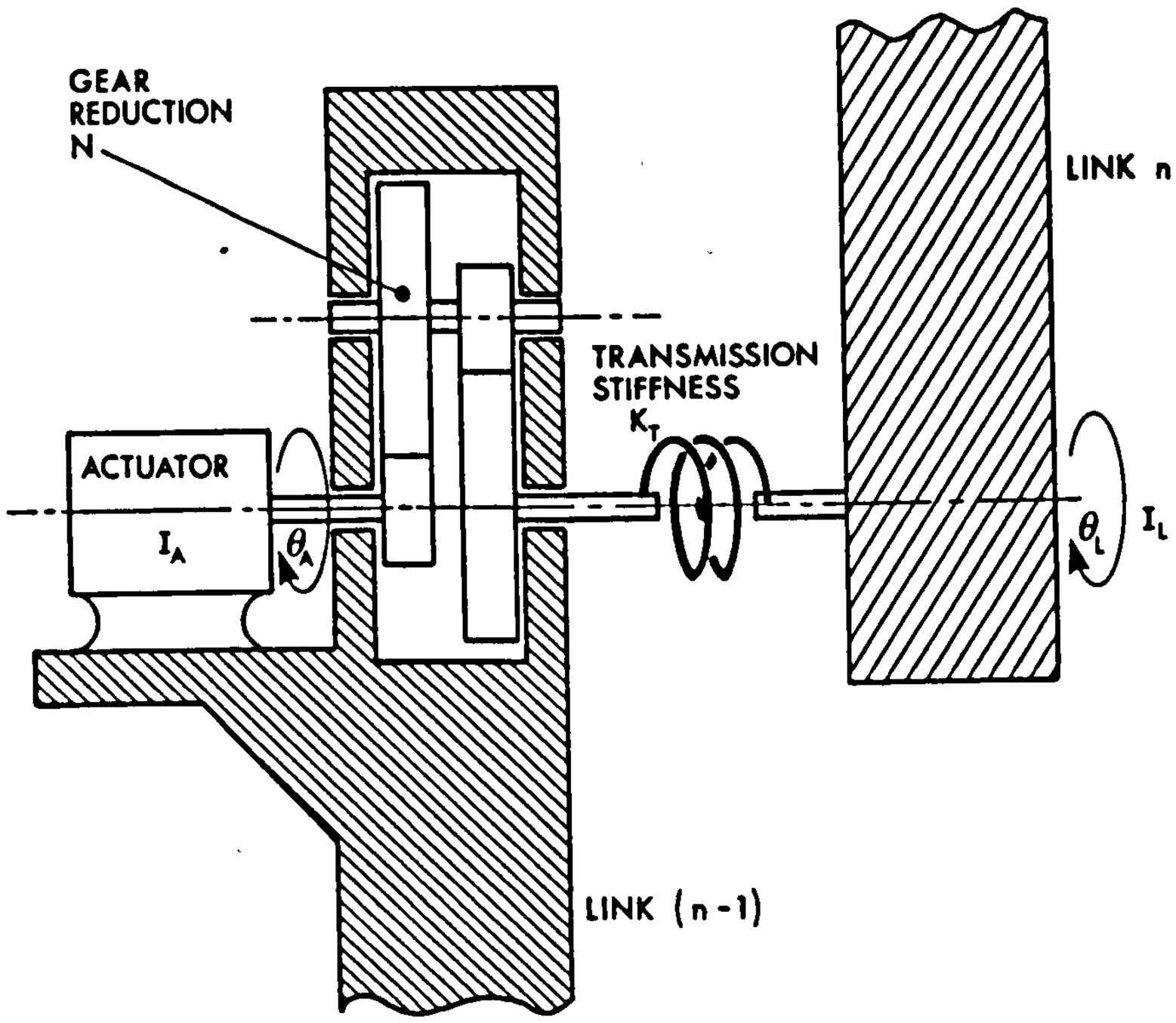
The dynamic equations of motion for passive force control are established firstly by considering the individual dynamic sub-systems for link, actuator and sensor. These are subsequently compounded to give the full dynamic equations shown in section 4.2.4.

4.2.1 Link Dynamics

Benatti et al [36,37] give the linearized equations of link motion of a typical six axes manipulator for small deflections as,

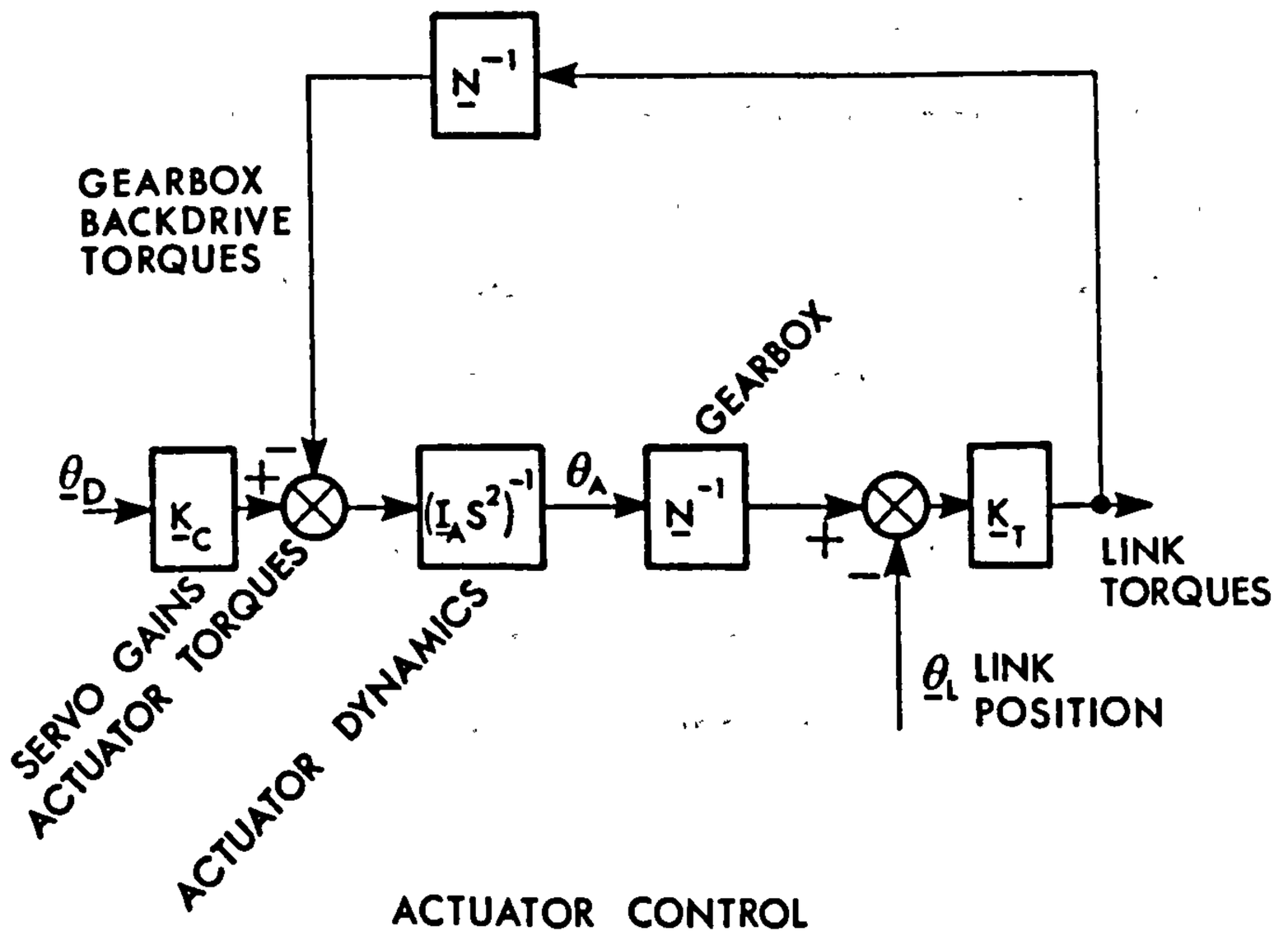
$$\underline{T} = \underline{I}_L.S^2\,\underline{\theta} + \underline{B}.S\,\underline{\theta}_L + \underline{f} + \underline{\mu} + \underline{g} \qquad (4.1)$$

This assumes that the links are rigid and the actuator dynamics negligible. Here $\underline{T}$ is the vector of torques acting on the manipulator links, $\underline{I}_L$ is the link's non-diagonal inertial matrix, $\underline{B}$ is the



TRANSMISSION MODEL

FIGURE 4.1



ACTUATOR CONTROL

FIGURE 4.2

non-diagonal matrix representing velocity dependent terms, $\underline{f}$ and $\underline{\mu}$ represent vectors of dynamic and static friction respectively, $\underline{g}$ is the gravity load vector and 'S' is the Laplace variable. In general all these parameters are configuration dependent indicating non-linear nature.

The manipulator model given by equation 4.1 takes no account of the inherent joint flexibilities typical of transmission systems. A more realistic model derives the joint torques from deformations of the joint transmission, see figure 4.1, giving:

$$\underline{T} = \underline{K}_T(\underline{N}^{-1}\ \underline{\theta}_A - \underline{\theta}_L) - \underline{T}_L' \qquad (4.2)$$

where $\underline{K}_T$ is the diagonal drive transmission stiffness, $\underline{N}$ the matrix of mechanical reduction of the respective transmissions and $\underline{T}_L'$ the external load torques vector acting on the individual links. $\underline{T}_L'$ is generally zero for a manipulator operating in positional control mode where the payload is negligible. The vector of incremental actuator positions is $\underline{\theta}_A$, which is measured from a mean position, as is $\underline{\theta}_L$, the vector of incremental link positions. Assuming that gravity loads can be removed by feed-forward compensation, and that velocity dependent terms, $\underline{f}$ and $\underline{B}$, for a slow moving manipulator, may be neglected. Also static friction, $\underline{\mu}$, (both stiction and Coulombic) which in practice is non-linear and difficult to model analytically, can be neglected where transmitted forces tend to be relatively large.

In the case where a force sensor is fitted $\underline{T}_L'$ represents the reactions of the manipulator/sensor system transformed to joint coordinates. Combining 4.1 and 4.2 gives the equations of motion for the link system as

$$\underline{I}_L.S^2\ \underline{\theta}_L = \underline{K}_T(\underline{N}^{-1}\ \underline{\theta}_A - \underline{\theta}_L) - \underline{T}_L' \qquad (4.3)$$

or

$$(\underline{I}_L.S^2 + \underline{K}_T)\underline{\theta}_L = \underline{K}_T\ \underline{N}^{-1}\ \underline{\theta}_A - \underline{T}_L' \qquad (4.4)$$

The left hand side of equation 4.4 represents the characteristic matrix of the link dynamics, whose eigenvalues are its roots.

4.2.2 Actuator Dynamics

In the control scheme adopted here, utilising the Hybrid technique, no local actuator feedback is provided, see figure 4.2.

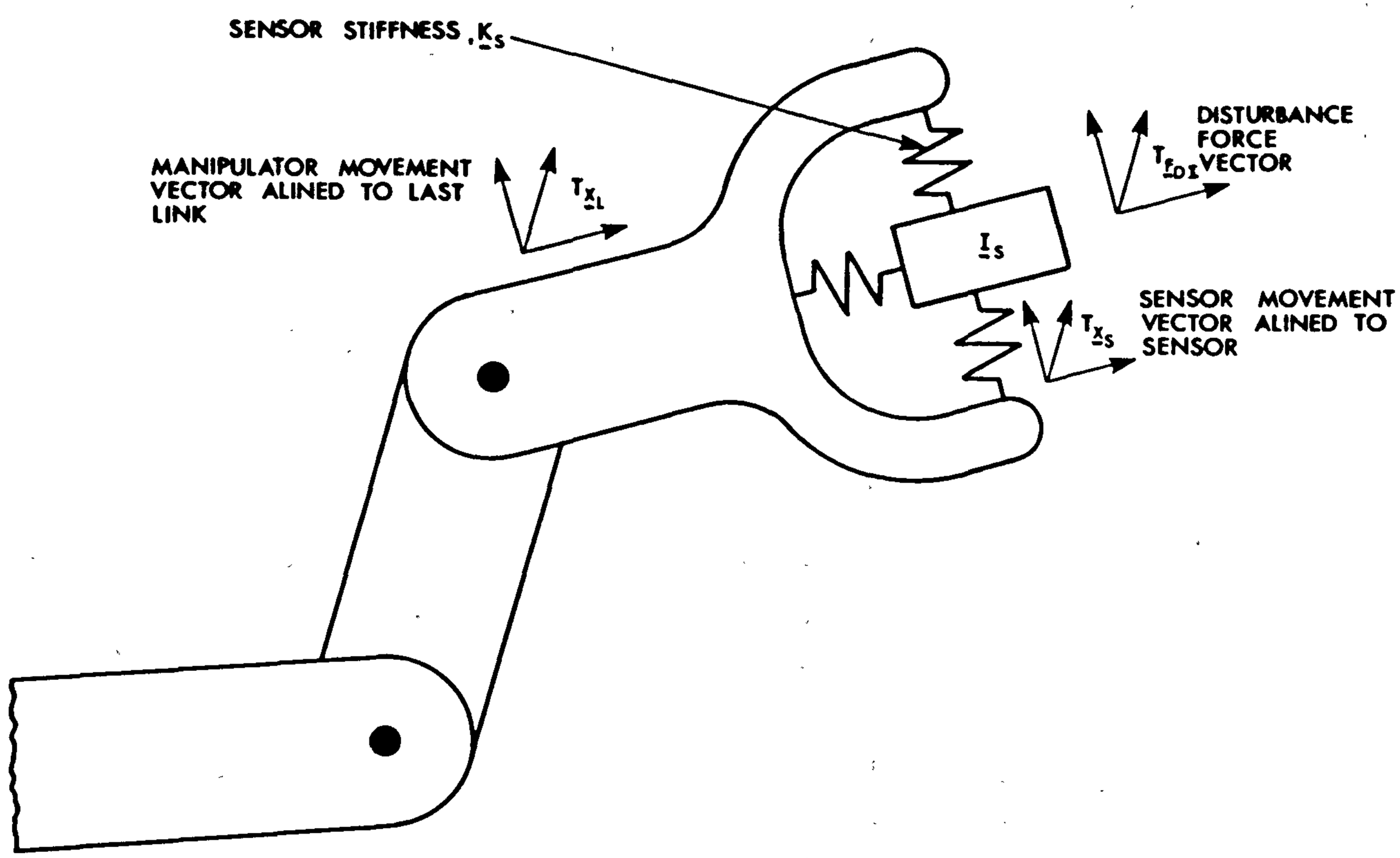
Note that the natural static and dynamic friction of the actuators have been ignored. Since the manipulator is to be slow moving the dynamic friction effects will be small helping to simplify the analysis. Static friction, if significant, will act as a disturbance to the system without affecting stability. Rate feedback is applied to the overall control loop to impart stability. Figure 4.2 gives the dynamic equation for actuator control as:

$$\underline{K}_C\ \underline{\theta}_D - \underline{N}^{-1}\ \underline{K}_T(\underline{N}^{-1}\ \underline{\theta}_A - \underline{\theta}_L) = \underline{I}_A.s^2\ \underline{\theta}_A$$

or
$$\underline{K}_C\ \underline{\theta}_D + \underline{N}^{-1}\ \underline{K}_T\ \underline{\theta}_L = [\underline{I}_A.s^2 + (\underline{N}^{-1})^2\ \underline{K}_T]\underline{\theta}_A \qquad (4.5)$$

where $\underline{\theta}_D$ is the desired incremental actuator position vector, produced by the controller as an input to the servo amplifier. The servo amplifier and associated gains are modelled as a diagonal matrix $\underline{K}_C$. $\underline{I}_A$ is the diagonal actuator inertia matrix.

Equation 4.5 shows the characteristic matrix to be diagonal with roots dependent on the gearing ratio, $\underline{N}$. Increasing $\underline{N}$ tends to reduce the effect $\underline{K}_T$ has on actuator dynamics, effectively isolating the actuators from the manipulator.



ROBOT/SENSOR ARRANGEMENT AND COORDINATE SYSTEMS

FIGURE 4.3

4.2.3 Wrist Sensor Dynamics

Consider the diagrammatic arrangement of a general compliant wrist force sensor, see figure 4.3.

The dynamics of the sensor can be considered to be represented by a non - diagonal inertial mass matrix $\underline{I}_S$, suspended by a spring of non - diagonal stiffness matrix, $\underline{K}_S$. Sensor movement is described by the vector ${}^T\underline{X}_S$, where superscript 'T' indicates the coordinate frame. Sensors in this case are aligned to the tool coordinate frame. The '$\underline{X}$' indicates the motion is Cartesian within the tool coordinate frame. Subscript 'S' indicates the sensor is the component being referred to. As with the previous two cases, the sensor dynamics are determined with respect to its natural coordinate system. Also since deflections of $\underline{K}_S$ are relatively small, the system may be assumed to be linear. The dynamics of the sensor can be written,

$$\underline{K}_S({}^T\underline{X}_L - {}^T\underline{X}_S) + {}^T\underline{F}_{DI} = \underline{I}_S\, s^2\, {}^T\underline{X}_S$$

where ${}^T\underline{X}_L$ is the vector of manipulator link motion resolved into tool Cartesian coordinates acting at the manipulator end, hence superscript 'T'. ${}^T\underline{F}_{DI}$ is the vector of external disturbance forces acting on the sensor in the tool coordinate system, rewritten as

$$\underline{K}_S\, {}^T\underline{X}_L + {}^T\underline{F}_{DI} = (\underline{I}_S.s^2 + \underline{K}_S){}^T\underline{X}_S \qquad (4.6)$$

The transformation between the link joint incremental movement, vector $\underline{\theta}_L$, and movements in the manipulator Cartesian frame, is the Jacobian $\underline{J}_M$ for small movements or velocities. $\underline{J}_S$ is the subsequent transformation from manipulator end to sensor end movement. This gives:

$$\underline{K}_S\, \underline{J}_S\, \underline{J}_M\, \underline{\theta}_L - {}^T\underline{F}_{DI} = (\underline{I}_S.s^2 + \underline{K}_S){}^T\underline{X}_S$$

since the manipulator terminal link incorporates an integral force sensor, the matrix $\underline{J}_S$ becomes the identity and $\underline{J}_M$ is subsequently referred to as $\underline{J}$.

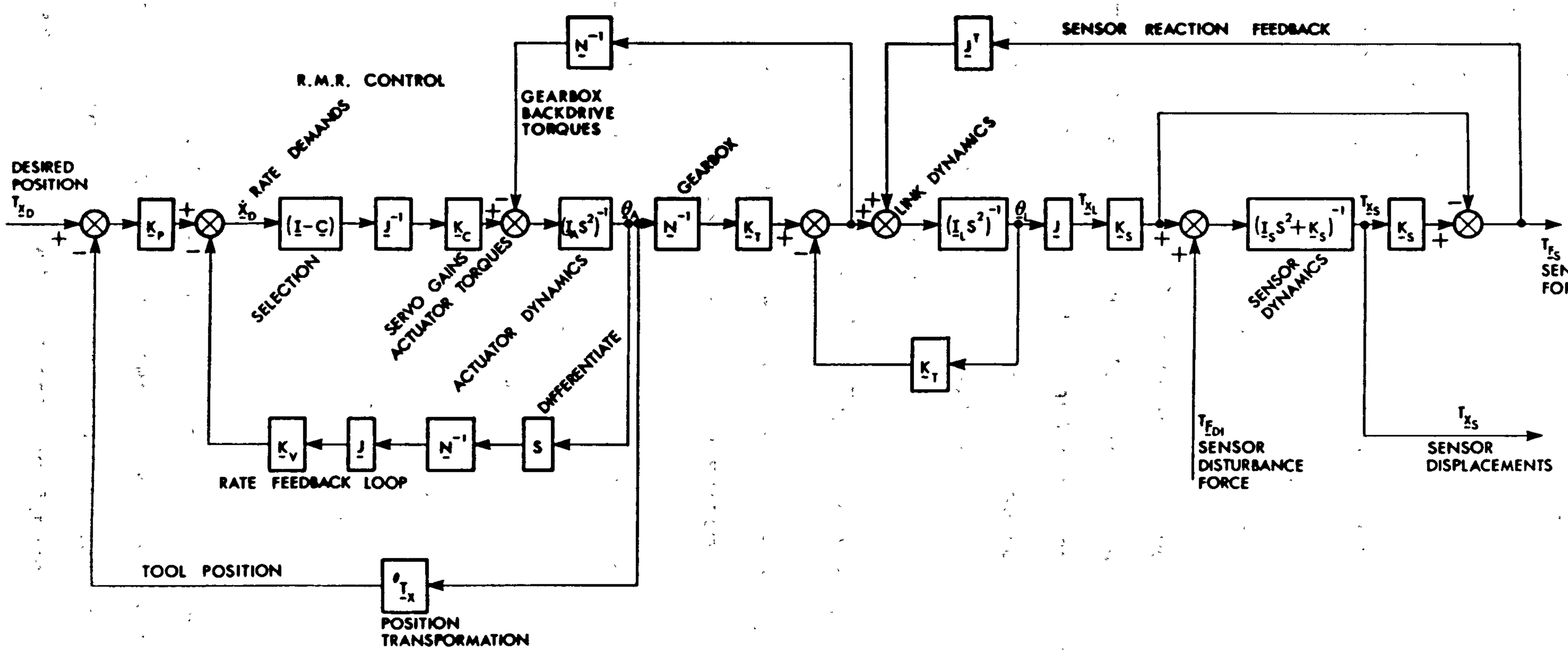
$$\underline{K}_S \, \underline{J} \, \underline{\theta}_L - {}^T\underline{F}_{DI} = (\underline{I}_S.S^2 + \underline{K}_S)\,{}^T\underline{X}_S \qquad (4.7)$$

4.2.4 Overall System

Combining the three dynamic systems of the actuators, link and sensor, each a second order system, gives the system represented in block diagram form shown in figure 4.4. The resulting system is sixth order, and gives the relationship between the sensor end position under the effect of the disturbance force ${}^T\underline{F}_{DI}$.

Figure 4.4 also shows the closed loop position control system containing the Resolved Motion Rate Control (RMRC) subsystem. Most implemented force control schemes employing conventional industrial robots [42,43] use this type of open-loop force control scheme. There are a few instances where the force control loop has been closed [1,4,6] by using the force sensor signal to modify the desired position demands according to force error.

In common with the Hybrid control scheme, compliance selection ($\underline{I}$-$\underline{C}$) is carried out in Cartesian coordinates. $\underline{C}$ is the diagonal compliance selection matrix. For the particular degrees of freedom selected as position control the respective element of $\underline{C}$ is zero (non-compliant). The position controller matrix $\underline{K}_p$ operates in local coordinates and is naturally diagonal. The dynamic equations show that all six degrees of freedom operate in position control, ie. force control is open loop. ${}^{\theta}\underline{T}_X$ represents the position transformation from actuator measurements to Cartesian coordinates.



R.M.R.C. — POSITION CONTROL, SUBJECTED TO SENSOR DISTURBANCE

Figure 4.4.

From the block diagram of figure 4.4 one can see the modifications to equation 4.5, caused by introducing resolved motion rate control (RMRC) [31], together with the position controller and position feedback. If the combined actuator and servo amplifier gains are represented by the diagonal matrix $\underline{K}_C$, then equation 4.5 becomes

$$\underline{K}_C\ \underline{J}^{-1}\ \underline{K}_P\ {}^T\underline{X}_D = [\underline{I}_A.S^2 + (\underline{N}^{-1})^2\ \underline{K}_T + \underline{K}_C\ J^{-1}\ \underline{K}_P\ {}^\Theta\underline{T}_X + \underline{K}_C\ \underline{J}^{-1}\ \underline{K}_V\ \underline{J}\ \underline{N}^{-1}.S]\underline{\Theta}_A - \underline{N}^{-1}\ \underline{K}_T\ \underline{\Theta}_L \qquad (4.8)$$

Equation 4.4 representing link dynamics is modified by replacing the $-\underline{T}_L'$ term by the sensor forces ${}^T\underline{F}_S$ reaction, which is transmitted back through the manipulator structure and represented by $\underline{J}^T\ {}^T\underline{F}_S$. $\underline{J}^T$ is the Jacobian transpose, the transformation of sensor forces to joint forces or torques [58], and $\underline{K}_S({}^T\underline{X}_S - \underline{J}\ \underline{\Theta}_L)$ is the wrist sensor force vector ${}^T\underline{F}_S$. Equation 4.4 therefore becomes,

$$\underline{K}_T(\underline{N}^{-1}\ \underline{\Theta}_A - \underline{\Theta}_L) + \underline{J}^T\ \underline{K}_S({}^T\underline{X}_S - \underline{J}\ \underline{\Theta}_L) = \underline{I}_L.S^2\ \underline{\Theta}_L \qquad (4.9)$$

Equations 4.8 and 4.9 are combined with equation 4.6, which represents actuator dynamics, to give the complete system in matrix form as:

$$\begin{bmatrix} [\underline{I}_A.S^2+(\underline{N}^{-1})^2\ \underline{K}_T+\underline{K}_C\underline{J}^{-1}\ \underline{K}_P\ {}^\Theta\underline{T}_X+\underline{K}_C\ \underline{J}^{-1}\ \underline{K}_V\ \underline{J}\ \underline{N}^{-1}.S] & [-\underline{N}^{-1}\ \underline{K}_T] & [0] \\ [-\underline{K}_T\ \underline{N}^{-1}] & [\underline{I}_L\ S^2+\underline{K}_T+\underline{J}^T\ \underline{K}_S\ \underline{J}] & [-\underline{J}^T\ \underline{K}_S] \\ [0] & [-\underline{K}_S\underline{J}] & [\underline{I}_S.S^2+\underline{K}_S] \end{bmatrix} \begin{bmatrix} [\underline{\Theta}_A] \\ [\underline{\Theta}_L] \\ [{}^T\underline{X}_S] \end{bmatrix}$$

$$= \begin{bmatrix} [\underline{K}_C\ \underline{J}^{-1}\ \underline{K}_P\ {}^T\underline{X}_D] \\ [0] \\ [{}^T\underline{F}_{DI}] \end{bmatrix} \qquad (4.10)$$

The matrix group $\underline{K}_C\ \underline{J}^{-1}\ \underline{K}_P\ {}^\Theta\underline{T}_X$ is the actuator servo position stiffness transformed to actuator coordinates. Likewise the $\underline{K}_C\ \underline{J}^{-1}\ \underline{K}_V\ \underline{J}\ \underline{N}^{-1}$ is the rate feedback transformed to actuator coordinates and $\underline{J}^T\ \underline{K}_S\ \underline{J}$ is sensor stiffness transformed to joint coordinates.

Vectors $\underline{\theta}_A$, $\underline{\theta}_L$, ${}^T\underline{X}_S$ are the dependent variables. Each has six degrees of freedom giving a total of eighteen dependent variables. There are also twelve independent or input variables, the desired position vector ${}^T\underline{X}_D$ and disturbance force vector ${}^T\underline{F}_{DI}$.

The eigenvalues of the characteristic matrix are found from its determinant, yielding six eigenvalues for each degree of freedom, thirty six in total. Unfortunately expressing the determinant in terms of the minor matrices, when the characteristic matrix is naturally partitioned as 3 x 3, is difficult and it is doubtful if meaningful conclusions can be drawn from the analytical results.

However, without proceeding further to determine the system eigenvalues, it is interesting to consider the steady state solution of 4.10 under the influence of the disturbing force vector ${}^T\underline{F}_{DI}$ applied at the sensor end.

4.2.5 Steady State Compliance/Stiffness

Steady state stiffness of the manipulator is defined by noting the small end effector deflections ${}^T\underline{X}_S$ under external force ${}^T\underline{F}_{DI}$. This can be determined by using equation 4.10 and the Laplace Final Value theorem to give three relationships in matrix form:

$$
\begin{array}{lllll}
(\underline{N}^{-2}\,\underline{K}_T + \underline{K}_C\,\underline{J}^{-1}\,\underline{K}_P\,{}^{\theta}\underline{T}_X)\underline{\theta}_A & - \underline{N}^{-1}\,\underline{K}_T\,\underline{\theta}_L & & = & 0 \\
- \underline{K}_T\,\underline{N}^{-1}\,\underline{\theta}_A & + (\underline{K}_T + \underline{J}^T\,\underline{K}_S\,\underline{J})\underline{\theta}_L & - \underline{J}^T\,\underline{K}_S\,{}^T\underline{X}_S & = & 0 \\
 & - \underline{K}_S\,\underline{J}\,\underline{\theta}_L & + \underline{K}_S\,{}^T\underline{X}_S & = & {}^T\underline{F}_{DI}
\end{array}
$$

Eliminating $\underline{\theta}_A$ and $\underline{\theta}_L$ between these equations, gives the end point compliance as:

$$ {}^{T}\underline{X}_S({}^{T}\underline{F}_{DI})^{-1} = \underline{K}_S^{-1} + \underline{J}\ \underline{N}^{-1}\ (\underline{K}_C\ \underline{J}^{-1}\ \underline{K}_P\ {}^{\theta}\underline{T}_X)^{-1}\ \underline{N}^{-1}\ \underline{J}^{T} + \underline{J}\ \underline{K}_T^{-1}\ \underline{J}^{T} \qquad (4.11) $$

Note that the total compliance is the sum of the individual components, for sensor, actuators and transmissions, each resolved to sensor coordinates. The compliance for the actuators seems a little contrived. Firstly the $\underline{K}_C\ \underline{J}^{-1}\ \underline{K}_P\ {}^{\theta}\underline{T}_X$, as explained earlier for equation 4.10, is the servo position stiffness transformed to actuator coordinates. This is inverted to obtain actuator compliance whose effect is transformed back to sensor coordinates. However, the presence of the $\underline{N}^{-1}$ matrices indicates that the effect of the actuator compliance, as seen from the sensor, is reduced by the effects of gearing squared.

Therefore, any static friction effects present in the transmission or actuator drive will be magnified by the gearing ratio so preventing back-driving of the manipulator and effectively creating a non-linearity in actuator compliance. That is, for small sensor disturbance forces ${}^{T}\underline{F}_{DI}$ actuator compliance is removed and equation 4.11 reduces to equation 4.12.

$$ {}^{T}\underline{X}_S({}^{T}\underline{F}_{DI})^{-1} = \underline{K}_S^{-1} + \underline{J}\ \underline{K}_T^{-1}\ \underline{J}^{T} \qquad (4.12) $$

For larger forces, sufficient to overcome actuator friction, actuator compliance control may operate and total sensor compliance will again resemble equation 4.11. This observation has important consequences for the concepts of Active Stiffness Control [38]. Simply that, it is confined to categories of manipulators which can easily back-drive, otherwise end point stiffness cannot be actively controlled.

Equation 4.12 indicates how the end-point stiffness can most effectively be controlled by a compliant wrist. A wrist is both accessible, to

enable instrumentation to be installed, and is without the problems of link dynamics and non-linearities. Perhaps effort should be directed within this approach towards a realistic solution to an adaptively compliant wrist [48,49]. Effectively this would add a miniature manipulator to the robot end effector, able to accommodate small movements. Computational effort behind such a wrist control scheme should not be too excessive as the wrist could be aligned to the current task's coordinate system and driven in a Cartesian frame. Also there would be minimal interference of the conventional industrial manipulator controller and certainly no increase in its computational burden. Possibly such a scheme could be based on the Remote Centre Compliance principle [44-46,49], actively re-positioning the centre of compliance and compliance vector as the task demands without interference of the controlling computer.

Although the above schemes assist in assembly by passively controlling the important contact forces they are essentially position control based. Contact forces can only be controlled reliably if the exact geometry of contact is known, together with the total end-point stiffness. Positional discrepancies are only accommodated if the end-point compliance is soft enough, implying force control is open-loop. Also large compliances at the end effector have the added disadvantage of causing loss in positional accuracy, especially when manipulating large masses or forces. This is particularly apparent in situations where robots are used to polish, fettle deburr etc. because large compliances, necessary to accommodate workpiece positional errors, are intolerant of the tool induced disturbances.

Overcoming disadvantages such as these is achieved by adopting active control of the contact force. This not only compensates for positional inaccuracies of contact but also precisely controls contact forces.

However, actively controlling contact force has its own set of control problems and it is to these problems that the next chapter is devoted.

CHAPTER 5

DYNAMIC FORCE CONTROL

5.1 Introduction

In the previous chapter the need is discussed for some form of compliance in the force sensing system. Compliance reduces the sharp rise in force experienced while tracking unknown irregularities in a high stiffness system. Also, the force sensing should be as near to the desired controlled force as possible. This reduces complications, caused by joint disturbances or link inertial effects, from interfering with the force measurement. For these reasons, sensing joint torques and transforming them to environment coordinates is unsuitable. Hence the selection of a wrist force sensor for reasons of both compliance and simplified dynamic effects.

5.1.1 Preliminary Force Control Results

Initial commissioning results using the Hybrid force/position control scheme with the experimental manipulator, showed the dynamics of the environment to be a significant parameter in the force control stability. As a consequence, environment dynamics are included in the model of the system dynamics.

To illustrate the above point, single axis force control was attempted using the single axis compliant force sensor. The surface being tracked was extremely compliant relative to the sensor and a good response to force demands was easily achieved using relatively high values of proportional gain, see figure 5.1. The same force controller used on a

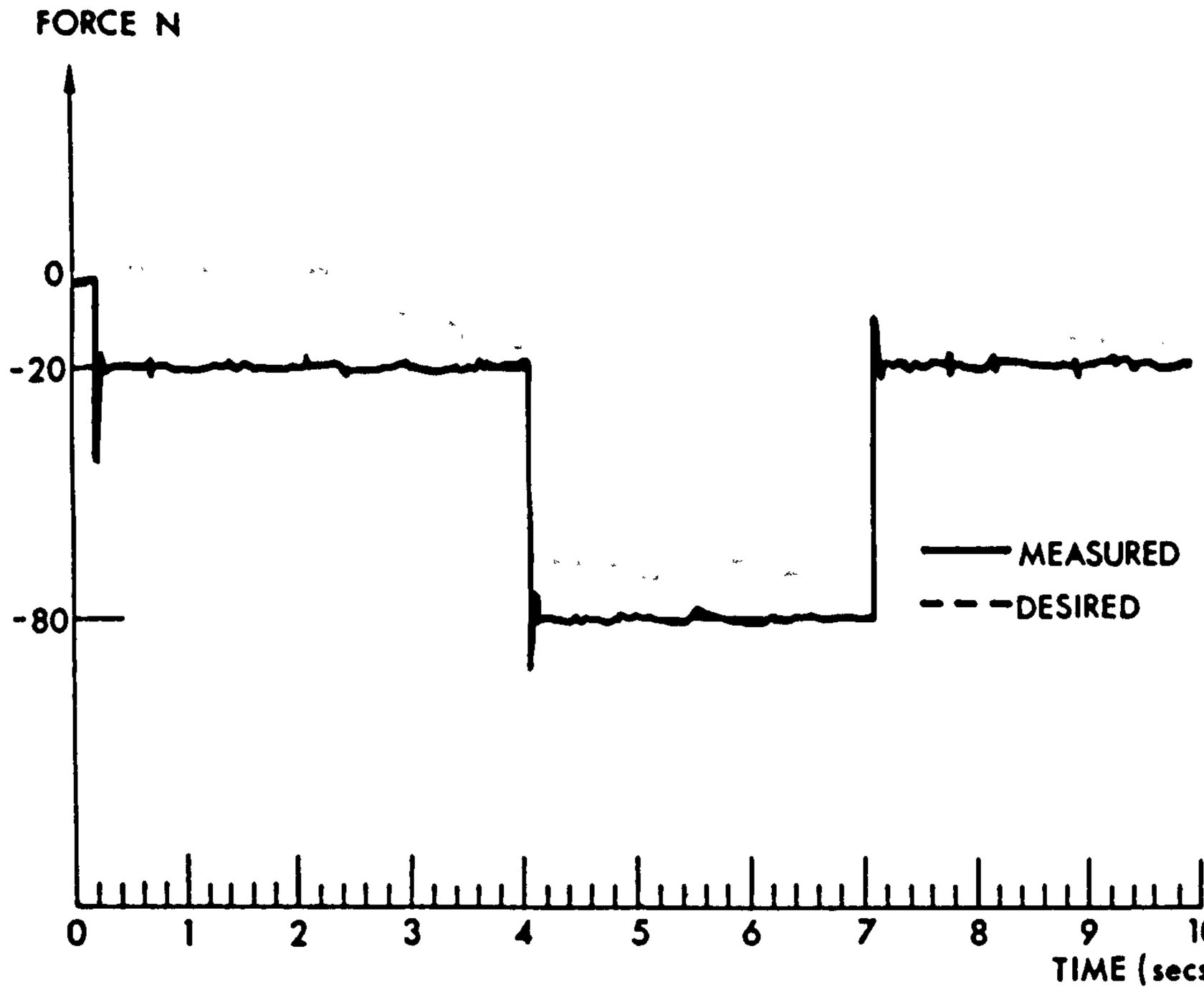


Figure 5.1. Initial experimental results showing force response for contact with a compliant environment using high proportional gain. K_P = 60.0units, $K_E < 10^3$N/m. The response is stable.

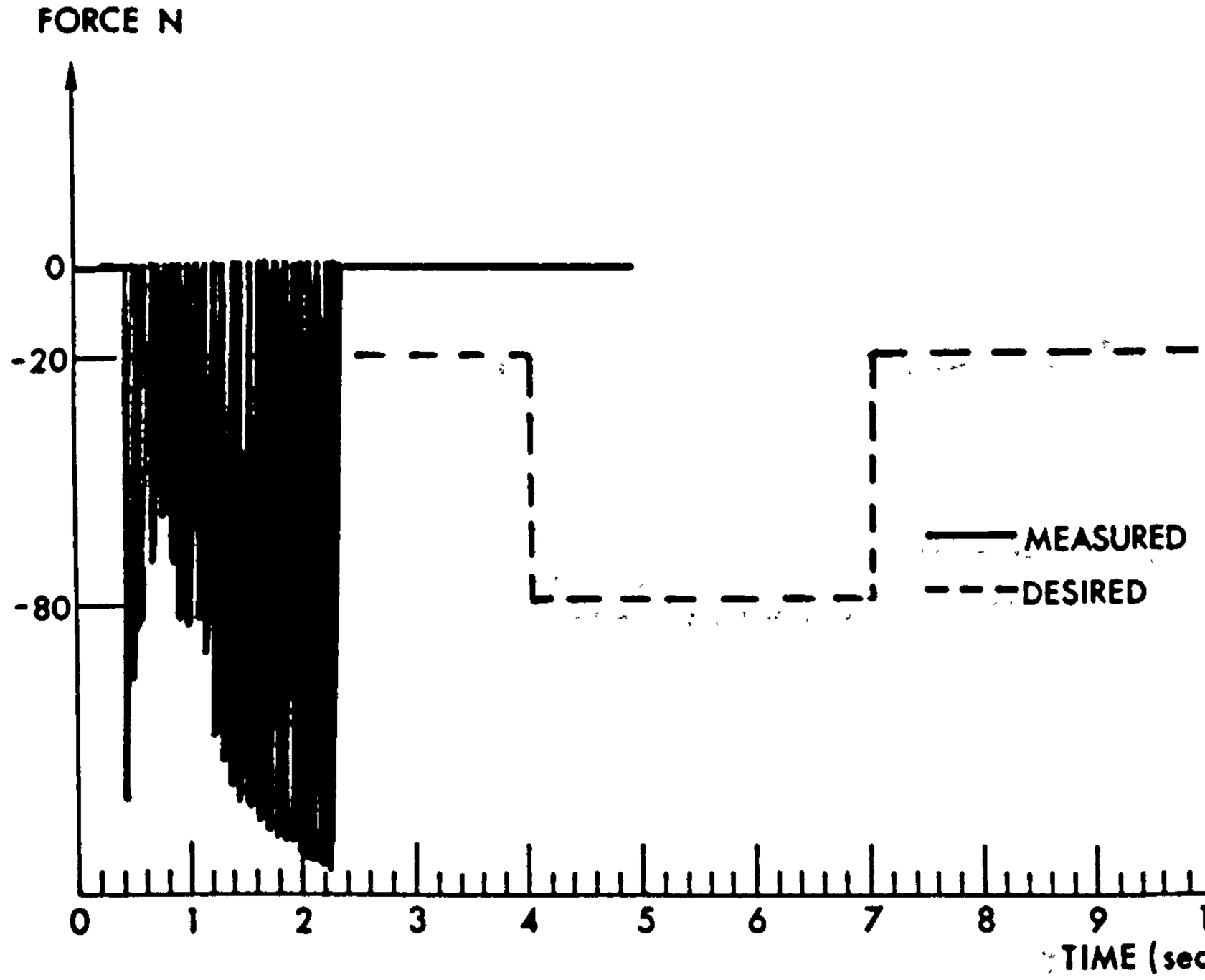


Figure 5.2. Initial experimental results showing force response for contact with a stiff environment using high proportional gain. K_P = 60.0units, $K_E > 10^6$N/m. The response is unstable.

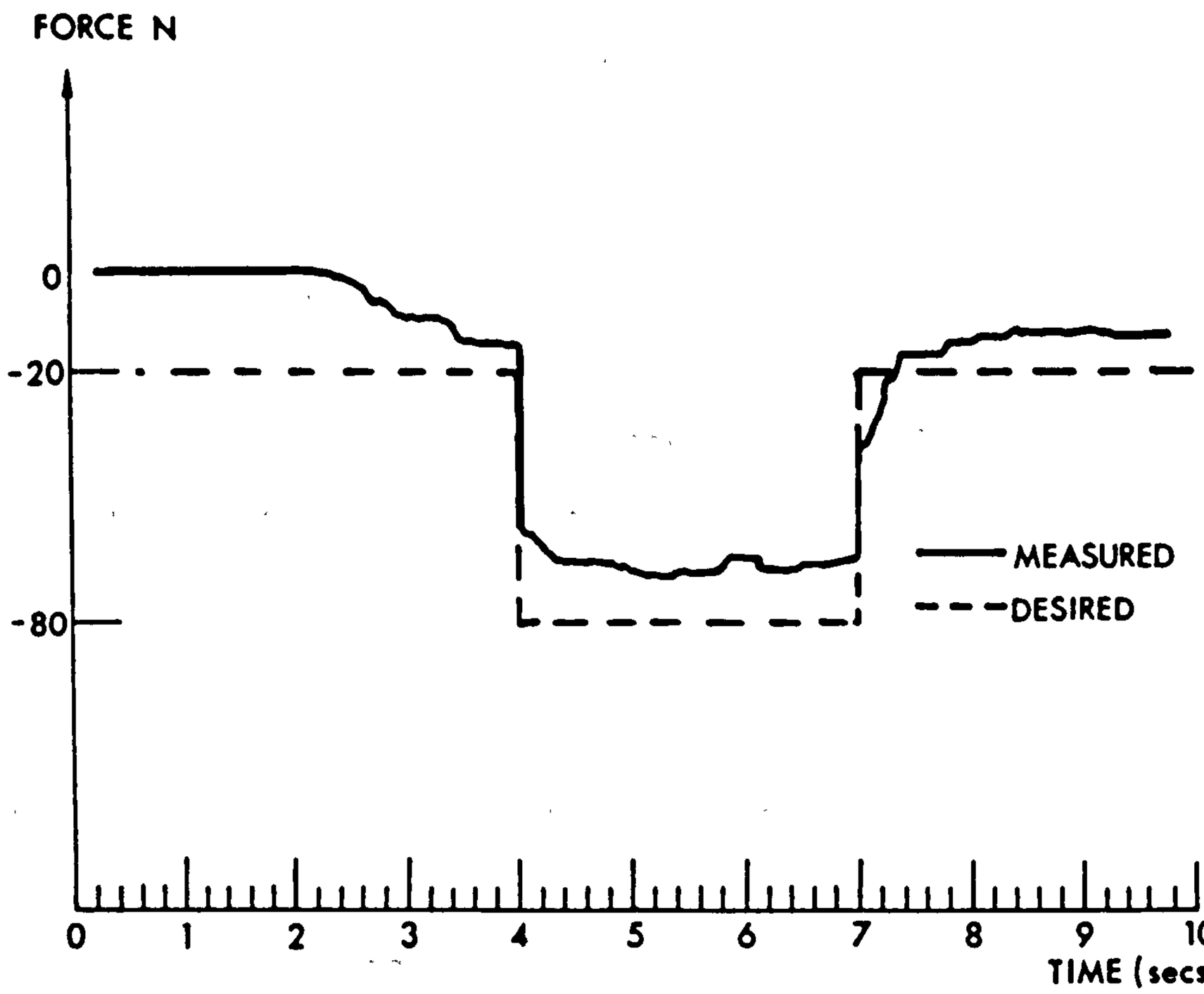


Figure 5.3. Initial experimental results showing force response for contact with a stiff environment using low proportional gain. K_p = 0.3units, $K_E > 10^6$N/m. The response is stable.

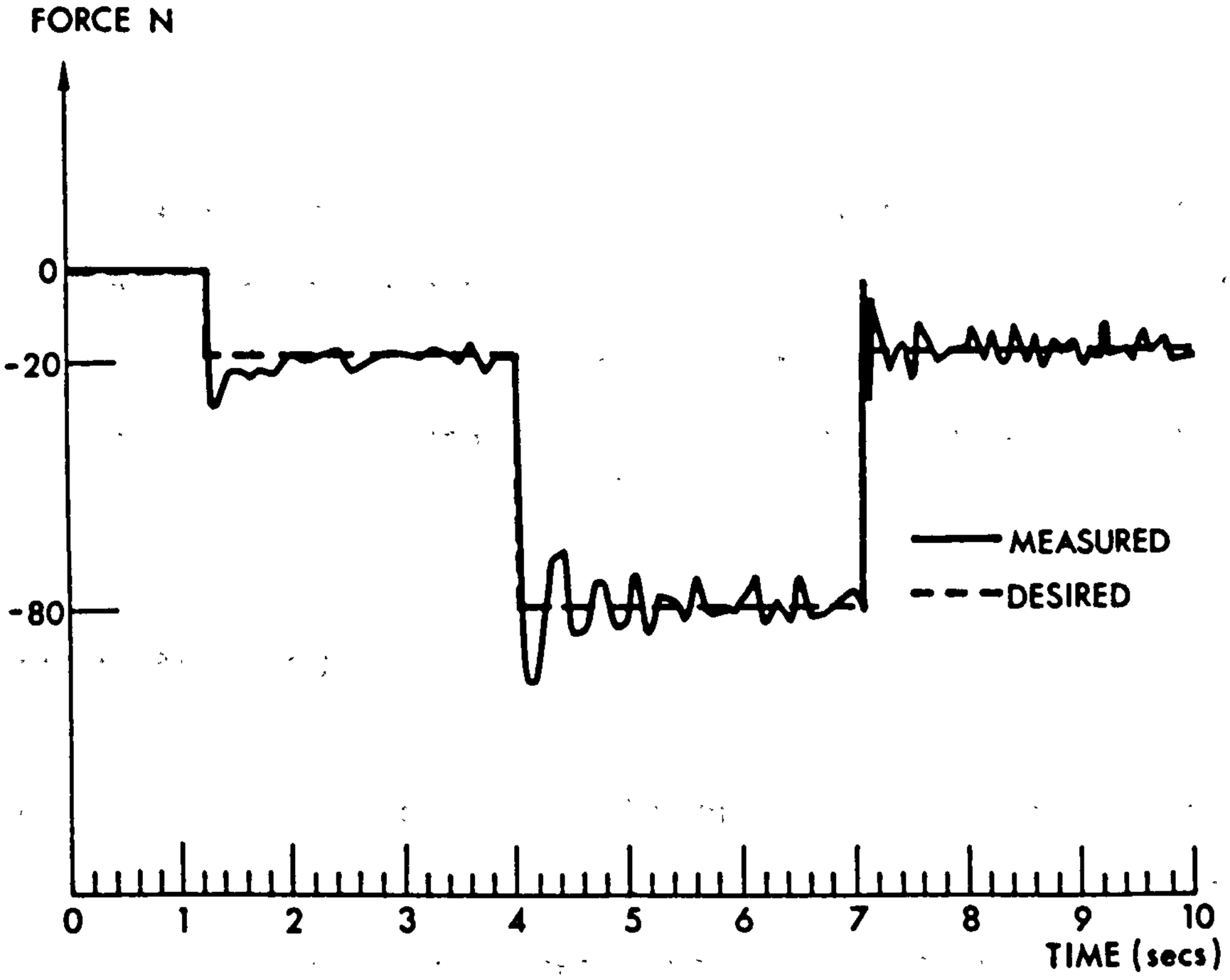


Figure 5.4. Initial experimental results showing force response for contact with a stiff environment using low proportional gain and integral control. K_p = 0.3units, $K_E > 10^6$N/m. The response is stable.

stiff surface was highly unstable displaying an uncontrolled limit cycle, shown in figure 5.2.

When tracking the stiff surface, force control stability could only be achieved by reducing the proportional gain. However, this allowed large steady state errors to remain, see figure 5.3. This error was subsequently removed by incorporating integral control into the force controller, a similar technique to that used by Raibert [34] employing saturation limits.

Integral control, although it is seen to remove the steady state error, produced a decaying oscillation with a long settling time, see figure 5.4, which is clearly undesirable.

These examples serve to illustrate the problems encountered when attempting to achieve force control under varying environmental conditions.

In the following analysis the role of the environment is identified within the dynamics of force control. The various sources of steady state errors are also studied. Resulting from this is a proposal for a new scheme to eliminate steady state error without recourse to integral control.

5.2 Force Control Analyses

The previous chapter establishes the individual dynamic systems for the actuator, link and sensor dynamics. These were finally combined to give the overall dynamic equations for general passive force control. In this chapter the same dynamic subsystems are employed but, in addition, the dynamics of the environment, which are shown to be significant, are also

included. Initially the dynamics of the general environment model are discussed. Subsequently the complete dynamics of general force controlled system are formed. These are then examined to identify parameter influence and steady state response.

The dynamics of force control are studied within the limitations of the generality of six degrees of freedom. In chapter 8 a more specific single degree of freedom analysis is considered together with simulations.

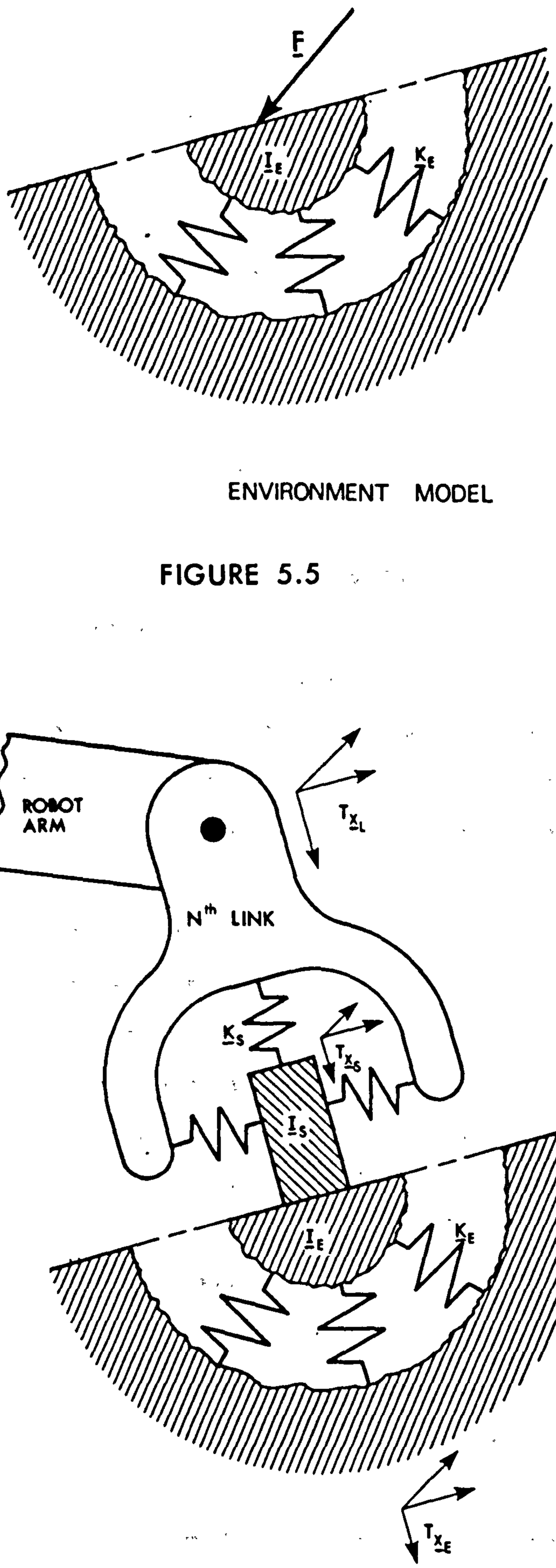
5.3 Environment Dynamics

Often the environment is modelled as too rigid and massive to display meaningful dynamic effects and tends to be ignored in analyses [62]. Such an environment, if it actually exists at all, is a special case of the general environment found in manipulative situations. Manipulators could be asked to perform force control with environments ranging from virtual sponge, thin walled vessels with the occasional rigid stiffeners, to perhaps, a block of steel on elastic foundations. To obtain a simple but effective environment model the model was assumed to be an equivalent mass, supported by a linear suspension, see figure 5.5.

A limitation of this model is that local elastic deformation is ignored, that is, the manipulator does not penetrate the environment. Any local elastic deformation is assumed small and its effect can be grouped with the stiffness of the suspension.

5.3.1 Coupling Effects

In general the environment properties are unknown and the coupling between the deflections of the various degrees of freedoms is assumed to



ENVIRONMENT MODEL

FIGURE 5.5

SENSOR/ENVIRONMENT MODEL, SHOWING BUILT-IN SENSOR

FIGURE 5.6

exist. This coupling may be illustrated by considering a cantilever. Applying a force transversely to the free end results in both a transverse deflection and a rotation of the beam end. These deflections are a function of beam geometry and point of load application. Likewise applying a pure moment to the free end of a cantilever beam causes a change in curvature and a resulting transverse deflection. Often these coupling effects may be minimal and it may be economical to neglect them. However here this is not assumed to be the case and the environment stiffness matrix $\underline{K}_E$ is non-diagonal.

Obviously the stiffness matrix $\underline{K}_E$ will remain unknown unless prior testing of the environment has been performed to establish it. Although measurement may be possible to determine $\underline{K}_E$ by using a suitably instrumented force sensor to measure these displacement coupling effects on-line, the practical and computational problems associated with an empirical up-to-date environment stiffness matrix may prove prohibitive.

The same comments may also be applied to the inertial properties of the environment, which may also suffer coupling effects. Thus the inertia matrix for the environment, $\underline{I}_E$, is also in general non-diagonal and unknown. Again the development of on-line sensing techniques, to determine the coupling effects of the inertial terms, will be extremely complex and perhaps impossible to evaluate in real-time situations.

5.3.2 Adopted Environment Model

To summarise, the matrices representing the inertial and stiffness properties of the environment are assumed non-diagonal. Further, when the manipulator's force sensor contacts the environment, the sensor inertia matrix $\underline{I}_S$ is assumed added to the environment model's inertia $\underline{I}_E$, the resulting matrix $[\underline{I}_E+\underline{I}_S]$ is also non-diagonal and is to be

represented by $\underline{I}_{E,S}$. Note the deflections of the combined system are measured in the same frame as the sensor applied forces, ${}^{T}\underline{F}_{S}$. Adding the respective inertias in this way is only possible if there is a mechanical coupling between them. Take for example a pencil point pressing normally on to the surface of a table. Normal force can easily be applied hence the axial component of the pencil's inertia can be added to the equivalent inertial component of the table's surface. All other degrees of freedom of the pencil remain free preventing the summation of the respective inertial components of the pencil and table.

However, in this analysis for simplicity, force control is assumed in all six degrees of freedom, that is, the sensor is assumed built into the surface. Therefore all the respective inertial elements of the sensor and environment are combined.

The environment/sensor model used can be represented diagrammatically, see figure 5.6, showing the sensor totally built-in.

In figure 5.6 $\underline{K}_{E}$ and $\underline{K}_{S}$ are the respective stiffness matrices of the environment and sensor respectively. ${}^{T}\underline{X}_{L}$ is the vector of incremental link movements resolved to tool Cartesian coordinates, and ${}^{T}\underline{X}_{S}$ is the vector of sensor/environment incremental movements. Since the sensor is integral with the final link, the two coordinate systems are taken to be aligned. Note the movement vector of environment support ${}^{T}\underline{X}_{E}$, is also aligned to tool Cartesian coordinates. ${}^{T}\underline{X}_{E}$ can be thought of as deliberate movement of the environment support, which is treated as an input to the system, while the manipulator attempts to track the contour.

Any form of velocity dependent damping in the environment/sensor system is probably minimal, unpredictable, and consequently ignored. Therefore

the dynamic equations representing environment movement may be written as

$$\underline{K}_S[{}^T\underline{X}_L - {}^T\underline{X}_S] + \underline{K}_E[{}^T\underline{X}_E - {}^T\underline{X}_S] = \underline{I}_{E,S}.s^2\ {}^T\underline{X}_S$$

rewritten as

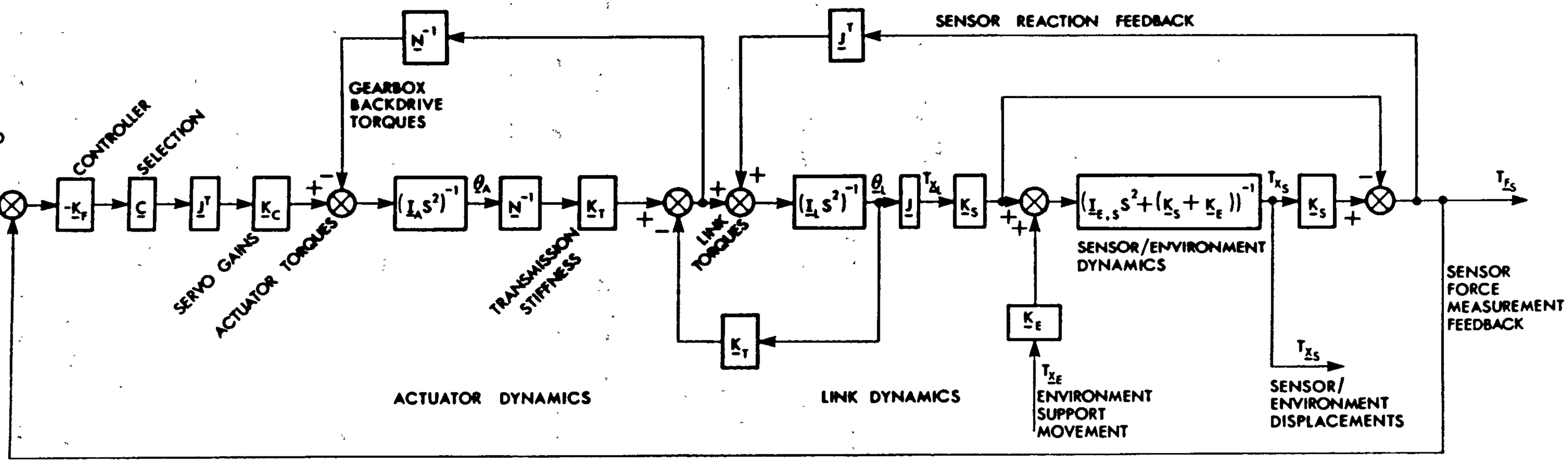
$$\underline{K}_S\ {}^T\underline{X}_L + \underline{K}_E\ {}^T\underline{X}_E = [\underline{I}_{E,S}.s^2 + (\underline{K}_S + \underline{K}_E)]\ {}^T\underline{X}_S \qquad (5.1)$$

where the force measured at the sensor is

$${}^T\underline{F}_S = \underline{K}_S[{}^T\underline{X}_S - {}^T\underline{X}_L]$$

5.4 Manipulator Force Control Dynamics

The block diagram of figure 5.7 is a modification of the block diagram of figure 4.4. This modification is achieved by replacing the position controller and associated feedback loops with a force controller with force feedback and adding the environment/sensor dynamics system, represented by equation 5.1. This block diagram, figure 5.7, now represents the system of manipulator force control dynamics. In this case ${}^T\underline{F}_D$ is the desired control force vector, expressed in tool coordinates. $-\underline{K}_F$ is the diagonal force controller matrix, the negative sign indicating a compressive force demand delivering positive torques. The diagonal compliance selection matrix $\underline{C}$ should have every element set to "1" indicating force control acts on all six degrees of freedom. $\underline{J}^T$, converts the selected cartesian forces to the respective joint torque commands to be input into the servo amplifiers. The overall gains of the servo systems are represented by diagonal matrix $\underline{K}_C$. As with position controlled dynamics of figure 4.4, mechanical damping dependent on velocity is ignored since, under force control, motions and velocities are small. Force rate damping is considered later in the analysis and is contained within controller $\underline{K}_F$.



FORCE CONTROLLED SYSTEM WITH TRANSMISSION STIFFNESSES

Figure 5.7

Equations representing dynamics of force control may be derived from figure 5.7, and written in their natural groups for actuator, link and sensor/environment dynamics give:

for actuator dynamics

$$-\underline{K}_C\ \underline{J}^T\ \underline{K}_F(^T\underline{F}_D - {}^T\underline{F}_S) + \underline{N}^{-1}\ \underline{K}_T(\underline{\Theta}_L - \underline{N}^{-1}\ \underline{\Theta}_A) = \underline{I}_A.S^2\ \underline{\Theta}_A \qquad (5.2)$$

for link dynamics

$$\underline{K}_T(\underline{N}^{-1}\ \underline{\Theta}_A - \underline{\Theta}_L) + \underline{J}^T\ {}^T\underline{F}_S = \underline{I}_L.S^2\ \underline{\Theta}_L \qquad (5.3)$$

and for sensor/environment dynamics by writing equation 5.1 as:

$$\underline{K}_S\ \underline{J}\ \underline{\Theta}_L + \underline{K}_E\ {}^T\underline{X}_E = [\underline{I}_{E,S}.S^2 + (\underline{K}_S + \underline{K}_E)]\ {}^T\underline{X}_S$$

If this latter equation is now written in terms of ${}^T\underline{F}_S$ by replacing ${}^T\underline{X}_S$ by $\underline{J}\ \underline{\Theta}_L + \underline{K}_S{}^{-1}\ {}^T\underline{F}_S$ it gives for sensor/environment dynamics:

$$\{\underline{K}_S - [\underline{I}_{E,S}.S^2 + (\underline{K}_S + \underline{K}_E)]\}\underline{J}\ \underline{\Theta}_L + \underline{K}_E\ {}^T\underline{X}_E = [\underline{I}_{E,S}.S^2 + (\underline{K}_S + \underline{K}_E)]\underline{K}_S{}^{-1}\ {}^T\underline{F}_S \qquad (5.4)$$

These equations, 5.2, 5.3, 5.4, can be presented in a similar form to equation 4.10 giving the partitioned matrix representation of dynamic equations governing force control as:

$$\begin{bmatrix} [\underline{I}_A.S^2+\underline{N}^{-2}\ \underline{K}_T] & [-\underline{N}^{-1}\ \underline{K}_T] & [-\underline{K}_C\ \underline{J}^T\ \underline{K}_F] \\ [-\underline{K}_T\ \underline{N}^{-1}] & [\underline{I}_L.S^2+\underline{K}_T] & [-\underline{J}^T] \\ [0] & [-(\underline{K}_S - \{\underline{I}_{E,S}.S^2+(\underline{K}_S+\underline{K}_E)\})\underline{J}] & [\{\underline{I}_{E,S}.S^2+(\underline{K}_S+\underline{K}_E)\}\underline{K}_S{}^{-1}] \end{bmatrix} \begin{bmatrix} [\underline{\Theta}_A] \\ [\underline{\Theta}_L] \\ [{}^T\underline{F}_S] \end{bmatrix}$$

$$= \begin{bmatrix} [-\underline{K}_C\ \underline{J}^T\ \underline{K}_F\ {}^T\underline{F}_D] \\ [0] \\ [\underline{K}_E\ {}^T\underline{X}_E] \end{bmatrix} \qquad (5.5)$$

vectors $\underline{\Theta}_A$, $\underline{\Theta}_L$, ${}^T\underline{F}_S$ are the dependent variables, each with six degrees of freedom, giving a total of eighteen dependent variables. The

independent input variables are the desired contact force vector ${}^{T}\underline{F}_{D}$ and the environment support position vector ${}^{T}\underline{X}_{E}$. The determinant of the characteristic matrix yields the eigenvalues, thirty six in total.

Complexity, at this general stage, prevents specific conclusions on the characteristics of the force controlled system from being drawn. However, in the following section, further analysis is made possible due to a justified simplification of the system, which reduces the system from eighteen to twelve dependent variables.

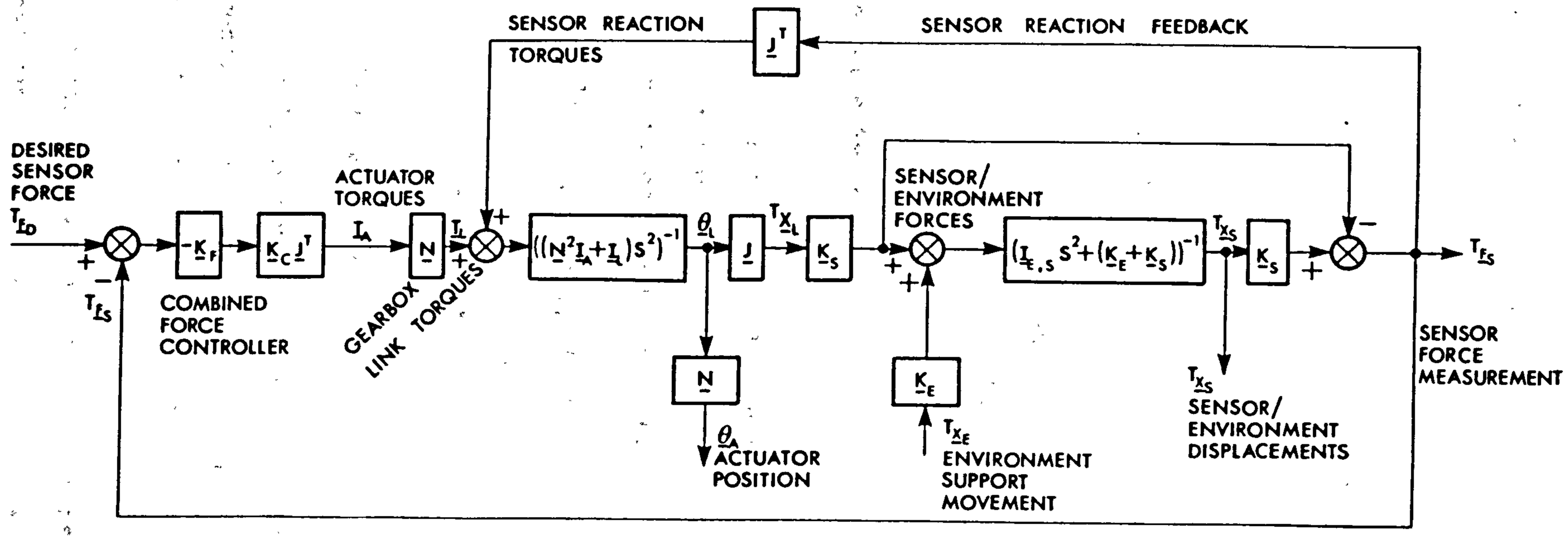
5.5 Reduced Force Control Dynamics

If the manipulator transmission stiffness $\underline{K}_{T}$ is large compared to both the wrist sensor stiffness $\underline{K}_{S}$ and actuator stiffness, then K_{T} may be eliminated combining the actuator inertia $\underline{I}_{A}$ with the manipulator link inertia $\underline{I}_{L}$. This reduces the order of the system from six to four and forms a more manageable expression of the determinant in terms of its partitioned minor matrices.

The experimental manipulator used here employs extremely rigid transmissions, tensioned rigid chains driving ballscrews, hence the above simplification is not an approximation. Following from this, it can be easily shown that actuator inertia $\underline{I}_{A}$, when transformed to link coordinates through a gear reduction of $\underline{N}$, becomes $\underline{N}^{2}\ \underline{I}_{A}$. That is the effective link inertia becomes $(\underline{N}^{2}\ \underline{I}_{A} + \underline{I}_{L})$, which is again non-diagonal. The reduced block diagram representing the force controlled system is thus shown in figure 5.8. The dynamic equations are now in terms of $\underline{\theta}_{L}$ and ${}^{T}\underline{F}_{S}$ only. This gives:

for link dynamics

$$-\underline{N}\ \underline{K}_{C}\ \underline{J}^{T}\ \underline{K}_{F}({}^{T}\underline{F}_{D} - {}^{T}\underline{F}_{S}) + \underline{J}^{T}\ {}^{T}\underline{F}_{S} = (\underline{N}^{2}\ \underline{I}_{A} + \underline{I}_{L})S^{2}\ \underline{\theta}_{L} \qquad (5.6)$$



FORCE CONTROLLED SYSTEM WITH RIGID TRANSMISSIONS

Figure 5.8.

and for sensor/environment dynamics

$$\underline{K}_S \, \underline{J} \, \underline{\theta}_L + \underline{K}_E \, {}^T\underline{X}_E = [\underline{I}_{E,S}.s^2 + (\underline{K}_S + \underline{K}_E)] \, {}^T\underline{X}_S \qquad (5.7)$$

again if ${}^T\underline{X}_S = \underline{J} \, \underline{\theta}_L + \underline{K}_S{}^{-1} \, {}^T\underline{F}_S$ then the equations can be written in matrix form as:

$$\begin{bmatrix} [(\underline{N}^2 \, \underline{I}_A + \underline{I}_L)s^2] & [-(\underline{J}^T + \underline{N} \, \underline{K}_C \, \underline{J}^T \, \underline{K}_F)] \\ [(\underline{I}_{E,S}.s^2 + \underline{K}_E) \, \underline{J}] & [\{\underline{I}_{E,S}.s^2 + (\underline{K}_S + \underline{K}_E)\}\underline{K}_S{}^{-1}] \end{bmatrix} \begin{bmatrix} [\underline{\theta}_L] \\ [{}^T\underline{F}_S] \end{bmatrix}$$

$$= \begin{bmatrix} [-\underline{N} \, \underline{K}_C \, \underline{J}^T \, \underline{K}_F \, {}^T\underline{F}_D] \\ [\underline{K}_E \, {}^T\underline{X}_E] \end{bmatrix} \qquad (5.8)$$

5.6 Steady State Force Response

Interesting results come from considering the steady state response of the force controlled system. Using the Laplace final value theorem, for time tending to infinity, equation 5.8 gives:

$$\underline{N} \, \underline{K}_C \, \underline{J}^T \, \underline{K}_F \, {}^T\underline{F}_D = (\underline{J}^T + \underline{N} \, \underline{K}_C \, \underline{J}^T \, \underline{K}_F) \, {}^T\underline{F}_S \qquad (5.9)$$

and

$$0 = \underline{K}_E \, \underline{J} \, \underline{\theta}_L + (\underline{K}_E + \underline{K}_S) \, \underline{K}_S{}^{-1} \, {}^T\underline{F}_S$$

written as

$$-\underline{K}_S \, (\underline{K}_E + \underline{K}_S)^{-1} \, \underline{K}_E \, \underline{J} \, \underline{\theta}_L = {}^T\underline{F}_S \qquad (5.10)$$

Equation 5.10 simply gives the relationship between manipulator Cartesian movement, $\underline{J} \, \underline{\theta}_L$ and $-\underline{K}_S \, (\underline{K}_E+\underline{K}_S)^{-1} \, \underline{K}_E$, the equivalent sensor/environment stiffness to sensor force vector ${}^T\underline{F}_S$.

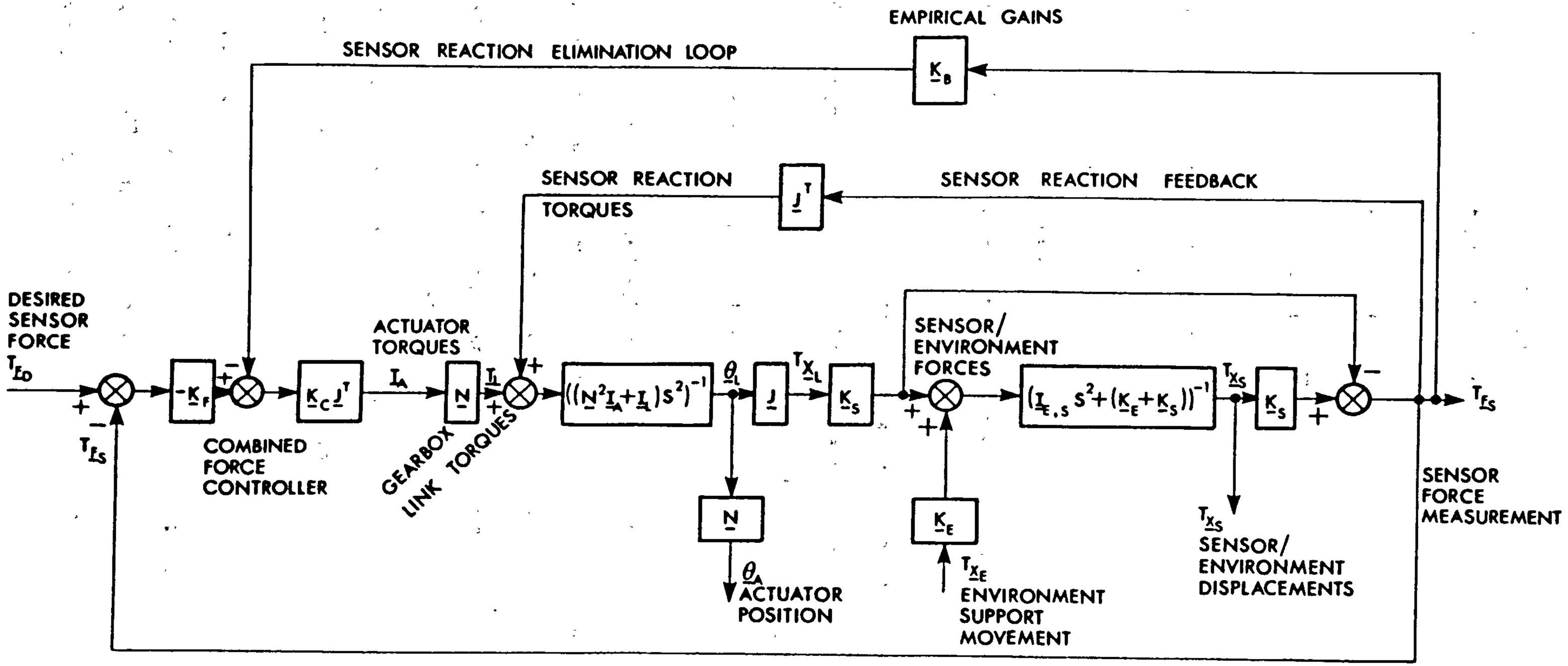
Equation 5.9 indicates steady state force response and agreement with experimental observations, in that, for a low values of $\underline{K}_F$ there is a force error caused by the presence of the $\underline{J}^T$ term. The $\underline{J}^T$ term represents the reaction of the sensor attempting to backdrive the

manipulator links. This restricts the links' full motion and hence produces reduced compression/tension of the sensor/environment system. Or, more conventionally, the error can be considered due to using a proportional controller, without integral action, which only produces actuator torques when an error in force demand is experienced. Preliminary results, in figure 5.4, and equation 5.9, show how integral control can remove all steady state error. Gearing reduction $\underline{N}$ can also contribute to a reduction in steady state error, but for stability reasons $\underline{K}_C$ and $\underline{K}_F$ may need to be reduced accordingly.

The presence of the $\underline{J}^T$ term in equation 5.9 is obviously significant and reducing it would greatly improve the steady state error for low values of $\underline{K}_F$, and thus allow for the complete elimination of integral control. Removing integral control also produces a further benefit of improving the settling time of the transient response.

5.6.1 Eliminating Natural Force Feedback

Modifying the block diagram of figure 5.8, by implementing an additional feedback loop which utilizes the force sensor signal transformed to joint coordinates, gives figure 5.9. This additional feedback loop is designed to oppose the natural feedback term $\underline{J}^T\ {}^T\underline{F}_S$, which appears in equation 5.8. In figure 5.9, the feedback loop summation is positioned after the force controller $\underline{K}_F$, but before the transformation $\underline{J}^T$. Consequently only simple additions are required without recourse to additional transformations. Hence there is no increase in computational burden. However, there are certain conditions which must be satisfied before the technique can function correctly, consider figure 5.9,



FORCE CONTROLLED SYSTEM WITH RIGID TRANSMISSIONS
+ SENSOR REACTION ELIMINATION

Figure 5.9.

showing the additional sensor reaction elimination loop. Under these conditions the summation of link torques gives:

$$\text{Total link torques} = \underline{N}\ \underline{K_C}\ \underline{J}^T[-\underline{K_F}({}^T\underline{F_D}-{}^T\underline{F_S}) + \underline{K_B}\ \underline{J}^T\ {}^T\underline{F_S}] - \underline{J}^T\ {}^T\underline{F_S}$$

$$=-\underline{N}\ \underline{K_C}\ \underline{J}^T\ \underline{K_F}({}^T\underline{F_D}-{}^T\underline{F_S}) + (\underline{N}\ \underline{K_C}\ \underline{J}^T\ \underline{K_B} - \underline{J}^T)\ {}^T\underline{F_S}$$

Therefore the sensor reaction forces may be eliminated if $\underline{N}\ \underline{K_C}\ \underline{J}^T\ \underline{K_B}=\underline{J}^T$. Hence $\underline{N}\ \underline{K_C}$ must be made the diagonal matrix $\underline{I}.NK_C$ where NK_C is now a scalar multiplier. Likewise the matrix $\underline{K_B}$ must be $\underline{I}.K_B$, where K_B is also a scalar. With this condition satisfied, K_B can then be evaluated as $K_B=(N.K_C)^{-1}$. Alternatively K_B may be found empirically.

Effectively the force sensor disturbance on the robot structure is removed. This could also have useful applications in full dynamic position control of manipulators [39] where unknown payload masses destroy the dynamic modelling of the manipulator.

Current industrial manipulators could also benefit from this technique by overcoming the dynamic effects experienced when moving large inertial loads, even though velocities and accelerations may be low and the inertial effects within the manipulator neglected. Elimination of this natural force feedback would also remove position deflections caused by payload mass without resort to integral control to counteract the steady state deflections at the actuators.

Since the group $(\underline{J}^T + \underline{N}\ \underline{K_C}\ \underline{J}^T\ \underline{K_F})$ in equation 5.8 represents the proportional feedback, where $\underline{J}^T$ contributes to the feedback, eliminating $\underline{J}^T$ should allow the proportional component in $\underline{K_F}$ to be increased, and so improve stability and force response.

The results of this technique are documented in the experimentation and

simulation chapters, sections 7.3 and 8.7 respectively. It is shown to be successful in eliminating the need for integral control by overcoming steady state error due to sensor reactions, apparent at low values of $\underline{K}_F$ as predicted. Also the force responses are seen to be slightly less oscillatory, indicating a reduction in proportional feedback.

Unfortunately not all the errors in force response are due to sensor reaction. Unmodelled friction in the robot's transmissions complicate the situation by creating static force errors independent of payload. However, the technique should still operate if high friction is absent in the transmissions, or if the robot is equipped with an invariance technique to eliminate it [12,61].

This natural feedback elimination technique has several advantages over that of feedforwarding the desired force ${}^T\underline{F}_D$. Using the feedforward approach at high force levels could be disastrous if contact is lost causing the robot to accelerate at high rate. Secondly, the overall proportional gains are reduced by the technique which allows the controller's proportional gains to be increased. Most importantly, the technique eliminates the effect of tool disturbances at the robot end effector, feedforward control does not.

5.7 Characteristic roots

The characteristic roots of the dynamic force controlled system are found from the determinant of the characteristic matrix of equation 5.8.

An expression can now be obtained for a reduced determinant using Schur's Formulae [68], which simply written gives the reduced

determinant of a partitioned matrix,

$$\text{DET} \begin{vmatrix} [\underline{A}] & [\underline{B}] \\ [\underline{C}] & [\underline{D}] \end{vmatrix}$$

where $\underline{A}$, $\underline{B}$, $\underline{C}$, $\underline{D}$ are the partitioned matrices, as:

$$\text{DET} \mid [\underline{A}].[\underline{D}] - [\underline{C}].[\underline{B}] \mid$$

providing both $\underline{C}$ and $\underline{D}$ are diagonal.

In this case the corresponding $\underline{C}$ term in equation 5.8 can be made diagonal if the $\underline{J}$ term is removed. This is achieved by transforming $\underline{\theta}_L$ to manipulator co-ordinates resolved to Cartesian axes, ${}^T\underline{X}_L$. The reduced determinant can then be written:

$$\text{DET} \mid (\underline{N}^2\ \underline{I}_A + \underline{I}_L).S^2\ \underline{J}^{-1}\ [\underline{I}_{E,S}.S^2 + (\underline{K}_E + \underline{K}_S)]\ \underline{K}_S^{-1} + (\underline{J}^T + \underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_F)\ (\underline{I}_{E,S}.S^2 + \underline{K}_E) \mid = 0$$

Rewriting the determinant as:

$$\text{DET} \mid \underline{K}_S^{-1} \mid . \mid (\underline{N}^2\ \underline{I}_A + \underline{I}_L)S^2\ \underline{J}^{-1}\ [\underline{I}_{E,S}.S^2 + (\underline{K}_E + \underline{K}_S)] + (\underline{J}^T + \underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_F).(\underline{I}_{E,S}.S^2 + \underline{K}_E)\ \underline{K}_S \mid = 0 \qquad (5.11)$$

Expanding gives:

$$\text{DET} \mid \underline{K}_S^{-1} \mid . \mid [(\underline{N}^2\ \underline{I}_A + \underline{I}_L)\ \underline{J}^{-1}\ \underline{I}_{E,S}]S^4 + [(\underline{N}^2\ \underline{I}_A + \underline{I}_L)\ \underline{J}^{-1}\ (\underline{K}_E + \underline{K}_S) + (\underline{J}^T + \underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_F)\ \underline{I}_{E,S}\ \underline{K}_S]S^2 + (\underline{J}^T + \underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_F)\ \underline{K}_E\ \underline{K}_S \mid = 0 \qquad (5.12)$$

5.8 Force Control Discussion and Analysis

The problems associated with maintaining force response despite a changing environment are discussed. Initially the problem is presented for the system without damping to allow the fundamentals of the dynamics

to be considered. Later the complexity is increased by the introduction of force rate damping, velocity damping and friction effects.

5.8.1 Stability

In considering the expanded determinant (equation 5.12) the roots of which represent the eigen-values of the system characteristic, there are no terms in odd powers of S. Any form of damping, whether it is viscous damping in the robot and/or sensor or force rate in the controller, would contribute odd terms of S, hence the possibility of stability. In this case, with no assumed damping, stability can only be achieved if the controller $\underline{K}_F$ becomes $(\underline{K}_P + \underline{K}_D.S)$, where $\underline{K}_P$ is a diagonal matrix containing proportional gain constants and $\underline{K}_D$ is also a diagonal matrix composed of derivative action gains necessary for stability. Initial experimental results have precluded the use of large values of $\underline{K}_D$ because errors in the digital derivation of the force signal cause an effect similar to high frequency noise, producing instability. An alternative means of obtaining accurate evaluation of force rate is required and direct measurement may be a possibility, using a force rate sensor. Such a sensor is currently being developed by the author and results are to be published at a later date.

5.8.2 Force Response with Proportional Control

Once the desired force response has been achieved for a given manipulator/environment configuration by empirical measurement and adjustment, it would be beneficial, if possible, to maintain such a response, by adapting the controller, $\underline{K}_P$, to the changing environment and manipulator dynamics. Take for example, the changing environment experienced whilst contour tracking a turbine blade. Firstly a changing environment is caused by the variation of the inertial and stiffness

properties of the surface. Therefore, in tracking a turbine blade, the stiffness $\underline{K}_E$ can change significantly (see chapter 7), thin and flexible at the trailing edge, rigid around the leading edge. Local inertial components may also change, but not as dramatically as with a solid component such as a blade. Blades are in general solid bodies and, as a result, the inertial components would tend to remain constant around that section, ignoring twist. Large inertial variations would possibly be experienced while manipulating structural components and not rigid bodies. Therefore, if manipulator movement is limited, resulting in an unchanging manipulator inertia matrix, then the S^4 term of equation 5.12 should also remain unchanged. These significant changes in force response can be concluded to be dependent on changes in environment stiffness $\underline{K}_E$ alone, which affects the S^2 and S^0 terms only of equation 5.12. As equation 5.12 shows, it is impossible to maintain both the S^2 and S^0 terms constant while the environment stiffness $\underline{K}_E$ varies. This can be easily illustrated in that, the criterion imposed on $\underline{K}_P$ for maintaining the S^2 terms constant, as $\underline{K}_E$ varies, is different from that for maintaining the S^0 terms constant. If a second variable is introduced, actively varying K_S, (although not possible in the experimentation conducted here), then the roots of equation 5.12 can be maintained and response preserved. The two criterion represented as simultaneous equations are:

$$\begin{aligned} S^2 \text{ term.} \quad & (\underline{N}^2\ \underline{I}_A + \underline{I}_L)\ \underline{J}^{-1}\ (\underline{K}_E + \underline{K}_S) + \underline{K}_S\ (\underline{J}^T + \underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_P)\ \underline{I}_{E,S} = \underline{P} \\ S^0 \text{ term.} \quad & (\underline{J}^T + \underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_P)\ \underline{K}_E\ \underline{K}_S = \underline{Q} \end{aligned} \tag{5.13}$$

In cases where robot configuration changes significantly while manipulating the empirical matrices P and Q should be evaluated for best response at every possible 'station', stored and used as a look up table while moving. Refinement of these 'station' positions depends on the required tolerance of the force responses. The uncontrollable variation

in the $\underline{J}$ terms and in the inertial terms of the manipulator as configuration changes will, in general, have an uncontrollable affect on transient force response using this proposed proportional controller. Such a controller operates in task coordinates of the environment and can therefore only compensate for the changes in environment dynamics catered for in the model. However, under certain conditions relating to the constant S^0 term, $(\underline{J}^T + \underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_P)\ \underline{K}_E\ \underline{K}_S$, high values of $\underline{K}_S$ and $\underline{K}_E$ may demand a low value in $\underline{K}_P$. Values of $\underline{K}_P$ so low that the natural feedback term $\underline{J}^T$ dominates in equation 5.13, preventing the desired response from being achieved. Again this illustrates the desire to eliminate the natural feedback term, $\underline{J}^T$.

The above considers the effect of proportional gain on force response. Under changing $\underline{K}_E$ it is impossible to maintain response if only $\underline{K}_P$ is adjustable. But response can be maintained if the manipulator configuration remains unchanged and $\underline{K}_S$ becomes adjustable. However, in these cases all forms of damping have been neglected, which implies that the system is always on the verge of instability and that the above discussions are not in fact practical. The cases where velocity dependent damping is considered are discussed later in section 5.10.

5.8.3 Force Rate Damping

Adding force rate damping to the controller, where $\underline{K}_F$ is replaced by $(\underline{K}_P+\underline{K}_D S)$, to improve stability, adds further complication to the characteristic matrix. Equation 5.12 becomes:

$$\begin{aligned} \text{DET}\Big|\ \underline{K}_S^{-1}\ \Big| . \Big|\ [(\underline{N}^2\ \underline{I}_A + \underline{I}_L)\ \underline{J}^{-1}\ \underline{I}_{E,S}].S^4 + [\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_D\ \underline{I}_{E,S}\ \underline{K}_S]S^3 \\ + [(\underline{N}^2\ \underline{I}_A + \underline{I}_L)\ \underline{J}^{-1}\ (\underline{K}_E + \underline{K}_S) + (\underline{J}^T + \underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_P)\ \underline{I}_{E,S}\ \underline{K}_S]S^2 \\ + [\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_D\ \underline{K}_E\ \underline{K}_S]S + (\underline{J}^T + \underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_P)\ \underline{K}_E\ \underline{K}_S\ \Big| \ = \ 0 \end{aligned} \tag{5.14}$$

There are now two additional terms in odd powers of S. In the case of the S^3 term it increases only with sensor stiffness and is independent of $\underline{K}_E$. Whereas the damping associated with the S term varies with the product of $\underline{K}_E$ $\underline{K}_S$, and increases with increasing sensor and environment stiffnesses. Therefore, the criterion for maintaining both odd terms constant, by adjusting $\underline{K}_D$ in the face of $\underline{K}_E$ variation, is impossible.

The fact that the S^3 term actually increases with increasing $\underline{K}_E$ and $\underline{K}_S$ should not cause problems with stability and may even prove desirable, increasing the damping effect as $\underline{K}_E$ increases. But in cases where force rate errors are apparent, this effect may itself cause instability. The effects of $\underline{K}_D$, $\underline{K}_E$ and $\underline{K}_S$ variation are considered in more detail in the single degree of freedom simulations of chapter 8.

5.8.4 Effect of Parameter Variation on Steady State Response

Recalling equation 5.9, steady state force response is given by:

$$\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_P\ {}^T\underline{F}_D = (\underline{J}^T + \underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_P)\ {}^T\underline{F}_S \qquad (5.9)$$

It is clearly seen that variation in $\underline{K}_P$ will affect steady state force response, enforcing the argument to maintain $\underline{K}_P$ as high as possible. Increasing $\underline{K}_P$ should also improve surface approach speeds and disturbance rejection, see section 7.4.

5.8.5 Measurement of $\underline{K}_E$

The above analysis presupposes a measurement environment stiffness $\underline{K}_E$. This could possibly be done by measuring the contact force and the corresponding environment displacement, hence evaluating $\underline{K}_E$. However the manipulation task operates in local co-ordinates and absolute positions are undefined. Therefore it is proposed to use the

measurement of change in force and the corresponding changes in displacement over the same period, thus calculating $\underline{K}_E$. Taken a step further, measurement of force rate and sensor velocity may be used. There are other problems involved with $\underline{K}_E$ determination, especially when constant forces are demanded. These are due to the need for reasonable changes in both force and position before an accurate value of $\underline{K}_E$ can be found. Logic will be required to limit the evaluation of $\underline{K}_E$ to the period of force transient, and also perhaps an averaging scheme to reduce possible errors.

5.9 Special Cases

The effects of two special cases are considered on the characteristic matrix. These are the case where the sensor stiffness is considerably softer than the environment and the case where the environment is considerably softer than the sensor. Both cases allow a certain degree of simplification to the governing dynamic equations.

5.9.1 Low Sensor Stiffness ($\underline{K}_S << \underline{K}_E$)

Often the environment is considerably stiffer than the sensor stiffness $\underline{K}_S$ and equation 5.11 can be written:

$$\text{DET} \mid \underline{K}_S^{-1} \mid . \mid (\underline{N}^2\ \underline{I}_A + \underline{I}_L)\ \underline{J}^{-1}.s^2 + (\underline{J}^T + \underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_F)\ \underline{K}_S \mid$$
$$. \mid (\underline{I}_{E,S}.s^2 + \underline{K}_E) \mid = 0 \qquad (5.15)$$

for $\underline{K}_S << \underline{K}_E$

This shows that the composition of the eigenvalues is due to both the manipulator force control oscillations and sensor/environment mass oscillations.

Environment/Sensor Oscillation

The part of the determinant of equation 5.15, whose eigenvalues relate to sensor/environment dynamics, indicates a possible explanation for Raibert's results [4] with the initial Hybrid work. These results gave a low frequency response of the manipulator to force demand, (indicating the bandwidth of the force control system), plus a superimposed low amplitude, high frequency vibration associated with the sensor mass. Raibert indicated the latter was due to the oscillation of the sensor mass alone, frequency, f, modelled by:

$$f = \frac{1}{2\pi}\left[\frac{K_E + K_S}{I_S}\right]^{\frac{1}{2}}$$

However this was probably an inaccurate model since I_S would have associated with it component of environment inertia I_E giving:

$$f = \frac{1}{2\pi}\left[\frac{K_E + K_S}{I_S + I_E}\right]^{\frac{1}{2}}$$

for the single degree of freedom case. The correlation Raibert found with his quoted model was probably within unquoted experimental error. Also, had sufficient force sensor damping been present, this sensor mass oscillation should have decayed. Certainly, in the initial results quoted here, such vibration has not been apparent, probably owing to the presence of such sensor damping, which at this stage is uncontrollable.

Oscillation of manipulator force control system.

The eigenvalues associated with the oscillation of force controlled

robot system are given by:

$$\text{DET} \mid [(\underline{N}^2\ \underline{I}_A + \underline{I}_L)\ \underline{J}^{-1}]S^2 + [\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_D\ \underline{K}_S]S + [\underline{J}^T + \underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_P\ \underline{K}_S] \mid = 0 \qquad (5.16)$$

for $\underline{K}_S << \underline{K}_E$.

Where the controller $\underline{K}_F$ now contains the derivative action, component $\underline{K}_D$, necessary for stability.

For a given configuration, ie. constant $\underline{J}$ and $\underline{I}_L$, the roots are determined by the value of force controller gains, $\underline{K}_F$, and are greatly influenced by the force sensor stiffness, $\underline{K}_S$ in the product $\underline{K}_P\ \underline{K}_S$. This work is further investigated by simulation in section 8.5.3.

The chief disadvantage of using a force sensor with low stiffness is the reduction in positional certainty. This will have two effects, the loss of positional accuracy while moving in positional mode and also the likelihood of tool induced vibrations affecting the payload mass. This will possibly result in tool bounce at the surface and the breakdown of the above analysis.

However, the advantages of low sensor stiffness for force control are many fold, summarized as follows.

a) Force control is no longer dependent on environment stiffness. Response is only affected by changes in manipulator geometry. This simplifies the requirements of controller parameter selection.

b) Low sensor stiffness allows higher values of proportional gain to be employed, so reducing steady state error and the effects of unmodelled stiction and coulombic friction in the transmissions.

c) Higher gains allow higher response speeds to be obtained without

the problems of instability. This has been observed by Roberts et al [62] using a single degree of freedom system with rigid environment and varying sensor stiffness. However their results were not supported by analytical justification.

5.9.2 High Sensor Stiffness $\underline{K}_S >> \underline{K}_E$

This section assumes that the sensor stiffness is allowed to increase and dominate the environment by at least an order of magnitude. Thus the previously referred to problems encountered by poor positional sensor restraint are overcome.

In this case the environment is relied upon to create the necessary compliance in the system to avoid producing the uncontrollable high frequency forces which are above the controller's bandwidth. Clearly, high $\underline{K}_E$ combined with high $\underline{K}_S$ is to be avoided for this reason alone. But, if this condition does arise, the analysis breaks down owing to neglected terms, eg. transmission and structural stiffnesses, becoming significant.

Equation 5.14 is slightly simplified by the condition $\underline{K}_S >> \underline{K}_E$ giving the determinant:

$$\begin{aligned} \text{DET}\,|\,\underline{K}_S^{-1}\,|\,.\,|\,&[(\underline{N}^2\,\underline{I}_A + \underline{I}_L)\,\underline{J}^{-1}\,\underline{I}_{E,S}]S^4 + [\underline{N}\,\underline{K}_C\,\underline{J}^T\,\underline{K}_D\,\underline{I}_{E,S}\,\underline{K}_S]S^3 \\ &+ [(\underline{N}^2\,\underline{I}_A + \underline{I}_L)\,\underline{J}^{-1}\,\underline{K}_S + (\underline{J}^T + \underline{N}\,\underline{K}_C\,\underline{J}^T\,\underline{K}_P)\,\underline{I}_{E,S}\,\underline{K}_S]S^2 \\ &+ [\underline{N}\,\underline{K}_C\,\underline{J}^T\,\underline{K}_D\,\underline{K}_E\,\underline{K}_S]S + (\underline{J}^T + \underline{N}\,\underline{K}_C\,\underline{J}^T\,\underline{K}_P)\,\underline{K}_E\,\underline{K}_S\,| \;=\; 0 \end{aligned} \tag{5.17}$$

noting the S^2 term is now no longer dependent on variations in $\underline{K}_E$, providing it remains much less than $\underline{K}_S$. The S^0 and S^1 terms are the only terms directly dependent on $\underline{K}_E$ and by varying parameters $\underline{K}_P$ and $\underline{K}_D$ in the controller these terms may be maintained constant. However, varying

$\underline{K}_P$ and $\underline{K}_D$ will also directly affect the S^2 and S^3 terms and thus the force response cannot be maintained while environment stiffness varies, despite the fact the environment stiffness is considerably less than $\underline{K}_S$. This effect is demonstrated experimentally, in section 7.6, by artificially lowering the environment stiffness, this must be done because the nature of the force sensor precludes such significant increases in its stiffness.

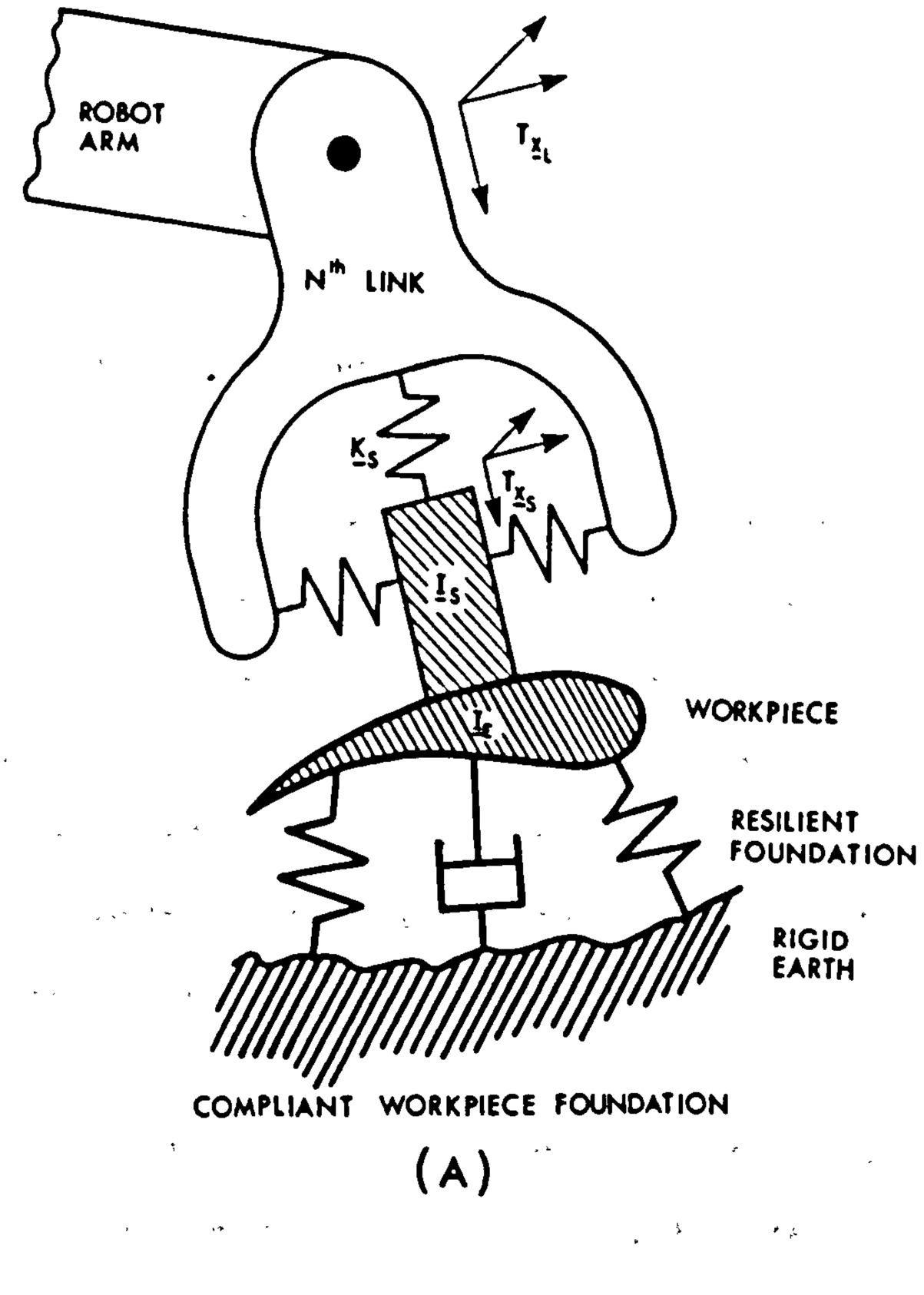
5.9.3 Constant Environment Stiffness

The advantages of artificially maintaining a low constant environment stiffness remove the problems associated with a changing environment described above. Under this constant environment condition equation 5.17, and hence force response, would remain unchanged for a given manipulator configuration. Therefore, if force response was found to vary significantly with changing configuration, or stability became a problem, then a technique similar to equation 5.13 could be employed to compensate for the configuration effects.

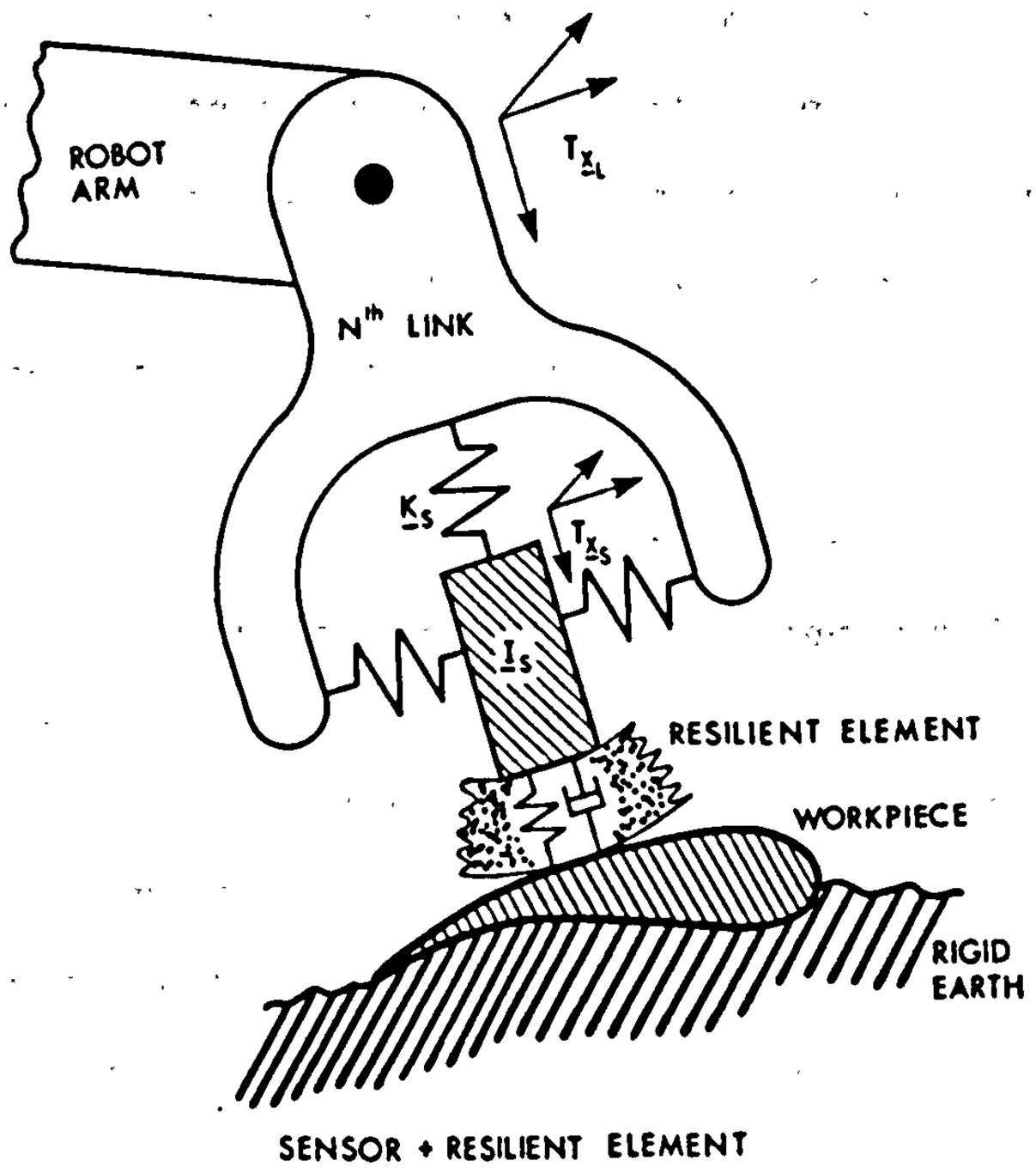
This soft environment can be achieved mechanically, consider the following two arrangements.

Compliant Workpiece Support

Figure 5.10(a) shows a solid body, such as a blade, supported by a uniform stiffness foundation. A carefully chosen rubber could easily be used in such an application. The inherent damping properties would be useful to damp tool induced vibrations, or alternatively a mechanical damping system could be attached to the workpiece. Choosing a low suspension stiffness would allow high values of proportional gain to be used, hence achieving good response and steady state force error. This



(A)



(B)

Figure 5.10. Two methods for maintaining the environment stiffness constant.

approach would be particularly advantageous if the polishing medium was unavoidably rigid, for example a grinding wheel.

Compliant Element at Sensor End

A different, but equivalent, approach to the suspended environment just described is to add a compliant or resilient element between the sensor and the environment. This arrangement is shown in figure 5.10(b) which indicates a similarity to the suspended environment of figure 5.10(a). In fact, a compliant polishing 'bob' or a rubber rimmed wheel supporting a polish belt, commonly used today in manual polishing, is exactly this, a compliant element between the tool and environment.

The characteristics of both the systems in figure 5.10 are equivalent, although physical parameters will differ. Figure 5.10(b) shows how a soft compliance, between the sensor and environment, isolates the environment from the control problem. The combined inertia term $\underline{I}_{E,S}$, is reduced to just $\underline{I}_S$, the sensor/tool inertia term. The environment, if possible, should be attached rigidly to earth. Reducing the possibility of the environment forming an additional system in the dynamic train and so avoiding consequential vibration problems.

Equipping the sensor end with a resilient element removes the need from the sensor or environment to provide the system compliance. This could allow higher sensor stiffnesses to be employed, so increasing the natural frequency of the supported tool and the positional accuracy of the end effector. In the unlikely event of tool vibration still posing a problem, a resilient medium with a high energy absorbing capacity could be used as the compliant element to attenuate such vibrations.

5.10 Velocity Damping and Friction Considerations

5.10.1 Joint Friction Effects

The joint friction considered here is composed of two main components. The first is a linear velocity damping component caused by back emf. or motor regulation. This effectively decreases motor current, and hence torque, with increasing motor speed. The retardation is analogous to viscous friction and is easily modelled using data from motor characteristics. The second component is made up of the less than desirable effects of friction in the transmissions. This type of friction is extremely difficult to model and to compensate for. Its highly non-linear nature and associated degrees of randomness makes modelling virtually impossible. Such randomness is due to a number of reasons, chiefly: manufacturing tolerances, surface finish, lubrication, dirt, etc. If the vector of joint friction torques $\underline{T}_{FRIC}$, composed of both modelled and unmodelled components, is included in the equation of joint dynamics, then equation 5.6 becomes:

$$-\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_F\ ({}^T\underline{F}_D - {}^T\underline{F}_S) + \underline{J}^T\ {}^T\underline{F}_S + \underline{T}_{FRIC} = [(N^2\ \underline{I}_A + \underline{I}_L)S^2]\ \underline{\theta}_L \qquad (5.18)$$

The effect of $\underline{T}_{FRIC}$ on controlling torques can now be seen.

Although little is known about the complexity of these frictional disturbances, they can be seen to have two main effects; (i) on transient response and steady state error, (ii) on stability. Unwanted friction torques interfere with the control torques, resulting in a deterioration of steady state error and force response, figures 7.1 and 7.2 are examples of this. In addition, force control stability will be affected if joint movement occurs, as it typically does in compliant conditions. These friction forces oppose motion and therefore force response is impeded. Energy is dissipated and so stability is improved.

Hence, the unpredictable nature of the friction creates a highly non-linear form of viscous damping, making it difficult to find a model equivalent [69] of a viscous damping term which could be used in the characteristic matrix, equation 5.8 above.

Owing to the complexity of the above problems modelling techniques have been unsuccessful in attempting to eliminate these friction effects completely. But, modelling apart, two main schools of thought have been directed at this friction problem; (i) utilising a direct drive [70] arm with inherent low friction at the joints, (ii) employing joint torque sensors in a local invariance loop around each joint [61]. Direct drive arms are limited by present motor technology, tending to be weak and thus unsuitable for tasks involving high manipulative forces. Invariance techniques appear to be the likely choice to eliminate joint friction, but to be realistic are beyond the scope of this research where static friction effects in the joints must be tolerated.

Clearly, in the results described, the friction forces have a substantial static component. Providing the proportional controlling torques, $\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_P\ ({}^T\underline{F}_D - {}^T\underline{F}_S)$, are much greater than the static friction torques, the effects can be ignored and the earlier analysis in this chapter holds. It also follows that, owing to the large static friction component, the energy dissipated in friction is dependent on joint velocity rather than on the transmitted manipulation forces. So, providing the compliances in the system are relatively low, velocities will also be low and the friction damping effects can be ignored. Under these conditions, while velocity damping terms may be neglected, the frictional interference on force response and steady state may still be significant. This judgement again depends on the relative magnitudes of static friction and the controlling torques. These problems are addressed in more detail in section 7.5 of the experimental chapter.

At this stage in force manipulation research there are significant problems with joint friction forces. A clear understanding of the friction's properties, and how to avoid the consequences, will be necessary in related future work.

5.10.2 Friction - Affect on Stability and Response

Section 5.10.1 discussed a model equivalent of joint friction because the viscous damping term was difficult, if not impossible, to evaluate owing to the random and unpredictable nature of the friction. If existence of such a viscous damping model is assumed, although largely unquantifiable, then the effect on the system eigenvalues may be considered.

The link dynamics equation 5.18 is modified to equation 5.19 by incorporating equivalent viscous matrix and static component vector, $\underline{C}_L$ and $\underline{T}_{STAT}$ respectively. Where the $\underline{T}_{FRIC}$ of equation 5.18 has been replaced by its individual components, equivalent viscous damping term $\underline{C}_L$ and static friction torques $\underline{T}_{STAT}$, both terms act in joint coordinates. Component $\underline{C}_L$ contains both the velocity dependent damping due to friction and motor back emf. Additional controller based velocity damping, $\underline{K}_V$ is also included in the derivative term of equation 5.19, this is software implemented and operates in local Cartesian coordinates. The velocity damping K_V is a matrix of diagonal action constants and uses the velocity of the manipulator end, determined from the differential of incremental link movement ${}^T\underline{X}_L$, to calculate the controller damping torques. Additional velocity damping has been included to improve the overall velocity dependent damping of the manipulator. This also helps to reduce the reliance of the joint based velocity dependent damping on manipulator configuration. Experiments

described in section 7.5 employ such a scheme.

$$-\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_F\ ({}^T\underline{F}_D - {}^T\underline{F}_S) + \underline{J}^T\ {}^T\underline{F}_S + \underline{T}_{STAT}$$
$$= [(\underline{N}^2\ \underline{I}_A + \underline{I}_L)S^2 + (\underline{C}_L + \underline{K}_V\ \underline{J})S]\ \underline{\theta}_L \qquad (5.19)$$

Note, the velocity damping in equation 5.19 is transformed to act in the same coordinate system as the joint frictions by using $\underline{J}\ \underline{\theta}_L$ to represent ${}^T\underline{X}_L$. Thus the complete set of dynamic equations incorporating viscous and velocity damping may be written in partition matrix form, similar to equation 5.8, to give equation 5.20. $\underline{T}_{STAT}$, the unknown static friction component, can be considered to be a disturbance to the system and as such appears on the left hand side of equation 5.20, alongside the desired torque vector.

$$\begin{bmatrix} [(\underline{N}^2\ \underline{I}_A + \underline{I}_L)S^2 + (\underline{C}_L + \underline{K}_V\ \underline{J})S] & [-(\underline{J}^T + \underline{N}\ \underline{K}_C\ \ \underline{J}^T\ \underline{K}_F)] \\ [(\underline{I}_{E,S}.S^2 + \underline{K}_E)\ \underline{J}] & [\{\underline{I}_{E,S}.S^2 + (\underline{K}_S + \underline{K}_E)\}\ \underline{K}_S{}^{-1}] \end{bmatrix} \begin{bmatrix} [\underline{\theta}_L] \\ [{}^T\underline{F}_S] \end{bmatrix}$$
$$= \begin{bmatrix} [-\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_F\ {}^T\underline{F}_D + \underline{T}_{STAT}] \\ [\underline{K}_E\ {}^T\underline{X}_E] \end{bmatrix} \qquad (5.20)$$

Expanding the determinant of equation 5.20 gives equation 5.21 which represents the eigenvalues of the system.

$$\text{DET}\ |\underline{K}_S{}^{-1}|.|[(\underline{N}^2\ \underline{I}_A + \underline{I}_L)\ \underline{J}^{-1}\ \underline{I}_{E,S}]S^4$$
$$+ [\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_D\ \underline{I}_{E,S}\ \underline{K}_S + (\underline{C}_L\ \underline{J}^{-1} + \underline{K}_V)\ \underline{I}_{E,S}]S^3$$
$$+ [(\underline{N}^2\ \underline{I}_A + \underline{I}_L)\ \underline{J}^{-1}\ (\underline{K}_E + \underline{K}_S) + (\underline{J}^T + \underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_P)\ \underline{I}_{E,S}\ \underline{K}_S]S^2$$
$$+ [\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_D\ \underline{K}_E\ \underline{K}_S + (\underline{C}_L\ \underline{J}^{-1} + \underline{K}_V)(\underline{K}_E + \underline{K}_S)]S$$
$$+ (\underline{J}^T + \underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_P)\ \underline{K}_E\ \underline{K}_S\ | \ = \ 0 \qquad (5.21)$$

Note the additional viscous damping terms, in odd powers of S, alongside the existing force rate damping terms. The force rate and velocity damping terms appear in local Cartesian coordinates while the equivalent viscous friction terms are in joint coordinates. Adding the velocity

damping term to the force controlled system is seen to contribute similar terms to that of the viscous joint friction alone.

Clearly the dynamics of the force controlled system are now considered to be extremely complex with many independent variables. Thus the task of maintaining force response, as environment stiffness varies, with the proposed controller is impossible.

CHAPTER 6

EXPERIMENTAL RIG

6.1 Introduction

Robot controlled polishing, as discussed in chapter 3, has been shown to be a demanding task from both a control and a mechanical view point. As a result of the polishing criterion specified in chapter 3, various manipulator designs have been considered.

Firstly, the manipulator must be capable of adequate articulation to perform negotiation of complex geometrical forms, such as turbine blading. High manipulator load capacity is also an important requirement, as polishing productivity can be increased by utilising larger abraiding surfaces at higher loads and velocities. But, in addition to the large loads and disturbance loads of the polishing process, the manipulator must also sustain the weight of tooling or workpiece, which can be considerable.

6.2 Manipulator Design Considerations

Owing to the mechanical and controller limitations of commercially available robots, a decision was made to design and build a research robot and controller able to meet the mechanical requirements of the polishing task and the control requirements of the Hybrid control scheme.

The choice of mechanical design was based on a review of current

manipulator technologies which encompassed serial and parallel link designs.

6.2.1 Serial Link Configuration

Highly articulated movement is possible with a well designed serial link manipulator. But from a mechanical view point there are serious limitations with such a design. The structure itself is complicated with complex actuation and transmissions. Also, as a consequence of having to carry its own weight, low payload to weight ratios are common. Position errors and end effector compliances are relatively large, owing primarily to the cumulative errors of the serial link design.

For the above reasons the performance of serial link manipulators in relation to force manipulation tasks was not considered desirable. A manipulator based on parallel configuration provided a solution.

6.2.2 Parallel Link Configuration

Various parallel link configured manipulators based on the Stewart Platform are discussed in chapter 2 [20-28].

The major disadvantages of serial link manipulators are overcome by parallel actuation. Each limb of the parallel manipulator is supported by the base and as a consequence load capacity and stiffness are extremely high. Since a single link connects the end effector to the base the cumulative joint errors of the serial design are eliminated and hence high positional accuracy is easily achievable.

Limitations of the parallel axes manipulator are geometrical, resulting in both a restricted operating envelope and articulated capacity. This

contrasts with the benefits of serial link manipulators. However, in manipulative force controlled situations a reduced operating envelope is not generally a serious limitation, but limited articulation could be, this depends on the task.

6.2.3 Combined Parallel and Serial Drives

The design approach was primarily conceived and adopted to combine the benefits of the parallel and serial drive technologies while minimizing their limitations.

Benefits from reductions in mechanical and computational complexity were gained by restricting the research manipulator to operate in a plane. Thus, full freedom of movement was achieved by using a three axes manipulator. These simplifications allowed greater computational capacity to be directed at the problems associated with simultaneous force and position control, which remains fundamentally unchanged by the simplifications.

The load and stiffness capability is provided by triangulated linear axes, consisting of twin ballscrews terminating at a joint at the apex of a triangle. The necessary articulation is provided by integrating a serial drive wrist structure into the apex joint. Thus the ballscrew drives provide the gross movement capacity of the manipulator, while the serial wrist provides articulation. Currently there are two known prototype manipulators, [26,27] based on similar principles of actuation. These have apparently been designed and constructed concurrently with this work and are discussed in chapter 2.

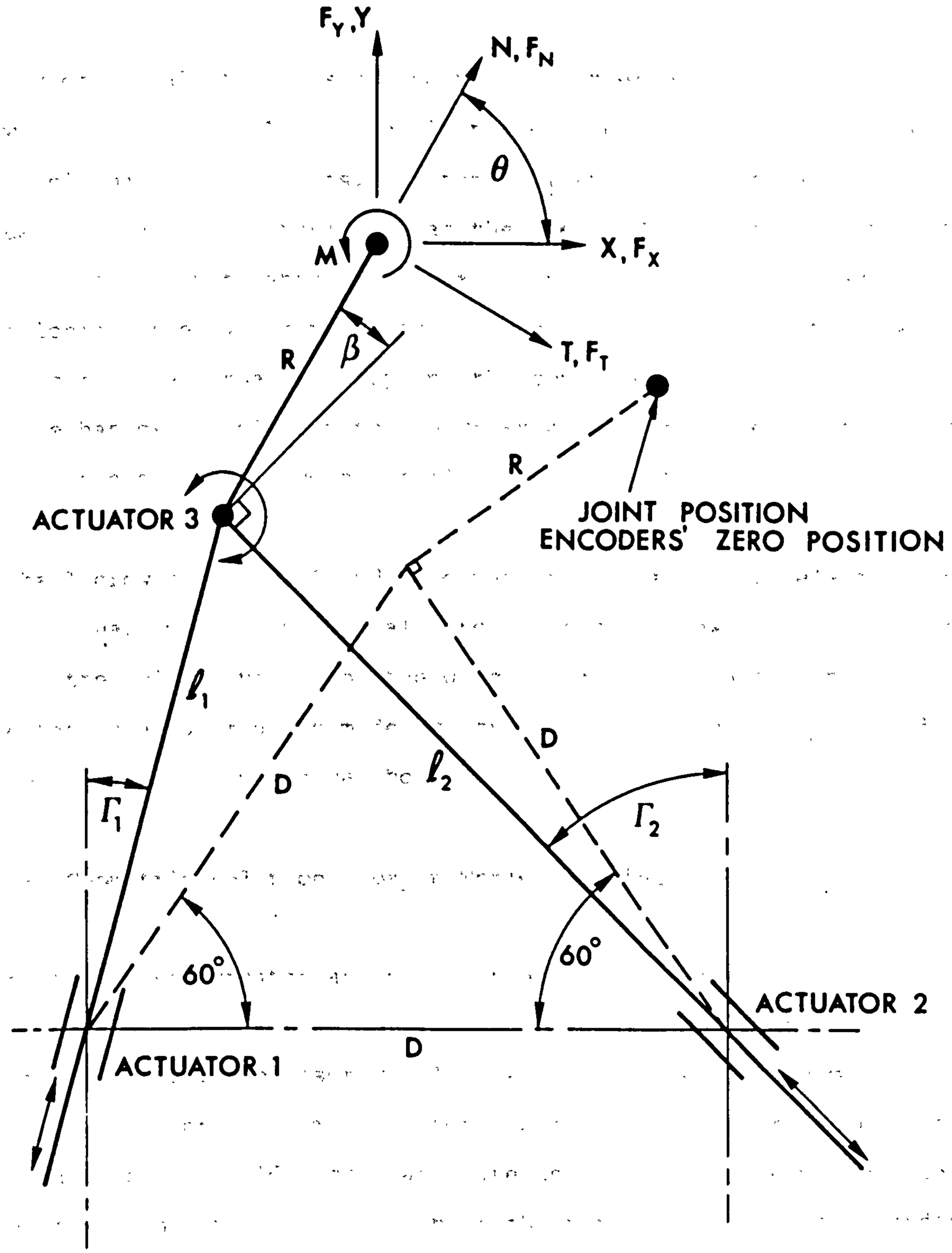


Figure 6.1. Geometrical arrangement of the experimental manipulator.

6.3 Geometry and Kinematics

The geometrical features of the planar experimental manipulator are shown in figure 6.1. As explained earlier its main features are parallel ballscrew drives, lengths l_1 and l_2, which are driven by actuators 1 and 2 and terminate at the apex. The serially driven wrist, of length R, pivots about the apex joint and is driven by actuator 3. Positioning the wrist pivot axis to be coincident with the ballscrew apex simplifies the geometry and the consequential computations, but adds mechanical complexity to the wrist actuation. The wrist actuation design is described in more detail in the section 6.4.3.

The ballscrew actuators 1 and 2 convert rotary motion of electric motors into linear motion in the ballscrew. As ballscrew lengths l_1 and l_2 vary, the pivot point of the wrist moves in planar space. Gross orientation adjustment is made by actuator 3, an electric motor driving the wrist via a harmonic gearbox.

6.3.1 Geometric Relationships in World Coordinates

Both the wrist actuator and position encoder are attached to ballscrew 2. Therefore, as shown in figure 6.1, wrist movement, β, is made relative to this ballscrew. l_1, l_2 and β together form the joint coordinate vector. The manipulator operates in two coordinate systems, world and tool. World coordinates are used to bring the manipulator and effector to the absolute position of the task. Whereas tool coordinates are primarily used during the task, by orientating the tool to the natural coordinate system of the task in hand.

World coordinates are shown as X and Y with θ, the absolute angle θ is measured from the X axis. Tool coordinates are represented by N and T,

when N is aligned to the wrist axis and T is orthogonal to the wrist, θ is the common angle coordinate for both tool and World coordinates.

Information for figure 6.1

l_1, l_2 are the lengths of the ballscrew 1 and 2 respectively, they are joint coordinates.

β is the angular displacement joint coordinate of the wrist relative to ballscrew 2.

X,Y are the World Cartesian coordinates of the end effector or wrist tip.

θ is the absolute orientation coordinate of the wrist.

N,T are the Cartesian tool coordinates. N is aligned to the wrist, T is orthogonal.

F_X,F_Y are the World coordinate frame components of force along the X and Y directions, acting at the wrist tip.

M is the moment acting in the θ direction at the wrist tip.

F_N,F_T are the tool coordinate frame components of force acting along the N and T directions, at the wrist tip.

R is the length of the wrist.

D is the distance between actuators 1 and 2., 700mm.

Γ_1, Γ_2 are the angles between ballscrews and the World Y axis.

6.3.2 World Coordinates

Absolute position measurement given by vector (X,Y,θ) is made relative to an arbitrary datum. The World position origin is defined as being the apex of the triangle, where l_1=l_2=D and the wrist is aligned to the X axis. This differs from the joint position zero indicated in figure

6.1. Hence the absolute positional relationships are given by:

$$X = D/2 - l_2 \sin \Gamma_2 + R \cos (\beta+\Gamma_2)$$

$$Y = -D.\sqrt{3}/2 + l_2 \cos \Gamma_2 + R \sin (\beta+\Gamma_2)$$

$$\Theta = \beta+\Gamma_2 \qquad (6.1)$$

where Γ_2 is completely defined in terms of l_1 and l_2 as:

$$\sin \Gamma_2 = \frac{1}{l_2} \left[\frac{D}{2} - \frac{(l_1^2-l_2^2)}{2D} \right] \qquad (6.2)$$

Rather than determine the complex analytical solution of X,Y,θ in terms of l_1, l_2 and β, it is far simpler to compute the values of Γ_2 and substitute into equation 6.1.

6.3.3 World Coordinate Transformations

The Jacobian matrix, $\underline{J}$, is fundamental to both force and position control as shown by the relationships:

$$\underline{F_Q} = \underline{J}^T \underline{F_X} \qquad (6.3)$$

$$\text{and } d\underline{X} = \underline{J}\, d\underline{Q} \qquad (6.4)$$

Where $d\underline{X}$ is the incremental changes in the controlled Cartesian coordinate system, which are either in World or Tool coordinates, and $d\underline{Q}$ is the corresponding incremental changes at the joints. Force/torque vector $\underline{F_X}$, which acts at the end effector in the Cartesian frame, can be transformed to the equivalent force/torque vector $\underline{F_Q}$ acting at the joints.

The elements of the Jacobian matrix and its inverse must be evaluated in real time for successful robot control. This task is simplified if the Jacobian is analytically invertable and thus only simple parameter evaluation is required at run time.

Elements of the Jacobian matrix may be determined analytically by evaluating $d\underline{X}/d\underline{Q}$ for each element. The procedure expresses the elements of $\underline{X}$, as shown in equation 6.1, in terms of the components of $\underline{Q}$ and is differentiated with respect to $\underline{Q}$.

The Jacobian transformation into World coordinates from joint coordinates, expressed as ${}_W\underline{J}_Q$, may be written:

$$d\underline{X} = {}_W\underline{J}_Q \, d\underline{Q}$$

where ${}_W\underline{J}_Q = [a_{i,j}] = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{bmatrix}$, i,j = 1 to 3

and the elements $a_{i,j}$ are:

$a_{1,1} = \cos \Gamma_2/S$

$a_{1,2} = \cos \Gamma_1/S$

$a_{1,3} = -R \sin \theta$

$a_{2,1} = \sin \Gamma_2/S$

$a_{2,2} = \sin \Gamma_1/S$

$a_{2,3} = R \cos \theta$

$a_{3,1} = -l_1/(Dl_2 \cos \Gamma_2)$

$a_{3,2} = (l_1^2+l_2^2-D^2)/(2Dl_2^2.\cos \Gamma_2)$

$a_{3,3} = 1$

and where $S = \sin (\Gamma_1+\Gamma_2)$

It is extremely difficult, if not impossible, to determine the analytic inverse of ${}_W\underline{J}_Q$ in its present form. However, if a static force analysis is performed on the manipulator to obtain an expression of joint forces in terms of end effector Cartesian forces, then equation 6.3 may be applied, to give the Jacobian transpose. Using this latter technique to

evaluate ${}_{W}\underline{J}_{Q}$, the terms $a_{3,1}$ and $a_{3,2}$, which are in fact the most complex of all the terms, are found to be insignificant in comparison to other terms and may be assumed to be zero. This simplification allows analytical solution to be determined for $({}_{W}\underline{J}_{Q})^{-1}$, the inverse Jacobian, which may be written as,

$$({}_{W}\underline{J}_{Q})^{-1} = {}_{Q}\underline{J}_{W} = [b_{i,j}] \quad , \quad i,j = 1 \text{ to } 3$$

where

$$b_{1,1} = \sin \Gamma_1$$

$$b_{1,2} = \cos \Gamma_1$$

$$b_{1,3} = -R \cos(\Theta+\Gamma_1)$$

$$b_{2,1} = -\sin \Gamma_2$$

$$b_{2,2} = \cos \Gamma_2$$

$$b_{2,3} = R \cos (\Theta-\Gamma_2)$$

$$b_{3,1} = 0$$

$$b_{3,2} = 0$$

$$b_{3,3} = 1$$

The neglection of the two terms $a_{3,1}$ $a_{3,2}$ in the Jacobian can also be justified physically. These two terms express the coupling relationship between linear velocities of the ballscrews and the angular velocity of the terminal wrist. This coupling is relatively minor in comparison to the linear velocities of the ballscrews effect on the linear Cartesian velocities of the manipulator. These omissions are prevented from causing position errors by the outer position control loop, as described in section 6.6, which continuously corrects any possible angular or position errors. Force control is also unaffected, as these particular matrix elements are responsible for wrist torque control, which is unused in this particular application.

6.3.4 Tool Transformations

Owing to the physical task constraints of force control it is not generally possible to control force in World coordinates. Therefore force control is implemented in tool coordinates, which are aligned to the local coordinates of the task at hand. Hence a further Jacobian matrix is required for transforming between joint coordinates and tool coordinates. This transformation matrix was determined by simply post multiplying the ${}_{W}\underline{J}_{Q}$ transformation by the World to tool coordinate transformation. This latter transformation is evaluated directly from the geometry of the wrist as

$$d\underline{X}_{T} = {}_{T}\underline{J}_{W}\, d\underline{X}_{W}$$

where $\underline{X}_{T}$ is the Cartesian tool position coordinate vector (T,N,θ) and $\underline{X}_{W}$ is (X,Y,θ), the World position vector. ${}_{T}\underline{J}_{W}$ is the Jacobian transformation from World to Tool coordinates and is given by

$${}_{T}\underline{J}_{W} = \begin{bmatrix} \sin\theta & -\cos\theta & 0 \\ \cos\theta & \sin\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The inverse, ${}_{W}\underline{J}_{T}$ is simply the transpose of ${}_{T}\underline{J}_{W}$.

Therefore the Jacobian ${}_{T}\underline{J}_{Q}$ which transforms from joint velocities to tool velocities is determined from:

$${}_{T}\underline{J}_{Q} = {}_{T}\underline{J}_{W} \cdot {}_{W}\underline{J}_{Q}$$

Likewise the inverse is given by

$$({}_{T}\underline{J}_{Q})^{-1} = {}_{Q}\underline{J}_{T} = {}_{Q}\underline{J}_{W} \cdot {}_{W}\underline{J}_{T}$$

which utilizes the tool to World coordinate transformation.

The complete set of transformation matrices for velocity and force transformation, and their inverses, are given in the main controller program listings presented in appendix D. The transformation matrices reside within the subroutines `TMATRICES' and `WMATRICES' within the Fortran program `MULTSAMP.'

6.4 Mechanical Design

6.4.1 Experimental Facility Outline

The photograph in figure 6.2 shows the experimental manipulator complete with controlling electronics, trajectory planning computer and controller computer. Major components are itemized as shown.

The manipulator base is a rigid steel frame (item 1) which is secured rigidly to earth by being bolted to a pedestal (item 2), itself bolted to earth. Twin ballscrew drive transmissions (item 3) are supported by their nuts in gimbals (item 4) which pivot freely in the supporting frame. One end of each ballscrew terminates at the pivot joint of the serial wrist (item 5). Ballscrew actuation is provided by DC electric motors (item 6) mounted on, and moving with, each gimbal. Wrist actuation is provided by a single DC electric motor (item 7) mounted axially with the pivot joint. Manipulator joint positions are measured using digital encoders (items 8). Lateral vibrations, thought insignificant at the design stage, are damped using a vertically mounted gas shock damper (item 9). The free end of the damper is fitted with a ball castor, enabling it to move freely over the floor.

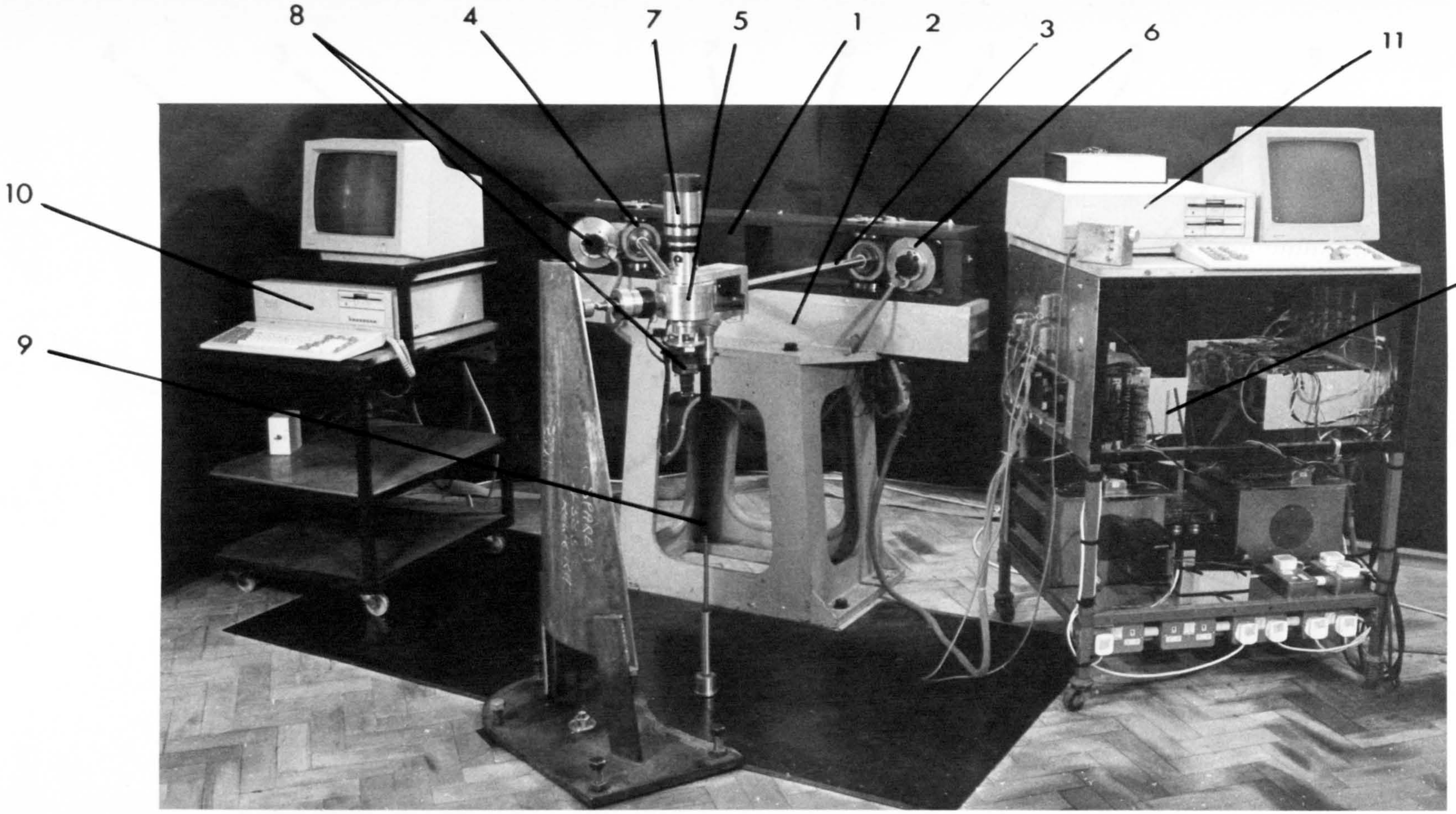


Figure 6.2 Complete experimental facility, showing computers, manipulator and controller.

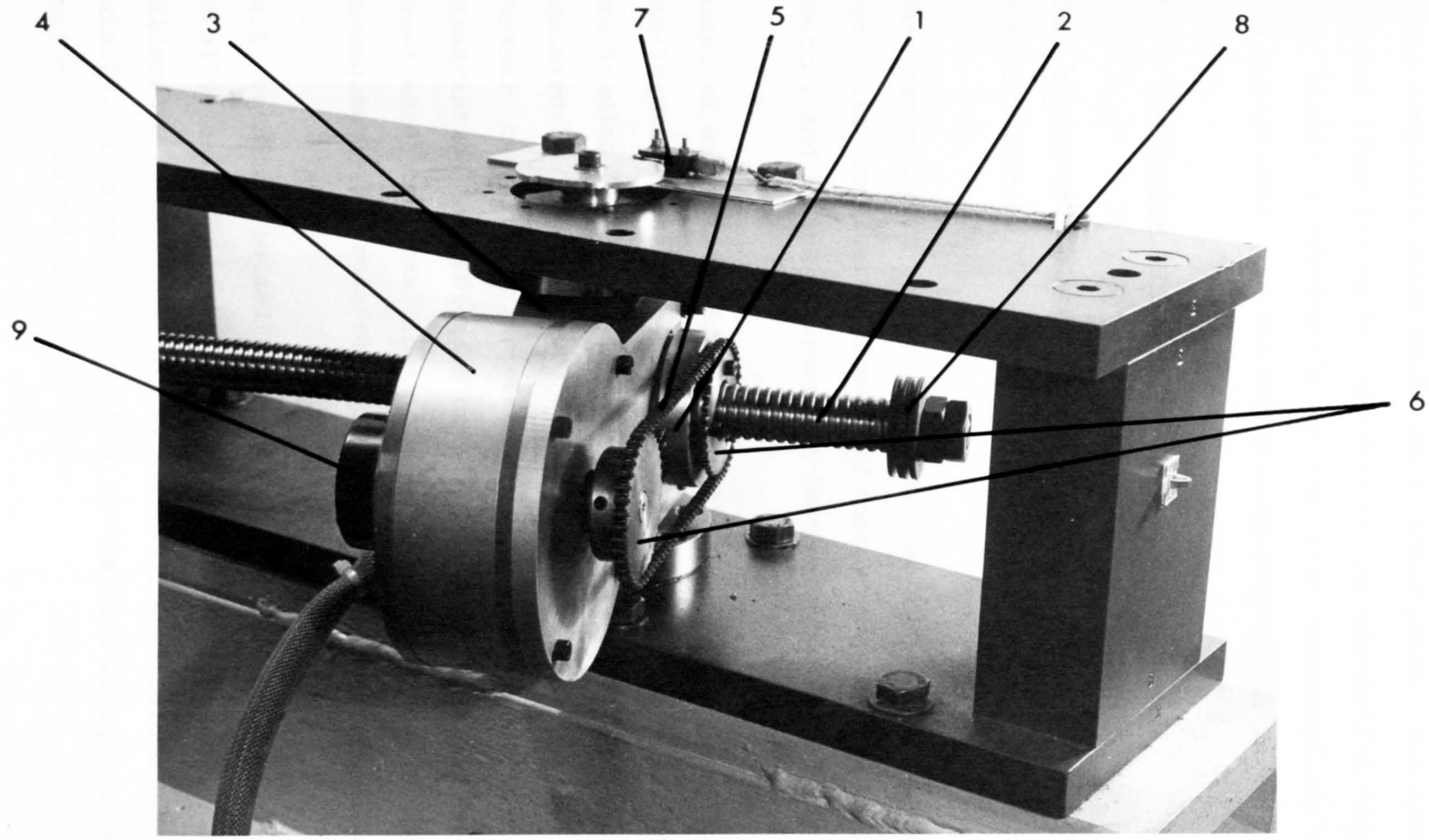


Figure 6.3. Ballscrew actuation system.

Twin computer control is achieved by employing two high speed Sperry micro computers (high speed IBM PC compatibles). One acts as trajectory planner (item 10) while the other is the controlling computer (item 11). The latter is responsible for low level communications and control of the manipulator, via the controlling electronics assembly (item 12).

6.4.2 Ballscrew Actuator

The photograph given in figure 6.3 shows details of the ballscrew actuator system.

The manipulator transmissions utilize ballscrews rather than linear hydraulic actuators because of their mechanical simplicity, ease of control and a high accuracy and load potential.

Backlash was eliminated in the ballnut assembly by employing two rigidly pre-loaded nuts (item 1) on each ballscrew (item 2). This form of nut pre-loading is only possible if high precision ground ballscrews are employed. The adoption of high precision ballscrews resulted in a further benefit of improved ballscrew accuracy, while incurring the disadvantage of increased cost. A key machined into the sides of the pre-loaded ballnuts locks them together without further adjustment. Consequently the nuts revolve as one around the ballscrew thread.

Ball bearings, housed in the gimbals (item 3), constrain the ballnuts axially allowing only simple rotation around the ballscrews. The ballscrews themselves are rotationally constrained but free to move axially. Thus the rotary motion of the ballnut is converted to a linear motion of the ballscrew.

The rotary motion of the DC electric motors (item 4) is transmitted to

the ballnuts via a tensioned high precision instrumentation chain (item 5). Tensioning is used to eliminate backlash in the chain. The chain has a fine pitch which eliminates cogging and both the motor and ballnut are fitted with sprockets (item 6) to allow smooth transmission.

The DC electric motors (details given in appendix B1) which drive the ballscrews are of printed armature construction. The armature resembles a printed circuit board, with copper tracks laid into a non-metallic thin disc. The large diameter radial construction offers low inertia, high torque and strength, ideal for high accelerations and high speed operations. The large radial commutator has a large number of segments, allowing controlled ripple free torque down to zero speed. Printed motor technology also has the added advantage of cheapness, more than an order of magnitude cheaper than an equivalent specification torque motor.

Angular swing of the ballscrews must not be excessive if mechanical damage is to be avoided. Thus, limit switches (item 7) have been added to monitor the swing of the ballscrews. These switches, if activated, inhibit the whole servo electrical system preventing all servo motor current supply. However, it is still possible to extend the reach axis of the manipulator to its limit without encountering the angular limit switches. This occurrence will not cause mechanical damage providing the ballscrews remain captive within the ballnuts. Stacked disc springs (item 8) mounted securely at the ballscrew ends, act as shock absorbers, preventing the ballscrews from leaving the ballnuts even under high speed crash conditions. In addition to these mechanical/electrical devices preventing undesirable manipulator configurations, there is comprehensive software monitoring of the geometry, inhibiting all motor current if a geometric limit is exceeded. Therefore, in common with most industrial manipulators, two independent levels of geometrical

protection are operative.

Digital encoders (item 9) are directly mounted to the free end of the motor shaft, measuring motor movement directly. These are a light emitting diode optical incremental encoder of low power consumption. They remain active continuously and have a battery backed up power supply. Thus they act as precision absolute encoders, at a fraction of the cost, and only require a single calibration at commissioning. A comprehensive specification of the encoders is given in section 6.5.6 and in appendix C.

6.4.3 Wrist Actuation

The photograph in figure 6.4 shows the details of the serial wrist drive arrangement. The wrist (item 1) has an integral compliant force sensor, which is detailed in section 6.5.5. Tooling can easily be attached to the wrist by the threaded portion at the wrist end and is shown here with the ball castor (item 2) attached. All the major wrist components have been machined from HE 30, a light but strong machinable aluminium alloy.

For the reasons given in section 6.2 all rotation associated with the wrist structure takes place about the common pivot axis. The wrist is attached to a cylindrical drum (item 3) which is driven internally by the Harmonic gearbox (see appendix B3 for details) to rotate the wrist about the pivot axis. The gearbox itself reacts against the end face of the drum which forms part of a fork assembly (item 4) attached to the ballscrew. This arrangement allows the ballscrew to pivot about the same pivot axis as the wrist. The second ballscrew is attached to an outer fork (item 5). This fork encloses the inner fork and is completely free to rotate about the pivot axis.

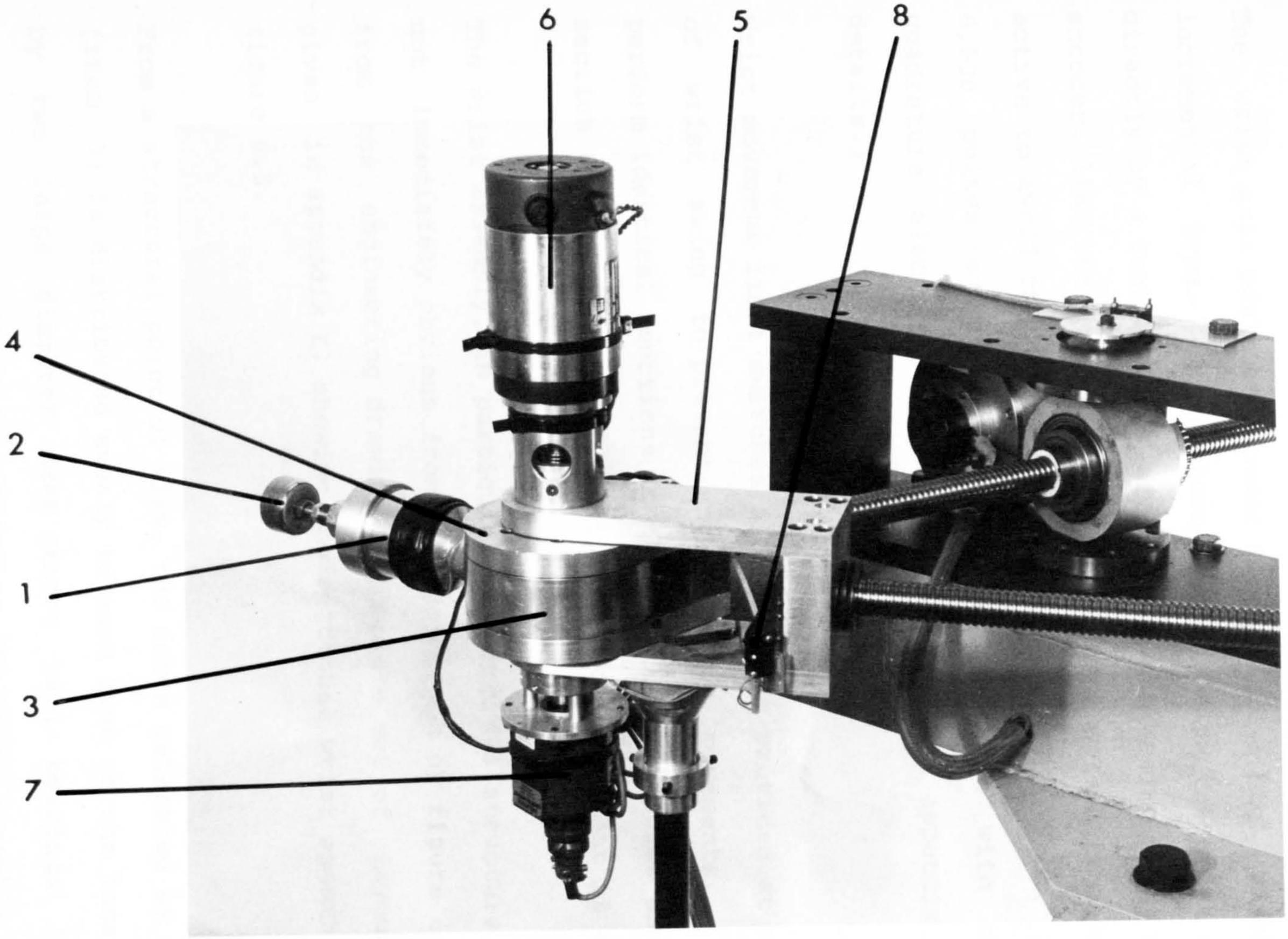


Figure 6.4 Wrist actuation system.

The DC electric servo motor (item 6) is of conventional copper wound armature construction (see appendix B2 for details). This motor drives the Harmonic gearbox directly via a bellows coupling attached to the central gearbox input shaft.

The wrist axis position encoder (item 7) is a high resolution optical incremental type. It is shown mounted below the wrist and is driven directly by a central shaft attached to the moving wrist assembly. This encoder, like the others, is battery backed up and remains continuously active to avoid the need for continual re-calibrations. A resolution of 6,000 pulses/rev is achieved in conjunction with the associated quadrature electronics, (see section 6.5.5 and appendix C for further details.)

Wrist movement limit switches (items 8) are provided at the extremities of wrist swing to prevent any damaging movements. These switches perform identical functions to those on the ballscrew axes, described in section 6.4.2, by inhibiting all servo motor current if activated.

The wrist assembly, in particular its internal structure, is complex and not immediately obvious from the photograph of figure 6.3. An extract from the engineering drawing (a complete set of layout drawings are given in appendix E) showing the sectional wrist assembly, is given in figure 6.5.

From a structural point of view, the force generated by the force sensor (item 1) is distributed evenly to each limb of the inner fork (item 2) by two large diameter deep groove ball bearings (items 3). The cylindrical drum (item 4) supporting the wrist assembly can thus be seen to act as a hollow shaft containing the Harmonic gearbox (item 5). The

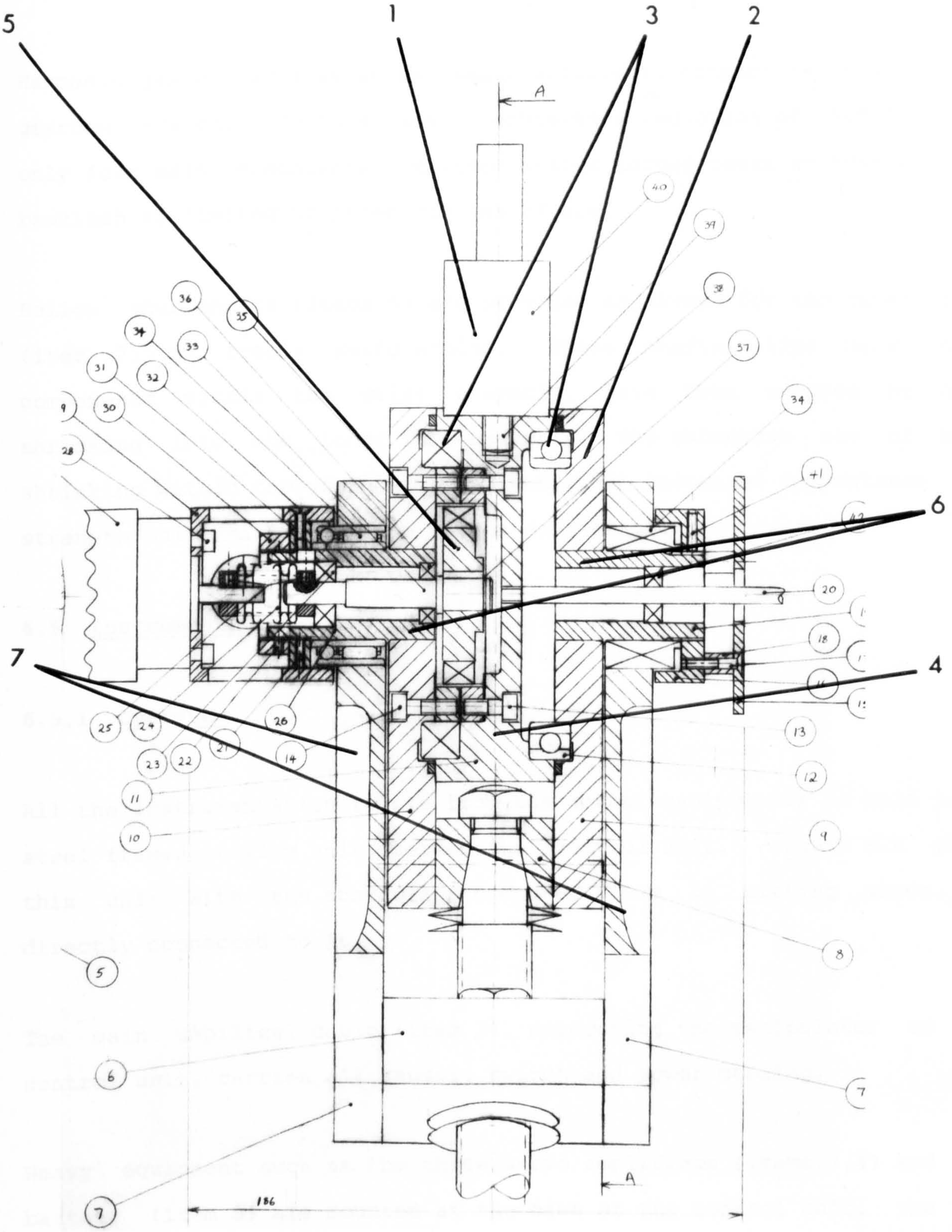


Figure 6.5 Details of wrist assembly.

Harmonic gearbox acts as an extremely efficient, compact and high torque gearbox system. In this case it achieves a reduction of 160:1 using only four major components. Maximum output torque peeks at 90Nm and the backlash is limited to three minutes of arc.

Hollow stub shafts (items 6) are provided as pivots for the outer fork (item 7) to freely swing around. These shafts, like many other components within the wrist assembly, have been secured by heat shrinking into the inner fork assembly. The extensive use of heat shrinking within the whole wrist assembly has increased compactness and strength while minimizing the use of fastenings.

6.5 Instrumentation

6.5.1 Overview

All the instrumentation remote from the actual manipulator is held in a steel framed trolley unit, termed the `control unit.' Figure 6.6 shows this unit with the controller computer (item 1) sitting above and directly connected to it.

The main umbilical cable (item 2), connecting the manipulator to the control unit, carries all sensor, switch and power cabling.

Heavy equipment such as the three servo amplifiers (items 3,4) and the battery (item 5) are mounted at the base of the control unit. The two servo amplifiers (items 3) provide power for the ballscrew axes, while the smaller central servo amplifier (item 4) controls the serial wrist axis. The standby battery and its trickle charging power supply (item 6) provide continuous power for the position encoders.

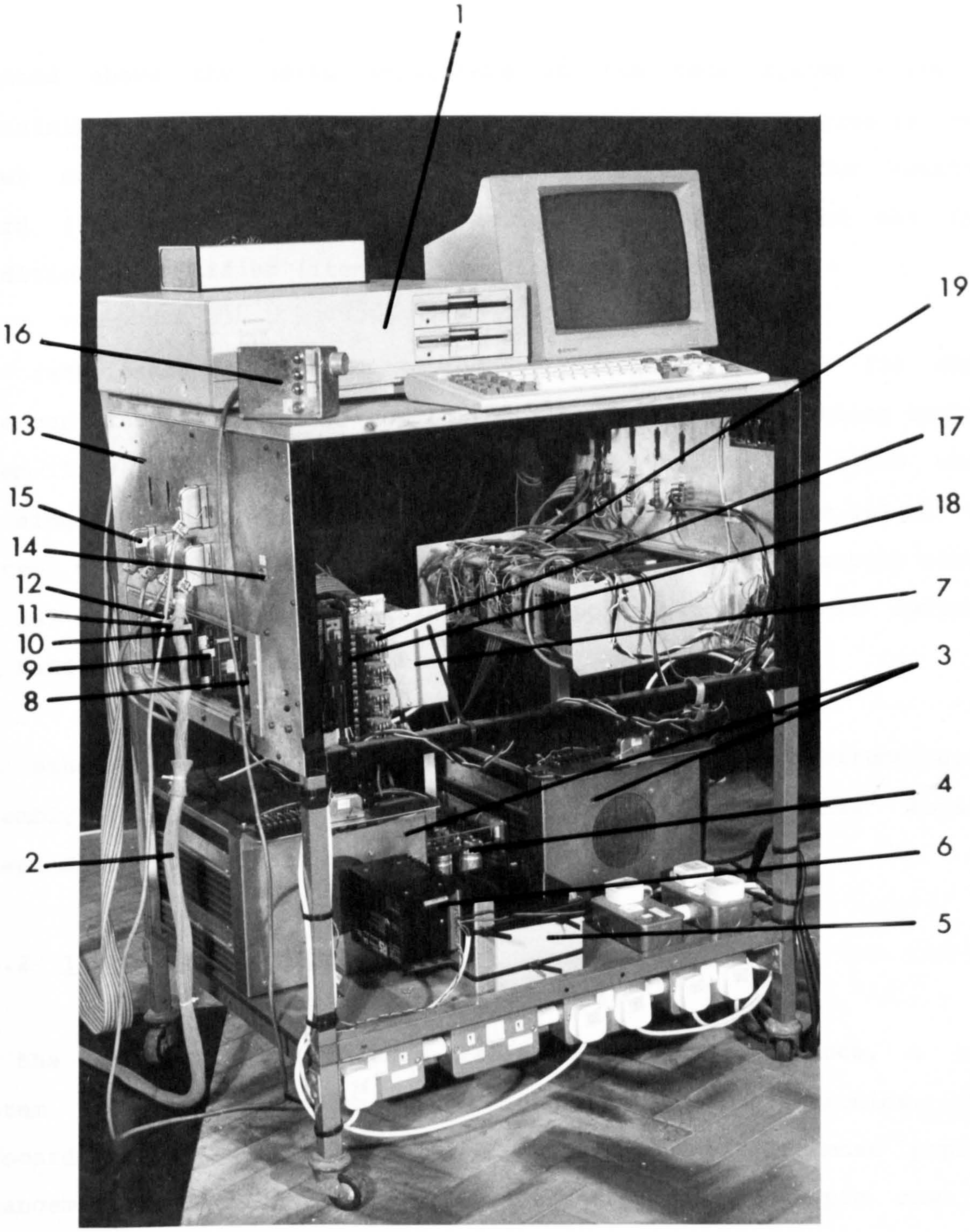


Figure 6.6 The manipulator control unit.

Mounted above the servo amplifiers is the rack system (item 7) containing the interface circuitry. The rack contains: plug-in rack power supplies (item 8), power supply board (item 9), encoder counting board (item 10), analogue interfacing board (item 11) and the LVDT conditioning amplifier (item 12) used for the force sensor.

The rack itself is mounted into the front facia panel (item 13) which also contains an emergency stop button (item 14) and the cabling sockets (item 15). The four lower sockets connect to the umbilical cable while the single upper socket connects the hand held pendant (item 16) to the control unit. Alongside the rack are the relays and power supply (items 17, 18) responsible for monitoring and responding to the limit switches and hand held pendant conditions.

The other rack (item 19), mounted at the rear of the instrumentation assembly, is used for another research project and is totally independent of this work.

6.5.2 Instrumentation Rack

In the interest of convenience, reliability and maintenance, a rack system is employed to house the instrumentation electronics. The Eurocard standard rack was selected as it is one of the most popular arrangements. Rack backplane connections are standard 64 pin row DIN 41612. These backplane connectors are crimped to 64 way ribbon cable and mounted to the rear of the rack. The power pin connections are to Eurocard standard, compatible with the rack power supplies. All the boards within the rack system are fitted with compatible 64 way edge connector plugs thus allowing simple insertion and removal. All the boards adopt common power connections, but their individual signal connections vary.

6.5.3 Rack Power Supplies

There are three independent rack supplies providing ±15V, 5V and battery backed 5V. The ±15V and 5V are provided by two mains powered supplies. These power supplies, by Bulgin, are of modular construction and plug directly into the Eurocard rack. The pin-out details of the rack power rails are given in appendix C1. The battery backed 5V power supply, provided solely for use by the position encoder electronics, was achieved by stepping the voltage down from the 12V lead acid battery to 5V. An integrated voltage regulator type LM338K was used to reduce this voltage. The regulator and associated control components were mounted on the power board of the instrumentation rack. The sealed lead acid battery is continuously charged by mains charger which has controlled output currents specifically designed for sealed lead acid batteries.

Visual indication of the four independent voltage levels of the rack is provided by light emitting diodes (LED's). The LED's are located at the outer edge of the power card, the same as that used for battery voltage regulation. The complete circuit for this power board is provided in appendix C2.

6.5.4 Limit Switches and Hand Held Pendant

Provision of the hand held pendent gives the operator absolute control over the manipulator. The pendent is uncomplicated in operation. Three switches are provided to independently enable or disable any axis of the manipulator. A red emergency stop button is provided on the pendent face alongside the three switches. Once depressed the emergency stop disables completely all the manipulator axes. The operation of the

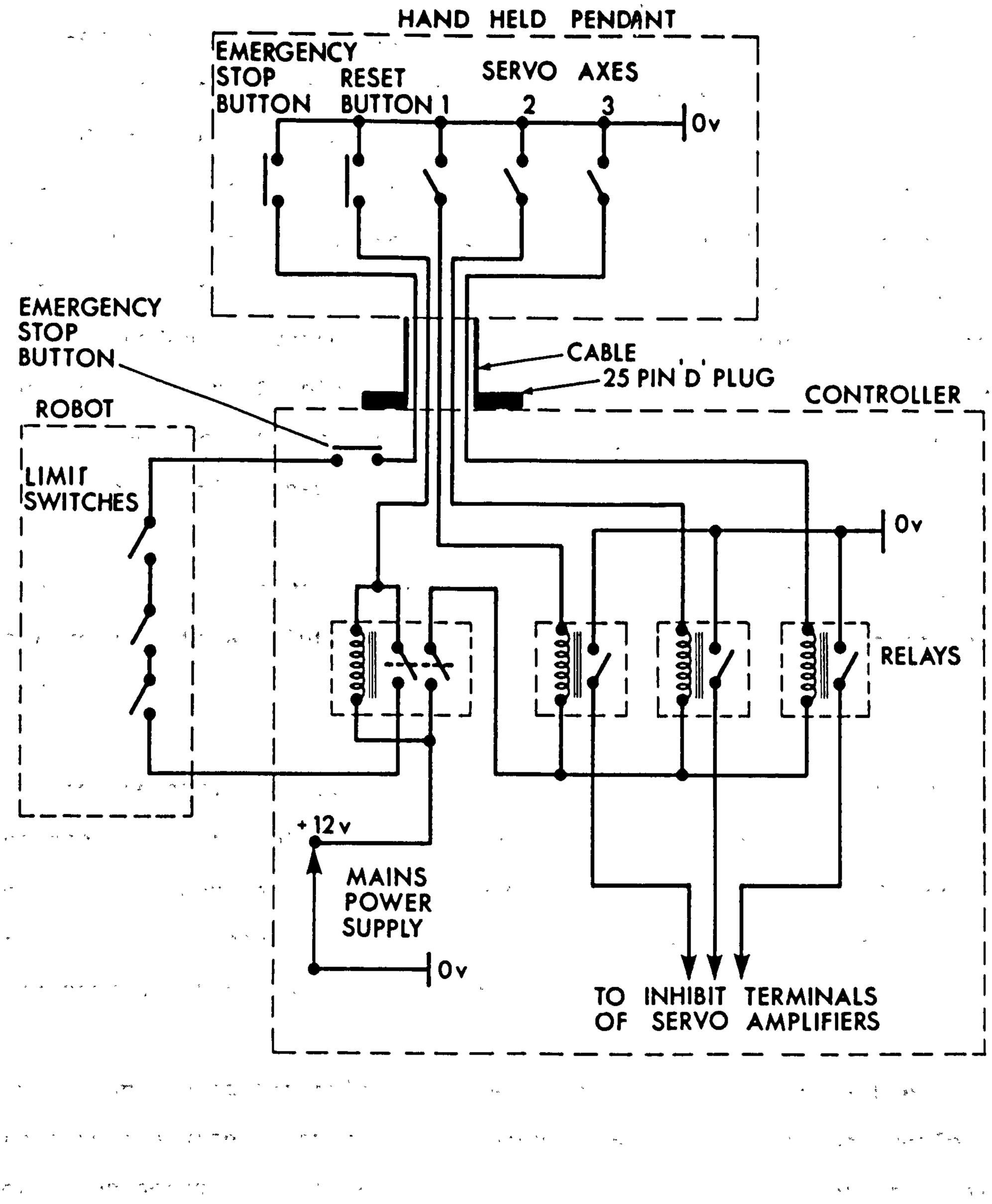


Figure 6.7 Wiring arrangement for limit switches and hand held pendent.

emergency stop button can be cancelled or reset by pressing the green reset button mounted on the end face of the pendent, shrouded and placed out of accidental reach. A further emergency stop button is provided on the facia of the controller unit. The geometrical limit switches described earlier in section 6.4 are identical in action to the emergency stop button's action, as all limit switches and emergency stop buttons are connected in series. This is indicated in figure 6.7, showing the wiring arrangements for the limit switches and pendent. The hand held pendent is connected to the controller unit facia by a 3 metre cable and a 25 pin `D' plug. As a further precaution, when the pendent is unplugged from the controller unit all manipulator operations are inhibited preventing unauthorized operation.

Figure 6.7 shows the wiring arrangement which is comprised of a single master relay with absolute control over the operation of the other three relays.

The three relays, controlled by the three pendent switches, allow for independent control of the servo amplifiers by switching their inhibit lines. Thus any axis of the robot can be simply disabled by the operation of a pendent switch. This facility was particularly useful during commissioning the manipulator.

On power-up the master relay must be first reset, this is achieved by depressing the green button on the pendent. Resetting must be performed before the servo amplifiers can be uninhibited and motion of the axes commence. Apart from this fail safe feature, the same mechanism protects if there is an unknown loss of limit switch power. Immediately there is a loss of power, the master relay opens and behaviour is exactly that of an emergency stop condition.

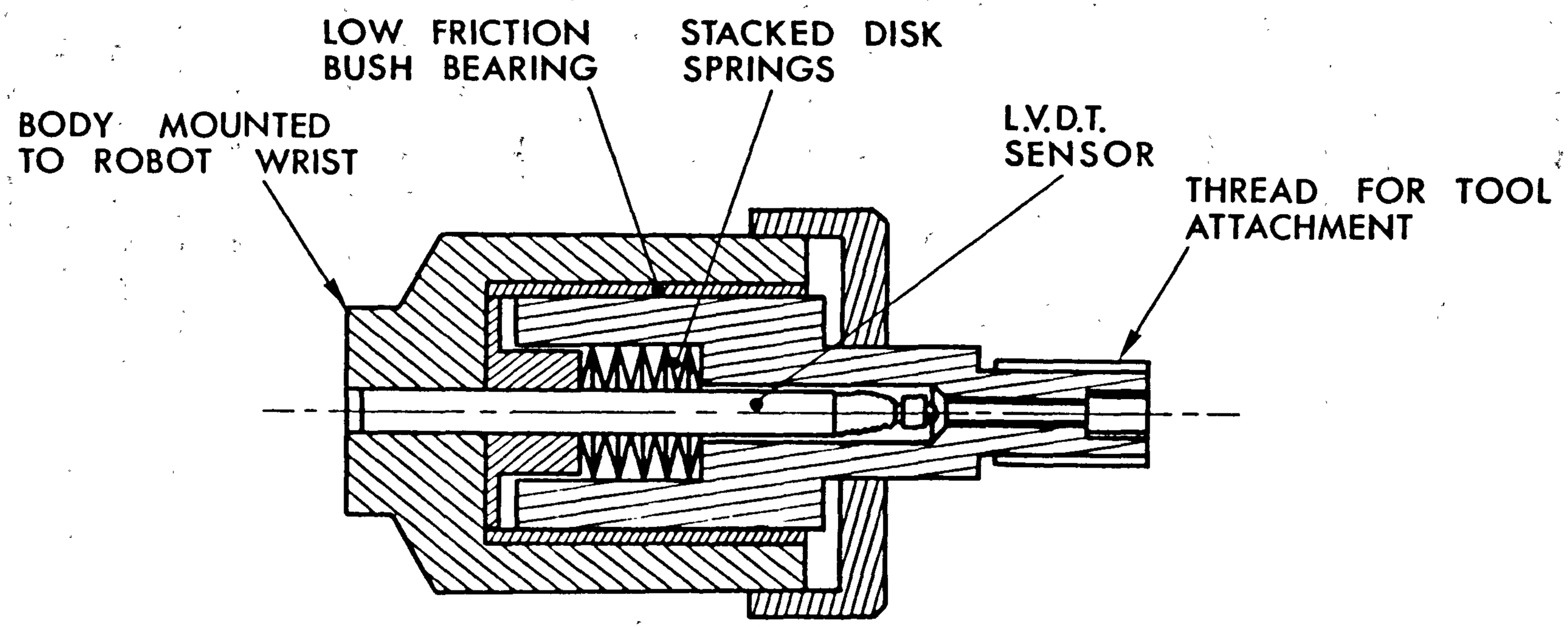


Figure 6.8 Force sensor features.

6.5.5 Force Sensing Wrist

The requirements of the force sensor are discussed in the Problem Analysis, chapter 3. Basically the requirements are to provide a single axial degree of freedom with a force sensing capability, while the remaining degrees of freedom are resistant to all force disturbances. With the planar 3 axes experimental manipulator this requirement is reduced to a single force measuring axis and to rigidity on the remaining two axes, all in one plane. However, in the event, it was simpler to produce a general force sensor with a single axis force sensing axis and five rigid axes. The sensor is designed to measure compressive forces only, this is not a limitation as the forces involved in this form of contact are themselves compressive. The force sensing axis has necessarily been designed to be compliant, where the compliance is adjustable by varying the spring stiffness. A detailed layout drawing of the force sensor is included in appendix E but the salient features are shown in figure 6.8 in schematic form. The force sensor, which forms the wrist structure of the robot, is bolted internally by its outer casing to the third axis of the robot. Tooling is attached to the moving element of the sensor by the threaded portion at its end. Hence the forces acting on the tooling are measured by the sensor, neglecting dynamic effects.

The moving element is guided by a close fitting sintered PTFE bush. The bush was chosen for its high load capacity and low friction properties. However, in practice, the bush displayed a significant viscous friction component. Steps are currently being taken to redesign the sensor to replace the bush bearing with a non-friction element.

Force Measurement

Force is measured by utilizing the compliance of the sensor. Axial deflection of the sensor is measured directly by an LVDT (linear variable differential transformer) mounted in the bore of the sensor. From a knowledge of the spring stiffness, the sensor force is calculated in software by the controlling program. Mechanical adjustment is provided for the LVDT by an axial set screw contained within the threaded portion of the sensor end.

The LVDT used is a Sangamo Gauging Transducer model AG/2.5. The 2.5 signifies ±2.5mm of mechanical movement. The conditioning amplifier used in conjunction with the LVDT sensor is a Sangamo CA series card. This card is of Eurocard size and is held within the controller instrumentation rack and takes its power from the backplane. Specification details of the LVDT and conditioning amplifier are given in appendix C3.

The LVDT is connected to the controller via a screened lead to a 25 pin `D' plug mounted on the controller unit facia. Given in appendix C4 are the 25 pin `D'plug connection details.

The diagram showing the electrical arrangement for the LVDT sensor is given in appendix C3. This shows the LVDT sensor connected to the conditioning amplifier by four wires, two for the primary circuit and two carrying the modulated signal from the secondary coils. Demodulation is then carried out by the amplifier to produce an analogue DC voltage ranging from -5V to +5V, according to transducer position.

Setting up the LVDT to measure force correctly requires a sequence of procedures given below.

i) Since the maximum movement of the force sensor is 5mm, the LVDT is mechanically adjusted to the -2.5mm position corresponding to zero force condition.

ii) The conditioning amplifier electrical offset is then adjusted for 100% offset, giving zero volts for zero force, termed an 'end zero' for the LVDT. So, instead of the usual centre zero ± 5V, a voltage swing between 0V and 10V is achieved for an end zero setting.

iii) Finally the gain of the amplifier is adjusted so that a 10V signal corresponds to the maximum 5mm deflection.

Unfortunately the analogue to digital conversion card is arranged to measure full 12 bit accuracy between -10V and +10V. Therefore the actual accuracy of the LVDT sensor is half the maximum, corresponding to 11 bit.

During the commissioning stage, ripple associated with the conditioning amplifier's modulation frequency of 5KHz was found on the DC output signal from the amplifier. To avoid the possibility of aliasing problems at the A to D stage, a first order passive filter was arranged at the amplifier output with cut-off frequency of 1KHz. This frequency is well above the 20Hz maximum response frequency of the manipulator and from a control point of view is ignored.

6.5.6 Position Encoders

All the encoders used for position measurements are of an optical incremental type. Two different models are used; a RENCO R80 for each of the ballscrew axes, a RENCO HDR70 for the wrist axis. Appendix C5 gives the technical specifications of the encoders. Sections 6.4.2 and

6.4.3 describe the location and function of the ballscrew and wrist encoders respectively. The wrist encoder is sited near to the tooling and workpiece and consequently there is a higher risk of incurring damage. For this reason the heavy duty, industrial encoder HDR70 was fitted.

The encoders fitted to the ballscrew axes have a resolution of 250 pulses per revolution. Since the pitch of the ballscrews is 5mm, this encoder resolution gives a measured precision of 0.02mm to the ballscrew axial movement. The wrist encoder produces 1500 pulses per revolution and is directly coupled to the moving wrist without gearing. Quadrature is used in this case for pulse counting giving an effective resolution of 6000 pulses per wrist revolution or 3.6 minutes of arc. This is comparable to the backlash limit of 3 minutes of arc for the wrist Harmonic drive.

All the encoders are wired to a single 25 pin `D' connector which plugs into the controlling unit facia. The pin arrangement of the plug is given in appendix C4. Each encoder is TTL compatible and produces two phase shifted square wave outputs which switch between the 0V and 5V levels.

Counting these encoder pulses is achieved by an integrated electronic device which receives the phased signals directly from the encoder. These sophisticated electronic devices are available on a single 28 pin chip by Texas Instruments, a SN74LS2000, see appendix C6 and [72] for details. Three of these devices, one for each encoder, have been mounted on a single card which is held in the controller rack. The circuit diagram of the complete encoder interface card is given in appendix C6.

These chips take the two phased signals from the encoders and by identifying the shift are able to count pulses up or down. The 16 bit counter is continuously updated. But if a count reading is required by the computer, the count is passed to the 8 bit output register as upper and lower bytes which must be sequentially accessed using computer instructions. A machine code routine has been written to communicate with these devices and transfer their counter readings to the high level Fortran program. See appendix D3 for complete listings of the machine code assembler programs. The communication with these chips and the computer is achieved using an additional parallel input/output (I/O) card plugged into the controlling computer. The control of this card is described in more detail in section 6.5.8.

All connections and communications to the board take place through its 64 pin end connector which plugs directly into the rack backplane bus. These connections include: power, input encoder signals, digital control and data lines from the computer. Various options are available to the counter chips, these are selected by the DIL switches in accordance with the chip operating instructions. In this case the switches are set to give quadrature for the wrist encoder counter and single counting for the ballscrew encoder counters. Each chip requires a clock pulse at frequency of at least four times the maximum count frequency, this allows for quadrature counting. A single pulse generating device by Texas Instruments, SN74LS625 [72] was fitted to the board and controlled to oscillate at 8 MHz. This frequency gave a reliable square wave and was in excess of the maximum required frequency. All three counting devices received their clock pulses from this single device.

6.5.7 Servo Amplifiers

Three DC analogue servo amplifiers are used to power the three servo axes. Two different types of amplifier are used. For the printed armature DC motors driving the ballscrew axes, the McLellan amplifier EM200 was used. For the wrist axes, an Electro-Craft LA 5400 was used. The technical specifications for these amplifiers are given in appendix C7. It is possible to achieve high speed from the printed armature DC motors if they are given high voltages. For this reason the EM 200 was chosen, capable of delivering ±50V at 9 amps continuous rating. These amplifiers also provided a 30 amps peak voltage which is thermally controlled to fold back to 9 amps after one second. The gain of these amplifiers was set at 50V/V. The cheaper and less powerful wrist axis servo amplifier, LA5400, operates in a current controlled mode with a gain of 6A/V. Terminal resistance of the wrist servo motor was specified as 0.7Ω, giving an effective amplifier gain of 4.2V/V under stalled conditions. The maximum terminal voltage of this amplifier was 44V, but under loaded condition (9 amps continuous) this voltage falls back to 35V. All the amplifiers have in-built control electronics and feedback loops. These facilities allow for use in stand alone applications where the servo amplifier can be arranged to perform the additional function of motor controller. All these features were totally disconnected or unused in this work. Thus the amplifiers were configured to function purely as power amplifiers.

Each amplifier was controlled by an individual DAC (digital to analogue converter) output from the controlling computer. To protect the DAC card against damage and to provide signal drivers for the servo amplifier inputs, three analogue amplifier circuits were arranged. The three circuits, which are completely independent of each other, are arranged on a single card which plugs into the Eurocard rack system.

The circuits which are composed of dual operational amplifiers, for double inversion, have their output voltages protected by two serially connected zenner diodes. This limits the output voltages to ±10. Gains are adjusted by potentiometers. Circuit diagrams for the analogue circuits are given in appendix C8.

The DAC devices have 12 bit accuracy and produce voltage swings of ±10V. Therefore to benefit from the highest resolution, the DAC signals serving the EM200 amplifiers were attenuated by a factor of 10 at their analogue amplifier buffers . This produced the maximum servo amplifier voltage for the maximum DAC voltage swing. This procedure was unnecessary for the LA5400 servo amplifier where the analogue buffer gain was set to unity.

6.5.8 Computers and Interfacing Cards

The two computers used to control the manipulator are Sperry PC's. They are both IBM-PC compatible but are switched to operate at a higher clock speed of 7.16 MHz. The improved clock speed improves the real-time computational capacity by 50% over the standard IBM-PC. Both computers used here are fitted with the mathematical co-processor, the INTEL 8087, which typically improves computational speed by 30 times over the single standard 8088 processor alone.

The machines have standard PC expansion slots which enable up to four additional cards to be plugged into the main processor board. The expandability, flexibility and standardization were the primary reasons for choosing the IBM-PC standard for this task.

Three different interface plug-in cards were used: the Tecmar Lab Master TM40, the Tecmar Dadio D to A and the Tecmar Baseboard.

Analogue to Digital Conversion

The Lab Master card has many features and options, see specification sheet in appendix C9, but here it is used solely for the task of analogue to digital conversion (A/D). The card has the features: 16 multiplexed A/D channels, 12 bit resolution and a conversion rate of 40 KH_Z, and is used to simply digitize a single analogue signal representing the force measured at the force sensor. The A/D converter reads voltages within the range -10v to +10v, converting them to a 2's complement integer in the range of -2048 to 2047 which is passed to the calling Fortran program.

The Lab Master consists of two basic components, the mother board which resides within the computer and the daughter board which is external to the computer. The daughter board, which has its own enclosure, is designed to be placed near to the analogue signal sources, remote from the computer and so limiting the effects of noise. The daughter board contains all the analogue acquisition circuitry including the multiplexer and analogue to digital converter. The digitized signal is returned via a ribbon cable to the mother board, immune to noise.

Sampling the analogue force signal is achieved using the machine code routine IADCON(channel), which is arranged to behave as a function call from Fortran. Section 6.6 describes the software requirements for the ADC's operation within the main controlling Fortran program. Suffice to say that the function, IADCON, returns an integer value between the ranges -2048 to 2047, depending on the force sensor's displacement.

Digital to Analogue Conversion

Two digital to analogue conversion (DAC) channels are available on the Lab Master board, but these were insufficient for this work which requires three. Therefore an additional DAC board was used; the Tecmar DAD10 board. This board is equipped with four independent channels of DAC, the details of the board's specification and connections are given in appendix C9.

Each DAC has 12 bit accuracy and can operate at 200 KH_Z, with latched voltage output. The previous section, 6.5.7, described how the DAC voltage outputs were first buffered and adjusted by analogue amplifiers before going on to control the servo amplifiers.

The DAC's are operated from the Fortran program by using the subroutine VOLTOUT which in turn uses the input/output port command OUT(port address, integer). The integer takes values between 0 and 4095 giving corresponding voltage swings between -10v and +10v.

Parallel Input/Output

Extensive use of the digital input/output (I/O) lines was made for both the parallel communication arranged between the two computers and for reading the position encoder counters. The computer's I/O lines were provided by the Tecmar Baseboard card. The device on the card which provides the I/O facility is the Intel 8255 chip. Each chip provides 24 lines of programmable I/O lines and the card is fitted with four devices, giving 96 I/O lines per card.

Two Baseboards have been used, one sited in the controller discussed above, the other situated in the trajectory planning computer. The

latter is used solely for the inter-computer communication. The 8255's are programmable devices which can be conveniently divided into three ports A,B and C, each 8 bits wide.

Position Encoder Counters

To read the encoder counters two counters were connected to the No.1 8255, while the remaining counter was connected to the No.3. Each 8255 on the Baseboard has adjacent to it a 40 pin ribbon connector these were used to directly connect the encoder counters and second computer. Appendix C9 shows a diagram of the Baseboard and its connector arrangements.

Each 40 way ribbon cable from the 8255's is connected to the 64 way socket of the encoder counter board in accordance with the pin connections given in appendix C6. The 8255 chips are programmed for their A and B ports to be inputs and take the 8 bit data from each counter. Port C is arranged to connect to two lines on each counter, one to control chip select and read, the other to the A/O line which controls the selection of the upper and lower byte of the 16 bit register. 'PORTIN' is the machine code routine which interfaces the main Fortran program to the counter's requirements. Its implementation is described in the following section 6.6 and is listed in appendix D3. Free running speed trials on PORTIN show that it is able to sample three 16 bit encoder counts at a rate of 18 KH_Z.

Parallel Communication

Arranging two computers to share the computational task was made possible by implementing the high speed communication of key parameters within the controlling task. This speed requirement excludes serial

communication in favour of faster parallel operation. The system finally adopted transfers single precision real numbers at a rate of 12 KH_z. Such real numbers require 4 bytes to define, 3 bytes for the mantissa and a single byte for the exponent. Using the adopted 16 bit ribbon cable to transfer data requires the transfer of two 16 bit words to transmit a single real number.

The wiring arrangement connecting the computers is a single 40 way ribbon between the No. 2 8255 chips of each Baseboard. Bi-directional data flow is made possible by running the appropriate routines on the trajectory planner and controller computers. If a real number is to be transferred out, then the machine code routine `WRDOUT' (word out) is used, while the corresponding computer must use `WRDIN' (word in). This is an absolute rule otherwise the programs will stop and control will be lost. Bi-directional data flow is possible because each time the routines `WRDOUT' and `WRDIN' are run, the A and B ports responsible for data transfer are reprogrammed as outputs or inputs respectively. The control over the data flow or `handshaking' is controlled by the C port which sends and acknowledges signals depending on the status of the data transfer. Further details on the handshake control are given in the assembler code listings in Appendix D3. Data transfer will `jam' or stop if there are delays or miss matches in the handshake status. Therefore it is extremely important that `WRDOUT' and `WRDIN' routines are executed in the correct sequence on each respective computer. This restriction effectively locks or synchronizes the two computer programs together, making the task of dual control easier to supervise. However, great care must be taken as to when and where data transfer occurs.

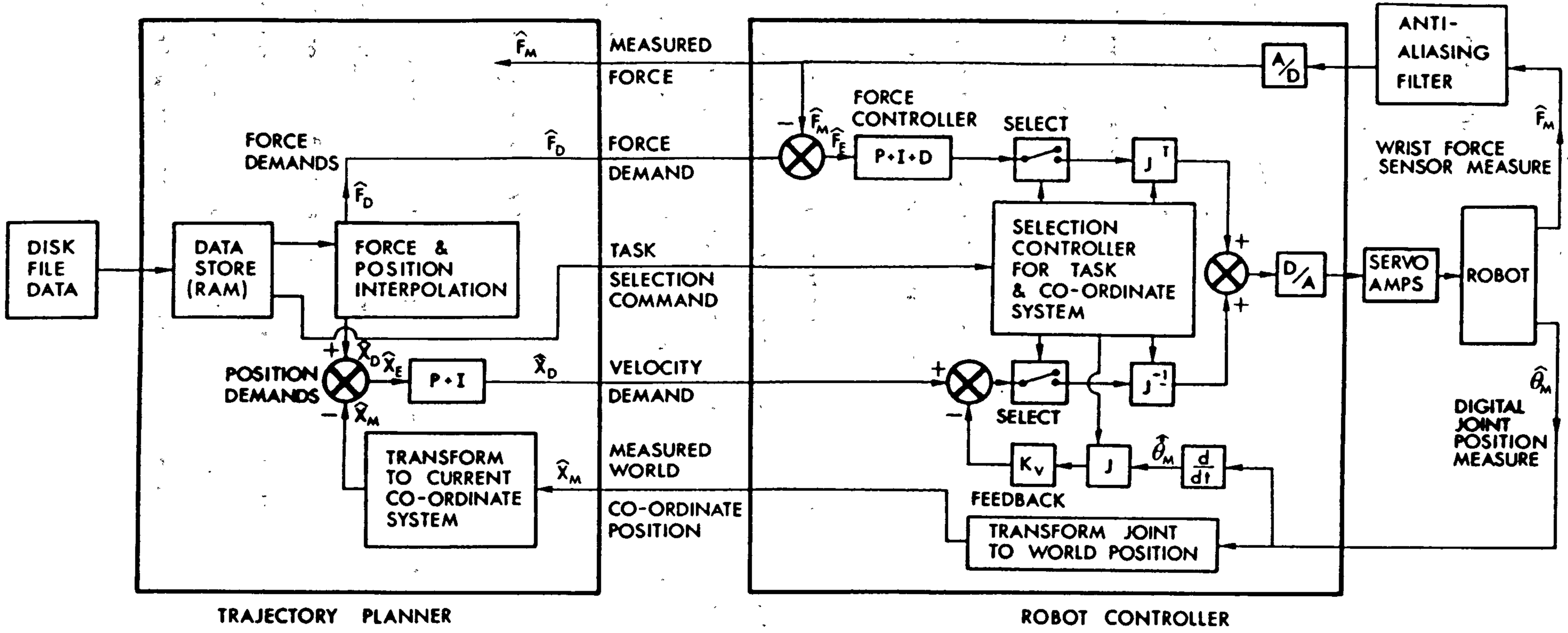


Figure 6.9 Schematic arrangement of the twin computer control system.

6.6 Control Algorithms

6.6.1 Introduction

The basic requirements of a manipulator controller can be summarized as follows.

1) To interpret the motion instructions from a pre-programmed task based language. Process the data before and/or during the task.
2) Directly control the motion or forces of the manipulator.
3) Communicate with low level sensory information and control accordingly.

In this work, task 1 is achieved by the Trajectory Planner computer. Tasks 2 and 3 are performed by the second or Controller computer.

The twin computer operation is indicated schematically in figure 6.9. This diagram shows the physical boundaries of the computers within which the controlling block diagrams are drawn. Communication between the computers is indicated, as well as that to the user's data file and to the robot.

6.6.2 Trajectory Planner

The trajectory planning program is called MULTIN. The listing is given in appendix D1 and the flow diagram in figure 6.10. It reads trajectory data directly from a magnetic disk into memory. The data is composed of both numeric data and simple language commands related to the task. The data input format is in ASCII code, which is easily edited to the format given below in table 6.1, which is an example of a data input file.

```
POSITION
GLOBAL
X                    Y                    θ
t
POSITION
TOOL
T                    N                    θ
t
FORCE
NOTRACK
T                    F_N                  θ
t
END
```

Example of Trajectory Planner Input File Information

Table 6.1

Three different options are shown in the example data. The top line shows Global position control is requested, where the numeric values, X, Y and θ are the next desired absolute positions to be moved to. Movement takes place from the current position in a time of t seconds. The next sequence requests position control in tool coordinates and takes as the starting position for the new movement the current measured tool position. Movements T, N and θ are measured from the current position, taking a time of t seconds to complete the move. The third option is normal force control selection which is only operable in tool coordinates. In this case, from the current measured tool coordinates, the tangential tool movement T, and angle θ, can be specified in addition to a normal force F_N. Again the time for change is specified by t. Finally if the word 'END' is input, the end of the data input is acknowledged by the computer confirming the input information and re-displaying it on the screen.

TRAJECTORY PLANNING ALGORITHM - 'MULTIN'

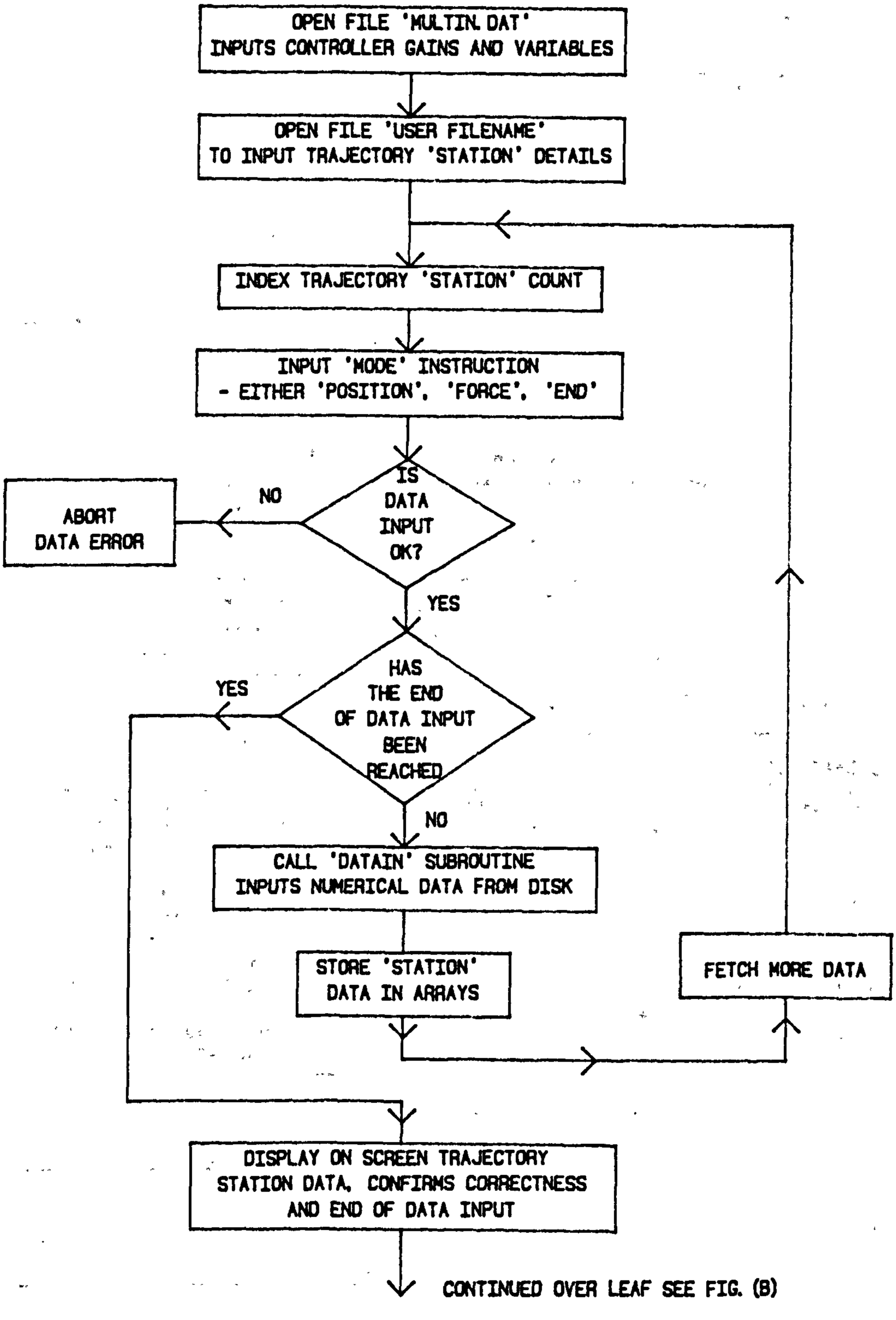


FIGURE 6.10(a)

ALGORTHM 'MULTIN' - CONTINUED

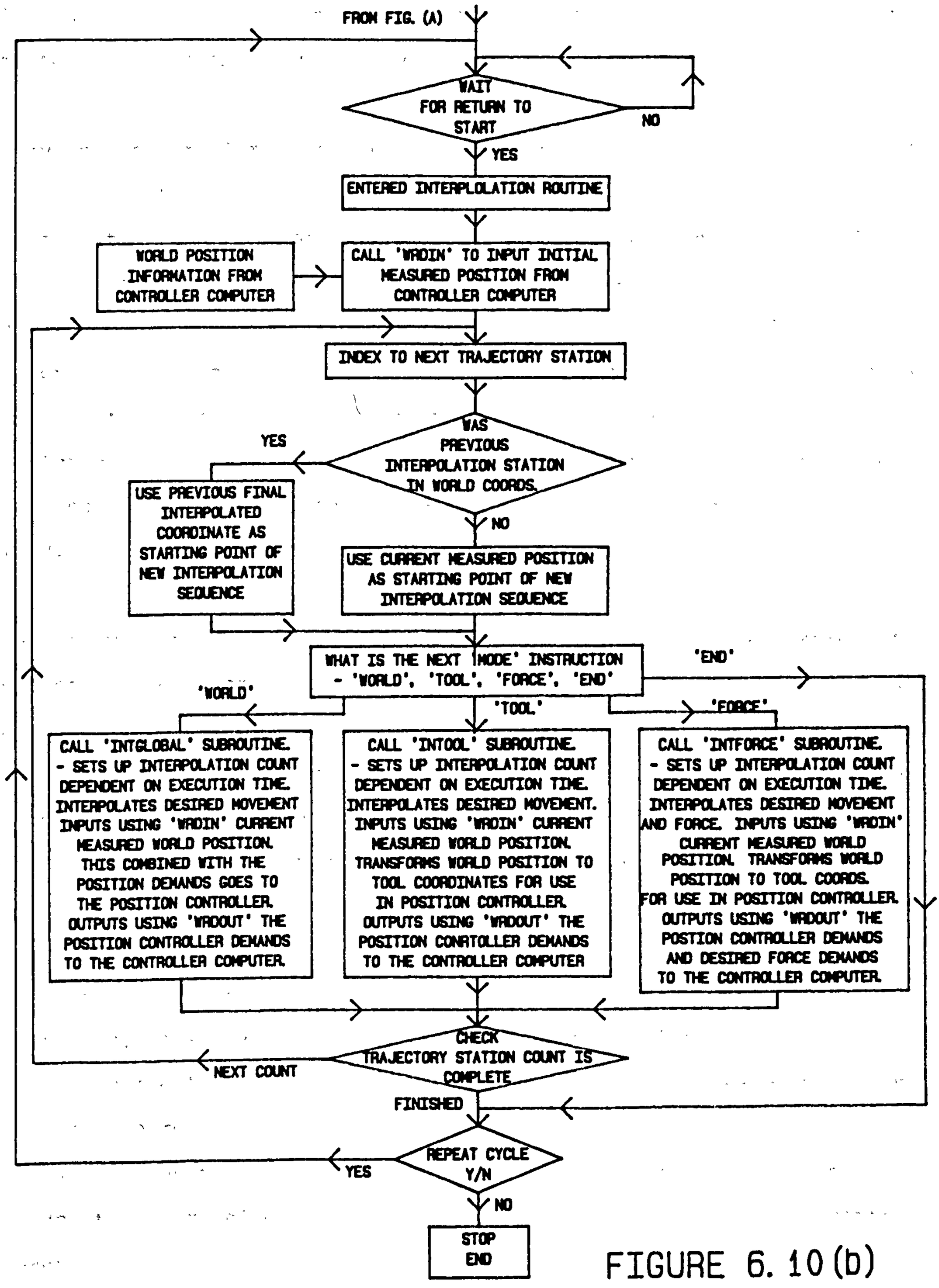


FIGURE 6.10(b)

All data is syntax error checked on input. Immediately any errors are detected the program is terminated and a data error warning given.

To avoid the problems associated with storing a mass of interpolated data, the data is actively interpolated on-line. Interpolation is carried out at control cycle speed, which for convenience is assumed to be 100H_Z, having a time period of 0.01 secs. In specifying a change in position or force the linear interpolation will ramp the desired trajectory between two extreme values in a time period which is a multiple of 0.01 secs. This gives the possibility of a virtual step, by specifying a desired change in 0.01 seconds. In the experimental work this technique was extensively used to generate step changes in force demand.

Potentially there is a major problem in switching from force control to position control because the precise whereabouts of the end effector at the end of a force controlled operation is unknown. No known literature has yet addressed this problem. In this case, the problem has been overcome by taking the final measured position of the manipulator under force control as being the interpolation starting point of the subsequent position controlled movement.

Rather than further overburden the Controller computer with additional computation, it was anticipated that further controller functions would be based in the Trajectory Planning computer. For this reason the outer position controller is based in the Trajectory Planner. Position demands are then passed to the Controller computer and input as velocity demands to the resolved motion rate controller (RMRC) [31] to control movement.

The position controller is relatively unsophisticated and generates rate demands proportional to the errors in position. Early attempts to add integral control to this controller proved unsatisfactory.

Measured force is also sent to the Trajectory Planner from the Controller. This was implemented in anticipation of adding an `intelligent' task switching algorithm which could sense end effector contact with the environment and switch to force or position control accordingly. However time restraints prevented its development.

The current measured position, in World coordinates, and the measured force are input in to the Trajectory Planner every control cycle by using the `WRDIN' routine. Likewise, the demands of the Trajectory Planner are passed down to the Controller using the `WRDOUT' function. The complementary routines `WRDIN' and `WRDOUT' are described in section 6.5.8 and 6.6.4.

The `MODE' instruction is a single instruction coded as a real number and passed from the Trajectory Planner to the Controller. It is passed immediately before the Trajectory Planner passes its control demands to the Controller. This process is repeated every control cycle. The `MODE' switching instruction is generated directly from the information stored on magnetic disk, and informs the Controller as to whether World, tool or force control is requested.

6.6.3 Control Program

The Controller computer runs the main controller program. The program can be effectively divided into two distinct sections. An initial section, which is in fact the main program, and which performs all the

MAIN CONTROLLER ALGORTHM 'MULTSAMP'

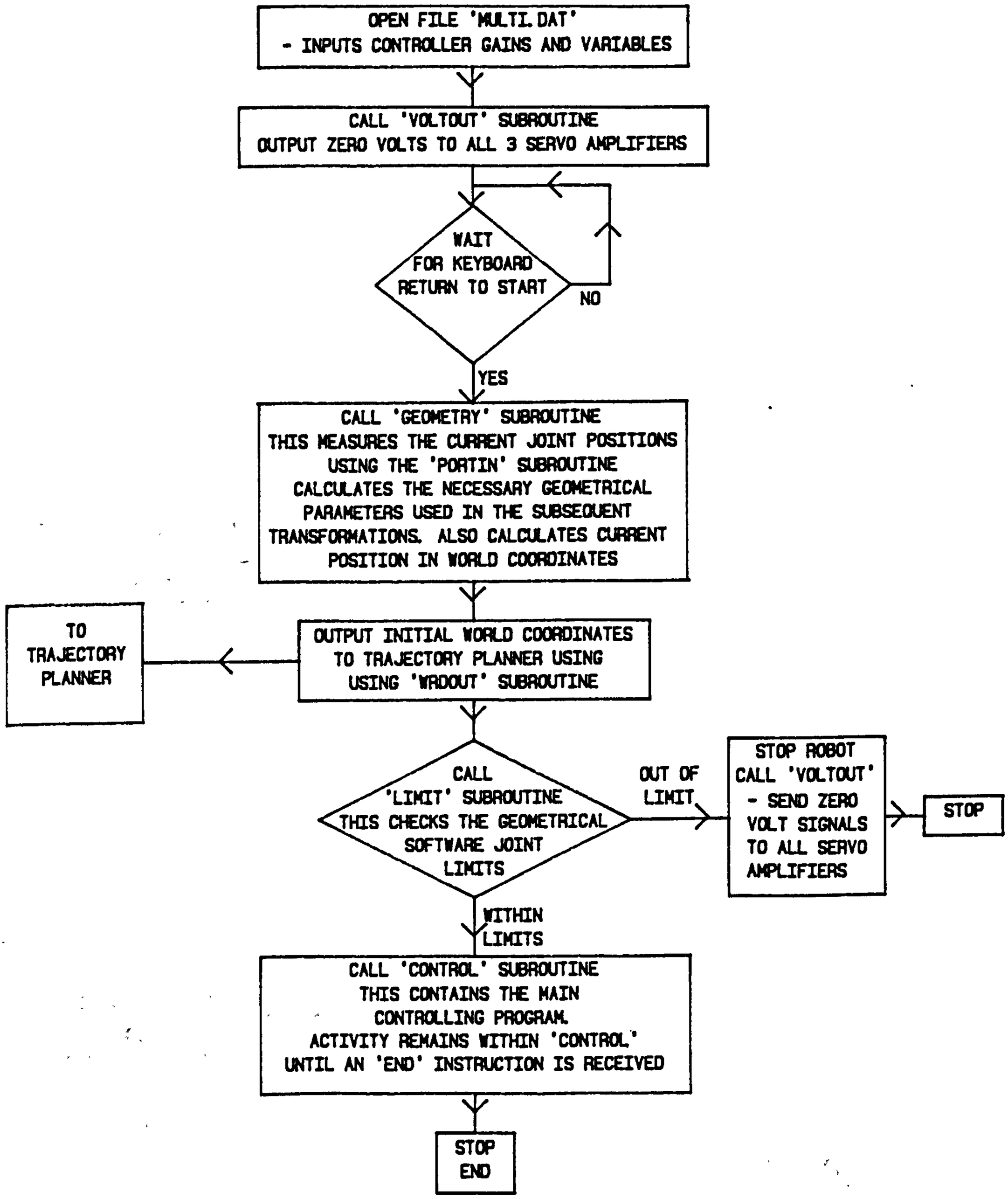


FIGURE 6.11

SUBROUTINE 'CONTROL'

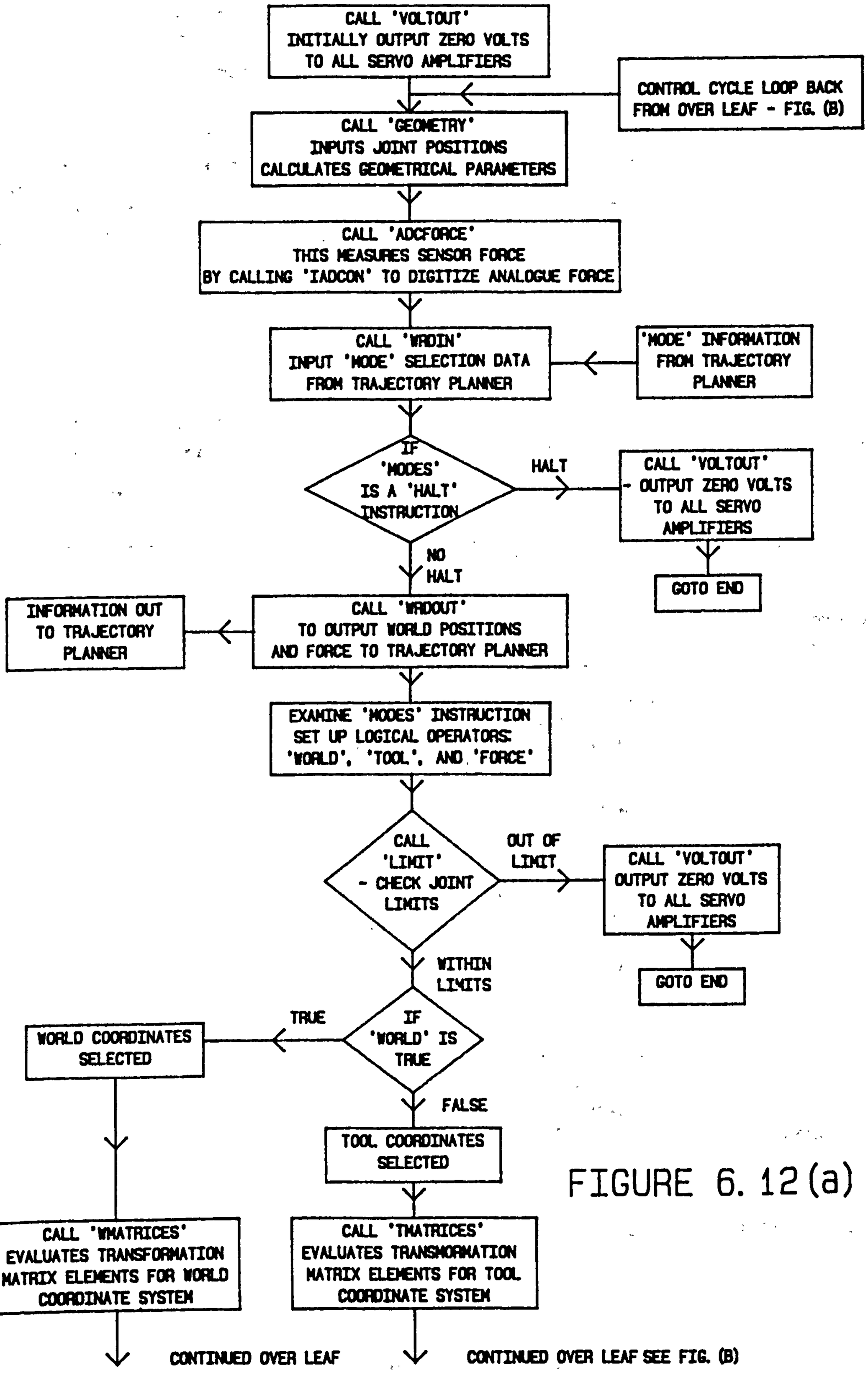


FIGURE 6.12(a)

SUBROUTINE 'CONTROL' - CONTINUED

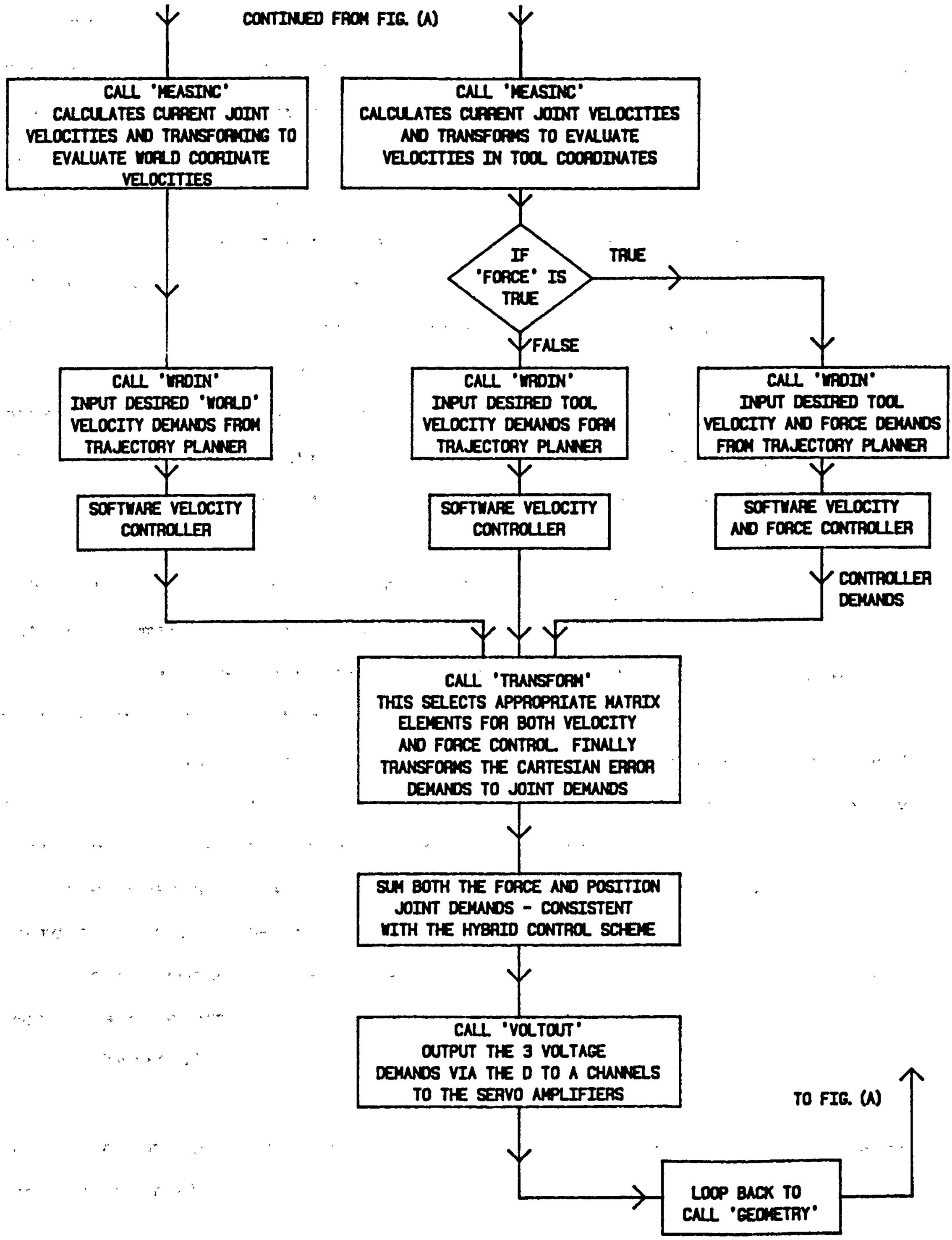


FIGURE 6.12(b)

initial pre-run demands, such as measuring the initial geometry, checking geometrical limits and communicating the initial positional information to the Trajectory Planning computer. Figure 6.11 shows a flow diagram of the complete program `MULTSAMP' displaying this initial stage of the algorithm. The complete Fortran listing of the program `MULTSAMP' is given in appendix D3. On successfully passing through these initial stages the program is allowed to advance to the subroutine `CONTROL' which forms the second and major section of the controlling algorithm. The flow diagram representing the subroutine `CONTROL' is given in figure 6.12 (a) and (b). Control of the robot under position control and/or force control is achieved within the subroutine `CONTROL'. Program control is only returned to the `MULTSAMP' program after a termination or `HALT' instruction has been received.

Within the subroutine `CONTROL' there are a number of important features summarized as follows.

1) All the matrices necessary for control under the Hybrid scheme are evaluated within the subroutine `CONTROL'.

2) The Jacobian, its transpose and inverse are necessary for combined velocity and force control. As a consequence, the force and velocity controllers are based in the Controller computer.

3) Force demands are sent from the Trajectory Planner to the Controller according to the force trajectory required.

4) Velocity demands are also sent from the Trajectory Planner. These demands are determined by the errors of an outer position control loop which is based within the Trajectory planner.

The force controller used within the `CONTROL' subroutine is discussed extensively in chapters 5, 7 and 8.

Unlike the original Hybrid control scheme proposed by Raibert,

computational economy is achieved by selectively evaluating the elements of the $\underline{J}^T$ and $\underline{J}^{-1}$ matrices necessary for the determination of the combined force and position control demands. This contrasts with Raibert's work which calculated all the matrix elements prior to selecting the elements required.

6.6.4 Machine Code Routines

Microsoft Fortran is used for the main `high' level control program. However, it is without low level I/O procedures. Therefore special purpose routines have been written in machine code, using Microsoft Assembler, to deal with communication to the peripheral electronics, as described in section 6.5. These routines are finally added into the main program at the `link' stage of program compilation.

The names and functions of the machine code routines are given below. Their assembler listings are given in appendix D3.

M/C code routine		Function
PORTIN	-	Reads sequentially the encoder counting devices.
OUT	-	Outputs a byte to an I/O location.
INP	-	Inputs a byte from an I/O location
IADCON	-	Starts conversion on a selected analogue to digital channel and returns with a 2's complement value.

WRDOUT - Transfers out a real number of 32 bits wide, in two 16 bit words, under handshake control. Operates in conjunction with WRDIN.

WRDIN - Transfers in a 32 bit real number in conjunction with WRDOUT.

A study of the two main program listings and corresponding flow diagrams, figures 6.9, 6.10, 6.11 and 6.12, shows how these routines are implemented.

CHAPTER 7

RESULTS & DISCUSSION

7.1 Introduction

The Hybrid technique is essentially kinematic in nature and as a consequence manipulator speeds are low to avoid the unmodelled dynamic effects. From an experimental research point of view, position control studies are unnecessary and have been avoided in this work.

Experimental work has been restricted to the largely unaddressed problems of manipulator force control. Some of these problems have already been introduced in the analytical theory, chapter 5, and will now be considered experimentally.

The experimental work has been conducted using the specially designed experimental manipulator equipped with the compliant wrist force sensor. Force control, environment and sensor properties were altered in various experiments. Judgements as to the suitability of the force control are made by considering transients of the force response. Such transient data is automatically logged by the controller computer while it simultaneously controls the robot. In this manner it is hoped to illustrate the salient features of manipulator force control.

The approach begins with initial force control attempts, from this comparisons to the considered analyses are drawn which attempt to identify the fundamental dynamic effects. The validity of the analytical model, and the effects of the proposed force control techniques, are discussed.

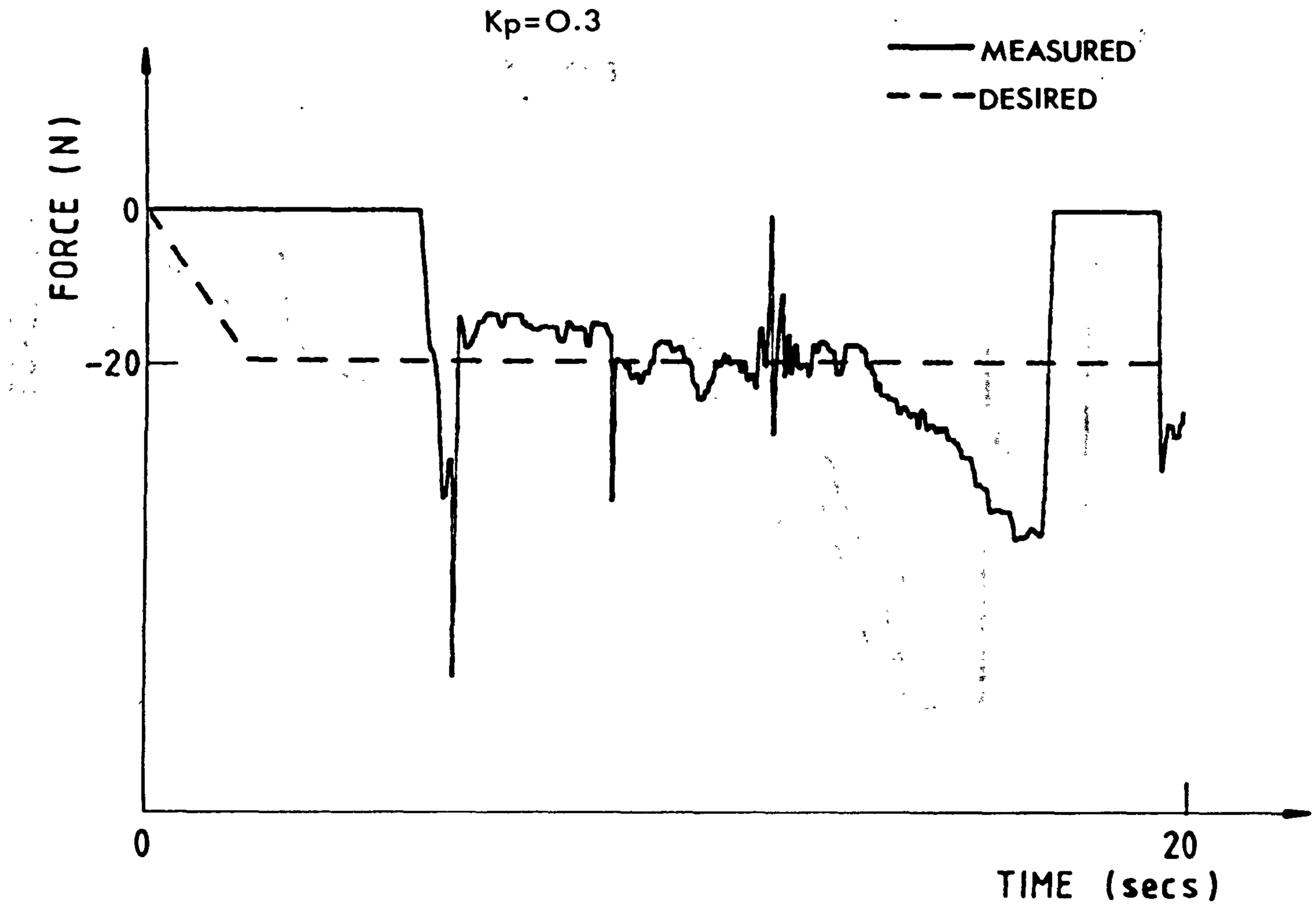


Figure 7.1. Experimental results of blade contour following under low force demands. Key parameters are $K_E > 10^6$N/m, $K_S = 10^5$N/m, K_P=0.3units.

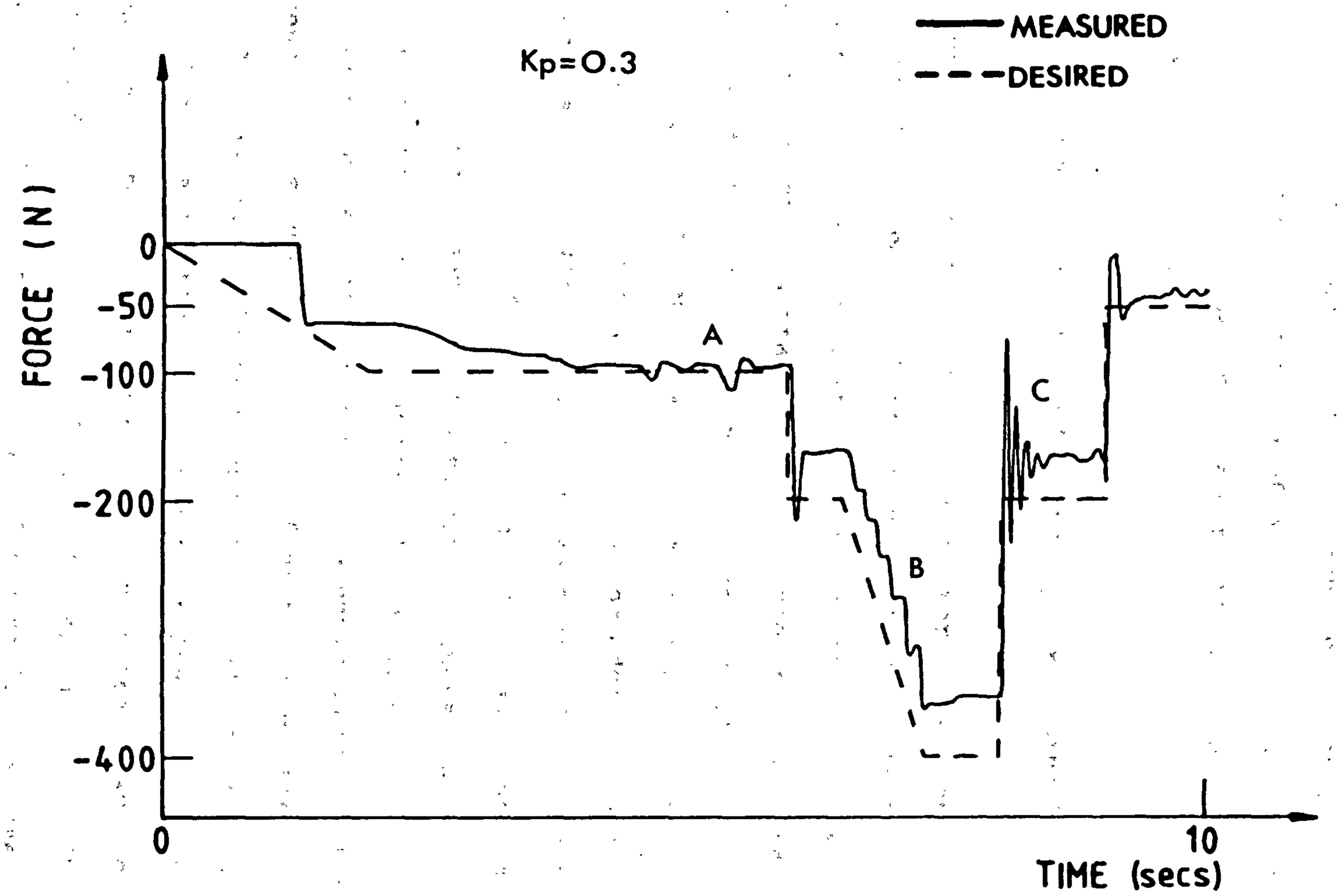


Figure 7.2. Experimental results of blade contour following under severe force demands. Key parameters are $K_E > 10^6$N/m, $K_S = 10^5$N/m, K_P=0.3units.

All controller gains have been determined empirically and, for the purpose of experimentation, are used for comparison purposes only. Later, in chapter 8, comparisons are drawn with analyses. Controller gains remained constant throughout each experiment. In the initial contour following work the value of proportional gain was increased for best response, to allow complete contour following, while remaining within the bounds of stability.

7.2 Simple Force Control & the Effects of Robot Joint Friction

To illustrate the problems of manipulator force control two simple results are described. Both tests involved the contour following of the fabricated turbine blade, starting at the trailing edge and tracking across the concaved surface and around the leading edge. The blade profile presented a variation in environment stiffness, from the lowest value of $K_E = 10^6$ N/m at the thin tip or trailing edge, to approximately 10^7 N/m around the leading edge. However, the sensor stiffness at 10^5 N/m is always at least an order of magnitude lower than the blades stiffness. Section 5.9.1 draws conclusions on the theoretical advantages of low sensor stiffness ($\underline{K}_S << \underline{K}_E$) and its relevance to these tests. Figures 7.1 and 7.2 employ identical force controller parameters and the same blade environment. Differences in the two results are dependent on the differing force demand profiles. In Figure 7.1 the maximum force demand is low and constant at -20N after an initial approach force ramp. However in Figure 7.2 force demands are arduous and at much higher levels: (i) an approach ramp to -100N, (ii) step to -200N, (iii) ramp to -400N, (iv) step back to -200N, (v) Finally a step to -50N. The latter test of figure 7.2 has been devised to produce a variety of useful test conditions, namely ramps and steps at both low and high force levels. The duration of test in figure 7.1 is 20 seconds, in figure 7.2 it is 10 seconds. In both tests the force

control parameters were the same proportional gain, K_p set at 0.3 units.

Clearly the response to low level force shown in figure 7.1 is inadequate. The poor response is due to the high coulombic friction and stiction components in the manipulator's transmissions, considered in the analysis chapter section 5.10. A measure of force response is only possible because the simultaneous position demands produce substantial joint movements, thus helping to overcome stiction.

Figure 7.2 shows a similarly poor response for low force demands during the initial ramp period. In a comparable manner to that mentioned above, the positional demands cause movements in the transmissions. These movements help to overcome friction so producing the force response portion marked 'A'. After the step to -200N, the response to the ramp to -400N displays two obvious effects: (i) a steady state error, (ii) small step changes in force on the ramp section. The cause of the steady state error will be considered in section 7.3. The steps in force response marked 'B' increase with increasing load. This indicates a growing static friction component in the transmissions which is superimposed on the residual static component. The mechanism behind this stepping can be described as: an increasing ramp force error causes an increase in actuator torques, eventually sufficient torque is reached to overcome the stiction and friction in the joint, movement occurs and the force response jumps to a new higher value. Again force ramp error builds and the process continues at higher joint loads, so larger steps are produced.

Where a step in force demand occurs, as opposed to a ramp demand, the force error is initially large and sufficient to overcome the friction and stiction in the joint. This produces movement and a reasonable force response. This effect is clearly seen in both figure 7.2 when

stepping between -100N and -200N and in the initial step force demand of -100N shown figure 7.3 .

A similar test to that shown in figure 7.2 was repeated and recorded in figure 7.3. Identical force controllers were used in both tests. The tests of figure 7.3 differed from 7.2 in that the manipulator did not follow the blade contour but instead remained in a held configuration while the force demands were experienced.

The prevention of position controlled movement was an attempt to remove the effects of change in manipulator dynamics, caused by configuration changes while tracking contours. As a consequence the joint stiction effects, which were previously reduced in figure 7.2 by the super imposition of positional movement demands, produced an extremely poor response to the initial low force ramp demand of figure 7.2. To overcome this problem the initial ramp was replaced by a step force demand sufficient to overcome joint stiction and so produce a reasonable initial force response.

The small steps in force response, which occur in figure 7.2 during the ramp demand at high force levels, are not apparent in figure 7.3. This again confirms the importance of movement in helping to overcome the sticking effects. One would expect the highly non-linear effects of static friction to display an increased effect at low force levels, as discussed in section 7.2.1. Figure 7.3 displays this effect, it shows decaying oscillations of over one cycle when stepping to the -200N level, in comparison to minimal oscillation when stepping up to the 50N level from -200N.

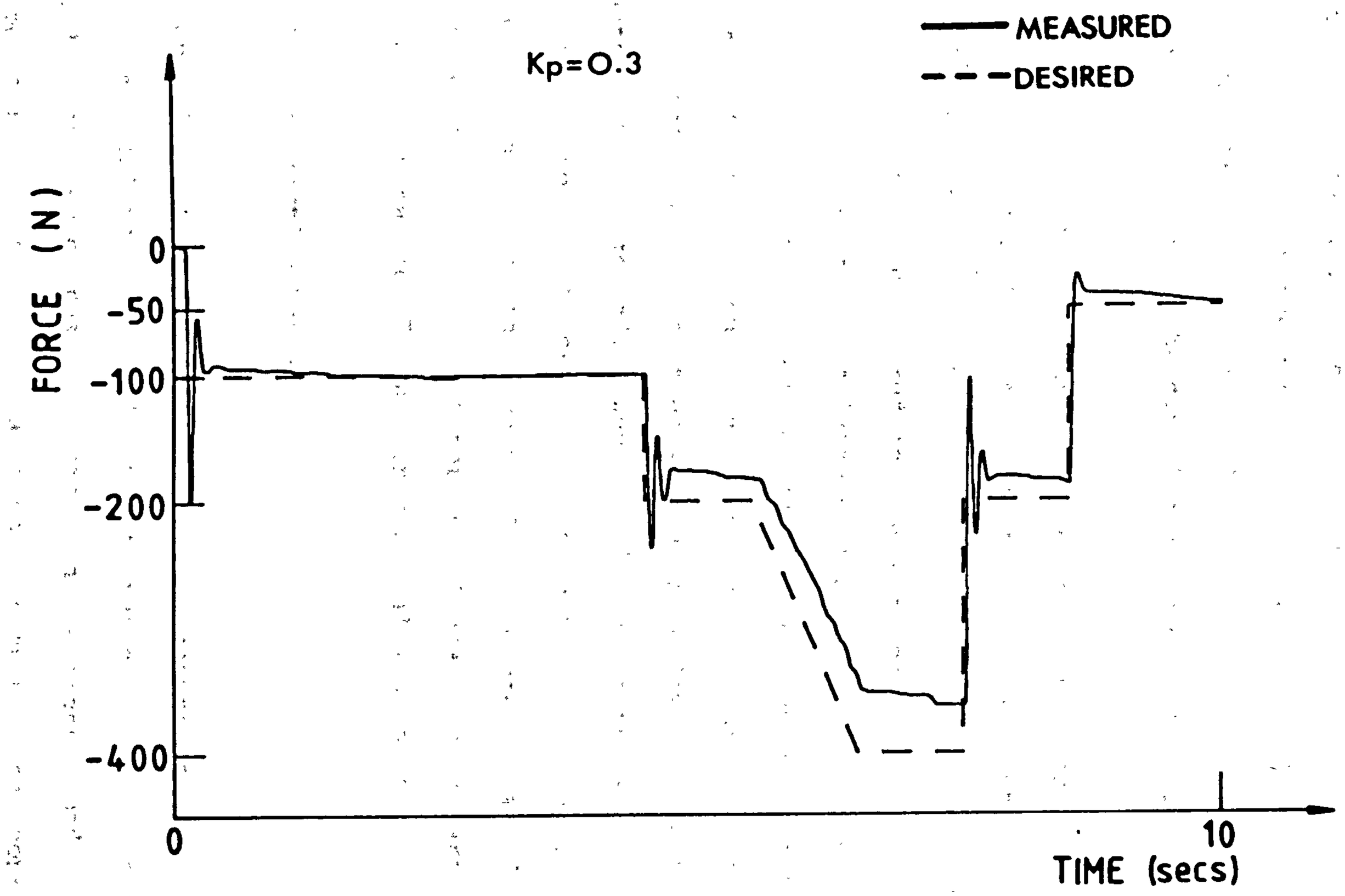


Figure 7.3. Experimental results of a held manipulator configuration during blade contact while undergoing severe force demands. Key parameters are $K_E > 10^6$N/m, $K_S = 10^5$N/m, K_P=0.3units.

The response of figure 7.2, particularly at the portion of transient marked 'C', shows an increased oscillation in the transient not apparent at the other positions around the blade. This effect is also absent in figure 7.3, where the manipulator is in a fixed configuration. Although not conclusively explained here, this effect will be shown in subsequent results, section 7.7, to be due to changes in manipulator configuration altering the dynamics of force control.

Clearly in the results described, the friction forces have a substantial static component. Provided the proportional controlling torques, $-\underline{N}\ \underline{K_C}\ \underline{J}^T\ \underline{K_P}({}^T\underline{F}_D - {}^T\underline{F}_S)$, are much greater than the static friction effects, the effects can be ignored and the earlier analysis holds. It also follows that, owing to the large static friction component, the energy dissipated in friction is dependent on joint velocity rather than the transmitted manipulation forces. Provided the compliances in the system are relatively low, velocities will also be low and the friction damping effects can be ignored. However, while velocity damping terms may be neglected, the frictional interference on force response and steady state may still be significant. This judgement again depends on the magnitude of the controlling torques. These problems are addressed in more detail in section 7.5 and in the analysis chapter, section 5.10.

7.3 Sensor Reaction Forces

Section 5.6 deals with the effect of the measured sensor reaction forces on the manipulator, particularly with respect to force control. It is argued that sensor reaction forces can be transformed to act at the manipulator joints so partially counteracting the controlled torque components. This effect results in reduced manipulator movement, producing reduced deflections of sensor and environment and a

consequential reduction in the net force within the robot structure. Hence the effect behaves as a 'natural force feedback' using the structure of the manipulator as the medium for force transmission. This is probably easier to illustrate by considering a robot moving under positional control while working against a payload, in this case the payload will cause joint torques and hence unwanted joint deflections. This additional proportional feedback, 'natural force feedback', acts as both a dynamic and static disturbance on the system.

Dynamic analysis of payload forces on position control, mentioned in section 5.6, is not relevant to this course of study, but nevertheless it should play a highly significant role in the important area of disturbed manipulator control. Such techniques could perhaps be used in continuing work to eliminate large disturbance effects on manipulator's structures. Here however, the investigation of the technique is confined to improving force response and in particular the steady state error.

7.3.1 Steady State Force Error

Steady state force error is a substantial problem, as shown in figures 7.2 and 7.3. At low force levels errors caused by the static friction effects in the transmissions appear random. At higher force levels the friction effects are relatively less and the majority of the error is due to sensor force reactions.

Attempts have been made, as illustrated in figure 5.4, to eliminate steady state error by implementing integral control. The attempts were not entirely successful in that integral control caused a highly oscillatory response which was slow to decay.

Equation 5.9 suggests that for a given controller gain, $\underline{K}_p$, steady state error is independent of environment and sensor stiffnesses. It also suggests that the parameters affecting steady state error are the proportional gains and the natural force feedback.

Section 5.6 describes an attempt to eliminate steady state force errors without using integral control or high proportional gains. This is achieved by the addition of a further feedback loop, see figure 5.9. Briefly, the additional loop serves to counteract the natural proportional feedback caused by the force sensor's reaction on the manipulator's structure. Successful elimination removes the single $\underline{J}^T$ term in equation 5.9 indicating the possibility of zero steady state error.

The gains in the feedback loop are represented by the diagonal matrix $\underline{I}.K_B$, shown in figure 5.9, where K_B is a scalar constant as discussed in section 5.6.1. However, in experimental work employing only single degree of freedom force control, the feedback gain can be reduced to a single scalar constant. Careful thought as to the positioning of this feedback loop in the overall control system has avoided the need for further transformations. Consequently feedback gains operate in local task coordinates, the same coordinate system as the force sensor. The adopted polishing approach places positional constraints on the sensor and causes it to be aligned to these local coordinates. Thus the task of implementing this control loop is extremely simple.

Choosing the optimum empirical gain for K_B is a cautious exercise. If K_B is too large, the system has high positive feedback, generating unstable forces which could easily damage the manipulator, sensor or environment. To avoid this, the gain K_B is gradually increased, noting each time its effect on the force response.

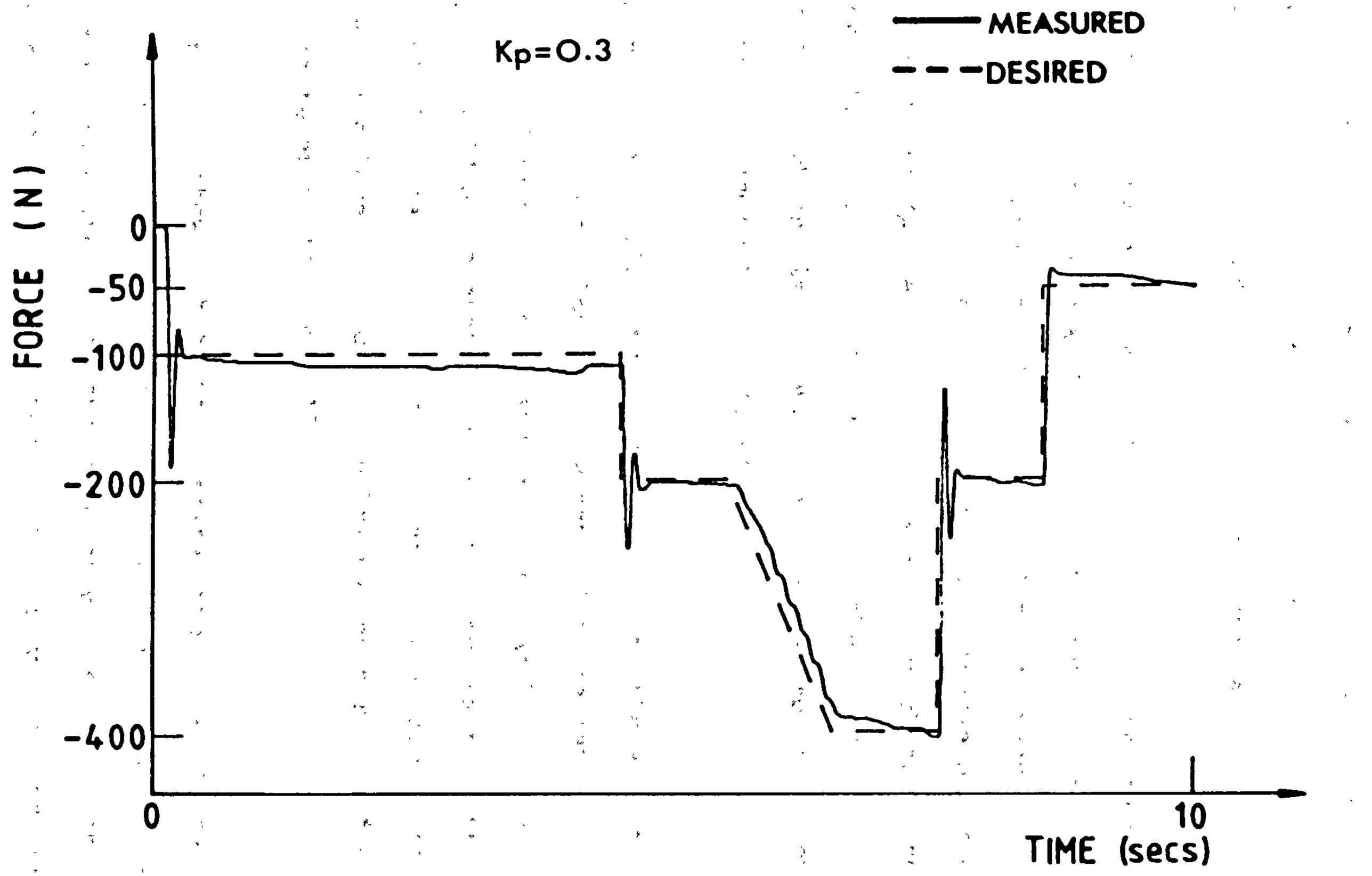


Figure 7.4. Experimental results of a held manipulator configuration during blade contact while undergoing severe force demands. Sensor reaction compensation is present, K_B = 0.035Nm/N. Key parameters are $K_E > 10^6$N/m, $K_S = 10^5$N/m, K_P=0.3units.

Figure 7.4 shows the results of K_B increased to a value of 0.035Nm/N, which is sufficient to remove the larger steady state error. All other control parameters, including the environment and force demand sequence, are identical to the test response shown in figure 7.3. Thus a direct comparison between figures 7.3 and 7.4 can be drawn, illustrating the effects of the proposed sensor reaction compensation.

Section 5.6 in the analysis chapter illustrates how the value of $\underline{K}_B$ can be theoretically chosen as $\underline{K}_B=(\underline{N}\ \underline{K}_C)^{-1}$ for total elimination of sensor reactions. The product $\underline{N}\ \underline{K}_C$ is the product of transmission gearing $\underline{N}$ and overall servo gains $\underline{K}_C$. For the single degree of freedom case the values of $\underline{N}$ and $\underline{K}_C$ are given in the simulation chapter, section 8.2.2, as $NK_C=31.7N/Nm$. Therefore K_B is theoretically evaluated to $K_B=0.032Nm/N$ which is extremely near to the empirical optimum, given above, so confirming the technique.

Again no clear conclusions can be drawn as to the effectiveness of the additional feedback loop at low control force levels, owing to the dominance of joint stiction and friction. At higher levels of force, -200N and -400N where control torques tend to dominate friction effects, the result is clearer. Steady state errors are reduced to within 5% of desired force levels using this technique, compared to figure 7.3 where steady state error is around 15%.

Equation 5.9 shows that steady state errors are large where K_p is necessarily small for stability considerations. Under these conditions this form of compensation should be effective in eliminating steady state errors. However, when dealing with compliant environments (to be covered later in this chapter) higher values of K_p may be tolerated. Consequently the steady state errors are less, and such compensation would be unnecessary.

7.3.2 Proportional Feedback

The combined proportional feedback from both the sensor reaction, via the manipulator structure, and the controller proportional feedback loop have an added effect on the transients, shown in figure 7.3. Eliminating the effect of the natural feedback, hence reducing the total proportional feedback, causes the transients to be less oscillatory, as is shown when a comparison is made between figure 7.4 and figure 7.3.

Section 5.6 considered the desirability of eliminating the natural feedback parameter $\underline{J}^T$ from the combined feedback term ($\underline{J}^T+\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_F$). The benefits of this technique over integral control to eliminate steady state force errors are numerous. The main ones being: simplicity of implementation, lack of unwanted side effects, and improvement rather than degradation, of stability. Potentially the technique has applications in all areas of robotics where end effector disturbances are problematic and not solely in the area of manipulator force control.

7.4 Environment Stiffness

As suggested in the force control analysis chapter, there should be a relationship between force controlled response and environmental properties. Also, changes in environment inertia are likely to be significantly less than changes in environment stiffness. Thus the transient response depends primarily on environment stiffness and it is to these effects attention is now drawn.

An experiment was devised to illustrate the dramatic effect changes in environment stiffness can have on force response and hence on control. Other variables, such as sensor stiffness, manipulator configuration,

environment inertia and controller gains, are held relatively constant.

7.4.1 Experimental Description

Widely varying environment stiffnesses were achieved by employing a cantilever beam and constraining the manipulator's force sensor to travel along the cantilever's length. The force sensor was again equipped with the ball castor end, so minimising tangential friction in the movement direction. The traverse was made from the free to the fixed end over a travel of 40cm in a time of 20 seconds. Step changes in desired force levels were requested every 2.5 seconds between -20N and -40N.

Although the manipulator configuration did change slightly during the experiment, the symmetry of the manipulator was exploited to minimise the effect. This is seen in figure 7.6 where the initial and final configurations were mirror images of one another.

Along the length of the beam, the stiffness varied significantly from around 1,000N/m at the free end to a theoretical infinity at the built-in end. Figure 7.5 shows both the theoretical and measured stiffnesses.

To confirm the insignificance of the changes in environment inertia and manipulator inertia a series of similar experiments were repeated using discrete lengths of cantilever beams. The object of these experiments was to duplicate the same experimental conditions as in the beam tracking, using exactly same controller and environment stiffness, but this time the manipulator configuration remained fixed. This was achieved by adjusting the position of the built-in support rather than by moving the manipulator. The force sensor was then applied to the

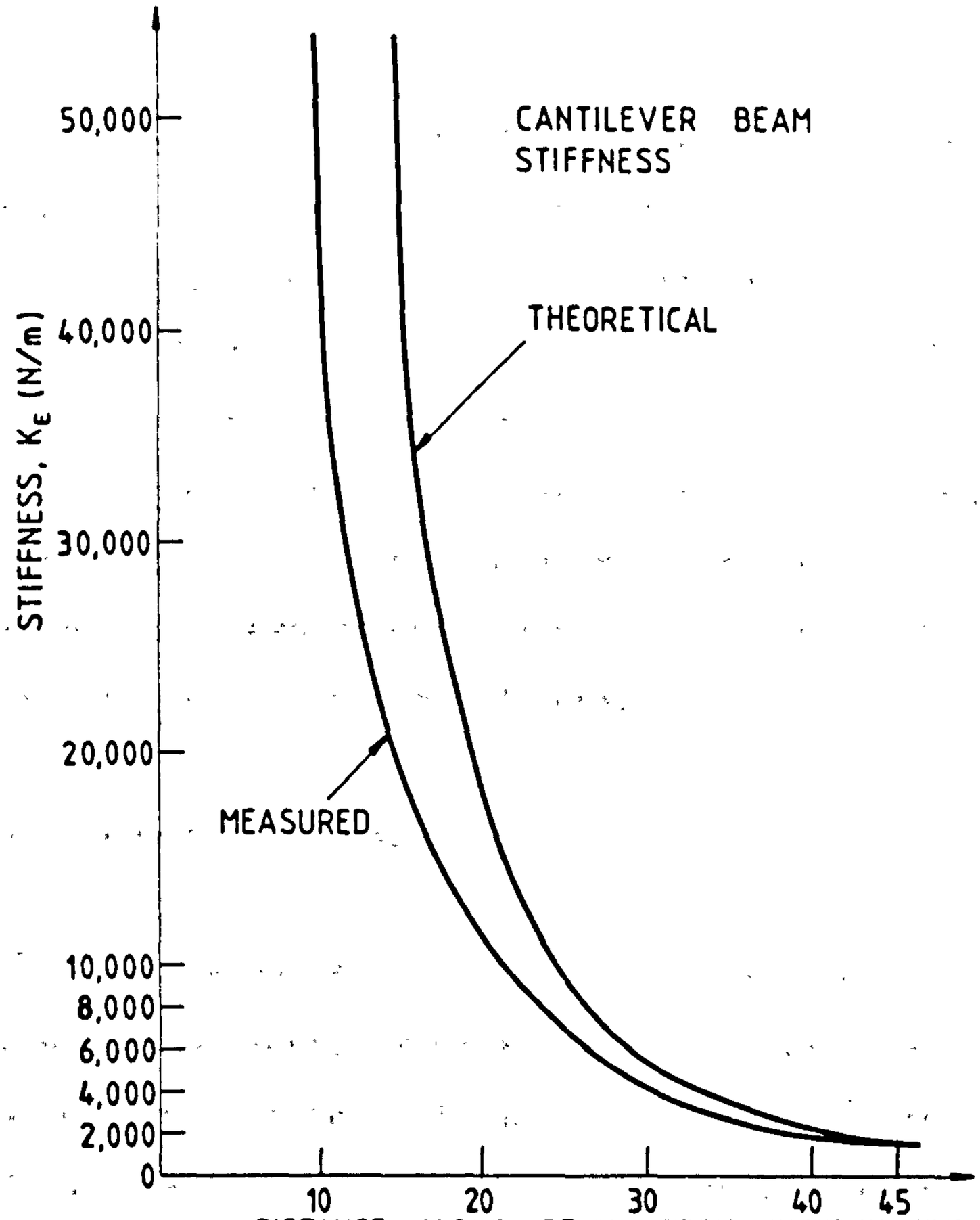


FIGURE 7.5

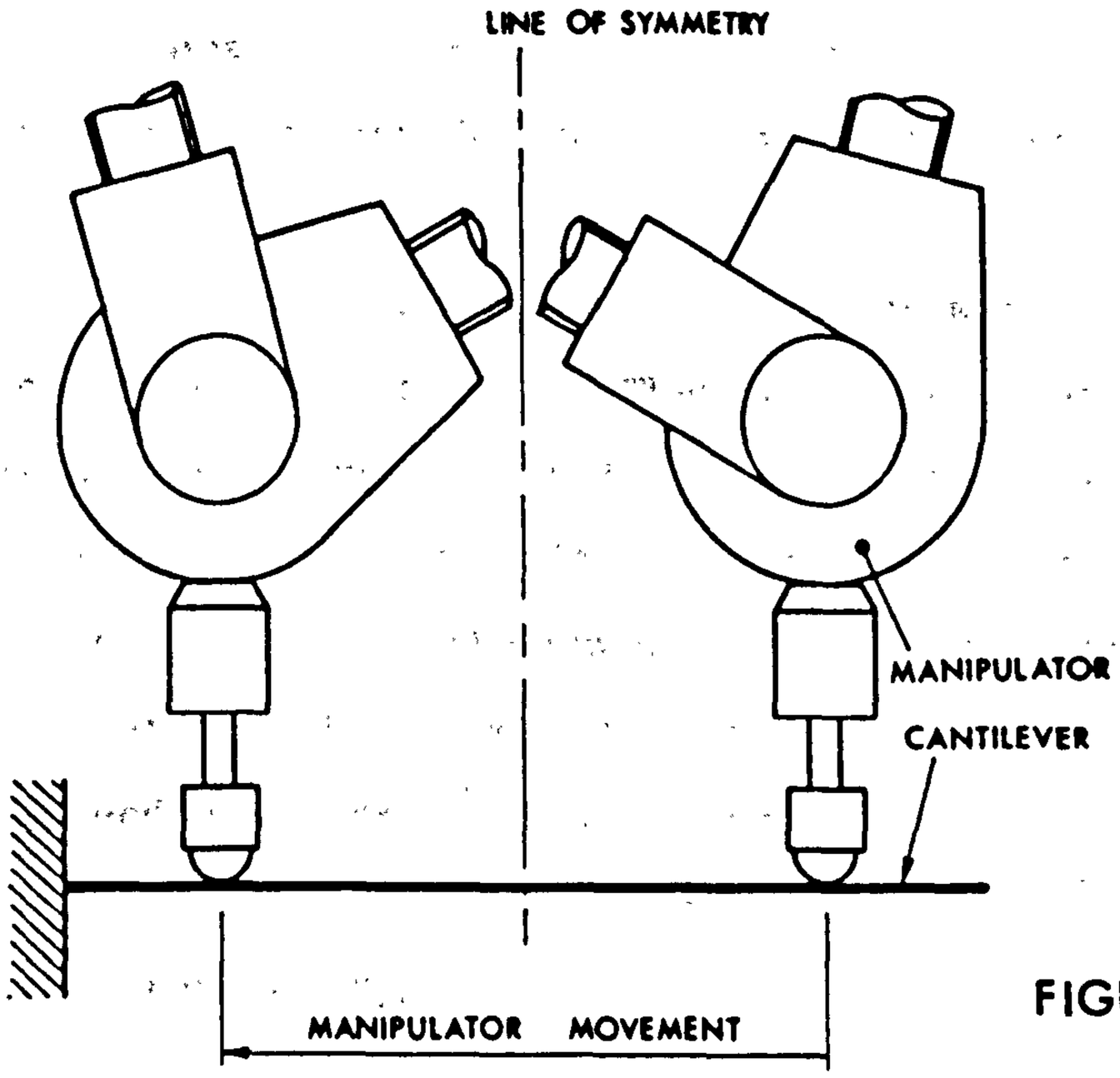


FIGURE 7.6

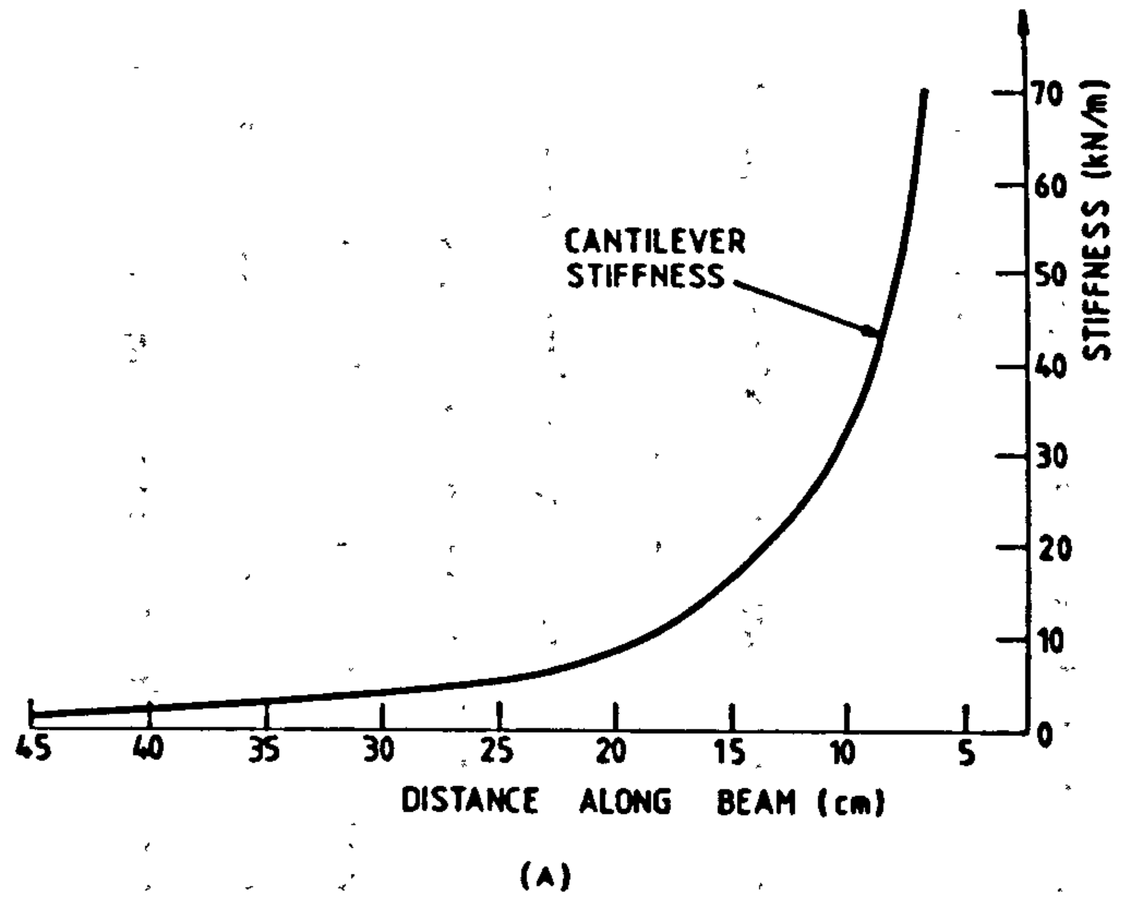
Illustrates the manipulator arrangement for the cantilever beam following experiments.

beam's free end which provided the desired environment stiffness. The environment inertia then differed from that in the beam tracking experiment, but no observable differences in force responses could be detected. From these results it can be concluded that the environment inertia plays an insignificant role within the beam tracking experiment and can therefore be ignored as a variable. The validity of the beam tracking experiment is therefore confirmed. The results of these discrete beam experiments are too numerous to be included and only confirm the results presented here.

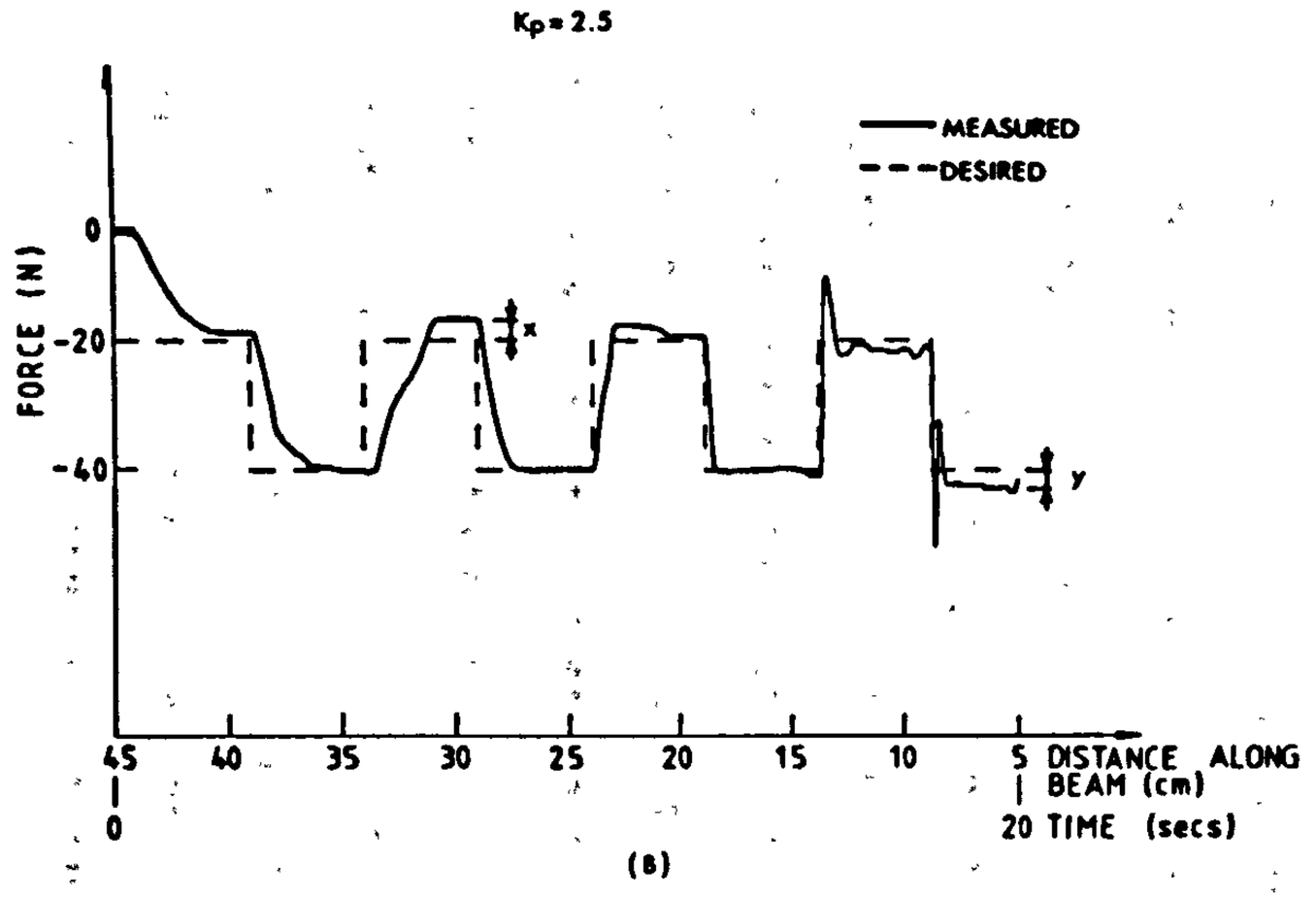
7.4.2 Results and Discussion

Figure 7.7 shows four individual graphs related to the cantilever tracking experiment. Figure 7.7(a) shows the same information as presented in figure 7.5, and indicates how the measured cantilever stiffness varies according to its length. The other three graphs give the force responses for three similar cantilever tracking experiments with three different values of proportional gain: K_P=2.5 in figure (b), K_P=5.0 in figure (c), and K_P=10.0 in figure(d). The natural feedback elimination gain remained constant throughout at F_B= 0.035 units, the previously determined optimum for best steady state response.

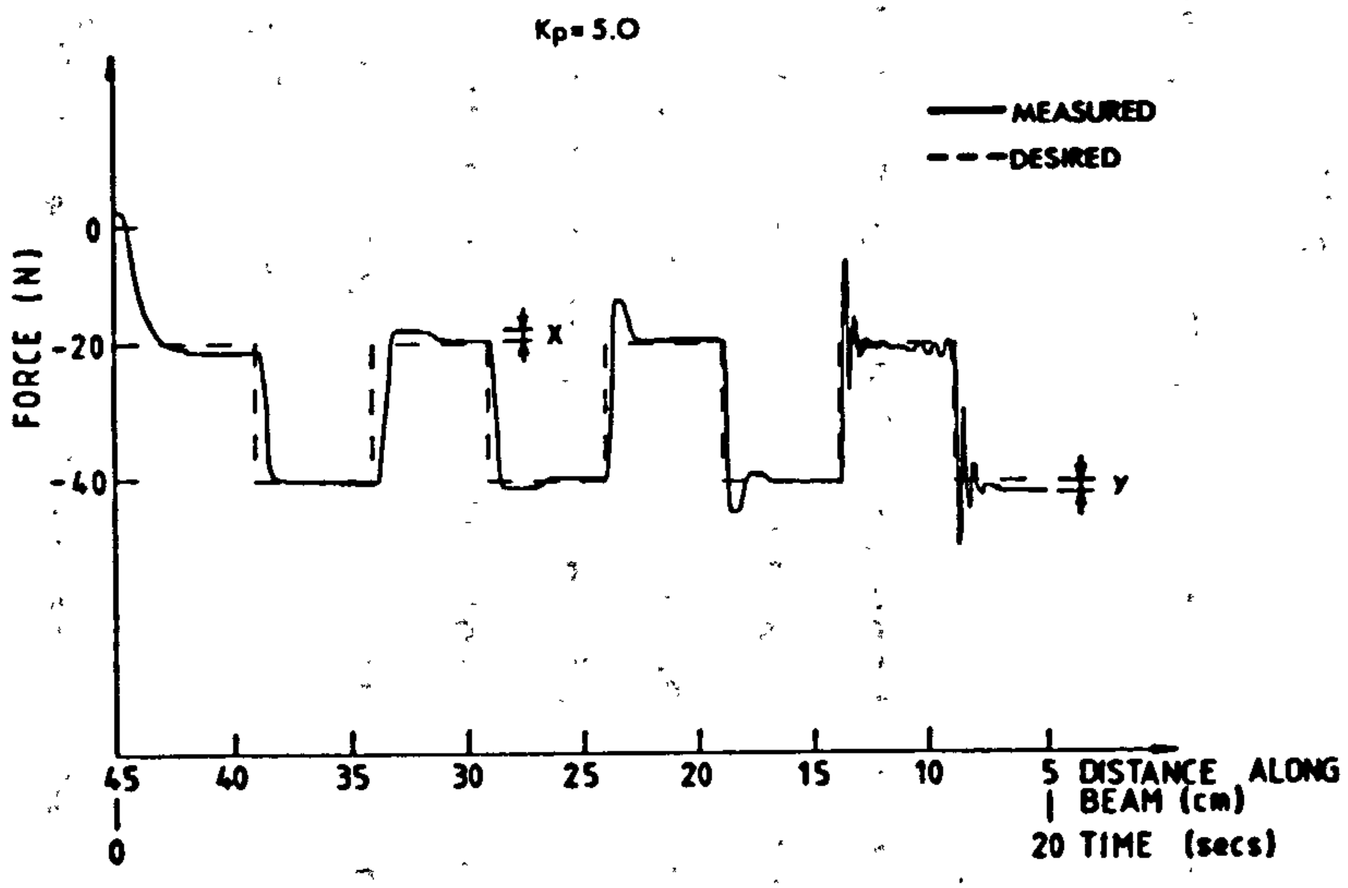
The results of figure 7.7 allow easy comparison with one another and variation in environment stiffness. They show that stability decreases with increasing environment stiffness K_E and with increasing proportional gain K_P. However, it is worthwhile observing how the results have been affected by the unwanted joint frictions, mentioned in section 7.3. Equation 5.6 has been further modified by the elimination of the natural feedback term $\underline{J}^T\ {}^T\underline{F}_S$, giving equation 7.1. This equation shows the effect of the controlling torques $\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_F({}^T\underline{F}_D - {}^T\underline{F}_S)$ on the unwanted friction torques $\underline{T}_{FRIC}$.



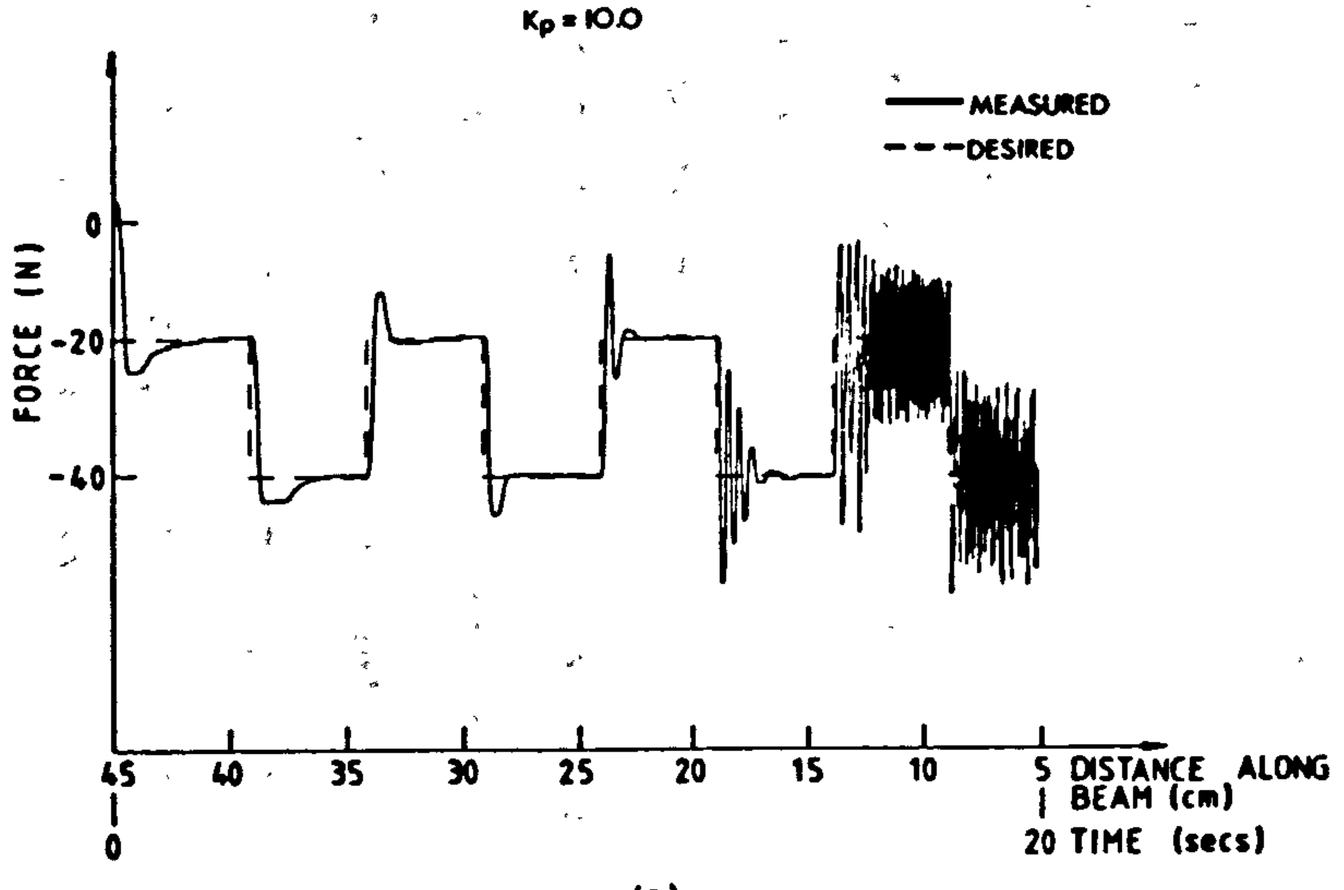
(A)



(B)



(C)



(D)

Figure 7.7. Results of a cantilever tracking experiment which present a large variation in environment stiffness, shown in figure (a). Key parameters are: K_E which varies approximately between 10^3 and 10^5N/m, and proportional gain K_p is changed in figures: (b)K_p = 2.5units, (c)K_p=5.0units, (d)K_p=10.0units.

$$-\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_F({}^T\underline{F}_D - {}^T\underline{F}_S) + \underline{T}_{FRIC} = [(\underline{N}^2\ \underline{I}_A + \underline{I}_L)S^2]\theta_L \quad (7.1)$$

Force errors in the transients are due to control dynamics, while the steady state errors are due only to unwanted joint static friction effects, as the natural force feedback is largely eliminated, (owing to K_B compensation and relatively high values of K_p). The maximum observable steady state errors of figure 7.7(a) and (b) are marked X and Y. They are approximately equal in each figure, indicating a consistent maximum static friction that can be tolerated by the force controller. The maximum steady state error in figure 7.7(c) is approximately half the steady state error shown in 7.7(b), whilst the proportional controller gain K_p used in figure 7.7(c) is twice that used in 7.7(b). Assuming joint friction forces are consistent in exactly similar manipulator configurations and duty cycles, although still generally unpredictable, then equation 7.1 is confirmed in that,

$$\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_F({}^T\underline{F}_D - {}^T\underline{F}_S) = \underline{T}_{STAT} \quad (7.2)$$

for steady state force response, where $\underline{T}_{STAT}$ is the static friction component of $\underline{T}_{FRIC}$.

These simple observations conform to the model proposed by equation 5.18 which shows steady state joint friction disturbances are independent of sensor and environment stiffnesses. Also if the value of proportional gain can be increased further, say to that used in figure 7.7(d) where K_p=10.0, then the steady state frictional disturbances can be neglected. This improved disturbance rejection is certainly an important advantage to be gained from high proportional gains.

The sensor stiffness of $1.4\text{x}10^5$N/m tends to dominate the environment stiffness in the three figures of figure 7.7, where the environment stiffness varies from $1.5\text{x}10^3$N/m at the beam free end to around

$1.0x10^5$N/m towards the support. The results should then reflect equation 5.21, shown here in modified form in equation 7.3 where the natural feedback has again been eliminated and $\underline{K}_D=0$.

$$\begin{aligned} \text{DET} \mid \underline{K}_S^{-1} \mid . \mid & [(\underline{N}^2\ \underline{I}_A + \underline{I}_L)J^{-1}\ \underline{I}_{E,S}]S^4 \\ & + [(\underline{C}_L\ \underline{J}^{-1} + \underline{K}_V)\underline{I}_{E,S}]S^3 \\ & + [(\underline{N}^2\ \underline{I}_A + \underline{I}_L)\underline{J}^{-1}(\underline{K}_E + \underline{K}_S) + (\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_P)\underline{I}_{E,S}\ \underline{K}_S]S^2 \\ & + [(\underline{C}_L\ \underline{J}^{-1} + \underline{K}_V)(\underline{K}_E + \underline{K}_S)]S \\ & + [\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_P\ \underline{K}_E\ \underline{K}_S] \mid = 0 \end{aligned} \qquad (7.3)$$

If $\underline{K}_E$ varies but remains very much less than $\underline{K}_S$, then only one parameter of equation 7.3 should be affected as the environment stiffness changes, namely the matrix of constants $\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_P\ \underline{K}_E\ \underline{K}_S$. Figure 7.7(b,c,d) all show that an increasing K_E is accompanied by an increase in transient frequency, indicating an increase in force response bandwidth. This improvement in response is due to the increase of the constant parameter in equation 7.3. Under these conditions the results show the introduction of environment compliance causes a degradation in force response. This is demonstrated in figure 7.7(b), where values of K_E and K_p are at their lowest. If the proportional gain is increased to $K_p=5.0$ in figure 7.7(c) and then to $K_p=10.0$ in figure 7.7(d), the response for the corresponding environment stiffness improves. Transient response frequency and hence bandwidth have increased but the response in 7.7(d) also shows signs of underdamping. The effect of increasing K_p in equation 7.3 changes other parameters as well as those affected by K_E and hence the response cannot be maintained as K_E varies. This is particularly unfortunate as it prevents a simple, model based, adaptive force controller to be used to maintain force response as the environment stiffness varies. High K_E accompanied by high proportional gain can, as figure 7.7(d) shows, lead to instability and should be avoided. Conversely, low values of K_p can lead to problems of unwanted joint frictions interfering with force response. Therefore in some

situations adequate force response may only be achieved at the expense of introducing mechanical compliance into the system.

These simple experiments show that, where the environment stiffness is less than force sensor stiffness, $\underline{K}_E$ is an extremely important variable in force control and cannot be ignored in general manipulation tasks. Chapter 8, involving single degree of freedom simulations, also confirms K_E as an important variable in force control dynamics.

7.5 Damping Effects (Friction & Velocity)

Compliance in the force controlled system, whether present in the force sensor or environment, allows manipulator movement while responding to a force demand. This is seen in the previous section, 7.4, where the free end of the beam environment displays low stiffness and manipulator movement occurs. At a force of 40N and an environment stiffness of 1500N/m a deflection of around 2.5 cm in the environment is is produced. Then, depending on manipulator configuration, the associated joint velocities could be significant, dissipating energy in friction, and hence damp force response. In certain situations, where force rate damping is not employed, frictional damping may be the only factor promoting stability. Certainly Raibert [34], in his Hybrid force work, relied on this property for stability where compliance in the force sensor allowed some degree of manipulator movement to dissipate energy in Joint friction. However, it is far from ideal to rely on friction damping for stability. Joint friction has many unpredictable components and since joint velocities are configuration dependent, the damping effects will not be consistent as manipulator position changes. It is proposed in this section to study these effects in more detail by using the analysis of section 5.10, illustrated by selected experimentation.

Joint friction effects are unknown and uncontrollable, but, in an attempt to appreciate their possible viscous effects, a series of experiments have been conducted. These experiments introduce controlled velocity damping to provide an analogy to the friction damping effects. However the analogy must be first justified.

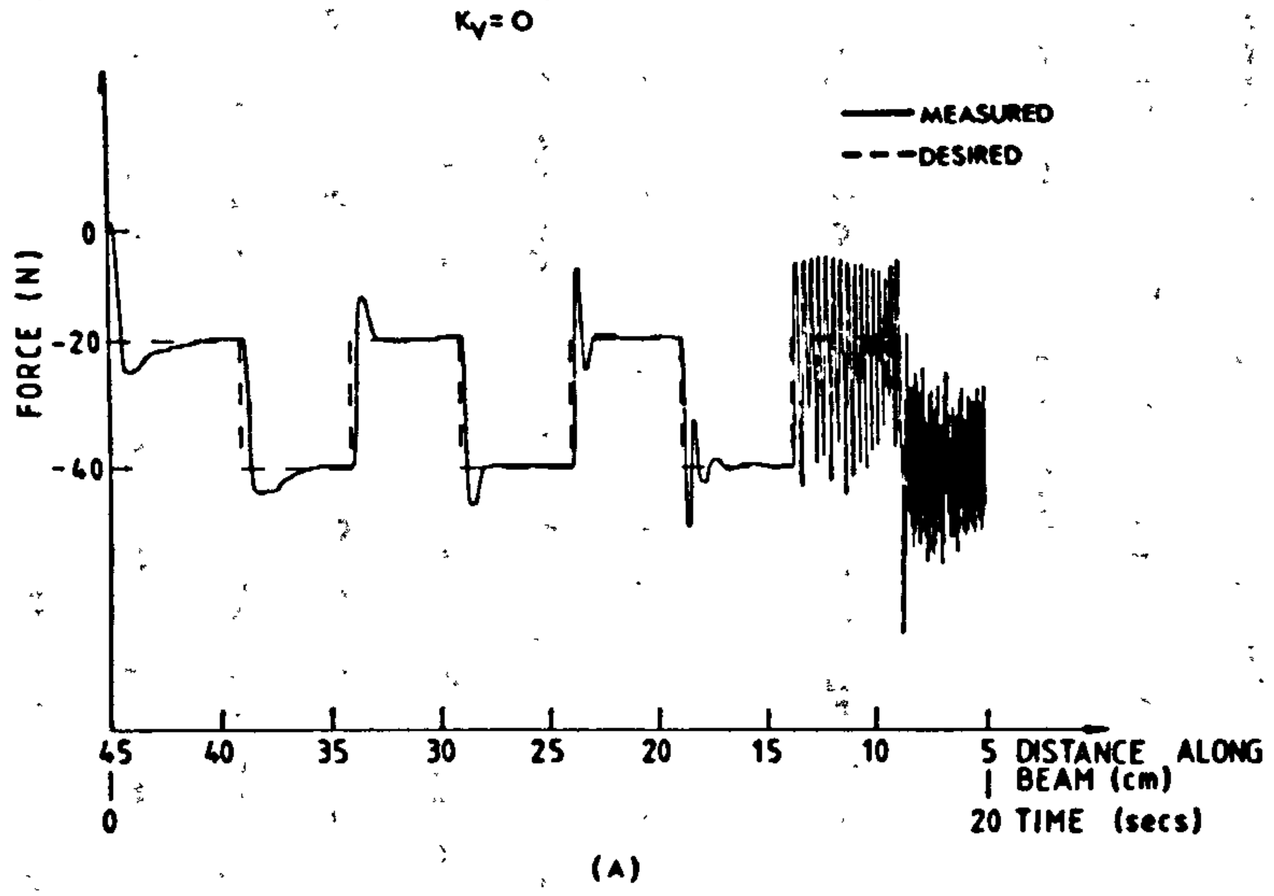
In section 5.10.1 viscous damping was shown to be difficult, if not impossible to evaluate owing to the unpredictable nature of the friction. Therefore, a model equivalent of joint friction was proposed.

Equation 5.21 shows, adding velocity feedback damping to the force controlled system contributes similar terms to those created by joint friction and motor regulation alone.

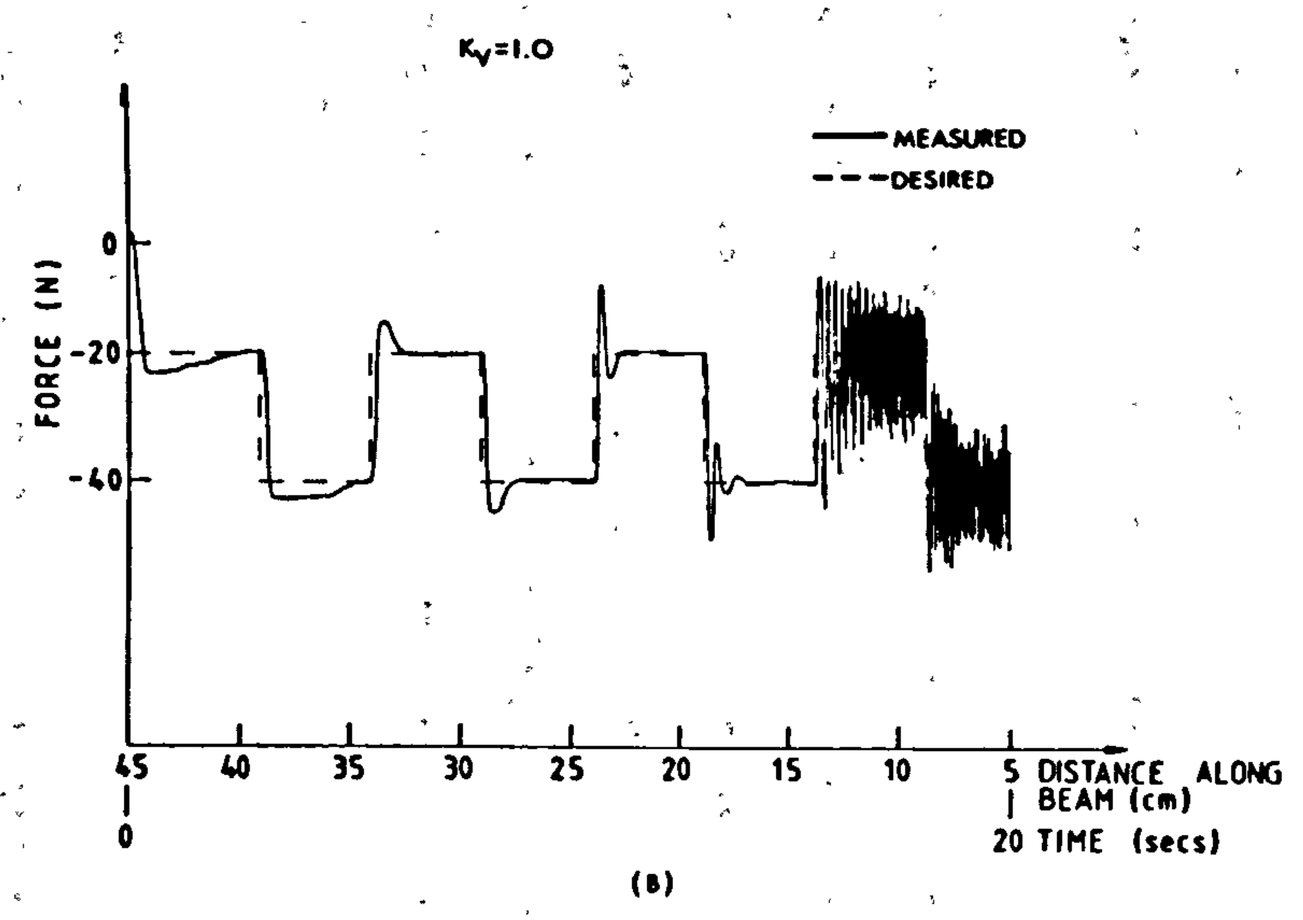
Therefore provided the limitations of the comparison are noted namely that, velocity and friction damping exist in different coordinate systems and that the friction based viscous damping component is unpredictably non-linear, then a general analogy between the two forms of damping may be made. That is a controlled variation in velocity damping can directly be compared to changes in joint based viscous damping. Thus the effect of changes in environment stiffness on a viscously damped system may be inferred by observing the effects on a system with controller based velocity damping.

7.5.1 Experimental Description

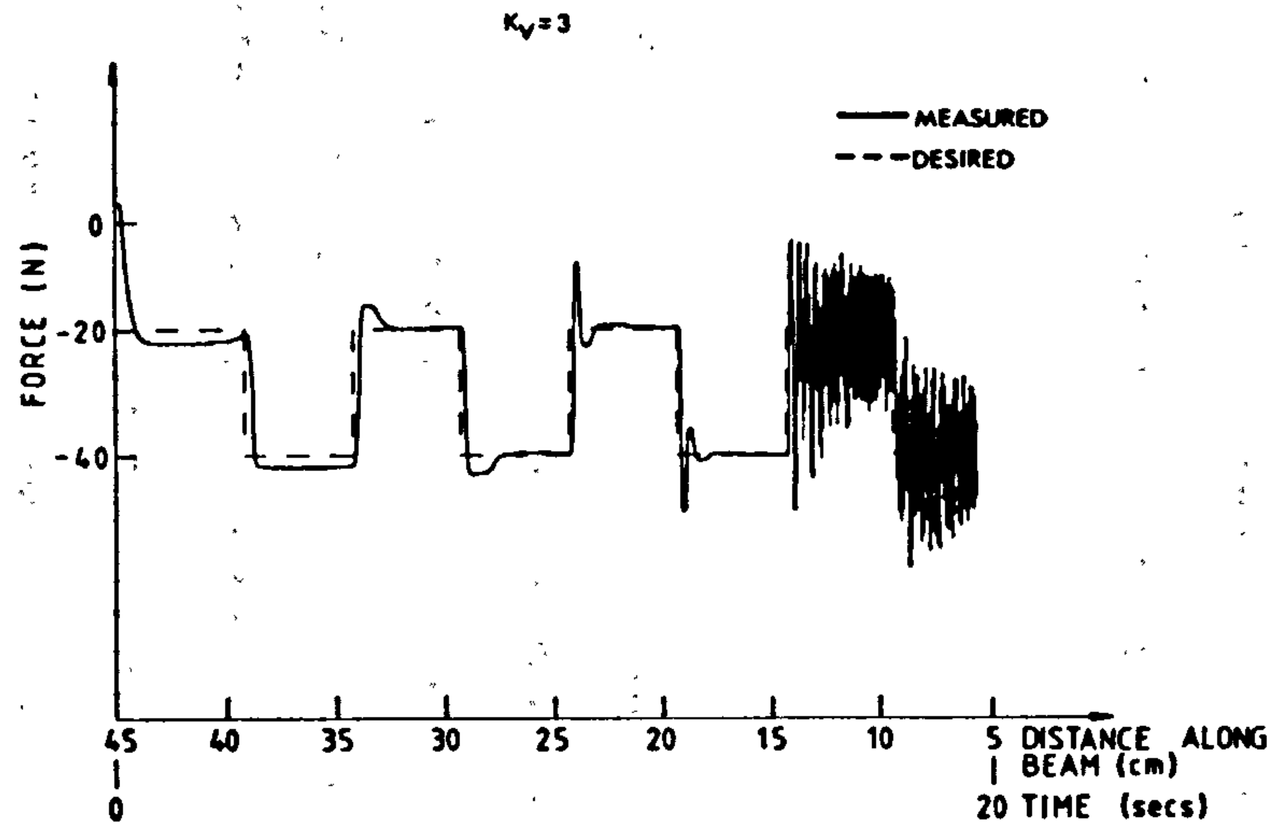
The physical arrangement of the robot and cantilever beam necessary to provide a variation in environment stiffness, was exactly the same as that described in section 7.4. As explained above, the purpose of the experiments was to introduce velocity damping into the single axis force control. The velocity damping was implemented in software running within



(A)



(B)



(C)

Figure 7.8. Results of a cantilever tracking experiment which present a large variation in environment stiffness. Key parameters are: K_E which varies approximately between 10^3 and 10^5N/m, and velocity feedback K_V which takes values in figures: (a)K_V=0, (b)K_V=1.0unit, (c)K_V=3.0units.

the controller computer. Simply, the calculated cartesian velocity, the derivative of the incremental manipulator movement, ${}^{T}X_{L}$, was multiplied by an action constant, K_V, and the product negated from the force controller demand. Thus velocity damping operated in the same local cartesian frame as the force controller, and not in joint coordinates.

The constant K_V, is a dimensionless gain holding no absolute significance and is used in a relative context only. The maximum value of K_V was 3 units, and these results are displayed in figure 7.8(c). The minimum value of K_V was zero, that is, no velocity damping was present and the results are presented in figure 7.8(a) for comparison. Figure 7.8(b) displays an intermediate result for velocity damping K_V of 1 unit.

In common with all the previous experiments employing the cantilever beam to achieve a variation in environment stiffness, stiffness increases towards the right in all figures, varying according to figure 7.7(a). Proportional gain K_p was constant throughout the three experiments and set at 10 units.

7.5.2 Results and Discussion

Comparing the three results of figure 7.8(a), (b) and (c), little difference in the force responses can be seen, except for the responses near to the free end of the cantilever, where K_E is lowest. This is to be expected, since a lower environment stiffness corresponds to higher manipulator velocities which therefore produces greater damping effect. Increasing velocity damping from K_V=0 in figure 7.8(a) to K_V=3 in 7.8(c), the transient is seen to loose its overshoot and the system becomes critically or overdamped, as shown in figure 7.8(c).

In this experiment, since $\underline{K}_E$ is significantly less than $\underline{K}_S$ then only one parameter group in equation 7.3 is affected by variation in $\underline{K}_E$, the proportional group. Force response damping is due to both the parameter groups in odd terms of S which are independent of K_E. This presents a possible explanation as to why velocity damping effects at relatively high values of K_E have negligible effect, shown in a comparison of figures 7.8(c) with 7.8(a). At high values of K_E, the proportional grouping of equation 7.3, $\underline{N}\ \underline{K}_C\ \underline{J}^T\ \underline{K}_P\ \underline{K}_E\ \underline{K}_S$, is likely to dominate the remainder of the characteristic equation. Transient response frequency is also seen to increase with K_E for the same reasons. This is confirmed in the simulations in the following chapter. The converse is also shown, that for low values of K_E, the responses become damped and display a lower transient frequency.

In all three cases displayed in figure 7.8, instability occurs at exactly the same point in the tracking cycle where the environment is relatively stiff. This point of instability appears to be unaffected by the addition of velocity damping.

These are extremely significant results, demonstrating that friction and velocity damping cannot be relied upon to create stability if the environment stiffness of the system is too high. The analyses in chapter 5 and chapter 8 show, force rate damping would appear to be a better proposition if reasonable force rate signal could be obtained, but unfortunately at this stage such a sensor is not available.

7.6 Artificially Compliant Environment

Considered in section 5.9.2 of the analysis chapter are the possible advantages of maintaining a constant environment stiffness. The major advantage of a constant environment stiffness is that it removes the

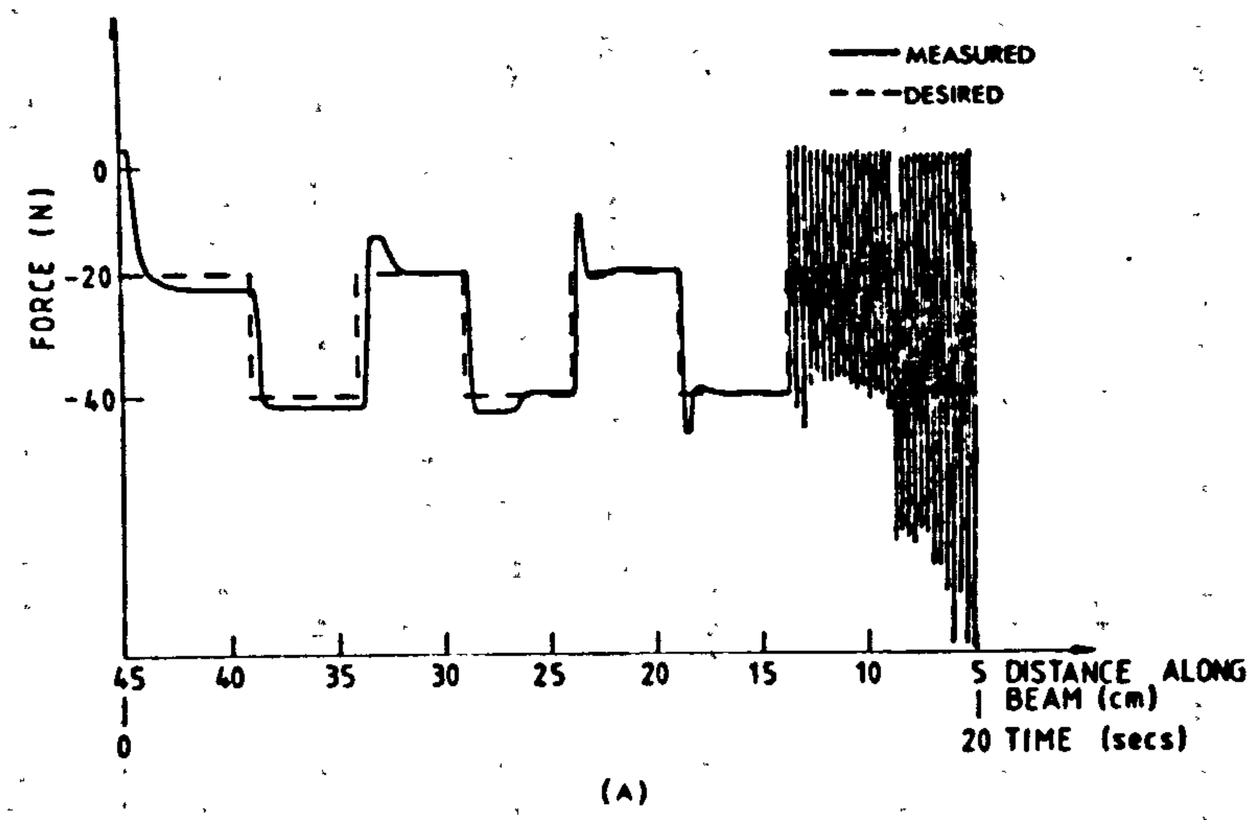
considerable dependence the dynamics of force control has on environment stiffness variation. Artificially lowering environment stiffness is one of the simplest ways of maintaining K_E constant. But this approach also has added advantages of, allowing higher values of proportional controller gains to operate and provides the necessary compliance required to avoid the high frequency disturbances above the force controller's bandwidth.

Two approaches were proposed in section 5.9.2 to achieve this condition of constant environment stiffness. The first, given in figure 5.10(a), involved supporting the immediate environment in a mechanical suspension. The second approach shown in figure 5.10(b), and adopted in the experimentation below, involved adding a compliant element between the actual environment and the force sensor. Then providing the actual environment's stiffness is much higher than the compliant element's stiffness, the environment is effectively isolated from the force control dynamics by the element.

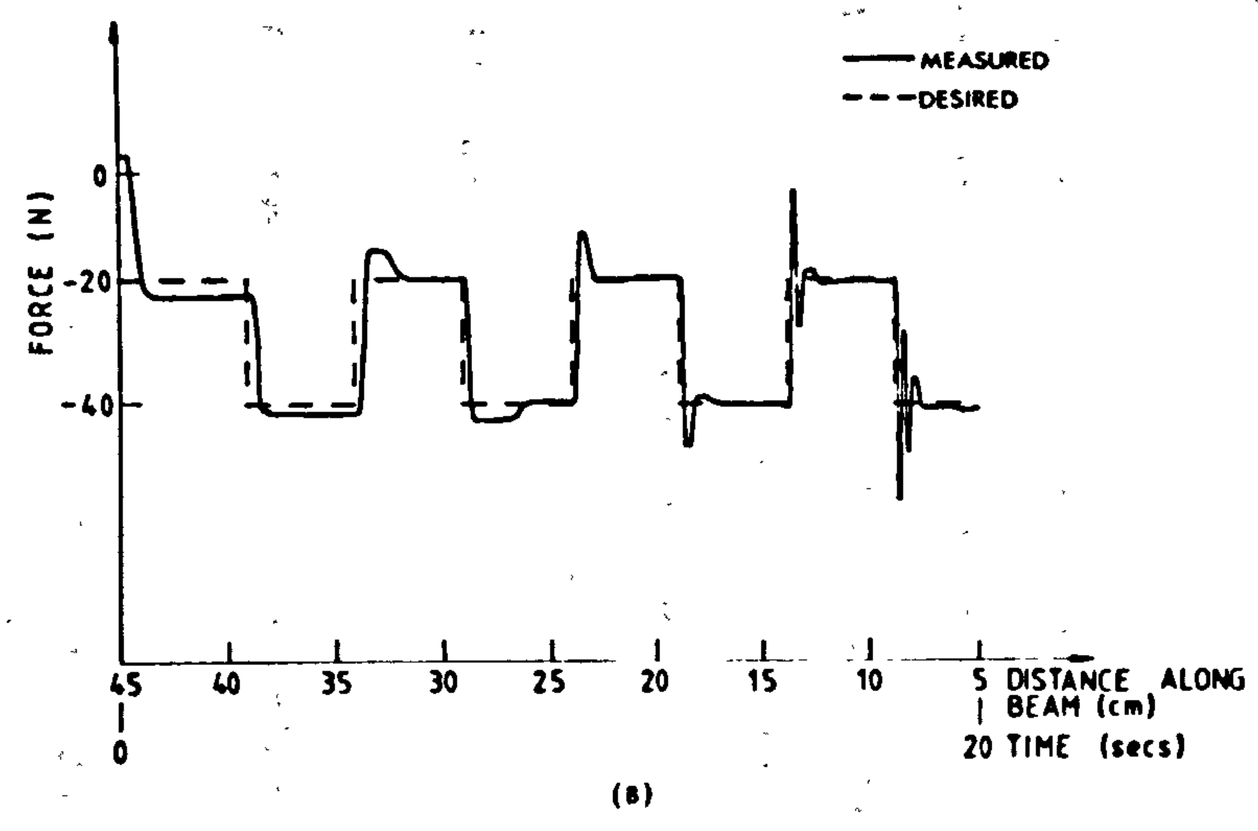
7.6.1 Description of Experiment

Single degree of freedom force control was used to investigate the compliant element approach. All other experimental parameters were held as near constant as possible, the exception being the environment stiffness. For these reasons the cantilever tracking experiment was once again used.

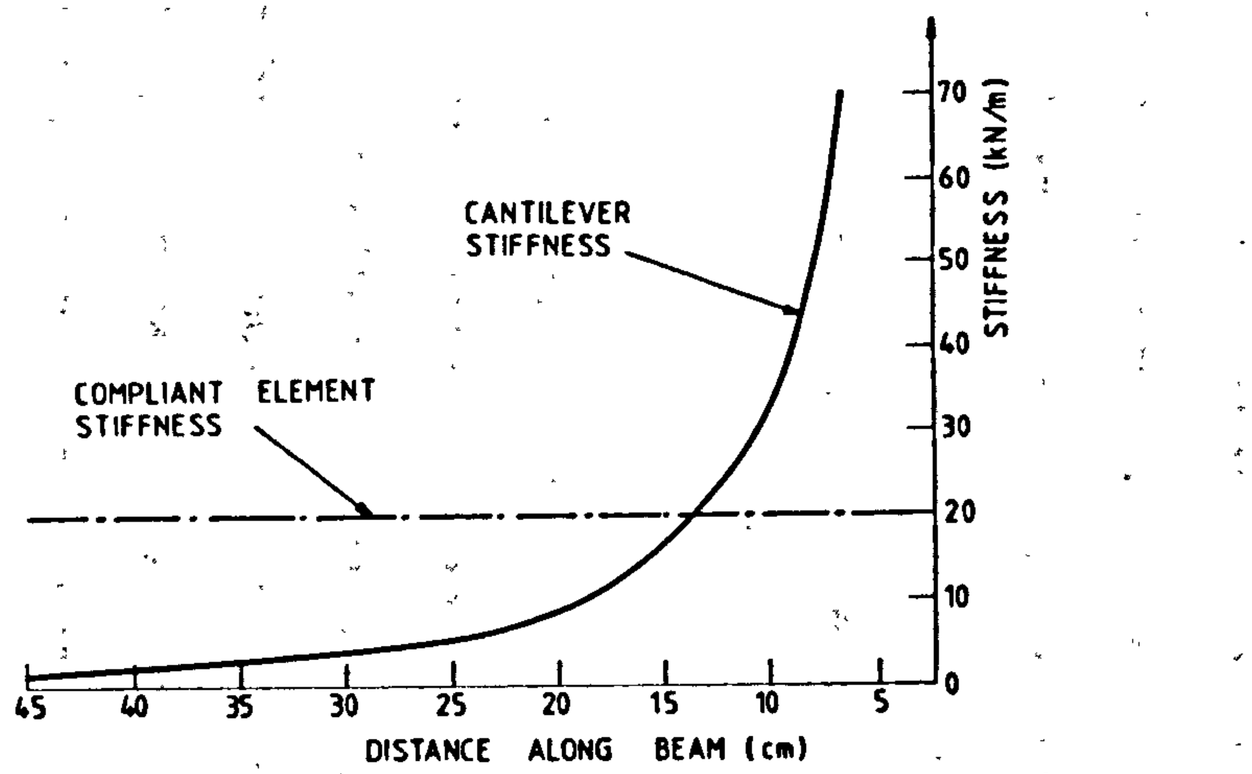
Two experiments were conducted. The first experiment, results given in 7.9(a), was a repeat of the previous cantilever following experiments using the force sensor in its normal guise-fitted with its rigidly mounted ball-castor end. In the second experiment, the sensor was fitted with an attachment containing the compliant element and which



(A)



(B)



(C)

Figure 7.9. Results of a cantilever tracking experiment with and without the compliant sensor element. Figure (c) shows a comparison of the complaint element stiffness to the varying environment stiffness. Figure (a) is without the compliant element and figure (b) with the element. K_p=7.5units in both cases.

also guided the now compliantly supported castor to move along the force sensing axis. This provides single degree of freedom compliance of the environment.

The compliant element used within the sensor was of rubber foam. Foam was used principally for ease of design, manufacture and assembly, where the foam can easily be attached by adhesive. The foam has a near linear stiffness characteristic over the range of loading conditions considered here and has negligible damping properties at such low frequencies. Thus the compliant element can be considered to be a simple spring with stiffness of $2.0x10^4$N/m.

7.6.2 Results and Discussion.

Figure 7.9(a) shows the results of the cantilever following experiment using the conventional force sensing arrangement, while figure 7.9(b) shows the results of the now compliant element arrangement. Figure 7.9(c) shows the variation in cantilever beam stiffness, exactly that given in figure as 7.5, but now with the compliant element stiffness indicated as a horizontal line at the $2.0x10^4$N/m mark.

Equation 5.21 is relevant to these results. It shows that if K_E can be made constant and all other parameters remain constant, then the force response should remain unchanged. This effect can be seen by comparing figure 7.9(a) to 7.9(b). The instability appearing in figure 7.9(a) due to increasing K_E is now completely eliminated, as shown in figure 7.9(b). The increased stability is due to the compliance of the element dominating that of the cantilever beam. Up to the point where the cantilever's stiffness is equal to the element's stiffness, there is very little differences between the responses of figure 7.9. After this point, the compliance of the element becomes more significant and the

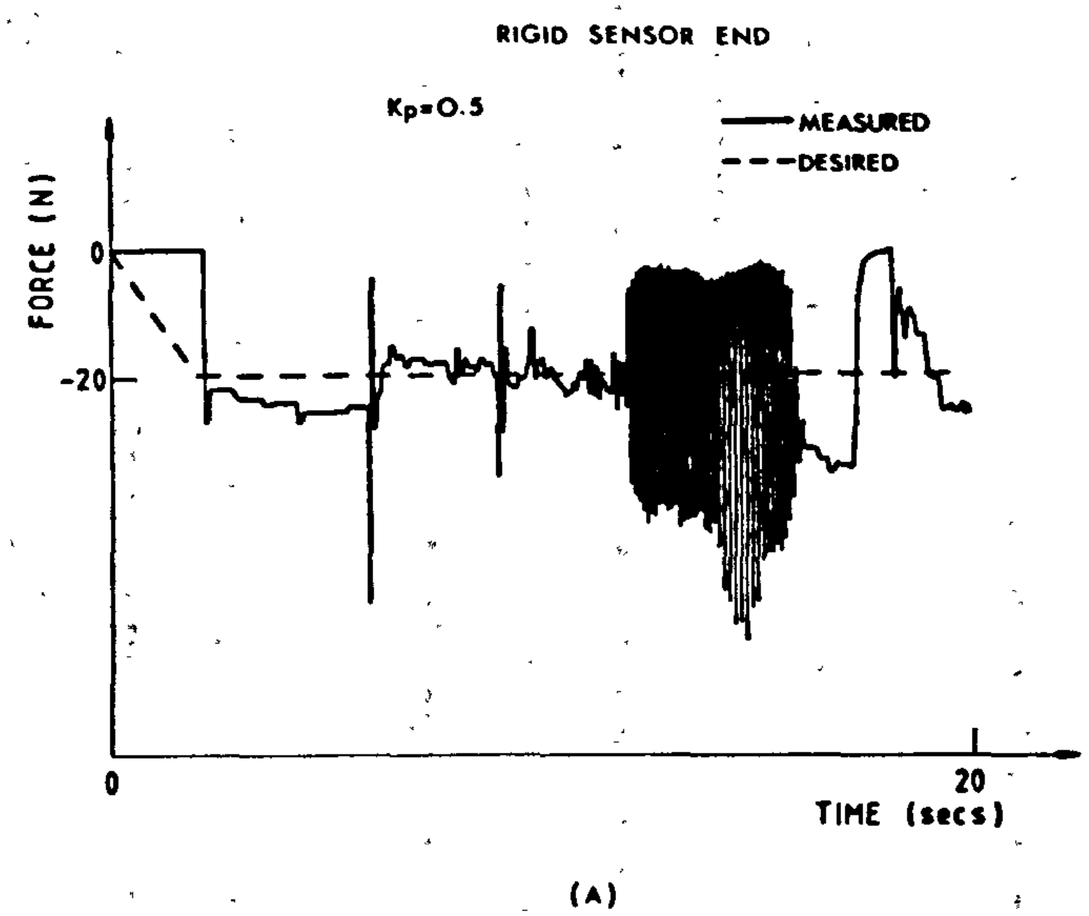
deflections of the cantilever increasingly less so. In this way the stiffness of the cantilever is progressively isolated from the dynamics of force response, while the constant stiffness of the element begins to dominate, maintaining a consistent response. The greater the differential between the environment and compliant element's stiffnesses, then the more consistent is the force response.

This approach to manipulator force control has direct application to manipulator polishing using a compliant bob, where the dynamics of the environment are isolated from the force control by the compliance of the bob.

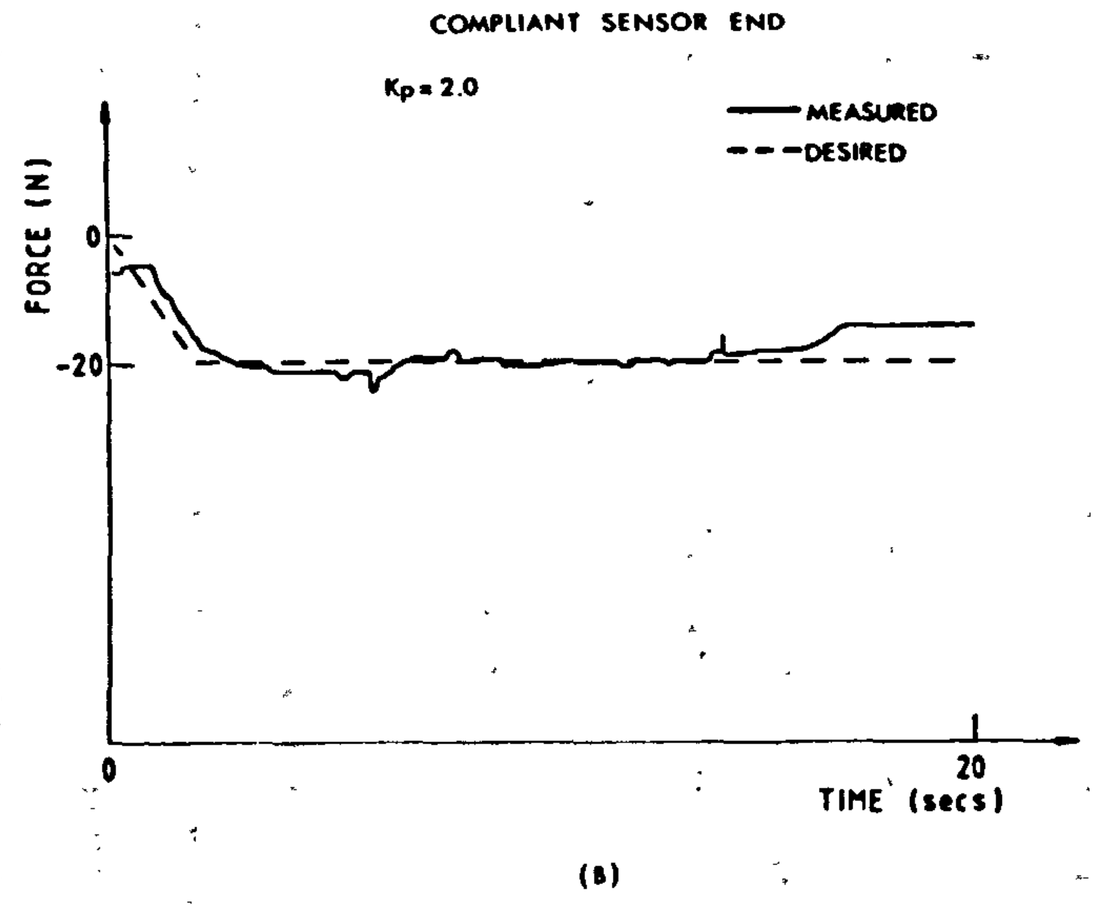
7.7 Manipulator Configuration Effects

In section 7.6, the unchanging manipulator configuration has been utilized to investigate the benefits of employing a compliant element to maintaining the dynamics of force control constant. Having established this fact, a similar approach can be used to isolate and investigate the effect of changing manipulator configuration on force response.

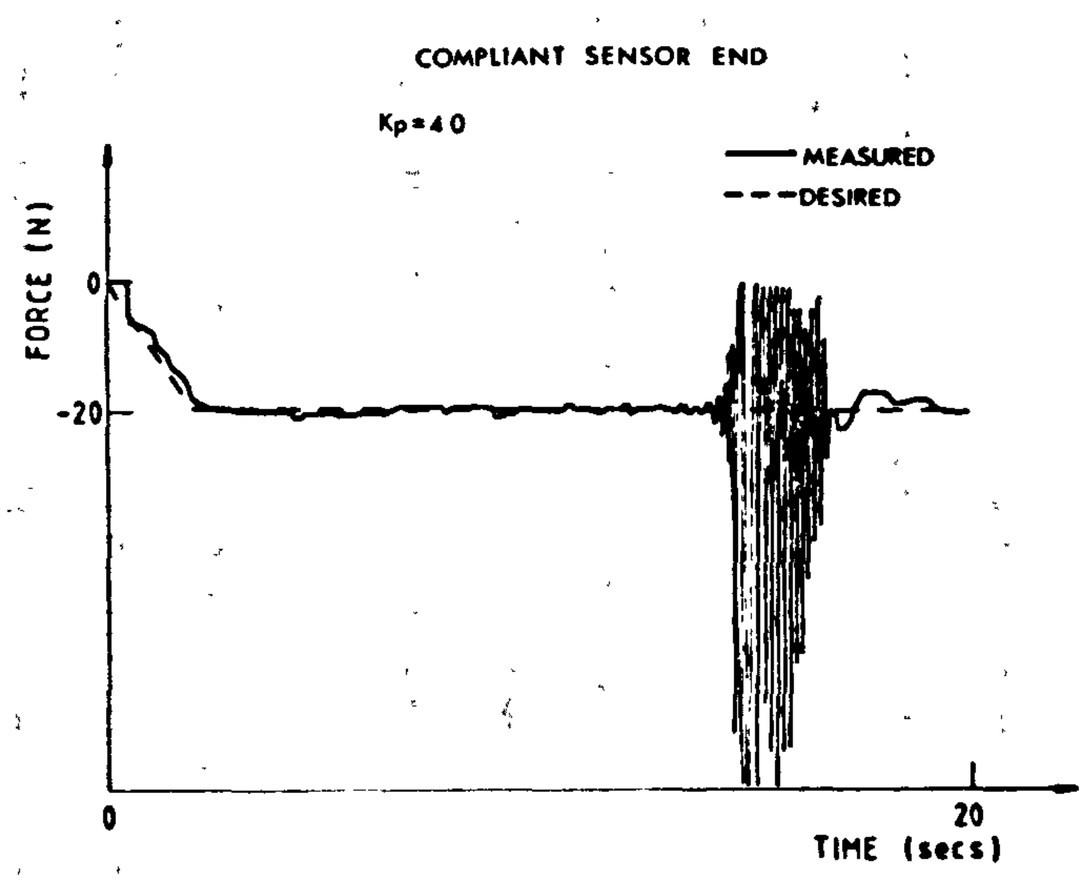
Equation 5.21 shows that if all variables, $\underline{K}_S$, $\underline{K}_E$, $\underline{K}_P$, $\underline{K}_D$, $\underline{K}_V$ remain constant then the only possible factors affecting force response are those dependent on manipulator configuration. Changes in manipulator configuration cause variation to the manipulator's inertial and Jacobian matrices and to joint based friction damping. As equation 5.21 indicates these terms should have a significant, if difficult to define, effect on force response. The following experiment has been designed to display this effects.



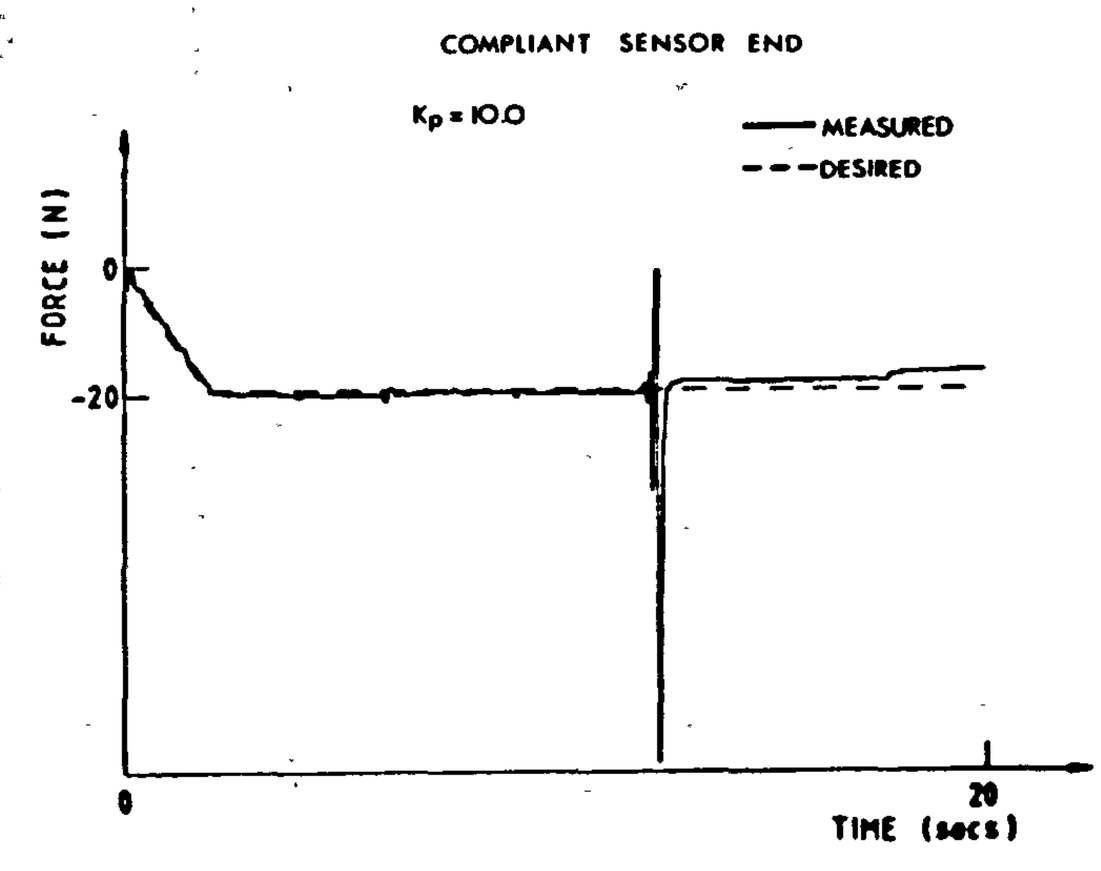
(A)



(B)



(C)



(D)

Figure 7.10. Experiment devised to display the changing manipulator configuration effects on force control. Experimental results are for the blade contour following under low force demands. Figure (a) is for a rigid sensor end and K_p=0.5units. Figures (b), (c) and (d) are results with the compliant sensor end fitted, and where gain K_p is varied as: (b) K_p=2.0units, (c) K_p=4.0units, (d) K_p=10.0units.

7.7.1 Experimental Description

The experiment involved tracking the turbine blade contour using the force sensor, complete with ball-castor and compliant element. In this way the changing stiffness of the turbine blade can be masked from the dynamics of force control. Only the changing configuration of the manipulator should then influence the results. The lowest stiffness of the turbine blade at 10^6N/m is always considerably stiffer than the compliant element's at $2x10^4$, therefore all the deflections of the blade can effectively be ignored.

The same blade following procedure described in section 7.2 and displayed in figure 7.1 is implemented here. After an initial ramp to -20N force demand, the force demand remains constant at -20N for 20 seconds, while simultaneously the manipulator is controlled to follow the blade profile.

Figure 7.10(a) is provided for comparison purposes giving the results for the rigidly mounted ball-castor. For stability, the proportional gain was reduced to K_p=0.5 units. In this case, the consequential force response was poor with a period of instability, from which the manipulator eventually recovers. The response is characterized by large force errors, caused mostly by joint friction. The superimposed small uncontrollable disturbances are caused by surface irregularities.

Figure 7.10(b), (c) and (d) feature the use of the compliantly supported ball-castor. They repeat the same experimental conditions as those used in figure 7.10(a), but differ from each other in that the proportional gain constant K_p is altered. Adding the compliant element allows an immediate increase in proportional gain, without encountering the problems of instability. Figure 7.10(b), shows the results of using

K_p=2.0 units. This shows a considerably improved force response to that shown in figure 7.10(a), less force error and less high frequency disturbance. Persistent force error is due largely to joint friction effects. However, these errors are significantly less than those in 7.10(a) owing to higher controller proportional gains.

Increasing the proportional gain further, in figure 7.10(c), where K_p=4.0 units, shows an improved force response with even less error than 7.10(b). However, instability now appears at a point where the manipulator is about to negotiate the leading edge of the blade, but recovers soon after passing this point.

Increasing proportional gain still further, gives the results of figure 7.10(d), which shows further improvement in force response over that of 7.10(c). But again instability is displayed which now occurs earlier than that in 7.10(c). The instability was too severe to proceed and consequently the manipulator was inhibited at this point to prevent damage.

7.7.2 Discussion of Results

This section illustrates the dramatic effect that the changing manipulator dynamics has on the stability of the force control. Physically, the point of instability occurs as the manipulator is contorting itself around the leading edge. This is not a kinematic singularity but an effect due to an obvious geometry change.

Stability can be improved by reducing the proportional gain constant K_p at the sacrifice of force response. In chapter 5 a similar approach to this was suggested to maintain stability everywhere. This would take the form of a look-up table, containing all the possible crude

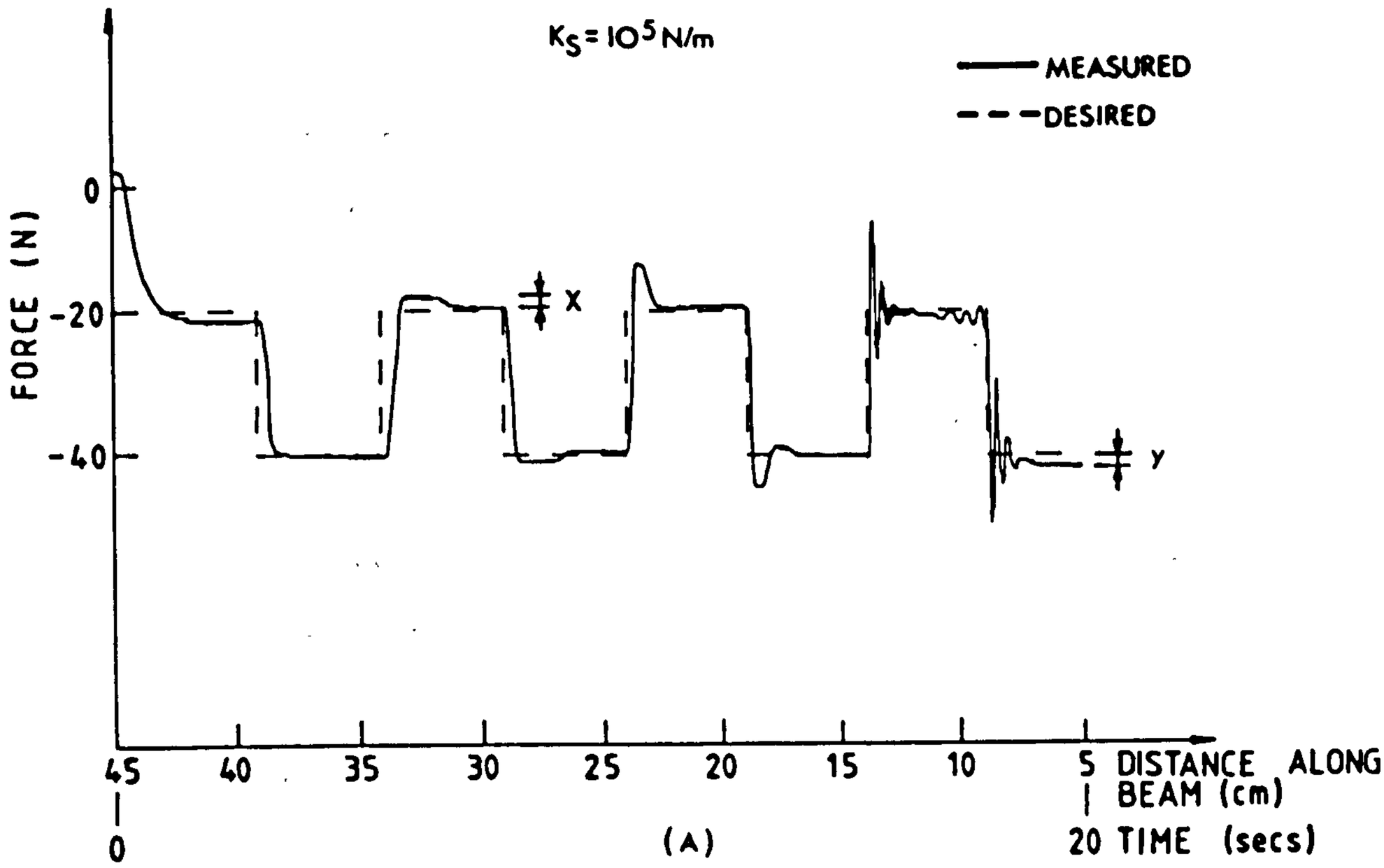
variations in manipulator geometry, against which the parameters for the stability envelope could be determined empirically. Implementing such a system has not been attempted here, but it is a suitable project for future work, and important to the general application of manipulator force control employing the Hybrid technique.

The other important observation is the reduced effect of the uncontrollable high frequency disturbance forces, displayed in figure 7.10(a) and virtually eradicated in figure 7.10(d), despite a proportional gain increase of 20 times. Again this displays an important benefit of local compliance in helping to overcome the problem of the inability to respond to disturbances above the controller's bandwidth.

7.8 Sensor Damping Effects

In the following are described results which at the time were inexplicable by the model proposed in the analysis chapter 5. The experiment used a lower value of sensor stiffness and contrary to expectations the system became less stable. Improvements in stability were expected as predicted by equation 5.16 and in [62] where a reduction in $\underline{K}_S$ should allow an increase in $\underline{K}_P$. But, as shown in section 8.6.4, reducing K_S allows the sensor damping C_S to become significant and influence the dynamics of force control. Under these conditions lowering K_E can also worsen the stability situation, contrary to earlier results of section 7.4.

A comprehensive analysis of sensor damping's influence on force control is given in single degree of freedom analysis and simulations in section 8.6.



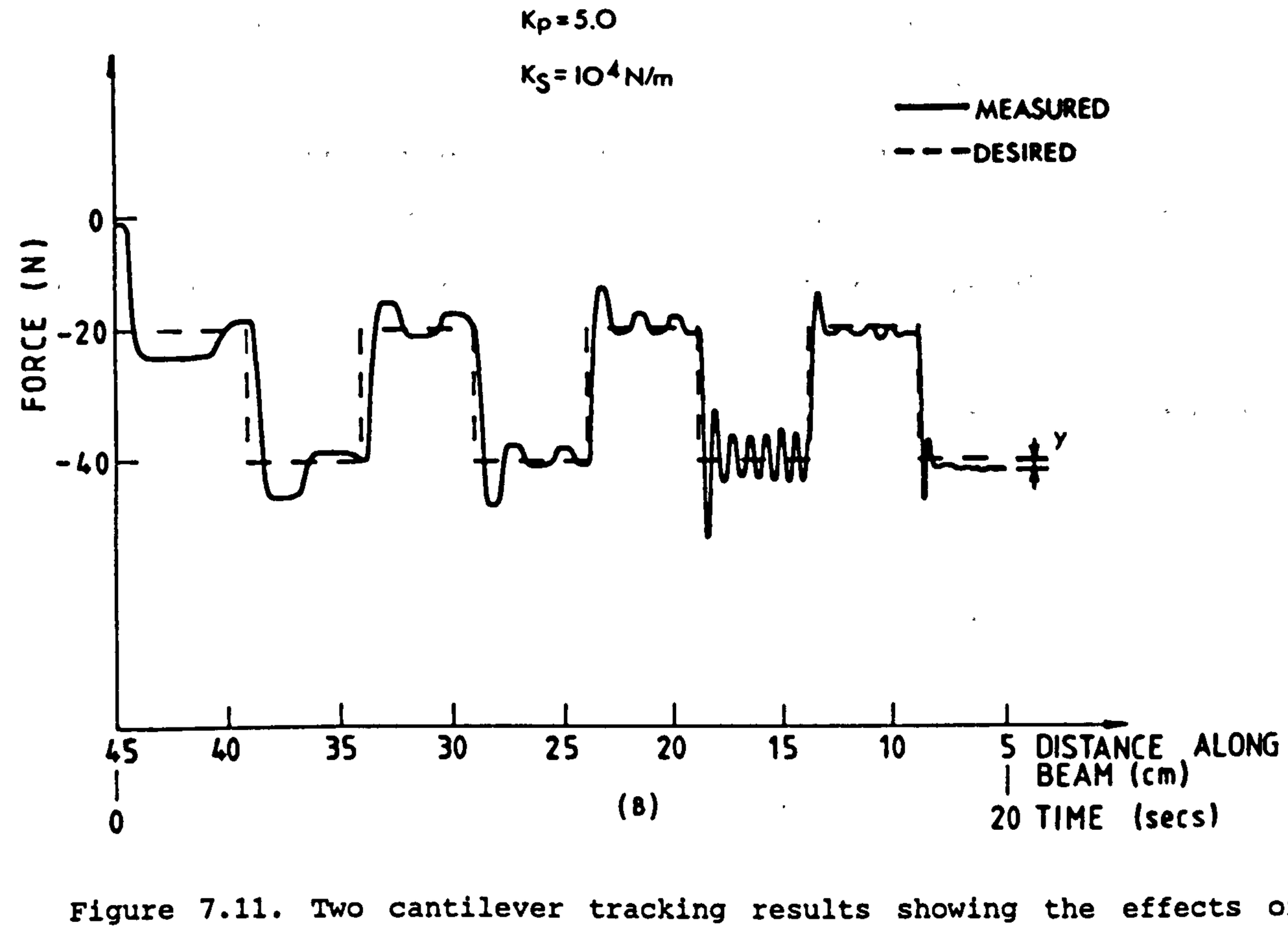


Figure 7.11. Two cantilever tracking results showing the effects of sensor damping on force control stability. All experimental conditions were identical between figures (a) and (b) except for the sensor stiffness being lowered from $K_S = 10^5$N/m in figure (a) to $K_S = 10^4$N/m in (b). This allowed the sensor damping to take effect as shown by the limit cycle in figure (b). K_p=5.0units in each case.

Two results are presented in figure 7.11. Figure 7.11(a) is a duplicate of figure 7.7(c) and is used as a comparison to figure 7.11(b). Both figures are the results from the usual cantilever following experiment, presenting an increasing environment stiffness to the manipulator as described in section 7.11. All experimental conditions were identical between figures 7.11(a) and (b) and with K_p=5.0 units in both cases. The only variable between the two experiments was the sensor stiffness, which was lowered from $K_S=10^5$N/m in figure 7.11(a) to $K_S=10^4$N/m in figure (b). This was achieved by replacing the compression spring within the force sensor with a less stiff element.

In a direct comparison between figures 7.11(a) and (b), the effect of C_S on force response is seen to be dramatic. The effect of lowering the sensor stiffness is to increase the oscillatory nature of the response in all situations except for the stiffest environment near to the cantilever support. In figure 7.11(a), for an increasing environment stiffness, the system response changes from being relatively overdamped at the cantilever free end to a more responsive system at the built-in end. The converse is true for figure 7.11(b), where stability returns at the built-in end from near instability at the free end. The effect of environment stiffness, under these conditions imposed by sensor damping, is further investigated by simulation in section 8.6.4.

A particular result for the cantilever mid-section in figure 7.11(b), where for $K_E=10^4$N/m, a lightly decaying transient of frequency 3.0±0.5Hz is displayed. This is confirmed by simulation in section 8.6.5 where the transient frequency is found to be 2.7Hz. In these simulations an improved manipulator model is used by incorporating an approximate joint viscous damping term C_R. The presence of this C_R term is shown, in simulation, to improve stability in situations where sensor damping is effective.

CHAPTER 8

SINGLE DEGREE OF FREEDOM ANALYSIS, SIMULATION AND DISCUSSIONS

8.1 Introduction

Frequently it is difficult, if not impossible, in experimentation to 'cleanly' control all independent variables, such as - joint friction and stiction, manipulator inertia, environment inertia and stiffness. As a result the experimental results are in general non-ideal and show variation from the basic analytical model. Using six degrees of freedom in the initial analysis and three degrees of freedom in subsequent experimental work, provided a useful insight into robot geometry dependencies. However, such complexities and complications limit the depth of conclusion on other fundamental aspects of the force control problem.

This chapter describes the use of single degree or freedom simulation to overcome some of these above mentioned problems and to improve the quality of the overall conclusions. The 'single degree of freedom' refers to the number of degrees of freedom constrained to force control within local coordinates. In the experimentation and simulations there is only a single coordinate constrained to force control, the normal coordinate. This is referred to as 'single degree of freedom force control'. Within this force controlled degree of freedom, which is assumed decoupled from the other coordinates, there naturally exists the usual multi-degrees of freedom of the dynamic train. Restricting the simulation analysis to a single degree of freedom allows further details to be added to the model before the problem becomes too complex. Thus an attempt can be made to explain unforeseen effects in the experimental work by simulation.

This analysis has tended to concentrate on the characteristics of manipulator force control. Therefore a simulation package 'SIMBOL' [71] has been used because of its excellent facilities in this respect. 'SIMBOL' allows the production of root locus diagrams for single degree of freedom systems. The root locus analysis shows clearly both whether or not the system is to be stable for all gains and the immediate features of the characteristic. The root locus diagrams are used throughout.

As the simulation analysis develops its relevance to the experimental and analytical studies is discussed.

The simulation analysis [65] Eppinger and Seering on the dynamics of a single degree of freedom force control problem, has been recently published. But the results are inconclusive and fail to explain the problems of achieving stability encountered within this work. Their simple lumped mass models representing both robot and environment, see figure 8.1 for details, showed stability to exist for all values of controller gain. Experimentation in this research has found this not to be strictly correct. This, in addition to other aspects of manipulator force control, is considered in the following single degree of freedom analyses, simulations and discussions.

8.2 Modelling

A linearized single degree of freedom model is presented here. This is a reasonable assumption because any resulting movement within force control tends to be limited and consequently linear. Thus system inertia's are unchanging and considered in the simulation model as lumped masses.

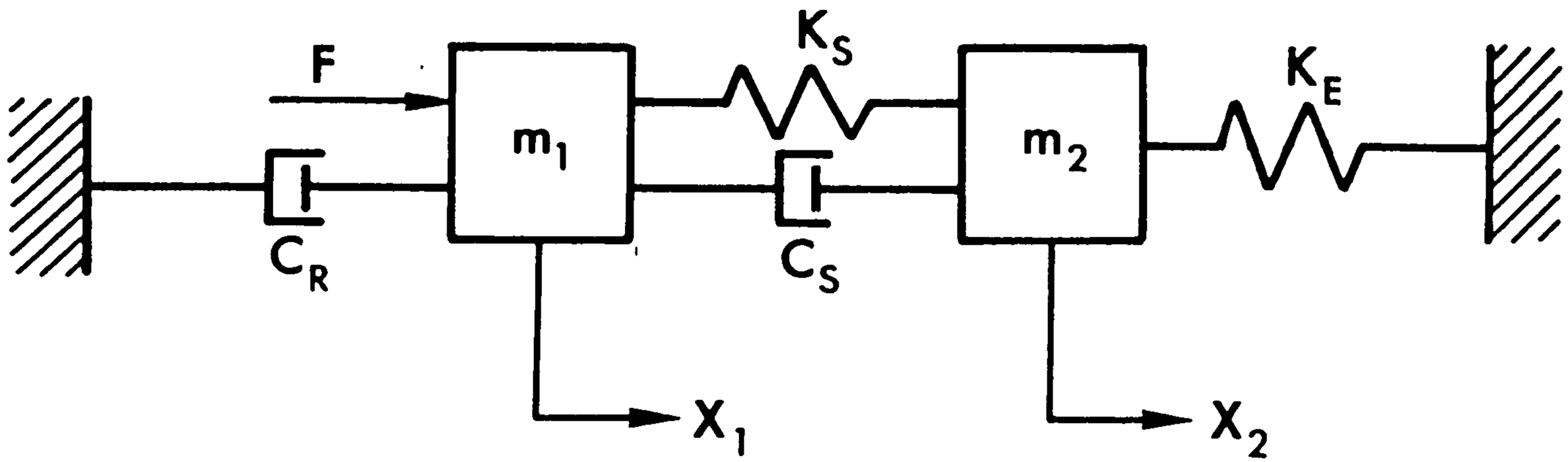


Figure 8.1. Basic mechanical model of the single force controlled degree of freedom system. Mass M_1 represents manipulator inertia, mass M_2 the combined sensor/environment inertia.

Chapter 5 discusses the reasons for considering any flexibilities in the manipulator's transmission as rigid. This allows the combination of the actuator and link masses, modelled as M_1, see figure 8.1. Sensor mass is combined with the effective environment mass and is represented by M_2. Sensor and environment stiffness are represented, as usual, by K_S and K_E respectively. Viscous damping in the robot and sensor is represented by C_R and C_S. The controlled actuator force F, delivered by the actuators, acts to produce a compressive sensor force according to:

$$F= -K_F(F_D-F_S) \qquad (8.1)$$

where K_F is the force controller containing only simple proportional gain, F_D is desired force and F_S the sensor force.

Mass displacements from mean positions are given as X_1 and X_2, referring to M_1 and M_2 respectively. Sensor force is determined from measuring the relative displacement (X_2-X_1) multiplied by the sensor stiffness, K_S.

The equations of motion can be written for the robot, mass M_1, as:

$$M_1\ddot{X}_2 = -K_F\{F_D-K_S(X_2-X_1)\} - (X_1-X_2)K_S - (\dot{X}_1-\dot{X}_2)C_S - \dot{X}_1C_R \qquad (8.2)$$

and for sensor mass, M_2 as:

$$M_2\ddot{X}_2 = (X_1-X_2)K_S + (\dot{X}_1-\dot{X}_2)C_S - K_EX_2 \qquad (8.3)$$

These equations are represented in matrix form as:

$$\begin{bmatrix} [M_1S^2+(C_R+C_S)S+K_S(1+K_F)] & -[C_S+K_S(1+K_F)] \\ -[C_SS+K_S] & [M_2S^2+C_SS+(K_E+K_S)] \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} -K_FF_D \\ 0 \end{bmatrix} \qquad (8.4)$$

where S is the Laplace transform variable.

If X_2 is replaced as a dependent variable by F_S where $X_2=F_SK_S^{-1}+X_1$, then the resultant equation is exactly comparable to equation 5.8, but now contains the additional damping terms, C_R and C_S. Thus the characteristic matrix is:

$$\begin{bmatrix} [M_1S^2+C_R] & -[\{C_S+K_S(1+K_F)\}K_S^{-1}] \\ [M_2S^2+K_E] & [\{M_2S^2+C_SS+(K_E+K_S)\}K_S^{-1}] \end{bmatrix} \begin{bmatrix} X_1 \\ F_S \end{bmatrix} = \begin{bmatrix} -K_FF_D \\ 0 \end{bmatrix} \tag{8.5}$$

8.2.1 Characteristic Equation

The force controller K_F is replaced by its individual components, where $K_F = (K_P+K_D.S)$, then the system eigenvalues are found from the roots of the determinant of equation 8.5. Expanding this determinant gives the characteristic equation as:

$$\begin{aligned} M_1M_2S^4 + [M_1C_S+M_2(C_R+C_S) + M_2K_S.N.K_C.K_D]S^3 \\ +[M_1(K_E+K_S) + M_2(NK_CK_P+1)K_S + C_RC_S]S^2 \\ +[C_R(K_E+K_S) + K_EC_S + K_EK_SNK_CK_D]S \\ + K_EK_S(1+NK_CK_P) = 0 \end{aligned} \tag{8.6}$$

This is directly comparable to equation 5.21, but with extra term dependent on C_S.

Equation 8.6 can be seen to contain odd powers of S, necessary to confirm stability. The odd terms are composed of K_D, C_S and C_R dependent terms. Hence if the K_D term is not actually present, stability may still be possible due to the presence of velocity dependent terms alone.

Attempts at a Routh-Hurwitz analysis of the complete system characteristic of equation 8.6, to investigate the combined role of all damping terms, proved extremely complex and inconclusive. Further Routh-

Hurwitz analyses, described in sections 8.4 and 8.6, investigate separately the individual effects of the C_R and C_S terms, and produce definite conclusions.

8.2.2 Modelling Parameters

The experimental rig was used as the basis for choosing the physical values of the simulation parameters. The model parameters of figure 8.1 and of the simulation are only approximations to those of the mechanical manipulator. The approximations arise from the influences of manipulator geometry. As the manipulator geometry varies, so the joint inertia and viscous effects, transformed to sensor coordinates, change also. In addition to this are the changing link inertia effects and the effects of variation in kinematic and force transmissions. But, providing the parameters are 'realistic', that is, they are of the correct order of magnitude and of relative value, then the simulations are adequate in expressing the fundamental trends of force control, rather than confirming any specific results. Also, the model parameters are assumed unchanging for the duration of the force controlled task. This being the case, the following are estimates of the manipulator's physical parameters.

The high reduction of the ballscrew drives has a surprisingly significant effect on the manipulator inertial parameter, M_1, previously referred to as $(\underline{N}^2\ \underline{I}_A+\underline{I}_L)$ in earlier chapters. The component I_L is approximately 10 to 15Kg, but the N^2I_A term is around 2500Kg owing to the amplification of I_A by the N^2 term. The combined sensor/tool and environment mass lies typically between 0.5kg and 3 Kg, depending on the nature of the contacting environment, and is very much less than the effective manipulator mass. Force sensor damping, C_S, due to the sensor guide is assumed to be viscous, and has been measured as $360N/ms^{-1}$. The

manipulator joint damping, C_R, as discussed earlier in chapter 5, is difficult to model, but a viscous component is assumed and will be investigated further in simulation.

Although there is an unknown viscous joint friction component within C_R whose effect is highly non-linear and unpredictable, there is also another viscous damping component due to the back emf. or torque regulation of the D.C. motors. Whether the latter effect dominates the friction viscous component or not, it nevertheless has a surprising effect once transformed from actuator to sensor coordinates. As a first approximation only the motor regulation effect is considered to be the minimum dominant term of C_R. Appendix B gives motor regulations as 1.21×10^{-3} Nm/rev/min, transformed to sensor coordinates it has an effective damping coefficient of $1.82 \times 10^{4} N/ms^{-1}$.

Realistic controller parameters used in the simulations are determined from the computer program implemented to control the experimental robot. To understand how the parameters are determined consider figure 8.2 which is a simplified control block diagram displaying the key features of the force controller. The robot and sensor dynamics have been omitted, these are given elsewhere in figure 8.5. The diagram shows the important combination of the proportional gains of the motor, servo amplifier and software joint gains to form the overall actuator gain K_C, which converts a motor torque demand into motor torque. The gain K_C represents the same property as that described in the analysis chapter 5 and here in its single degree of freedom form takes the numerical value 0.025. Combining K_C with the gain N of the gearbox, or ballscrews in this case where N = 1256[N/Nm], gives an overall joint gain of NK_C = 31.7[N/Nm]. This gain is finally transformed to local or tool coordinates by the robot structure, the inverse of the J^T transform. Therefore when duplicating the experimental gains in the simulations,

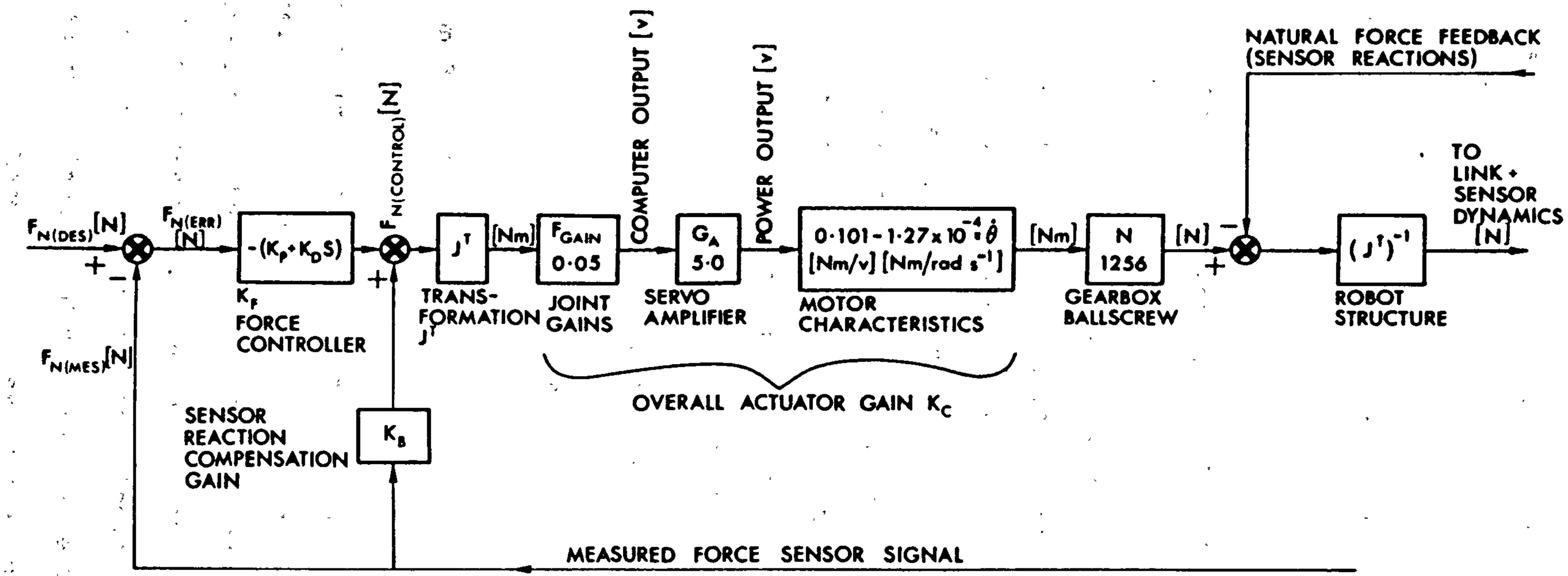


Figure 8.2. Block diagram representing the main features of the experimental manipulator control system.

the experimental force controller parameters should be multiplied by NK_C = 31.7 to obtain the equivalent single degree of freedom model transformed to tool coordinates.

The list of nominal parameters used in the simulations is given in table 8.1

Manipulator inertia, M_1=2500[Kg]

Sensor inertia, M_2=0.5to3[Kg]

Sensor damping C_S=360[N/ms$_{-1}$]

Robot damping C_R=1.8x10^4[N/ms^{-1}]

Overall actuator gains, NK_C=31.7[N/Nm]

Sensor stiffness K_S=variable[N/m]

Environment stiffness K_E=variable[N/m]

Proportional controller gain K_P=variable

Derivative controller gain K_D=variable

Sensor reaction elimination constant K_B=variable

Nominal Simulation Parameters

Table 8.1

8.2.3 Simulation Block Diagrams

Control system information is input into 'Simbol' in the form of control block diagrams. Each block within the control system can either be a polynomial representing the dynamics of the block, or be a block diagram itself, called a sub-system. The control system flow paths can be achieved graphically by connecting the blocks at will. Additions, subtractions, function inputs and multi-variable plotting are functions readily available. These facilities create an extremely easy and understandable program environment for data input and output. In the

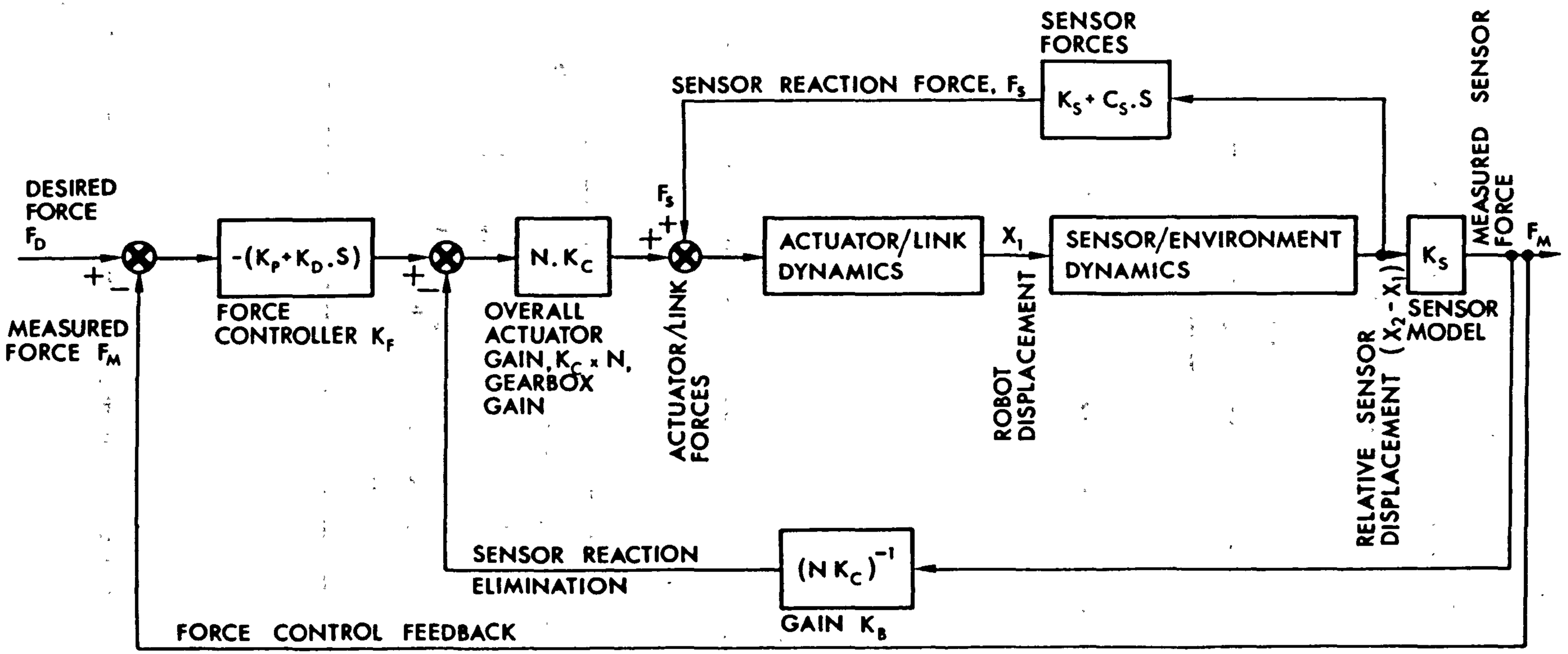


Figure 8.3. Block diagram representing the simulated single degree of freedom force controlled system.

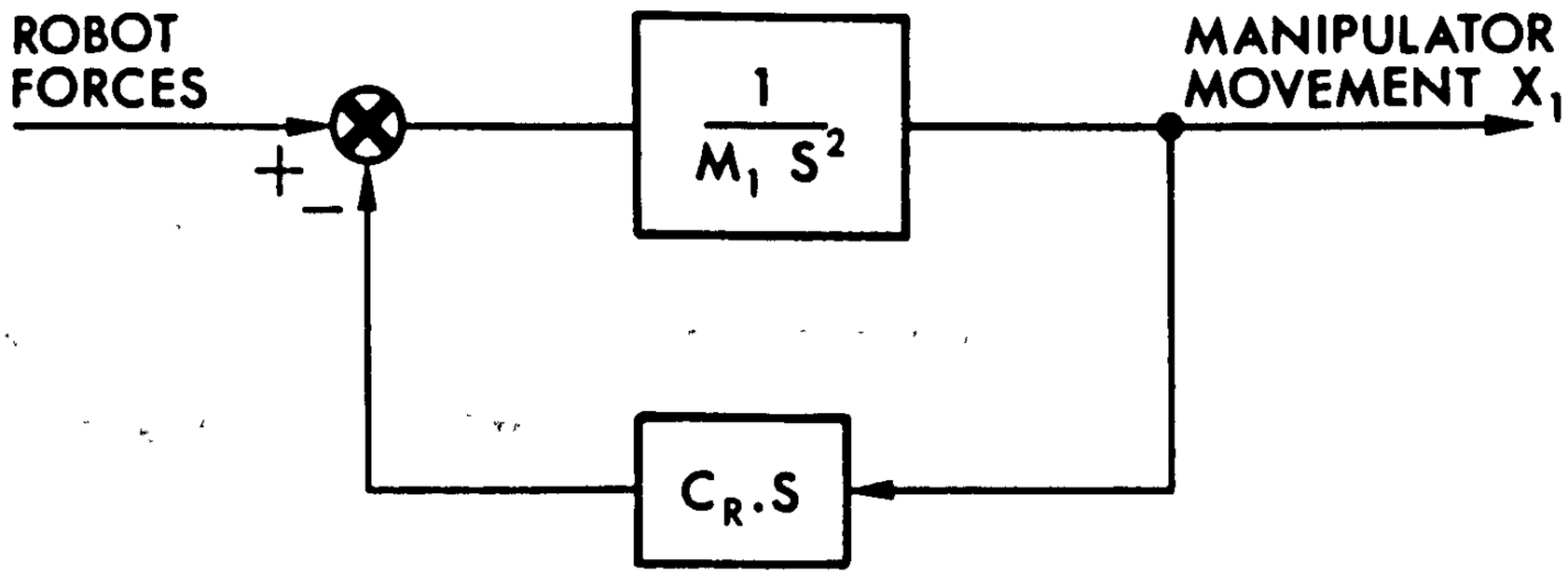


Figure 8.4. Functional block diagram of actuator/manipulator subsystem of figure 8.3

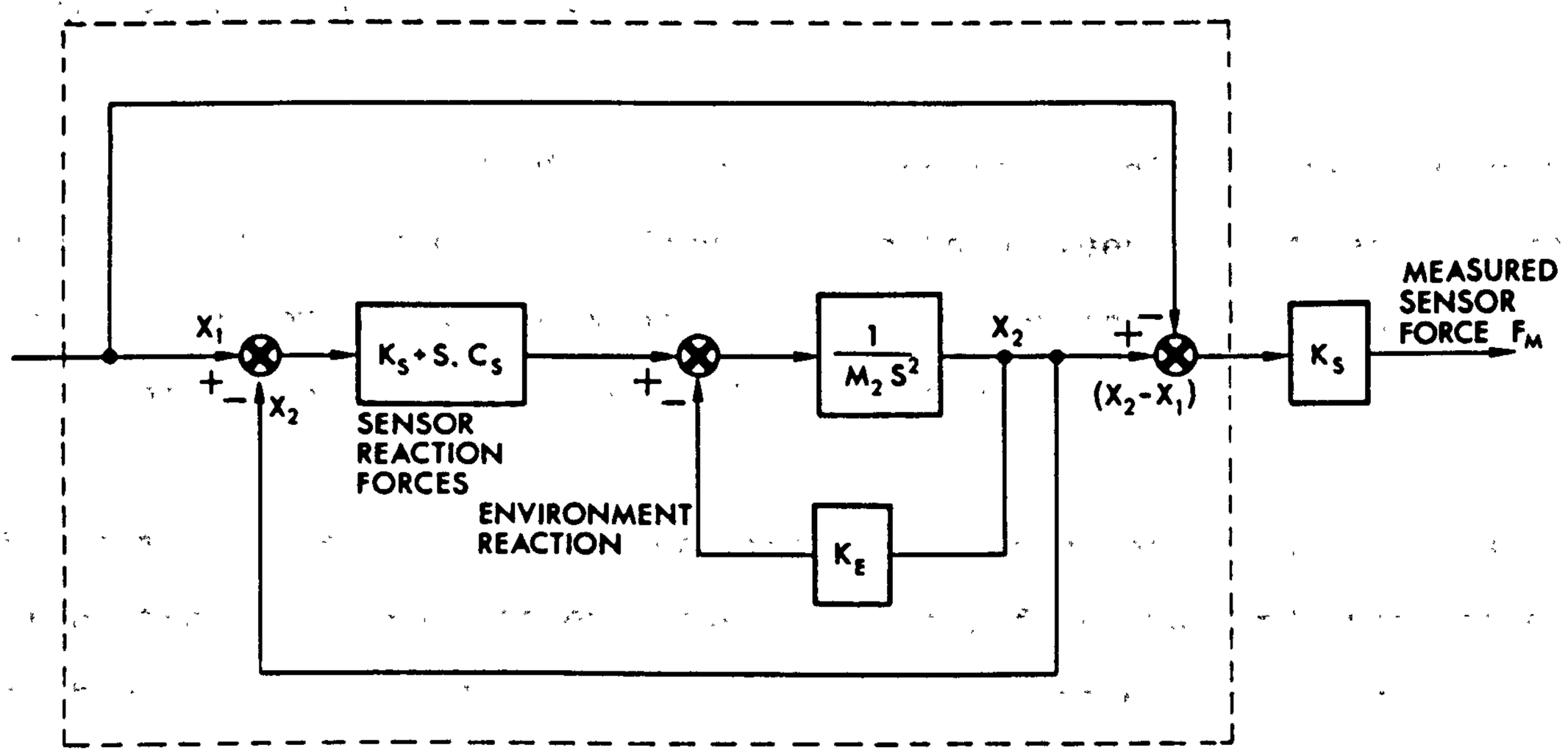


Figure 8.5. Functional block diagram of environment/sensor subsystem of figure 8.3.

analysis considered here root locus diagrams are particularly useful and easily generated by Simbol.

The complete control system shown in figure 8.3 can be conveniently broken into its subsystems representing, the controller, manipulator and environment/sensor dynamics.

The force controller is shown with its two basic elements, the proportional term K_p and the derivative action represented by K_D. The force controller produces an actuator force F. Actuator/manipulator dynamics are seen to be represented by the mass M_1 of figure 8.1 producing a displacement X_1 under the combined action of the force controller force F and sensor reaction forces. The actuator/manipulator subsystem is shown in more detail in figure 8.4, where provision is made to incorporated manipulator damping C_R.

For the single degree of freedom case and for manipulators employing parallel drive systems, the damping C_R acts between mass M_1 and earth. But for manipulators employing serial drive configurations, C_R acts on the relative velocities between adjacent links.

The combined sensor/environment inertia M_2 experiences two forces, a force from the environment, due to the stiffness K_E, and the sensor reaction forces. The sensor is considered to have stiffness K_S, as before, and also a viscous damping term C_S. The functional block diagram for the sensor/environment is arranged as figure 8.5. This shows the production of a measured sensor force F_M created by a relative manipulator/sensor movement (X_2-X_1) and stiffness K_S. The sensor reaction forces F_S are caused by the relative displacement of X_1 and X_2 together with their relative velocities.

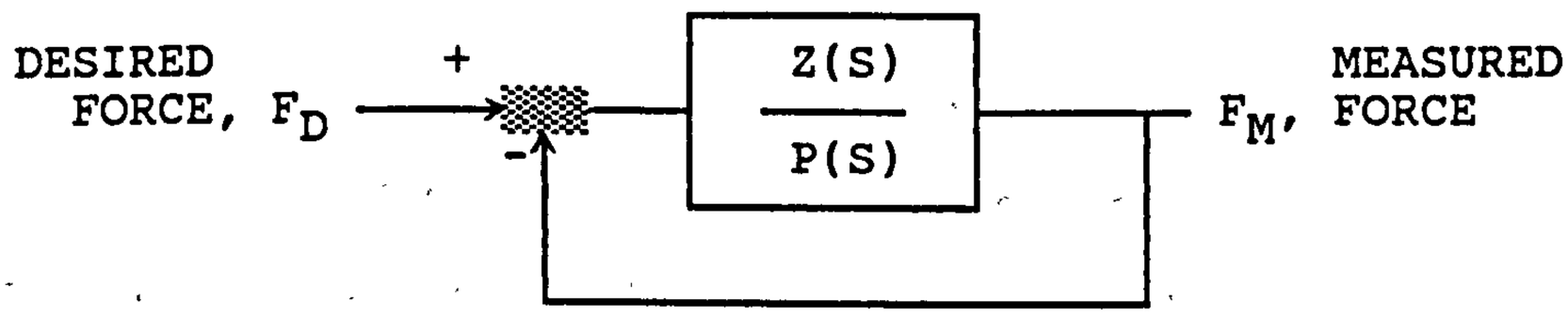
8.2.4 Root Loci Diagrams

The root locus diagrams presented as simulation results show the locus of the roots to the closed loop characteristic equation, as the overall open loop gain is increased from zero to infinity. The locus begins at a pole, marked by an 'X', and ends at a zero marked by an 'O' or tends to infinity along an asymptote. In keeping with convention, the real axis is shown as the abscissa - and the imaginary axis as the ordinate. Roots lying on the right hand side of the imaginary axis of the locus diagram are unstable roots.

Using Simbol it is possible to pick off the actual transient frequency of the loci by positioning the cursor on the locus and asking the program to generate the gain of the system for that point. At this point the ordinate of the cursor is displayed as the transient frequency at that gain.

8.2.5 Open Loop Poles and Zeros

Excluding the sensor reaction compensation loop from figure 8.3 and reducing the figure further to a single block system with unity feedback, gives figure 8.6.



where:

$$Z(S) = (M_2.S^2+K_E)(K_P+K_D.S)$$

$$P(S) = (M_1.S^2+C_R.S)(M_2.S^2+C_S.S+K_E+K_S) + (K_S+C_S.S)(M_2.S^2+K_E)$$

Z(S) are the open loop zeros of the system and P(S) the open loop poles.

Figure 8.6

The zeros Z(S) are seen to be dependent on the force controller, sensor mass dynamics and environment stiffness only. The poles are in general dependent on all system parameters. The placement of the zeros raises an interesting problem. The zero associated with the controller indicates a negative real value always, whereas the zeros associated with the sensor mass are always imaginary, undamped oscillatory placements despite the presence of C_R, C_S, and K_D terms. These zeros should move off the imaginary axes if sufficient environment damping is present. In practice this condition is unlikely and certainly cannot be relied upon. Therefore, the force controlled system should always tend to the limit of stability as the overall system gain approaches infinity and the roots tend to the terminating zeros.

Interestingly, the forward path proportional gain is the product $NK_CK_SK_EK_P$ which under certain circumstances can dominate the characteristic equation if the individual terms become excessively large. In agreement with this increases in proportional gain, environment and sensor stiffnesses were shown experimentally to both raise transient frequency and reduce damping of the transient response.

8.3 Undamped System

A useful exercise for later comparisons is to firstly consider the important aspects of the fundamental force controlled system freed from the influences of any form of damping. The simulation model is the same as that of figure 8.6 except all forms of damping, C_S, C_R and K_D are set to zero.

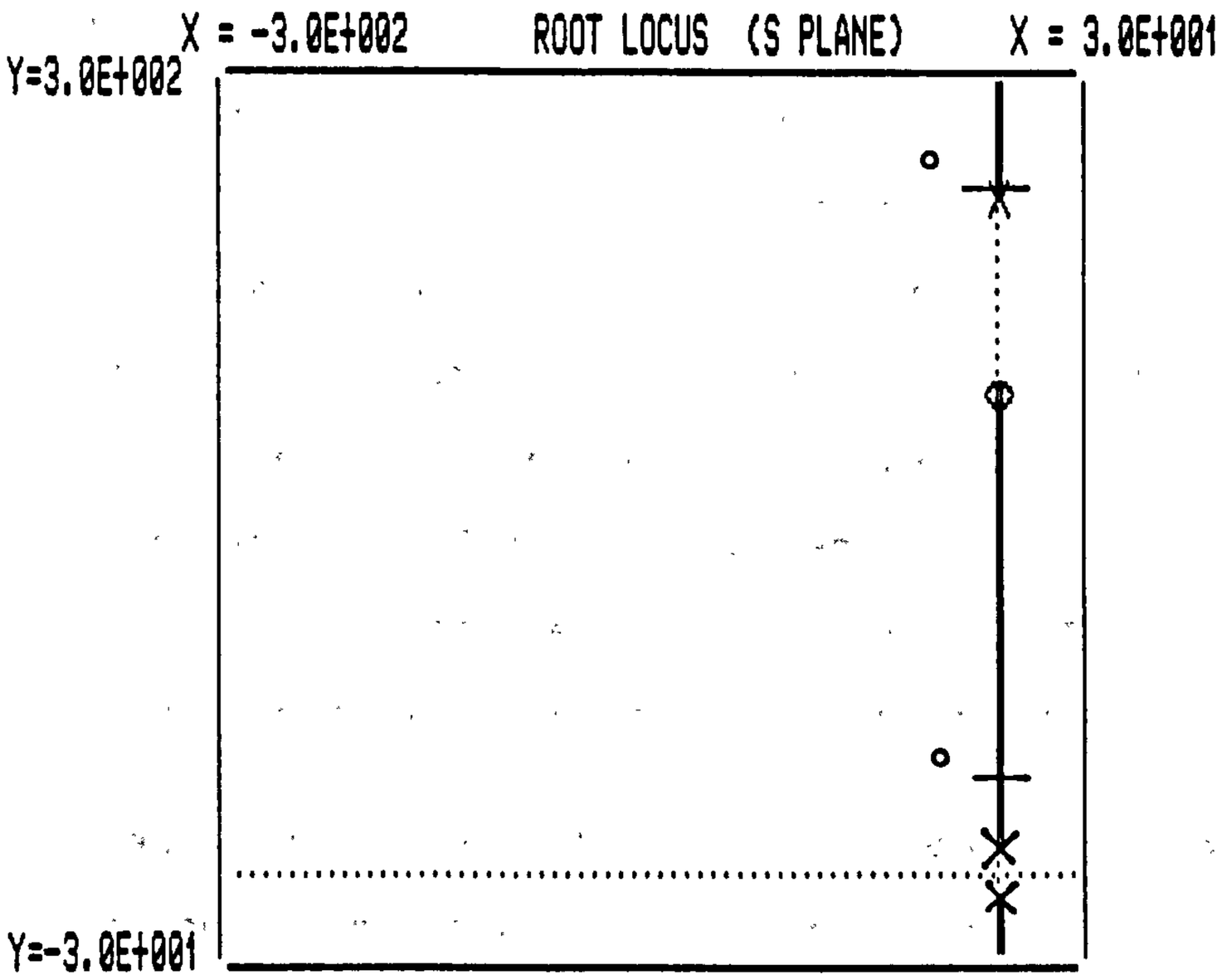


Figure 8.7. Simulation showing the root locus plot of the basic undamped, single degree of freedom, force controlled system. $K_S=10^5$N/m, $K_D=0$, $C_R=0$, $K_E=10^4$N/m, operating point is shown at $K_P=0.3$ units. All other parameters take values defined in table 8.1.

Reducing figure 8.6 to a single block diagram gives the characteristic equation as:

$$M_1M_2S^4 + \{M_1(K_E+K_S) + M_2(K_PK_CN+1)\}S^2 + K_EK_S(1+K_PK_CN) = 0 \qquad (8.7)$$

Having no odd terms in 'S' shows the system to be at the limit of stability. In fact the root locus shown in figure 8.7 exists only along the imaginary axis, indicating zero damping on all roots, confirming the theoretical limit of stability for all system gain. However in the absence of negative real parts to the roots, instability will arise in practice. The parameters used in figure 8.7 are: $K_P=0.3$, $K_E=10^4$, $K_S=10^5$. The diagram, as expected, displays 4 poles and 2 zeros, indicating 2 asymptotes which lie along the imaginary axes. The root loci are divided effectively into two transients; a low frequency locus starting close to the real axis, and a much higher frequency locus with poles at around 190 rad/sec and extending to infinity. The low frequency transient is associated with the gross force control of the manipulator system, having transient frequency of approximately 13 rad/sec. The upper frequency transient is associated with sensor/environment vibration which is effectively a harmonic on the fundamental transient. At a frequency of approximately 200 rad/sec it is easily identified as the natural frequency, w, of the sensor system, which is given by:

$$w = \left[\frac{K_E+K_S}{M_2} \right]^{\frac{1}{2}}$$

8.4 Viscous Damping

Adding to the undamped system of section 8.3 the velocity damping component C_R, caused by the motor back emf. regulation, gives the system represented by equation 8.6. The motor regulation has been transformed to sensor coordinates. Both the C_S and K_D component remain set to

zero. The characteristic equation then becomes:

$$M_1M_2S^4 + M_2C_RS^3 + \{M_1(K_E+K_S) + M_2(K_PK_CN+1)\}S^2 + \{C_R(K_E+K_S)\}S + K_EK_S(1+K_PK_CN) = 0 \quad (8.8)$$

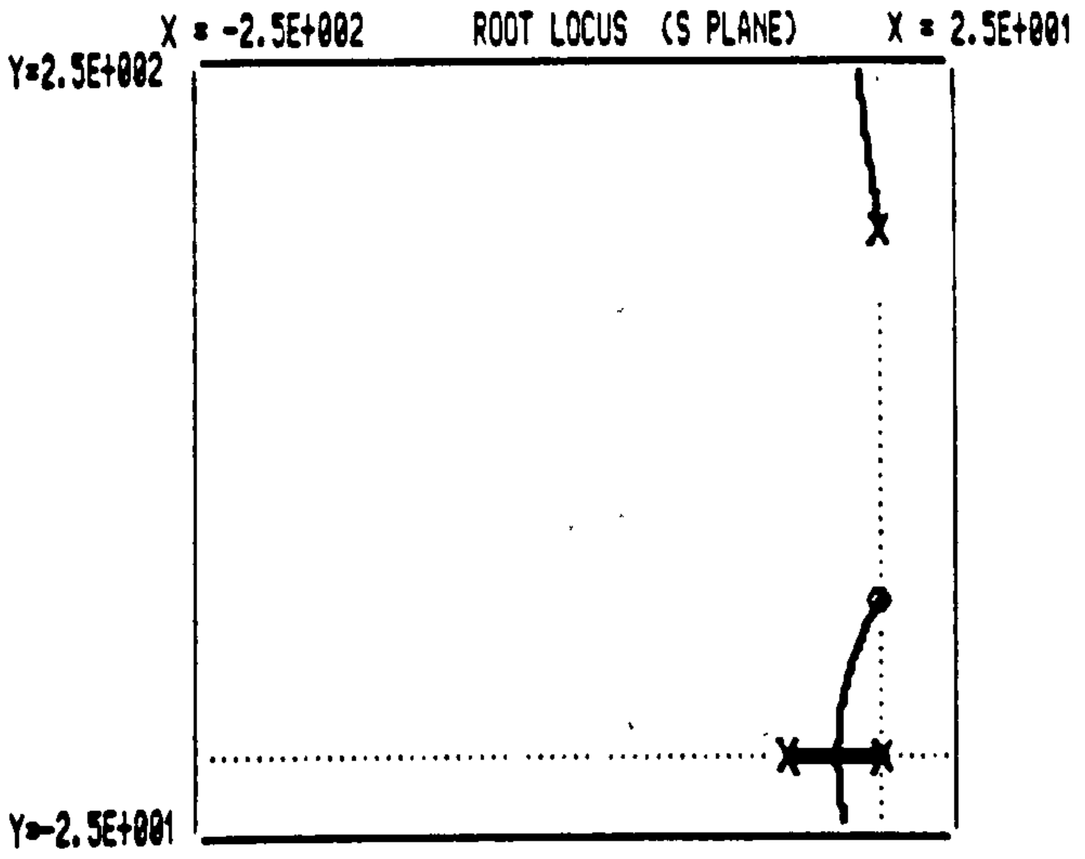
The Routh Hurwitz analysis given in Appendix A shows that under these conditions the system is always stable providing

$$M_2^2C_R^2K_S^2(K_PK_CN+1) > 0$$

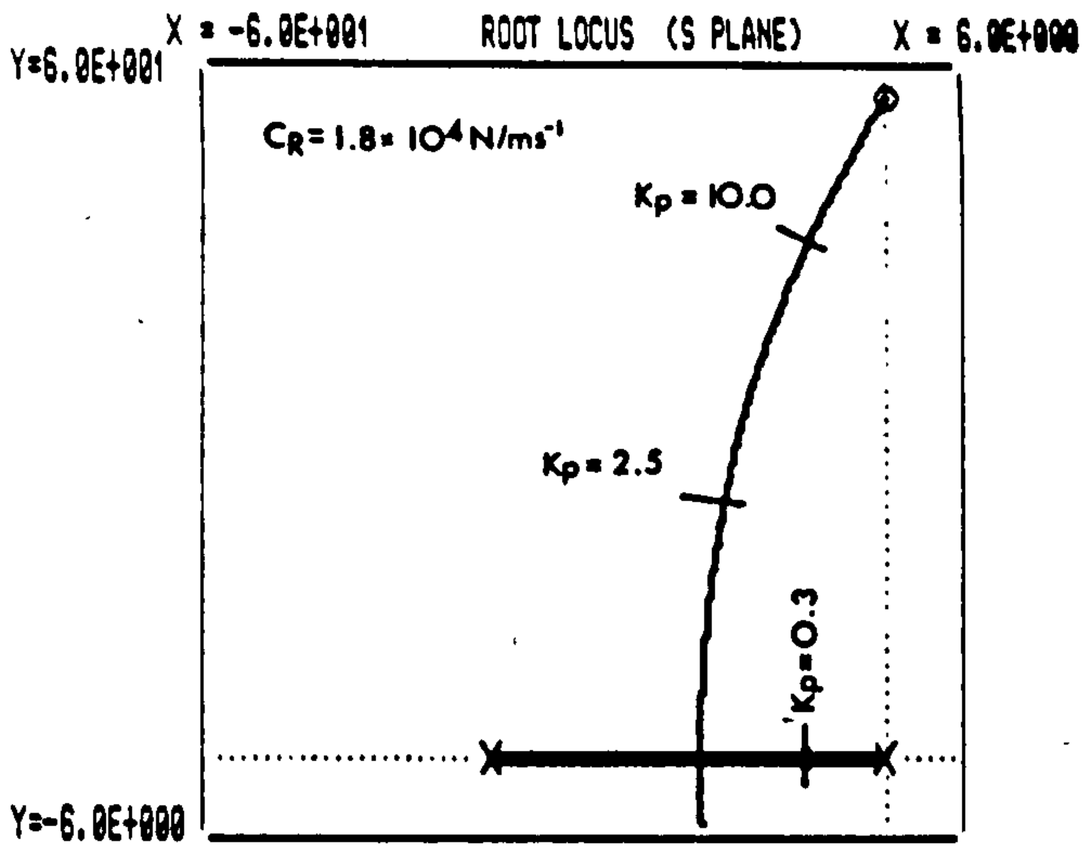
ie. for values of $K_PK_CN>-1$.

Figure 8.8(a) shows the simulated system with viscous robot damping at $C_R=1.8x10^4N/ms^{-1}$. The system displays 4 poles and 2 zeros, equal in number to those of the undamped system. However, the asymptotes of the loci have moved in to the negative half of the 'S' plane, away from the imaginary axis, indicating C_R damping improves stability. The most significant change is to the low frequency locus component. This locus has been moved significantly towards the left and at a proportional gain set at $K_P=0.3$, the system is considerably over damped. As the proportional gain of the system K_P is increased further, then the system begins to become oscillatory, agreeing with the sequence of results shown in figure 7.7 of the experimental chapter.

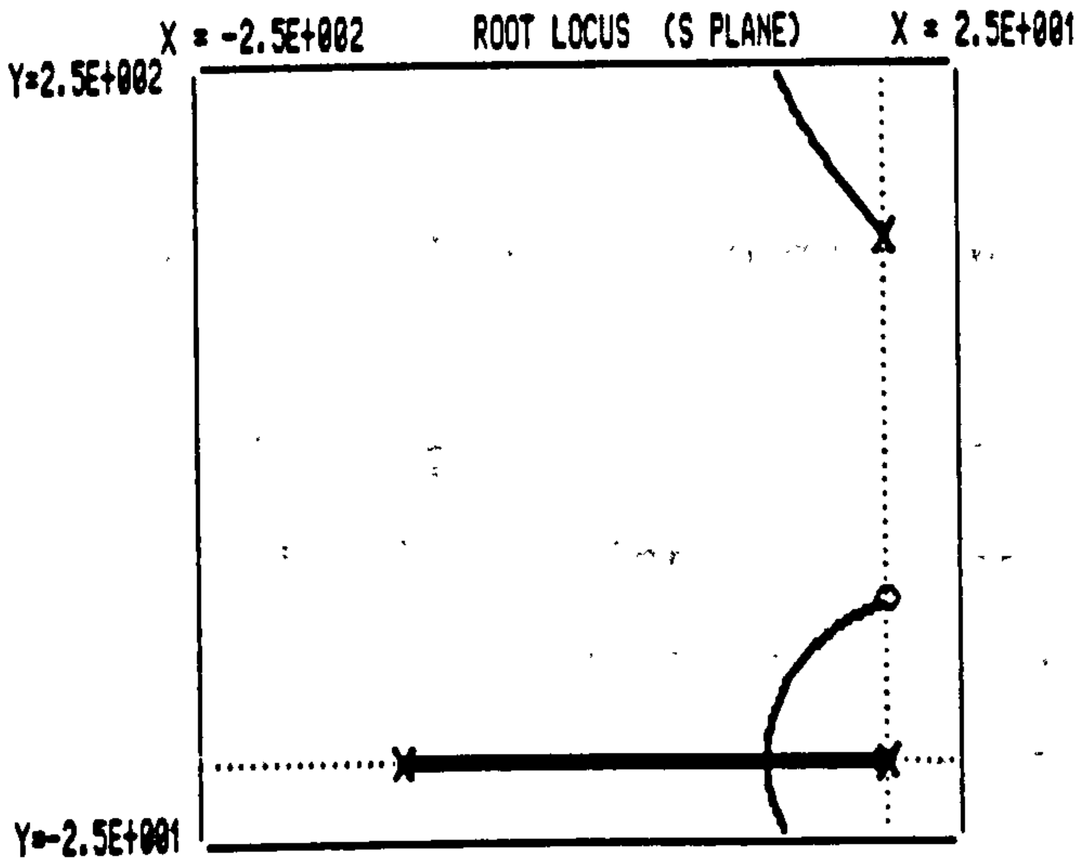
Figure 8.8(b) shows the same result as 8.8(a) but the locus of the low frequency transient is enlarged for clarity. Three points are marked on this locus which correspond to $K_P=0.3$, 2.5, 10.0, indicating increasing transient frequency for increasing gain. Figure 8.8(c) shows the effect of increasing the value of C_R five fold to $9.0x10^4$. Significantly, only one pole is affected by increasing C_R, moving it increasingly towards



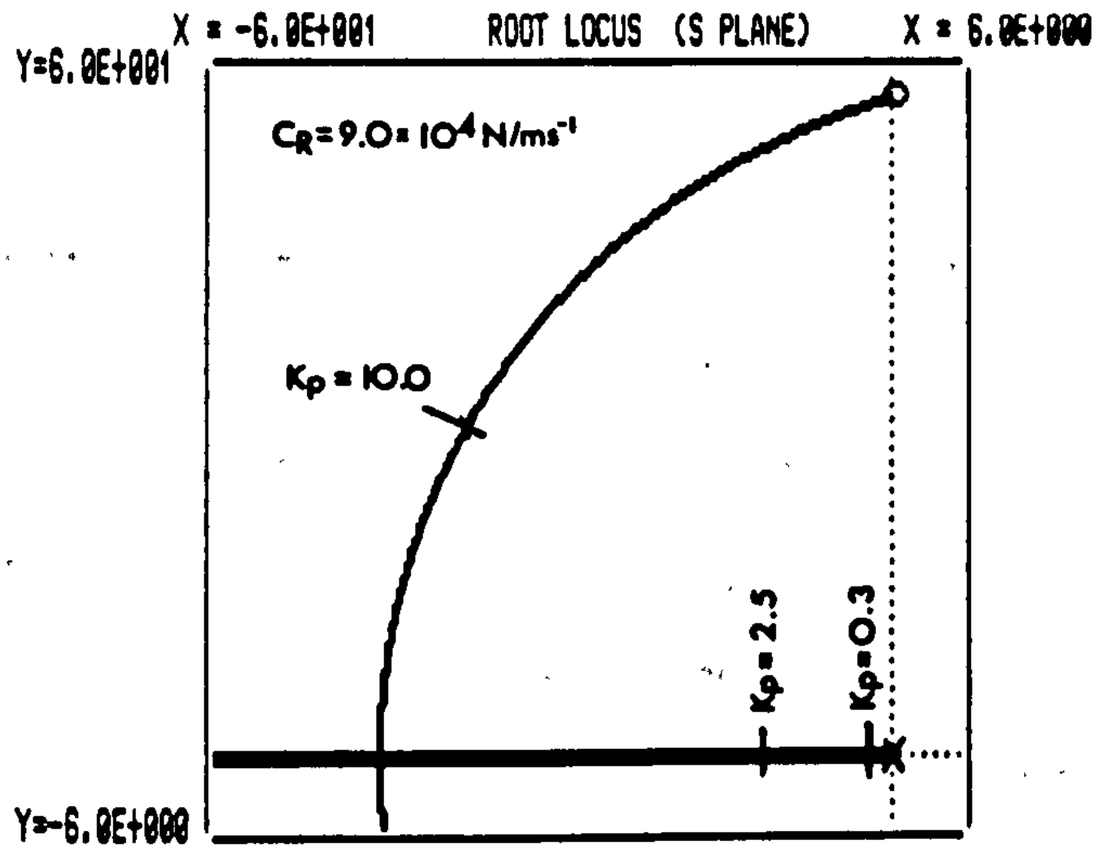
(A)



(B)



(C)



(D)

Figures 8.8(a) to (d). Simulations showing the root locus plots of the viscously damped, single degree of freedom, force controlled system. Viscous damping C_R is the variable.

$K_S=10^5$N/m, $C_S=0$, $K_D=0$, $K_E=10^4$N/m. Different operating points are shown at $K_P=0.3$, 2.5, 10.0 units. All other parameters take values defined in table 8.1. Figures: (a)&(b) $C_R = 1.8\times10^4$N/ms^{-1}, (c)&(d) $C_R = 9.0\times10^4$N/ms^{-1}

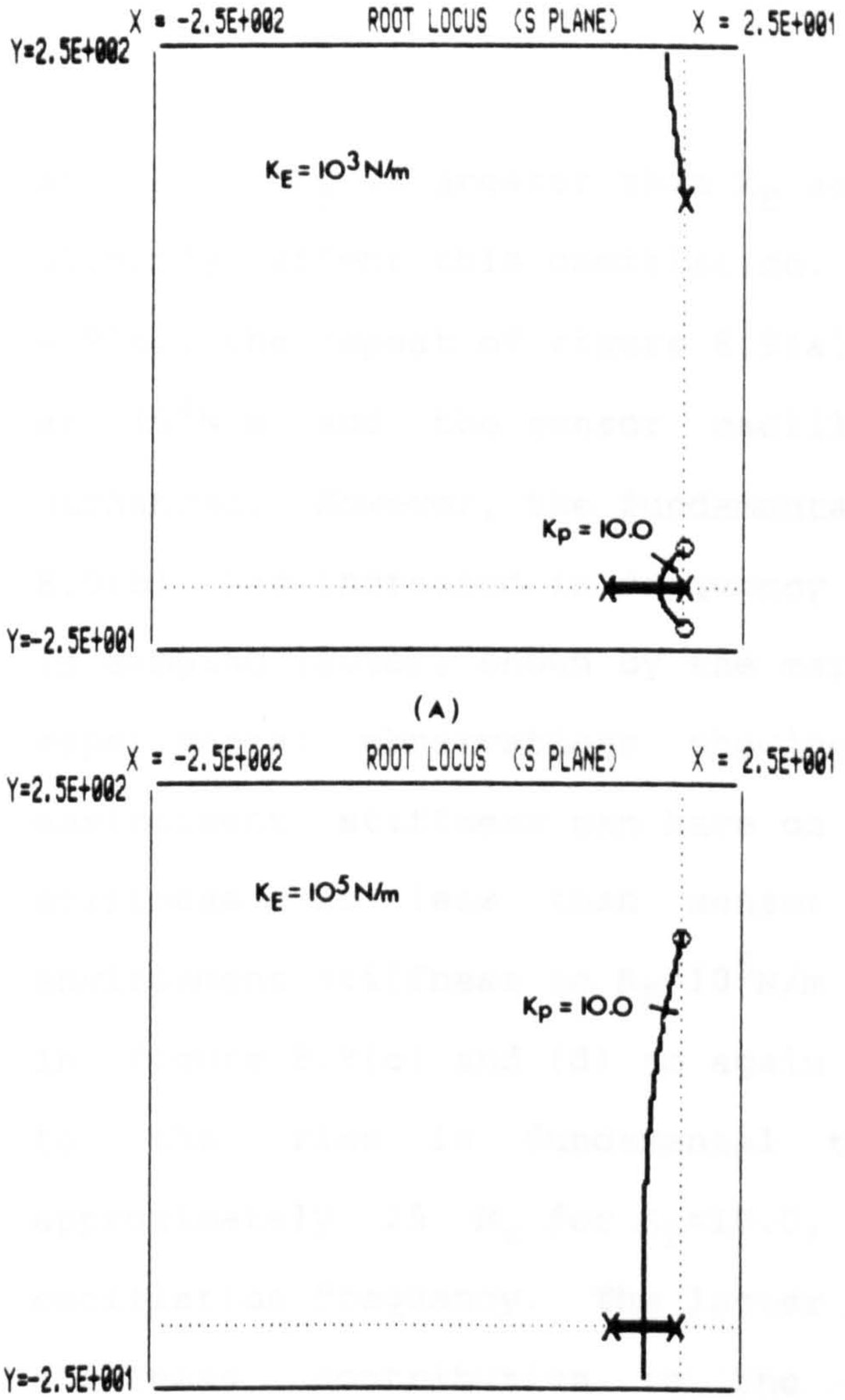
the left. This also results in the asymptotes being moved towards the left. The points of proportional gain K_p=0.3, 2.5, 10.0 are correspondingly marked on figure 8.8(d). This shows increasing C_R reduces the transient frequency and increases the damping factor. In fact K_p=0.3 and 2.5 both correspond to overdamped systems. This trend confirms the effect shown in figure 7.8, in the experimentation chapter, where increases in velocity damping, to simulate viscous friction, increasingly damped the response for low values of environment stiffness K_E.

8.4.2 Effect of Environment and Sensor Stiffness on Viscous Damping

The following sequence of simulations shows how variation in environment stiffness affects force response where viscous type damping is present. The system studied is again based on the usual model described in figure 8.6 but where $K_S=10^5$N/m, $C_R=1.8x10^4$N/ms^{-1}, C_S=0.0 and K_D=0.0, unless otherwise stated.

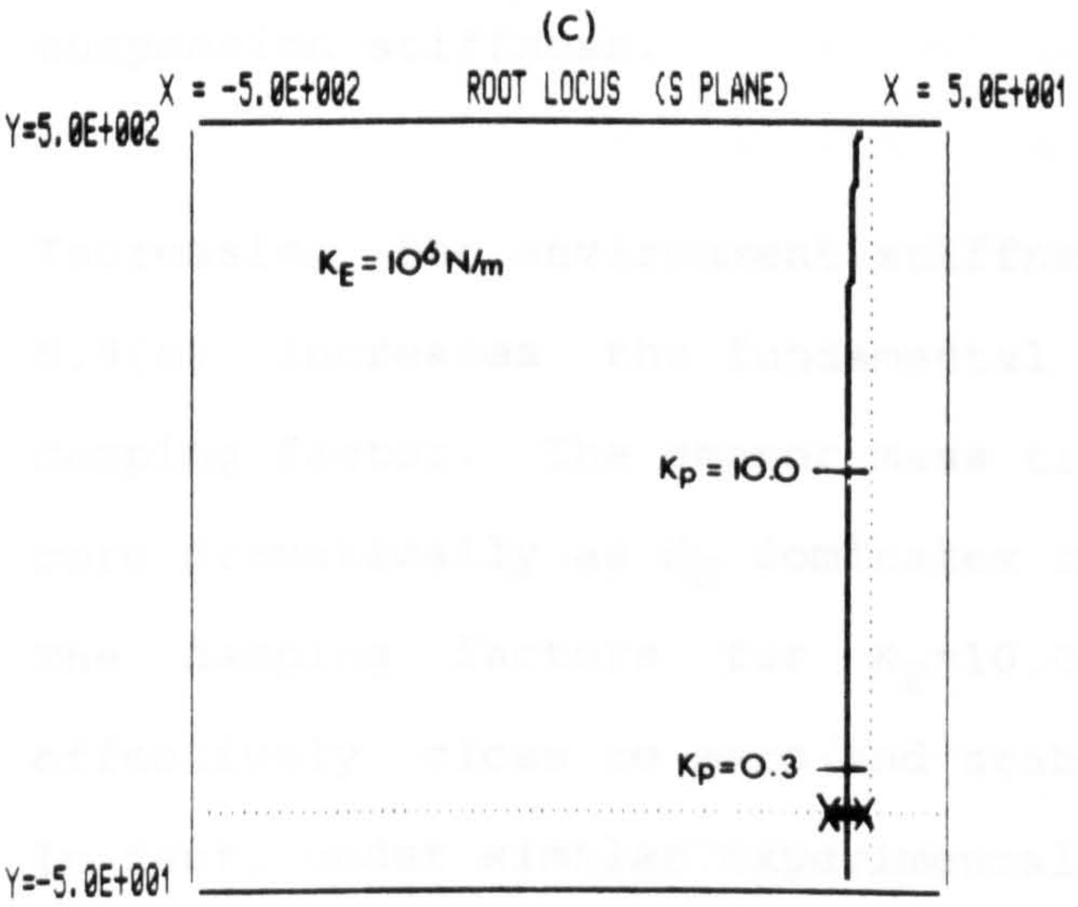
Figure 8.9(a) shows the root loci for an environment stiffness of 10^3N/m, approximately equal to that of the free end of the cantilever used in the experimentation.

The root locus diagram again displays two distinct loci, a low frequency locus associated with the fundamental force response and a high frequency locus associated with sensor mass vibration. On the low frequency fundamental locus, the point corresponding to K_p=10.0 is marked and shows the transient to be oscillatory, underdamped and with a damping factor of around 0.7. This approximately corresponds to figure 7.7(d) of the experimental chapter, which shows a low frequency transient with slight overshoot. The high frequency sensor/environment oscillation is determined by the combined stiffness of its supports, K_S

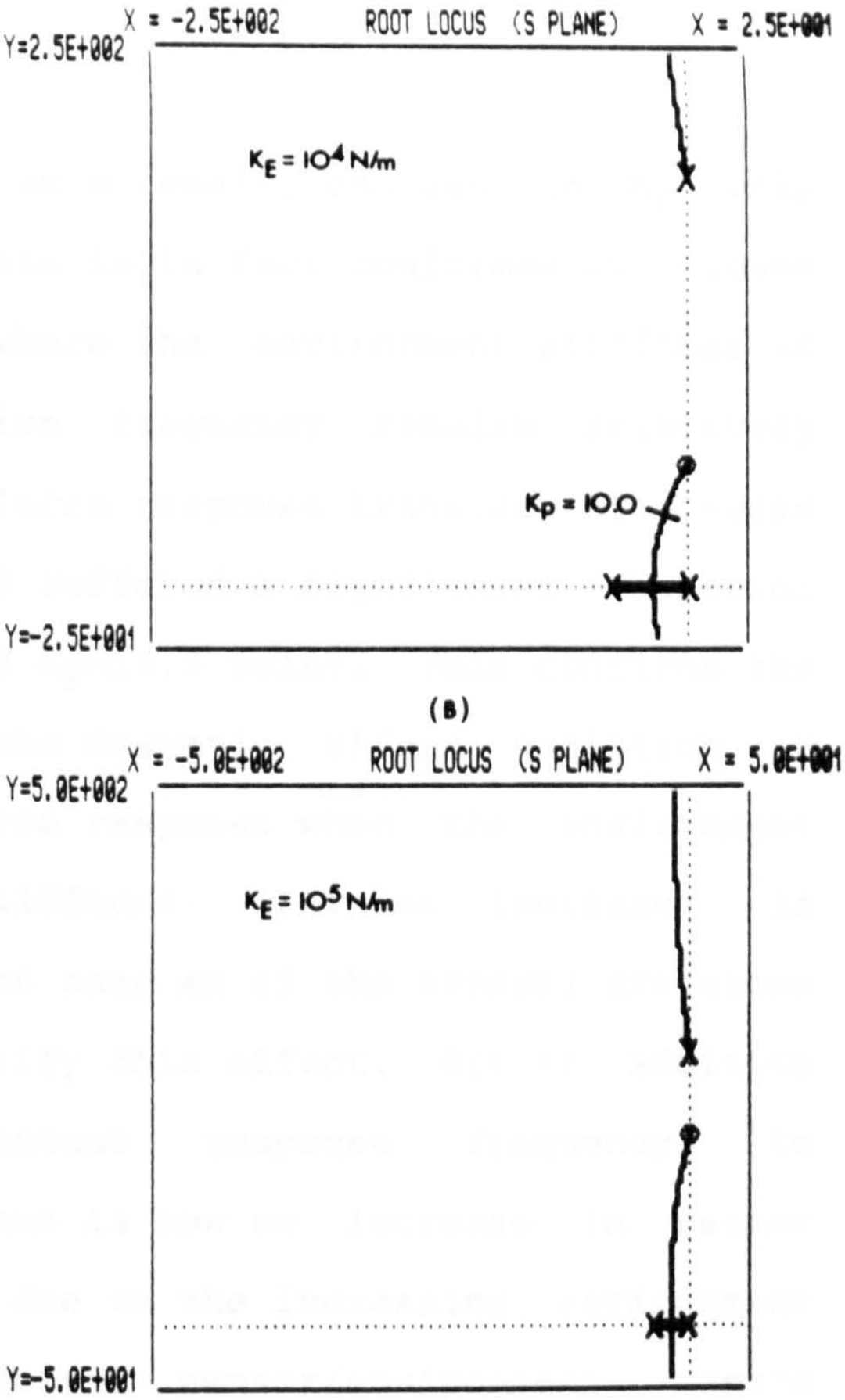


(A)

(C)

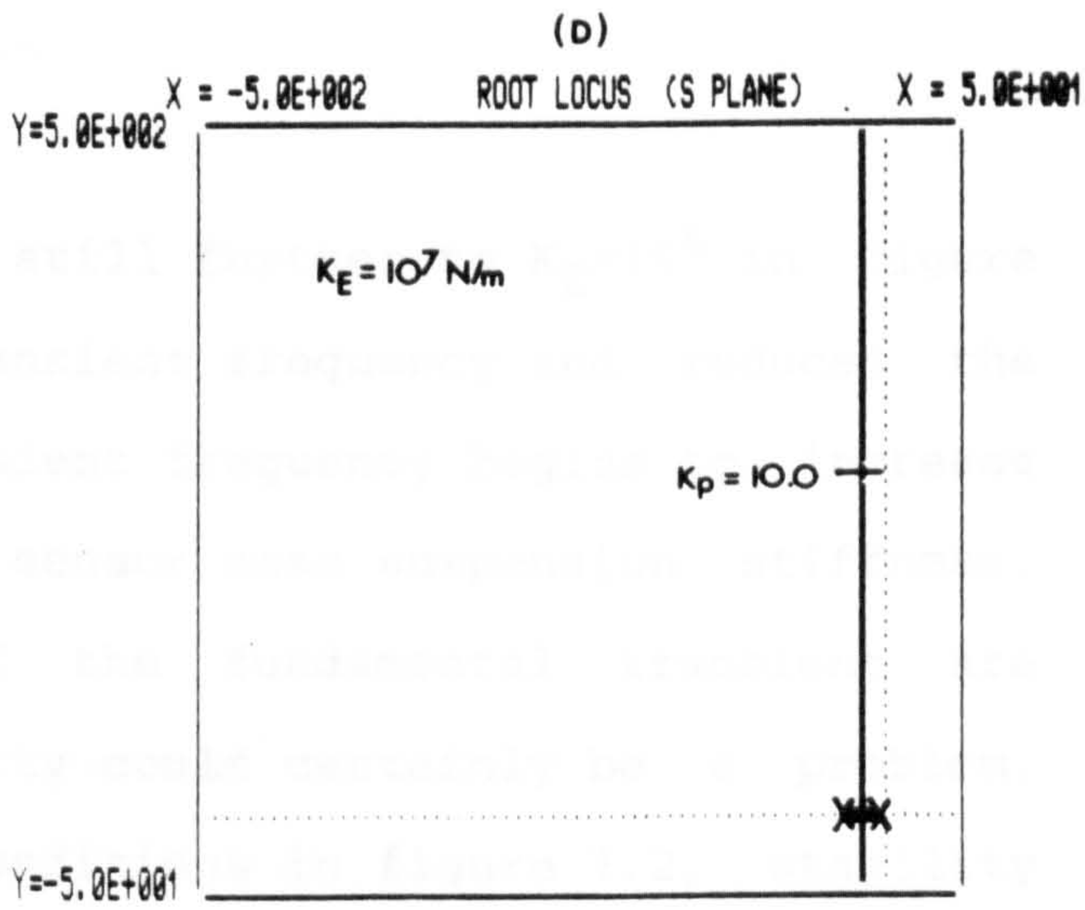


(E)



(B)

(D)



(F)

Figures 8.9(a) to (f). Simulations showing the root locus plots of the viscously damped, single degree of freedom, force controlled system. Environment stiffness K_E is the variable.
$K_S=10^5$N/m, $C_S=0$, $C_R=1.8\times10^4$N/ms^{-1}, $K_D=0$. Different operating points are shown at $K_P=0.3$, 10.0 units. All other parameters take values defined in table 8.1. Figures: (a) $K_E=10^3$N/m, (b) $K_E=10^4$N/m, (c)&(d) $K_E=10^5$N/m, (e) $K_E=10^6$N/m, (f) $K_E=10^7$N/m.

and K_E. K_S is greater than K_E and, as a result, changes in K_E only slightly affect this oscillation. This is in fact confirmed by figure 8.9(b), the repeat of figure 8.9(a), where the environment stiffness is at 10^4N/m and the sensor oscillation frequency remains relatively unchanged. However, the fundamental force response transient of figure 8.9(b) has increased in frequency and suffered a significant reduction in damping factor, shown by the marked K_p=10.0 point. This confirms the experimental observations showing the dramatic effect variation in environment stiffness can have on force response when the environment stiffness is less than sensor stiffness. Further increases in environment stiffness to $K_E=10^5$N/m (the same as of the sensor) are shown in figure 8.9(c) and (d) to again verify this effect. But in addition to the rise in fundamental transient response frequency, to approximately 25 H_z for K_p=10.0, there is now an increase in sensor oscillation frequency. The latter is due to the increasing environment stiffness contribution to the combined sensor/environment mass's suspension stiffness.

Increasing the environment stiffness still further to $K_E=10^6$ in figure 8.9(e) increases the fundamental transient frequency and reduces the damping factor. The sensor mass transient frequency begins to increase more dramatically as K_E dominates the sensor mass suspension stiffness. The damping factors for K_p=10.0 of the fundamental transient are effectively close to zero and stability could certainly be a problem. In fact, under similar experimental conditions in figure 7.2, stability was only possible if K_p was reduced to around 0.3.

Under these conditions, where the sensor stiffness is significantly less than environment stiffness and while neglecting viscous damping, equation 8.8 can be written as:

$$(M_1S^2 + C_RS + (K_pK_CN+1))(M_2S^2 + K_E) = 0$$

This shows the characteristic to be composed of two distinct equations, as indeed equation 5.15 in the general analysis chapter did. The fundamental force response, the equation associated with M_1, is seen now to be totally independent of K_E. Whereas the oscillation associated with M_2 is totally dependent on K_E. Increasing K_E to 10^7N/m, figure 8.9(f), agrees with this prediction, as is shown by the fundamental transient frequency remaining unchanged between figures 8.9(e) and (f). Comparing these simulation results with the experimental results of figure 7.2, where for K_p=0.3, the simulations give a transient frequency of 6.9Hz and experimentally it is found to be 7 to 9 Hz.

This latter result confirms that when environment stiffness is greater than sensor stiffness, then the fundamental response is independent of environment stiffness and is influenced only by sensor stiffness and proportional gain. The converse is also true, as shown in figures 8.9(a), (b) and (c), if the environment stiffness is considerably less than the sensor stiffness then the environment stiffness has been shown to be a highly influential parameter in determining the fundamental response frequency.

8.5 Force Rate Damping

Software implementation of force rate damping has been unsuccessfully attempted experimentally and has been discussed in section 5.8.3. The errors associated with its derivation proved troublesome and therefore the gains were necessarily low and the force rate effects inconclusive.

Work is continuing on the development of a force rate sensor and ultimately it is hoped it will show benefit. The following simulations are designed to illustrate some of the possible advantages of using

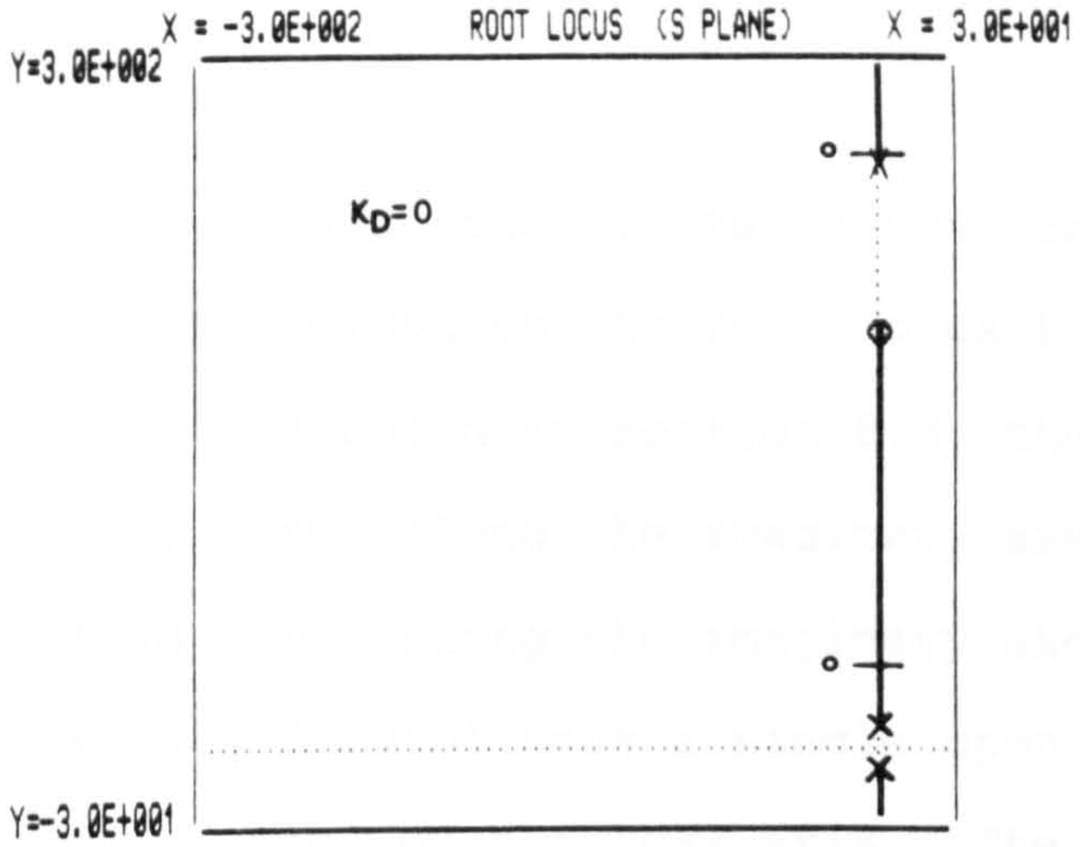
substantial force rate signals to improve force stability and response.

Adding a force rate damping term K_D to the force controller K_F of figure 8.3 gives the system described in figure 8.6, and has the effect of adding an open-loop zero to the system. This extra zero reduces the number of asymptotes from two to one. The remaining asymptote is then directed along the negative real axis. Having only a single asymptote tends to improve stability as the loci which were previously near to the imaginary axis are moved towards the single asymptote.

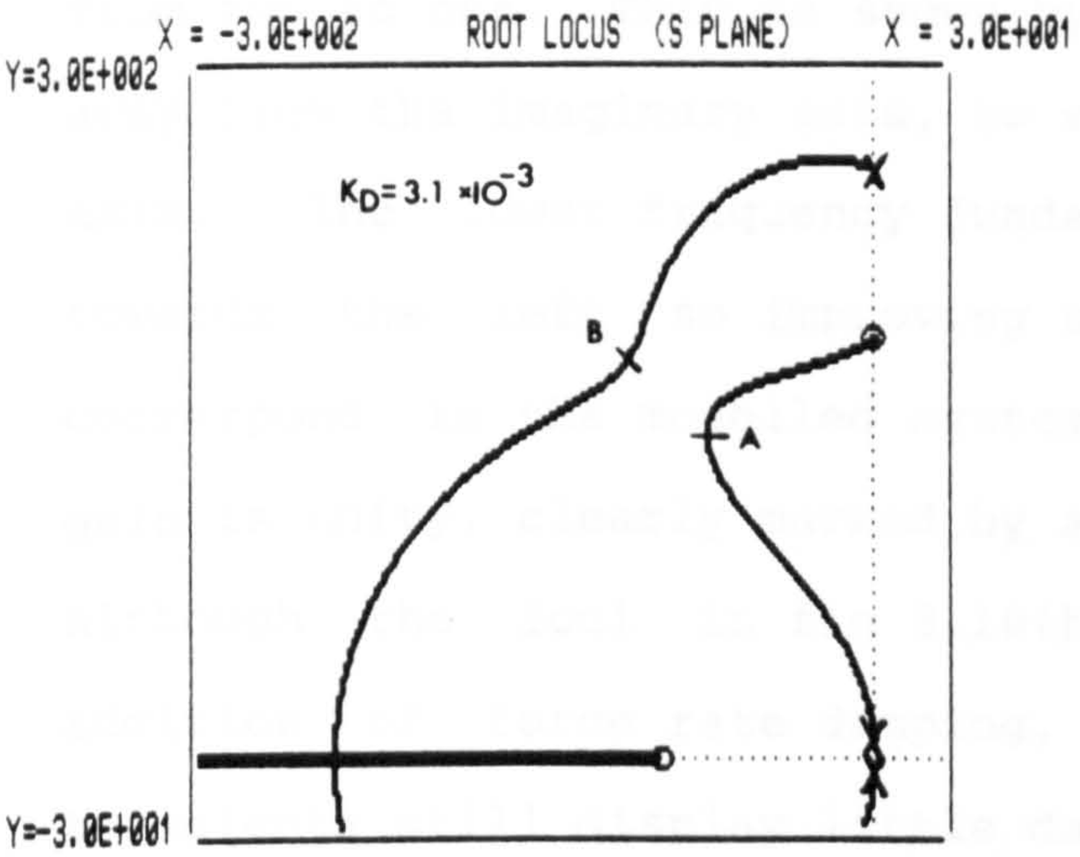
Initially the effect on force response by force rate damping alone is studied, without the beneficial effects of velocity dependent damping interfering with the results. The result obtained is later used as a comparison to systems which include the effects of velocity dependent damping. The effectiveness of force rate damping under differing operating conditions is also considered. This involves studying the effect of variation in environmental and sensor stiffnesses on the properties of force rate. Finally the section is concluded by a discussion on the implications of force rate signals to the many aspects of force control.

8.5.1 Pure Force Rate Damping

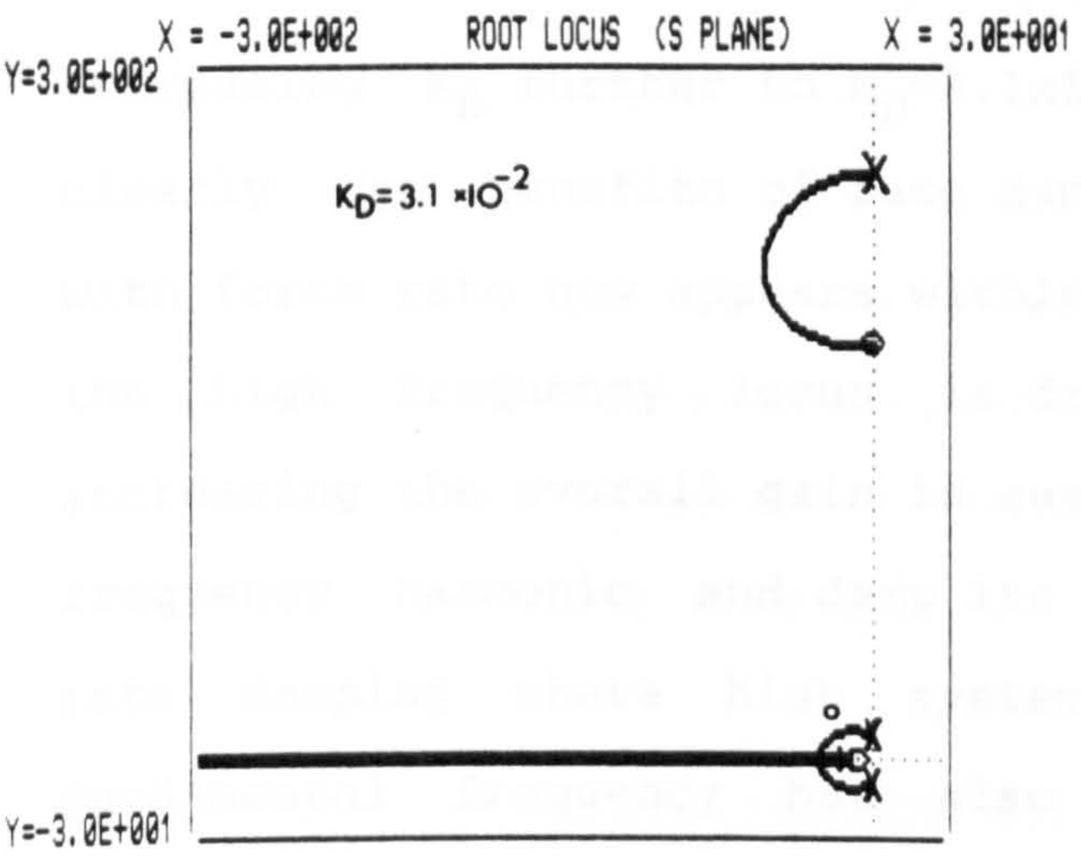
The effect of force rate damping variation is considered firstly in isolation. That is, the velocity dependent damping C_R is set to zero and the environment and sensor stiffnesses are fixed at typically intermediate values, $K_S=K_E=10^5$N/m. The system model was again the same as that described earlier in section 8.2. Proportional gain was set at $K_P=0.3$ and remained constant throughout. The root locus diagrams were, as usual, generated by increasing the overall open-loop gain of the



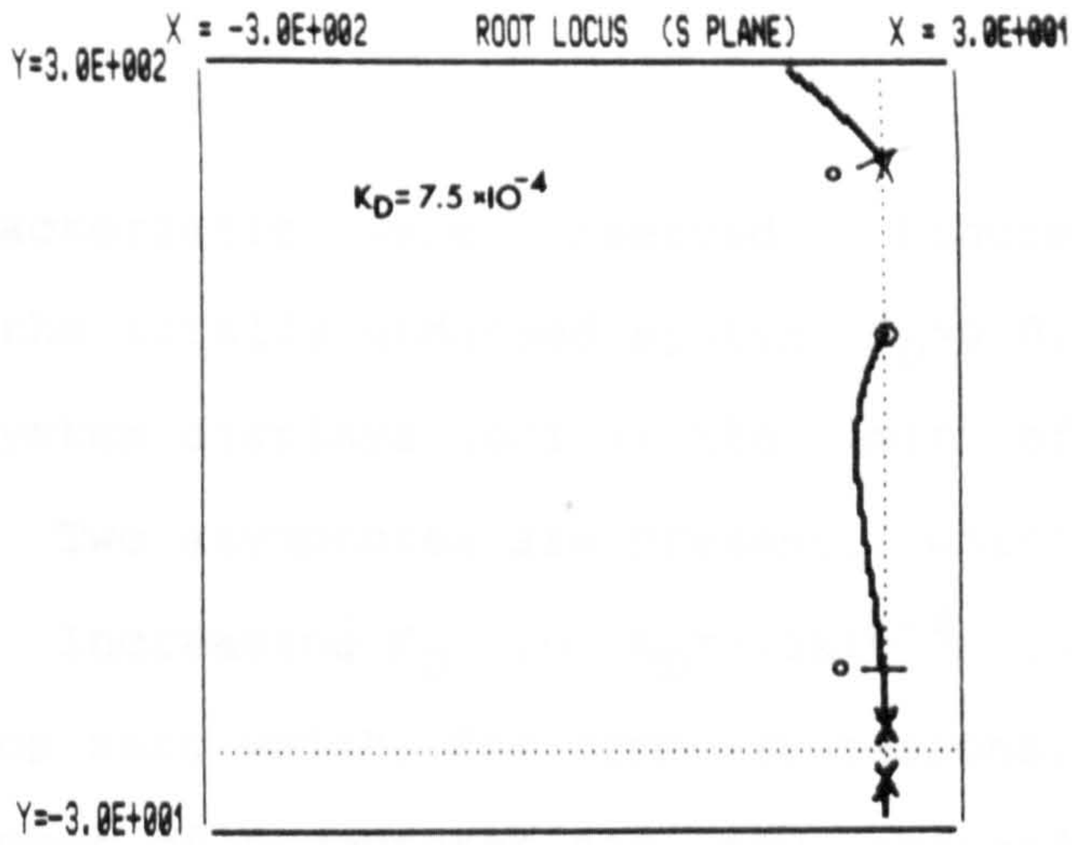
(A)



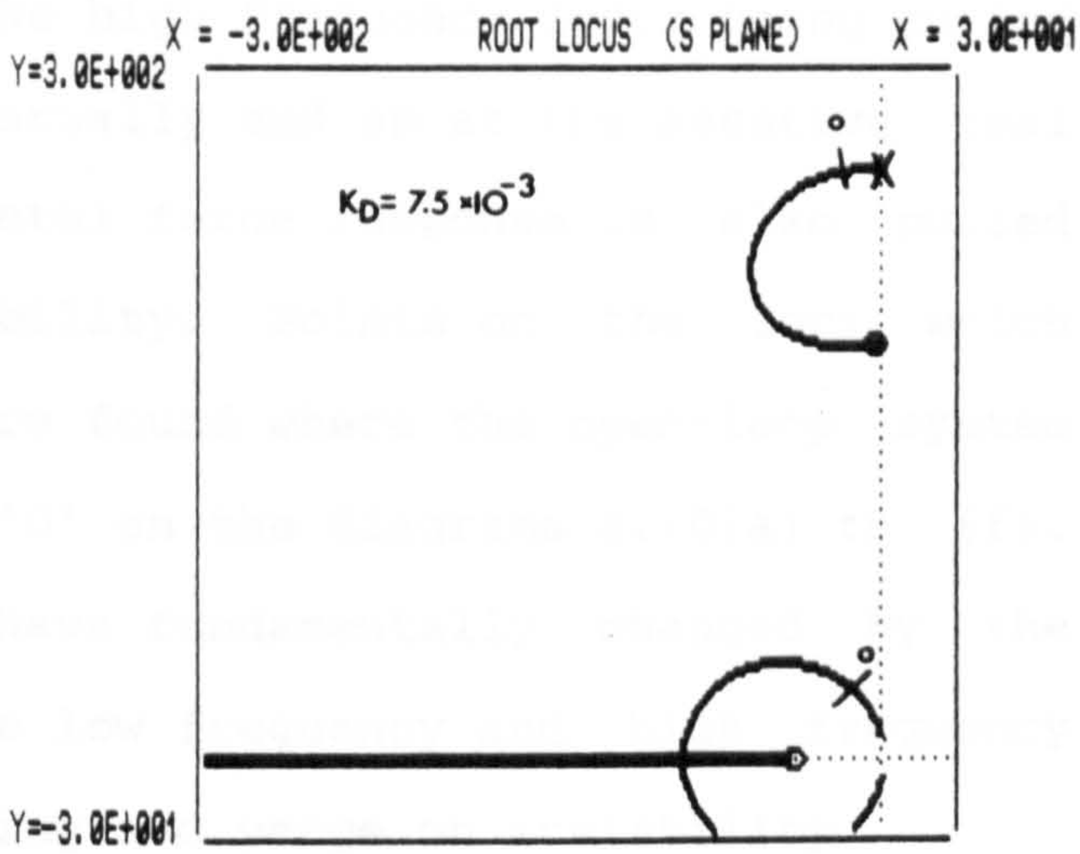
(C)



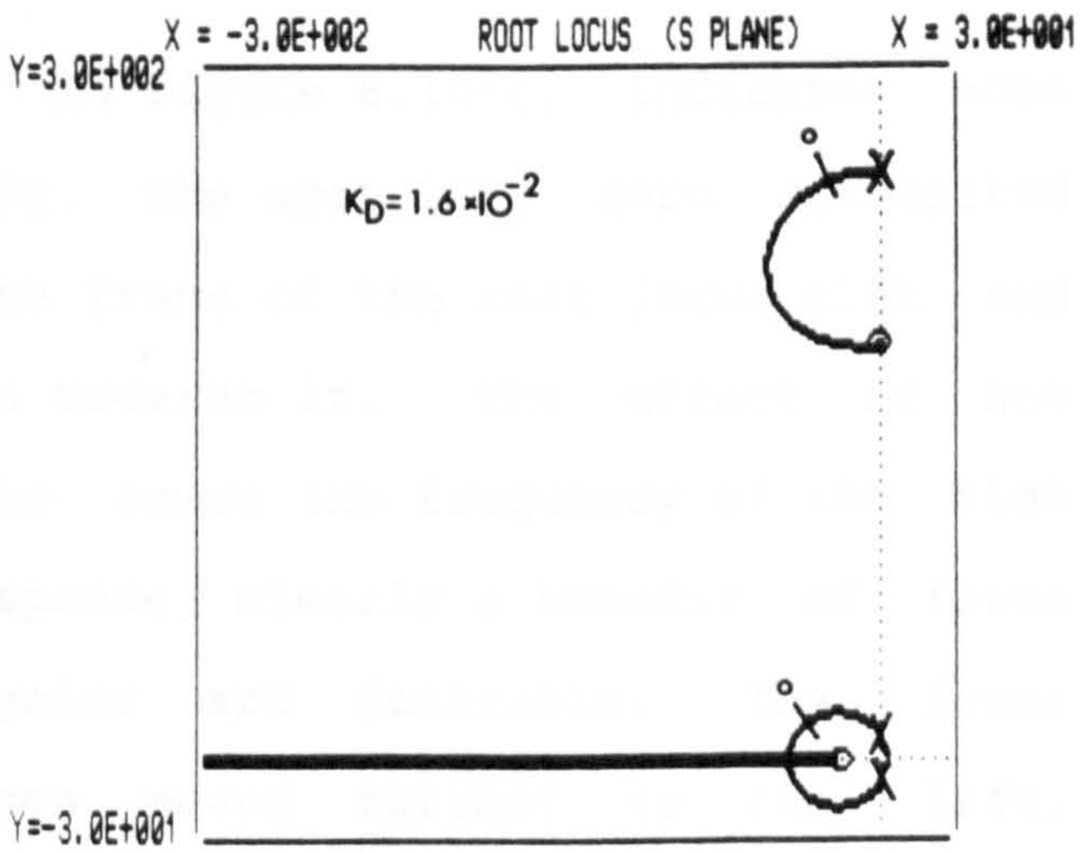
(E)



(B)



(D)



(F)

Figures 8.10(a) to (f). Simulations showing the root locus plots of the force rate damped, single degree of freedom, force controlled system. Force rate damping K_D is the variable.

$K_S=K_E=10^5$N/m, $C_S=0$, $C_R=0$, $K_P=0.3$ units. The operating points are shown by O. All other parameters take values defined in table 8.1. Figures: (a) $K_D=0$, (b) $K_D=7.5\text{x}10^{-4}$, (c) $K_D=3.1\text{x}10^{-3}$, (d) $K_D=7.5\text{x}10^{-3}$, (e) $K_D=3.1\text{x}10^{-2}$, (f) $K_D=1.6\text{x}10^{-2}$ units.

system and the roots of the characteristic were observed. Figure 8.10(a) displays the root locus for the totally undamped system, K_D=0.0, and, as shown in section 8.3, the system displays loci at the limit of stability, along the imaginary axes. Two asymptotes are present, which also run along the imaginary axes. Increasing K_D to K_D=7.5x10^{-4} in figure 8.10(b) adds a single open loop zero which, for symmetry reasons, must lie on the real axis. The number of asymptotes has now reduced from two to one. This is shown by the high frequency locus being curled away from the imaginary axis, to eventually end up at the negative real axis. The lower frequency fundamental force response is also pulled towards the left so improving stability. Points on the loci which correspond to the modelled system are found where the open-loop system gain is unity, clearly marked by an 'O' on the diagrams 8.10(a) to (f). Although the loci in fig 8.10(b) have fundamentally changed by the addition of force rate damping, the low frequency and high frequency transients still display little damping and verge on instability.

Increasing K_D further to K_D=3.1x10^{-3} in figure 8.10(c) indicates more clearly the benefits of rate damping. The open loop zero associated with force rate now appears within the frame of the root locus plot and the high frequency locus is drawn towards it. The effect of now increasing the overall gain is seen to lower the frequency of the high frequency harmonic and damp its response, clearly a benefit of force rate damping where high system gains are desirable. The lower fundamental frequency has also been moved further to the left, increasing damping of the transient. But in the latter case increasing the overall gain increases the transient frequency of the fundamental and marginally improves damping to the point marked 'A'. Thereafter the damping reduces as the locus moves back towards the imaginary axis and instability. A feature of figure 8.10(c) is that two kinks appear on both loci at points 'A' and 'B'. Increasing K_D further to K_D=7.5x10^{-3}

causes a joining of the loci at these points and a formation of two new loci with completely different characteristics, as shown in figure 8.10(d). This transition is significant in that the fundamental transient will now decay for all values of system gains. But the same cannot now be said of the higher frequency harmonic whose zero and pole both lie on the imaginary axis. Thus the system could boarder on instability for near infinite system gains.

Increasing K_D further to $K_D=1.6x10^{-2}$ and $3.1x10^{-2}$ in figure 8.10(e) and (f) respectively, shows the upper harmonic to be relatively unaffected by further increases in K_D, although the marked operating points do show increased damping. However, the low frequency fundamental transient reduces in frequency and increases in damping. Finally in figure 8.10(f) the fundamental is shown to be completely overdamped.

The benefits of force rate damping over velocity dependent damping can be clearly seen. Force rate damping provides a robust controller able to cope with high system gains often found where environment and sensor stiffnesses are high and controller gains are necessarily large to minimise the effects of joint stiction. Velocity dependent damping provides none of these advantages. Providing sufficient rate damping is employed to take K_D above the locus transition, then stability should be easily achieved with rate damping alone.

8.5.2 Variation in Environment Stiffness

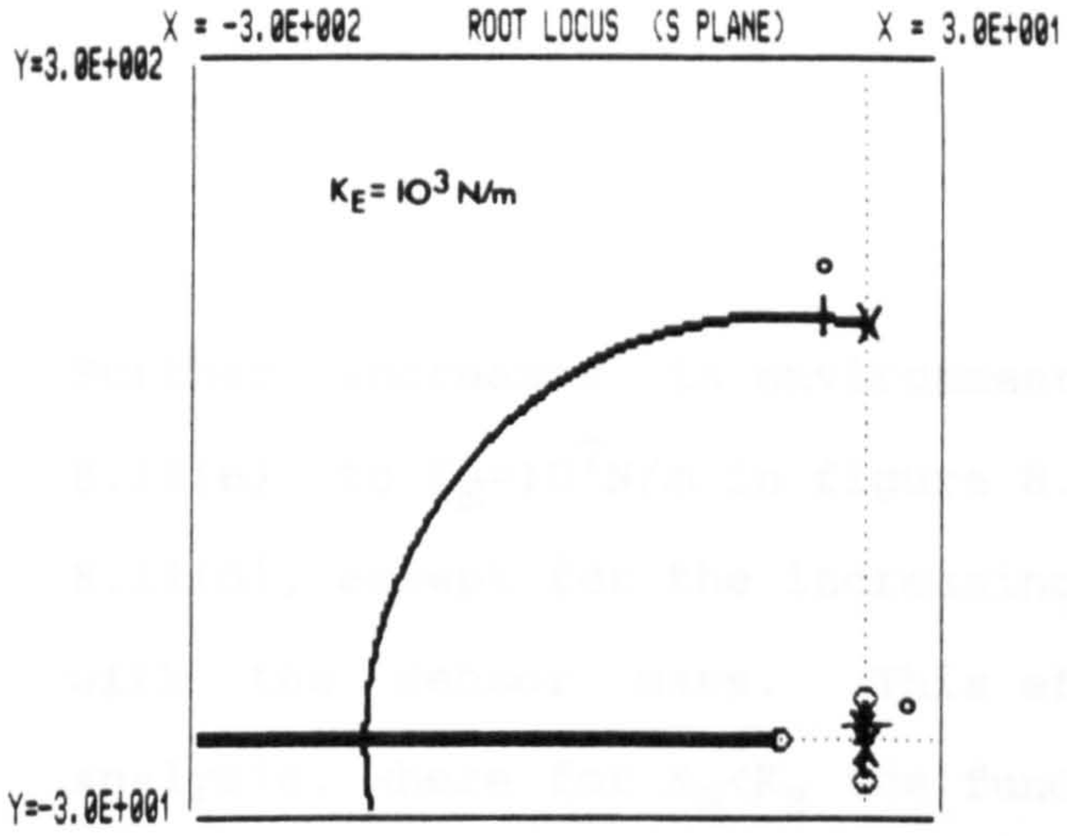
The properties of force rate damping are affected by variation in environment stiffness, it is therefore important to study this aspect.

The same force rate controlled system as that used in the previous section 8.5.1, was again used here, where: $C_R=0.0$, $K_S=10^5$N/m, $K_P=0.3$ and

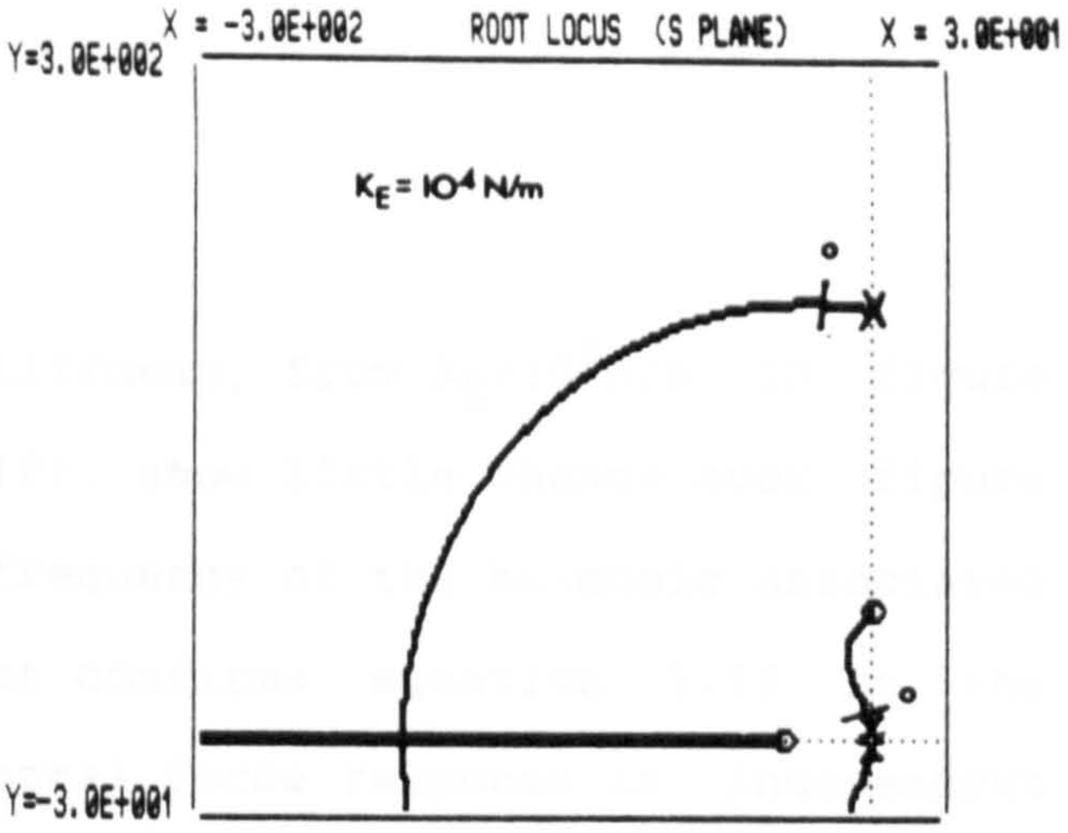
the rate damping term remained constant throughout at $K_D=7.5x10^{-3}$. In this case environment stiffness became the variable.

Environment stiffness was varied between two realistic extremes. A soft environment, as provided by the free end of the experimental cantilever beam, gave the lower limit at $K_E=10^3$N/m. The upper limit was taken as $K_E=10^7$N/m, approximately that of the supported turbine blade. Conveniently mid-range is at $K_E=10^5$N/m, the same stiffness as the sensor. Figure 8.11(a) gives the root locus analysis for $K_E=10^3$N/m, and shows the low frequency locus to be at the point of instability. The exact operating point on the locus is indicated by the usual line marking. The figure appears completely different in characteristic to figure 8.10(d) where the same value of K_D is used. This indicates a significant dependence of the force rate damping effectiveness on the environment stiffness. This fact is further confirmed in figure 8.11(b) where the environment stiffness has been increased to $K_E=10^4$N/m. The low frequency fundamental locus has now moved to the left, affording a degree of stability together with a slight increase in transient frequency. The root locus of figure 8.11(b) shows a similar pattern to that of figure 8.10(b) which, for low values of damping, shows the high frequency locus directed towards the zero on the negative real axis.

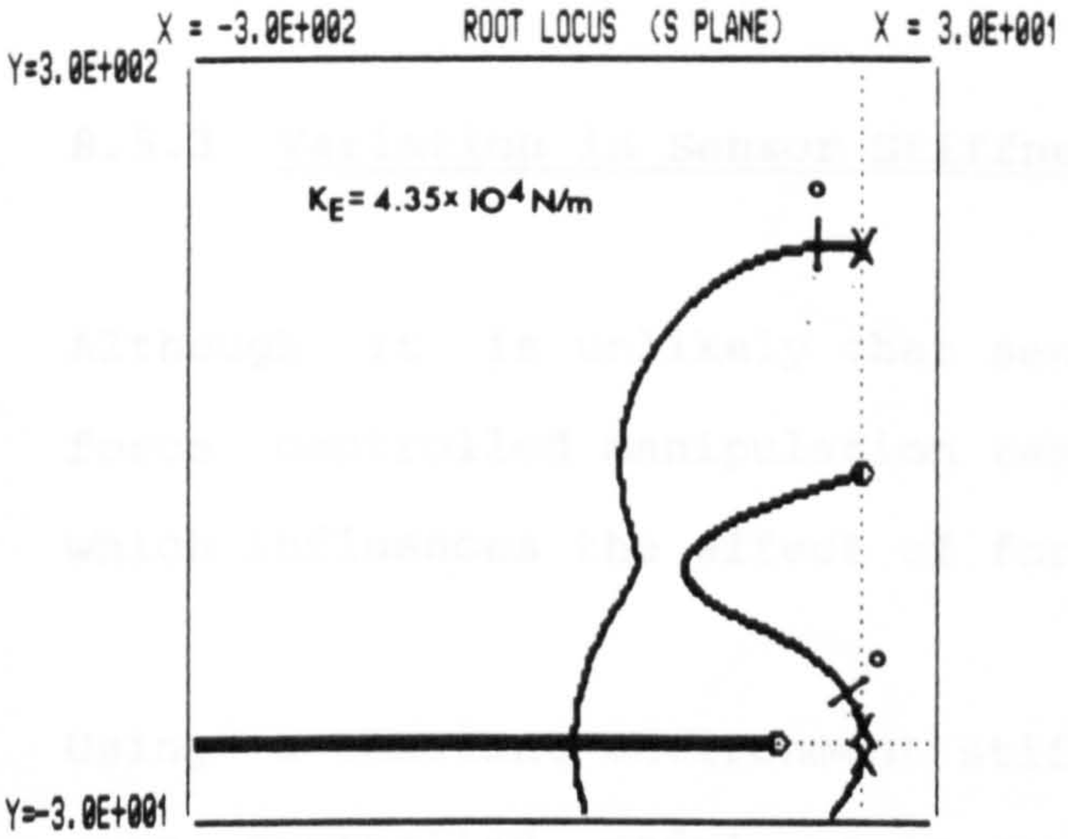
Increasing K_E further in figure 8.11(c), shows a similar transition to figure 8.10(c), where the whole nature of the locus changes. The value of K_E at the point of change is found to be near to $K_E=4.35x10^4$N/m. Beyond this point further increases in K_E show the two loci to completely divide into the low frequency fundamental, and a higher frequency sensor oscillation. The low frequency fundamental is now found to be stable for all system gain and is increasingly damped for increasing gain. This is clearly confirmed in figure 8.11(d) where $K_E=10^5$N/m.



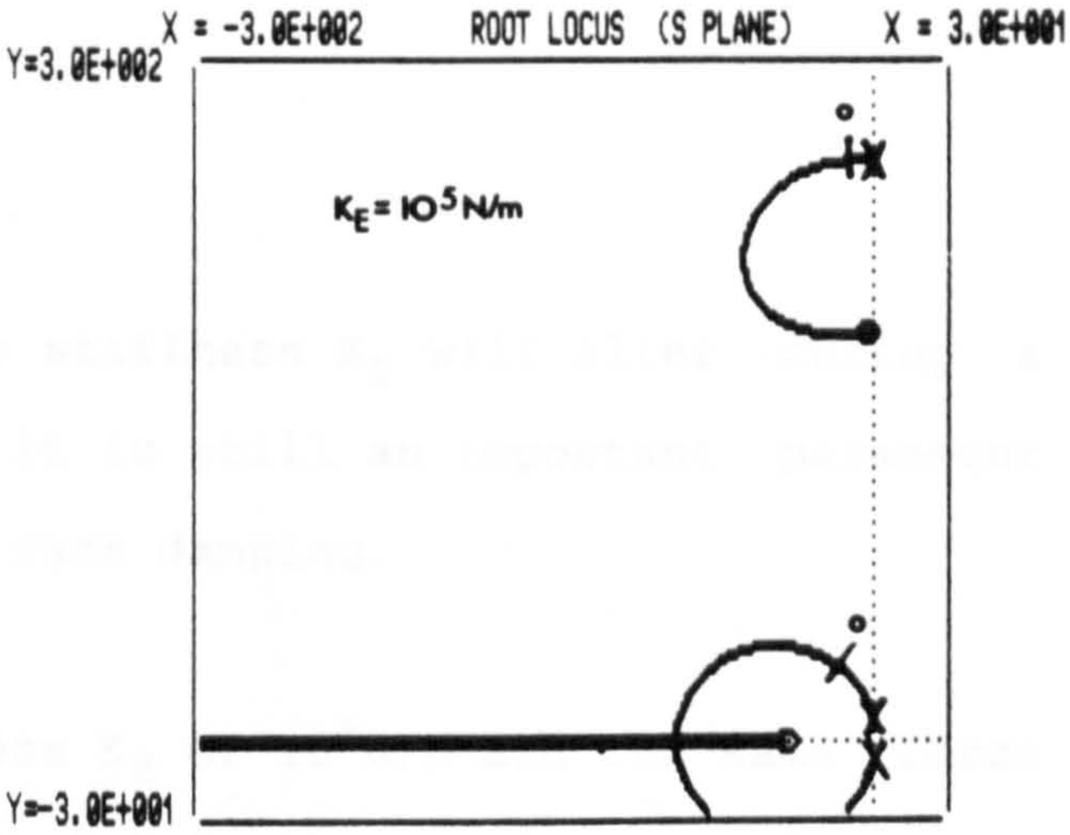
(A)



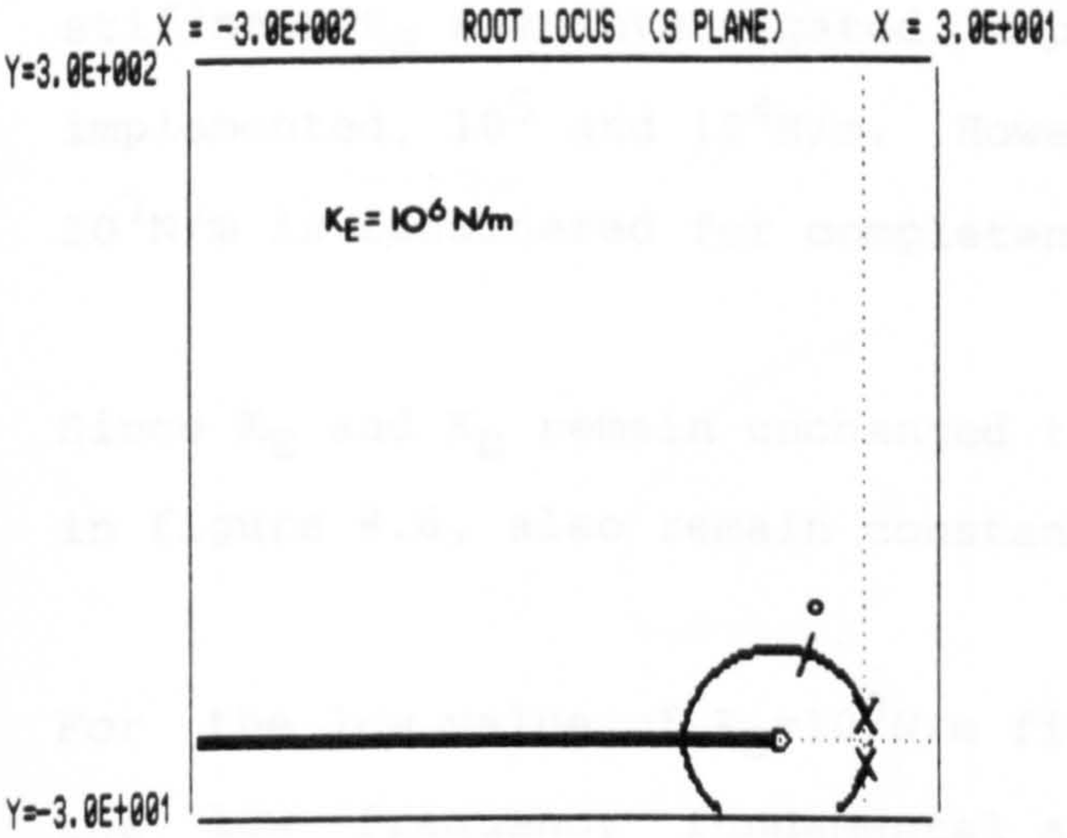
(B)



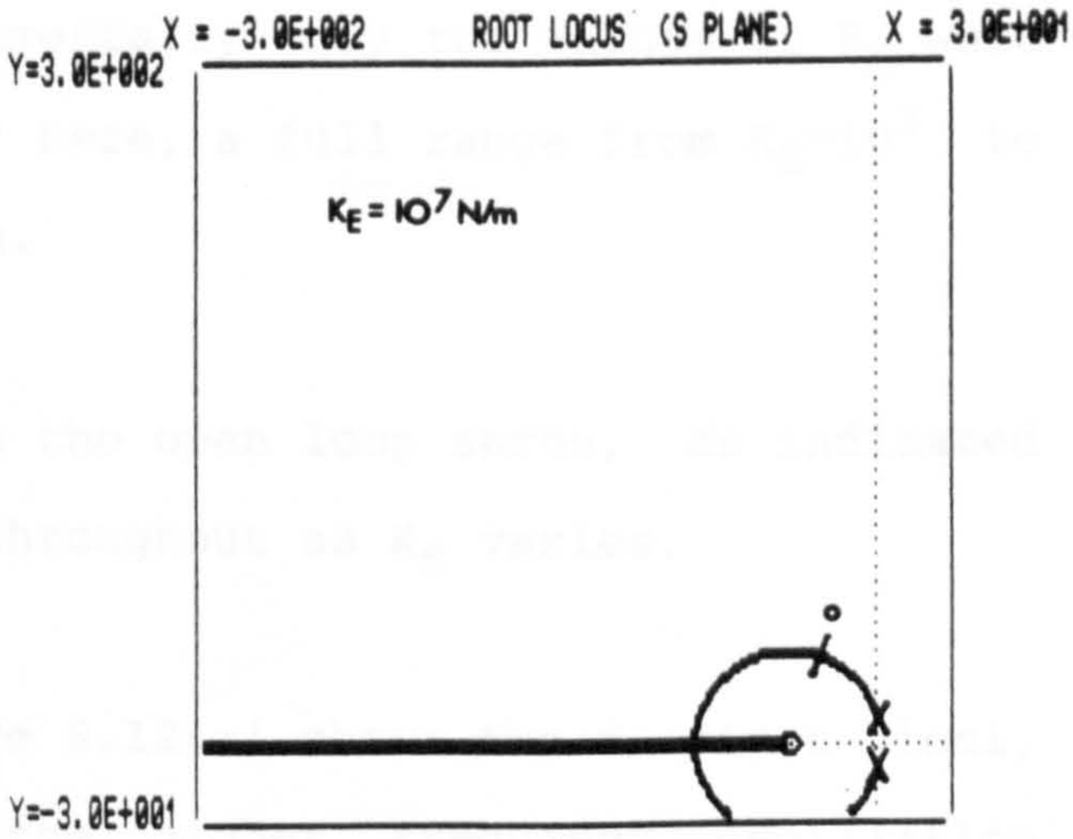
(C)



(D)



(E)



(F)

Figures 8.11(a) to (f). Simulations showing the root locus plots of the force rate damped, single degree of freedom, force controlled system. Environment stiffness K_E is the variable.
$K_D=7.5\times10^{-3}$units, $K_S=10^5$N/m, $C_S=0$, $C_R=0$, $K_P=0.3$ units. The operating points are shown by O. All other parameters take values defined in table 8.1. Figures: (a) $K_E=10^3$N/m, (b) $K_E=10^4$N/m, (c) $K_E=4.35\times10^4$N/m, (d) $K_E=10^{5}$N/m, (e) $K_E=10^6$N/m, (f) $K_E=10^7$N/m.

Further increases in environment stiffness, from $K_E=10^6$N/m in figure 8.11(e) to $K_E=10^7$N/m in figure 8.11(f), show little change over figure 8.11(d), except for the increasing frequency of the harmonic associated with the sensor mass. This effect confirms equation 5.16 in the analysis, where for $K_S<K_E$ the fundamental force response is independent of K_E.

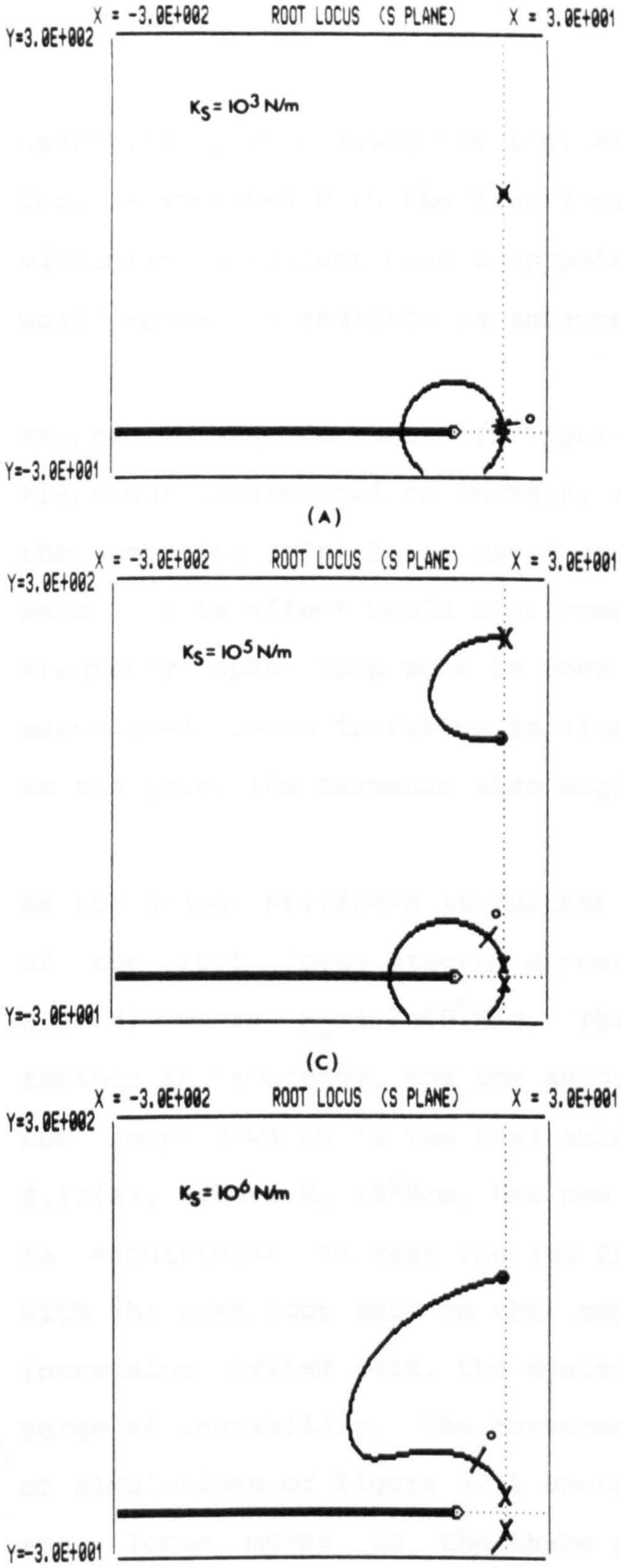
8.5.3 Variation in Sensor Stiffness

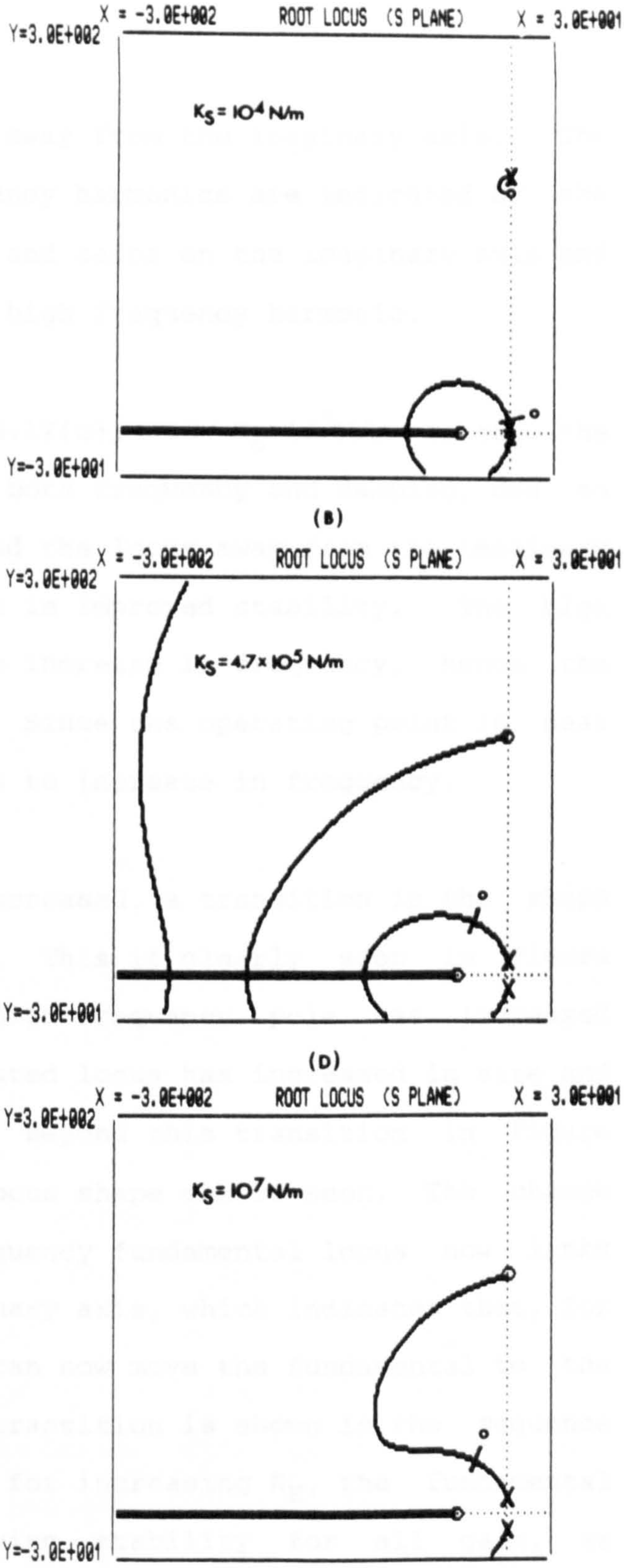
Although it is unlikely that sensor stiffness K_S will alter during a force controlled manipulation task, it is still an important parameter which influences the effect of force rate damping.

Using a constant environment stiffness K_E of 10^5N/m and the same force rate controlled model as in section 8.5.2, the variation of sensor stiffness K_S was investigated. Experimentally only two values of K_S were implemented, 10^5 and 10^4N/m. However here, a full range from $K_S=10^3$ to 10^7N/m is considered for completeness.

Since K_E and K_D remain unchanged then the open loop zeros, as indicated in figure 8.6, also remain constant throughout as K_S varies.

For the low value of $K_S=10^3$N/m figure 8.12(a) shows two distinct loci, the low frequency fundamental and the higher frequency oscillation associated with the sensor mass. Little change can be seen in figure 8.12(b), where $K_S=10^4$N/m, except for a slight change in the operating point on the locus, due to the increase in system gain caused by K_S increasing. In both cases (a) and (b) the system operating points are close to the imaginary axis and verge on instability. This situation could be improved by increasing system gain, causing a movement of the





Figures 8.12(a) to (f). Simulations showing the root locus plots of the force rate damped, single degree of freedom, force controlled system. Sensor stiffness K_S is the variable. $K_D=7.5\text{x}10^{-3}$units, $K_E=10^5$N/m, $C_S=0$, $C_R=0$, $K_P=0.3$ units. The operating points are shown by O. All other parameters take values defined in table 8.1. Figures: (a) $K_S=10^3$N/m, (b) $K_S=10^4$N/m, (c) $K_S=10^5$N/m, (d) $K_S=4.7\text{x}10^5$N/m, (e) $K_S=10^6$N/m, (f) $K_S=10^7$N/m.

operating points along the loci and away from the imaginary axis. The loci associated with the high frequency harmonics are indicated by the virtually coincident open loop poles and zeros on the imaginary axis and would appear to indicate an undamped high frequency harmonic.

Increasing K_S further in figure 8.12(c), to $K_S=10^5$N/m causes the transient fundamental to increase in both frequency and damping, due to the operating point being moved around the locus away from the imaginary axis. This effect would also result in improved stability. The high frequency open loop pole is seen to increase in frequency, hence the associated locus increases in size. Since the operating point is near to the pole, the harmonic also begins to increase in frequency.

As the sensor stiffness is further increased, a transition in the shape of the root locus diagram appears. This is clearly seen in figure 8.12(d) where $K_S=4.7\text{x}10^5$N/m. The high frequency pole has increased further in frequency, and the associated locus has increased in size and now loops down on to the real axis. Beyond this transition in figure 8.12(e), where $K_S=10^6$N/m, the new locus shape can be seen. The change is significant in that the low frequency fundamental locus now links with the open loop zero on the imaginary axis, which indicates that, for increasing system gain, the system can now move the fundamental to the verge of instability. The converse transition is shown in the sequence of simulations of figure 8.11 where, for increasing K_E, the fundamental root locus moves to the shape giving stability for all gain, as discussed in section 8.5.2.

Further increases in sensor stiffness to $K_S=10^7$N/m in figure 8.12(f), have little effect on the shape of the fundamental root locus, showing that the response is now largely determined by the lower stiffness of the environment, rather than that of the sensor.

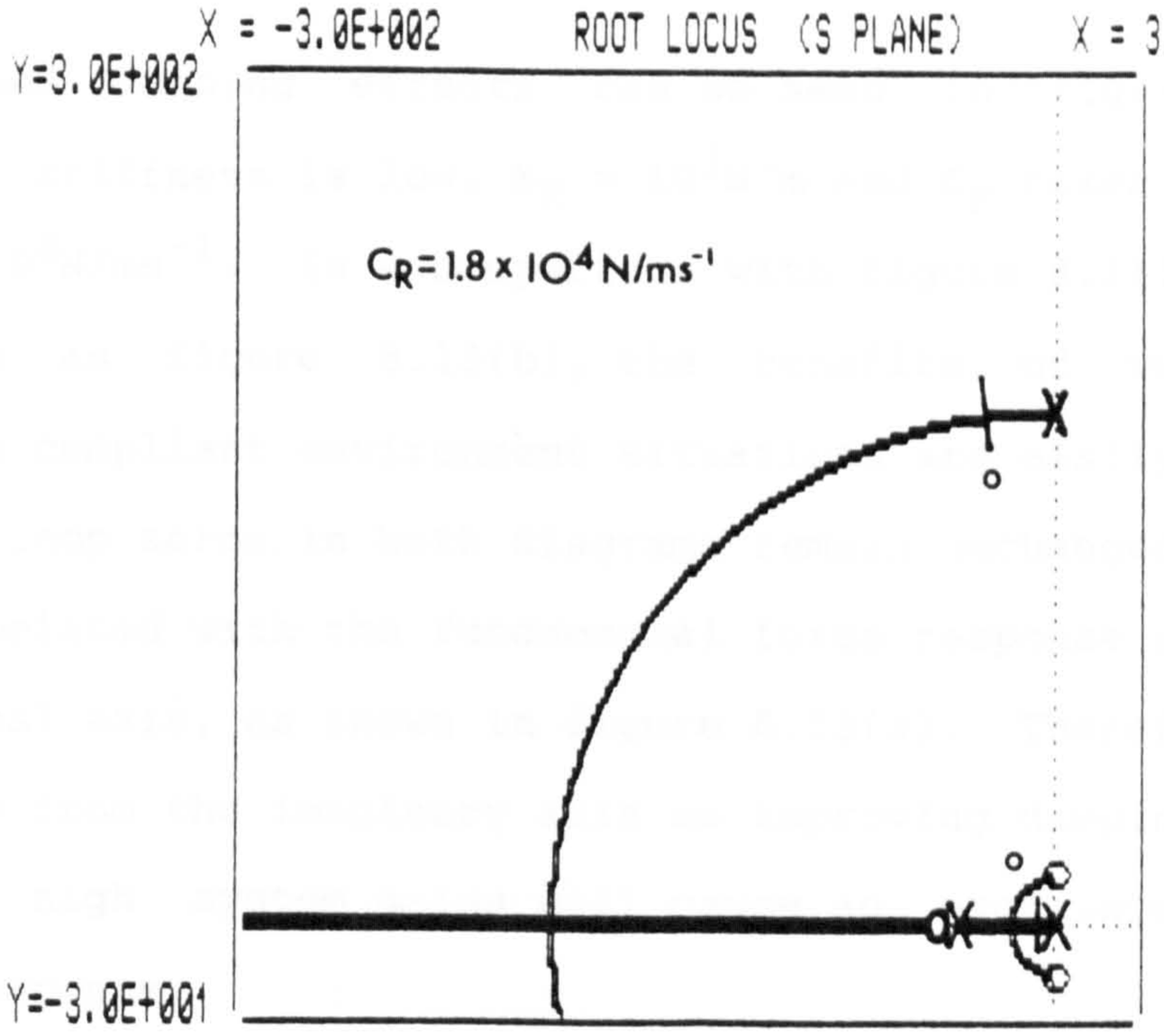
8.5.4 Combined Velocity Dependent and Force Rate Dependent Damping

Equation 8.6 represents the characteristic equation of the complete force controlled system. The terms necessary for stability, in odd powers of S, are composed of C_R, C_S, and K_D parameters. Sensor damping C_S, is considered later in section 8.6 and is assumed here to be zero. Only variation in C_R and K_D are considered. Equation 8.6 can be rewritten as equation 8.9, excluding the effects of sensor damping.

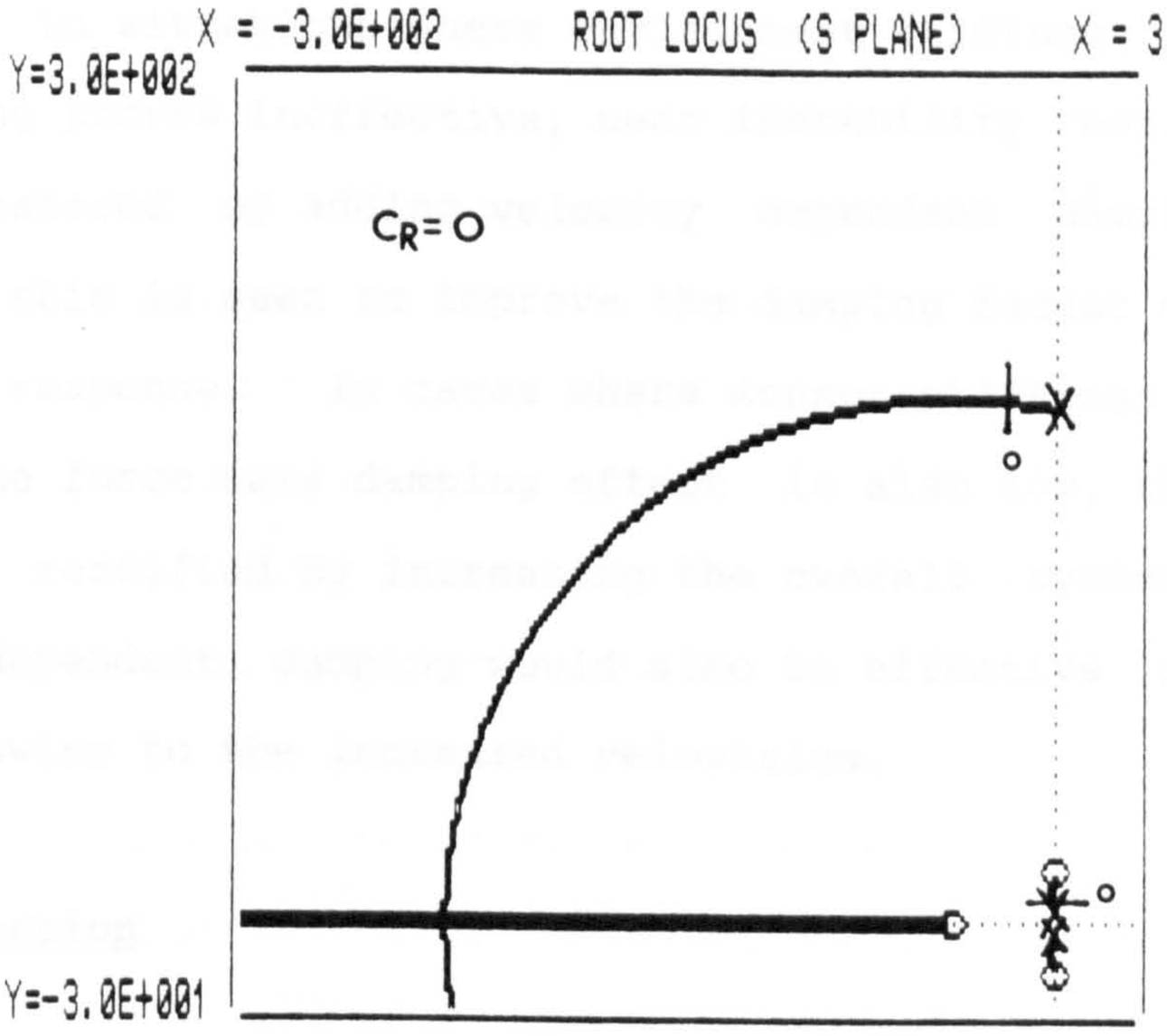
$$M_1M_2S^4 + [M_2C_R+M_2K_SNK_CK_D]S^3 + [M_1(K_E+K_S) + M_2(NK_CK_P+1)K_S]S^2 + [C_R(K_E+K_S) + K_EK_SNK_CK_D]S + K_EK_S(1+NK_CK_P) \quad (8.9)$$

The odd terms of S depend on terms of K_D and C_R alone.

For low values of K_D, K_E, K_S or indeed system gain, the force rate damping effect as shown by figures 8.11(a) and 8.12(a) is small. This can be inferred from equation 8.9 where the force rate dependent terms in odd powers of S are also dependent on K_S or K_E. As the parameters K_S and K_C increase, so the system damping increases and stability improves as indicated by figures 8.11(f) and 8.12(f). However, for velocity dependent damping the converse is true. In cases where K_p, K_S and K_E are low the K_D dependent terms of equation 8.9 are small, the velocity dependent terms of C_R should then dominate. This has been confirmed both in simulations in section 8.4 and experimentally in section 7.5 where, for low values of K_E or K_S or K_p, the system shows significant damping. It would therefore seem beneficial to have both forms of damping present; force rate damping necessary to maintain stability where system gains are high, and velocity dependent damping where stiffnesses are low and velocities are high.



(A)



(B)

Figures 8.13(a) and (b). Simulations showing the root locus plots of the force rate damped, single degree of freedom force controlled system, with and without viscous damping C_R. Viscous damping C_R is the variable. Shows the benefits of added viscous damping in compliant situations. $K_D=7.5x10^{-3}$units, $K_S=10^5$N/m, $K_E=10^3$N/m, $C_S=0$, $K_P=0.3$ units. The operating points are shown by O. All other parameters take values defined in table 8.1. Figures: (a) $C_R=1.8x10^4$N/ms^{-1}, (b) $C_R=0$.

The combined damping effects can be seen in figure 8.13(a) where environment stiffness is low, $K_E = 10^3 N/m$ and C_R takes the usual value, $C_R = 1.8x10^4 N/ms^{-1}$. In a comparison with figure 8.11(a), repeated for convenience as figure 8.13(b), the benefits of velocity dependent damping in compliant environment situations are easily seen. Although the open loop zeros in both diagrams remain unchanged, the open loop poles associated with the fundamental force response are moved to the negative real axis, as shown in figure 8.13(a). Therefore this locus is moved away from the imaginary axis so improving damping and stability. Only very high system gains will cause an oscillatory and possibly unstable response.

Therefore, in situations where environment stiffness is low, and force rate damping proves ineffective, near instability results, the situation can be restored by adding velocity dependent damping. Under these conditions this is seen to improve the damping factor of the fundamental transient response. In cases where sensor stiffness is low, figure 8.12(a), the force rate damping effect is also low, this situation can easily be rectified by increasing the overall system gain. However velocity dependent damping would also be effective in improving this situation owing to the increased velocities.

8.5.5 Discussion

Force rate damping shows important benefits where variations in environment and sensor stiffness occur. In all cases increases in environment and sensor stiffnesses showed beneficial growth in the negative real component of the fundamental transient response. Where K_E and/or K_S were low, then the damping component is reduced and the operating point on the locus moves near to the imaginary axis and towards instability. In the case of low sensor stiffness, increasing

the overall controller gain would increase the damping factor and restore stability. This follows since increasing the overall gain in effect increases the individual force controller components of K_P and K_D as seen from equation 8.9. In this situation varying overall system gain would appear an extremely simple but useful means of compensating for variations in sensor stiffness. However, its effectiveness largely depends on the value of K_D used and hence the quality of the force rate signal, as suggested in the experimental work.

Force rate damping at high system gains and stiffnesses appears to have extremely important benefits. But in situations where K_S or K_E are low, at around 10^3N/m, then the benefits are considerably reduced. Under the condition of a compliant environment as shown by figure 8.13, stability can be maintained if velocity dependent damping is also present. Velocity damping naturally occurs in both friction and motor back emf. and hence functions in joint coordinates. This causes the velocity damping to be configuration dependent which has its own associated problems as discussed in sections 5.10 and 7.7. Adding velocity damping to the cartesian tool based frame would therefore be beneficial, particularly in compliant situations.

Adding increasing amounts of force rate damping to a system in general moves the loci to the left, improving stability. But there exists a transition in the shape of the root loci and, for values of K_D below this transition value, instability of the dominant low frequency fundamental is still possible for high system gains. Increasing K_D above the transition point changes the loci shape so that a theoretically infinite rise in system gain should not cause instability of the fundamental. Whether or not this transition value of K_D can be achieved practically remains to be seen. The transition point is also dependent on the values of K_E and K_S. It would appear from figures 8.11

and 8.12 that a favourable transition occurs when $K_S<K_E$ or when K_S is low or K_E is high. Again this displays a further benefit of situations where sensor stiffness is less than environment stiffness. Lowering sensor stiffness also allows the system gain to be easily increased, which improves response.

8.6 Sensor Damping

Adding further viscous damping to the force controlled system, in the form of viscous damping of the sensor mass, would intuitively appear to be beneficial. Certainly current literature on the subject by Seering and Eppinger [65] considers this to be so. However, experimental work conducted here, and the following analysis, indicate the situation is not so simple, and that distinct conditional instability can arise from the presence of sensor damping C_S.

An obvious benefit to be gained from adding sensor damping is that it is an extremely effective way of preventing sensor or tool vibration. This property may be essential if a vibrating or rotating tooling is attached to the robot end effector. This situation is common in polishing, grinding or general force controlled machining tasks.

8.6.1 Conditional Stability

A Routh Hurwitz analysis was used to examine the characteristic equation 8.6. Equation 8.6 contains several parameters representing damping: force rate damping K_D, velocity dependent damping C_R, and viscous sensor damping C_S. To consider the combined effects of all these parameters in a Routh Hurwitz analysis is extremely complex giving results which are difficult, if not impossible, to meaningfully interpret. It is far easier and clearer to consider the fundamental effect of C_S alone on

stability, without the added complications of K_D and C_R. Therefore K_D and C_R have been eliminated from the Routh Hurwitz analysis which contains only C_S as the damping term. Earlier analyses have shown the presence of K_D and C_R to improve stability without condition and the same effect is expected here.

Equation 8.10 is a reduced version of equation 8.6, where the only damping present is that of the sensor. The Routh Hurwitz analysis of equation 8.10 is given in Appendix A.

$$M_1M_2S^4 + [M_1C_S+M_2C_S]S^3 + [M_1(K_E+K_S) + M_2(NK_CK_P+1)K_S]S^2 + K_EC_SS + K_EK_S(NK_CK_P+1) = 0 \qquad (8.10)$$

A single condition of instability arises from the Routh Hurwitz analysis of equation 8.10, given as:

$$M_1K_E = NK_CK_PK_S(M_1+M_2) \qquad (8.11)$$

Surprisingly the condition is independent of C_S, however omitting C_S from equation 8.10 removes the condition of instability. Therefore, even for extremely small values of C_S, theoretical conditional instability could still arise. In practice however this is meaningless, since the system has no damping and the root loci run along the imaginary axis bordering on instability. Under these conditions adding velocity or force rate damping could easily achieve stability.

The result of the Routh Hurwitz analysis in equation 8.11 simplifies if $M_1>>M_2$, as in this case, to a simple relationship expressing the ratio

of environment to sensor stiffness as:

$$\frac{K_E}{K_S} = NK_CK_P \qquad (8.12)$$

which states that, increases in environment stiffness can be accompanied by increases in proportional gain or sensor stiffness. This condition is the exact opposite to that found in earlier simulations and experiments. However, in the last series of experiments in section 7.8, where the sensor stiffness was lowered, unusual and previously unexplained results were obtained. Virtual limit cycles existed experimentally for low values of K_E, for the same controller gains which had previously been shown to be stable. The system's characteristic had obviously been affected by lowering the sensor stiffness from 10^5N/m to 10^4N/m. Where K_E increased, stability is shown to return, see experiment figure 7.11.

Another interesting fact concluded from equation 8.12, where $K_D=C_R=0$ and $M_1>M_2$, is that the conditional stability is independent of system inertias M_1 and M_2. This fact could possibly be exploited where changes in environment/sensor mass M_2 and robot inertia M_1 occur. However, in practice this result may be masked by other factors, such as, variation in K_E and manipulator configuration affecting the viscous damping component of the joints.

8.6.2 Variation in Sensor Damping

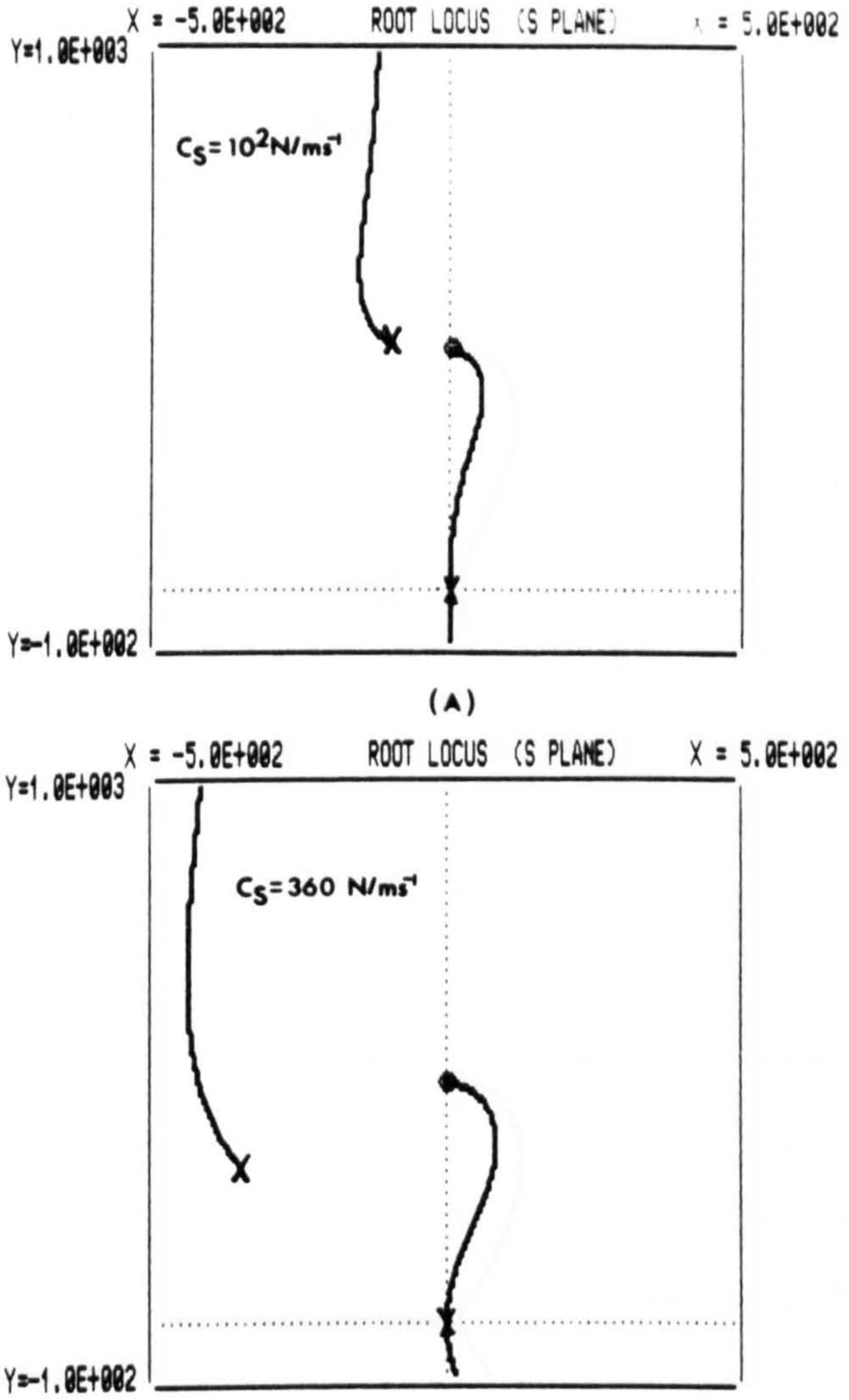
The following sequence of simulations considers the effect of variation in sensor damping on the system root locus and hence on the system response. The system is exactly that described in 8.6.1 above, where C_R and K_D are zero, thus the effect of C_S variation is considered independently. In drawing a comparison to the cantilever beam following

experiments, the sensor mass is considered to be 0.5Kg and not the 3Kg considered earlier. This is because the beam is less massive than the turbine blade. Four simulations have been carried out using the usual parameters, but importantly $K_E=10^5$N/m and $K_S=10^4$N/m and $K_p=0.3$ units. The variable C_S, takes four differing values; 10^2, 360, 10^3 and 10^4N/ms^{-1}. The experimentally determined value for C_S is 360 N/ms^{-1}.

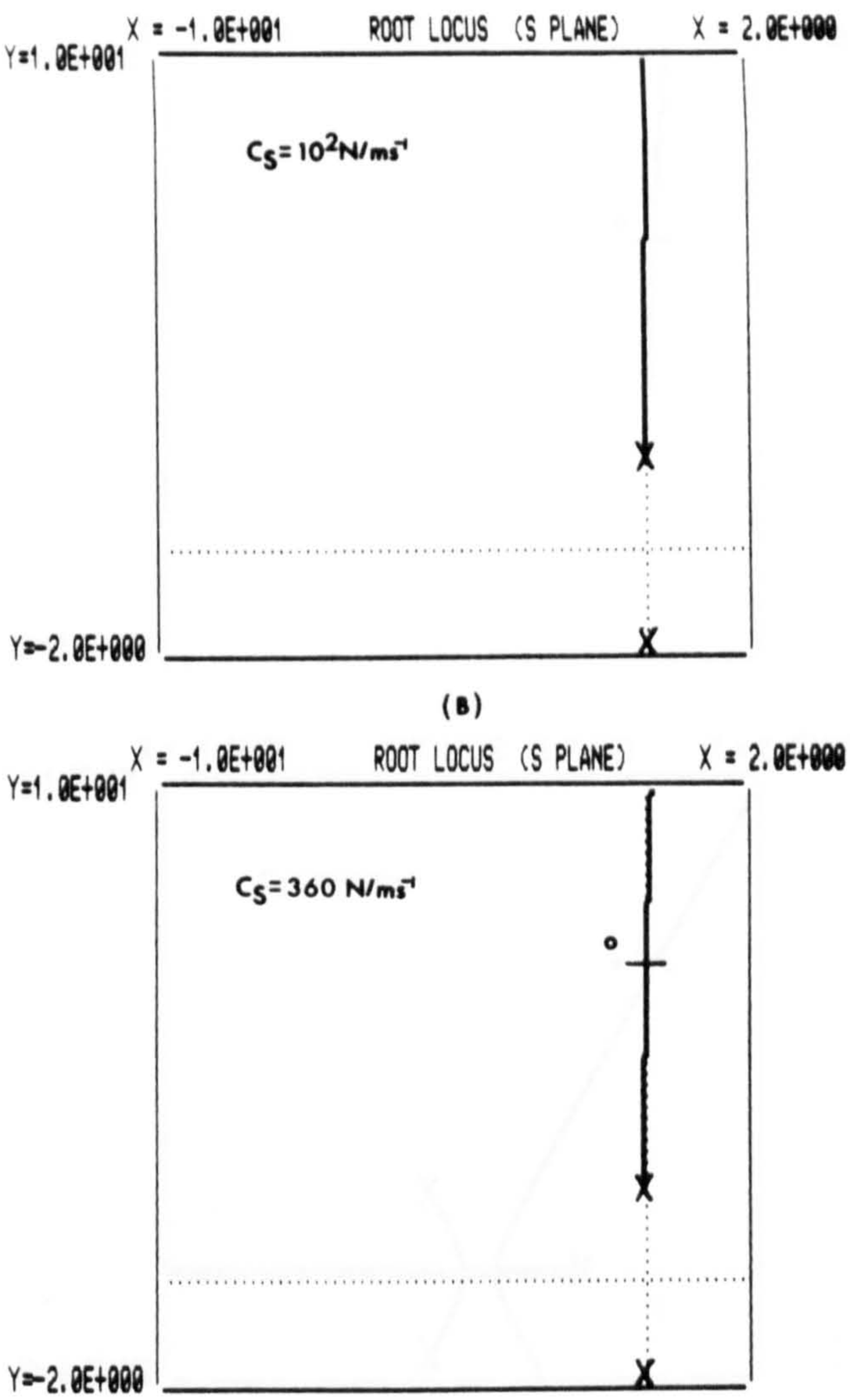
Adding sensor damping to the system does not increase the number of open-loop zeros. Therefore there are two open-loop zeros and four open-loop poles. This indicates two asymptotes one at 90 and the other at 270 degrees.

The first simulation, where $C_S=10^2$N/ms^{-1} shown in figure 8.14(a), indicates two significant effects. The open-loop poles associated with high frequency oscillation of the sensor mass are moved to the left away from the imaginary axis, indicating the effect the damping has on the sensor mass oscillation. The second and more important effect is the apparent instability of the fundamental and dominant locus. The locus is seen to start at the origin and to bend into the right hand side of the 'S' plane, terminating at the open loop zeros on the imaginary axes. Closer inspection near to the origin, in figure 8.14(b), indicates the locus crosses the imaginary axis, although the exact point is not clearly shown at such low values of C_S.

Increasing the sensor damping further to the actual experimental value, $C_S=360$N/ms^{-1} in figure 8.14(c), shows the sensor mass harmonic to be increasingly damped and the fundamental force response to be moved further into the right hand side of the 'S' plane. Closer examination of the fundamental response near to the origin in figure 8.14(d) reveals a definite crossing of the imaginary axis by the locus. The point of



(A)

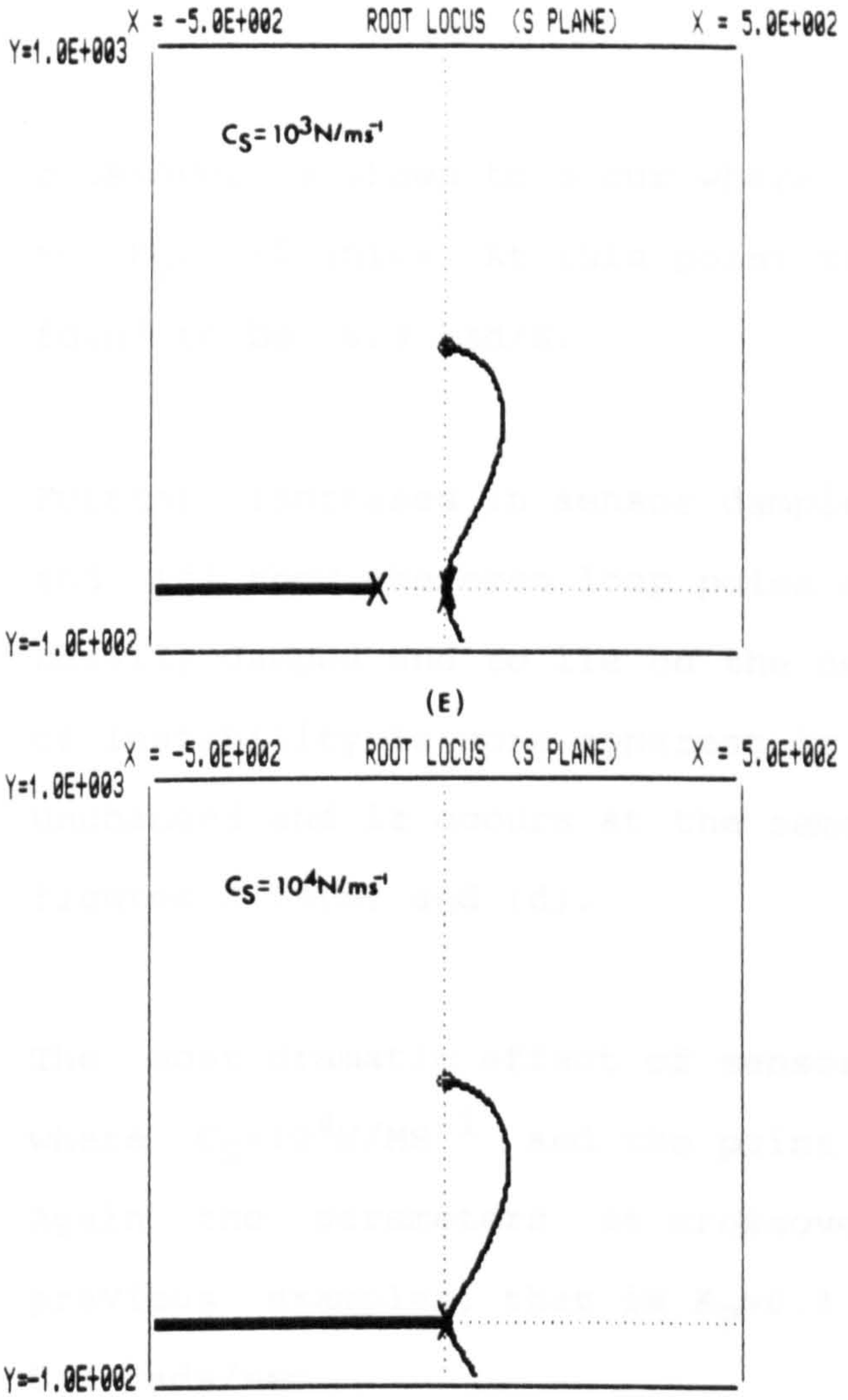


(B)

(C)

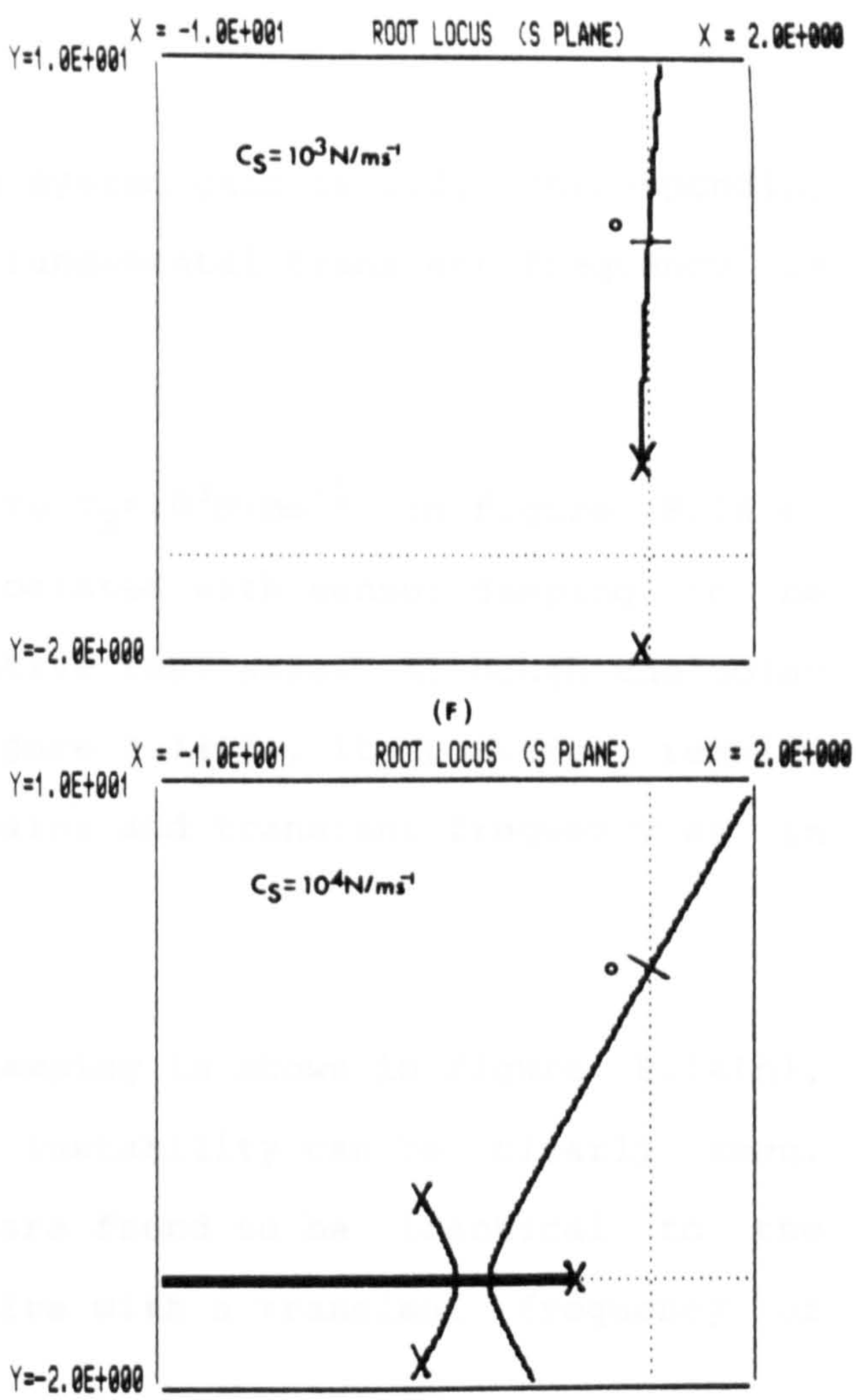
(D)

Figure 8.14 (a) to (h), continued overleaf.



(E)

(G)



(F)

(H)

Figures 8.14(a) to (h). Simulations showing the root locus plots of the single degree of freedom force controlled system with sensor damping C_S variation. Conditional stability is displayed.
M_1=0.5Kg, $K_S=10^4$N/m, $K_E=10^5$N/m, K_D=0, C_R=0, K_P=0.3 units. The operating points are shown by O. All other parameters take values defined in table 8.1. Figures: (a)&(b) $C_S=10^2 N/ms^{-1}$ (c)&(d) $C_S=360 N/ms^{-1}$, (e)&(f) $C_S=10^3 N/ms^{-1}$, (g)&(h) $C_S=10^4 N/ms^{-1}$.
Points of instability occur on all figures at K_P=0.33units at fundamental transient frequency of 6.4 rads./sec.

crossover is shown to occur where the system gain is 1.0, corresponding to K_P=0.30 units. At this point the fundamental transient frequency is found to be 6.3 rad/S.

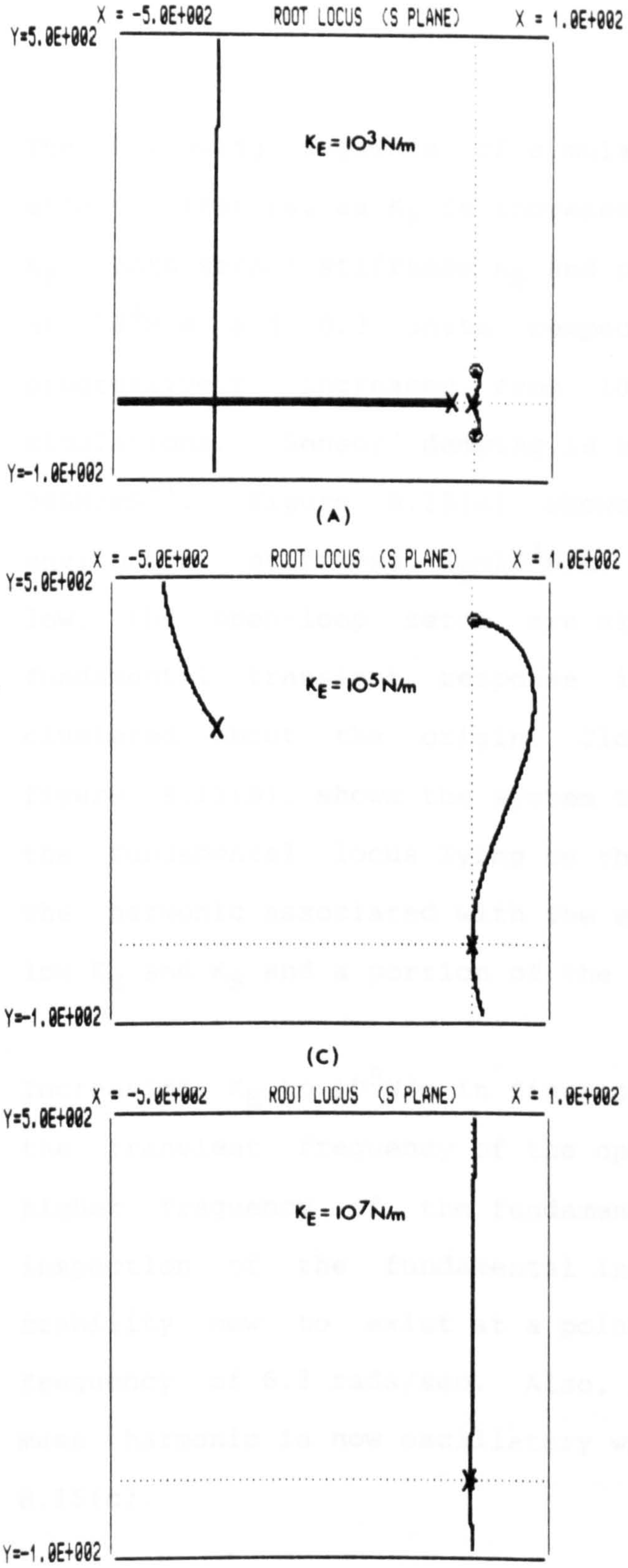
Further increases in sensor damping to $C_S=10^3 N/ms^{-1}$ in figure 8.14(e) and (f) show the open loop poles associated with sensor damping to be heavily damped and to lie on the negative real axis. Although the point of instability is more apparent in figure 8.14(f), its position remains unchanged and it occurs at the same gains and transient frequency as in figures 8.14(b) and (d).

The most dramatic effect of sensor damping is shown in figure 8.14(h), where $C_S=10^4 N/MS^{-1}$ and the point of instability can be clearly seen. Again the parameters at crossover are found to be identical to the previous examples, that is K_P=0.3 units with a transient frequency of 6.3 rads/sec.

These results confirm the conditions gained from the Routh Hurwitz analysis, $K_E=NK_CK_PK_S$ for the point of instability, showing that the condition is independent of C_S. The group NK_CK_P is the combined total proportional gain, where $NK_C=31.7N/Nm^{-}$, and K_P=0.3 at the point of crossover, giving a total proportional gain of 9.5 units. Hence the Routh Hurwitz condition $K_E=NK_CK_PK_S$ is confirmed for $K_S=10^4 N/m$ and $k_E=10^5 N/m$.

8.6.3 Variation in Environment Stiffness

For the system described, the effect of variation in K_E can be explained by equation 8.12, given as $K_E=NK_CK_P.K_S$.



Figures 8.15(a) to (f). Simulations showing the root locus plots of the single degree of freedom force controlled system with sensor damping. The effect of K_E variation on condition stability is shown. M_1=0.5Kg, K_S=10^4N/m, C_S=360N/ms^{-1}, K_D=0, C_R=0, K_P=0.3 units. The operating points are shown by O. All other parameters take values defined in table 8.1. Figures: (a)&(b) K_E=10^3N/m, (c)&(d) K_E=10^5N/m, (e)&(f) K_E=10^7N/m.

The following sequence of simulations are designed to confirm this effect, that is, as K_E is increased so stability improves for a given K_p. Both sensor stiffness K_S and proportional gain K_p are held constant at 10^4N/m and 0.3 units respectively. Environment stiffness is progressively increased from 10^3N/m, 10^5N/m and 10^7N/m in the simulations. Sensor damping is set at the experimental norm of C_S = 360N/mS^{-1}. Figure 8.15(a) shows the root locus diagram for low environment stiffness, $K_E=10^3$N/m. Since the environment stiffness is low, the open-loop zeros are also small, hence the locus of the fundamental transient response is shown to be low frequency and clustered about the origin. Closer analysis near to the origin in figure 8.15(b), shows the system to be unstable for $K_p=0.3$ units, with the fundamental locus lying on the right hand side of the 'S' plane. The harmonic associated with the sensor mass is overdamped, owing to a low K_E and K_S and a portion of the locus lies on the negative real axis.

Increasing K_E to 10^5N/m in figures 8.15(c) & (d) shows an increase in the transient frequency of the open loop zeros, hence a potentially higher frequency of the fundamental response could result. Closer inspection of the fundamental in figure 8.15(d), shows conditional stability now to exist at a point where $K_p=0.33$, with a transient frequency of 6.3 rads/sec. Also, due to the increased K_E, the sensor mass harmonic is now oscillatory with reduced damping, shown in figure 8.15(c).

Where K_E is finally increased to 10^7N/m, figure 8.15(e), the open loop zeros have large imaginary components and are consequently off the diagram. As a result the sensor mass harmonic is both high frequency and increasingly underdamped.

Closer inspection of the fundamental transient response, in figure 8.15(f), near to the point of imaginary axis cross over, indicates the limit of stability is given by a proportional gain of 33 units with a transient frequency of 63 rads/sec.

Comparing the results of figure 8.15(d) and (f) confirms the Routh Hurwitz condition for stability given in equation 8.12. Confirmation is seen by increasing K_E from 10^5 to 10^7N/m, this shows a corresponding increase in K_p from 0.33 to 33 units at the point of instability on the imaginary axis. Therefore, in the experimentation where K_p is maintained constant as K_E varies, for high K_E the sensor damping effect no longer becomes the mechanism of instability. However, if under the same conditions K_E is reduced, then K_p must be reduced to avoid instability, as shown by figure 8.15(b).

This explanation confirms the experimental effects displayed in figure 7.11, where using a soft force sensor, K_S=10^4N/m, near instability arose when tracking the free end of the cantilever, despite the increased velocity damping effects. The situation improved as K_E increased towards the built in end, where the system can tolerate higher values of proportional gain.

For low values of K_E sensor damping is significant and is likely to cause instability. For high values of K_E, the C_S effect is unlikely to cause instability, as instability is more typically caused by the effects already discussed, in particular the reduction in velocity damping caused by the increased system stiffness.

Certainly the practical system is further complicated by the inherent velocity damping terms, section 8.6.5 gives brief consideration to this.

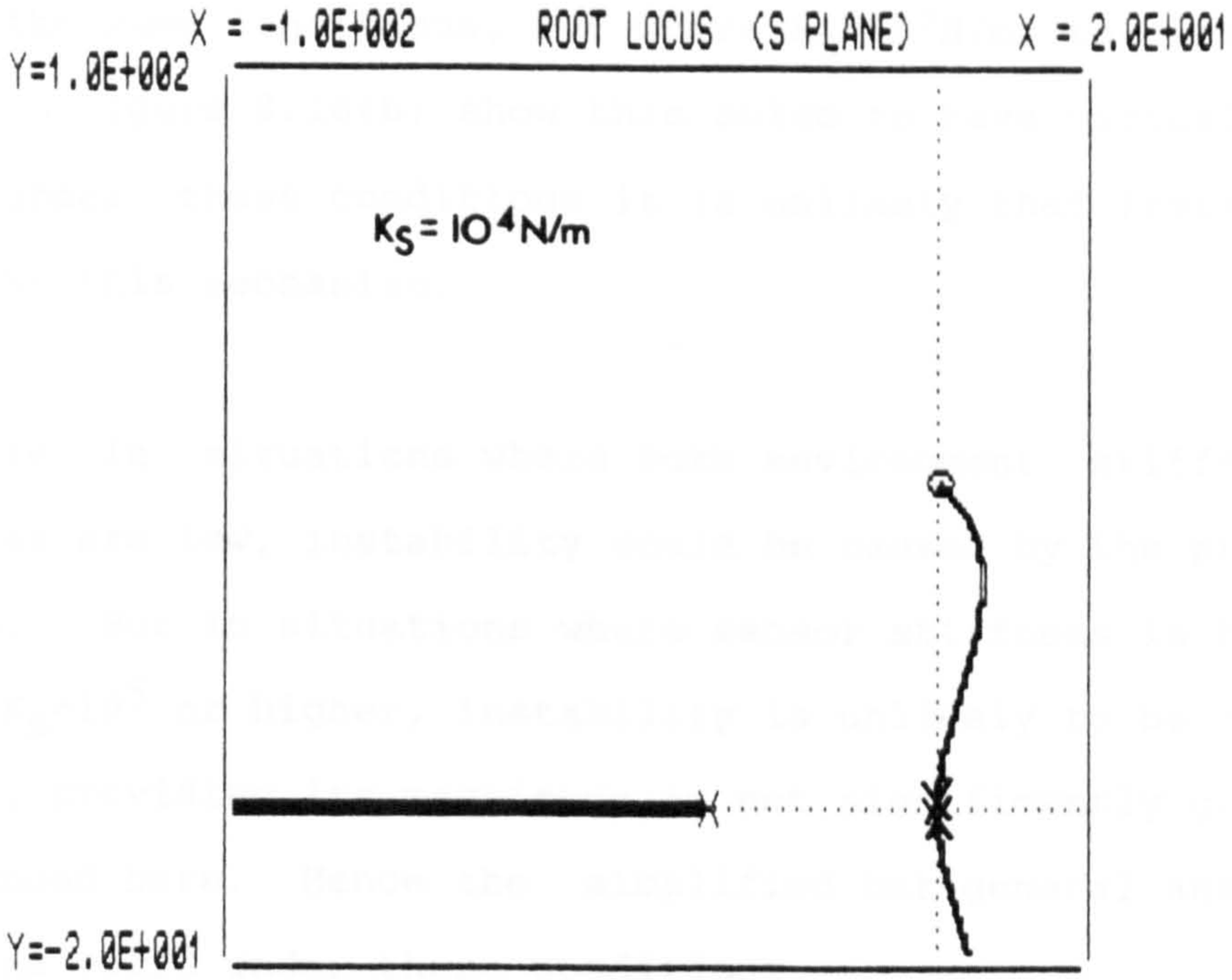
8.6.4 Variation in Sensor Stiffness

The experimental work of section 7.8 shows unusual effects in force response as the sensor stiffness is reduced from 10^5N/m to 10^4N/m. The effects have been postulated to be due to the sensor damping C_S becoming more significant as the sensor stiffness is reduced.

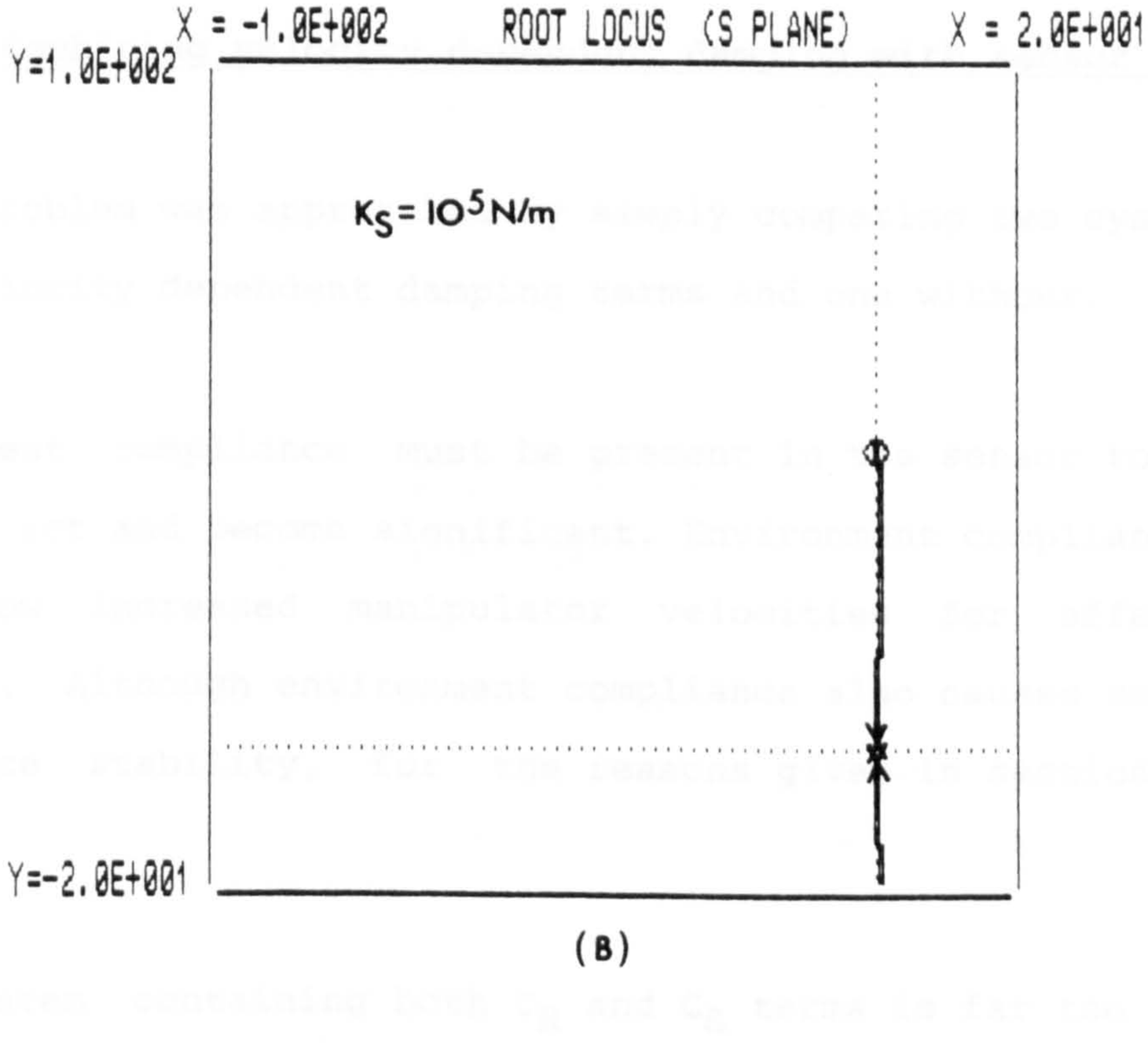
The following sequence of simulations have been designed to investigate the apparent dependence of the sensor damping effect on sensor stiffness, and also whether the assumption of the earlier analysis, in ignoring the presence of C_S, was indeed justified for the case where $K_S=10^5$N/m.

The two relevant experiments both employ the cantilever as the varying environment stiffness. The experimental results are given in figures 7.11(a) and (b) showing the results for $K_S=10^5$N/m and for the soft sensor at $K_S=10^4$N/m respectively. In both experiments K_P remains constant at $K_P=5.0$ units. The effects of the soft sensor appear to be more apparent at the free end where $K_E=10^3$N/m, compared to those results where $K_S=10^5$N/m. This is explained by the analysis given above in section 8.6.3, covering variation in environment stiffness. The effect of lowering sensor stiffness is shown to increase the oscillatory nature of the response and to cause a near instability at the cantilever free end. The instability takes the form of a low frequency limit cycle.

Two simulations display the effect of sensor stiffness on the conditional stability caused by sensor damping. Both simulations have the same proportional gain, $K_P=0.3$ and environment stiffness, $K_{E=}10^3$N/m. Figure 8.16(a) shows the simulation results for a soft sensor, $K_S=10^4$N/m, showing the bulge of the fundamental force response locus into the right hand side of the 'S' plane, hence causing instability.



(A)



(B)

Figures 8.16(a) and (b). Simulations showing the root locus plots of the single degree of freedom force controlled system with sensor damping. The effect of K_S variation on the degree of instability is displayed for a compliant environment condition.
M_1=0.5Kg, K_E=10^3N/m, C_S=360N/ms^{-1}, K_D=0, C_R=0, K_P=0.3 units. The operating points are shown by O. All other parameters take values defined in table 8.1. Figures: (a) K_S=10^4N/m, (b) K_S=10^5N/m.

Under the same conditions, but where $K_S=10^5$N/m, the simulation results shown in figure 8.16(b) show this bulge to have virtually disappeared. Hence under these conditions it is unlikely that instability will be caused by this mechanism.

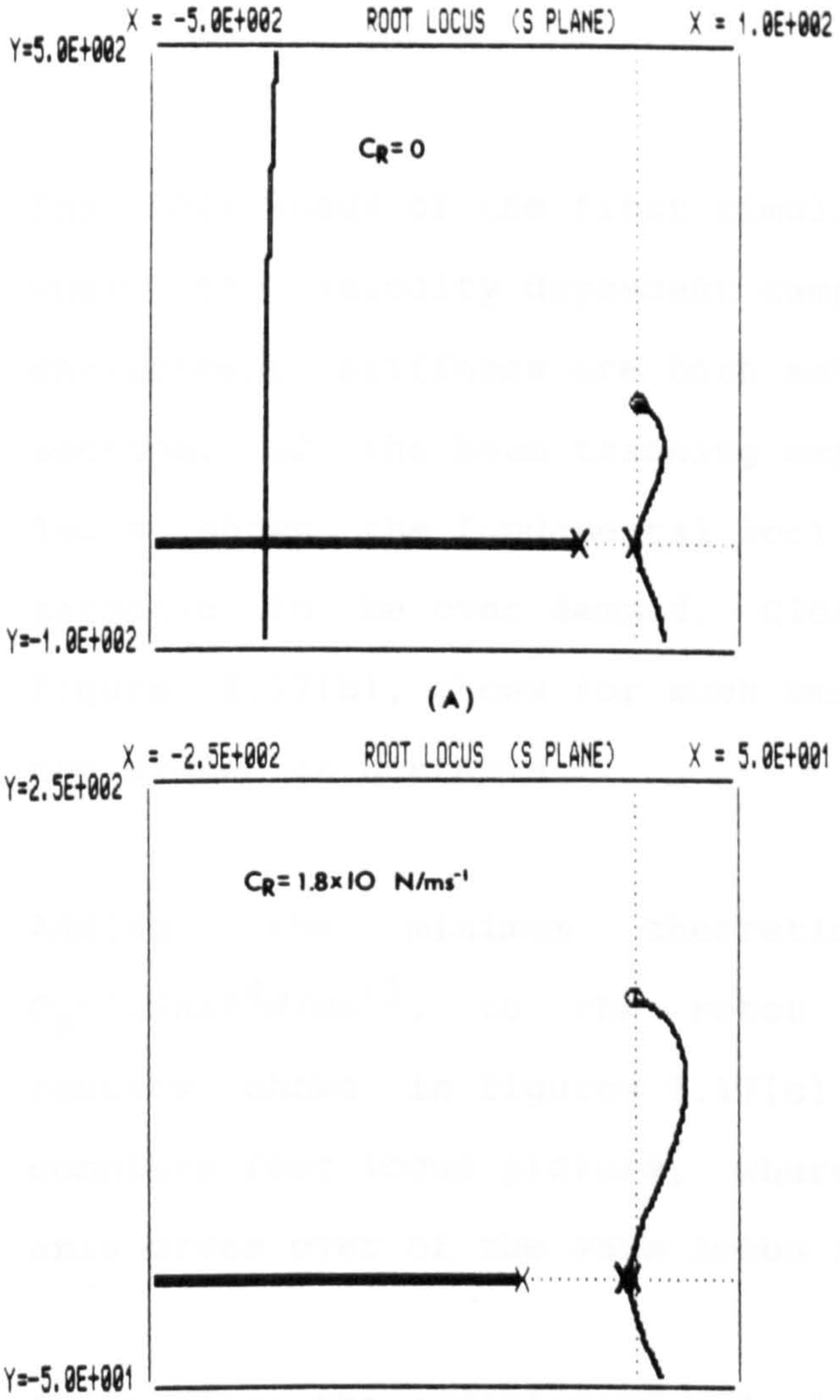
Therefore in situations where both environment stiffness and sensor stiffness are low, instability could be caused by the presence of sensor damping. But in situations where sensor stiffness is high, as in this case, $K_S=10^5$ or higher, instability is unlikely to be caused by sensor damping, providing its magnitude is not significantly greater than that experienced here. Hence the simplified but general analysis of chapter 5 remains valid under these conditions.

8.6.5 Combining velocity dependent damping with sensor damping

This problem was approached by simply comparing two systems, one with velocity dependent damping terms and one without.

Sufficient compliance must be present in the sensor to enable the C_S term to act and become significant. Environment compliance is necessary to allow increased manipulator velocities for effective velocity damping. Although environment compliance also causes sensor damping to influence stability, for the reasons given in sections 8.6.3 and 4 above.

The system containing both C_R and C_S terms is far too complex to be considered by a Routh Hurwitz analysis. Simulation was therefore proposed to perform further investigations. Although rigorous results cannot be obtained from these brief simulations, they are sufficient to illustrate the combined effects of C_S and C_R.



Figures 8.17(a) to (d). Simulations showing the root locus plots of the single degree of freedom force controlled system with sensor damping. The effect of velocity dependent damping, C_R, on stability is investigated.

M_1=0.5Kg, K_E=10^4N/m, K_S=10^4N/m, C_S=360N/ms^{-1}, K_D=0. Operating points are shown at K_P=0.3, 5.0 and 7.2 units. All other parameters take values defined in table 8.1. Figures: (a)&(d) C_R=0, (c)&(d) C_R=1.8x10^4N/ms^{-1}.

The root locus of the first simulation is presented in figure 8.17(a) where the velocity dependent damping term C_R is omitted. Sensor and environment stiffness are both set at 10^4N/m, typical of the beam mid-section, of the beam tracking experiment, figure 7.11(b). The root locus shows the fundamental loci to be unstable and the sensor mass harmonic to be over damped. Closer inspection of the fundamental in figure 8.17(b), shows for such small proportional gains where K_p=0.3, the system is unstable.

Adding the minimum theoretical velocity dependent damping, C_R=1.8x10^4N/ms^{-1}, to the robot model in the simulation, gives the results shown in figures 8.17(c) and (d). Figure 8.17(c) shows the complete root locus picture, whereas figure 8.17(d) shows the imaginary axis cross over of the same locus in more detail.

It is clear the addition of velocity dependent damping has a significant effect on this form of instability. The system is completely stable for gains of K_p=0.3 or less. But in the experimental work, figure 7.11, K_p=5.0 units, the corresponding point is indicated on figure 8.17(d) and shows a lightly damped transient oscillation of 17 rads/sec or 2.7Hz. Taking approximate measurements from the experimental figure, at the point corresponding to an environment stiffness of 10^4N/m, shows a lightly damped oscillation of approximately 3Hz. Such agreement between simulation and experiment helps to confirm the validity of the model.

8.6.6 Sensor Damping Modelling

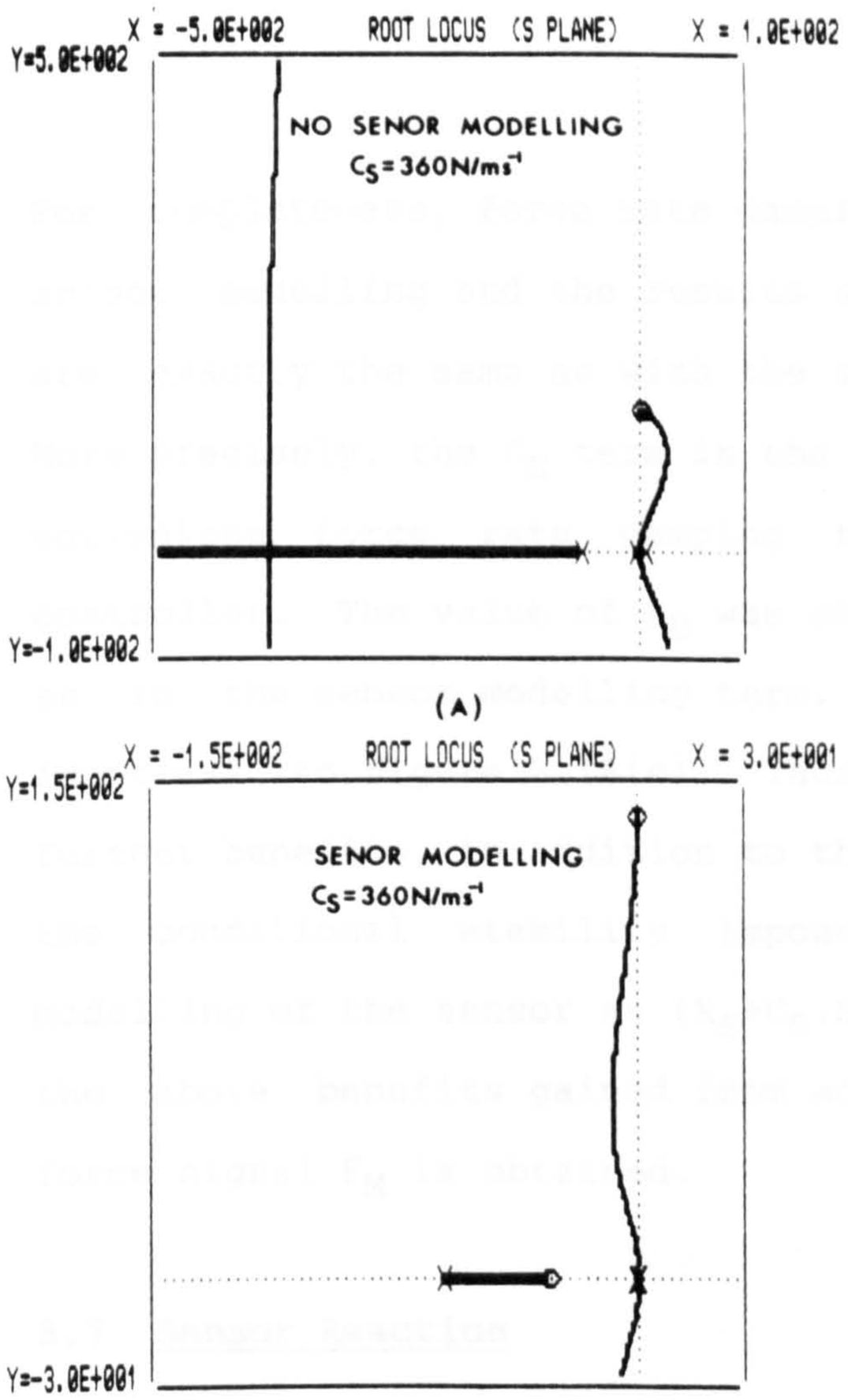
Throughout this work the force in the sensor has been measured by using the sensor stiffness K_S and the measured sensor deflection (X_2-X_1) as indicated in figure 8.3. The component of sensor force dependent on sensor velocity and C_S, has been ignored in the calculation of measured

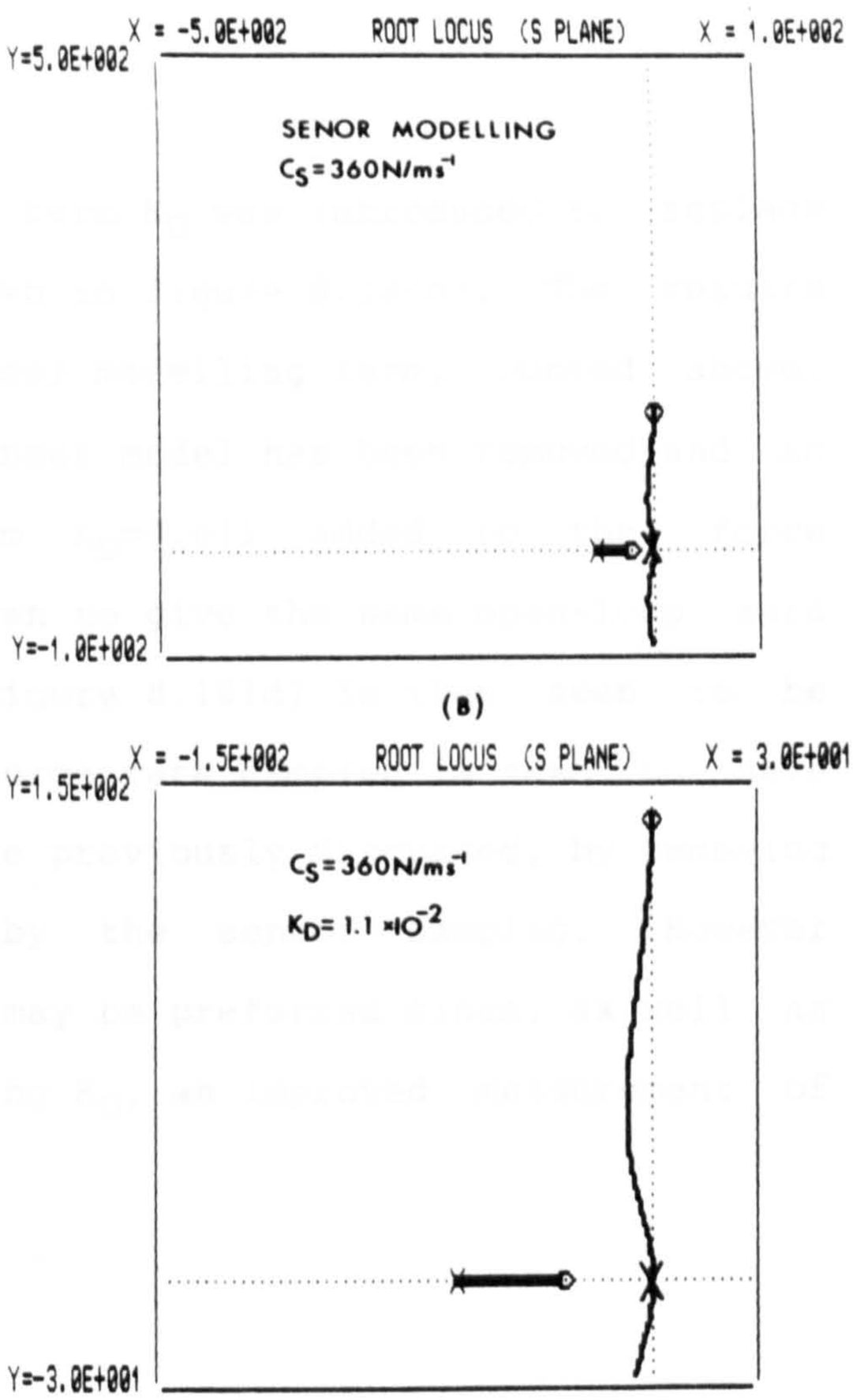
sensor force, F_M. It is thought the error between the actual sensor force F_S and the measured sensor force F_M could be the cause of the conditional stability associated with the presence of C_S. To this end, the following simulations have been carried out in an attempt to confirm this mechanism.

Taking figure 8.3, but without the sensor reaction elimination loop, and replacing the sensor model, K_S, by a more accurate model (K_S+C_SS) has the effect of adding an open loop zero to the system. This effect is identical to adding a force rate term K_D to the force controller. Unfortunately a Routh Hurwitz analysis proves too complex to derive meaningful and specific conclusions under these conditions. Therefore a detailed analysis is left to future work. Let it suffice at this stage to introduce the benefits of such a technique by implementing an improved sensor model.

Figure 8.18(a) shows the root locus of an identical system to that used in figure 8.17(a) of the previous 8.6.5 section. This figure displays the usual conditional stability associated with C_S.

The results of the simulations which include the improved sensor modelling are shown in figure 8.18(b), where C_S in the model takes the same value as system, $C_S=360N/ms^{-1}$. In a comparison with figure 8.18(a), figure 8.18(b) shows definite improvement in stability. The additional third open-loop zero is seen to reduce the number of asymptotes from two to one. Closer inspection of figure 8.18(b), displayed under higher magnification in figure 8.18(c), shows the conditional stability to now be completely removed. The locus of the fundamental response is seen to exist totally in the left hand side of the 'S' plane and does not traverse the imaginary axis.





(C) (D)

Figures 8.18(a) to (d). Simulations showing the root locus plots of the single degree of freedom force controlled system with sensor damping. The effect of sensor modelling is investigated. M_1=0.5Kg, K_E=10^4N/m, K_S=10^4N/m, C_S=360N/ms^{-1}, K_D=0, C_R=0. All other parameters take values defined in table 8.1. Figures: (a) no sensor modelling, (b)&(c) sensor modelling, (d) no sensor modelling with K_D=1.1x10^{-2}units.

For completeness, force rate damping term K_D was introduced to replace sensor modelling and the results shown in figure 8.18(d). The results are exactly the same as with the sensor modelling term, quoted above. More precisely, the C_S term in the sensor model has been removed and an equivalent force rate damping term K_D=0.011 added to the force controller. The value of K_D was chosen to give the same open-loop zero as in the sensor modelling term. Figure 8.18(d) is thus seen to be identical to figure 8.18(c). Thus force rate damping is seen to have further benefits, in addition to those previously discussed, by removing the conditional stability imposed by the sensor damping. However modelling of the sensor as (K_S+C_S.S) may be preferred since, as well as the above benefits gained from adding K_D, an improved measurement of force signal F_M is obtained.

8.7 Sensor Reaction

Considered in the analysis chapter are the possible benefits to be gained from eliminating the sensor reaction effects. The experimental chapter demonstrated these benefits by implementing a practical scheme to eliminate the sensor reactions. Although the results were successful at high force levels at low force levels the effect of joint frictions cloud the results.

Therefore, to confirm the previous results and the mechanism of steady state error, single degree of freedom simulation has been implemented in this study.

Figure 8.19(a) shows the time response to a unit step in force of the typical force control system. The proportional gain of the system has been reduced to K_p=0.03 to emphasis the effect of steady state error. The figure shows the measured force to have reached 49% of the desired

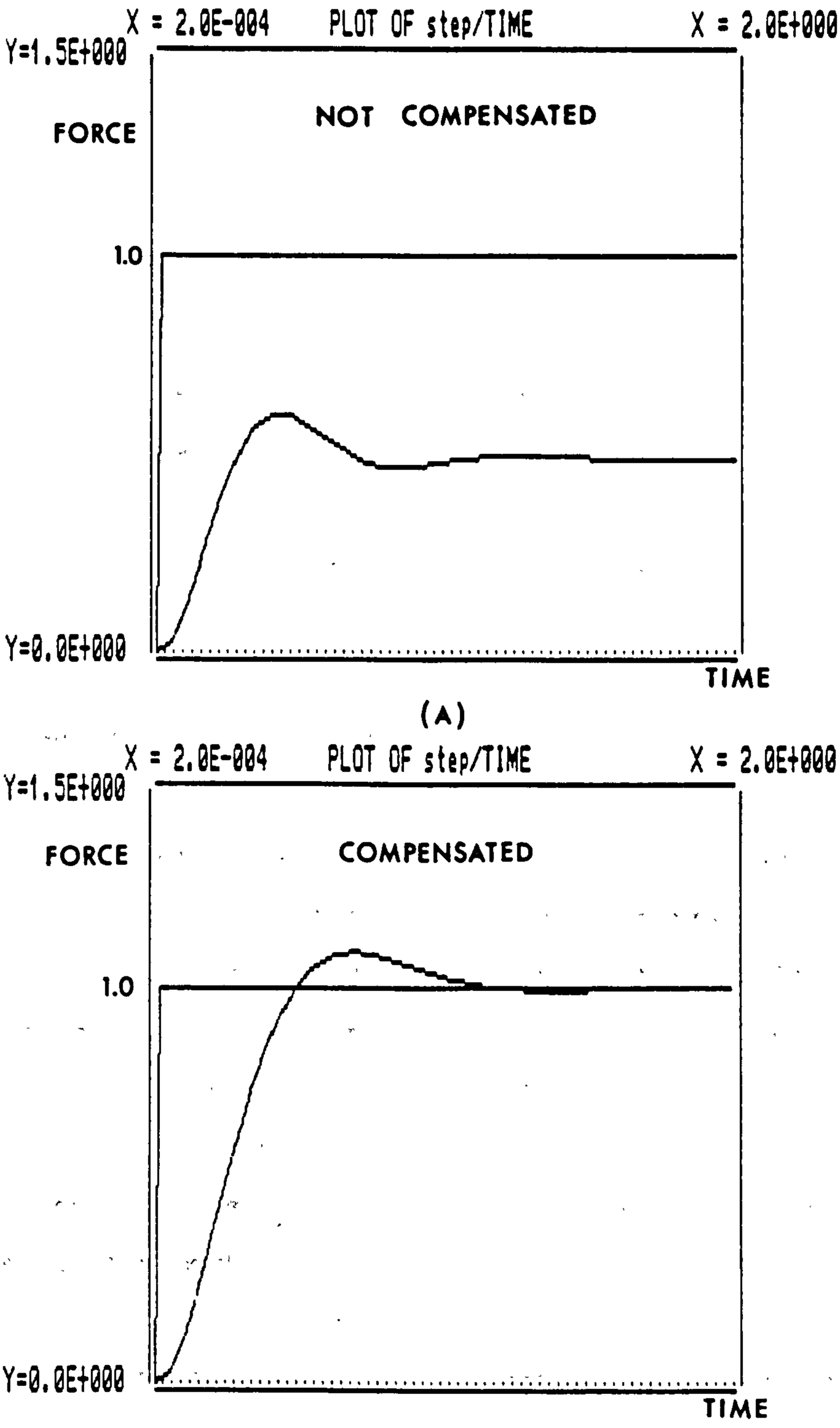
value. Rewriting equation 5.9, from the analysis chapter, in terms of the single degree of freedom analysis gives:

$$N.K_C.K_P.F_D=(1+NK_CK_P).F_M \qquad (8.10)$$

where F_D and F_M are the desired and measured forces respectively. Since N and K_C are constant, steady state error is dependent only on the magnitude of K_p.

Therefore equation 8.10, for the conditions given, gives the theoretical steady state force output as 48.7% of the desired value, confirming the simulation and analyses.

Implementing the sensor reaction compensation loop described in figure 8.3 into simulation gives the result shown in figure 8.19(b). All parameters including K_p remain unchanged between figures 8.19(a) and (b). The steady state error has been reduced to near zero without affecting stability. In fact, the response of figure 8.19(b) can be seen to have slightly improved stability, displaying a less oscillatory response with a lower transient frequency. This is caused by an effective reduction of proportional feedback by the elimination of the unity term in equation 8.10.



(A)

(B)

Figures 8.19(a) and (b). Simulations showing the force response to a unit step input for the single degree of freedom force controlled system. Force sensor reaction compensation is investigated and shows steady state error elimination and improved stability.
M_1=0.5Kg, K_E=10^6N/m, K_S=10^4N/m, C_S=360N/ms^{-1}, K_D=0, C_R=1.8x10^4N/ms^{-1}.
All other parameters take values defined in table 8.1. Figures: (a) not compensated, (b) compensated.

CHAPTER 9

GENERAL DISCUSSIONS

9.1 General

The research work has been directed towards the problem of automatically polishing turbine blading. However the same techniques could be applied to most surface processing where control of force is required and disturbance forces resisted. Four main aspects to the work have been considered, they are: (a) the polishing task approach, (b) the experimental facility, (c) simultaneous force and position control scheme and (d) force control stability.

(a) The polishing task approach imposes the additional positional constraint of maintaining the force sensor normal to the blade surface. This has distinct advantages: primarily it decouples the unknown and unwanted polishing friction forces from the normal controlled force, it allows the utilization of a simple single degree of freedom force sensor, no transformations are required between the sensor and local coordinates of the surface, the friction forces can be neglected and resisted by the mechanical structure of the robot.

(b) The experimental facility was restricted to a planar three degree of freedom manipulator able to operate around a two dimensional blade profile. Manipulator design exploited the unique aspect of combined parallel and serial drives. The triangulated parallel drive gives the manipulator high strength, accuracy and stiffness, and enables it to develop forces of up to 4000N at the end effector. The manipulator is controlled by two synchronized microcomputers. One computer is the low

level position/force controller while the other behaves as the trajectory planner.

(c) The simultaneous force and position control scheme, although fundamentally the same as the Hybrid scheme devised by Raibert and Craig, differs in a major aspect. Unlike the original Hybrid scheme, where force control operates in joint coordinates, here the force controller operates in local Cartesian coordinates. Subsequent force control analyses show this to be a natural coordinate system with computational advantages. Also the compliance selection is achieved by selecting the relevant elements of the transformation matrices, rather than using the complete matrix as Raibert did. This has significant computational advantages.

(d) Force control stability has proved to be the major obstacle preventing progress of the research towards satisfactory simultaneous force and position control. Elsewhere, limited attention has been given to force control stability and even less to related experimental work. The manipulator designed here has been shown to be an extremely useful tool in evaluating the influential properties and parameters of force control. Sufficient understanding of this area must be gained before work can be usefully and reliably used in surface processing applications. To this end a substantial part of the research has been directed at force control, beginning with a general analysis, followed by experimental work and a more specific single degree of freedom analysis and digital simulations. The following summary is devoted to a major part of the thesis, the factors influencing force control.

9.2 Force Control

Initially, force control based on passive impedance control techniques was considered. The passive aspect of the techniques presents major disadvantages. Reactive forces, generated by errors in position control, are controlled by adjustments in desired position, based on measured contact forces. Since such a system cannot operate at servo cycle speed, bandwidth is low. Also, friction in the robot transmissions was shown to be a serious limitation as it impedes back driving. The technique is therefore only suitable for directly driven manipulators or those with limited transmission gearing. Consequently these design limitations restrict the technique to low force applications.

Active control of end effector force, utilising a wrist force sensor, was next considered. In principle this technique can be used with manipulators possessing transmissions and hence large forces can be achieved. Dynamic equations governing six degrees of freedom were established. Modelling the effects of transmission stiffnesses produced a sixth order system. But for the experimental manipulator transmission stiffnesses are extremely high in comparison to that of the sensor and are therefore assumed infinite. Thus grouping the inertia of the actuator and links permitted a reduction in the order of the system to four. The subsequent simplification allowed the the characteristic matrix to be determined in partitioned form. The effects of individual parameters on the system's characteristics are discussed.

A more specific analysis was performed by reducing the force controlled system to a single degree of freedom case which involved some simulation. At this single degree of freedom stage, sensor damping is introduced in an attempt to explain the unpredictable experimental results experienced when the sensor stiffness was lowered.

Analysis showed the response to be dependent on many parameters belonging to the manipulator, sensor, environment and force controller. Influential parameters are : (1) manipulator's inertia, geometry and damping, (2) sensor's stiffness, inertia and damping, (3) environment's stiffness and inertia, (4) force controller's gains for proportional, force rate and velocity damping.

One of the most significant problems in force control is the effect of unpredictable variation in the environment's stiffness. Environment stiffness proves to be an extremely significant parameter and is shown experimentally to have a dramatic effect on stability and force response. A means of artificially maintaining the environment stiffness constant was implemented experimentally. By the addition of a compliant element to the sensor end, the problems associated with the varying environment stiffness were shown to be removed. This technique was exploited to display the effect a changing manipulator configuration has on force control. A means of automatically adjusting the force controller parameters under changing manipulator geometry was shown to be necessary. A configuration dependent look-up table was suggested as a possible solution. These results have direct application to grinding/polishing using a compliant bob whose effect can be considered to lower the environment stiffness.

The analysis of the single degree of freedom system showed that the basic force controlled system is composed of two open-loop zeros and four open-loop poles with unity feedback. The zero placements are dependent on the combined sensor and environment inertia and the environment stiffness. Since environment damping in most engineering applications is negligible and in any case cannot be relied upon, the two open-loop zeros have no real parts and lie on the imaginary axis.

Hence for large values of system gain the system tends to zero damping and instability, confirming experimental findings. Adding a force rate term to the controller leaves these two zeros unaffected and adds a further zero to the negative real axis.

Open-loop pole placements are dependent on all system parameters except those of the force controller. Where velocity damping is not present, all poles lie on the the imaginary axis and hence the system proves unstable. In practice however, friction and velocity dependent damping are always present in the actuators, but under certain conditions instability can still occur. Such cases of instability have been given considerable attention in simulation analyses and experimental work.

The forward proportional gain of the open-loop system is found to be the product $N.K_C.K_P.K_E.K_S$ which, under certain circumstances, can dominate the characteristic equation if the individual terms become excessively large. In agreement with this, increases in proportional gain, environment and sensor stiffnesses were shown experimentally to raise transient frequency and reduce damping of the transient response, leading to instability in some cases. Under these conditions analysis shows conditional stability should not arise but, if the proportional gain reaches extremes, damping is shown in simulations to become virtually zero and instability soon occurs.

Steady state error is a feature of the force controlled system. By a particular arrangement of the characteristic matrix and block control diagram, the effect is shown to be caused by sensor reactions disturbing the mechanical manipulator. Experimental attempts to introduce integral control to eliminate these errors were not entirely successful, as the integral term caused the force response to develop a low frequency oscillation which was slow to decay. Analysis showed the relationship

between steady state measured and desired forces using a P+D controller as:

$$(\underline{J}^T + \underline{N}.\underline{K}_C.\underline{J}^T.\underline{K}_p)^{-1}.(\underline{N}.\underline{K}_C.\underline{J}^T.\underline{K}_p) = {}^T\underline{F}_S.({}^T\underline{F}_D)^{-1}$$

This was confirmed by experiment and in simulation for the single degree of freedom case.

By implementing a new and original technique of measuring the force disturbance at the force sensor and feeding it back into the force controller to counteract the sensor reaction disturbance, steady state error due to this cause was eliminated. The technique was confirmed by both simulation and experiment. Implementation of the technique is extremely simple in practice and effective. The application of this technique should not be limited to force control but have applications to all areas of disturbed robotic control.

Other steady state errors in force response were identified in the experimentation as being due to unwanted static friction forces in the transmissions, largely random in nature. These errors were shown to be reduced in direct proportion to an increasing proportional gain and were to be less significant at higher transmitted force levels. This property made force control at low force levels and with low proportional gains particularly troublesome. The problem disappeared when higher values of proportional gain were permitted, for example when dealing with an environment or sensor of low stiffness. Successful modelling of static friction is difficult, if not impossible. The implementation of an invariance technique to eliminate or reduce these friction effects acting at the joints is therefore suggested.

Velocity dependent damping was shown experimentally, analytically and in simulations to be a significant parameter in maintaining stability. The

majority of the damping was shown to act in joint coordinates resulting in configuration dependent damping effects. Overall mechanical system compliance greatly influences the velocities of the manipulator joints and hence these velocity damping effects. Thus, where sensor and environment stiffnesses are high, stability has been shown to be difficult to achieve experimentally. This has been confirmed by both simulation and analysis. Therefore, if velocity dependent damping is the only term promoting stability then mechanical stiffness must be lowered for it to be effective. It is suggested to introduce velocity feedback based in sensor coordinates. This is controlled to dominate all velocity damping and thus preserve a measure of velocity damping independent of configuration. However, experimentally the resolution of the velocity signals limited the benefits to compliant situations.

An additional form of damping was proposed, force rate damping. Unfortunately the quality of the experimental force rate signal was poor, preventing conclusive results. However, simulation and analysis show potentially important benefits to be gained from its adoption. Force rate terms in the characteristic equation increase with increasing environment and sensor stiffnesses, this holds the possibility of stability even at high stiffness levels. But at low stiffness levels velocity damping is shown, in simulation at least, to be the most effective of the two. Combining both force rate and velocity based damping would give the best performance under hard and soft environment conditions. But until a suitable force rate sensor is available, velocity based damping remains essential in maintaining stability.

Lowering sensor stiffness gave surprising experimental results where, instead of the expected improvement in stability, instability occurred. This can be explained by considering the sensor's viscous damping

effects becoming increasingly dominant as the sensor's movement increases. The presence of sensor damping has been shown to create conditional stability. Analysis shows for the single degree of freedom case, where actuator velocity damping is ignored, conditional stability is given by:

$$N.K_C.K_P = \frac{K_E}{K_S}$$

This implies that instability occurs as the environment stiffness is decreased, the opposite condition to that where sensor damping is considered insignificant. This trend was confirmed experimentally and in simulation. Including the effects of actuator based velocity dependent damping in simulations, simply shows a movement of the characteristic roots to the left, helping to promote stability.

Attempts to model sensor damping to gain a better estimate of sensor force shows, in simulation, the possibility of eliminating these detrimental effects completely. Modelling sensor damping adds a open-loop zero to the system, this is identical in principle to adding force rate damping. Therefore, in situations where sensor damping is desirable, for example to damp tool vibrations, the introduction of force rate damping provides further benefits.

CHAPTER 10

CONCLUSIONS

10.1 General

A unique 3 degree of freedom manipulator has been designed and developed capable of applying controlled end effector force whilst negotiating a 2 dimensional contour. The manipulator employs a triangulated ballscrew structure which can apply extremely high loads, and also provide for the gross motion of the manipulator.

Twin computers communicate in a synchronized manner to perform the complete control function. One computer has overall control and performs the trajectory planning function. This passes instructions down to the second computer which performs the low level control and interface communications with the manipulator.

A variant of the Hybrid force/position control scheme has been successfully implemented. It differs from the original scheme in that the force and position controllers are software based and operate in the current Cartesian coordinate system. This approach reduces the demands of the Hybrid controller and, with force control, proves to be a more natural approach.

Friction forces within the existing force sensor have prevented actual surface polishing from being attempted here. However, a polishing 'philosophy' has been proposed and implemented, which eliminates many problems associated with force control under these conditions. The adopted approach continuously orientates the force sensor normal to the

surface while controlling normal contact force and tangential movement. This decouples at source the unwanted tangential friction force of the polishing force from the desired controlled force component.

10.2 Force Control

Early experimental work showed that force control stability presented major problems and as a consequence the dynamics of force control stability has been extensively investigated. Initially, in chapter 4, a general 6 degree of freedom analysis involving passive force control was performed. This was followed, in chapter 5, by a general analysis of the dynamics of actively controlling end effector force. Subsequently a more specific analysis of the dynamics of actively controlling force was performed in single degree of freedom experimentation and simulations, (chapters 7 and 8 respectively).

Arising from this work are a number of conclusions conveniently presented in the following list.

(a) Simultaneous force and position control has been implemented on a 3 degree of freedom manipulator using a modified Hybrid scheme which is able to track and apply normal forces to planar two dimensional contours.

(b) Analyses showed the parameters affecting force response are : (i) manipulator's inertia, geometry and damping. (ii) sensor's stiffness, inertia and damping. (iii) environment's stiffness and inertia. (iv) all force controller gains. In general maintaining force response is impossible using the proposed force controller. Only under special conditions can force response be maintained constant while parameters vary.

(c) Mechanical compliances within the sensor and environment have been shown to be extremely important elements by appearing in the overall gain of the force controlled system. High stiffnesses in these elements necessitate lower controller gains.

(d) For the mechanical system presented stability of the force controlled system is due to the presence of velocity dependent damping effects. Sufficient mechanical compliance must be present to allow the velocity dependent damping terms to be effective.

(e) Sensor damping may provide benefit in limiting end effector tool vibrations, but its presence has been shown to give rise to conditional instability. Under these conditions increases in environment stiffness improves stability. These findings have been confirmed by experiment and simulation.

(f) Friction in the transmissions has two main effects, the viscous component tends to promote stability (contributing to velocity dependent damping), while the static components introduce unpredictable but repeatable errors in force response. Providing increases in proportional force controller gain can be tolerated, then a proportionate reduction in these errors is achieved.

(g) Force rate damping has been shown in the simulations and analyses to be particularly effective in producing stability in situations where mechanical stiffnesses of the sensor and environment are high and velocities low. But, owing to the poor quality of the force rate signals, this could not be confirmed experimentally. Combining both force rate and velocity dependent damping was shown in simulation to have the greatest potential for maintaining stability over a wide range of mechanical stiffnesses.

(h) The mechanism causing steady state error in the controlled force response has been analysed and has been shown to be caused by sensor reactions. As a consequence a simple but effective technique has been developed and implemented to eliminate steady state error in the force response. The technique involves minimal additional computations by feeding back a portion of the measured sensor force. This technique has been used here in preference to integral control or feed forward techniques which display unwanted side effects.

(i) By adding to the end of the force sensor a compliant element to simulate a compliant environment or `bob' of the polishing process, the dependency of force control dynamics on environment stiffness was eliminated and larger controller gains could thus be employed. This result is particularly useful since some form of compliant element invariably exists as a support for the abrasive medium used in a polishing process.

CHAPTER 11

RECOMMENDATIONS FOR FUTURE WORK

Recommendations for future work can perhaps be conveniently divided into two categories, short term and long term. That work which has immediate application to the continuation of this work, in particular polishing, can be classed as short term. Whereas the longer term research objectives, towards which the results of this work can make important contributions, are to improve the general understanding this subject of simultaneous force and position control.

11.1 Immediate Research Work

Most importantly the force sensor should be redesigned to remove its static friction component which occurs under tangential loading. Once this is achieved blade polishing can be attempted and investigated further. Currently a single axis compliant force sensor without the problematic bearing element is being developed by the author, initial results are promising.

Improved joint velocity signals are required. Since present velocity dependent damping operates in joint coordinates, it is manipulator configuration dependent. Transforming these measured joint velocity signals to the controller based coordinate system will improve the stability of both force and position control.

Introducing an analogue force rate sensor holds the greatest promise for improved force control. Both simulation and analyses indicate significant improvements in the degree of stability are possible by

implementing a suitable force rate signal. Also the instability caused by the presence of sensor damping effects can be removed by force rate damping. Such a force rate sensor is currently under development by the author, initial results show the device to function correctly. Practical implementation will be more difficult since the nature of any rate measuring device is that signal amplitude increases with frequency, therefore unwanted high frequency disturbances have greatest effect. Those frequencies required for robot force control are low, in general less than 30Hz, therefore careful signal filtering will be required to avoid aliasing problems.

If, after implementing both the new force rate signal and improved velocity signals, the force control stability is still not robust enough to cope with all variations in environment and manipulator dynamics, then a configuration dependent look-up table should be implemented. The elements of the table are selected as the force controller gains most suitable for a particular manipulator configuration. These look-up gains and their associated configuration envelope, should be determined empirically by trial and error.

Once the successful completion of the above work has been achieved, the next stage in this work should be towards polishing general 3 dimensional contours. Naturally this would require a prototype 6 axis manipulator capable of moving whilst applying force. A suitable manipulator could be based on the manipulator constructed within this research, but with 3 ballscrew drives arranged as a type of tripod and with three orientating axes at the apex. Alternatively, if funds allow, a large powerful industrial manipulator may be used and a special Hybrid force/position controller developed along the lines of that used in this work.

Eventually attempts should be made to extend the work to investigate techniques of holding the blade in the end effector and offering it to various abrading mediums. This technique may prove to be more economic in terms of blade polishing cycle times.

11.2 Long Term Research Work

The long term implications of this work, beyond polishing, are in general in applications where manipulators require a sense of `touch' or `feel'. This work will inevitably lead to active assembly, unachievable at present. An example of such a task could perhaps be, locating a nut on a thread. This is an extremely complex operation, with many touches and interactions before firstly the stud and subsequently the thread start are correctly located. Natural coordinates, as defined by Mason [32], will certainly require automatic selection during this procedure. Various methods may be used to distinguish between a contact and a false contact. False contacts, perhaps caused by frictional or inertial resistance effects at the end effector, could easily fool the controller. A possible approach to correctly distinguish contact may be achieved by using a combination of contact force and force rate signals. If, for example, contact is made while moving in a positional sense with a slow `seeking' velocity, there will be a rise in force accompanied by a constant force rate. Recognizing this condition would result in the correct detection of contact, and the consequential switching to correct transformation matrices within the Hybrid scheme.

A further interesting aspect of this work would be to develop a control strategy to allow the manipulator to track, under force control, unknown contours in much the same way as a blind person would. This possibly has no immediate practical application, apart from checking the geometry of some component, but it could eventually be used in automatic location

and hence assembly operations.

Sample delay effects on force control, caused by the numerically based control system, should receive extensive study. This should establish the precise bounds of stability in this respect, and establish the relationship between sample rate and the other parameters of the dynamic force controlled system.

The extreme long term objective of this and related research, is to achieve stable full dynamic force and position control. Problems currently exist when manipulators are driven `fast' whilst being controlled by the Hybrid scheme. Hybrid control does not cater for the dynamic effects caused by fast movements and thus there is coupling between the dynamics of fast movement and those of force control. This results in transient errors in both force and position control. This work is immensely complex as both the dynamics of the manipulator and environment (as has been shown by this work) need to be fully considered before true dynamic decoupling can be achieved.

REFERENCES

[1] Stute, G. and Erne, H. "The Control Design of an Industrial Robot with Advanced Tactile Sensitivity". 9th Int Symp on Industrial Robots, March 1979.

[2] Mortensen, Asbjorn. "Automatic Grinding", 13th Int Symp on Industrial Robots, Vol 1, Section 8, April 1983.

[3] Kramer, B.M. et al. "Robotic Deburring", Robotics and Computer Integrated Manufacture, Vol 1, No 3/4, 1984, pp 365-374.

[4] Spur, G. and Felsing, W. "Development of a Robot-Integrated Workingplace for the Brushing of Rubber/Metal Parts". 12th Int Symp on Industrial Robots, 1982.

[5] "Grinding", SERC Publication, UK, 1982.

[6] Stepien, T.M., Sweet, L.M., Good, M.G. "Control of Tool/Workpiece Contact Force with Application to Robot Deburring". Procs. IEEE Robotics and Automation, 1985, pp 670-679.

[7] Asada, H., Goldfine, N. "Optimal Compliance Design for Grinding Tool Holders". Procs. IEEE Robotics and Automation, 1985, pp 316-322.

[8] Graham, D., Woodwark J.R., "Automated Polishing of Complex Geomertry Components". Procs. 4th British Robot Association Annual Conf., Brighton, May 1980, pp 133-142.

[9] Johnson, C.D. "Accommodation of Exteral Diturbances in Linear Regulator and Servomechanism Problems", IEEE Trans. on Automatic Control, AC-16, No. 6, 1971, pp 635-644.

[10] Davidson, E.J. "The Output Control of Linear Time-invariant Multivariable Systems with Unmeasurable Arbitrary Disturbances", IEEE Trans. on Automatic Control, AC-17, No. 5, 1972, pp 621-629.

[11] Burdess, J.S., Metcalfe, A.V., "Active Control of Forced Vibration Produced by Arbitrary Disturbances", ASME Journal of Vibration, Acoustics, Stress and Reliability in Design, Vol. 107, No.1, 1976, pp 35-47.

[12] Hewit, J.R. and Burdess, J.S. "Fast Dynamic Decoupled Control for Robotics Using Active Force Control", Mechanism and Machine Theory, Vol 16, No5, 1981.

[13] Hewit, J.R., Tan, N. "Dynamic Co-ordination of Robot Movement ". 4th CIBM-IFToMM Symp. Theory and Practice of Robots and Maipulators, Warsaw, Sept. 1981.

[14] Hewit, J.R., Burdess, J.S. "An Active Method for the Control of Mechanical Systems in the Presence of Unmeasurable Forcing", Mechanism and Machine Theory, Vol. 21, No. 5, 1986, pp 393-400.

[15] Galatis, G., Hadzistylis A., Hewit, J.R. "Experimental Investigation of Active Force Control of Robot and Manipulator Arms". Proc. of 6th CIS-IFToMM Symp. ROMANSY-86. Cracow, Poland, Sept. 1986, pp 309-320.

[16] Paul R.P., Wu, C. " Resolved Motion Force Control of Robot Manipulator", IEEE Transactions on Systems, Man and Cybernetics, Vol. 12, No. 3, May/June 1982.

[17] Paul, F.W., Gettys. T.K., Thomas, J.D. "Definning of Iron Castings Using a Robot Positioned Chipper", Trans. ASME Robotics Research and Advanced Applications Proc, Nov 1982.

[18] Zalucky, A., Hardt, D.E. "Active Control of Robot Structure Deflections", ASME Procs. Robotics Research and Advanced Applications, Nov. 1982.

[19] West, H., Asada, H. A Method for the Design of Hybrid Position/Force controllers for Manipulators Constrained by Contact with the Environment", IEEE Procs. Robotics and Automation, 1985, pp 251-259.

[20] Stewart, D., "A Platform with Six Degrees of Freedom", Procs. Inst. of Mech. Engrs., Vol 180, Part 1, No 15, 1965, pp 371-378.

[21] Potton, S.L., "GEC Advanced Device for Assembly". 15th GRP Intl. Seminar on Manuf. Systems, Amherst, Mass. USA, June 1983.

[22] Hunt, K.H., "Structural Kinematics of In-Parallel-Actuated-Robot-Arms". Trans. ASME Journal of Mechanisms, Transmissions and Automation in Design, Vol. 105, Dec 1983, pp 705-712.

[23] Yang D.C.H., Lee, T.W., "Feasibility Study of a Platform Type of Robotic Manipulators from a Kinematic Viewpoint". Trans. ASME Journal of Mechanisms, Transmissions and Automation Design, Vol. 106, June 1984, pp 191-198.

[24] Fichter, E.F., "A Stewart Platform-Based Manipulator: General Theory and Practical Construction". Intl. Journal of Robotics Research, Vol. 5, No. 2, 1986, pp 157-182.

[25] Steven, A. Hewit, J.R., "Hybrid Position and Force Control Applied to Robotic Polishing of Turbine Blading". Proc 3rd Intl. Conf. on Advanced Robotics, Versailles, France, Oct. 1987, pp 493-501.

[26] Thorton, G.S., "The GEC Tetrabot - A New Serial-Parallel Assembley Robot". Proc. IEEE Intl. Conf. or Robotics and Automation, Philadelphia, Pen., USA, April 1988, pp 437-439.

[27] Rooke, M.W., Lewis, P.H., "Kinematics of a Triangular Structured Robot". IEEE Conf. on Robotics and Automation, San Francisco, California, USA, April 1986, pp 684-688.

[28] Rathbun, G.P., Dunlop, G.R., "Commensurate Positioning of a Stepmotor Actuated Stewart Platform". Procs. 7th World Congress, Theory of Machines and Mechanisms, Sevilla, Spain, Sept. 1987, pp 1481-1483.

[29] Paul, R.P., Shimano, B., "Compliance and Control". Procs. of Joint Automatic Control Conf., Lafayette, Indiana, USA, July 1976, pp 694-699.

[30] Whitney, D.E., "Force Feedback Control of Manipulator Fine Motions". Journal of Dynamic Systems, Measurement, and Control, June 1977, pp 91-97.

[31] Whitney, D.E., "Resolved Motion Rate Control of Manipulators and Human Prostheses." IEEE Trans. on Man-Machine Systems, Vol. 10, No.2, June 1969, pp 47-53.

[32] Mason, M.T., "Compliance and Force Control for Computer Controlled Manipulators". IEEE Trans. on Systems, Man and Cybernetics, Vol. 11, No.6, June 1981, pp 418-432.

[33] Mason, M.T., "Compliance and Force Control for Computer Controlled Manipulators". MIT Anteficial Intelligence Laboratory Memo 515, April 1979.

[34] Raibert, M.H., Craig, J.J., "Hybrid Position/Force Control of Manipulators". ASME Journal of Dynamic, Systems, Measurement and Control Vol. 102, June 1981, pp 126-133.

[35] Raibert, M.H., Craig, J.J., "Hybrid Position/Force Control of Manipulators." ASME Winter Meeting, Chicargo, IL, USA, Nov. 1980.

[36] Benati, M., et al, "Anthropomorphic Robotics, Part I". Biol. Cybernetics, Vol. 38, 1980, pp 132-140.

[37] Benati, M., et al, "Anthropomorphic Robotics, Part II". Biol. Cybernetics, Vol. 38, 1980, pp 142-150.

[38] Salisbury, J.K., "Active Stiffness Control of a Manipulator in Cartesian Coordinates". 19th IEEE Conf. on Decision and Control, Albuquerque, N.M., Dec 1980.

[39] Asada, H., Slotine, J.J.E, "Robot Analysis and Control", John Wiley and Sons, pp 185-216.

[40] Whitney, D.E., "Historical Perspective and State of the Art" in Robot Force Control". IEEE Conf. on Robotics and Automation, St. Louis, Missouri, USA, March 1985, pp 262-268.

[41] Maples, J.A., Becker J.J., "Experiments in Force Control of Robotic Manipulators." IEEE Conf. on Robotics and Automation, Philadelphia, Pen., USA, April 1986, pp 695-702.

[42] Takeyasu, K., Goto, T., Inoyama, T., "Precision Insertion Control and its Application". ASME Journal of Engineering for Industry, Nov. 1976, pp 1313-1318.

[43] Drake, S.H., Watson, D.C., Simunovic S.H., "High Speed Robot Assembly of Precision Parts Used in Compliance Instead of Sensory Feedback". Proc. 7th Intl. Symp. on Industrial Robots, Tokyo, Japan, 1977, pp 87-97.

[44] Watson, P.C., "Remote Centre Compliance System". U.S. patent No. 4,098,001, 4th July 1978.

[45] Whitney, D.E., Nevins, J.L., "What is the Remote Centred Compliance (RCC) and What Can It Do?". Internal Report P-728, The Charles Stark Draper Lab., Cambridge, MA, USA, Nov. 1978.

[46] Whitney, D.E., "Quasi-Static Assembly of Compliantly Supported Rigid Parts". ASME Journal of Dynamic Systems, Measurement and Control, Vol. 104, March 1982.

[47] McCallion, H., Alexander, K.V., Pham, D.T., "Aids for Automatic Assembly". 1st Intl. Conf. on Assembly Automation, Brighton, UK, March 1980.

[48] Van Brussel, H., Simons, J., "The Adaptable Compliance Concept and Its Use for Automatic Assembly by Active Force Feedback Accommodations". 9th Intl. Symp. on Industrial Robots, Washington, D.C., USA, March 1979, pp 167-181.

[49] Cutkosky, M., Wright, P.K., "External Position Control of Industrial Manipulators". ASME Computers in Engineering, San Diego, CA, USA, 1982, pp 113-118.

[50] Hollis, R.L., "A Planar XY Robotic Fine Positioning Device". IEEE Conf. on Robotics and Automation, St. Louis, Missouri, USA, March 1985, pp 329-336.

[51] Zang, H., Paul, R.P., "Hybrid Control of Robot Manipulators". IEEE Conf. on Robotics and Automation, St. Louis, Missouri, USA, March 1985, pp 602-607.

[52] Townsend, W.T., Salisbury, J.K., "The Effect of Coulomb Friction and Stiction on Force Control". IEEE Conf. on Robotics and Automation, Raleigh, North Carolina, USA, April 1987, pp 883-889.

[53] Hogan, N., "Impedance Control of Industrial Robots". Robotics and Computer Integrated Manufacture, Vol. 1, No.1, 1984, pp 97-113.

[54] Kazerooni, H., Houpt, P.K., Sheridan, T.B., "Robust Compliant Motion for Manipulators, Parts I and II". IEEE Journal of Robotics and Automation, Vol. 2, No. 2, June 1986, pp 83-105.

[55] Tsai, C-K., Orin, D.E., "Modified Hybrid Control for an Electro-Hydraulic Robot Leg". ASME Conf. Computers in Engineering, Chicago, IL, USA, July 1986.

[56] Hogan, N., "Impedance Control: An Approach to Manipulation, Parts I, II, III". Journal Dynamics Systems Measurement and Control, Vol 107, March 1985, pp 1-23.

[57] Inoue, H., "Force Feedback in Precise Assembly Tasks". Technical Report AIM-308, Artificial Intelligence Laboratory, MIT, Cambridge, MA, USA, Aug. 1974.

[58] Featherstone, R., "Calculation of Robot Joint Rates and Actuator Torques from End Effector Velocities and Applied Forces". Mechanism and Machine Theory, Vol. 18, No. 3, 1983, pp 193-198.

[59] Paul, R.P., "Problems and Research Issues Associated with the Hybrid Control of Force and Displacement". IEEE Conf. on Robotics and Automation, Raleigh, North Carolina, USA, April 1987, pp 1966-1971.

[60] Uchiyama, M., Iwasawa, N., Hakomori, K., "Hybrid Position/Force Control of a Two Arm Robot". IEEE Conf. on Robotics and Automation, Raleigh, North Carolina, USA, April 1987, pp 1242-1247.

[61] Luh, J.Y.S., Fisher, W.D., Paul, R.P.C.," Joint Torque Control by a Direct Feedback of Industrial Robots". IEEE Trans. on Automatic Control, Vol. 28, No. 2, Feb 1983.

[62] Roberts, R.K., Paul, R.P., Hillberry, B.M.," The Effect of Wrist Force Sensor Stiffness on the Control of Robot Manipulators". IEEE Intl. Conf. or Robotics and Automation, St. Louis, Missouri, USA, March 1985, pp 269-274.

[63] Fukuda, T., Kitamura, N., "Adaptive Force Control of Grippers with Consideration of Dynamics of Objects". RoManSy86, CraKow, Poland, Sept. 1986, pp 235-249.

[64] Khatib, O., Burdick, J., "Force Control of Robot Manipulators". 7th World Congress, Sevilla, Spain, Sept 1987, pp 1213-1218.

[65] Seering, W.P., Eppinger, S.D. "The Dynamics of Robot Force Control". 7th World Congress, Sevilla, Spain, Sept 1987, pp 1223-1226.

[66] Chae, A.H., Hollerbach, J.M., "Kinematic Stability Issues in Force Control of Manipulators". IEEE Intl. Conf. on Robotics and Automation, Raleigh, North Carolina, USA, April 1987, pp 897-903.

[67] Chae, A.H., Hollerbach, J.M., "Dynamic Stability Issues in Force Control of Manipulators". IEEE Intl. Conf. or Robotics and Automation, Raleigh, North Carolina, USA, April 1987, pp 890-896.

[68] Barnett, S., Storey, C., "Matrix Methods in Stability". Thomas Nelson and Sons Ltd., pp 8-13.

[69] Thompson, W.T.,"Vibration Theory and Applications". George Allen and Unwin Ltd., 1978, pp 68-75.

[70] Asada, H., Kanade, T., Takeyama, I., "Control of a Direct Drive Arm". ASME Trans. on Journal of Dynamic Systems, Measurement and Control, Vol. 105, Sept. 1983, pp 136-143.

[71] Wright, S., Wright, S., Geake, V., "Symbol for MS-DOS Personal Computers - User Guide". Cambridge Control Ltd., Cambridge, Version 1.2, March 1987.

[72] Texas Instruments, "The TTL Data Book for Design Engineers". Vol. 1, 1984.

APPENDIX A

THE ROUTH HURWITZ STABILITY CRITERION

OF SINGLE DEGREE OF FREEDOM FORCE CONTROL

The Routh-Hurwitz stability criterion for a characteristic equation expressed in polynomial form:

$$a_0S^n + a_1S^{n-1} + a_2S^{n-2} + \ldots + a_{n-1}S + a_n = 0$$

Providing the coefficients a_0 to a_n are none zero and of the same sign, then the stability array may be expressed as:

	1	2	3	4	5	6		
S^n	a_0	a_2	a_4	a_6	..	..	..	..
S^{n-1}	a_1	a_3	a_5	a_7	..	..	..	
S^{n-2}	b_1	b_2	b_3	b_5	..	..		
S^{n-3}	c_1	c_2	c_3	..	..			
S^{n-4}	d_1	d_2	..	..				
.	..	..	..					
.	..	..						
.								
S^0								

where the array coefficients are defined as:

$$b_1 = \frac{a_2a_1-a_0a_3}{a_1}, \quad b_2 = \frac{a_4a_1-a_0a_5}{a_1}, \quad b_3 = \frac{a_6a_1-a_0a_7}{a_1}, \ldots\ldots$$

$$c_1 = \frac{a_3b_1-a_1b_2}{b_1}, \quad c_2 = \frac{a_5b_1-a_1b_3}{b_1}, \ldots\ldots$$

$$d_1 = \frac{b_2c_1-b_1c_2}{c_1}, \ \ldots\text{etc.}$$

For every change in sign in the first column of coefficients there is an

associated unstable root.

Using this definition for the Routh-Hurwitz criterion, two equations from chapter 8 were investigated.

Effects of Sensor and Transmission Damping

The Routh-Hurwitz stability criterion is used to investigate the effects of the viscous damping, due to the sensor and transmissions, on force control stability. The single degree of freedom characteristic equation 8.6, expressed here as equation A.1, is used for the investigation. However, the rate damping term K_D is set to zero to limit complexity.

$$M_1M_2S^4 + [M_1C_S+M_2(C_R+C_S)]S^3 + [M_1(K_E+K_S) + M_2(NK_CK_P+1)K_S + C_RC_S]S^2 + [C_R(K_E+K_S) + K_EC_S]S + K_EK_S(1+NK_CK_P) = 0 \qquad (A.1)$$

The Routh-Hurwitz analysis performed on equation A.1 gives the b_1 coefficient from the first column of the array as:

$$b_1 = \left[M_1^2(K_E+K_S) + M_1M_2(NK_CK_P+2)K_SC_S + M_2^2(NK_CK_P+1)K_S(C_R+C_S) + (M_1+M_2)C_S + M_2C_R^2C_S \right] \div \{(M_1+M_2)C_S+M_2C_R\}$$

For normal gains and mechanical dynamics this coefficient remains positive always and so does not indicate unstable roots. The next coefficient in the array's first column is c_1, which is given as:

$$c_1 = \Big[\{M_1^2C_SK_E(C_R+C_S)(K_E+K_S) + M_1M_2K_EK_SC_S(C_R+C_S)(NK_CK_P+2)$$
$$+ M_2^2K_EK_S(NK_CK_P+1)(C_S+C_R)^2 + (M_1+M_2)K_EC_RC_S(C_R+C_S)^2$$
$$+ M_1^2C_RC_SK_S(K_E+K_S) + M_1M_2C_RC_SK_S^2(NK_CK_P+2)$$
$$+ M_2^2C_RK_S^2(NK_CK_P+1)(C_S+C_R) + (M_1+M_2)C_R^2C_SK_S(C_S+C_R)\}$$
$$- \{K_SK_E(1+NK_CK_P)\{(M_1+M2)M_1C_S^2 - (M_1+M_2)M_2C_S(C_SC_R)$$
$$+ M_2M_1C_RC_S + M_2^2C_R(C_S+C_R)\}\Big] + b_1$$

At this stage the evaluation of coefficient c_1 is indeterminate. Therefore c_1 is simplified by considering the the effects of C_S and C_R separately.

C_R Damping Alone

Setting the C_S terms to zero reduces the c_1 coefficient to:

$$c_1 = \Big[M_2^2K_EK_S(NK_CK_P+1) + M_2^2C_R^2K_S^2(NK_CK_P+1) - M_2^2K_EK_S(NK_CK_P+1)\Big]$$
$$+ b_1$$

This further reduces, giving the condition for instability as a sign change of the term $M_2C_R^2K_S^2$, which will never occur in practice.

C_S Damping Alone

Setting all the C_R terms to zero reduces the c_1 to:

$$c_1 = \Big[M_1^2C_S^2K_E(K_E+K_S) + M_1M_2K_EK_SC_S^2 - K_SK_E(NK_CK_P+1)(M_1+M_2)M_1C_S^2\Big]$$
$$+ b_1$$

Which gives the conditional stability as a sign change in the term $[M_1K_E - NK_CK_PK_S(M_1+M_2)]$, written as:

$$\frac{M_1K_E}{(M_1+M_2)} = NK_CK_PK_S$$

If, as in this case, the manipulator mass, $M_1 > M_2$, the sensor/environment mass then, the ratio of environment to sensor stiffness is given by:

$$\frac{K_E}{K_S} = NK_CK_PK_S$$

APPENDIX B

TRANSMISSION COMPONENTS

B1 Printed Armature DC. Servomotor

B2 Electro-Craft DC. Servomotor

B3 Harmonic Drive, HDUF 25

Printed Armature DC Servo Motor

G12M4

MOTOR RATINGS

Continuous torque at rated speed	55 oz in
Pulse torque (50 ms at 1% duty cycle)	1200 oz in
Rated speed	3650 rev/min
Rated voltage[1]	48 Vdc
Power output at rated speed	147 W
Rated current	4.4 A
Maximum continuous stall current	7.5 A
Terminal resistance	0.75Ω

MOTOR CONSTANTS

Torque constant (K_t)	15.6 oz in/A
Emf constant (K_e)	11.5 V/1000 rev/min
Damping constant (K_d)	3.1 oz in/1000 rev/min
Total inertia (J_m)	0.020 oz in s^2
Regulation at constant voltage (R_m)[2]	5.85 rev/min/oz in
Armature inductance (L_a)	<100 μH
Average friction torque (T_f)	4.0 oz in
Mechanical time constant[2]	0.0126 s
Power rate[3]	507 kW/s

THERMAL RESISTANCE

Uncooled	
Armature-to-case ($\theta_{a\text{-}c}$)	1.15 deg C/W
Case-to-ambient ($\theta_{c\text{-}a}$)	
With 8 × 16 × ⅜ in aluminium heat sink	0.87 deg C/W
With 14 × 14 × ⅜ in aluminium heat sink	0.70 deg C/W
Forced cooling	
Armature-to-ambient (θ_t)	
With mass air flow of 0.4 lb/min	0.8 deg C/W
With mass air flow of 0.8 lb/min	0.51 deg C/W
With mass air flow of 2.0 lb/min	0.28 deg C/W

The figures quoted above for the motor constants are typical and cannot be guaranteed unless a technical specification has been negotiated.

WEIGHT

8 lb (3.63 kg)

NOTES

1. Motor is tested at this voltage for convenience. Other voltages may be used provided maximum armature dissipation is not exceeded. ($P_{max} = P_{in} - P_{out} = \text{constant}$).
2. The speed-torque curve is obtained by using the maximum terminal resistance of the motor at 150 °C armature temperature (worst condition).
3. Calculated from the formula:

$$7.01 \times 10^{-6} \times \frac{(\text{Pulse torque})^2}{\text{Inertia}}$$

GENERAL

1. Maximum allowable armature dissipation:

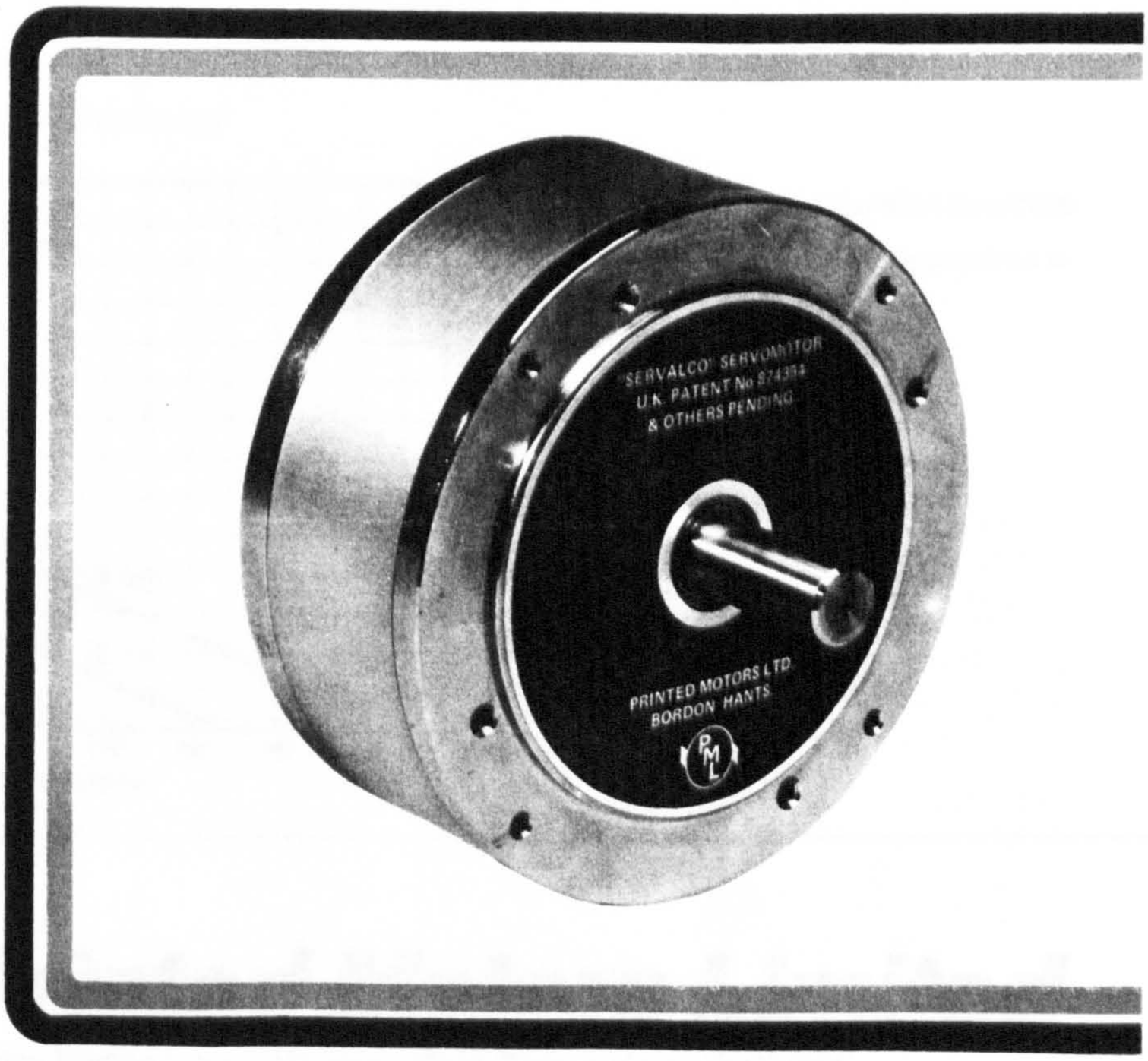
$$P_{max} = \frac{150\ ^\circ\text{C} - T_{ambient}\ (^\circ\text{C})}{\theta_{a\text{-}c} + \theta_{c\text{-}a}}$$

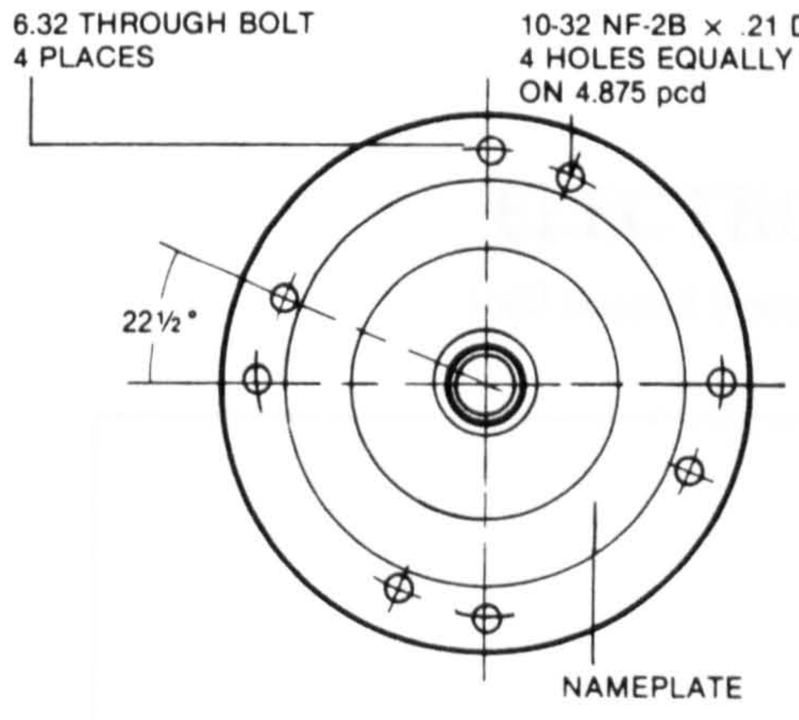
2. The curves for forced cooling operation were obtained by modifying the mechanical configuration of the motor to accept the required air flow. These motors are available on special request.
   The maximum allowable armature dissipation in this case is calculated as follows:

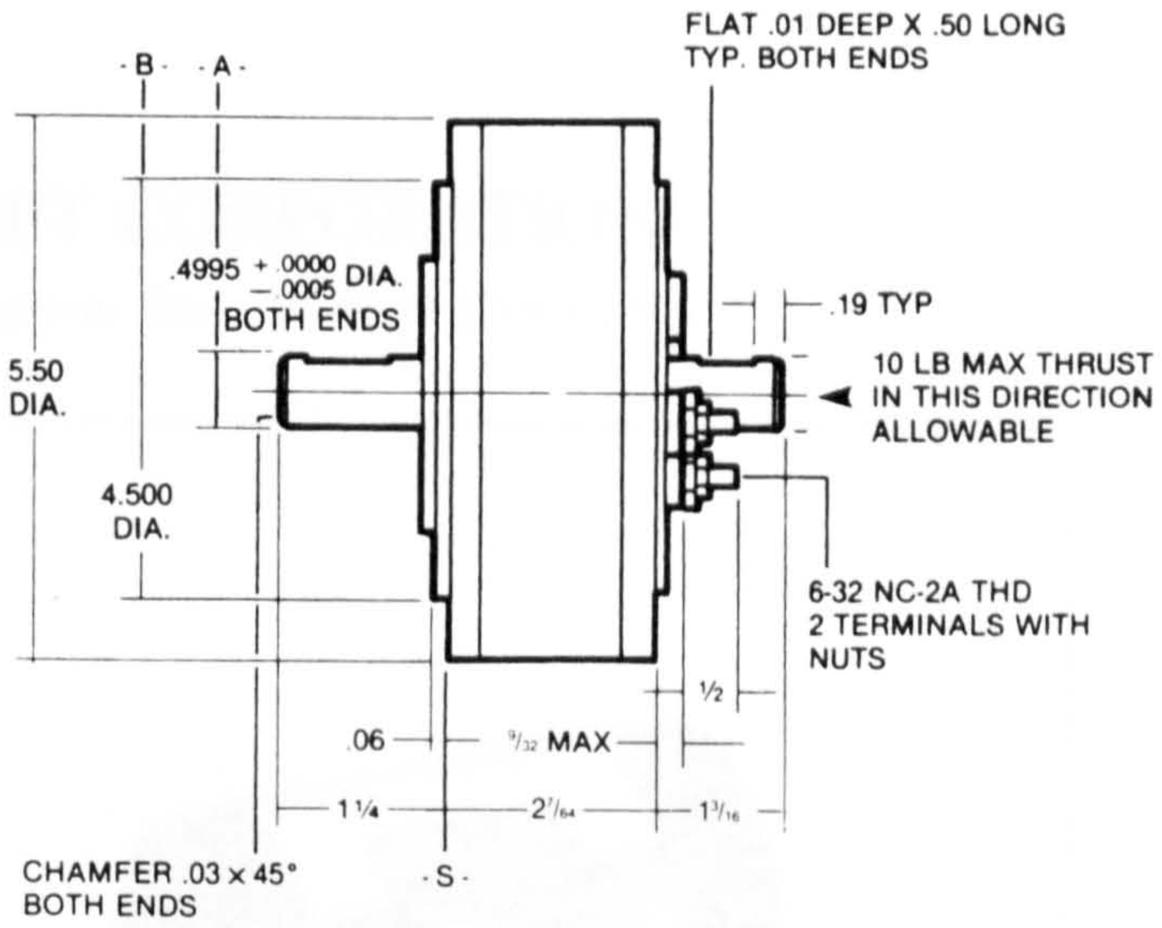
$$P_{max} = \frac{150\ ^\circ\text{C} - T_{ambient}\ (^\circ\text{C})}{\theta_t}$$

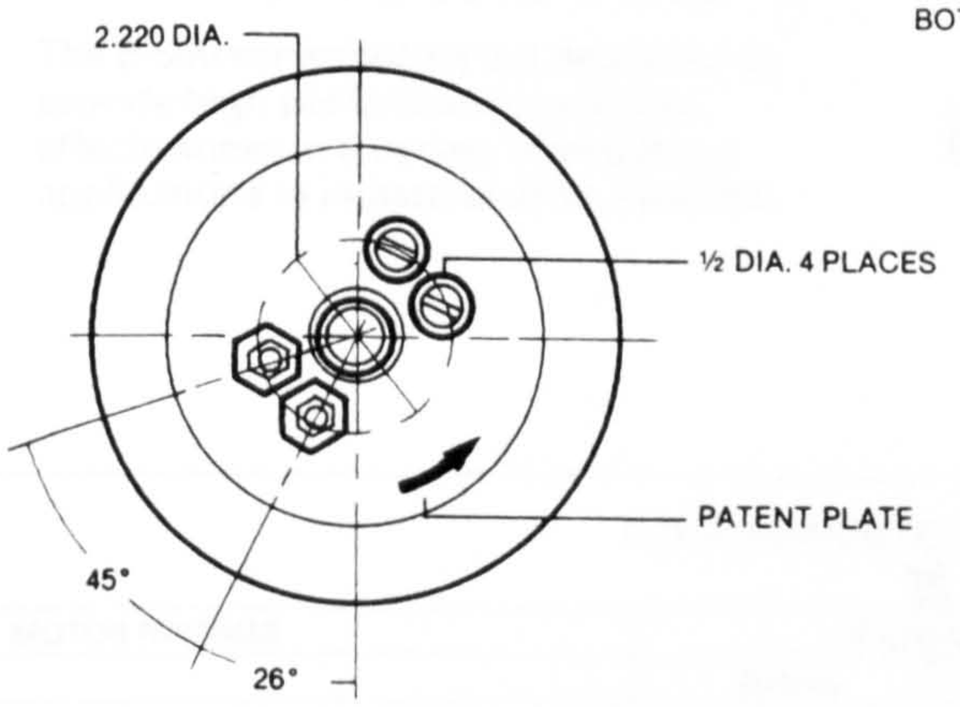
3. Mass air flow (lb/min) = Air volume (ft^3/min) × density (lb/ft^3)

All nominal values at 25 °C ambient except where otherwise stated.









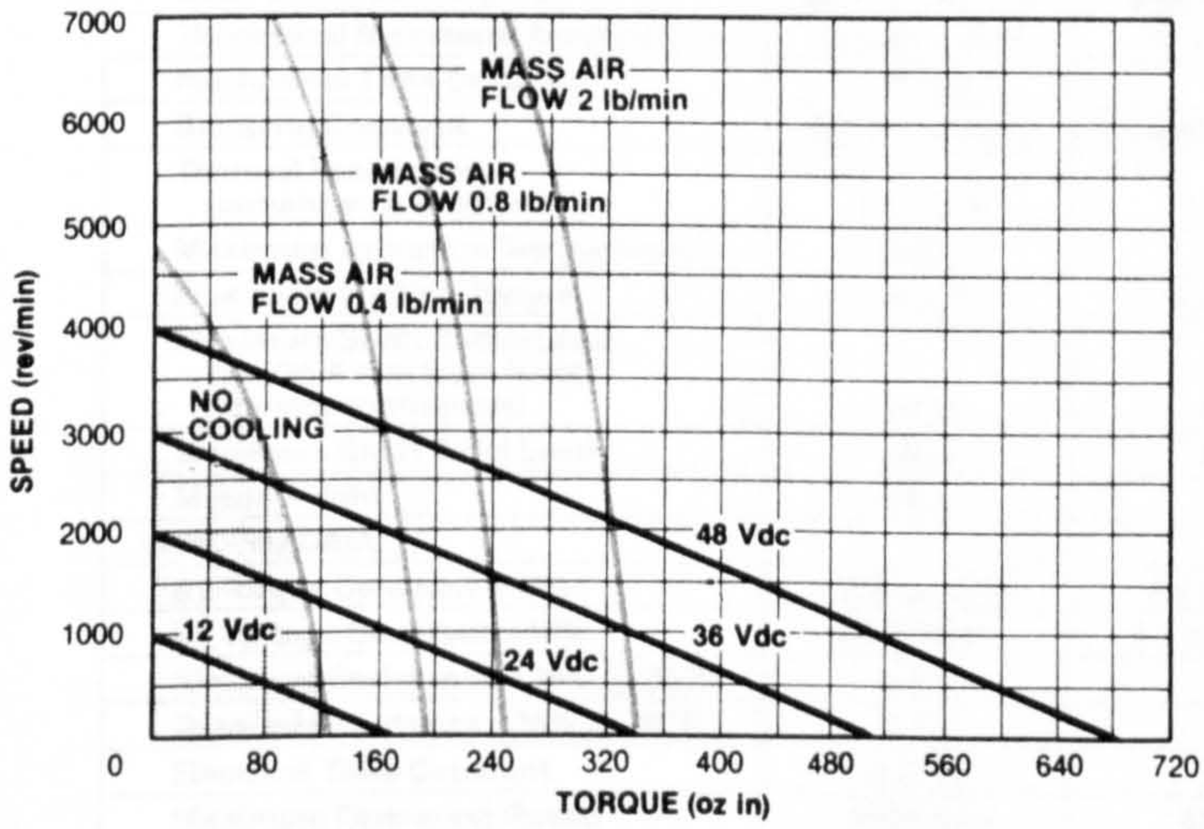
Dimensions must not be worked to unless certified.

Dimensions in inches

G12 M4 MECHANICAL SPECIFICATIONS

1. Shaft diameter 'A' runout not to exceed 0.001 in per inch.
2. Pilot diameter 'B' concentric to 'A' within 0.003 in. TIR.
3. Mounting surface 'S' perpendicular to 'A' within 0.007 in.
4. Shaft end play 0.004 in maximum under a reversal of 5 pounds thrust.
5. Maximum allowable radial load of 30 pounds at rated speed.

AVERAGE PERFORMANCE CHARACTERISTICS



☐ Limit of continuous operation (uncooled)

The run current at any operating condition is obtained as follows:

$$I_{RUN} = \frac{K_d \times \frac{N}{1000} + T_f + T_l}{K_t}$$

Printed Motors Limited

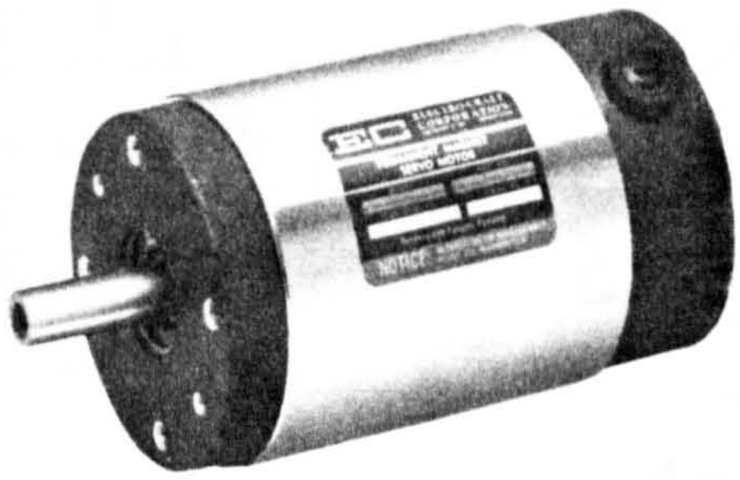
Bordon Trading Estate, Oakhanger Road, Bordon, Hants. GU35 9HY.
(Registered Office and Works)
Telephone: Bordon (04203) 3033 (4 lines) Telex: 858768
A MEMBER OF THE Technograph GROUP

ELECTRO-CRAFT CORPORATION

1600 Second Street South, Hopkins, Minn. 55343 • (612) 931-2700

E-660
DC SERVOMOTOR

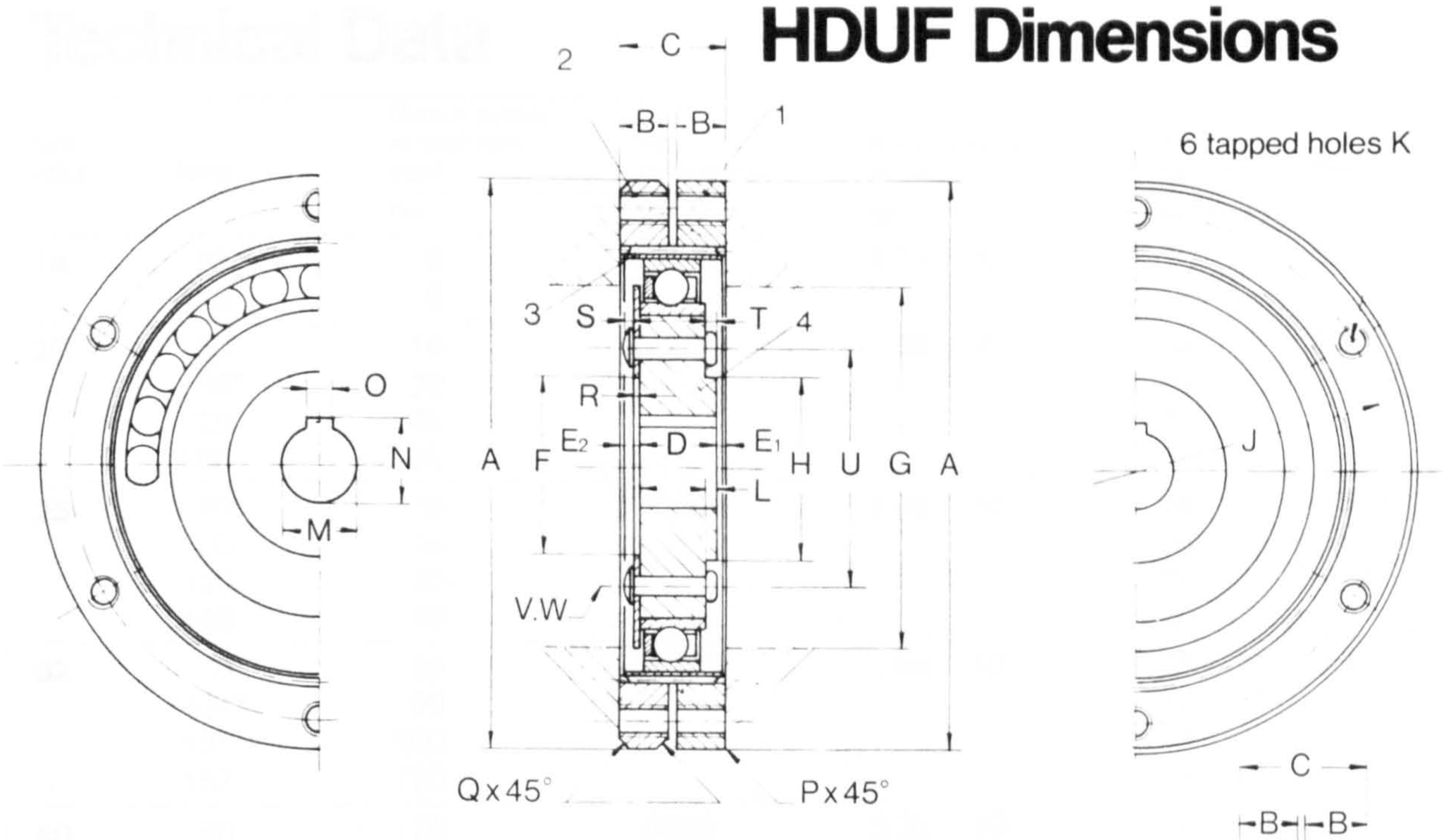
The E-660 servomotors are designed to provide high performance and cost-effectiveness in a variety of precision applications in industrial drive systems.

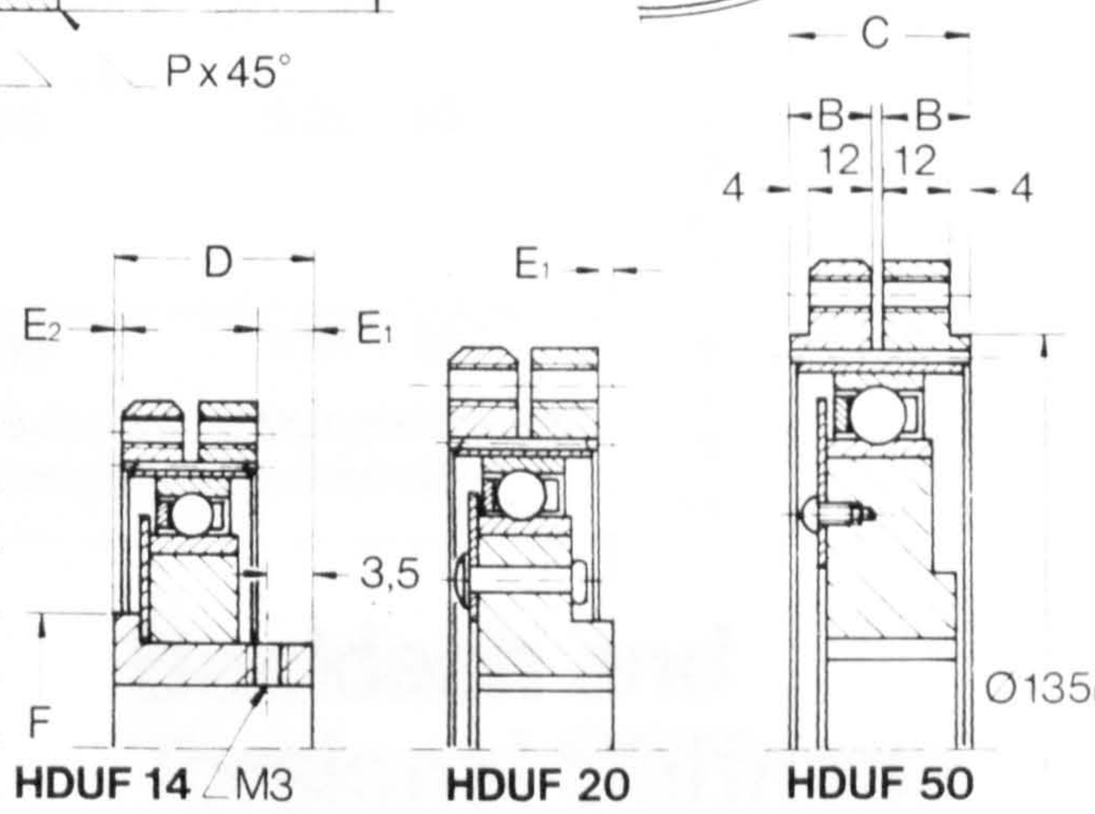


PERFORMANCE CHARACTERISTICS

MOTOR RATINGS	E-660-SA		E-660-SB	
	British	SI (Metric)	British	SI (Metric)
Continuous Output Power	240 w	240 w	240 w	240 w
Continuous Stall Torque	100 oz-in	0.7 Nm	120 oz-in	0.85 Nm
Peak Torque	385 oz-in	2.7 Nm	500 oz-in	3.5 Nm
Maximum Terminal Voltage	80 V	80 V	80 V	80 V
Stall Current	6.02 A	6.02 A	3.7 A	3.7 A
MECHANICAL DATA				
Rotor Moment of Inertia	3.2×10^{-2} oz-in-s^2	2.26×10^{-4} kg m^2	3.2×10^{-2} oz-in-s^2	2.26×10^{-4} kg m^2
Theoretical Maximum Acceleration	12×10^3 rad/s^2	12×10^3 rad/s^2	15×10^3 rad/s^2	15×10^3 rad/s^2
Mechanical Time Constant	22 ms	22 ms	10.5 ms	10.5 ms
Damping Constant	0.7 oz-in/krpm	4.7×10^{-5} Nm/rad•s^{-1}	0.7 oz-in/krpm	4.7×10^{-5} Nm/rad•s^{-1}
Thermal Resistance (armature-to-ambient)	2.8 °C/W	2.8 °C/W	2.8 °C/W	2.8 °C/W
Maximum Armature Temperature	155 °C	155 °C	155 °C	155 °C
Maximum Friction Torque	6 oz-in	4×10^{-2} Nm	6 oz-in	4×10^{-2} Nm
Maximum Shaft Radial Load (1 in/25.4 mm from front bearing, continuous)	30 lb	13.6 kg	30 lb	13.6 kg
Maximum Shaft Axial Load	10 lb	4.54 kg	10 lb	4.54 kg
Motor Weight	6.6 lb	3 kg	6.6 lb	3 kg
WINDING DATA				
K_T Torque Constant ±10%	12.8 oz-in/A	9×10^{-2} Nm/A	27 oz-in/A	0.19 Nm/A
K_E Voltage Constant ±10%	9.5 V/krpm	9×10^{-2} V/rad•s^1	20 V/krpm	0.19 V/rad•s^1
Armature Resistance ±15% @ 25 °C	0.3 Ω	0.3 Ω	1.34 Ω	1.34 Ω
Terminal Resistance ±15% @ 25 °C	0.7 Ω	0.7 Ω	1.7 Ω	1.7 Ω
Electrical Time Constant	3.7 ms	3.7 ms	3.7 ms	3.7 ms
Maximum Operating Speed	4000 rpm	420 rad/s	3400 rpm	357 rad/s
Maximum No-Load Speed	4000 rpm	420 rad/s	4000 rpm	420 rad/s
Maximum Pulse Current (before demagnetization)	53 A	53 A	25 A	25 A
Armature Inductance	1.2 mH	1.2 mH	5.4 mH	5.4 mH

HDUF Dimensions





Part	Quantity	Description
1	1	Circular Spline (CS)
2	1	Dynamic Spline (DS)
3	1	Flexspline (FS)
4	1	Wave Generator (WG)

Dimensions in mm

HDUF	øA g7	B	C	D	E1	E2	øF	øG	øH	øJ	K	L	øM H7	N	O JS7	P	Q	R	S	T	U	V	W
14	50	5	10.5	15.0	3.75	0.75	18	29	14	44	M3	–	6	–	–	0.2	1.0	–	–	–	–	–	–
20	70	6	12.5	11.4	0.95	2.05	20	42	20	60	M4	3	9	10.4 +0.1 -0.0	3	0.2	1.0	1	1.8	2	27	2	o 3
25	85	8	16.5	12.8	0.35	3.35	24	53	25	75	M5	3	11	12.8 +0.1 -0.0	4	0.2	1.5	1	1.8	2	34	2	o 3
32	110	10	20.5	15.6	0.95	3.95	34	69	36	100	M6	3	14	16.3 +0.1 -0.0	5	0.2	1.5	1	1.8	2	46	2	o 3
40	135	13	27.0	19.4	1.80	5.80	40	84	42	120	M8	4	14	16.3 +0.1 -0.0	5	0.4	2.0	1	2.6	–	56	2	M4
50	170	16	33.0	23.2	2.90	6.90	50	105	54	150	M10	4	19	21.8 +0.1 -0.0	6	0.4	2.0	1	2.6	–	72	3	M4

Installation Instructions

The Dynamic Spline is distinguished by chamfered edges.

HDUF gear sets can be installed in any attitude. To ensure trouble free service the input and output shafts must be properly supported in suitable bearings, and the following additional precautions should be taken.

1. The Circular Spline (1) and Dynamic Spline (2), must be held concentric with a maximum run out of 0,03 mm. Any deviation of the centres of the Dynamic Spline and Circular Spline rings must not exceed 0,015 mm from the centre line of the gearset. In addition Circular Spline and Dynamic Spline must be normal to the drive axis. Maximum deviation of 0,1 mm in 100 mm is permissible.
2. The Flexspline must abutt a surface of at least HB 260 (HRC 26.4) hardness to stop it working towards the Circular Spline side or the Dynamic Spline side. The Dynamic Spline is designated by chamfered edges (Q).
3. G = maximum housing I.D.

Technical Data

Size HDUF	Ratio	Output torque at 1450 rpm input	Max. input speed	Wave Generator inertia (J*)	Max. backlash	Weight
	i	Nm	rpm	kgm²	in minutes	kg
14	88•	8	3600	$3{,}30 \cdot 10^{-6}$	36	0,1
	110	8			36	
20	80	16	3600	$1{,}38 \cdot 10^{-5}$	33	0,3
	96•	20			33	
	128	25			33	
	160	30			33	
25	80	30	3600	$4{,}45 \cdot 10^{-5}$	28	0,5
	100	35			28	
	120	40			28	
	160	45			28	
32	78	65	3600	$1{,}68 \cdot 10^{-4}$	23	1,1
	104•	90			23	
	131	100			23	
	157	120			23	
40	80	120	3600	$5{,}05 \cdot 10^{-4}$	23	2,0
	96•	140			23	
	128	160			23	
	160	180			23	
50	80	200	3600	$1{,}51 \cdot 10^{-3}$	16	3.3
	100	260			16	
	120	300			16	
	160	340			16	

$^{*}J = \frac{GD^2}{4}$

•) Ratios will be changed in i = 100. Consult before ordering.

Overloads

Short-period overloads up to 100% of the rated load at 1450 rpm have little effect on service life provided they last only seconds and are not repeated more than four or five times per hour. If more severe or continuous overload conditions are anticipated – particularly rapid starts and stops, then a drive must be selected capable of handling the peak torque.

HDUF drives are limited in life by the use of special tooth forms to give the necessary difference in tooth numbers between the Circular Spline and Dynamic Spline which are the same diameter. This means that HDUF units are not interchangeable with HDUC types.

HDUF ratings are based on a minimum design life of 3000 hours at 1450 rpm input.

Backlash and Torsional Stiffness

Lost motion is measured under similar conditions to that shown for HDUC units on page 1.07, and may be reduced to three minutes of arc for special applications. Such drives are designated with the suffix BL3, and are not normally held in stock. It must be noted that they are made up as matched sets and parts may not be interchanged without loss of performance even among BL3 types.

HDUF torsional stiffness data is difficult to determine as it is very much dependent on overall tolerance. As a general rule the best units will approach the stiffness of HDUC types (page 1.07), and the worst approximately 50% of those values.

Service Factors

Service factors are given in the HDUC section on page 1.03.

APPENDIX C

INSTRUMENTATION

C1 Rack Power Supplies - 64 Pin Edge Connector.

C2 Power Board with Voltage Regulator.

C3 LVDT. and Conditioning Amplifier Specification.

C4 Facia Plug Connection Details.

C5 Optical Encoder Specifications.

C6 Encoder Counter Interface Board and the SN74LS2000 Specification.

C7 Servo Amplifier Specification.

C8 DAC.'s Analogue Interface Buffer Board.

C9 Computer Interface Cards.

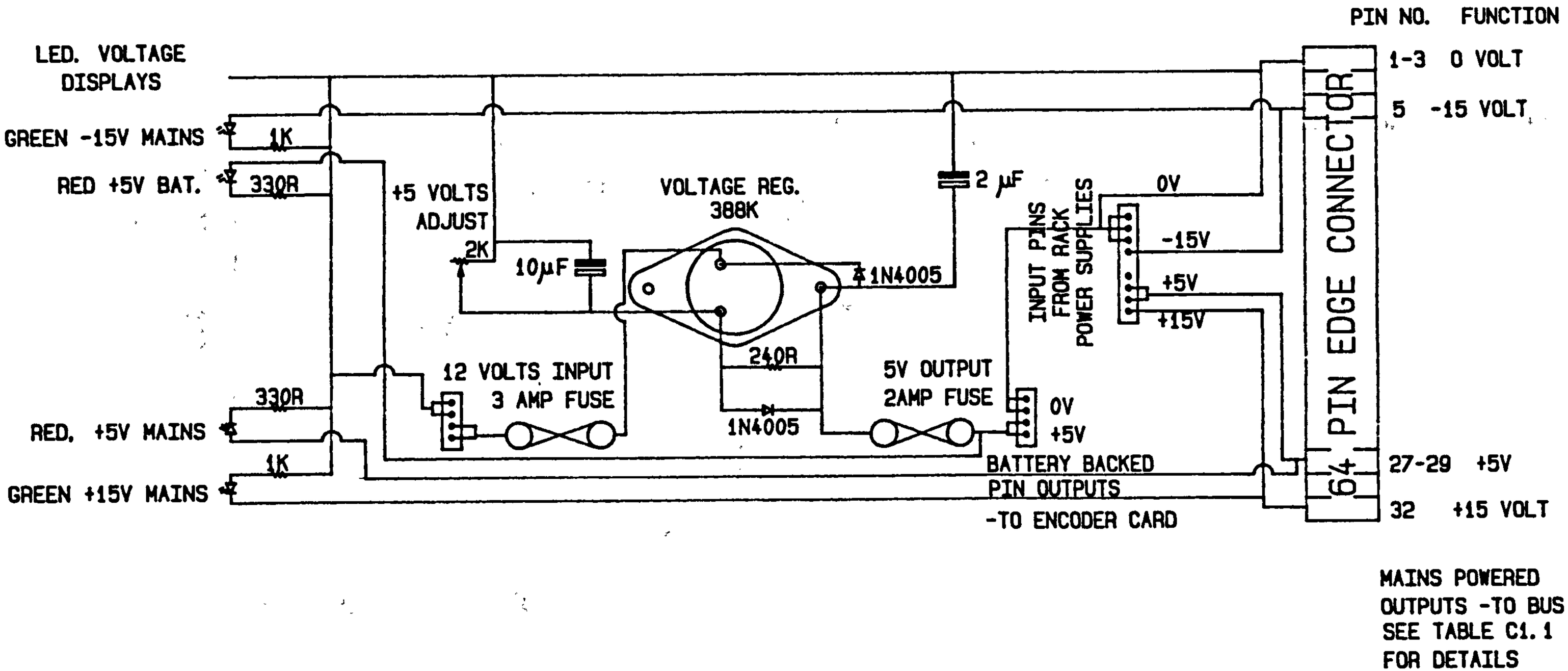
C1.1 Rack Power Supplies - 64 Pin Edge Connector Details

Two Bulgin power supplies fit into, and supply power to, the Eurocard rack assembly.

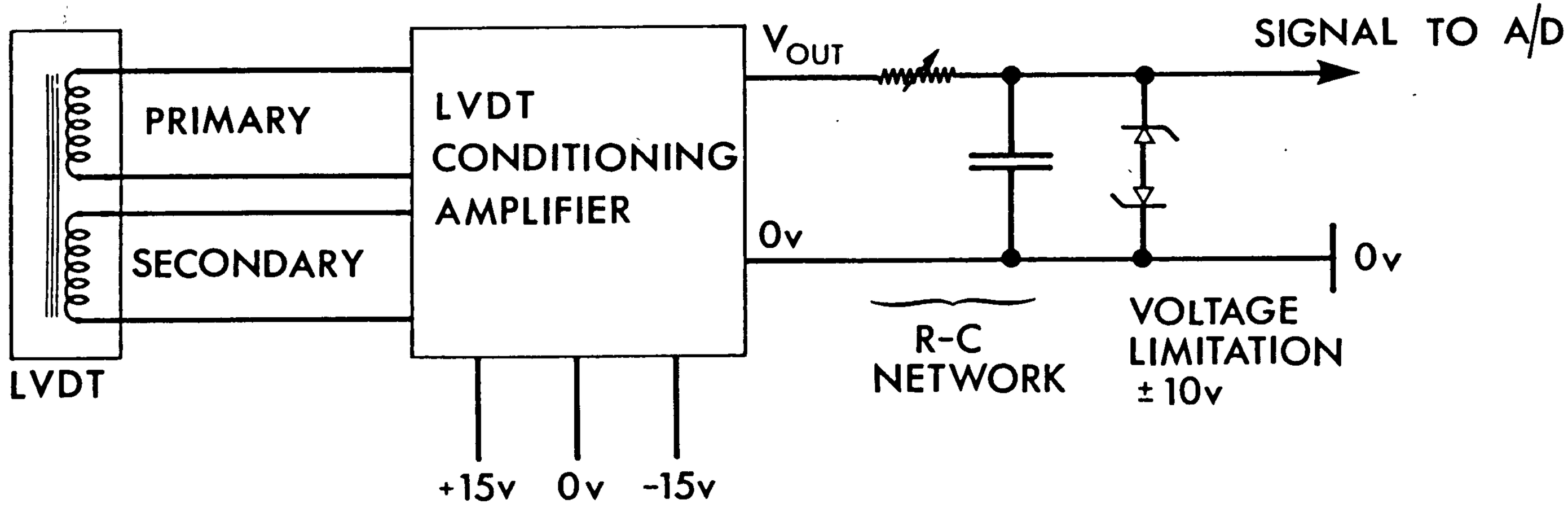
Rack power pin connections are as shown in table C1.1

Table C1.1

Edge connector pin Number	Function
1 to 3, A & B	0 Volts, common
7, A & B	Earth
5, A & B	-15 Volts
24, A & B	Earth
27 to 29, A & B	+5 Volts
30, A & B	0 Volts
32, A & B	+15 Volts



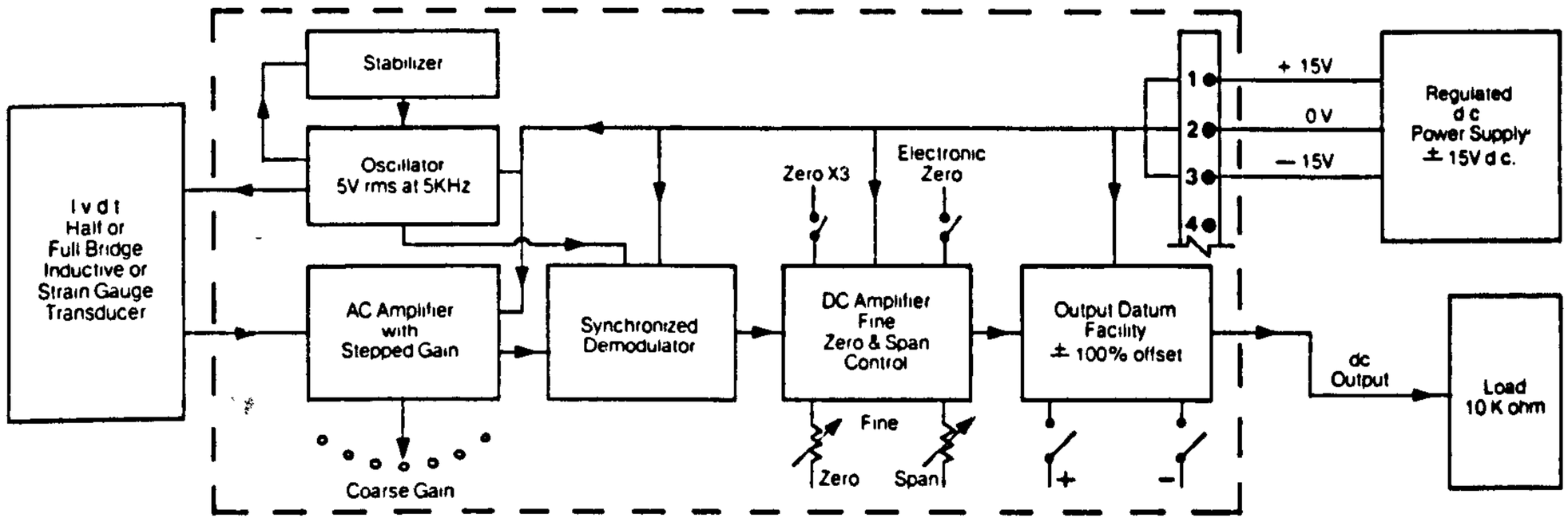
POWER BOARD WITH VOLTAGE REGULATOR
FIGURE C2.1



FORCE SENSOR LVDT WIRING ARRANGEMENT

FIGURE C3.1

Block diagram of CA Series card

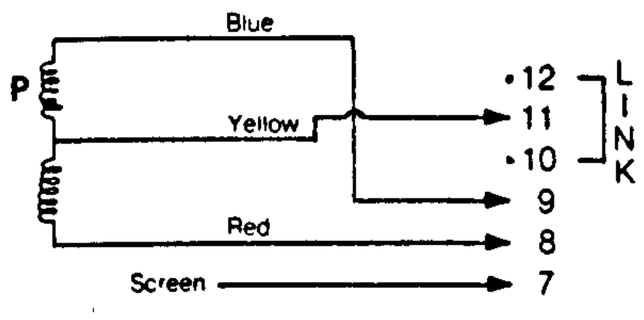


Mechanical

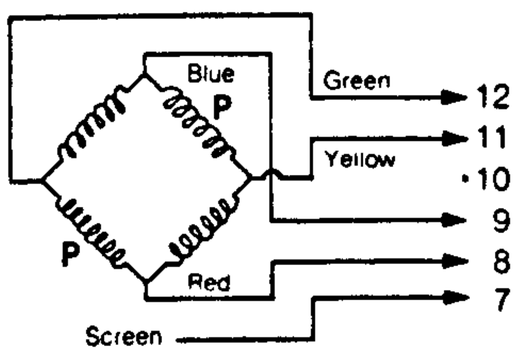
Type	
CA1	Avdel lugs (solder) for connection
X *CA2* X	Screw-down terminals (Klippon Mk8 series)
CA3	Pye Modulo 100 edge connector
CA4	DIN41612 edge connector
Dimensions	100mm wide 153mm long 18mm highest component

Transducer Connections

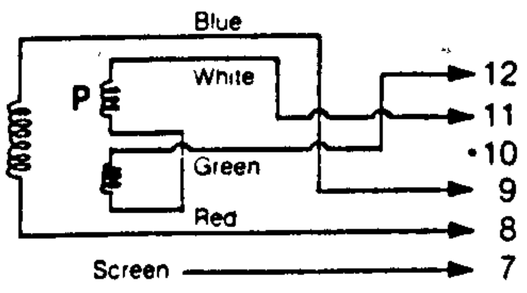
Half bridge



Full bridge



LVDT



Electrical

Power supply	±15V ±0·6V dc at 1% regulation or better
Current	+20, −40mA no load +40, −60mA max load
Input protection	Reverse polarity protected
Oscillator output	5V rms at 5kHz sinusoid max rated 50mA
Input sensitivity for rated output	0·5mV/V to 692mV/V in 9 coarse gain positions
Range of gain control	Fine: 3:1
Range of zero control	±20% max
Output	±5V into 10kΩ over-range linearity to ±10V
Output shift	±100% switchable
Output protection	Open/short circuit protected
Output ripple	< 10mA pk/pk at 10kHz
Output filter	2nd order low pass fc at 500Hz
Linearity	> 0·1% of rated output
Temperature range	0°C to 70°C
Temperature coefficient	Zero: > 0·01% fro/°C Gain: >0·01% fro/°C

Due to continuous improvements and changes in design we reserve the right to amend any specification without notice

SANGAMO
Schlumberger

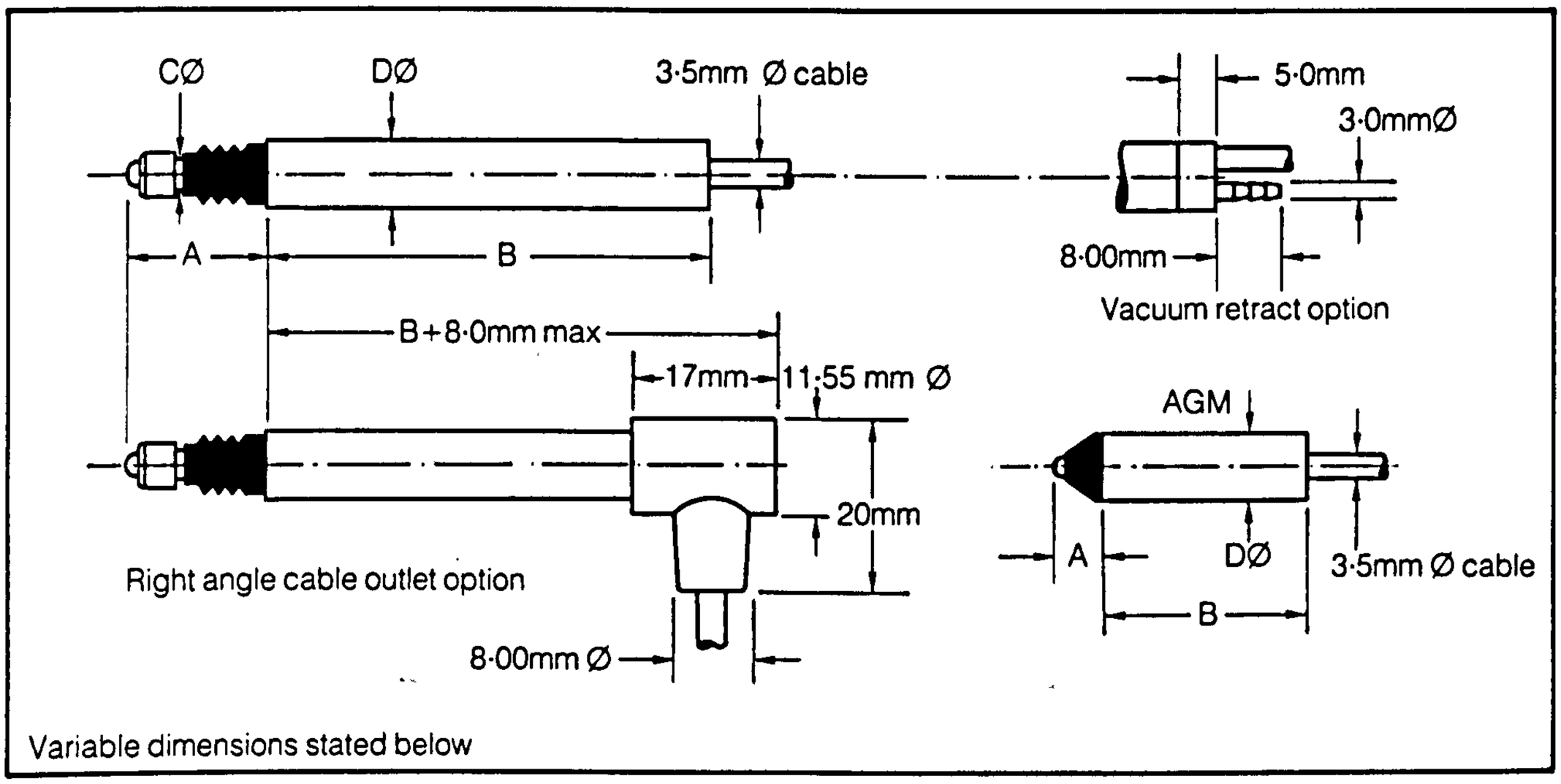
SANGAMO TRANSDUCERS
NORTH BERSTED, BOGNOR REGIS
W. SUSSEX, PO22 9BS, ENGLAND
TEL: (0243) 825011 TELEX: 86827

LO 202

Designed by Ian Burt. Printed in England by Gemini Press Ltd.

AG Series Gauging Transducers

The AG series offers a complete range of ball bearing movement gauging transducers with measuring capabilities up to 10mm. This series forms the base of all Sangamo Gauging Transducers, the rugged yet precise construction being shown in the exploded view opposite.



Variable dimensions stated below

Mechanical

Lvdt type	*AGM/0·5*	*AGS/0·5*	*A6G/1*	*AG/1*	*AG/1·5*	*AG/2·5*	*AG/5*
Half-bridge type	*AGMH/0·5*	*AGSH/0·5*	*A6GH/1*	*AGH/1*	*AGH/1·5*	*AGH/2·5*	*AGH/5*
Linear stroke	±0·5mm	±0·5mm	±1mm	±1mm	±1·5mm	±2·5mm	±5mm
Dimensions							
A. at electrical zero	5·0mm	15·0mm	14·0mm	12·5mm	16·0mm	16·0mm	23·0mm
B	24mm	40mm	48mm	51mm	72mm	77mm	96mm
C	—	4mm	3mm	4mm	4mm	4mm	4mm
D	8mm H6	8mm H6	6mm H6	8mm H6	8mm H6	8mm H6	8mm H6
Maximum stroke	±1·3mm	±1·5mm	±1·15mm	±1·5mm	±1·65mm	±2·65mm	±5·15mm
Spring rate	15g/mm	15g/mm	15g/mm	15g/mm	10g/mm	13g/mm	10g/mm
Force at electrical zero	70g	70g	70g	70g	70g	90g	105g
Temperature range	−40°C to +100°C						
Temperature coefficient	Zero: less than 0·005%/°C. Sensitivity: less than 0·005%/°C						
Linearity	0·1% 0·3% and 0·5% are available						

Electrical

Lvdt type	*AGM/0·5*	*AGS/0·5*	*A6G/1*	*AG/1*	*AG/1·5*	*AG/2·5*	*AG/5*
Current at 5Vrms 5kHz	12mA	12mA	30mA	20mA	10mA	10mA	8mA
Input/output phase shift	20°	20°	12°	9°	2°	2°	0°
Zero phase frequency	16kHz	16kHz	13·5kHz	12kHz	4kHs	4kHs	5kHs
Sensitivity into 100kΩ (per mm)	281mV/V	281mV/V	248mV/V	220mV/V	150mV/V	150mV/V	110mV/V
Half-bridge type	*AGMH/0·5*	*AGSH/0·5*	*A6GH/1*	*AGH/1*	*AGH/1·5*	*AGH/2·5*	*AGH/5*
Current at 5Vrms 10kHz	5mA	5mA	6mA	4mA	5mA	8mA	6mA
Input/output phase shift	3°	3°	2°	0°	2°	5°	3°
Zero phase frequency	12kHz	12kHz	13kHz	10kHz	12kHz	20kHz	14kHz
Sensitivity to 1kΩ (per mm)	84mV/V	84mV/V	88mV/V	80mV/V	86mV/V	84mV/V	63mV/V

C4 Facia Plug Connections - 25 pin 'D' details

C4.1 Hand Held Pendant - 25 pin 'D' Plug Connections

Table C4

25 pin 'D' plug, pin no.	Function
1	0V, common
15	Emergency stop
18	No.2 inhibit switch
20	Reset
21	No.1 switch
24	No.3 inhibit switch

C4.2 LVDT - 25 pin 'D' Plug Connections

Table C4.2

25 pin 'D' plug, pin no.	Function	Connected to conditioner amp pin no.
3	0V, screen	2
5	Primary energizing voltage	8
7	Primary energizing voltage	9
9	Secondary return signal	11
11	Secondary return signal	12

C4.3 Position Encoder - 25 pin 'D' Plug Connections

Table C4.3

25 pin 'D' plug pin no.	Function	Connected to encoder counter card edge connector pin no.
25	0V common	23A
24	A phase encoder 3	26B
12	B phase encoder 3	26A
23	A phase encoder 2	25A
11	B phase encoder 2	26B
22	A phase encoder 1	24A
10	B phase encoder 1	24B
13	+5V	27-29

C4.4 Limit Switches - 25 pin 'D' Plug Connections

Table C4.4

25 pin 'D' plug pin no.	Function	Connects with
25 & 23	Limit switch circuit	Serial circuit to manipulator switches from cut-out relays

C4.5 Motor Power - 25 pin 'D' Plug Connections

Table C4.5

25 pin 'D' plug pin no.	Function	Connects with
25-24 13-12	Motor no.1 power line	Connects motor to servo amplifier
22-23 10-11	Motor no.1 common return	Connects motor to servo amplifier
21-20 9-8	Motor no.2 power line	Connects motor to servo amplifier
19-18 7-6	Motor no.2 common return	Connects motor to servo amplifier
17-16 5-4	Motor no.3 power line	Connects motor to servo amplifier
15-14 3-2	Motor no.3 common return	Connects motor to servo amplifier

MODEL R-80
MODULAR ENCODER

The R-80 Series is a low cost 2 inch modular encoder. It incorporates a solid state LED light source and Radial-Line Sensor Array to insure precision output signals. By minimizing edge variations the R-80 is less sensitive to temperature and power supply variations encountered with conventional circumferential type sensor placement kit encoders.

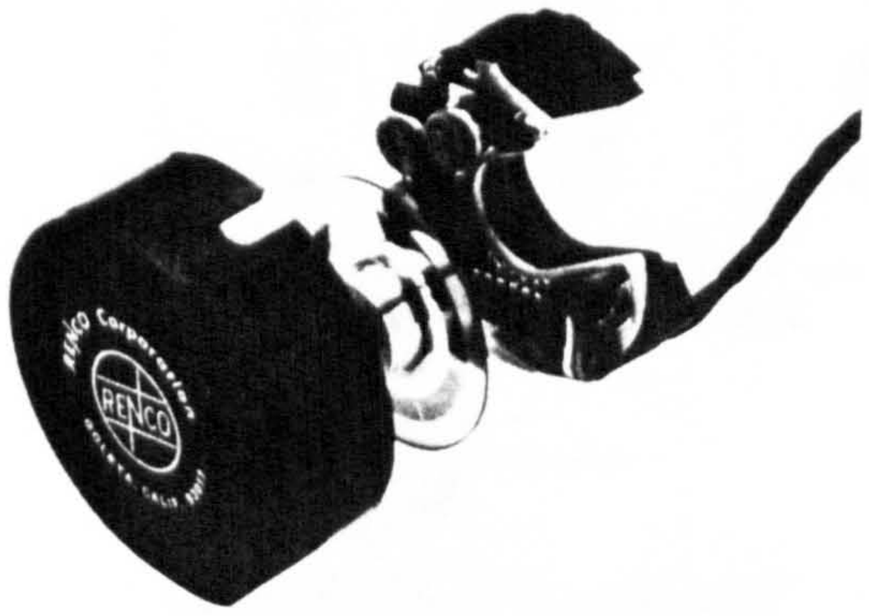
The R-80 comes as an easy to install kit that will fit all of the popular mounting hole patterns, thereby replacing existing kit encoders without requiring costly redrilling of motor mounting holes. Its radial-line sensor array is prealigned at the factory, and requires only a minimal alignment at time of installation. Furthermore, the reticle to disk air gap is less critical than that required with conventional kit encoders.

This modular encoder is available with standard resolutions up to 2500 CPR (higher resolutions available upon special request), also optional index and English/metric dual resolutions are featured. Differential sinewave, amplified sinewave, TTL, CMOS, and line drivers outputs are available. (Consult factory for specials.)

The R-80 encoder was designed for those cost conscious computer peripheral and process control applications where price and performance are major considerations. The R-80 Series presents convenient and inexpensive means of sensing direction, angular speed and position.

FEATURES

- Easy installation and alignment.
- Chrome on glass standard; mylar disk optional.
- Popular mounting patterns.
- Solid state LED source.
- Radial-line sensor array.
- Optional index marker pulse.
- Many standard pulse rates.
- 4 types of standard electrical outputs including linedrivers.
- Dual and English metric resolutions.
- Many other options.



MECHANICAL

Mechanical Outline ..... **See Figure 1**
Moment of Inertia ..... 100(10^{-6}) oz-in-sec^2
Angular Acceleration ..... 100,000 rad/sec^2
Slew Speed ..... 9,000 RPM
Hub Sizes ..... See Table I
Air Gap Setting ..... See Table II
Cable Wiring ..... See Table IV

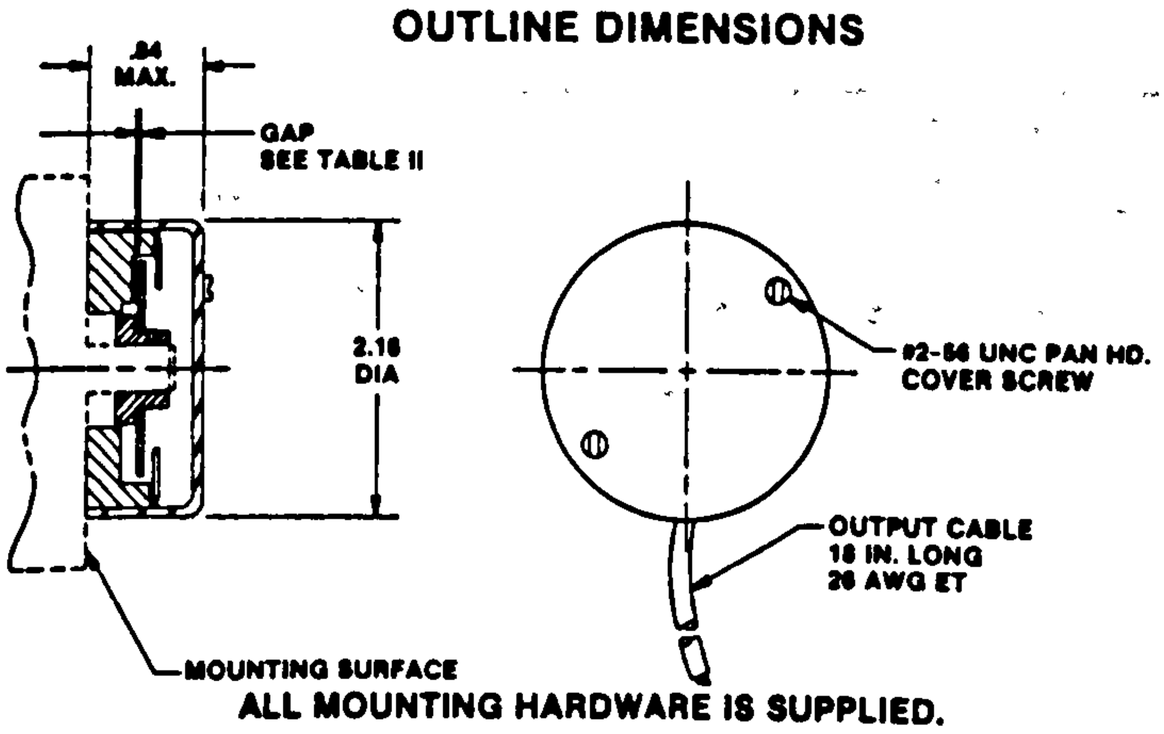
ELECTRICAL

Input Power Requirements
Type 1 Sine Wave Models ..... 5 vdc ± 5% @ 75 ma
Type 2 Amp. Sine Wave Models +15 vdc ± 10% @ 90 ma
-15 vdc ± 10% @ 30 ma
Type 3 Square Wave Models T^{2}L ..... 5 vdc ± 5% @ 100 ma max.
Type 4 CMOS Compatible ..... +12 or +15 vdc ± 10% @ 90 ma max.
Phasing ..... A leads B by 90° ± 36° electrically for CCW rotation (90° ± 9° available Type 3 and 4 only)
Code ..... Incremental
Illumination Source ..... L E D (for extra long life)
Sensors ..... Phototransistor radial array
Index ..... Available option
Operating frequency
Data ..... 75 Khz max.
Index ..... 20 Khz max.

ENVIRONMENTAL

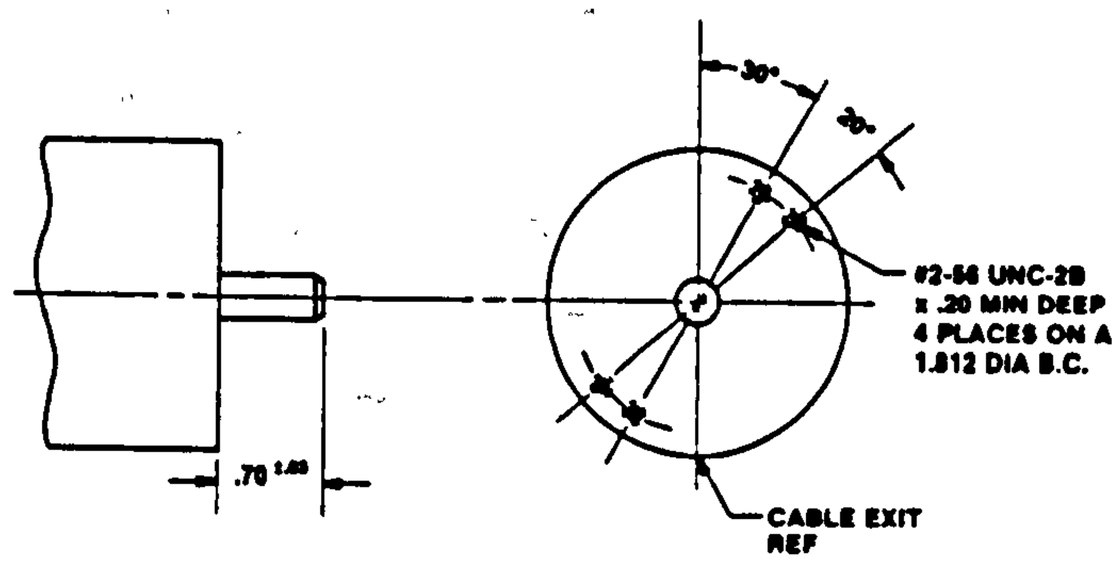
Operating Temperature ..... 0° to +70° C
Storage Temperature ..... -30° to +90° C
Humidity ..... 90% relative, no condensation

FIGURE 1
OUTLINE DIMENSIONS



ALL MOUNTING HARDWARE IS SUPPLIED.

FIGURE 2
STANDARD MOUNTING PATTERN



(Specify "H" for hole in cover)

TABLE I

HUB SIZE	
SHAFT +.0000 SIZE -.0003	SPECIFY
.2497	1/4
.3122	5/16
.3747	3/8
.4372	7/16
.4997	1/2
.2360	6MM
.3148	8MM
.3935	10MM
.4722	12MM

TABLE II

GAP SETTING			
RESOUTION	GAP TOLERANCE	GAP	
		TYPE 1,3,&4	TYPE 2
0 to 250	±.005	020	.020
251 to 750	+.002 −.000	.008	010
751 to 1024	+ 002 − 000	.006	006
1025 to 1500	+ 001 −.000	.004	.004
1501 to 2500	+ 001 − 000	003	.003

TABLE III

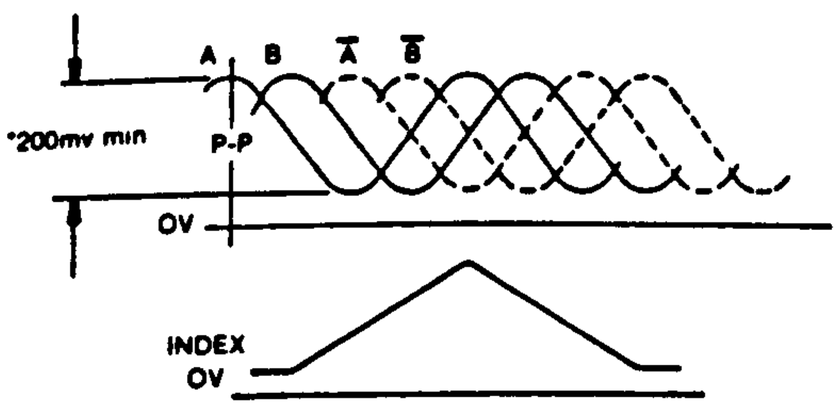
OPERATING VOLTAGE OPTION	
VOLTAGE	DESIGNATION
5.0	5
12.0	12
15.0	15

TABLE IV

OUTPUTS				
TYPE 1	TYPE 2	TYPE 3 & 4	TYPE 3 LINE DRIVERS	LEAD COLOR
A	A	A	A	WHT
$\overline{A}$			$\overline{A}$	YEL
B	B	B	B	GRN
$\overline{B}$			$\overline{B}$	BLU
INDEX	INDEX	INDEX	INDEX	ORG
+5V	+15V	Vcc	Vcc	RED
	−15V			BLU
GND	GND	GND	GND	BLK
			$\overline{\text{INDEX}}$	BRN

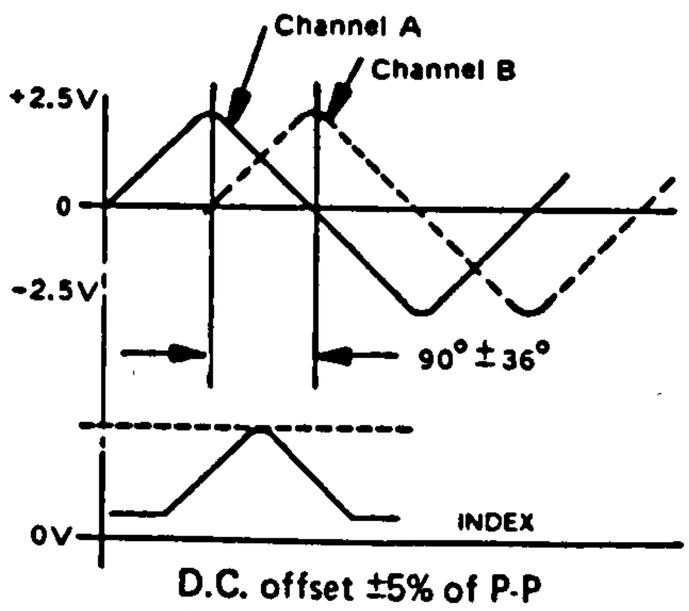
OUTPUT CONFIGURATION

TYPE 1 OUTPUT: Differential Sine Wave



Output load impedance 470Ω
*Typical at 10 kHz

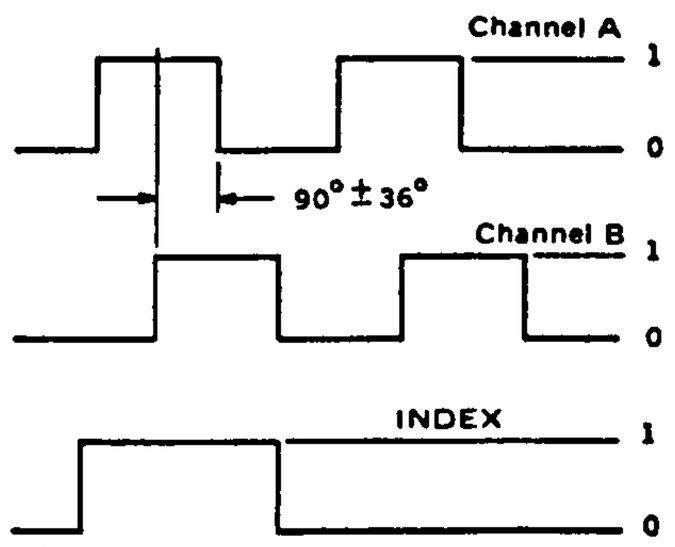
TYPE 2 OUTPUT: Amplified Sine Wave



D.C. offset ±5% of P-P
Output load > 1kΩ

TYPE 3 OUTPUT: T²L Compatible

TYPE 4 OUTPUT: CMOS Compatible



T²L Logic 1 = Vcc @ 2KΩ
Logic 0 = .7V max. @ 6.0 ma sink

CMOS Logic 1 = Vcc −.10V @ 10kΩ
Logic 0 = .7V max. @ 2.5 ma sink

ORDERING INFORMATION

To order, supply the appropriate designation; see codes shown below.

R—802 ____ — ____ — ____ — ____ — ____ — ____ — **LD** — **S**

Type Output 1 Sensor, differential 2 Amplified zero-crossing sine wave 3 TTL compatible 4 CMOS 9 Custom design circuit	Outputs A Single B Dual 90° quadrature C Single + index D Dual + index	Resolution Defines number of cycles in one revolution of code disk	Shaft Size See Table I	H = Hole in cover with shaft seal Blank = No hole	Input Voltage See Table III	LD = Line Drivers Type SN75158	Special Options Consult Factory

RENCO CORP. 26 Coromar Drive Goleta, CA 93117 Phone (805) 968-1525 TWX 910-334-1180
Division of Electro-Craft Corporation

MODEL HDR70 HEAVY DUTY INDUSTRIAL ENCODER

The HDR70 Series is a rugged encoder designed for industrial applications where many encoders cannot survive. This Size 25 Encoder features a rugged cast aluminum housing and cover along with heavy duty sealed bearings and a 3/8" CRES shaft. Popular flange or servo mounting, rear or side mount connector are offered. Shaft seal is an available option.

This HDR70 Series features a solid state L E D light source and Radial-Line Sensor Array to insure stable output signals and reliability under harsh machine environment. Standard output signals conform to TTL or CMOS logic levels. Available options are line drivers, direction sensing logic with X 2 or X 4 count multiplication and external selectable English metric count.

This Heavy Duty Encoder is designed to provide extremely reliable positioning information under adverse industrial environmental conditions. Typical applications are NC machine tools, synchronization of flow lines, solar trackers, material processing equipment, oil drilling equipment, robots, flow measuring systems, automatic welding equipment, chemical batching equipment, automated assembly line transfer machines and many more.

Special output configurations providing common mode noise rejection are available for use in electro-magnetic noise environments. Consult factory for additional information on amplified sinewave or differential amplified sinewave outputs.

FEATURES

- Cast aluminum housing and cover.
- Popular flange or servo mounting.
- Radial-Line Sensor Array.
- Direction sensing logic.
- Standard resolution to 5000 PPR.
- English/metric switch selectable.
- Pin for Pin—bolt for bolt inter-changeable with standard industrial encoders.

MECHANICAL:

Dimensions	See Figures 1 & 2
Shaft size	.3748 +.0000 -.0003
Shaft run out	.0005 TIR
Starting torque	
Models without shaft seal (sealed bearings standard)	1.0 oz in. max.
Models with shaft seal	5.0 oz in. max.
Moment of inertia	$4.0(10^{-4})$ oz-in. sec^2 max.
Slewing speed	
Standard	5000 RPM max.
Special order	10,000 RPM max.
Angular acceleration	10^6 rad/sec^2
Bearing life	10^8 Revolution: min.
Shaft loading	
Radial	35 lb. max.
Axial	40 lb. max.
Shaft rotation	Continuous & reversible
Housing & Cover	Rugged cast aluminum
Weight	20 oz max.

ELECTRICAL:

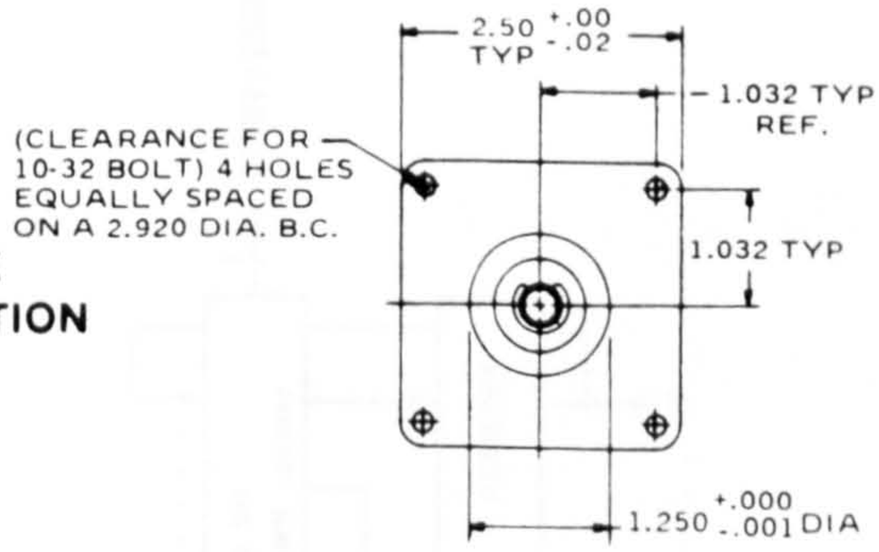
Output circuitry	TTL or CMOS compatible for sinewave models (amplified & differential) consult factory.
Output logic levels	
TTL	Logic "1": Vcc thru 2KΩ Logic "0": .4V max. at 15 ma sink
CMOS	Logic "1": Vcc—1.0 V
Line driver output device	Type, DS8830
Operating frequency	
Data	75 Khz max.
Index	20 Khz max.
Input power requirements	5V, 12V, 15V & 24V units require 125 ma max.
Illumination source	
Type	L E D (filament free for vibration environments)
Useful life	100,000 hrs min.
Encode MTBF	41,000 hours

ENVIRONMENTAL:

Temperature:	
Operating:	0° C to +70° C
Storage	-25° C to 90° C
Humidity	Up to 98% RH
Vibration	5 to 2000 Hz @ 20 G's
Shock	50 G's for 11 MS duration

PHYSICAL CHARACTERISTICS

"F" FLANGE MOUNTING OPTION



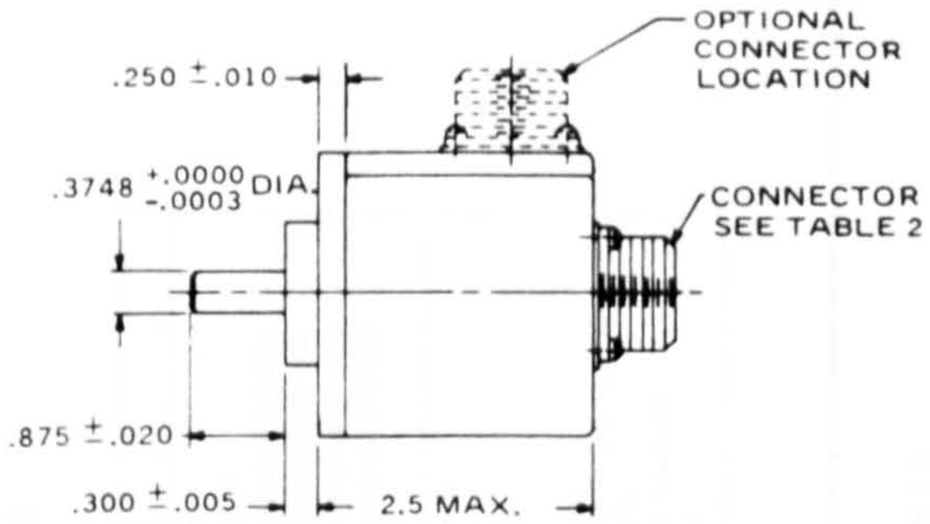
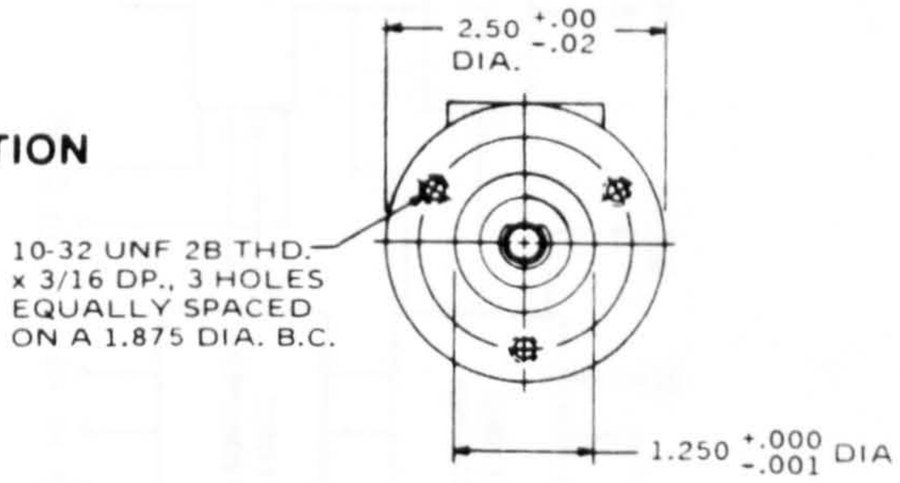


FIGURE 1

"S" SERVO MOUNTING OPTION



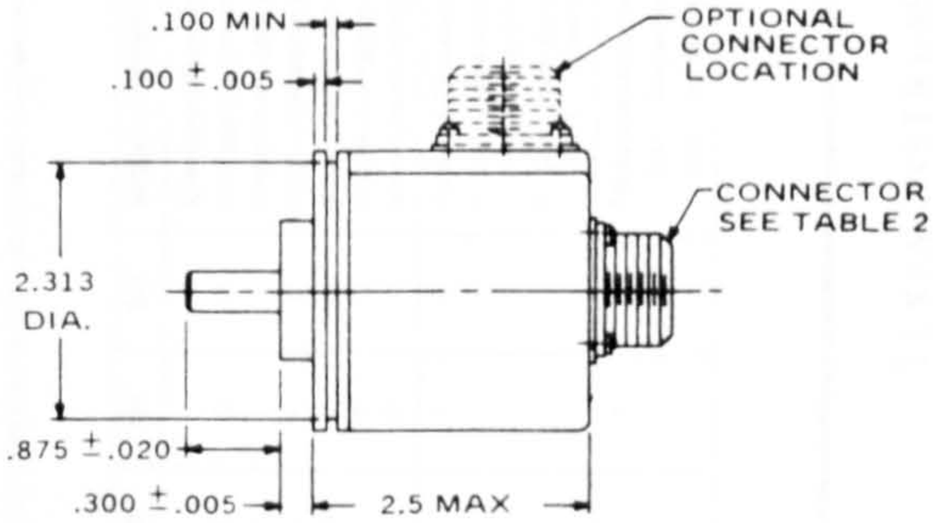


FIGURE 2

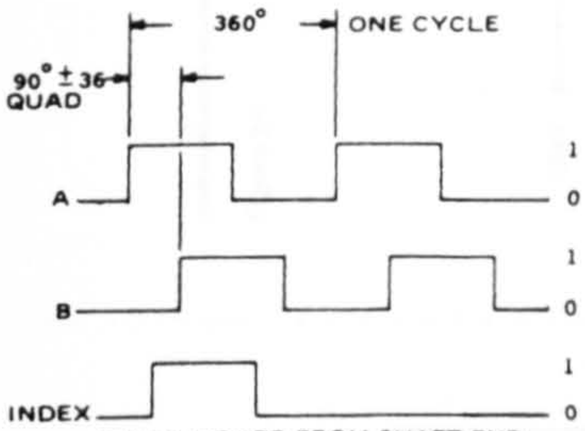
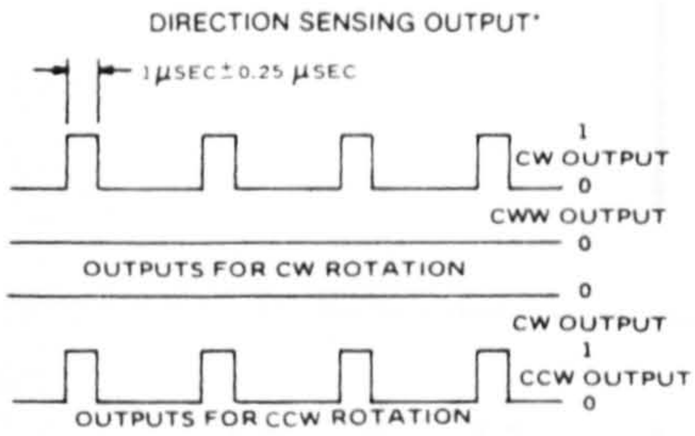


TABLE 1

CODE FORMAT	
SPECIFY	**OUTPUT**
A	SINGLE CHANNEL
B	DUAL QUADRATURE CHANNELS
C	SINGLE CHANNEL WITH INDEX
D	DUAL QUADRATURE WITH INDEX
X1	PULSED OUTPUT with direction sensing
X2	2X MULTIPLICATION with direction sensing
X4	4X MULTIPLICATION with direction sensing



* DIRECTION SENSING UNITS HAVE LINE DRIVERS WTIHOUT INDEX AS STANDARD OUTPUTS

TABLE 2 CONNECTOR

CONNECTOR	MS3102A16S-1P					MS3102A18-1P
OUTPUT OPTION	**CHANNELS A,B, AND INDEX**	**CH. A&B WITH LINE DRIVERS**	**CH. A INDEX WITH LINE DRIVERS**	**ENGLISH METRIC**	**DIRECTION SENSING**	**CH. A,B, & INDEX WITH LINE DRIVERS**
PIN:						
A	CH. A	A	$\overline{A}$	A	CW	A
B	CH. B	B	$\overline{A}$	B	CCW	B
C	INDEX	$\overline{A}$	INDEX	ENGLISH SELECT	$\overline{CW}$	INDEX
D	+Vcc	+Vcc	+Vcc	+Vcc	+Vcc	+V cc
E	—	$\overline{B}$	$\overline{INDEX}$	METRIC SELECT	$\overline{CCW}$	NO CONN
F	GND	GND	GND	GND	GND	GND
G	CASE GND	CASE GND	CASE GND	CASE GND	CASE GND	CASE GND
H	N/A	N/A	N/A	N/A	N/A	$\overline{A}$
I	N/A	N/A	N/A	N/A	N/A	$\overline{B}$
J	N/A	N/A	N/A	N/A	N/A	$\overline{INDEX}$

ORDERING INFORMATION

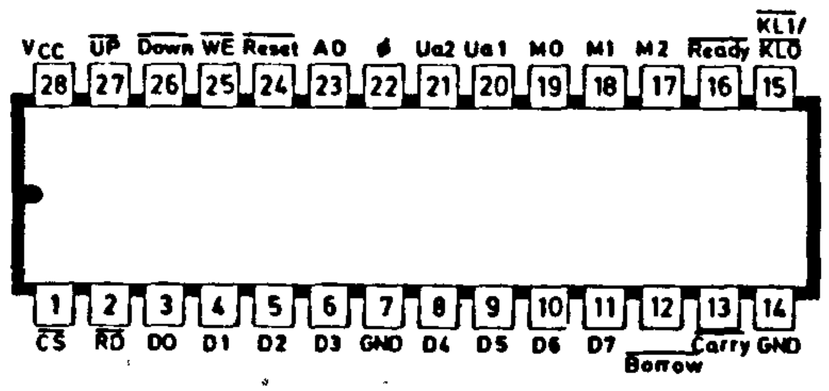
HDR70	D	1000	F	R	S	12	LD
	CODE FORMAT SEE TABLE I	CYCLES PER SHAFT REV	MOUNTING SERVO ..... S FLANGE ..... F	CONNECTOR LOCATION SIDE MOUNTING ..... S REAR MOUNTING ..... R	SHAFT SEAL SEALED BEARINGS STD SHAFT SEAL ..... S	SUPPLY VOLTAGE TTL ..... 5 CMOS ..... 12 CMOS ..... 15 24V LOGIC 24	LINE DRIVE OPTION SPECIFY ..... LD
EXAMPLE OF MODEL NO	QUADRATURE OUTPUTS WITH INDEX	1000 CYCLES	FLANGE MOUNTING	REAR CONNECTOR	SHAFT SEAL	12 VOLT INPUT VOLTAGE	LINE DRIVER OUTPUT

RENCO CORP. 26 Coromar Drive Goleta, CA 93117 Phone (805) 968-1525 TWX 910-334-1180

Division of Electro-Craft Corporation

TYPE SN74LS2000
(DIRECTION DISCRIMINATOR)

JORN PACKAGE



Features

- Direction Discriminator to identify forward/backward direction.
- Separate 16 BIT cascadable up/down counter.
- Pulse-width measurement with either forward or backward counting.
- Frequency measurement.
- 8-Bit parallel tri-state data bus.
- Simple write and read procedure.
- All inputs and outputs TTL compatible.
- Single +5 Volt supply.

INTRODUCTION

The SN74LS2000N DIRECTION DISCRIMINATOR is a device designed for use with TEXAS INSTRUMENTS or other microprocessors in all kinds of applications where it is required to evaluate the output signals, received from an incremental length measurement system or any other incremental transducer (like robots, automatic turning lathe, etc).

Additional features allow it to be used also in microprocessor-systems where automatically pulse-width frequency measurement or up/down counting is required

The SN74LS2000N is a 28 Pin, single supply (+5 V_{CC}), low power schottky array technology device, with all inputs and outputs - TTL compatible

SPECIFICATION AND FUNCTIONAL DESCRIPTION

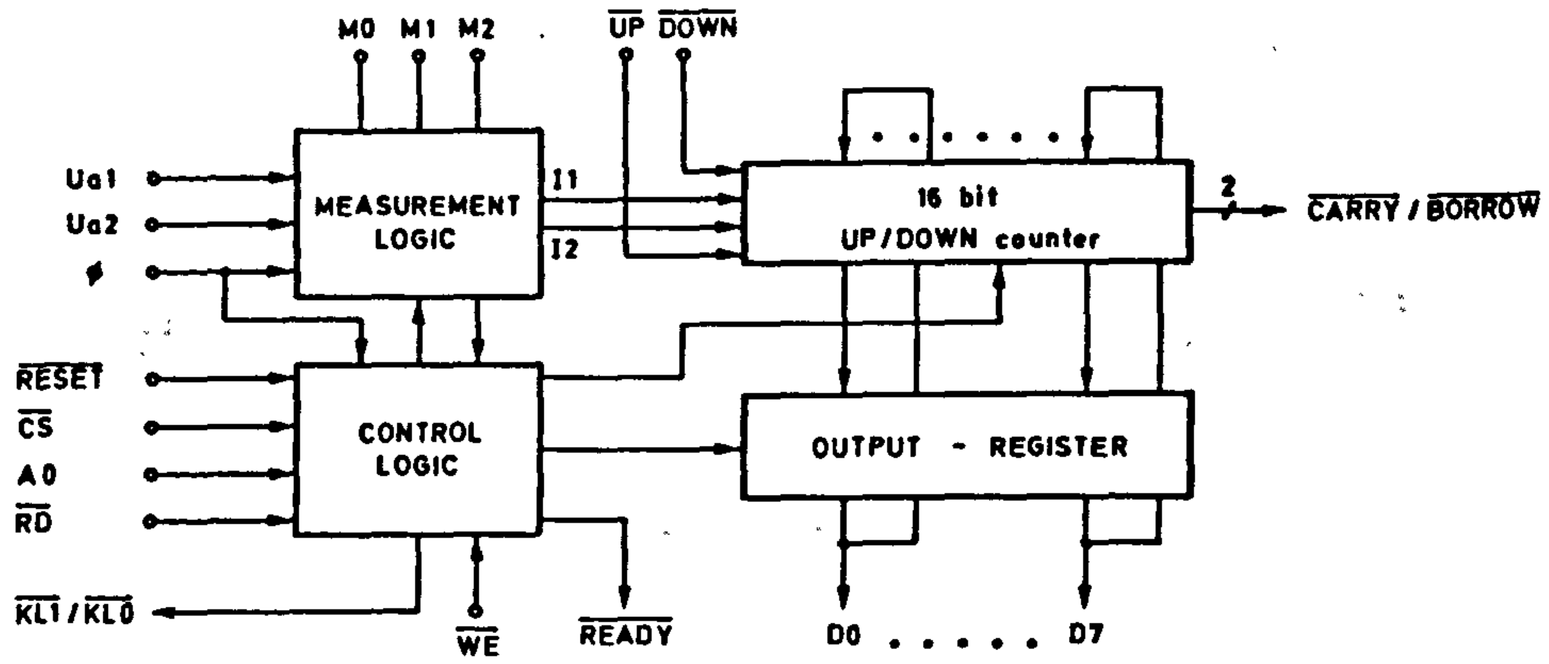
Architecture

Figure 2-1 shows the functional block-diagram of the SN74LS2000N The direction discriminator consists of the measurement-logic, control-logic 16 BIT up/down counter and output-register The measurement-logic generates, depending on the input signals Ua1 Ua2 and the mode state the internal I1/I2 pulses for the up/down count

The control-logic circuitry is the interface between controlsignals of the processor and the SN74LS2000N This logic generates internal control signals to prevent faultless behavior of the device

The output-register which operates as a one word memory for the 16 BIT evaluated information, prevents wrong information to be read in consequence of the 2 Byte reading operation In both the pulse-width and the frequency measurement mode it enables the measurement circuitry to make a new sample

Figure 2-1

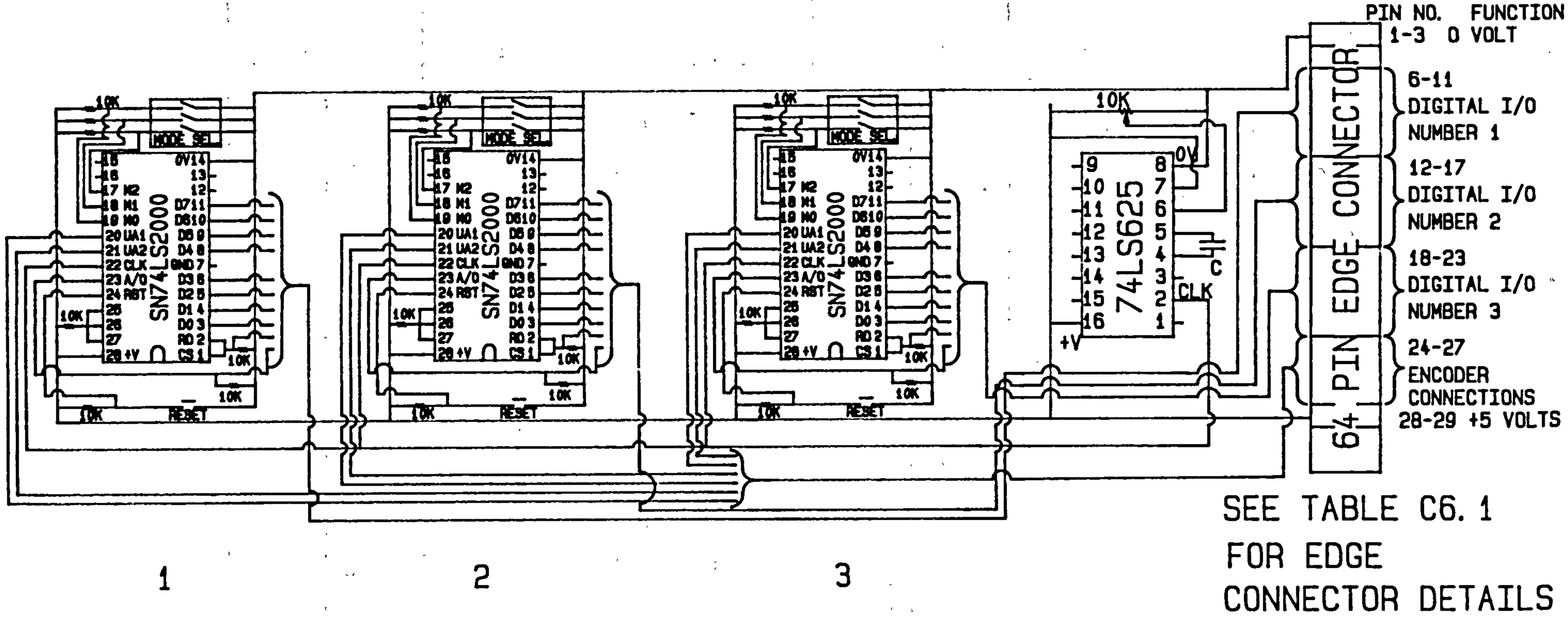


Mode Selection

Table 2-1 shows the different mode functions of the SN74LS2000N (see also fig 2-2)

Table 2-1

M2	M1	M0	MODE DESCRIPTION
			Direction Discriminator
0	0	0	direction discriminator inhibit
0	0	1	single count pulse, synchronous with Ua1
0	1	0	single count pulse synchronous with Ua2
0	1	1	double count pulse synchronous with Ua1
1	0	0	double count pulse synchronous with Ua2
1	0	1	quadruple counting
1	1	0	**Pulse width measurement** 1. Ua1 = gate signal Ua2 'H' = up counting 2 Ua1 = gate signal Ua2 'L' = down counting
1	1	1	**Frequency measurement** Ua1 = frequency to be measured Ua2 = start/stop gate



SEE TABLE C6.1
FOR EDGE
CONNECTOR DETAILS

ENCODER COUNTER INTERFACE BOARD
FIGURE C6.1

C6.1 Encoder Counter Interface Board - 64 Pin Edge Connector Board

The edge connector provides 32 rows of two pins, marked A and B, giving a total of 64 pins.

Table C6.1

Edge connector pin no.	Function	Connected to
1-3 A & B	0V, common	0V, common of power supply
6B	No.1 counter output bit 0	Pin 21 port A, no.1 8255
6A	No.1 counter output bit 1	Pin 22 port A, no.1 8255
7B	No.1 counter output bit 2	Pin 23 port A, no.1 8255
7A	No.1 counter output bit 3	Pin 24 port A, no.1 8255
8B	No.1 counter output bit 4	Pin 25 port A, no.1 8255
8A	No.1 counter output bit 5	Pin 26 port A, no.1 8255
9B	No.1 counter output bit 6	Pin 27 port A, no.1 8255
9A	No.1 counter output bit 7	Pin 28 port A, no.1 8255
10B	INPUT chip select (CS & RD)	29 port C, no.1 8255
10A	INPUT high/low byte select	30 port C, no.1 8255
11B	Spare	Pin 31 port C
11A	Common	33 common no.1 8255
12B	No.2 counter output bit 0	Pin 13 port B, no.1 8255
12A	No.2 counter output bit 1	Pin 14 port B, no.1 8255
13B	No.2 counter output bit 2	Pin 15 port B, no.1 8255
13A	No.2 counter output bit 3	Pin 16 port B, no.1 8255
14B	No.2 counter output bit 4	Pin 17 port B, no.1 8255
14A	No.2 counter output bit 5	Pin 18 port B, no.1 8255
15B	No.2 counter output bit 6	Pin 19 port B, no.1 8255
15A	No.2 counter output bit 7	Pin 20 port B, no.1 8255

Table C6.1 - continued

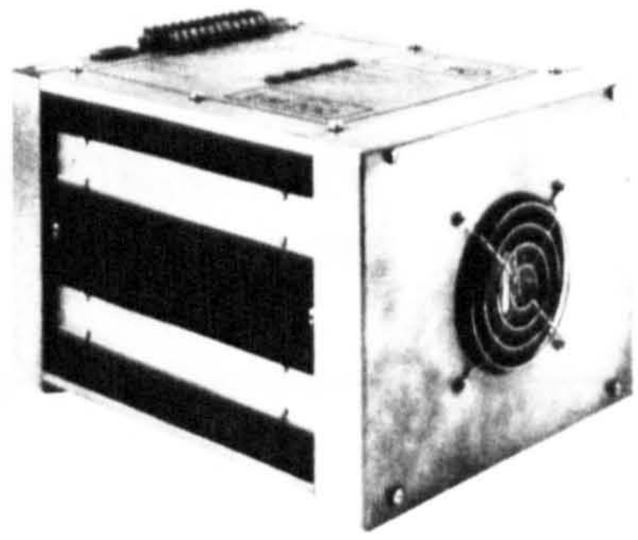
16B	INPUT chip select (CS & RD)	Pin 12 port C, no.1 8255
16A	INPUT high/low byte select	Pin 11 port C, no.1 8255
17B	Spare	Pin 10 port A, no.1 8255
17A	Common	Pin 8 common, no.1 8255
18B	No.3 counter output bit 0	Pin 21 port A, no.2 8255
18A	No.3 counter output bit 1	Pin 22 port A, no.2 8255
19B	No.3 counter output bit 2	Pin 23 port A, no.2 8255
19A	No.3 counter output bit 3	Pin 24 port A, no.2 8255
20B	No.3 counter output bit 4	Pin 25 port A, no.2 8255
20A	No.3 counter output bit 5	Pin 26 port A, no.2 8255
21B	No.3 counter output bit 6	Pin 27 port A, no.2 8255
21A	No.3 counter output bit 7	Pin 28 port A, no.2 8255
22B	INPUT chip select (CS & RD)	29 port C, no.2 8255
22A	INPUT high/low byte select	30 port C, no.2 8255
23B	Spare	31 port C, no.2 8255
23A	Common	33 common, no.2 8255
24B	INPUT phase A	Optical Encoder no.1
24A	INPUT phase B	Optical Encoder no.1
25B	INPUT phase A	Optical Encoder no.2
25A	INPUT phase B	Optical Encoder no.2
26B	INPUT phase A	Optical Encoder no.3
26A	INPUT phase B	Optical Encoder no.3
27B	0V	Encoder's common
27A	+5V	Encoder's power supply
28-29 A & B	+5V supply	+5V battery backed supply

450 watt dc servo system

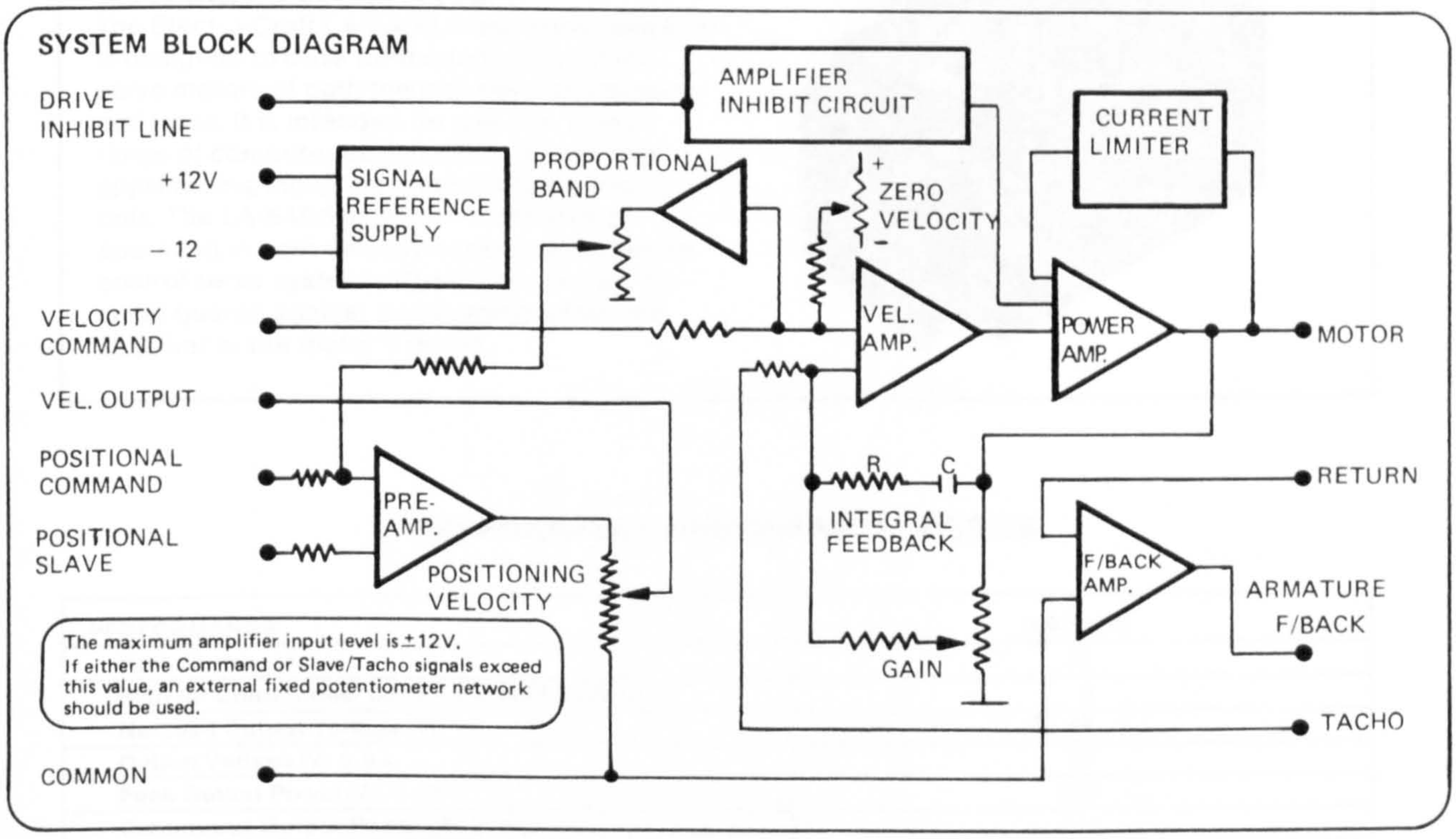
EM 200

The EM200 system provides bi-directional velocity or positional control of permanent magnet dc servo motors up to 300 watts (0.4 hp) output power.

Provided with an integral power supply, the EM200 utilises linear dc amplifier techniques to ensure rapid motor response and smooth shaft rotation at full torque, even when very low drive speeds are selected.



The system operates from remote command signals, a ± 12Vdc reference supply being provided for command and slave potentiometer excitation.



SPECIFICATION

Supply	:	240, 220, 115 Vac 50 or 60 Hz
Output Voltage	:	±50 Vdc max.
Output Current		
Continuous Rating	:	9 amps max.
Peak Rating	:	30 amps max.
Reference Supply	:	± 12 Vdc
Rating	:	15 mA
Stability	:	0.006 V/deg. C.
Resolution	≥	2000:1

The system will detect changes in input signals of 1 mV. The value quoted reflects typical tacho and slave transducer senstitivity.

FEATURES

- Bi-directional velocity control
- Zero speed adjustment with separate zero select line (amplifier inhibit)
- Pre-amplifier to provide positional control and adjustable deceleration
- Control of velocity in positioning mode
- Adjustable gain and feedback circuitry
- Current limit circuitry with temperature monitoring to protect amplifier and motor
- Peak current facility for rapid motor response
- Robust industrial enclosure with integral fan
- Separately fused power stages for high reliability

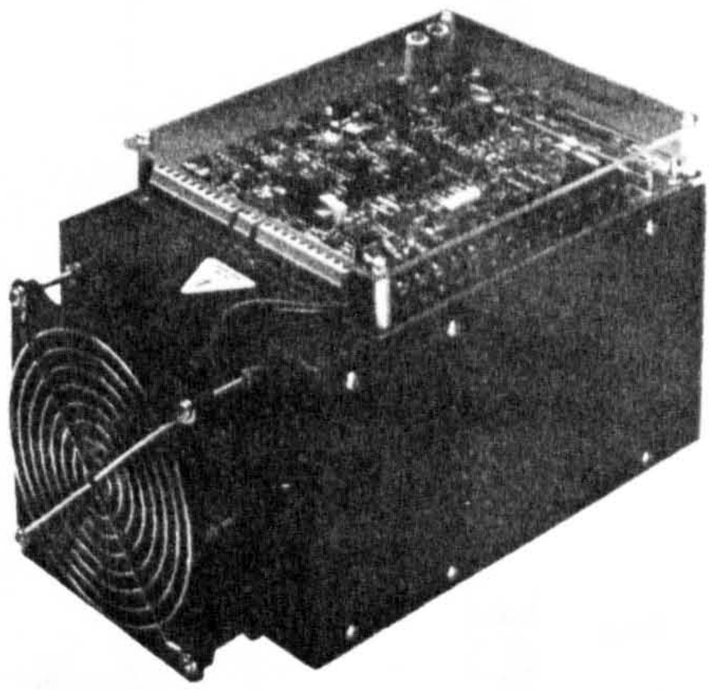
ELECTRO-CRAFT CORPORATION

1600 Second Street South, Hopkins, Minn. 55343 • (612) 931-2700

MAXIS™

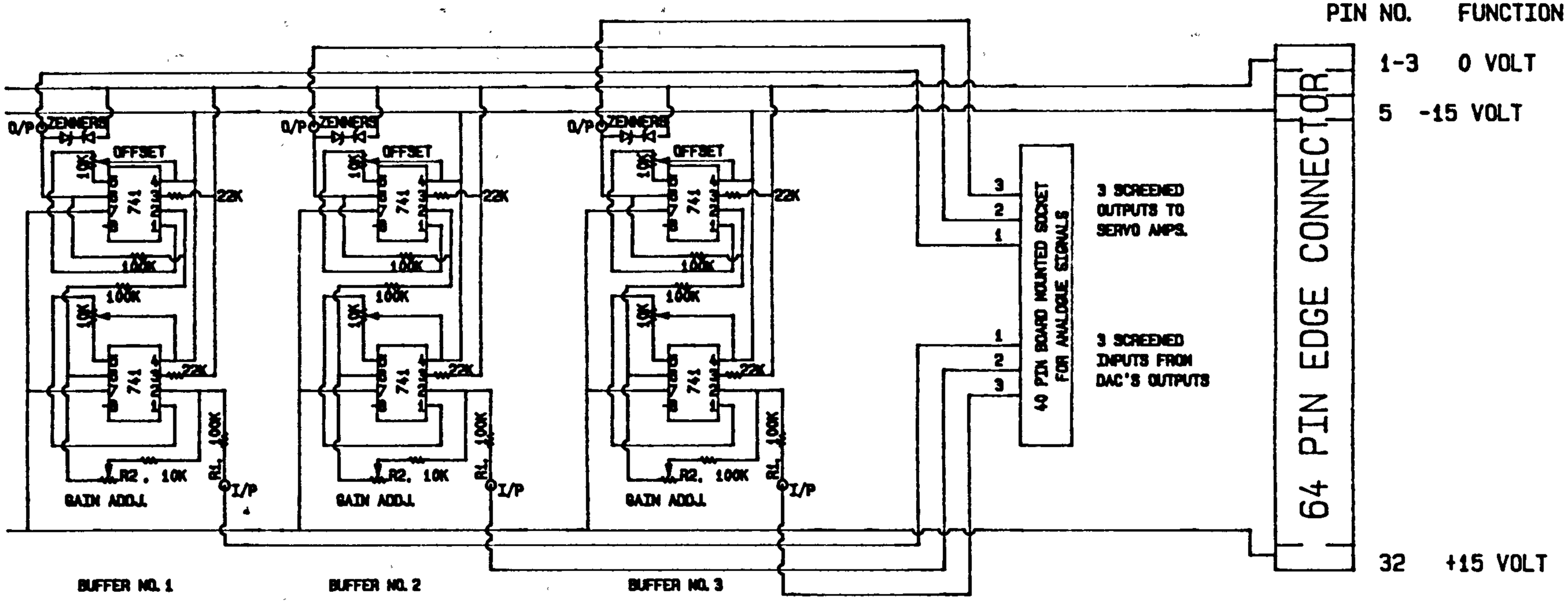
LA-5400
Linear Amplifier

The Electro-Craft LA-5400 linear servo amplifier is designed to drive permanent magnet DC servo motors of both the iron core and moving coil types. It is intended for use in a broad range of computer peripheral and industrial applications, including machine tools and robots. The LA-5400 amplifier is capable of operating in both velocity control and position control servo systems. Special protection circuitry guards against overloading either the amplifier or the motor it drives.



PERFORMANCE AND CHARACTERISTICS

PERFORMANCE	LA-5400
Peak Current (A)	22
RMS (or Continuous) Output Current (A) at 50 C	9
No-Load Output Voltage (V)	± 44
Output Voltage (V) @ 9 A	35
Peak Output Power (W) @ 25 V	550
Continuous Output Power (W) @ 35 V	315
ELECTRICAL CHARACTERISTICS	
Open Loop DC Gain (A/V)	
Velocity Command Signal - VCS	1200
Auxiliary Signal	1200
Tachometer Signal	1200
Bandwidth (Hz)	DC-2000
VCS Input Impedance (kΩ)	14 minimum
Input Amplifier Drift (μV/ C)	15
Operating Temperature (C)	0-50
VCS Input Signal Range (V)	±10 nominal; ±60 maximum
PHYSICAL CHARACTERISTICS	
Height (in)	6.0
Width (in)	6.5
Length (in)	10.5
Weight (lb)	15.7
INPUT POWER REQUIREMENTS	115 V or 230 V, 50/60 Hz, 1φ



DAC. 'S ANALOGUE INTERFACE BUFFER BOARD
FIGURE C8. 1

C9 Computer Interface Cards

All three types of interface cards operate on a memory mapped I/O segment basis. The individual addresses of the various cards and functions are given in the relevant machine code listings in appendix D3.

C9.1 Tecmar Labmaster

The Tecmar Labmaster interface card provides a multitude of interface functions. But in this work is used solely to convert a single channel from an analogue signal to a 12 bit digital representation. The analogue signal is provided by the conditioning amplifier of the LVDT force sensor. This signal is measured by channel 15 which is located at pin 5 on the 40 pin connection, P1D, within the daughter board. The 0V return wire is connected to the common pins, 21-40.

C9.2 Tecmar Dadio Board

The Tecmar Dadio board (digital to analogue and digital input/output) provides 4 channels of 12 bit digital to analogue conversion. Various voltage ranges are jumper selectable on the board. It is set, in this work, to produce voltage swings of ±10V for 12 bit variation.

The outputs from the DAC's terminate at a 9 pin 'D' edge connector socket. Output channel '0' is unused and channels 1, 2 and 3 are connected as table C9.2.1.

Table C9.2.1

9 pin 'D' edge connector, pin no.	Function	Connections to input of analogue buffer number
9	Channel 3 DAC	3
8	Channel 2 DAC	2
7	Channel 1 DAC	1
1-5	common return	0V

C9.3 Tecmar Baseboard

Tecmar Baseboard interface card provides 96 channels of I/O. These are arranged as 4 ports of 24 I/O lines. Each port terminates at board mounted 40 pin connectors, P1 and P4. The baseboard mounted in the Controller computer is responsible for both position encoder interface and the 16 bit parallel inter-computer communications. The baseboard mounted in the trajectory planner is used solely for parallel inter-computer communications.

The 40 pin connectors are arranged as table C9.3.1 for encoder interface and table C9.3.2 for parallel communication.

Table C9.3.1

Position Encoder Baseboard Connections

(a) No.1 8255, 40 pin channel connection

No.1 40 pin channel pin no.	Function Port A, B, C or 0V	Connected to Encoder board 64 pin edge connector pin no.
8	0V, common	17A
9	C4	N/C
10	C6	17B
11	C0	16A
12	C2	16B
13	B0	12B
14	B1	12A
15	B2	13B
16	B3	13A
17	B4	14B
18	B5	14A
19	B6	15B
20	B7	15A
21	A0	6B
22	A1	6A
23	A2	7B
24	A3	7A
25	A4	8B
26	A5	8A
27	A6	9B
28	A7	9A
29	C5	10B
30	C7	10A

Table C9.3.1 continued

31	C1	11B
32	C3	N/C
33	0V, common	11A

(b) No.2 8255, 40 pin channel connection

No.2 40 pin channel pin no.	Function Port A, B, C or 0V	Connected to Encoder board 64 pin edge connector pin no.
21	A0	18B
22	A1	18A
23	A2	19B
24	A3	19A
25	A4	20B
26	A5	20A
27	A6	21B
28	A7	21A
29	C5	22B
30	C7	22A
31	C1	23B
32	C3	N/C
33	0V, common	23A

C9.3.2 Parallel Communication

Parallel communication is achieved by utilizing two machine code routines, 'WRDOUT' and 'WRDIN', which are written to allow bi-directional data transfer.

The number three 40 pin board mounted sockets, on each baseboard within both computers, are connected by a 40 pin ribbon. This connects together the A, B and C ports of the number three channel of each board.

All 40 pins are connected for convenience except the 12V and 5V lines, to avoid the possibility of circulating power.

The relevant pin connections are given in table 9.3.2

Table C9.3.2

Parallel Communications - baseboard connections

Channel 3 40 pin connector, pin no.	Function
6-8	common, 0V
9	C4 handshake line
10	C6 not used
11	C0 handshake line
12	C2 not used
13-20	B port data byte
21-28	A port data byte
29	C5 handshake line
30	C7 not used
31	C1 handshake line
32	C3 not used
33-35	common, 0V

APPENDIX D

COMPUTER LISTINGS

D2 Trajectory Planning Program, 'MULTIN'

D1 Controller Program, 'MULTISAMP'

D3 Assembler Program Listings: 'PORTIN', 'OUT', 'INP', 'IADCON', 'WRDOUT', 'WRDIN'

APPENDIX D1

TRAJECTORY PLANNING PROGRAM 'MULTIN'

```
C********************** PROGRAM MULTIN ***********************************
C**** READS POINT COORDINATES AND CONTROL WORDS FROM A FILE **********
C**** AND IN REAL TIME LINEARLY INTERPOLATES THESE ********************
C**** TO FORM THE DESIRED POSITION OF THE CONTROLLER ******************
C**** READS IN BOTH GLOBAL AND TOOL DATA FROM A FILE ******************
C**** ALSO INTERPOLATES GIVING DESIRED FORCE DATA **********************
C
      PROGRAM MULTIN
C
C
C
      COMMON NCOUNT,NSTART,TDELT,XTRAJ,YTRAJ,THTRAJ,XCOORD,YCOORD,
     @THCOORD,TIME,MODE,COORD,NT,NINT,NTRAJ,NLP,XM,YM,THETAM,XERR,
     @YERR,THERR,KXVEL,KYVEL,KTHVEL,XDOT,YDOT,THDOT,KTVEL,KNVEL,KBVEL,
     @TM,NM,FNCORD,FNINC,FNINTL,DESFN,MESFN,SAMPLE,NSAMPLE,SWITCH
     @XERI,KXERI,YERI,KYERI,THERI,KTHERI,IXER,IYER,ITHER,KIXER,KIYER,
     @KITHER,TERI,KTERI,NERI,KNERI,BERI,KBERI,ITER,INER,IBER,KITER,
     @KINER,KIBER,IXLIM,IYLIM,ITHLIM,ITLIM,INLIM,IBLIM
C
      COMMON /A1/ XMES,YMES,THMES,FNMES,DFNMES,XDES,YDES,THDES,FNDES
C
      REAL XCOORD,YCOORD,THCOORD,XTRAJ,YTRAJ,THTRAJ,TDELT,
     @TIME,XINC,YINC,THINC,X,Y,TH,XM,YM,THETAM,XERR,YERR,THERR,
     @KXVEL,KYVEL,KTHVEL,XDOT,YDOT,THDOT,N,T,TINC,NINC,X1,Y1,DELXM,
     @DELYM,CTH1,STH1,DELTM,DELNM,TM,NM,TERR,NERR,TDOT,NDOT,NCOORD,
     @TCOORD,KTVEL,KNVEL,KBVEL,FNCORD,FNINC,FNINTL,DESFN,MESFN,
     @XMES,YMES,THMES,FNMES,DFNMES,XDES,YDES,THDES,FNDES
     @XERI,KXERI,YERI,KYERI,THERI,KTHERI,IXER,IYER,ITHER,KIXER,KIYER,
     @KITHER,TERI,KTERI,NERI,KNERI,BERI,KBERI,ITER,INER,IBER,KITER,
     @KINER,KIBER,IXLIM,IYLIM,ITHLIM,ITLIM,INLIM,IBLIM
C
      DIMENSION XTRAJ(4),YTRAJ(4),THTRAJ(4),MODE(99),
     @COORD(99),ERROR(99),XCOORD(99),YCOORD(99),THCOORD(99),
     @TIME(99),NCOORD(99),TCOORD(99),FNCORD(99),XMES(1000),
     @YMES(1000),THMES(1000),FNMES(1000),DFNMES(1000),XDES(1000),
     @YDES(1000),THDES(1000),FNDES(1000)
C
      INTEGER*4 NT,NINT,NCOUNT,NSTART,NTRAJ,NLP,NSAMPLE
C
      INTEGER*2 SWITCH
C
      LOGICAL*2 MEMORY,SAMPLE
C
      CHARACTER*10 MODE,COORD,ERROR,FNAME
C
C**** INPUT CONTROL GAINS ***************************************************
C
```

```
C**** LIST OF CONTROL PARAMETERS ***********************************************
C
C**** PURE POSITION GAINS
C**** KXVEL,KYVEL,KTHVEL,KTVEL,KNVEL,KBVEL
C**** INTEGRAL RAMP FACTORS
C**** KXERI,KYERI,KTHERI,KTERI,KNERI,KBERI
C**** PURE INTEGRAL GAINS
C**** KITER,KIXER,KIYER,KITHER,KINER,KIBER
C**** INTEGRATOR LIMITS
C**** IXLIM,IYLIM,ITHLIM,ITLIM,INLIM,IBLIM
C
      OPEN (5,FILE='MULTIN.DAT',STATUS='OLD')
      READ (5,4) KXVEL,KYVEL,KTHVEL,KTVEL,KNVEL,KBVEL,
     @KXERI,KYERI,KTHERI,KTERI,KNERI,KBERI,KITER,KIXER,KIYER,KITHER,
     @KINER,KIBER,IXLIM,IYLIM,ITHLIM,ITLIM,INLIM,IBLIM
4     FORMAT ( F12.4,23(/F12.4))
      WRITE(*,3)
3     FORMAT ( '     CONTROLLER GAINS INPUT FROM MULTIN.DAT '//)
C
C
C**** INITIALISE INTERPOLATION COUNTER *****************************************
C
      NTRAJ = 1
      NINT = 1
C
C**** SET INTEGATORS' INITIAL CONDITIONS ***************************************
C
      XIER = 0.0
      YIER = 0.0
      THIER = 0.0
      TIER = 0.0
      NIER = 0.0
      BIER = 0.0
C
C**** SAMPLING TIME OF ROBOT CONTROLLER = TDELT *********************************
C
      TDELT = 0.01
C
C**** GET TRAJECTORY DATA ******************************************************
C
C**** OPEN INPUT FILE **********************************************************
C
      WRITE(*,50)
50    FORMAT('       DATA-FILE NAME   =   '\)
      READ(*,60) FNAME
60    FORMAT(A)
C
C**** OPEN DESIRED FILE ********************************************************
C
      OPEN(2,FILE=FNAME)
C
100   NTRAJ = NTRAJ + 1
C
      READ (2,200) MODE (NTRAJ)
C
200   FORMAT (A10)
```

```
C
C**** CHECK NUMBER OF TRAJECTORY CYCLES ***********************************
C
      IF ( NTRAJ .GE. 99 ) THEN
         WRITE (*,1790)
1790     FORMAT ( /,'          MAX. No. OF TRAJECTORIES EXCEEDED,  '
     @   'PROGRAM TERMINATES !!! ')
         MODE (NTRAJ) = 'END'
         GOTO 2000
      ENDIF
C
      IF ( MODE(NTRAJ) .EQ. 'END') THEN
C
         GOTO 1000
      ENDIF
C
      IF ( MODE (NTRAJ) .EQ. 'POSITION') THEN
         GOTO 500
      ENDIF
C
      IF ( MODE (NTRAJ) .EQ. 'FORCE') THEN
         GOTO 500
      ELSE
         WRITE (*,*) MODE(NTRAJ)
         WRITE (*,305)
305      FORMAT (//,'  INPUT ERROR ???     ',/,
     @   '   ERROR IN FILE, "A:TRAJ2.DAT", PROGRAM TERMINATED !!',/)
         GOTO 2000
      ENDIF
C
C
C**** SELECT COORDINATE SYSTEM OF TRAJECTORY, TOOL/GLOBAL OR FORCE
C**** AUTOMATIC CONTOUR TRACKING OR NOT ***********************************
C
500   CONTINUE
C
      READ (2,700) COORD (NTRAJ)
C
700   FORMAT (A10)
C
      IF ( COORD (NTRAJ) .EQ. 'GLOBAL' ) THEN
C
C**** CALL TO READ IN GLOBAL MOVEMENT DATA ********************************
C
         CALL DATAIN
C
C**** RETURN WITH UPDATED NTRAJ VALUE. GOTO READ IN NEXT TRAJECTORY **
C
         GOTO 100
      ENDIF
C
      IF ( COORD (NTRAJ) .EQ. 'TOOL' ) THEN
C
C**** CALL TO READ IN TOOL MOVEMENT DATA **********************************
C
         CALL DATAIN
```

```
C
C**** RETURN WITH UPDATED NTRAJ VALUE. GOTO READ IN NEXT TRAJECTORY **
C
         GOTO 100
       ENDIF
C
       IF ( COORD (NTRAJ) .EQ. 'NOTRACK' ) THEN
C
C**** CALL TO READ IN TOOL FORCE/MOVEMENT DATA ***************************
C
         CALL DATAIN
C
C**** RETURN WITH UPDATED NTRAJ VALUE. GOTO READ IN NEXT TRAJECTORY **
C
         GOTO 100
       ENDIF
C
       IF ( COORD (NTRAJ) .EQ. 'TRACK' ) THEN
C
C**** CALL TO READ IN TOOL FORCE/MOVEMENT DATA ***************************
C
         CALL DATAIN
C
C**** RETURN WITH UPDATED NTRAJ VALUE. GOTO READ IN NEXT TRAJECTORY **
C
         GOTO 100
       ELSE
         WRITE (*,*) MODE(NTRAJ)
         WRITE (*,805)
805      FORMAT (//,'  INPUT ERROR ???,   PROGRAM TERMINATED !!! ')
         GOTO 2000
       ENDIF
C
1000   CONTINUE
C
       CLOSE (2, STATUS = 'KEEP')
C
C**** WRITE OUT INPUTTED VALUES *****************************************
C
       DO 1800, NT = 2,(NTRAJ - 1)
C
       WRITE (*,1765) MODE(NT),COORD(NT)
1765   FORMAT (/,'    ',A10,/,'    ',A10)
       WRITE (*,1770) NT,
      @XCOORD(NT),YCOORD(NT),THCOORD(NT)
1770   FORMAT ( /,13,'     XCOORDS          YCOORDS          THCOORDS',/,
      @F12.5,2('    ',F12.5))
C
       WRITE (*,1780) TIME (NT)
1780   FORMAT (F12.5)
C
1800   CONTINUE
C
C**** NOW INTERPOLATE AND GENERATE COORDS. ******************************
C
C
```

```
1807   NSAMPLE = 0
       WRITE (*,1810)
1810   FORMAT (//,'      DATA INPUT COMPLETE..... PRESS RETURN '/
      @'      TO ENTER CONTROL ALGORITHM',\)
       READ (*,1820) L
1820   FORMAT ( I1)
C
C**** NUMBER OF INDIVIDUAL TRAJECTORIES = NTRAJ, SET LOOP COUNTER ****
C
C       WRITE (*,*) 'NTRAJ = ',NTRAJ
C
C**** INPUT ROBOTS CURRENT POSITION IMEDIATELY BEFORE ENTERING THE
C**** INTERPOLATION ROUTINE ***********************************************
C
          CALL WRDIN (XM)
          CALL WRDIN (YM)
          CALL WRDIN (THETAM)
C
C**** SET INITIAL STARTING POINT ******************************************
C
          XCOORD(1) = XM
          YCOORD(1) = YM
          THCOORD(1) = THETAM
C
C**** ENTER TRAJECTORY CONTROL ********************************************
C
       DO 1895, NLP = 2,NTRAJ
C
C       WRITE (*,*) '     NLP = ',NLP
C       WRITE (*,*) '     MODE = ',MODE(NLP)
C
       IF ( MODE(NLP) .EQ. 'POSITION' ) THEN
          IF ( COORD(NLP) .EQ. 'GLOBAL') THEN
             CALL INTGLOBAL
          ELSE
C
C**** 'TOOL' CONTROL ******************************************************
C
             CALL INTTOOL
          ENDIF
       ENDIF
C
       IF ( MODE(NLP) .EQ. 'FORCE' ) THEN
          IF ( COORD(NLP) .EQ. 'NOTRACK') THEN
             CALL INTFORCE
          ELSE
C
C**** CONTOUR TRACKING REQUIRED *******************************************
C
C             CALL TINTFORCE
          ENDIF
       ENDIF
C
C**** SPECIAL HOLD CONDITIONS WHEN 'END' IS ENCOUNTERED ***************
C
       IF ( MODE(NLP) .EQ. 'END' ) THEN
```

```
         IF ( MODE(NLP - 1) .EQ. 'POSITION' ) THEN
C
C**** HOLD PREVIOUS POSITION AT (NLP - 1) ********************************
C
C              CALL POSHOLD
C
C**** OUTPUT STOP STATEMENT TO COMP. 2 ***********************************
C
               CALL WRDOUT (100.0)
            ENDIF
C
         IF ( MODE(NLP - 1) .EQ. 'FORCE' ) THEN
C
C**** ENTER SOME ROUTINE TO HANDLE HOLD OF 'FORCE/POSITION/POLISHING'
C              CALL FORHOLD
C
C**** OUTPUT STOP STATEMENT TO COMP. 2 ***********************************
C
               CALL WRDOUT (100.0)
C
            ENDIF
         ENDIF
1895     CONTINUE
1900     CONTINUE
C
         WRITE (*,1905)
1905     FORMAT (//'       DO YOU WISH STORE DATA     Y/N   '\)
         READ (*,1910) ERROR(1)
1910     FORMAT (A10)
         IF ((ERROR(1) .EQ. 'Y').OR.(ERROR(1) .EQ. 'y')) THEN
            CALL STORE
            GOTO 1940
         ENDIF
         IF ((ERROR(1) .EQ. 'N').OR.(ERROR(1) .EQ. 'n')) THEN
            GOTO 1940
         ELSE
            WRITE (*,1915)
1915        FORMAT ('      INPUT ERROR ? ......Try again!!   '//)
            GOTO 1900
         ENDIF
C
1940     CONTINUE
C
         WRITE (*,1950)
1950     FORMAT (//'       DO YOU WISH TO REPEAT CYCLE      Y/N   '\)
         READ (*,1955) ERROR(1)
1955     FORMAT (A10)
         IF ((ERROR(1) .EQ. 'Y').OR.(ERROR(1) .EQ. 'y')) GOTO 1807
         IF ((ERROR(1) .EQ. 'N').OR.(ERROR(1) .EQ. 'n')) THEN
            GOTO 2000
         ELSE
            WRITE (*,1970)
1970        FORMAT ('      INPUT ERROR ? ......Try again!!   '//)
            GOTO 1940
         ENDIF
C
```

```
2000  CONTINUE
      WRITE (*,2010)
2010  FORMAT ( //'                    END' )
C
      STOP
      END



C**** SUBROUTINE INTERPOLATES BETWEEN THE INPUTTED GLOBAL COORDS. ****
C
      SUBROUTINE INTGLOBAL
C
C
      COMMON NCOUNT,NSTART,TDELT,XTRAJ,YTRAJ,THTRAJ,XCOORD,YCOORD,
     @THCOORD,TIME,MODE,COORD,NT,NINT,NTRAJ,NLP,XM,YM,THETAM,XERR,
     @YERR,THERR,KXVEL,KYVEL,KTHVEL,XDOT,YDOT,THDOT,KTVEL,KNVEL,KBVEL,
     @TM,NM,FNCORD,FNINC,FNINTL,DESFN,MESFN,SAMPLE,NSAMPLE,SWITCH
     @XERI,KXERI,YERI,KYERI,THERI,KTHERI,IXER,IYER,ITHER,KIXER,KIYER,
     @KITHER,TERI,KTERI,NERI,KNERI,BERI,KBERI,ITER,INER,IBER,KITER,
     @KINER,KIBER,IXLIM,IYLIM,ITHLIM,ITLIM,INLIM,IBLIM
C
      COMMON /A1/ XMES,YMES,THMES,FNMES,DFNMES,XDES,YDES,THDES,FNDES
C
      REAL XCOORD,YCOORD,THCOORD,XTRAJ,YTRAJ,THTRAJ,TDELT,
     @TIME,XINC,YINC,THINC,X,Y,TH,XM,YM,THETAM,XERR,YERR,THERR,
     @KXVEL,KYVEL,KTHVEL,XDOT,YDOT,THDOT,N,T,TINC,NINC,X1,Y1,DELXM,
     @DELYM,CTH1,STH1,DELTM,DELNM,TM,NM,TERR,NERR,TDOT,NDOT,NCOORD,
     @TCOORD,KTVEL,KNVEL,KBVEL,FNCORD,FNINC,FNINTL,DESFN,MESFN,
     @XMES,YMES,THMES,FNMES,DFNMES,XDES,YDES,THDES,FNDES
     @XERI,KXERI,YERI,KYERI,THERI,KTHERI,IXER,IYER,ITHER,KIXER,KIYER,
     @KITHER,TERI,KTERI,NERI,KNERI,BERI,KBERI,ITER,INER,IBER,KITER,
     @KINER,KIBER,IXLIM,IYLIM,ITHLIM,ITLIM,INLIM,IBLIM
C
      DIMENSION XTRAJ(4),YTRAJ(4),THTRAJ(4),MODE(99),
     @COORD(99),ERROR(99),XCOORD(99),YCOORD(99),THCOORD(99),
     @TIME(99),NCOORD(99),TCOORD(99),FNCORD(99),XMES(1000),
     @YMES(1000),THMES(1000),FNMES(1000),DFNMES(1000),XDES(10),
     @YDES(1000),THDES(1000),FNDES(1000)
C
      INTEGER*4 NT,NINT,NCOUNT,NSTART,NTRAJ,NLP,NSAMPLE
C
      INTEGER*2 SWITCH
C
      LOGICAL*2 MEMORY,SAMPLE
C
      CHARACTER*10 MODE,COORD,ERROR,FNAME
C
C
C**** CALCULATE THE NUMBER OF INCREMENTS ***********************************
C
      NINT = INT ( TIME(NLP)/TDELT )
C
C**** INTERPOLATION BEGINS *************************************************
C
```

```
C**** IF THE LAST ROUTINE WAS IN TOOL COORDS, THEN CURRENT POSITION **
C**** IS DIFFICULT TO PREDICT ACCURATELY, THEREFORE IT IS MEASURED ***
C**** SET LAST MEASUREMENT TO STARTING COORDS OF NEXT INTERPOLATION **
C**** THIS AVOIDS PROBLEMS OF RETURNING FROM AN UNKNOWN COORDINATE TO
C**** ACCURATE GLOBAL CONTROL **********************************************
C
C**** CHECKING TO SEE IF THE PREVIOUS TRAJECTORY WAS IN GLOBAL COORDS
C
      IF (COORD(NLP - 1) .EQ. 'GLOBAL') THEN
C
          XINC = (XCOORD(NLP) - XCOORD(NLP - 1))/NINT
          YINC = (YCOORD(NLP) - YCOORD(NLP - 1))/NINT
          THINC = (THCOORD(NLP) - THCOORD(NLP - 1))/NINT
C
C**** DESIRED COORDINATES OF TRAJECTORY ARE (X,Y,TH) ******************
C**** INITIAL COORDINATE IS LAST COORD. OF PREVIOUS CYCLE *************
C
          X = XCOORD(NLP - 1)
          Y = YCOORD(NLP - 1)
          TH = THCOORD(NLP - 1)
C
      ELSE
C
          XINC = (XCOORD(NLP) - XM)/NINT
          YINC = (YCOORD(NLP) - YM)/NINT
          THINC = (THCOORD(NLP) - THETAM)/NINT
C
C**** DESIRED COORDINATES OF TRAJECTORY ARE (X,Y,TH) ******************
C**** INITIAL COORDINATE IS LAST COORD. OF PREVIOUS CYCLE *************
C
          X = XM
          Y = YM
          TH = THETAM
C
      ENDIF
C
C**** INTERPOLATION LOOP ***************************************************
C
      DO 100, NCOUNT = 1, NINT
C
C**** SEND 'GLOBAL' COORDINATE INSTRUCTION TO COMP. 2 ******************
C
      CALL WRDOUT (10.0)
C
C**** INPUT CURRENT POSITION FROM COMP. 2 AS EARLY AS POSSIBLE TO NOT
C**** RISK ANY DELAY IN CONTROLLERS ALGORITHM ******************************
C
      CALL WRDIN (XM)
      CALL WRDIN (YM)
      CALL WRDIN (THETAM)
      CALL WRDIN (MESFN)
C
C      WRITE (*,*) '  GLOBAL CONT. X = ',X,' Y = ',Y,' TH = ',TH
C      WRITE (*,*) '  XM = ',XM,'  YM = ',YM,'  THETAM = ',THETAM
C
C**** CALCULATE DESIRED POSITION ***************************************
```

```
C
      X = X + XINC
      Y = Y + YINC
      TH = TH + THINC
C
C**** SAMPLE I/O PORT, CHECK SAMPLING EXTERNAL SWITCH 'ON' OR 'OFF' **
C**** LOOK FOR BINARY '00000101' FOR 'ON' ANYTHING ELSE 'OFF' ********
C
      SWITCH = INP(0529)
      IF ( SWITCH .EQ. 5) THEN
         SAMPLE = .TRUE.
      ELSE
         SAMPLE = .FALSE.
      ENDIF
C
C**** STORE DATA FROM COMPUTER 2 IN MEMORY IF 'SAMPLE' IS TRUE *******
C
      IF (SAMPLE .AND. (NSAMPLE .LT.1000 )) THEN
         NSAMPLE = NSAMPLE + 1
         XMES(NSAMPLE) = XM
         YMES(NSAMPLE) = YM
         THMES(NSAMPLE) = THETAM
         FNMES(NSAMPLE) = MESFN
         DFNMES(NSAMPLE) = FNMES(NSAMPLE) - FNMES(NSAMPLE - 1)
         XDES(NSAMPLE) = X
         YDES(NSAMPLE) = Y
         THDES(NSAMPLE) = TH
         FNDES(NSAMPLE) = DESFN
      ENDIF
C
C
C      WRITE (*,90) X,Y,TH
C90    FORMAT (3F12.5)
C      WRITE (*,*) '  GLOBAL CONT. X = ',X,' Y = ',Y,' TH = ',TH
C
C**** CALCALATE POSITION ERROR OF CURRENT POSITION *********************
C
      XERR = X - XM
      YERR = Y - YM
      THERR = TH - THETAM
C
C**** CALCULATE CONTROLLING VELOCITY OUTPUTS ****************************
C**** FOR PROPORTIONAL PLUS INTEGRAL ************************************
C
C
      XERI = XERR*KXERI
      YERI = YERR*KYERI
      THERI = THERR*KTHERI
C
      IXER = IXER + XERI
      IYER = IYER + YERI
      ITHER = ITHER + THERI
C
```

```
C**** CHECK THE INTEGRATION LIMITS ************************************
C
      IF (ABS(IXER).GE.IXLIM)  IXER = IXER - XERI
      IF (ABS(IYER).GE.IYLIM)  IYER = IYER - YERI
      IF (ABS(ITHER).GE.ITHLIM)  ITHER = ITHER - THERI
C
      XDOT = XERR*KXVEL + KIXER*IXER
      YDOT = YERR*KYVEL + KIYER*IYER
      THDOT = THERR*KTHVEL + KITHER*ITHER
C
C**** OUTPUT VELOCITY COMMANDS TO COMPUTER 2 ***************************
C
C     WRITE (*,*) '  XDOT = ',XDOT,'  YDOT = ',YDOT,'  THDOT = ',THDOT
C
      CALL WRDOUT(XDOT)
      CALL WRDOUT(YDOT)
      CALL WRDOUT(THDOT)
C
100   CONTINUE
C
C
1000  CONTINUE
C
      END



C**** SUBROUTINE INTERPOLATES BETWEEN THE INPUTTED TOOL COORDS. ******
C
      SUBROUTINE INTTOOL
C
C
C
      COMMON NCOUNT,NSTART,TDELT,XTRAJ,YTRAJ,THTRAJ,XCOORD,YCOORD,
     @THCOORD,TIME,MODE,COORD,NT,NINT,NTRAJ,NLP,XM,YM,THETAM,XERR,
     @YERR,THERR,KXVEL,KYVEL,KTHVEL,XDOT,YDOT,THDOT,KTVEL,KNVEL,KBVEL,
     @TM,NM,FNCORD,FNINC,FNINTL,DESFN,MESFN,SAMPLE,NSAMPLE,SWITCH
     @XERI,KXERI,YERI,KYERI,THERI,KTHERI,IXER,IYER,ITHER,KIXER,KIYER,
     @KITHER,TERI,KTERI,NERI,KNERI,BERI,KBERI,ITER,INER,IBER,KITER,
     @KINER,KIBER,IXLIM,IYLIM,ITHLIM,ITLIM,INLIM,IBLIM
C
      COMMON /A1/ XMES,YMES,THMES,FNMES,DFNMES,XDES,YDES,THDES,FNDES
C
      REAL XCOORD,YCOORD,THCOORD,XTRAJ,YTRAJ,THTRAJ,TDELT,
     @TIME,XINC,YINC,THINC,X,Y,TH,XM,YM,THETAM,XERR,YERR,THERR,
     @KXVEL,KYVEL,KTHVEL,XDOT,YDOT,THDOT,N,T,TINC,NINC,X1,Y1,DELXM,
     @DELYM,CTH1,STH1,DELTM,DELNM,TM,NM,TERR,NERR,TDOT,NDOT,NCOORD,
     @TCOORD,KTVEL,KNVEL,KBVEL,FNCORD,FNINC,FNINTL,DESFN,MESFN,
     @XMES,YMES,THMES,FNMES,DFNMES,XDES,YDES,THDES,FNDES
     @XERI,KXERI,YERI,KYERI,THERI,KTHERI,IXER,IYER,ITHER,KIXER,KIYER,
     @KITHER,TERI,KTERI,NERI,KNERI,BERI,KBERI,ITER,INER,IBER,KITER,
     @KINER,KIBER,IXLIM,IYLIM,ITHLIM,ITLIM,INLIM,IBLIM
C
      DIMENSION XTRAJ(4),YTRAJ(4),THTRAJ(4),MODE(99),
     @COORD(99),ERROR(99),XCOORD(99),YCOORD(99),THCOORD(99),
     @TIME(99),NCOORD(99),TCOORD(99),FNCORD(99),XMES(1000),
```

```
     @YMES(1000),THMES(1000),FNMES(1000),DFNMES(1000),XDES(1000),
     @YDES(1000),THDES(1000),FNDES(1000)
C
      INTEGER*4 NT,NINT,NCOUNT,NSTART,NTRAJ,NLP,NSAMPLE
C
      INTEGER*2 SWITCH
C
      LOGICAL*2 MEMORY,SAMPLE
C
      CHARACTER*10 MODE,COORD,ERROR,FNAME
C
C
C**** CALCULATE THE NUMBER OF INCREMENTS ********************************
C
      NINT = INT ( TIME(NLP)/TDELT )
C
C**** INTERPOLATION BEGINS ***********************************************
C**** SET INITIAL CONDITIONS**********************************************
C
      TCOORD(NLP) = XCOORD(NLP)
      NCOORD(NLP) = YCOORD(NLP)
C
      TINC = TCOORD(NLP)/NINT
      NINC = NCOORD(NLP)/NINT
      THINC = (THCOORD(NLP) - THETAM)/NINT
C
C**** DESIRED COORDINATES OF TRAJECTORY ARE (T,N,TH) ********************
C**** SETTING INITIAL VALUES *********************************************
C
      T = 0.0
      N = 0.0
      TH = THETAM
      TM = 0.0
      NM = 0.0
C
C**** INTERPOLATION LOOP *************************************************
C
      DO 100, NCOUNT = 1, NINT
C
C**** INITIALISE STARTING COORDINATE FOR MEASURED INCREMENT CALCS. ***
C**** SETTING TO PREVIOUSLY MEASURED COORDINATE, TO STARTING POINT ***
C
      X1 = XM
      Y1 = YM
      TH1 = THETAM
C
C**** SEND 'TOOL' COORDINATE INSTRUCTION TO COMP. 2 **********************
C
      CALL WRDOUT (20.0)
C
C**** INPUT CURRENT POSITION FROM COMP. 2 AS EARLY AS POSSIBLE TO NOT
C**** RISK ANY DELAY IN CONTROLLERS ALGORITHM ****************************
C
```

```
      CALL WRDIN (XM)
      CALL WRDIN (YM)
      CALL WRDIN (THETAM)
      CALL WRDIN (MESFN)
C
C      WRITE (*,*) '   TOOL CONT. T = ',T,' N = ',N,' TH = ',TH
C      WRITE (*,*) '  XM = ',XM,'  YM = ',YM,'  THETAM = ',THETAM
C
C**** CALCULATE DESIRED TOOL MOVEMENTS ************************************
C
      T = T + TINC
      N = N + NINC
      TH = TH + THINC
C
C      WRITE (*,90) T,N,TH
C90    FORMAT ('   TOOL CONTROL ',3F12.5)
C      WRITE (*,*) '  TOOL CONT. T = ',T,' N = ',N,' TH = ',TH
C
C**** CALCALATE POSITION ERROR OF CURRENT POSITION ***********************
C
      DELXM = XM - X1
      DELYM = YM - Y1
C
      CTH1 = COS(TH1)
      STH1 = SIN(TH1)
C
C**** CALCULATE CHANGES IN NORMAL AND TANGENTIAL POSITION *************
C**** NOTE THE PREVIOUS ANGLE VALUE IS USED, ANY CALCULATION ERROR IS
C**** MINIMISED WITH A LARGE NUMBER OF CYCLES *****************************
C
C**** TRANSFORMING GLOBAL TO TOOL AXES ************************************
C
      DELTM = DELXM*STH1 - DELYM*CTH1
      DELNM = DELXM*CTH1 + DELYM*STH1
C
C**** ADD TO INTEGRATE TOOL MOVEMENTS, WHEREVER THETA MAY WANDER *****
C
      TM = TM + DELTM
      NM = NM + DELNM
C
C**** SAMPLE I/O PORT, CHECK SAMPLING EXTERNAL SWITCH 'ON' OR 'OFF' **
C**** LOOK FOR BINARY '00000101' FOR 'ON' ANYTHING ELSE 'OFF' ********
C
      SWITCH = INP(0529)
      IF ( SWITCH .EQ. 5) THEN
         SAMPLE = .TRUE.
      ELSE
         SAMPLE = .FALSE.
      ENDIF
C
C**** STORE DATA FROM COMPUTER 2 IN MEMORY IF 'SAMPLE' IS TRUE *******
C
      IF (SAMPLE .AND. (NSAMPLE .LT.1000 )) THEN
         NSAMPLE = NSAMPLE + 1
         XMES(NSAMPLE) = TM
         YMES(NSAMPLE) = NM
```

```
            THMES(NSAMPLE) = THETAM
            FNMES(NSAMPLE) = MESFN
            DFNMES(NSAMPLE) = FNMES(NSAMPLE) - FNMES(NSAMPLE - 1)
            XDES(NSAMPLE) = T
            YDES(NSAMPLE) = N
            THDES(NSAMPLE) = TH
            FNDES(NSAMPLE) = DESFN
         ENDIF
C
C
C**** CALCULATE TOOL MOVEMENT ERRORS ************************************
C
         TERR = T - TM
         NERR = N - NM
         THERR = TH - THETAM
C
C**** CALCULATE CONTROLLING VELOCITY OUTPUTS ****************************
C**** FOR PROPORTIONAL PLUS INTEGRAL ************************************
C
         TERI = TERR*KTERI
         NERI = NERR*KNERI
         BERI = BERR*KBERI
C
         ITER = ITER + TERI
         INER = INER + NERI
         IBER = IBER + BERI
C
C**** CHECK THE INTEGRATION LIMITS **************************************
C
         IF (ABS(ITER).GE.ITLIM)  ITER = ITER - TERI
         IF (ABS(INER).GE.INLIM)  INER = INER - NERI
         IF (ABS(IBER).GE.IBLIM)  IBER = IBER - BERI
C
         TDOT = TERR*KTVEL + KITER*ITER
         NDOT = NERR*KNVEL + KIYNER*INER
         THDOT = THERR*KBVEL + KIBER*IBER
C
C**** OUTPUT VELOCITY COMMANDS TO COMPUTER 2 ****************************
C
C         WRITE (*,*) '  TDOT = ',TDOT,'  NDOT = ',NDOT,'  THDOT = ',THDOT
C
         CALL WRDOUT(TDOT)
         CALL WRDOUT(NDOT)
         CALL WRDOUT(THDOT)
C
100      CONTINUE
C
1000     CONTINUE
C
         END
```

```
C**** SUBROUTINE INTERPOLATES BETWEEN THE INPUTTED TOOL COORDS. IN THE
C**** TANGENTIAL & ANGULAR DIRECTIONS TOGETHER WITH NORMAL FORCE *****
C
      SUBROUTINE INTFORCE
C
C
      COMMON NCOUNT,NSTART,TDELT,XTRAJ,YTRAJ,THTRAJ,XCOORD,YCOORD,
     @THCOORD,TIME,MODE,COORD,NT,NINT,NTRAJ,NLP,XM,YM,THETAM,XERR,
     @YERR,THERR,KXVEL,KYVEL,KTHVEL,XDOT,YDOT,THDOT,KTVEL,KNVEL,KBVEL,
     @TM,NM,FNCORD,FNINC,FNINTL,DESFN,MESFN,SAMPLE,NSAMPLE,SWITCH
     @XERI,KXERI,YERI,KYERI,THERI,KTHERI,IXER,IYER,ITHER,KIXER,KIYER,
     @KITHER,TERI,KTERI,NERI,KNERI,BERI,KBERI,ITER,INER,IBER,KITER,
     @KINER,KIBER,IXLIM,IYLIM,ITHLIM,ITLIM,INLIM,IBLIM
C
      COMMON /A1/ XMES,YMES,THMES,FNMES,DFNMES,XDES,YDES,THDES,FNDES
C
      REAL XCOORD,YCOORD,THCOORD,XTRAJ,YTRAJ,THTRAJ,TDELT,
     @TIME,XINC,YINC,THINC,X,Y,TH,XM,YM,THETAM,XERR,YERR,THERR,
     @KXVEL,KYVEL,KTHVEL,XDOT,YDOT,THDOT,N,T,TINC,NINC,X1,Y1,DELXM,
     @DELYM,CTH1,STH1,DELTM,DELNM,TM,NM,TERR,NERR,TDOT,NDOT,NCOORD,
     @TCOORD,KTVEL,KNVEL,KBVEL,FNCORD,FNINC,FNINTL,DESFN,MESFN,
     @XMES,YMES,THMES,FNMES,DFNMES,XDES,YDES,THDES,FNDES
     @XERI,KXERI,YERI,KYERI,THERI,KTHERI,IXER,IYER,ITHER,KIXER,KIYER,
     @KITHER,TERI,KTERI,NERI,KNERI,BERI,KBERI,ITER,INER,IBER,KITER,
     @KINER,KIBER,IXLIM,IYLIM,ITHLIM,ITLIM,INLIM,IBLIM
C
      DIMENSION XTRAJ(4),YTRAJ(4),THTRAJ(4),MODE(99),
     @COORD(99),ERROR(99),XCOORD(99),YCOORD(99),THCOORD(99),
     @TIME(99),NCOORD(99),TCOORD(99),FNCORD(99),XMES(1000),
     @YMES(1000),THMES(1000),FNMES(1000),DFNMES(1000),XDES(1000),
     @YDES(1000),THDES(1000),FNDES(1000)
C
      INTEGER*4 NT,NINT,NCOUNT,NSTART,NTRAJ,NLP,NSAMPLE
C
      INTEGER*2 SWITCH
C
      LOGICAL*2 MEMORY,SAMPLE
C
      CHARACTER*10 MODE,COORD,ERROR,FNAME
C
C
C**** CALCULATE THE NUMBER OF INCREMENTS ********************************
C
      NINT = INT ( TIME(NLP)/TDELT )
C
C**** INTERPOLATION BEGINS ***********************************************
C**** SET INITIAL CONDITIONS**********************************************
C
      TCOORD(NLP) = XCOORD(NLP)
      FNCORD(NLP) = YCOORD(NLP)
C
C**** CHECK CONDITION TO ASSIGN A VALUE TO INITIAL FORCE ***************
C
      IF ( MODE(NLP - 1) .EQ. 'FORCE' ) THEN
         FNINTL = FNCORD (NLP - 1)
      ELSE
```

```
         FNINTL = 0.0
      ENDIF
C
      TINC = TCOORD(NLP)/NINT
      FNINC = (FNCORD(NLP) - FNINTL)/NINT
      THINC = (THCOORD(NLP) - THETAM)/NINT
C
C**** DESIRED COORDINATES OF TRAJECTORY ARE (T,FN,TH) ****************
C**** SETTING INITIAL VALUES ******************************************
C
      T = 0.0
      N = 0.0
      TH = THETAM
      TM = 0.0
      NM = 0.0
C
C**** INTERPOLATION LOOP **********************************************
C
      DO 100, NCOUNT = 1, NINT
C
C**** INITIALISE STARTING COORDINATE FOR MEASURED INCREMENT CALCS. ***
C**** SETTING TO PREVIOUSLY MEASURED COORDINATE, TO STARTING POINT ***
C
      X1 = XM
      Y1 = YM
      TH1 = THETAM
C
C**** SEND 'TOOL/FORCE' MODE INSTRUCTION TO COMP. 2 ********************
C
      CALL WRDOUT (30.0)
C
C**** INPUT CURRENT POSITION & FORCE DATA FROM COMP. 2 AS EARLY AS
C**** POSSIBLE TO NOT RISK ANY DELAY IN CONTROLLERS ALGORITHM ********
C
      CALL WRDIN (XM)
      CALL WRDIN (YM)
      CALL WRDIN (THETAM)
      CALL WRDIN (MESFN)
C
C      WRITE (*,*) '   TOOL CONT. T = ',T,' N = ',N,' TH = ',TH
C      WRITE (*,*) '  XM = ',XM,'  YM = ',YM,'  THETAM = ',THETAM
C
C**** CALCULATE DESIRED TOOL FORCES & MOVEMENTS ************************
C
      T = T + TINC
      FNINTL = FNINTL + FNINC
      TH = TH + THINC
      DESFN = FNINTL
C
C      WRITE (*,90) T,DESFN,TH
C90    FORMAT ('   TOOL CONTROL ',3F12.5)
C      WRITE (*,*) '  TOOL CONT. T = ',T,' DESFN = ',DESFN,' TH = ',TH
C
C**** CALCALATE POSITION ERROR OF CURRENT POSITION **********************
C
      DELXM = XM - X1
```

```
      DELYM = YM - Y1
C
      CTH1 = COS(TH1)
      STH1 = SIN(TH1)
C
C**** CALCULATE CHANGES IN NORMAL AND TANGENTIAL POSITION ************
C**** NOTE THE PREVIOUS ANGLE VALUE IS USED, ANY CALCULATION ERROR IS
C**** MINIMISED WITH A LARGE NUMBER OF CYCLES ***************************
C
C**** TRANSFORMING GLOBAL TO TOOL AXES *********************************
C
      DELTM = DELXM*STH1 - DELYM*CTH1
      DELNM = DELXM*CTH1 + DELYM*STH1
C
C**** ADD TO INTEGRATE TOOL MOVEMENTS, WHEREVER THETA MAY WANDER *****
C
      TM = TM + DELTM
      NM = NM + DELNM
C
C**** SAMPLE I/O PORT, CHECK SAMPLING EXTERNAL SWITCH 'ON' OR 'OFF' **
C**** LOOK FOR BINARY '00000101' FOR 'ON' ANYTHING ELSE 'OFF' ********
C
      SWITCH = INP(0529)
      IF ( SWITCH .EQ. 5) THEN
         SAMPLE = .TRUE.
      ELSE
         SAMPLE = .FALSE.
      ENDIF
C
C**** STORE DATA FROM COMPUTER 2 IN MEMORY IF 'SAMPLE' IS TRUE *******
C
      IF (SAMPLE .AND. (NSAMPLE .LT. 1000 )) THEN
         NSAMPLE = NSAMPLE + 1
         XMES(NSAMPLE) = TM
         YMES(NSAMPLE) = NM
         THMES(NSAMPLE) = THETAM
         FNMES(NSAMPLE) = MESFN
         DFNMES(NSAMPLE) = FNMES(NSAMPLE) - FNMES(NSAMPLE - 1)
         XDES(NSAMPLE) = T
         YDES(NSAMPLE) = N
         THDES(NSAMPLE) = TH
         FNDES(NSAMPLE) = DESFN
      ENDIF
C
C
C**** CALCULATE TOOL MOVEMENT ERRORS ***********************************
C
      TERR = T - TM
      THERR = TH - THETAM
C
C**** CALCULATE CONTROLLING VELOCITY OUTPUTS ****************************
C**** FOR PROPORTIONAL PLUS INTEGRAL ************************************
C
      TERI = TERR*KTERI
      BERI = BERR*KBERI
C
```

```
      ITER = ITER + TERI
      IBER = IBER + BERI
C
C**** CHECK THE INTEGRATION LIMITS ************************************
C
      IF (ABS(ITER).GE.ITLIM)  ITER = ITER - TERI
      IF (ABS(IBER).GE.IBLIM)  IBER = IBER - BERI
C
      TDOT = TERR*KTVEL + KITER*ITER
      THDOT = THERR*KBVEL + KIBER*IBER
C
C**** OUTPUT VELOCITY & FORCE COMMANDS TO COMPUTER 2 ******************
C
C      WRITE (*,*) '  TDOT = ',TDOT,'  DESFN = ',DESFN,'  THDOT = ',
C      @THDOT
C
      CALL WRDOUT(TDOT)
      CALL WRDOUT(DESFN)
      CALL WRDOUT(THDOT)
C
100   CONTINUE
C
1000  CONTINUE
C
      END
```

```
C********************* SUBROUTINE DATAIN ***********************************
C**** THIS ROUTINE INPUTS DATA TO BE USED TO DRIVE THE ROBOT IN TOOL
C**** OR GLOBAL COORDINATES *************************************************
C
      SUBROUTINE DATAIN
C
C
      COMMON NCOUNT,NSTART,TDELT,XTRAJ,YTRAJ,THTRAJ,XCOORD,YCOORD,
     @THCOORD,TIME,MODE,COORD,NT,NINT,NTRAJ,NLP,XM,YM,THETAM,XERR,
     @YERR,THERR,KXVEL,KYVEL,KTHVEL,XDOT,YDOT,THDOT,KTVEL,KNVEL,KBVEL,
     @TM,NM,FNCORD,FNINC,FNINTL,DESFN,MESFN,SAMPLE,NSAMPLE,SWITCH
     @XERI,KXERI,YERI,KYERI,THERI,KTHERI,IXER,IYER,ITHER,KIXER,KIYER,
     @KITHER,TERI,KTERI,NERI,KNERI,BERI,KBERI,ITER,INER,IBER,KITER,
     @KINER,KIBER,IXLIM,IYLIM,ITHLIM,ITLIM,INLIM,IBLIM
C
      COMMON /A1/ XMES,YMES,THMES,FNMES,DFNMES,XDES,YDES,THDES,FNDES
C
      REAL XCOORD,YCOORD,THCOORD,XTRAJ,YTRAJ,THTRAJ,TDELT,
     @TIME,XINC,YINC,THINC,X,Y,TH,XM,YM,THETAM,XERR,YERR,THERR,
     @KXVEL,KYVEL,KTHVEL,XDOT,YDOT,THDOT,N,T,TINC,NINC,X1,Y1,DELXM,
     @DELYM,CTH1,STH1,DELTM,DELNM,TM,NM,TERR,NERR,TDOT,NDOT,NCOORD,
     @TCOORD,KTVEL,KNVEL,KBVEL,FNCORD,FNINC,FNINTL,DESFN,MESFN,
     @XMES,YMES,THMES,FNMES,DFNMES,XDES,YDES,THDES,FNDES
     @XERI,KXERI,YERI,KYERI,THERI,KTHERI,IXER,IYER,ITHER,KIXER,KIYER,
     @KITHER,TERI,KTERI,NERI,KNERI,BERI,KBERI,ITER,INER,IBER,KITER,
     @KINER,KIBER,IXLIM,IYLIM,ITHLIM,ITLIM,INLIM,IBLIM
C
      DIMENSION XTRAJ(4),YTRAJ(4),THTRAJ(4),MODE(99),
```

```
     @COORD(99),ERROR(99),XCOORD(99),YCOORD(99),THCOORD(99),
     @TIME(99),NCOORD(99),TCOORD(99),FNCORD(99),XMES(1000),
     @YMES(1000),THMES(1000),FNMES(1000),DFNMES(1000),XDES(1000),
     @YDES(1000),THDES(1000),FNDES(1000)
C
      INTEGER*4 NT,NINT,NCOUNT,NSTART,NTRAJ,NLP,NSAMPLE
C
      INTEGER*2 SWITCH
C
      LOGICAL*2 MEMORY,SAMPLE
C
      CHARACTER*10 MODE,COORD,ERROR,FNAME
C
C
C**** INPUT FIRST COORDINATE OF TRAJECTORY. THIS SHOULD BE THE ROBOT'S
C**** STARTING POSITION OTHERWISE THERE WILL BE UNCONTROLLED MOVEMENT
C**** TOWARDS THE PLANNED TRAJECTORY ************************************
C
C
C**** No. OF TRAJECTORYS COUNTER 'NTRAJ' ********************************
C
C************* GO AND INPUT NEXT DESTINATION COORDINATE ***************
C
1120  READ (2,1200) XCOORD (NTRAJ),YCOORD (NTRAJ),
     @THCOORD (NTRAJ)
1200  FORMAT (3F12.5)
C
C**** INPUT TIME OF POINT TO POINT MOVEMENT *****************************
C
C1720  WRITE (*,1750)
C1750  FORMAT(//,'   INPUT POINT TO POINT MOVEMENT TIME, "T"secs. = '\)
C
1755  READ (2,1760) TIME (NTRAJ)
1760  FORMAT (F12.5)
C
C**** RETURN AND LOOK FOR NEXT CONTROL WORD *********************************
C
      END


C************************ SUBROUTINE STORE ***********************************
C**** STORES MEMORY DATA TO HARD DISK C:\DATA\SAMPLE.DAT FILE*********
C
      SUBROUTINE STORE
C
C
      COMMON NCOUNT,NSTART,TDELT,XTRAJ,YTRAJ,THTRAJ,XCOORD,YCOORD,
     @THCOORD,TIME,MODE,COORD,NT,NINT,NTRAJ,NLP,XM,YM,THETAM,XERR,
     @YERR,THERR,KXVEL,KYVEL,KTHVEL,XDOT,YDOT,THDOT,KTVEL,KNVEL,KBVEL,
     @TM,NM,FNCORD,FNINC,FNINTL,DESFN,MESFN,SAMPLE,NSAMPLE,SWITCH
     @XERI,KXERI,YERI,KYERI,THERI,KTHERI,IXER,IYER,ITHER,KIXER,KIYER,
     @KITHER,TERI,KTERI,NERI,KNERI,BERI,KBERI,ITER,INER,IBER,KITER,
     @KINER,KIBER,IXLIM,IYLIM,ITHLIM,ITLIM,INLIM,IBLIM
C
      COMMON /A1/ XMES,YMES,THMES,FNMES,DFNMES,XDES,YDES,THDES,FNDES
C
```

```
      REAL XCOORD,YCOORD,THCOORD,XTRAJ,YTRAJ,THTRAJ,TDELT,
     @TIME,XINC,YINC,THINC,X,Y,TH,XM,YM,THETAM,XERR,YERR,THERR,
     @KXVEL,KYVEL,KTHVEL,XDOT,YDOT,THDOT,N,T,TINC,NINC,X1,Y1,DELXM,
     @DELYM,CTH1,STH1,DELTM,DELNM,TM,NM,TERR,NERR,TDOT,NDOT,NCOORD,
     @TCOORD,KTVEL,KNVEL,KBVEL,FNCORD,FNINC,FNINTL,DESFN,MESFN,
     @XMES,YMES,THMES,FNMES,DFNMES,XDES,YDES,THDES,FNDES
     @XERI,KXERI,YERI,KYERI,THERI,KTHERI,IXER,IYER,ITHER,KIXER,KIYER,
     @KITHER,TERI,KTERI,NERI,KNERI,BERI,KBERI,ITER,INER,IBER,KITER,
     @KINER,KIBER,IXLIM,IYLIM,ITHLIM,ITLIM,INLIM,IBLIM
C
      DIMENSION XTRAJ(4),YTRAJ(4),THTRAJ(4),MODE(99),
     @COORD(99),ERROR(99),XCOORD(99),YCOORD(99),THCOORD(99),
     @TIME(99),NCOORD(99),TCOORD(99),FNCORD(99),XMES(1000),
     @YMES(1000),THMES(1000),FNMES(1000),DFNMES(1000),XDES(1000),
     @YDES(1000),THDES(1000),FNDES(1000)
C
      INTEGER*4 NT,NINT,NCOUNT,NSTART,NTRAJ,NLP,NSAMPLE,K1
C
      INTEGER*2 SWITCH
C
      LOGICAL*2 MEMORY,SAMPLE
C
      CHARACTER*10 MODE,COORD,ERROR,FNAME
C
C**** WRITE NUMBER OF DATA STORAGE CYCLES **********************************
C
      WRITE (*,*) '   No. DATA STORAGE CYCLES = ',NSAMPLE
C
C**** CREATE FILE TO STORE DATA, 'SAMPLE.DAT' ******************************
C
      OPEN (3,FILE = 'C:SAMPLE.DAT',STATUS = 'NEW')
C
      DO 600  K1 = 1,NSAMPLE
C
C**** PARAMETERS WRITEN TO FILE ARE :
C
C**** XMES(1000),YMES(1000),THMES(1000),FNMES(1000),DFNMES(1000),
C**** XDES(1000),YDES(1000),THDES(1000),FNDES(1000)
C
C
      WRITE (3,10) XMES(K1),YMES(K1),THMES(K1),
     @XDES(K1),YDES(K1),THDES(K1),FNDES(K1),FNMES(K1),DFNMES(K1)
10    FORMAT(9(E9.3))
C
600   CONTINUE
C
C**** CLOSE FILE ***********************************************************
C
      CLOSE (3, STATUS = 'KEEP')
C
      WRITE (*,700)
700   FORMAT (//,'      WRITING TO FILE C:SAMPLE.DAT COMPLETE')
C
      END
```

APPENDIX D2

CONTROLLER PROGRAM 'MULTSAMP'

```
C ******************** MAIN PROGRAM MULTSAMP **********************************
C ******* SAMPLES FORCE DATA UPTO 1000 CYCLES ****************************
C ***** CONTROLS ROBOT ARM WITH VELOCITY ********************************
C ***** CONTROL IN WORLD COORDS. ALINED WITH CARTESIAN COORDS ********
C ***** CONTROL IN TOOL COORDS. ALINED WITH TOOL FRAME ****************
C ***** ALSO FORCE/VELOCITY CONTROL ALINED WITH TOOL COORDS. *********
C
      PROGRAM MULTSAMP
C
C
      COMMON D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,GEOROR,L1INTL,L2INTL,BEINTL,BVELBACK,
     @ERRLIM,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,FINISH,
     @XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,INCRIM,
     @IPORT1,IPORT2,IPORT3,SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,L1M,
     @L2M,BETAM,INVJ(3,3),F(3,3),J(3,3),NDOT1,MTDOT,MNDOT,MTHDOT,LEN1,
     @LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,EBTDOT,EF1,
     @EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,LEN2INTL,
     @BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,XJM,YJM,KFN,DFN,
     @IFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,
     @XFINISH,YFINISH,THFINISH,HOLD,POSITION,FORCE,SN,ST,SB,SELECT,
     @MODES,WORLD,LOCAL,KVX,KVY,KVTH,KEX,KEY,KETH,KET,KEN,KEB,KEFN,
     @NSAMPLE,SAMP,SAMPLE,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,IMPACT
C
      COMMON /ASTORE/ SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2
C
      COMMON /FNFEEDBAK/ MESFN,KVFN,FNFBAK
C
      REAL D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,L1C,L2C,BETAC,XJC,YJC,XD,LC,L1SQ,L2SQ,
     @BVELBACK,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,EL1,EL2,EBETA,
     @VL1,VL2,VBETA,XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,
     @SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,INVJ,F,J,NDOT1,MTDOT,MNDOT,
     @MTHDOT,LEN1,LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,
     @EBTDOT,EF1,EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,
     @LEN2INTL,BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,ML1INC,
     @ML2INC,XJM,YJM,KFN,DFN,IFN,MESFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,
     @KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,MODES,KVX,KVY,KVTH,KEX,KEY,KETH,
     @KET,KEN,KEB,KEFN,SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,KVFN,
     @FNFBAK
C
      INTEGER*2 IL1C,IL2C,IBETAC,L1INTL,L2INTL,BEINTL,L1M,
     @L2M,BETAM,L1VEL,L2VEL,BETVEL,ERRLIM,EL1P,EL2P,EBEP,INCRIM,
     @IPORT1,IPORT2,IPORT3,ST,SN,SB,SELECT,I,IADCON,IADC,CHANNEL
C
      INTEGER NSAMPLE
```

```
C
      LOGICAL*2 XFINISH,YFINISH,THFINISH,FINISH,GEOROR,HOLD,POSITION,
     @FORCE,WORLD,LOCAL,SAMP,SAMPLE,IMPACT
C
      DIMENSION SFNCONTROL(1005),SERRFN(1005),SIFN(1005),SMNINC(1005),
     @SEL1DOT(1005),SEL2DOT(1005),SEBTDOT(1005),SEF1(1005),SEF2(1005),
     @SEM2(1005)
C
      CHARACTER*10 ERROR(1)
C
      R = 0.265
      D = 0.70
      D1 = 0.35
      LMAX = 1.08
      LMIN = 0.48
      BETMIN = -0.3
      BETMAX = 2.79
C
C**** OUTPUT ZERO VOLTS, REMOVES ANY INITIAL VOLTAGES ******************
C
      CALL VOLTOUT (1,0.0)
      CALL VOLTOUT (2,0.0)
      CALL VOLTOUT (3,0.0)
C
C**** INPUT CONTROLLER GAINS ***************************************************
C
C**** TOOL VELOCITY CONTROL FEEDBACK PARAMS.
C**** KVT =    10.0
C**** KVN =    10.0
C**** KVB =    10.0
C**** TOOL FORCE CONTROL FEEDBACK PARAMS.
C**** KFN =    1.0
C**** WORLD VELOCITY CONTROL FEEDBACK PARAMS.
C**** KVX =    5.0
C**** KVY =    5.0
C**** KVTH =   5.0
C**** WORLD COORD. ERROR GAINS
C**** KEX =    1.0
C**** KEY =    1.0
C**** KETH =   1.0
C**** TOOL COORD. ERROR GAINS
C**** KET =    1.0
C**** KEN =    1.0
C**** KEB =    1.0
C**** FORCE ERROR GAIN
C**** KEFN =   1.0
C**** JOINT CONTROL GAINS, POSITION CONTROL
C**** VGAIN = 1000.0
C**** WVGAIN = 300.0
C**** JOINT CONTROL GAINS, FORCE CONTROL
C**** FGAIN =  0.05
C**** WFGAIN =  1.0
C**** FORCE CONTROLLER PARAMS.
C**** FORCE INTEGRATION LIMITS
C**** IFNLIM =  0.0
C**** FORCE ERROR PROPORTIONAL GAIN
```

```
C**** KPFN =      0.3
C**** FORCE INTEGRAL GAIN
C**** KIFN =      0.0
C**** FORCE RATE GAIN OF ERROR
C**** KDFN =      0.0
C**** VELOCITY FEEDBACK IN NORMAL DIRECTION, SIMILAR EFFECT TO KDFN
C**** KNINC =     0000.0
C**** INTEGRAL RAMP CONTROL
C**** KIER =      0.0
C**** FEDFORWARD FACTOR
C**** FNFWD =     0.0
C**** FNFBAK = 0.035
C
      OPEN (2,FILE='MULTI.DAT',STATUS='OLD')
      READ (2,4) KVT,KVN,KVB,KFN,KVX,KVY,KVTH,KEX,KEY,KETH,KET,KEN,KEB
     a,KEFN,VGAIN,WVGAIN,FGAIN,WFGAIN,IFNLIM,KPFN,KIFN,KDFN,KNINC,KIER
     a,FNFWD,FNFBAK
4     FORMAT ( F12.4,25(/F12.4))
      WRITE(*,3)
3     FORMAT ( '    CONTROLLER GAINS INPUT FROM MULTI.DAT '//)
C
100   CONTINUE
C
5     WRITE(*,10)
10    FORMAT (/'           PRESS RETURN TO START CONTROL !!!!  '\)
      READ (*,15) I10
15    FORMAT (I2)
C
C**** CALCULATE INITIAL VALUES OF GEOMETRY FOR CURRENT ROBOT POSITION
C
      IF (GEOROR) THEN
         GOTO 110
      ELSE
         HOLD = .FALSE.
         GEOROR = .FALSE.
         SAMP = .FALSE.
         NSAMPLE = 0
         IFN = 0.0
         IMPACT = .FALSE.
      ENDIF
C
110   CALL GEOMETRY
      LEN1INTL = LEN1
      LEN2INTL = LEN2
      BETAINTL = BETA
C
C**** CHECK GEOMETRICAL ERROR CONDITION BEFORE OUTPUTTING DATA
C**** THIS PREVENTS HANDSHAKE ERROR ***************************************
C
      IF (GEOROR) THEN
         CALL WRDIN (SCRAP)
         CALL WRDIN (SCRAP)
         CALL WRDIN (SCRAP)
         GEOROR = .FALSE.
         GOTO 600
      ENDIF
```

```
C
C**** OUTPUT STARTING COORDINATES TO COMPUTER 1 ************************
C
      CALL WRDOUT (XM)
      CALL WRDOUT (YM)
      CALL WRDOUT (THETAM)
C
C**** CHECK CURRENT POSITION TO BE WITHIN GEOMETRICAL LIMITS *********
C
      CALL LIMIT(LEN1,LEN2,BETA)
      IF (GEOROR) THEN
      CALL VOLTOUT (1,0)
      CALL VOLTOUT (2,0)
      CALL VOLTOUT (3,0)
C
500   WRITE (*,501) XM,YM,THETAM,LEN1,LEN2,BETA
501   FORMAT ('   GEOMETRICAL LIMIT EXCEEDED @   ',/
     @'  WORLD COORDS.  (',F8.3,',',F8.3,',',F8.3,')',/
     @'  JOINT COODSS.  (',F8.3,',',F8.3,',',F8.3,')')
C
      WRITE (*,300)
300   FORMAT ( '  GEOMETRICAL LIMIT EXCEEDED, AT INITIAL POSITION')
      GOTO 1000
      ENDIF
C
C
C**** CALL VELOCITY CONTROL ROUTINE ***********************************
C
600   CONTINUE
C
      CALL CONTROL
C
      IF (GEOROR) THEN
         GOTO 100
      ELSE
         IF (SAMP) CALL STORE
         GOTO 100
      ENDIF
C
1000  CONTINUE
C
      WRITE(*,605)
605   FORMAT( '       END HERE')
C
      STOP
      END
```

```
C********************* SUBROUTINE CONTROL ***********************************
C****** CONTROLS ROBOT IN VELOCITY & FORCE IN TOOL COORDS. ************
C**** CONTROLS IN VELOCITY ONLY IN WORLD COORDS ***********************
C**** TAKES POSITION AND FORCE SINGALS DIRECTLY FROM COMP. 1 **********
C
      SUBROUTINE CONTROL
C
C
      COMMON D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,GEOROR,L1INTL,L2INTL,BEINTL,BVELBACK,
     @ERRLIM,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,FINISH,
     @XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,INCRIM,
     @IPORT1,IPORT2,IPORT3,SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,L1M,
     @L2M,BETAM,INVJ(3,3),F(3,3),J(3,3),NDOT1,MTDOT,MNDOT,MTHDOT,LEN1,
     @LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,EBTDOT,EF1,
     @EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,LEN2INTL,
     @BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,XJM,YJM,KFN,DFN,
     @IFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,
     @XFINISH,YFINISH,THFINISH,HOLD,POSITION,FORCE,SN,ST,SB,SELECT,
     @MODES,WORLD,LOCAL,KVX,KVY,KVTH,KEX,KEY,KETH,KET,KEN,KEB,KEFN,
     @NSAMPLE,SAMP,SAMPLE,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,IMPACT
C
      COMMON /ASTORE/ SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2
C
      COMMON /FNFEEDBAK/ MESFN,KVFN,FNFBAK
C
      REAL D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,L1C,L2C,BETAC,XJC,YJC,XD,LC,L1SQ,L2SQ,
     @BVELBACK,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,EL1,EL2,EBETA,
     @VL1,VL2,VBETA,XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,
     @SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,INVJ,F,J,NDOT1,MTDOT,MNDOT,
     @MTHDOT,LEN1,LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,
     @EBTDOT,EF1,EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,
     @LEN2INTL,BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,ML1INC,
     @ML2INC,XJM,YJM,KFN,DFN,IFN,MESFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,
     @KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,MODES,KVX,KVY,KVTH,KEX,KEY,KETH,
     @KET,KEN,KEB,KEFN,SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,KVFN,
     @FNFBAK
C
      INTEGER*2 IL1C,IL2C,IBETAC,L1INTL,L2INTL,BEINTL,L1M,
     @L2M,BETAM,L1VEL,L2VEL,BETVEL,ERRLIM,EL1P,EL2P,EBEP,INCRIM,
     @IPORT1,IPORT2,IPORT3,ST,SN,SB,SELECT,I,IADCON,IADC,CHANNEL
C
      INTEGER NSAMPLE
C
      LOGICAL*2 XFINISH,YFINISH,THFINISH,FINISH,GEOROR,HOLD,POSITION,
     @FORCE,WORLD,LOCAL,SAMP,SAMPLE,IMPACT
C
      DIMENSION SFNCONTROL(1005),SERRFN(1005),SIFN(1005),SMNINC(1005),
     @SEL1DOT(1005),SEL2DOT(1005),SEBTDOT(1005),SEF1(1005),SEF2(1005),
     @SEM2(1005)
C
      CHARACTER*10 ERROR(1)
```

```
C
C
C**** OUTPUT ZERO VOLTS, ELIMINATES ANY RANDOM VOLTAGES ***************
C
      CALL VOLTOUT (1,0.0)
      CALL VOLTOUT (2,0.0)
      CALL VOLTOUT (3,0.0)
C
C**** MEASURES CURRENT POSITION AND EVALUATES GEOMETRICAL PARAMMETERS
C
100   CALL GEOMETRY
C
C**** INPUT ANALOGUE FORCE MEASUEMENT ***********************************
C
      CALL ADCFORCE
C
C**** INPUT STRATEGY DECISION FROM COMPUTER ONE CHOOSE : WORLD OR
C**** LOCAL COORDS AND FORCE/POSITION CONTROL ****************************
C
      CALL WRDIN (MODES)
C
C      WRITE (*,*) '       MODE No. =',MODES
C
C**** CHECK TO STOP *****************************************************
C
      IF ( MODES .EQ. 100.0 ) THEN
          CALL VOLTOUT (1,0)
          CALL VOLTOUT (2,0)
          CALL VOLTOUT (3,0)
C
      WRITE (*,*) '            HALT INSTRUCTION RECEIVED !!!!!!!!!! '
C
          GOTO 1000
C
      ENDIF
C
C**** OUTPUT CURRENT POSITION TO COMP. 1 *********************************
C
      CALL WRDOUT (XM)
      CALL WRDOUT (YM)
      CALL WRDOUT (THETAM)
      CALL WRDOUT (MESFN)
C
C**** CHECK MODES AND ASSIGN LOGIC VALUES ********************************
C
      IF ( MODES .EQ. 10.0 ) THEN
         WORLD = .TRUE.
         LOCAL = .FALSE.
         FORCE = .FALSE.
      ENDIF
C
      IF ( MODES .EQ. 20.0 ) THEN
         WORLD = .FALSE.
         LOCAL = .TRUE.
         FORCE = .FALSE.
      ENDIF
```

```
C
      IF ( MODES .EQ. 30.0 ) THEN
         WORLD = .FALSE.
         LOCAL = .TRUE.
         FORCE = .TRUE.
      ENDIF
C
C**** CHECK CURRENT POSITION TO BE WITHIN GEOMETRICAL LIMITS **********
C
      CALL LIMIT(LEN1,LEN2,BETA)
      IF (GEOROR) THEN
       CALL VOLTOUT (1,0)
       CALL VOLTOUT (2,0)
       CALL VOLTOUT (3,0)
C
500    WRITE (*,501) XM,YM,THETAM,LEN1,LEN2,BETA
501    FORMAT ('   GEOMETRICAL LIMIT EXCEEDED @    ',/
     @'  WORLD COORDS.  (',F8.3,',',F8.3,',',F8.3,')',/
     @'  JOINT COODSS.  (',F8.3,',',F8.3,',',F8.3,')')
C
       WRITE (*,300)
300    FORMAT ( '  GEOMETRICAL LIMIT EXCEEDED, CONTROL RETURNED TO MAIN
     @CALLING PROGRAM ')
       GOTO 1000
       ENDIF
C
C**** SELECT & EVALUATE WORLD OR TOOL TRANSFORMION MATRICES **********
C
      IF ( WORLD ) THEN
         CALL WMATRICES
C
C**** FEED BACK CURRENT POSITION BY USING MEASURED JOINT COORDS.
C**** TRANSFORMED INTO WORLD COORDS. BY USE OF JACOBIAN. **************
C
         CALL MEASINC
C
C**** INPUT DESIRED VELOCITY COMMANDS FROM COMPUTER 1 ******************
C
         CALL WRDIN (DTINC)
         CALL WRDIN (DNINC)
         CALL WRDIN (DBTINC)
         DESFN = 0.0
C
C       WRITE (*,110) DTINC,DNINC,DBTINC
110    FORMAT(' DTINC = ',F12.5,'  DNINC = ',F12.5,'  DBTINC = ',F12.5)
C
C**** CALCULATE LOCAL POSITION ERROR DEMANDS ******************************
C**** KV REPRESENTS RATE FEEDBACK, NONLINEAR PROBLEMS MAY ARISE IF KV
C**** IS TOO LARGE ********************************************************
C
C**** VELOCITY DEMANDS ARE OPPOSITE TO FORCE DEMANDS ********************
C
         ERTDOT = -(DTINC - KVX*MTINC)*KEX
         ERNDOT = -(DNINC - KVY*MNINC)*KEY
         ETHDOT = -(DBTINC - KVTH*MBTINC)*KETH
C
```

```
C**** BUT IF TOOL COORDINATES ARE REQUESTED ****************************
C
      ELSE
C
         CALL TMATRICES
C
C**** FEED BACK CURRENT POSITION BY USING MEASURED JOINT COORDS.
C**** TRANSFORMED INTO TOOL COORDS. BY USE OF JACOBIAN.
C**** FORCE TRANSFORMATION IS UNNECESSARY AS FORCE SENSOR IS ALINED
C**** TO TOOL COORDS. ************************************************
C
         CALL MEASINC
C
C**** IF IN TOOL COORDS AND FORCE IS REQUESTED ************************
C
         IF( FORCE ) THEN
C
C**** INPUT DESIRED FORCE & VELOCITY COMMANDS FROM COMP1 **************
C
            CALL WRDIN (DTINC)
            CALL WRDIN (DESFN)
            CALL WRDIN (DBTINC)
            DNINC = 0.0
C
C       WRITE (*,130) DTINC,DESFN,DBTINC
130    FORMAT ('  DTINC = ',F12.5,'  DESFN = ',F12.5,
     @'  DBTINC = ',F12.5)
C
C**** POSITION MOVEMENT IS IN THE OPPOSITE SENCE TO FORCE *************
C
            ERTDOT = -(DTINC - KVT*MTINC)*KET
            ERRFN  =  (DESFN - KFN*MESFN)*KEFN
            ETHDOT = -(DBTINC - KVB*MBTINC)*KEB
C
C       WRITE (*,135) ERTDOT,ERRFN,ETHDOT
135    FORMAT ('  ERTDOT = ',F12.5,'  ERRFN = ',F12.5,
     @'  ETHDOT = ',F12.5)
C
C**** MORE EFFICIENT TO APPLY FORCE POSITION CONTROLLER BEFORE
C**** TRANSFORMING . REQUIRES 3 STATEMENTS INSTEAD OF 6 ***************
C
            CALL FCONTROLLER
C
         ELSE
C
C**** INPUT DESIRED TOOL VELOCITY COMMANDS FROM COMPUTER 1 ************
C
            CALL WRDIN (DTINC)
            CALL WRDIN (DNINC)
            CALL WRDIN (DBTINC)
            DESFN = 0.0
C
```

```
C**** TOOL COORDINATE ERROR GENERATION ***********************************
C
            ERTDOT = -(DTINC - KVT*MTINC)*KET
            ERNDOT = -(DNINC - KVN*MNINC)*KEN
            ETHDOT = -(DBTINC - KVB*MBTINC)*KEB
C
         ENDIF
      ENDIF
C
C**** TRANSFORM THESE LOCAL CONTROLLER DEMANDS INTO JOINT DEMANDS
C**** FOR FORCE AND POSITION IF NECESSARY ********************************
C
      CALL TRANSFORM
C
C**** THESE SEPERATE JOINT DEMANDS FOR FORCE AND POSITION CONTROL
C**** ARE PASSED TO TO BE COMBINED AS APPROPRIATE AND SIGNALS
C**** OUTPUT TO ANALOGUE CIRCUITS
C
C**** SUMMATION OF POSITION AND FORCE CONTROL SIGNALS .
C**** OUTPUT VOLTAGE ROUTINES, USING VGAIN AS VELOCITY GAIN
C**** AND FGAIN AS FORCE GAIN ********************************************
C
      VL1 = EL1DOT*VGAIN + EF1*FGAIN
      VL2 = EL2DOT*VGAIN + EF2*FGAIN
      VBETA = EBTDOT*WVGAIN + EM2*WFGAIN
C
C**** WRITE DATA TO MEMORY ***********************************************
C
C**** LOOK FOR FORCE CONTROL TO BE ACCESSED FOR 'ON' *********************
C**** ANYTHING ELSE 'OFF' ************************************************
C
C**** STORE DATA FROM COMPUTER 2 IN MEMORY IF 'FORCE' IS TRUE ********
C
      IF (FORCE .AND. (NSAMPLE .LT. 1000 )) THEN
         NSAMPLE = NSAMPLE + 1
         IF ((MESFN .LT. 0.0) .OR. IMPACT ) THEN
            IF (IMPACT) THEN
               MN = MN + MNINC
               KSTIFF = MESFN/MN
            ELSE
               MN = .000001
               IMPACT = .TRUE.
            ENDIF
         ENDIF
C
C
C          WRITE (*,*) '     NSAMPLE = ',NSAMPLE
C
C
         SFNCONTROL(NSAMPLE) = FNCONTROL
         SERRFN(NSAMPLE) = ERRFN
         SIFN(NSAMPLE) = IFN
         SMNINC(NSAMPLE) = MNINC
         SEL1DOT(NSAMPLE) = DFN
         SEL2DOT(NSAMPLE) = MESFN
         SEBTDOT(NSAMPLE) = DESFN
         SEF1(NSAMPLE) = MN
```

```
            SEF2(NSAMPLE) = KSTIFF
            SEM2(NSAMPLE) = 0.0
C
C            SEL1DOT(NSAMPLE) = EL1DOT
C            SEL2DOT(NSAMPLE) = EL2DOT
C            SEBTDOT(NSAMPLE) = EBTDOT
C            SEF1(NSAMPLE) = EF1
C            SEF2(NSAMPLE) = EF2
C            SEM2(NSAMPLE) = EM2
C
         ENDIF
C
         CALL VOLTOUT(1,VL1)
         CALL VOLTOUT(2,VL2)
         CALL VOLTOUT(3,VBETA)
C
C         WRITE (*,70) VL1,VL2,VBETA
70       FORMAT(' OUTPUT INTEGER VOLTAGES ARE :',/
        @'     VL1 = ',F8.4,'    VL2 = ',F8.4,'    VBETA = ',F8.4)
C
         GOTO 100
C
2000     CONTINUE
C
1000     CONTINUE
C
         CALL VOLTOUT (1,0)
         CALL VOLTOUT (2,0)
         CALL VOLTOUT (3,0)
C
         WRITE(*,*) '    NSAMPLE = ',NSAMPLE
         END


C************************ SUBROUTINE GEOMETRY ********************************
C
C***** MEASURES AT TIME OF CALLING, THE GLOBAL POSITION OF THE E.F. **
C***** AND CALCULATES ALL GOEMETRY NECESSARY FOR MATRIX CALCS . ******
C
         SUBROUTINE GEOMETRY
C
C
         COMMON D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
        @LMAX,LMIN,BETMAX,BETMIN,GEOROR,L1INTL,L2INTL,BEINTL,BVELBACK,
        @ERRLIM,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,FINISH,
        @XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,INCRIM,
        @IPORT1,IPORT2,IPORT3,SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,L1M,
        @L2M,BETAM,INVJ(3,3),F(3,3),J(3,3),NDOT1,MTDOT,MNDOT,MTHDOT,LEN1,
        @LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,EBTDOT,EF1,
        @EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,LEN2INTL,
        @BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,XJM,YJM,KFN,DFN,
        @IFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,
        @XFINISH,YFINISH,THFINISH,HOLD,POSITION,FORCE,SN,ST,SB,SELECT,
        @MODES,WORLD,LOCAL,KVX,KVY,KVTH,KEX,KEY,KETH,KET,KEN,KEB,KEFN,
        @NSAMPLE,SAMP,SAMPLE,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,IMPACT
C
```

```
      COMMON /ASTORE/ SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2
C
      COMMON /FNFEEDBAK/ MESFN,KVFN,FNFBAK
C
      REAL D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,L1C,L2C,BETAC,XJC,YJC,XD,LC,L1SQ,L2SQ,
     @BVELBACK,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,EL1,EL2,EBETA,
     @VL1,VL2,VBETA,XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,
     @SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,INVJ,F,J,NDOT1,MTDOT,MNDOT,
     @MTHDOT,LEN1,LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,
     @EBTDOT,EF1,EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,
     @LEN2INTL,BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,ML1INC,
     @ML2INC,XJM,YJM,KFN,DFN,IFN,MESFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,
     @KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,MODES,KVX,KVY,KVTH,KEX,KEY,KETH,
     @KET,KEN,KEB,KEFN,SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,KVFN,
     @FNFBAK
C
      INTEGER*2 IL1C,IL2C,IBETAC,L1INTL,L2INTL,BEINTL,L1M,
     @L2M,BETAM,L1VEL,L2VEL,BETVEL,ERRLIM,EL1P,EL2P,EBEP,INCRIM,
     @IPORT1,IPORT2,IPORT3,ST,SN,SB,SELECT,I,IADCON,IADC,CHANNEL
C
      INTEGER NSAMPLE
C
      LOGICAL*2 XFINISH,YFINISH,THFINISH,FINISH,GEOROR,HOLD,POSITION,
     @FORCE,WORLD,LOCAL,SAMP,SAMPLE,IMPACT
C
      DIMENSION SFNCONTROL(1005),SERRFN(1005),SIFN(1005),SMNINC(1005),
     @SEL1DOT(1005),SEL2DOT(1005),SEBTDOT(1005),SEF1(1005),SEF2(1005),
     @SEM2(1005)
C
      CHARACTER*10 ERROR(1)
C
C??????????? STATEMENTS TO BE REMOVED ??????????????????????????????
C       WRITE (*,200)
200   FORMAT ( '  ENTERED GEOMETRY CALCS. ROUTINE')
C?????????????????????????????????????????????????????????????????????
C
C**** FIND CURRENT POSITION OF ROBOT ************************************
C
      CALL PORTIN (L1M,L2M,BETAM)
C
C**** CONVERT FROM INTEGER COUNTS INTO CORRECT UNITS, IE. REALS ******
C
      LEN1 = -2.0E-05*L1M + 0.7
      LEN2 = -2.0E-05*L2M + 0.7
      BETA = -1.0471983E-03*BETAM
C
```

```
C**** CALCULATE WORLD GEOMETRY AND ANGLES ETC. ***************************
C
      XJM = (LEN1*LEN1 - LEN2*LEN2)/1.40
      SGAM1 = (XJM + D1)/LEN1
      SGAM2 = (D1 - XJM)/LEN2
      CGAM1 = SQRT(1 - SGAM1*SGAM1)
      CGAM2 = SQRT(1 - SGAM2*SGAM2)
      GAM2 = ASIN(SGAM2)
      YJM = LEN2*COS(GAM2) - 0.6062178
      THETAM = BETA + GAM2
      STHETM = SIN(THETAM)
      CTHETM = COS(THETAM)
      YM = YJM + R*STHETM
      XM = XJM + R*CTHETM
C
C      WRITE (*,300) XM,YM,THETAM,LEN1,LEN2,BETA

300   FORMAT ('  XM = ',F12.5,'  YM = ',F12.5,'  THETAM = ',F12.5,/
     a'  LEN1 = ',F12.5,'  LEN2 = ',F12.5,'  BETA = ',F12.5)
C      WRITE (*,305) SGAM1,SGAM2,CGAM1,CGAM2,XJM,YJM
305   FORMAT (' SGAM1= ',F12.5,' SGAM2= ',F12.5,'  CGAM1= ',F12.5,
     a' CGAM2= ',F12.5,/'  XJM = ',F12.5,'  YJM = ',F12.5)
C
      END

C     ************** SUBROUTINE LIMIT ***********************************
C     ** CHECKS ROBOT GEOMETRICAL JOINT COODS. LIMITS, RETURNS TO ****
C     ** INPUT IF ERROR (GEOROR = .TRUE.) ******************************
C
      SUBROUTINE LIMIT (L1C,L2C,BETAC)
C
C
C
      COMMON D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     aLMAX,LMIN,BETMAX,BETMIN,GEOROR,L1INTL,L2INTL,BEINTL,BVELBACK,
     aERRLIM,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,FINISH,
     aXDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,INCRIM,
     aIPORT1,IPORT2,IPORT3,SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,L1M,
     aL2M,BETAM,INVJ(3,3),F(3,3),J(3,3),NDOT1,MTDOT,MNDOT,MTHDOT,LEN1,
     aLEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,EBTDOT,EF1,
     aEF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,LEN2INTL,
     aBETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,XJM,YJM,KFN,DFN,
     aIFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,
     aXFINISH,YFINISH,THFINISH,HOLD,POSITION,FORCE,SN,ST,SB,SELECT,
     aMODES,WORLD,LOCAL,KVX,KVY,KVTH,KEX,KEY,KETH,KET,KEN,KEB,KEFN,
     aNSAMPLE,SAMP,SAMPLE,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,IMPACT
C
      COMMON /ASTORE/ SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     aSEBTDOT,SEF1,SEF2,SEM2
C
      COMMON /FNFEEDBAK/ MESFN,KVFN,FNFBAK
C
      REAL D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     aLMAX,LMIN,BETMAX,BETMIN,L1C,L2C,BETAC,XJC,YJC,XD,LC,L1SQ,L2SQ,
     aBVELBACK,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,EL1,EL2,EBETA,
     aVL1,VL2,VBETA,XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,
```

```
     @SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,INVJ,F,J,NDOT1,MTDOT,MNDOT,
     @MTHDOT,LEN1,LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,
     @EBTDOT,EF1,EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,
     @LEN2INTL,BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,ML1INC,
     @ML2INC,XJM,YJM,KFN,DFN,IFN,MESFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,
     @KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,MODES,KVX,KVY,KVTH,KEX,KEY,KETH,
     @KET,KEN,KEB,KEFN,SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,KVFN,
     @FNFBAK
C
      INTEGER*2 IL1C,IL2C,IBETAC,L1INTL,L2INTL,BEINTL,L1M,
     @L2M,BETAM,L1VEL,L2VEL,BETVEL,ERRLIM,EL1P,EL2P,EBEP,INCRIM,
     @IPORT1,IPORT2,IPORT3,ST,SN,SB,SELECT,I,IADCON,IADC,CHANNEL
C
      INTEGER NSAMPLE
C
      LOGICAL*2 XFINISH,YFINISH,THFINISH,FINISH,GEOROR,HOLD,POSITION,
     @FORCE,WORLD,LOCAL,SAMP,SAMPLE,IMPACT
C
      DIMENSION SFNCONTROL(1005),SERRFN(1005),SIFN(1005),SMNINC(1005),
     @SEL1DOT(1005),SEL2DOT(1005),SEBTDOT(1005),SEF1(1005),SEF2(1005),
     @SEM2(1005)
C
      CHARACTER*10 ERROR(1)
C
C**** CHECK GEOMETRICAL ERROR LIMITS OF ROBOT JOINTS ******************
C
      GEOROR = .FALSE.
      IF (LMAX.LT.L1C) GOTO 500
      IF (LMAX.LT.L2C) GOTO 500
      IF (LMIN.GT.L1C) GOTO 500
      IF (LMIN.GT.L2C) GOTO 500
      IF (BETMAX.LT.BETAC) GOTO 500
      IF (BETMIN.GT.BETAC) GOTO 500
C
      GOTO 600
C
C**** SET ERROR INDICATOR 'GEOROR = TRUE' ***************************
C
500   GEOROR = .TRUE.
600   CONTINUE
C     RETURN
      END

C******************************* SUBROUTINE WMATRICES ********************
C**** EVALUATES AT CURRENT POSITION : JACOBIAN, INV. JACOBIAN AND ****
C**** FORCE TRANSFORMATION MATRICES, FOR WORLD COORDS.  ***************
C
      SUBROUTINE WMATRICES
C
C
      COMMON D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,GEOROR,L1INTL,L2INTL,BEINTL,BVELBACK,
```

```
     @ERRLIM,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,FINISH,
     @XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,INCRIM,
     @IPORT1,IPORT2,IPORT3,SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,L1M,
     @L2M,BETAM,INVJ(3,3),F(3,3),J(3,3),NDOT1,MTDOT,MNDOT,MTHDOT,LEN1,
     @LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,EBTDOT,EF1,
     @EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,LEN2INTL,
     @BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,XJM,YJM,KFN,DFN,
     @IFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,
     @XFINISH,YFINISH,THFINISH,HOLD,POSITION,FORCE,SN,ST,SB,SELECT,
     @MODES,WORLD,LOCAL,KVX,KVY,KVTH,KEX,KEY,KETH,KET,KEN,KEB,KEFN,
     @NSAMPLE,SAMP,SAMPLE,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,IMPACT
C
      COMMON /ASTORE/ SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2
C
      COMMON /FNFEEDBAK/ MESFN,KVFN,FNFBAK
C
      REAL D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,L1C,L2C,BETAC,XJC,YJC,XD,LC,L1SQ,L2SQ,
     @BVELBACK,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,EL1,EL2,EBETA,
     @VL1,VL2,VBETA,XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,
     @SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,INVJ,F,J,NDOT1,MTDOT,MNDOT,
     @MTHDOT,LEN1,LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,
     @EBTDOT,EF1,EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,
     @LEN2INTL,BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,ML1INC,
     @ML2INC,XJM,YJM,KFN,DFN,IFN,MESFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,
     @KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,MODES,KVX,KVY,KVTH,KEX,KEY,KETH,
     @KET,KEN,KEB,KEFN,SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,KVFN,
     @FNFBAK
C
      INTEGER*2 IL1C,IL2C,IBETAC,L1INTL,L2INTL,BEINTL,L1M,
     @L2M,BETAM,L1VEL,L2VEL,BETVEL,ERRLIM,EL1P,EL2P,EBEP,INCRIM,
     @IPORT1,IPORT2,IPORT3,ST,SN,SB,SELECT,I,IADCON,IADC,CHANNEL
C
      INTEGER NSAMPLE
C
      LOGICAL*2 XFINISH,YFINISH,THFINISH,FINISH,GEOROR,HOLD,POSITION,
     @FORCE,WORLD,LOCAL,SAMP,SAMPLE,IMPACT
C
      DIMENSION SFNCONTROL(1005),SERRFN(1005),SIFN(1005),SMNINC(1005),
     @SEL1DOT(1005),SEL2DOT(1005),SEBTDOT(1005),SEF1(1005),SEF2(1005),
     @SEM2(1005)
C
      CHARACTER*10 ERROR(1)
C
C
C**** EVALUATE JACOBIAN INVERSE FOR WORLD COORDS

C     ( (1,1) (1,2) (1,3) )
C     ( (2,1) (2,2) (2,3) )
C     ( (3,1) (3,2) (3,3) )

       INVJ(1,1) = SGAM1
       INVJ(1,2) = CGAM1
       INVJ(1,3) = R*(STHETM*SGAM1 - CTHEM*CGAM1)
```

```
      INVJ(2,1) = -SGAM2
      INVJ(2,2) = CGAM2
      INVJ(2,3) = -R*(STHETM*SGAM2 + CTHETM*CGAM2)
      INVJ(3,1) = 0
      INVJ(3,2) = 0
      INVJ(3,3) = 1

C      WRITE(*,100)INVJ
 100  FORMAT(' INVJ(1,1),INVJ(2,1),INVJ(3,1).....(3,3),= ',/9F8.4)

C EVALUATE FORCE MATRIX,ie. JACOBIAN TRANSPOSE
      FDET = (SGAM1*CGAM2 + SGAM2*CGAM1)
      F(1,1) = INVJ(2,2)/FDET
      F(1,2) = -INVJ(2,1)/FDET
      F(1,3) = 0
      F(2,1) = -INVJ(1,2)/FDET
      F(2,2) = INVJ(1,1)/FDET
      F(2,3) = 0
      F(3,1) = -R*STHETM
      F(3,2) = R*CTHETM
      F(3,3) = 1

C      WRITE(*,110)FDET,F
 110  FORMAT(' FDET & F(1,1),F(2,1),F(3,1)....(3,3),= ',F8.4,/
     @9F8.4)
C EVALUATE JACOBIAN MATRIX,USED FOR WORLD POSITION FEEDBACK
      J(1,1) = F(1,1)
      J(1,2) = F(2,1)
      J(1,3) = F(3,1)
      J(2,1) = F(1,2)
      J(2,2) = F(2,2)
      J(2,3) = F(3,2)
      J(3,1) = 0
      J(3,2) = 0
      J(3,3) = 1

C      WRITE(*,120)J
 120  FORMAT(' J(1,1),J(2,1),J(3,1)....(3,3),= ',/9F8.4)
C
      END


C****************************** SUBROUTINE TMATRICES **********************
C**** EVALUATES AT CURRENT POSITION : JACOBIAN, INV. JACOBIAN AND ****
C**** FORCE TRANSFORMATION MATRICES , IN TOOL COORDS ********************
C
      SUBROUTINE TMATRICES
C
C
      COMMON D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,GEOROR,L1INTL,L2INTL,BEINTL,BVELBACK,
     @ERRLIM,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,FINISH,
     @XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,INCRIM,
     @IPORT1,IPORT2,IPORT3,SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,L1M,
```

```
     @L2M,BETAM,INVJ(3,3),F(3,3),J(3,3),NDOT1,MTDOT,MNDOT,MTHDOT,LEN1,
     @LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,EBTDOT,EF1,
     @EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,LEN2INTL,
     @BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,XJM,YJM,KFN,DFN,
     @IFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,
     @XFINISH,YFINISH,THFINISH,HOLD,POSITION,FORCE,SN,ST,SB,SELECT,
     @MODES,WORLD,LOCAL,KVX,KVY,KVTH,KEX,KEY,KETH,KET,KEN,KEB,KEFN,
     @NSAMPLE,SAMP,SAMPLE,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,IMPACT
C
      COMMON /ASTORE/ SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2
C
      COMMON /FNFEEDBAK/ MESFN,KVFN,FNFBAK
C
      REAL D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,L1C,L2C,BETAC,XJC,YJC,XD,LC,L1SQ,L2SQ,
     @BVELBACK,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,EL1,EL2,EBETA,
     @VL1,VL2,VBETA,XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,
     @SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,INVJ,F,J,NDOT1,MTDOT,MNDOT,
     @MTHDOT,LEN1,LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,
     @EBTDOT,EF1,EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,
     @LEN2INTL,BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,ML1INC,
     @ML2INC,XJM,YJM,KFN,DFN,IFN,MESFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,
     @KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,MODES,KVX,KVY,KVTH,KEX,KEY,KETH,
     @KET,KEN,KEB,KEFN,SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,KVFN,
     @FNFBAK
C
      INTEGER*2 IL1C,IL2C,IBETAC,L1INTL,L2INTL,BEINTL,L1M,
     @L2M,BETAM,L1VEL,L2VEL,BETVEL,ERRLIM,EL1P,EL2P,EBEP,INCRIM,
     @IPORT1,IPORT2,IPORT3,ST,SN,SB,SELECT,I,IADCON,IADC,CHANNEL
C
      INTEGER NSAMPLE
C
      LOGICAL*2 XFINISH,YFINISH,THFINISH,FINISH,GEORGR,HOLD,POSITION,
     @FORCE,WORLD,LOCAL,SAMP,SAMPLE,IMPACT
C
      DIMENSION SFNCONTROL(1005),SERRFN(1005),SIFN(1005),SMNINC(1005),
     @SEL1DOT(1005),SEL2DOT(1005),SEBTDOT(1005),SEF1(1005),SEF2(1005),
     @SEM2(1005)
C
      CHARACTER*10 ERROR(1)
C
C --->
C**** EVALUATE JACOBIAN INVERSE FOR TOOL COORDS
C
C     ( (1,1) (1,2) (1,3) )
C     ( (2,1) (2,2) (2,3) )
C     ( (3,1) (3,2) (3,3) )

      INVJ(1,1) = STHETM*SGAM1 - CTHETM*CGAM1
      INVJ(1,2) = CTHETM*SGAM1 + STHETM*CGAM1
      INVJ(1,3) = R*INVJ(1,1)
      INVJ(2,1) = -(STHETM*SGAM2 + CTHETM*CGAM2)
      INVJ(2,2) = STHETM*CGAM2 - CTHETM*SGAM2
      INVJ(2,3) = R*INVJ(2,1)
```

```
      INVJ(3,1) = 0
      INVJ(3,2) = 0
      INVJ(3,3) = 1
C      WRITE(*,100)INVJ
 100  FORMAT('                        ENTERING MATRICES ROUTINE'/
     @' INVJ(1,1),INVJ(2,1),INVJ(3,1).....(3,3),= ',/9F8.4)

C EVALUATE FORCE MATRIX,ie. JACOBIAN TRANSPOSE

      FDET = (SGAM1*CGAM2 + SGAM2*CGAM1)
      F(1,1) = INVJ(2,2)/FDET
      F(1,2) = -INVJ(2,1)/FDET
      F(1,3) = 0
      F(2,1) = -INVJ(1,2)/FDET
      F(2,2) = INVJ(1,1)/FDET
      F(2,3) = 0
      F(3,1) = -R
      F(3,2) = 0
      F(3,3) = 1

C      WRITE(*,110)FDET,F
 110  FORMAT(' FDET & F(1,1),F(2,1),F(3,1)....(3,3),= ',F8.4,/
     @9F8.4)
C EVALUATE JACOBIAN MATRIX,USED FOR LOCAL POSITION FEEDBACK

      J(1,1) = F(1,1)
      J(1,2) = F(2,1)
      J(1,3) = F(3,1)
      J(2,1) = F(1,2)
      J(2,2) = F(2,2)
      J(2,3) = 0
      J(3,1) = 0
      J(3,2) = 0
      J(3,3) = 1

C      WRITE(*,120)J
 120  FORMAT(' J(1,1),J(2,1),J(3,1)....(3,3),= ',/9F8.4)
C
      END


C************************** SUBROUTINE MEASINC ****************************
C**** USES CURRENT MEASREMENT OF JOINT POSITIONS, (LEN1,LEN2,BETA) ***
C***** TO GENERATE CURRENT LOCAL POSITION MOVEMENT INCRIMENTS *********
C
      SUBROUTINE MEASINC
C
C
C
      COMMON D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,GEOROR,L1INTL,L2INTL,BEINTL,BVELBACK,
     @ERRLIM,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,FINISH,
     @XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,INCRIM,
     @IPORT1,IPORT2,IPORT3,SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,L1M,
     @L2M,BETAM,INVJ(3,3),F(3,3),J(3,3),NDOT1,MTDOT,MNDOT,MTHDOT,LEN1,
     @LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,EBTDOT,EF1,
```

```
     @EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,LEN2INTL,
     @BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,XJM,YJM,KFN,DFN,
     @IFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,
     @XFINISH,YFINISH,THFINISH,HOLD,POSITION,FORCE,SN,ST,SB,SELECT,
     @MODES,WORLD,LOCAL,KVX,KVY,KVTH,KEX,KEY,KETH,KET,KEN,KEB,KEFN,
     @NSAMPLE,SAMP,SAMPLE,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,IMPACT
C
      COMMON /ASTORE/ SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2
C
      COMMON /FNFEEDBAK/ MESFN,KVFN,FNFBAK
C
       REAL D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,L1C,L2C,BETAC,XJC,YJC,XD,LC,L1SQ,L2SQ,
     @BVELBACK,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,EL1,EL2,EBETA,
     @VL1,VL2,VBETA,XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,
     @SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,INVJ,F,J,NDOT1,MTDOT,MNDOT,
     @MTHDOT,LEN1,LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,
     @EBTDOT,EF1,EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,
     @LEN2INTL,BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,ML1INC,
     @ML2INC,XJM,YJM,KFN,DFN,IFN,MESFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,
     @KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,MODES,KVX,KVY,KVTH,KEX,KEY,KETH,
     @KET,KEN,KEB,KEFN,SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,KVFN,
     @FNFBAK
C
       INTEGER*2 IL1C,IL2C,IBETAC,L1INTL,L2INTL,BEINTL,L1M,
     @L2M,BETAM,L1VEL,L2VEL,BETVEL,ERRLIM,EL1P,EL2P,EBEP,INCRIM,
     @IPORT1,IPORT2,IPORT3,ST,SN,SB,SELECT,I,IADCON,IADC,CHANNEL
C
       INTEGER NSAMPLE
C
       LOGICAL*2 XFINISH,YFINISH,THFINISH,FINISH,GEOROR,HOLD,POSITION,
     @FORCE,WORLD,LOCAL,SAMP,SAMPLE,IMPACT
C
       DIMENSION SFNCONTROL(1005),SERRFN(1005),SIFN(1005),SMNINC(1005),
     @SEL1DOT(1005),SEL2DOT(1005),SEBTDOT(1005),SEF1(1005),SEF2(1005),
     @SEM2(1005)
C
       CHARACTER*10 ERROR(1)
C
C**** CALCULATE MEASURED JOINT INCRIMENT (PROPORTIONAL TO JOINT VEL.)
C
       ML1INC = LEN1 - LEN1INTL
       ML2INC = LEN2 - LEN2INTL
       MBTINC = BETA - BETAINTL

C
C**** ROBOT GEOMETRICAL SIMULATION, TO OBTAIN ACTUAL LOCAL COORDS
C
       MTINC = J(1,1)*ML1INC + J(1,2)*ML2INC + J(1,3)*MBTINC
       MNINC = J(2,1)*ML1INC + J(2,2)*ML2INC
       MBTINC = MBTINC
C
C       WRITE (*,100) ML1INC,ML2INC,MBTINC,MTINC,MNINC,MBTINC
100    FORMAT ('  ML1INC = ',F12.5,'  ML2INC = ',F12.5,'  MBTINC = ',F
```

```
     @12.5,/'  MTINC = ',F12.5,'  MNINC = ',F12.5,'  MTHINC = ',F12.5)
C
C**** SIMULATION OF ROBOT STRUCTURE TO PRODUCE LOCAL FORCE
C
C       FN1 = INVJ(1,2)*EF1 + INVJ(2,2)*EF2
C
C**** RESET NEXT CYCLE'S INITIAL VALUES ************************************
C
       LEN1INTL = LEN1
       LEN2INTL = LEN2
       BETAINTL = BETA
C
       END



C********************* SUBROUTINE TRANSFORM ***********************************
C**** DEPENDING FORCE/POSITION SELECTION, APPROPIATE ACTION IS TAKEN
C**** FOR FORCE/POSITION TRANSFORMATION OR JUST POSITION
C**** TRANSFORMATION INTO JOINT DEMANDS ****************************************
C
C
      SUBROUTINE TRANSFORM
C
C
C
       COMMON D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,GEOROR,L1INTL,L2INTL,BEINTL,BVELBACK,
      @ERRLIM,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,FINISH,
      @XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,INCRIM,
      @IPORT1,IPORT2,IPORT3,SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,L1M,
      @L2M,BETAM,INVJ(3,3),F(3,3),J(3,3),NDOT1,MTDOT,MNDOT,MTHDOT,LEN1,
      @LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,EBTDOT,EF1,
      @EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,LEN2INTL,
      @BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,XJM,YJM,KFN,DFN,
      @IFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,
      @XFINISH,YFINISH,THFINISH,HOLD,POSITION,FORCE,SN,ST,SB,SELECT,
      @MODES,WORLD,LOCAL,KVX,KVY,KVTH,KEX,KEY,KETH,KET,KEN,KEB,KEFN,
      @NSAMPLE,SAMP,SAMPLE,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,IMPACT
C
       COMMON /ASTORE/ SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
      @SEBTDOT,SEF1,SEF2,SEM2
C
       COMMON /FNFEEDBAK/ MESFN,KVFN,FNFBAK
C
       REAL D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
      @LMAX,LMIN,BETMAX,BETMIN,L1C,L2C,BETAC,XJC,YJC,XD,LC,L1SQ,L2SQ,
      @BVELBACK,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,EL1,EL2,EBETA,
      @VL1,VL2,VBETA,XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,
      @SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,INVJ,F,J,NDOT1,MTDOT,MNDOT,
      @MTHDOT,LEN1,LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,
      @EBTDOT,EF1,EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,
      @LEN2INTL,BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,ML1INC,
      @ML2INC,XJM,YJM,KFN,DFN,IFN,MESFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,
      @KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,MODES,KVX,KVY,KVTH,KEX,KEY,KETH,
      @KET,KEN,KEB,KEFN,SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
```

```
     @SEBTDOT,SEF1,SEF2,SEM2,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,KVFN,
     @FNFBAK
C
      INTEGER*2 IL1C,IL2C,IBETAC,L1INTL,L2INTL,BEINTL,L1M,
     @L2M,BETAM,L1VEL,L2VEL,BETVEL,ERRLIM,EL1P,EL2P,EBEP,INCRIM,
     @IPORT1,IPORT2,IPORT3,ST,SN,SB,SELECT,I,IADCON,IADC,CHANNEL
C
      INTEGER NSAMPLE
C
      LOGICAL*2 XFINISH,YFINISH,THFINISH,FINISH,GEOROR,HOLD,POSITION,
     @FORCE,WORLD,LOCAL,SAMP,SAMPLE,IMPACT
C
       DIMENSION SFNCONTROL(1005),SERRFN(1005),SIFN(1005),SMNINC(1005),
     @SEL1DOT(1005),SEL2DOT(1005),SEBTDOT(1005),SEF1(1005),SEF2(1005),
     @SEM2(1005)
C
      CHARACTER*10 ERROR(1)
C
C**** EXAMINE FOR FORCE POSITION SELECTION ********************************
C
      IF (FORCE) GOTO 500
C
C**** POSITION VELOCITY TRANSFORMATIONS ONLY ******************************
C
C
C**** GENERATING JOINT SIGNALS FROM LOCAL VELOCITY POSITION SIGNALS **
C
      EL1DOT = INVJ(1,1)*ERTDOT + INVJ(1,2)*ERNDOT + INVJ(1,3)*ETHDOT
      EL2DOT = INVJ(2,1)*ERTDOT + INVJ(2,2)*ERNDOT + INVJ(2,3)*ETHDOT
      EBTDOT =                                       ETHDOT
C
C**** NO LOCAL FORCE  REQUIREMENT,THEREFORE NO JOINT FORCE COMMANDS **
C
      EF1 = 0
      EF2 = 0
      EM2 = 0
C
C      WRITE(*,106)EL1DOT,EL2DOT,EBTDOT,EF1,EF2,EM2
 106  FORMAT('  EL1DOT,EL2DOT,EBTDOT,EF1,EF2,EM2,=',6F8.4)
C
      GOTO 600
C
500   CONTINUE
C
C**** COMBINING THE RELEVANT PORTIONS OF THE FORCE AND POSITION
C**** TRANSFORMATIONS TO ACHIEVE A SIMULTANEOUS FORCE/POSITION
C**** TRANSFORMATION FROM LOCAL TO JOINT DEMANDS **************************
C
C
C**** JOINT TORQUE SIGNALS OWING TO LOCAL NORMAL FORCE REQUIREMENT,
C**** FNCONTROL ***********************************************************
C
```

```
C**** ADDING POSITIVE SENSOR FORCE FEEDBACK, TO ELIMINATE FORCE ******
C**** DISTURBANCE MEASURED AT SENSOR, MESFN ********************************
C
      FACTUR = FNCONTROL + (FNFBAK*MESFN)
      EF1 = FACTUR*F(1,2)
      EF2 = FACTUR*F(2,2)
      EM2 = 0.0
C     F(3,2) = 0
C
C
C**** GENERATING JOINT SIGNALS FROM LOCAL VELOCITY POSITION SIGNALS
C**** IN BOTH THE TANGENTIAL AND BETA (WRIST) DIRCTION *****************
C
      EL1DOT = INVJ(1,1)*ERTDOT + INVJ(1,3)*ETHDOT
      EL2DOT = INVJ(2,1)*ERTDOT + INVJ(2,3)*ETHDOT
      EBTDOT =                    ETHDOT
C
C     WRITE(*,106)EL1DOT,EL2DOT,EBTDOT,EF1,EF2,EM2
C
600   CONTINUE
C
      END

C     ****************** SUBROUTINE VOLTOUT ************************************
C      ************ PROGRAMME VOLTAGE OUTPUT TO DADIO BOARD ******************
C
      SUBROUTINE VOLTOUT(PORT,VOLTS)
C
      REAL*4 VOLTS,V1,P
      INTEGER*2 IPORT,IP,IA,IH,L,PORT,PORT2
C
10    CONTINUE
C     DO 180 I=1,10000
C
150   IF(VOLTS.LE.-10.0) VOLTS = -10.0
      IF(VOLTS.GE. 10.0) VOLTS = 10.0
C
      V1 = (VOLTS + 10)*204.75
      IP = INT(V1)
      P  = V1/16
      IH = INT(P)
      L  = 16*(IP - IH*16)
C*************** IA IS BASE ADDR.******************
      PORT2 = PORT + PORT
      IA = 1840 + PORT2
C************ OUTPUT VOLTS ***************************
      IPORT = IA + 1
C
      CALL OUT(IPORT,IH)
      CALL OUT(IA,L)
180   CONTINUE
C     RETURN
      END
```

```
C************************** SUBROUTINE FCONTROLLER ************************
C**** TAKES ERRORS IN FORCE AND/OR POSITION, APPLYING P.I.D. CONTROL *
C***** THEIR EFFECT GOVERNED BY KPFN, KIFN, KDFN, RESPECTIVELY ********
C
      SUBROUTINE FCONTROLLER
C
C
      COMMON D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,GEOROR,L1INTL,L2INTL,BEINTL,BVELBACK,
     @ERRLIM,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,FINISH,
     @XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,INCRIM,
     @IPORT1,IPORT2,IPORT3,SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,L1M,
     @L2M,BETAM,INVJ(3,3),F(3,3),J(3,3),NDOT1,MTDOT,MNDOT,MTHDOT,LEN1,
     @LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,EBTDOT,EF1,
     @EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,LEN2INTL,
     @BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,XJM,YJM,KFN,DFN,
     @IFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,
     @XFINISH,YFINISH,THFINISH,HOLD,POSITION,FORCE,SN,ST,SB,SELECT,
     @MODES,WORLD,LOCAL,KVX,KVY,KVTH,KEX,KEY,KETH,KET,KEN,KEB,KEFN,
     @NSAMPLE,SAMP,SAMPLE,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,IMPACT
C
      COMMON /ASTORE/ SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2
C
      COMMON /FNFEEDBAK/ MESFN,KVFN,FNFBAK
C
      REAL D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,L1C,L2C,BETAC,XJC,YJC,XD,LC,L1SQ,L2SQ,
     @BVELBACK,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,EL1,EL2,EBETA,
     @VL1,VL2,VBETA,XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,
     @SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,INVJ,F,J,NDOT1,MTDOT,MNDOT,
     @MTHDOT,LEN1,LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,
     @EBTDOT,EF1,EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,
     @LEN2INTL,BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,ML1INC,
     @ML2INC,XJM,YJM,KFN,DFN,IFN,MESFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,
     @KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,MODES,KVX,KVY,KVTH,KEX,KEY,KETH,
     @KET,KEN,KEB,KEFN,SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,KVFN,
     @FNFBAK
C
      INTEGER*2 IL1C,IL2C,IBETAC,L1INTL,L2INTL,BEINTL,L1M,
     @L2M,BETAM,L1VEL,L2VEL,BETVEL,ERRLIM,EL1P,EL2P,EBEP,INCRIM,
     @IPORT1,IPORT2,IPORT3,ST,SN,SB,SELECT,I,IADCON,IADC,CHANNEL
C
      INTEGER NSAMPLE
C
      LOGICAL*2 XFINISH,YFINISH,THFINISH,FINISH,GEOROR,HOLD,POSITION,
     @FORCE,WORLD,LOCAL,SAMP,SAMPLE,IMPACT
C
      DIMENSION SFNCONTROL(1005),SERRFN(1005),SIFN(1005),SMNINC(1005),
     @SEL1DOT(1005),SEL2DOT(1005),SEBTDOT(1005),SEF1(1005),SEF2(1005),
     @SEM2(1005)
C
      CHARACTER*10 ERROR(1)
C
```

```
C
C**** SET SAMPLING FLAG ***************************************************
C
      SAMP = .TRUE.
C
C**** EVALUATE DIFFERENTIAL COMPONENT *************************************
C
      DFN = ERRFN - ERRFINTL
C
C**** EVALUATE INTEGRAL USING KIER TO CONTROL THE ERROR RAMP *********
C
      ERRFNI = ERRFN*KIER
      IFN = IFN + ERRFNI
C
C**** EVALUATE INTEGRAL USING KIER TO CONTROL THE ERROR RAMP *********
C**** IFNLIM INTEGRATION LIMIT ********************************************
C**** ADJUSTABLE INTEGRATION LIMITS ***************************************
C
      IF (ABS(IFN).GE.ABS(DESFN)) THEN
          IFN = IFN - ERRFNI
      ENDIF
C
C**** P.I.D. CONTROLLER WITH GAINS KPFN, KIFN, KDFN, KNINC ***********
C
      FNCONTROL = ERRFN*KPFN + IFN*KIFN + DFN*KDFN + KNINC*MNINC
     @           + DESFN*FNFWD
C
C      WRITE (*,10) ERRFN,ERRFINTL,IFN,MNINC,KPFN,KIFN,KDFN,FNCONTROL
10     FORMAT ('              ENTERED FCONTROLLER ROUTINE',/
     @ '  ERRFN = ',F12.5,'  ERRFINTL = ',F12.5,'  IFN = ',
     @F12.5,/'  MNINC = ',F12.5,'  KPFN = ',F12.5,'  KIFN = ',F12.5,
     @/'  KDFN = ',F12.5,'  FNCONTROL = ',F12.5 )
C
C**** RE-INITIALISE ERRFN FOR DERIVATIVE ACTION ***************************
C
      ERRFINTL = ERRFN
C
      END



C***************************** SUBROUTINE ADCFORCE ***************************
C**** MEASURES VOLTAGE FROM ADC. PORT 15 AND CONVERTS TO FORCE (N) ***
C
      SUBROUTINE ADCFORCE
C
C
      COMMON D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,GEOROR,L1INTL,L2INTL,BEINTL,BVELBACK,
     @ERRLIM,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,FINISH,
     @XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,INCRIM,
     @IPORT1,IPORT2,IPORT3,SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,L1M,
     @L2M,BETAM,INVJ(3,3),F(3,3),J(3,3),NDOT1,MTDOT,MNDOT,MTHDOT,LEN1,
     @LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,EBTDOT,EF1,
     @EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,LEN2INTL,
     @BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,XJM,YJM,KFN,DFN,
```

```
     @IFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,
     @XFINISH,YFINISH,THFINISH,HOLD,POSITION,FORCE,SN,ST,SB,SELECT,
     @MODES,WORLD,LOCAL,KVX,KVY,KVTH,KEX,KEY,KETH,KET,KEN,KEB,KEFN,
     @NSAMPLE,SAMP,SAMPLE,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,IMPACT
C
      COMMON /ASTORE/ SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2
C
      COMMON /FNFEEDBAK/ MESFN,KVFN,FNFBAK
C
      REAL D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
     @LMAX,LMIN,BETMAX,BETMIN,L1C,L2C,BETAC,XJC,YJC,XD,LC,L1SQ,L2SQ,
     @BVELBACK,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,EL1,EL2,EBETA,
     @VL1,VL2,VBETA,XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,
     @SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,INVJ,F,J,NDOT1,MTDOT,MNDOT,
     @MTHDOT,LEN1,LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,
     @EBTDOT,EF1,EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,
     @LEN2INTL,BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,ML1INC,
     @ML2INC,XJM,YJM,KFN,DFN,IFN,MESFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,
     @KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,MODES,KVX,KVY,KVTH,KEX,KEY,KETH,
     @KET,KEN,KEB,KEFN,SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
     @SEBTDOT,SEF1,SEF2,SEM2,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,KVFN,
     @FNFBAK
C
      INTEGER*2 IL1C,IL2C,IBETAC,L1INTL,L2INTL,BEINTL,L1M,
     @L2M,BETAM,L1VEL,L2VEL,BETVEL,ERRLIM,EL1P,EL2P,EBEP,INCRIM,
     @IPORT1,IPORT2,IPORT3,ST,SN,SB,SELECT,I,IADCON,IADC,CHANNEL
C
      INTEGER NSAMPLE
C
      LOGICAL*2 XFINISH,YFINISH,THFINISH,FINISH,GEOROR,HOLD,POSITION,
     @FORCE,WORLD,LOCAL,SAMP,SAMPLE,IMPACT
C
      DIMENSION SFNCONTROL(1005),SERRFN(1005),SIFN(1005),SMNINC(1005),
     @SEL1DOT(1005),SEL2DOT(1005),SEBTDOT(1005),SEF1(1005),SEF2(1005),
     @SEM2(1005)
C
      CHARACTER*10 ERROR(1)
C
C
C**** READ FORCE MEASURING CHANNEL 15 OF ADC. ****************************
C
      CHANNEL = 15
      IADC = IADCON (CHANNEL)
C
C**** THE VALUE OF IADC RANGES FROM -2048 TO +2047
C**** COMPRESSIVE FORCES GIVE A +VE. VOLTAGE SWING FROM 0 TO 10V MAX.
C**** TENSILE FORCE CANNOT BE MEASURED, THEREFOR THE -VE COMPONENT
C**** IS REMOVED . KVFN IS THE IS THE FACTOR RELATING IADC & FORCE ***
C
      IF (IADC .LT. 0) IADC = 0
C
C**** CONVERT INTEGER IADC TO FORCE IN NEWTONS ***************************
C
      KVFN   = -0.31573
      MESFN = KVFN*IADC
```

```
C
C       WRITE (*,5)  IADC,MESFN
5       FORMAT ('    ENTERED FORCE SAMPLING ROUTINE',/
      @'     IADC = ',I8,'     MESFN = ',F12.5)
C
        END

C********************** SUBROUTINE STORE ************************************
C**** STORES MEMORY DATA TO HARD DISK A:SAMPLE.DAT FILE ***************
C
        SUBROUTINE STORE
C
C
        COMMON D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
      @LMAX,LMIN,BETMAX,BETMIN,GEOROR,L1INTL,L2INTL,BEINTL,BVELBACK,
      @ERRLIM,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,FINISH,
      @XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,INCRIM,
      @IPORT1,IPORT2,IPORT3,SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,L1M,
      @L2M,BETAM,INVJ(3,3),F(3,3),J(3,3),NDOT1,MTDOT,MNDOT,MTHDOT,LEN1,
      @LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,EBTDOT,EF1,
      @EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,LEN2INTL,
      @BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,XJM,YJM,KFN,DFN,
      @IFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,
      @XFINISH,YFINISH,THFINISH,HOLD,POSITION,FORCE,SN,ST,SB,SELECT,
      @MODES,WORLD,LOCAL,KVX,KVY,KVTH,KEX,KEY,KETH,KET,KEN,KEB,KEFN,
      @NSAMPLE,SAMP,SAMPLE,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,IMPACT
C
        COMMON /ASTORE/ SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
      @SEBTDOT,SEF1,SEF2,SEM2
C
        COMMON /FNFEEDBAK/ MESFN,KVFN,FNFBAK
C
        REAL D,D1,R,GAM1C,GAM2C,SGAM1C,SGAM2C,CGAM1C,CGAM2C,X,Y,THETA,
      @LMAX,LMIN,BETMAX,BETMIN,L1C,L2C,BETAC,XJC,YJC,XD,LC,L1SQ,L2SQ,
      @BVELBACK,VGAIN,XFINAL,YFINAL,THFINAL,XM,YM,THETAM,EL1,EL2,EBETA,
      @VL1,VL2,VBETA,XDELT,YDELT,THDELT,XINC,YINC,THINC,WVGAIN,WVELBACK,
      @SGAM1,SGAM2,CGAM1,CGAM2,STHETM,CTHETM,INVJ,F,J,NDOT1,MTDOT,MNDOT,
      @MTHDOT,LEN1,LEN2,BETA,DSTDOT,DSNDOT,DTHDOT,DESFN,EL1DOT,EL2DOT,
      @EBTDOT,EF1,EF2,EM2,KVT,KVN,KVB,DTINC,DNINC,DBTINC,LEN1INTL,
      @LEN2INTL,BETAINTL,MTINC,MNINC,MBTINC,ERTDOT,ERNDOT,ETHDOT,ML1INC,
      @ML2INC,XJM,YJM,KFN,DFN,IFN,MESFN,ERRFN,ERRFINTL,FNCONTROL,KDFN,
      @KIFN,KPFN,IFNLIM,FGAIN,WFGAIN,MODES,KVX,KVY,KVTH,KEX,KEY,KETH,
      @KET,KEN,KEB,KEFN,SFNCONTROL,SERRFN,SIFN,SMNINC,SEL1DOT,SEL2DOT,
      @SEBTDOT,SEF1,SEF2,SEM2,KNINC,ERRFNI,KIER,FNFWD,MN,KSTIFF,KVFN,
      @FNFBAK
C
        INTEGER*2 IL1C,IL2C,IBETAC,L1INTL,L2INTL,BEINTL,L1M,
      @L2M,BETAM,L1VEL,L2VEL,BETVEL,ERRLIM,EL1P,EL2P,EBEP,INCRIM,
      @IPORT1,IPORT2,IPORT3,ST,SN,SB,SELECT,I,IADCON,IADC,CHANNEL
C
        INTEGER NSAMPLE
C
        LOGICAL*2 XFINISH,YFINISH,THFINISH,FINISH,GEOROR,HOLD,POSITION,
      @FORCE,WORLD,LOCAL,SAMP,SAMPLE,IMPACT
C
        DIMENSION SFNCONTROL(1005),SERRFN(1005),SIFN(1005),SMNINC(1005),
```

```
     @SEL1DOT(1005),SEL2DOT(1005),SEBTDOT(1005),SEF1(1005),SEF2(1005),
     @SEM2(1005)
C
      CHARACTER*10 ERROR(1)
C
C
1900  WRITE (*,1905)
1905  FORMAT (//'        DO YOU WISH STORE DATA      Y/N  '\)
      READ (*,1910) ERROR(1)
1910  FORMAT (A10)
      IF ((ERROR(1) .EQ. 'Y').OR.(ERROR(1) .EQ. 'y')) THEN
         GOTO 1940
      ENDIF
      IF ((ERROR(1) .EQ. 'N').OR.(ERROR(1) .EQ. 'n')) THEN
         GOTO 1000
      ELSE
         WRITE (*,1915)
1915     FORMAT ('       INPUT ERROR ? ......Try again!!   '//)
         GOTO 1900
      ENDIF
C
1940  CONTINUE
C
C**** WRITE NUMBER OF DATA STORAGE CYCLES ********************************
C
      WRITE (*,*) '    No. DATA STORAGE CYCLES = ',NSAMPLE
C
C**** CREATE FILE TO STORE DATA, 'SAMPLE.DAT' ****************************
C
      OPEN (3,FILE = 'A:SAMPLE',STATUS = 'NEW')
C
      DO 600  K1 = 1,NSAMPLE
C
C**** PARAMETERS WRITEN TO FILE ARE :
C
C**** SFNCONTROL(1000),SERRFN(1000),SIFN(1000),SMNINC(1000),
C**** SEL1DOT(1000),SEL2DOT(1000),SEBTDOT(1000),SEF1(1000),SEF2(1000),
C**** SEM2(1000)
C
      WRITE (3,10)  SFNCONTROL(K1),SERRFN(K1),SIFN(K1),SMNINC(K1),
     @SEL1DOT(K1),SEL2DOT(K1),SEBTDOT(K1),SEF1(K1),SEF2(K1),
     @SEM2(K1)
10    FORMAT(10(E9.3))
C
600   CONTINUE
C
C**** CLOSE FILE *********************************************************
C
      CLOSE (3, STATUS = 'KEEP')
C
      WRITE (*,700)
700   FORMAT (//,'       WRITING TO FILE A:SAMPLE.DAT COMPLETE')
C
1000  CONTINUE
C
      END
```

APPENDIX D3

ASSEMBLER PROGRAM LISTINGS

D3.1 ASSEMBLER ROUTINE 'PORTIN'

```
;***** ENCODER SAMPLING ROUTINE ,THREE ENCODERS 16 BIT RESOLUTION
;************************** PORTIN ************************************
;
;
DATA    SEGMENT PARA PUBLIC 'DATA'
        DB      256 DUP (0)
DATA    ENDS
;
CODE    SEGMENT PARA PUBLIC 'CODE'
;
PUBLIC  PORTIN
PORTIN  PROC FAR
        ASSUME CS:CODE ; SETS UP REGISTER/SEGMENT RELATION
        ASSUME DS:DATA ;ESTAB. DATA SEG ADDRESS
;
        PUSH BP         ;SAVES CALLER FRAMEPOINTER ON STACK
                        ;[REQD. FOR SUBRTS.]
        MOV BP,SP       ;SAVES SP IN FRAMEPOINTER, BP
        PUSH BP         ;SAVES SP
;        PUSH BX         ; ALL THE FOLLOWING REGISTERS SHOULD
;        PUSH DX         ; BE SAVED BY THE CALLING FORTRAN ROUTINE
;        PUSH CX         ; BUT IF NOT SAVE THEM
;        PUSH DI
;        PUSH SI
        PUSH DS
        PUSH SS
;
INPUT1: MOV DX,0210H    ; LOADS INTEGER, BASE ADDR.
                        ;OF 8255 INTO DX REGISTER
;
        CALL TRIGGER
;
        INC  DX           ;DX = BASE+2
        IN   AL,DX        ; TAKES ENCODER MS BYTE PORT B OF NO.1 8255
;        CALL DISPCHAR
        MOV  BX,[BP+10]   ;GETS ADDR. OF 2ND. PARAM.
        MOV  DS:[BX]+1,AL  ; MOVE BYTE TO PARAM. ADDR.
                           ;(SET NATURALLY TO LS BYTE IN MEM.)
;        PUSH BX           ;SAVE ADDR. OF 2ND. PARAM.
;
        INC  DX           ;DX=BASE+3
        IN   AL,DX        ;TAKES ENCODER MS BYTE PORT A,NO.1 8255
;        CALL DISPCHAR
;        MOV  BX,[BP+14]   ;GETS ADDR. OF 1ST. PARAM
```

```
        MOV   DS:[BX]-1,AL   ;MOVE BYTE TO PARAM ADDR.
;
        MOV   AL,81H
        SUB   DX,2           ;DX = BASE+1
        OUT   DX,AL          ;OUTPUT TO PORT C GENERATES LS BYTE
;
        ADD   DX,2           ;DX. = BASE + 3
        IN    AL,DX          ;TAKES LS BYTE  PORT A,NO.1 8255
;        CALL DISPCHAR
       MOV   DS:[BX]-2,AL ;MOVES BYTE TO PARAM. ADDR.
;
       DEC   DX              ;DX = BASE+2
;        POP   BX
       IN    AL,DX          ;PORT B LS BYTE
;        CALL DISPCHAR
       MOV   DS:[BX],AL
;
;
       MOV DX,0218H    ;BASE ADDR. OF NO.3 8255
;
       CALL TRIGGER
;
       ADD  DX,2           ;DX = BASE + 3
       IN   AL,DX          ;MS BYTE PORT A, NO.3 8255
;        CALL DISPCHAR
;        MOV   BX,[BP+6]
       MOV  DS:[BX]+3,AL   ;MOVES BYTE TO PARAM. ADDR.
;
       MOV   AL,81H
       SUB   DX,2       ;DX = BASE +1
       OUT   DX,AL      ;RESETS TO READ LS BYTE
;
       ADD   DX,2
       IN    AL,DX    ;LS BYTE PORT A, NO.3 8255
;        CALL DISPCHAR
       MOV   DS:[BX]+2,AL


;
;        JMP   SHORT INPUT1 ; ONE BYTE JUMP, TO START
                           ;REPEAT SAMPLING OF PORTS
;
       POP SS
       POP DS
;        POP SI
;        POP DI
;        POP CX
;        POP DX
;        POP BX
        POP BP
        MOV SP,BP          ;RESTORES ANY UNEXPECTED
                           ;MOVEMENT OF STACK POINTER
        POP BP             ;RESTORES CALLERS FRAME POINTER
        RET 12             ;RETURNS AND ALLOWS THE THE RETURN 4BYTE ADDR.
                           ; TO BE TAKEN OFF STACK
;
; SUBROUTINE TO DISPLAY A CHARACTER ON THE SCREEN
```

```
; ENTER WITH AL = CHARACTER TO BE DISPLAYED
; USES VIDEO INTERFACE IN BIOS
DISPCHAR PROC    NEAR
         PUSH BX
         MOV  BX,0
         MOV  AH,14
         INT  10H
         POP  BX
         RET
DISPCHAR ENDP
;
;ROUTINE SETS UP PORT A&B IN C OUT AND ALSO TO GIVE MS BYTE
;
TRIGGER PROC NEAR
;        DX IS SET TO BASE ADDR. = (CONTROL PORT)
         MOV AL,92H      ; SETS UP PORT NO. 1/3 8255 AS A&B IN C OUT
         OUT DX,AL       ; OUTPUTS BYTE TO CONTROL PORT
;
         INC DX          ; DX = BASE + 1  (C BYTE OF PORT)
         MOV AL,24H
         OUT DX,AL       ; SENDS HIGH PULSE TO PORT C, TRIGGERS
                         ;ENCODERS REGISTER
         MOV AL,0H
         OUT DX,AL       ; SENDS LOW PULSE TO PORT C,
                         ;ENABLES READ FROM ENCODER
         RET
TRIGGER ENDP
;
;
PORTIN   ENDP            ;END PROCEDURE
CODE     ENDS            ;END SEGMENT
         END
```

D3.2 ASSEMBLER ROUTINE 'OUT'

```
;****** SUBROUTINE OUT(ADDR.[WORD],NO.[BYTE]) TO BE CALLED FROM FORTRAN

DATA    SEGMENT PARA PUBLIC  'DATA'
DATA    ENDS
CODE    SEGMENT PARA PUBLIC  'CODE'
;
PUBLIC  OUT
OUT     PROC    FAR
        ASSUME  CS:CODE
        ASSUME  DS:DATA
;
        PUSH    BP
        MOV     BP,SP
        PUSH    BP          ;SAVES SP ON STACK
        PUSH    DS
        PUSH    SS
;
        MOV     BX,[BP+10]  ;TAKES PORT PARAM. ADDR.
        MOV     DX,DS:[BX]  ;LOADS INTEGR. PARAM. INTO DX
        MOV     AX,0
        MOV     BX,[BP+6]
        MOV     AL,DS:[BX] ;MOVES BYTE TO BE OUTPUT INTO AL
        OUT     DX,AL       ;OUTPUT BYTE AT PORT DX
;
        POP     SS
        POP     DS
        POP     BP
        MOV     SP,BP
        POP     BP
;
        RET     8           ;REMOVES THE LAST 8 BYTES PARAM.
                            ;ADDRS.FROM STACK
OUT     ENDP
CODE    ENDS
        END
```

D3.3 ASSEMBLER ROUTINE 'INP'

```
;******* FUNCTION INP(ADDR.[WORD]) TO BE CALLED FROM FORTRAN ***********
DATA    SEGMENT  PARA PUBLIC  'DATA'
DATA    ENDS
DGROUP  GROUP DATA
CODE    SEGMENT 'CODE'
        ASSUME   CS:CODE
        ASSUME   DS:DGROUP
;
PUBLIC  INP
;
INP     PROC     FAR
;
        PUSH     BP
        MOV      BP,SP
        PUSH     BP          ;SAVES SP ON STACK
        PUSH     DS
        PUSH     SS
;
        MOV      BX,[BP+6]   ;TAKE PARAM. ADDR.
        MOV      DX,DS:[BX]  ;LOADS INTEGR. PARAM. INTO DX
        MOV      AX,0
        IN       AL,DX       ;INPUT BYTE FROM PORT DX
        MOV      DX,0        ;ALLOWS INTEGER*4 TO BE USED
;
        POP      SS
        POP      DS
        POP      BP
        MOV      SP,BP
        POP      BP
;
        RET      4           ;REMOVES THE LAST 4 BYTE PARAM. FROM STACK
INP     ENDP
CODE    ENDS
        END
```

D3.4 ASSEMBLER ROUTINE 'IADCON'

```
;***** SAMPLING ANALOGUE TO DIGITAL CONVERTER BOARD ROUTINE *****
;************** LAB-MASTER 12 BIT ACCURACY ****************************
;************************** IADCON FUNCTION*2 DEFINITION **********
;
;
STACK   SEGMENT PARA STACK 'STACK'
        DB      25 DUP (0)
STACK   ENDS            ; SACK SEG. ENDS
;
DATA    SEGMENT PARA PUBLIC 'DATA'
        DB      256 DUP (0)
DATA    ENDS
;
CODE    SEGMENT PARA PUBLIC 'CODE'
;
PUBLIC  IADCON
IADCON  PROC FAR
        ASSUME CS:CODE ; SETS UP REGISTER/SEGMENT RELATION
        ASSUME DS:DATA ; ESTAB. DATA SEG ADDRESS
        ASSUME ES:DATA ; ESTAB. EXTRA SEG. ADDRESS TO DATA
;
        PUSH BP         ;SAVES CALLER FRAMEPOINTER ON STACK
                        ;[REQD. FOR SUBRTS.]
        MOV BP,SP       ;SAVES SP IN FRAMEPOINTER, BP
        PUSH BP         ;SAVES SP
;        PUSH BX         ; ALL THE FOLLOWING REGISTERS SHOULD
;        PUSH DX         ; BE SAVED BY THE CALLING FORTRAN ROUTINE
;        PUSH CX         ; BUT IF NOT SAVE THEM
;        PUSH DI
;        PUSH SI
        PUSH DS
        PUSH SS
;
INPUT1:  MOV DX,1808     ; LOADS INTEGER, BASE ADDR. OF ADC. CARD
                         ;INTO DX REGISTER
         ADD  DX,4       ; ADD 4 TO OBTAIN BASE ADDR. OF ADC.
                         ;PORTS (DX+4)
         MOV  AL,128     ; PUT A '1' IN BIT 7 OF CONTROL PORT,
                         ;SETS ADC TO DESCRETE SAMPLING
         OUT  DX,AL      ; OUTPUTS BYTE TO CONTROL PORT OF ADC.
;
         INC  DX         ; DX+5, PORT ADDR. OF 'CHANNEL TO BE SAMPLED'
;
         MOV  BX,[BP+6]  ; GETS DATA OFFSET ADDR. OF
                         ;CHANNEL No. PARAMETER
         MOV  AL,DS:[BX] ; MOVES BYTE PARAMETER INTO AX REGISTER
;         CALL DISPCHAR   ; DISPLAYS ASCII CHARACTER
;
;
         OUT  DX,AL      ; OUTPUT ADC. CHANNEL No. TO PORT
;
         INC  DX         ; DX+6, START CONVERSION PORT
;
```

```
        OUT   DX,AL         ; OUTPUT ANYTHING TO THIS PORT TO
                            ;START CONVERSION
; NOW TEST AND WAIT TILL CONVERSION IS COMPLETED
;
        SUB   DX,2          ; DX+6, TAKES US BACK TO CONTROL PORT
;
WAIT:   IN    AL,DX         ; LOOKING FOR '1' IN BIT 7 OF CONTROL BYTE
;
        TEST AL,128         ; TESTING FOR BIT 7, 'ZF=1' WHEN NOT FOUND
;        CALL DISPCHAR
        JZ    WAIT          ; JUMPS TO LABLE 'WAIT' WHEN ZF=1
;        CALL DISPCHAR
;
; CONVERSION COMPLETE. PROCEED
;
        INC   DX            ; DX+5, LOW  BYTE PORT ADDR.
        IN    AL,DX         ; GETS LOW BYTE FROM ADC.
;        CALL DISPCHAR
        MOV   CL,AL         ; STORING LOW AL BYTE IN CL
;
        INC   DX            ; DX+6, HIGH PORT ADDR.
        IN    AL,DX         ; GETS HIGH BYTE FROM ADC. ALSO RESETS ADC.
;        CALL DISPCHAR
        MOV   AH,AL         ; MOVE BYTE INTO HIGH BYTE READY FOR TRANSFER
        MOV   AL,CL         ; MOVE LOW BYTE BACK INTO AL FOR RETURN VALUE
        MOV   DX,0          ; MOVE 0 INTO DX FOR RETURN
;
;
        POP SS
        POP DS
;        POP SI
;        POP DI
;        POP CX
;        POP DX
;        POP BX
        POP BP
        MOV SP,BP           ;RESTORES ANY UNEXPECTED MOVEMENT
                            ;OF STACK POINTER
        POP BP              ;RESTORES CALLERS FRAME POINTER
        RET 4               ;RETURNS AND ALLOWS THE THE RETURN 4 BYTE ADDR.
                            ;TO BE TAKEN OFF STACK TOGETHER
                            ;WITH 4 BYTE PARAMETER
;
; SUBROUTINE TO DISPLAY A CHARACTER ON THE SCREEN
; ENTER WITH AL = CHARACTER TO BE DISPLAYED
; USES VIDEO INTERFACE IN BIOS
DISPCHAR PROC    NEAR
        PUSH AX
        PUSH BX
        MOV  BX,0
        MOV  AH,14
        INT  10H
        POP  BX
        POP  AX
        RET
DISPCHAR ENDP
```

```
;
IADCON   ENDP                ;END PROCEDURE
CODE     ENDS                ;END SEGMENT
         END
```

D3.5 ASSEMBLER ROUTINE 'WRDOUT'

```
;***** TRANSFERS REAL NUMBERS FROM COMPUTER TO COMPUTER *********
;************* SINGLE PRECISION REALS (32 BIT ) TRANSFER *********
;*********** TRANSFERS 16 BITS AT A TIME (WORD) ********************
;******************** SUBROUTINE WRDOUT(REAL No.) ********************
;***************** RUNS ON TRANSMITTING COMPUTER  ********************
;
;
STACK   SEGMENT PARA STACK 'STACK'
        DB      25 DUP (0)
STACK   ENDS            ; SACK SEG. ENDS
;
DATA    SEGMENT PARA PUBLIC 'DATA'
        DB      256 DUP (0)
DATA    ENDS
;
CODE    SEGMENT PARA PUBLIC 'CODE'
;
PUBLIC  WRDOUT
WRDOUT  PROC FAR
        ASSUME CS:CODE ; SETS UP REGISTER/SEGMENT RELATION
        ASSUME DS:DATA ; ESTAB. DATA SEG ADDRESS
        ASSUME ES:DATA ; ESTAB. EXTRA SEG. ADDRESS TO DATA
        ASSUME SS:DATA
;
        PUSH BP         ;SAVES CALLER FRAMEPOINTER ON STACK
                        ;[REQD. FOR SUBRTS.]
        MOV BP,SP       ;SAVES SP IN FRAMEPOINTER, BP
        PUSH BP         ;SAVES SP
;        PUSH BX         ; ALL THE FOLLOWING REGISTERS SHOULD
;        PUSH DX         ; BE SAVED BY THE CALLING FORTRAN ROUTINE
;        PUSH CX         ; BUT IF NOT SAVE THEM
;        PUSH DI
;        PUSH SI
;        PUSH DS
;        PUSH SS
;
INPUT1:  MOV DX,0532     ; BASE ADDR. OF No. 2 8255 INTO DX REGISTER
;
         MOV  AL,129     ; CONTROL BYTE TO SET 8255 AS: B PORT OUT,
                         ;A PORT OUT
                         ; PORT C 0-3 AS IN & C 5-7 AS OUT
;
         OUT  DX,AL      ; OUTPUTS BYTE TO CONTROL PORT OF 8255
;
         MOV  BX,[BP+6]  ; GET ADDR. OF REAL PARAMETER TO BE TRANSFERED
         MOV  CX,02      ; INITIATE CX LOOP COUNTER
;
         INC  DX         ; DX+1, 'STATUS' PORT C ADDR.
;
; OUTPUT DATA NOT VALID SIGNAL, C PORT BIT 4 LOW BIT 5 HIGH
;
         MOV AL,32       ; BIT 5 HIGH 4 LOW
```

```
          OUT DX,AL          ; OUTPUT DATA NOT VALID SIGNAL TO C PORT,
                             ;OUTPUTS LATCH
;
NEXT:     ; SEND NEXT 2 BYTES OF REAL No.
;
;
; LOOK FOR 'READY FOR DATA SIGNAL'
;
READY:    IN AL,DX           ; INPUT STATUS OF C PORT, IF READY,
                             ;0 IS HIGH 1 IS LOW
          XOR AL,01          ; INVERTS BIT 0 OF C PORT, IF READY 0
                             ;& 1 ARE BOTH LOW
          TEST AL,03
          JNZ  READY         ; JUMPS IF EITHER BIT 0 OR 1 IS HIGH,
                             ; RECEIVER IS READY
;
; IF READY FOR DATA PROCEED
;
; PROCEED TO TRANSFER WORD AT A TIME
;
          MOV  AX,DS:[BX]    ; MOVES WORD PARAMETER INTO AX
                             ; REGISTER FROM MEMORY
;
          INC  DX            ; DX+2, PORT B ADDR.
;
          OUT  DX,AL         ; OUTPUT BYTE TO PORT B
;
;          CALL DISPCHAR     ; DISPLAYS ASCII CHARACTER OF AL REG.
;
          INC  DX            ; DX+3, PORT A ADDR.
;
          MOV  AL,AH         ; MOVES MS. BYTE OF WORD TO AL REGISTER
;
          OUT  DX,AL         ; OUTPUTS MS. BYTE TO PORT A
;
;          CALL DISPCHAR
;
          SUB  DX,02         ; DX+1, RESTORES VALUE TO ADDR. PORT C
;
          MOV  AL,16         ; OUTPUT 'DATA VALID SIGNAL', BIT 4 HIGH 5 LOW
          OUT  DX,AL         ; OUTPUTS BYTE
;
          ADD  BX,02         ; INCRIMENT BX TO FETCH NEXT M.S.
                             ; WORD OF REAL No.
;
; LOOK TO SEE IF RECEIVER HAS RECEIVED DATA, LOOKS FOR NOT READY SIGNAL
;
NREADY:   IN   AL,DX         ; INPUT FROM PORT C
          XOR  AL,02         ; INVERT BIT 1, SHOULD BE HIGH IF NOT READY
          TEST AL,03         ; CHECKING BITS 1 & 0 TO BE LOW
          JNZ  NREADY        ; JUMP IF DATA HAS NOT BEEN READ BY RECEIVER
;
; OUTPUT DATA NOT VALID SIGNAL, C PORT BIT 4 LOW BIT 5 HIGH
;
          MOV AL,32          ; BIT 5 HIGH 4 LOW
          OUT DX,AL          ; OUTPUT DATA NOT VALID SIGNAL TO C PORT,
```

```
                                ; OUTPUTS LATCH
;
        LOOP NEXT               ; CX = CX-1, LOOP IF NOT ZERO
;
;
;       POP SS
;       POP DS
;       POP SI
;       POP DI
;       POP CX
;       POP DX
;        POP BX
        POP BP
        MOV SP,BP       ;RESTORES ANY UNEXPECTED MOVEMENT OF
                        ;STACK POINTER
        POP BP          ;RESTORES CALLERS FRAME POINTER
        RET 4           ;RETURNS AND ALLOWS THE THE RETURN 4 BYTE ADDR.
                        ;TO BE TAKEN OFF STACK TOGETHER WITH 4 BYTE
                        ;PARAMETER
;
; SUBROUTINE TO DISPLAY A CHARACTER ON THE SCREEN
; ENTER WITH AL = CHARACTER TO BE DISPLAYED
; USES VIDEO INTERFACE IN BIOS
DISPCHAR PROC   NEAR
        PUSH AX
        PUSH BX
        MOV  BX,0
        MOV  AH,14
        INT  10H
        POP  BX
        POP  AX
        RET
DISPCHAR ENDP
;
WRDOUT  ENDP            ;END PROCEDURE
CODE    ENDS            ;END SEGMENT
        END
```

D3.6 ASSEMBLER ROUTINE 'WRDOUT'

```
;*** INPUT TRANSFER OF REAL NUMBERS FROM COMPUTER TO COMPUTER ***
;************** SINGLE PRECISION REALS (32 BIT ) TRANSFER *********
;************** TRANSFERS 16 BITS AT A TIME (WORD) *****************
;********************* SUBROUTINE WRDIN(REAL No.) ******************
;******** RUNS ON RECEIVING COMPUTER ***********************************
;
;
STACK   SEGMENT PARA STACK 'STACK'
        DB      25 DUP (0)
STACK   ENDS            ; STACK SEG. ENDS
;
DATA    SEGMENT PARA PUBLIC 'DATA'
        DB      256 DUP (0)
DATA    ENDS
;
CODE    SEGMENT PARA PUBLIC 'CODE'
;
PUBLIC  WRDIN
WRDIN   PROC FAR
        ASSUME CS:CODE ; SETS UP REGISTER/SEGMENT RELATION
        ASSUME DS:DATA ; ESTAB. DATA SEG ADDRESS
        ASSUME ES:DATA ; ESTAB. EXTRA SEG. ADDRESS TO DATA
        ASSUME SS:DATA ;
;
        PUSH BP         ;SAVES CALLER FRAMEPOINTER ON STACK
                        ;[REQD. FOR SUBRTS.]
        MOV BP,SP       ;SAVES SP IN FRAMEPOINTER, BP
        PUSH BP         ;SAVES SP
;
;        PUSH AX
;        PUSH BX         ; ALL THE FOLLOWING REGISTERS SHOULD
;        PUSH DX         ; BE SAVED BY THE CALLING FORTRAN ROUTINE
;        PUSH CX         ; BUT IF NOT SAVE THEM
;        PUSH DI
;        PUSH SI
;        PUSH DS
;        PUSH SS
;
INPUT1:  MOV DX,0532     ; BASE ADDR. OF 8255 No. 2, INTO DX REGISTER
;
         MOV  AL,154     ; CONTROL BYTE TO SET 8255 AS: A PORT IN,
                         ; B PORT IN
                         ; C PORT LOWER OUT & CPORT UPPER IN
;
         OUT  DX,AL      ; OUTPUTS BYTE TO CONTROL PORT OF 8255
;
         MOV  BX,[BP+6]  ; GET ADDR. OF REAL PARAMETER TO BE TRANSFERED
         MOV  CX,02      ; INITIATE CX LOOP COUNTER
;
         INC  DX         ; DX+1, 'STATUS' PORT C ADDR.
;
NEXT:    ; GET NEXT WORD OF REAL No.
```

```
;
; OUTPUT 'READY FOR DATA'
;
        MOV   AL,01         ; OUTPUT BIT 0 HIGH & 1 LOW, INDICATES
                            ; READY FOR DATA
        OUT   DX,AL         ; OUTPUTS BYTE TO C PORT
;
; CHECK TO IF DATA IS VALID LOOK AT BITS 4 TO BE HIGH & 5 TO BE
; LOW ON C PORT
;
VALID:  IN    AL,DX         ; INPUT PORT C STATUS
        XOR   AL,16         ; INVERT C4 FROM HIGH TO LOW
        TEST  AL,48         ; TESTS BITS 4 & 5 TO BE LOW
        JNZ   VALID         ; JUMPS IF DATA IS NOT VALID
;
; DATA IS VALID PROCEED
;
; PROCEED TO TRANSFER 2 BYTES AT A TIME
;
        ADD   DX,02         ; DX+3, PORT A ADDR.
;
        IN    AL,DX         ; INPUTS MS. BYTE FROM PORT A
;
;        CALL DISPCHAR    ; DISPLAYS ASCII CHARACTER
;
        MOV   AH,AL         ; PUTS MS. BYTE IN ACCU. HIGH REGISTER
;
        DEC   DX            ; DX+2, PORT B ADDR.
;
        IN    AL,DX         ; INPUTS LS. BYTE FROM PORT B
;
;         CALL  DISPCHAR
;
        MOV   DS:[BX],AX ; MOVES WORD PARAMETER INTO MEMORY @ BX ADDR.
;
        DEC   DX            ; DX+1, RESTORES VALUE TO ADDR. PORT C
;
; OUTPUT 'NOT READY FOR DATA', INDICATES DATA RECEIVED
;
        MOV   AL,02         ; NOT READY FOR DATA, 1 IN BIT 1,
                            ; 0 IN BIT 0 OF PORT C
        OUT   DX,AL         ; OUTPUTS BYTE TO PORT C
;
        ADD   BX,2          ; INCRIMENT BX TO SEND NEXT M.S. WORD
                            ; OF REAL No. TO MEMORY
;
; LOOK FOR 'DATA NOT VALID' FROM TRANSMITTER, INDICATES END OF CYCLE
; LOOK FOR BITS 4 TO BE LOW AND 5 TO BE HIGH
;
NVALID: IN    AL,DX         ; INPUT PORT C, LOOK FOR C4 TO BE LOW, C5 HIGH
        XOR   AL,32         ; INVERT C5, SHOULD NOW BE LOW
        TEST  AL,48         ; TESTS BITS C4 & C5 TO BE LOW
        JNZ   NVALID        ; JUMPS IF DATA IS STILL VALID
;
        LOOP  NEXT          ; CX = CX-1, LOOP IF NOT ZERO
;
```

```
;
;           POP SS
;           POP DS
;           POP SI
;           POP DI
;           POP CX
;           POP DX
;           POP BX
;           POP AX
;
          POP BP
          MOV SP,BP       ;RESTORES ANY UNEXPECTED MOVEMENT OF
                          ;STACK POINTER
          POP BP          ;RESTORES CALLERS FRAME POINTER
          RET 4           ;RETURNS AND ALLOWS THE THE RETURN 4 BYTE ADDR.
                          ;TO BE TAKEN OFF STACK TOGETHER WITH 4
                          ;BYTE PARA. ADDR.
;
; SUBROUTINE TO DISPLAY A CHARACTER ON THE SCREEN
; ENTER WITH AL = CHARACTER TO BE DISPLAYED
; USES VIDEO INTERFACE IN BIOS
DISPCHAR PROC    NEAR
          PUSH AX
          PUSH BX
          MOV  BX,0
          MOV  AH,14
          INT  10H
          POP  BX
          POP  AX
          RET
DISPCHAR ENDP
;
WRDIN     ENDP              ;END PROCEDURE
CODE      ENDS              ;END SEGMENT
          END
```

APPENDIX E

LAYOUT DRAWINGS OF EXPERIMENTAL RIG