POSITION AND FORCE CONTROL OF DIRECT DRIVE ROBOT ARMS

BY

MOHAMMAD SAHIRAD M.ENG. B.SC. (ENG.)

A THESIS SUBMITTED FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

OF THE UNIVERSITY OF LONDON

AND THE DIPLOMA OF IMPERIAL COLLEGE

DECEMBER 1987

MANUFACTURING TECHNOLOGY SECTION

DEPARTMENT OF MECHANICAL ENGINEERING

IMPERIAL COLLEGE OF SCIENCE AND TECHNOLOGY

LONDON SW7 2BX

ENGLAND

ABSTRACT

A direct-drive arm is a mechanical arm in which the joints are directly coupled to the rotors of high performance motors. Since the arm does not contain any gears or transmission mechanism between the motors and their loads, the drive systems have no backlash and low friction, all of which are desirable for fast, accurate and versatile robots.

In a conventional serial linkage arm, dynamic interactions exist between the arm links, which act as disturbance loads on the actuators. This causes significant tracking errors. The direct-drive arm at Imperial College has been designed and built, based on a parallel linkage mechanism in which direct drive is achieved for the arm's shoulder and elbow joints using two stationary d.c. motors. Apart from making the moving parts of the arm lighter, it has been calculated that by selecting a certain mass distribution within the arm, invariant and decoupled effective arm inertias may be realised.

With the increasing acceptance of industrial robots by the industry the sophistication of the tasks that they are demanded to perform has increased too. In particular the emphasis has been shifted from pick-and-place operations to automating assembly tasks, in which mating of closely fitting parts is required. Because the robots have poor ability to deal with task uncertainties, current installations typically rely on the extremely high positioning accuracy of the robot arm and on the accurate positioning of the workpiece itself.

Sensory feedback is an obvious answer to the problem of robotic assembly. While

vision is considered to be of limited value in this context due to its limited resolution, properly used force feedback seems to be much more important.

When compared to the conventionally designed robots, direct drive robots have the following advantages:

-High speed

-Compliance can be easily controlled by varying controller gain.

-Force exerted by the gripper can be easily related to the current armature via the Jacobian matrix.

With these qualities in hand it becomes possible to construct a reliable and inexpensive high performance robotic system. In such a system the robotic arm would be designed to be light and fast but simple in construction, while the complications of sensory control will be taken on by the controller software. In this way the system would not have to rely on sophisticated and expensive sensors for its operation.

The present work deals with the hardware, software and control strategies required for the implementation of such a system.

"In fact, whenever precision is required man flies to the machine at once, as far preferable to himself. Our sum-engines never drop a figure, nor our looms a stitch; the machine is brisk and active, when the man is weary; it is clear-headed and collected, when the man is stupid and dull; it needs no slumber, when man must sleep or drop; ever at its post, ever ready for work, its alacrity never flags, its patience never gives in; its might is stronger than the combined hundreds...."

Samuel Butler (Erewhon, 1872)

ACKNOWLEDGEMENTS

First of all, I would like to thank my supervisor, Dr. C. B. Besant for his help, encouragement and support during the project. Also, I would like to thank Dr. H.A. Pak, now in the U.S.A. who provided me with the stimulus for much of the work, designed the arm and guided me before his departure. The final member of staff I wish to acknowledge is Dr. M. Ristic who gave me much help and support in the writing of this thesis.

Discussions with my colleagues at the Nuclear Power Laboratory and other members of Imperial College were invaluable help throughout the work. Special credit goes to Dr. Peter Turner and Peter Lansiaux.

On a personal note, I would like to express my profound thanks to my parents for their interest in and encouragement of my education throughout the years.

Finally, I would like to thank the Committee of Vice Chancellors and Principals and the Department of Mechanical Engineering for providing the funding to carry out the research.

CONTENTS

	Page
ABSTRACT	2
ACKNOWLEDGEMENTS	5
CONTENTS	6
LIST OF ABBREVIATIONS	10
LIST OF FIGURES AND TABLES	11
1 INTRODUCTION	14
1.1 Assembly direction breakdown	14
1.2 General arm configurations	15
1.2.1 General arm configurations for assembly robots	19
1.3 Research objectives and the structure of the thesis	21
2 KINEMATICS	22
2.1 Overview	22
2.1.1 Direct kinematic analysis of a two d.o.f. serial drive arm	23
2.1.2 Inverse kinematics of the two d.o.f. serial arm	25
2.2.1 Direct kinematics of the two d.o.f. parallelogram arm	26
2.2.2 Inverse kinematics of the two d.o.f. parallelogram arm	27
2.3 End-effector trajectory	29
2.3.1 Long move	30
2.3.2 Short move	31

Page

3 DYNAMICS 32

3.1 A comparison of direct-drive robotic arms and conventional robots 32

3.2 Lagrange and Newton-Euler methods 35

3.2.1 Dynamic Analysis of serial drive manipulators 37
(Lagrange approach)

3.2.2 Recursive Newton-Euler approach (serial drive mechanism) 41

3.3 Parallelogram drive mechanism 46

3.3.1 Parallelogram drive solutions 47

3.3.2 The recursive Newton-Euler solutions 49
(parallelogram mechanism)

3.3.3 Dynamics of the parallelogram drive mechanism (Lagrange) 52

3.4 Invariant and decoupled arm dynamics 53

4 SYSTEM IMPLEMENTATION 56

4.1 Introduction 56

4.2 Hardware details 56

4.2.1 Direct-drive arm 58

4.2.2 System's Electronics hardware 63

4.3 System software 67

4.3.1 Syntax of the PPIL 68

4.3.2 Program repertoire 72

4.3.3 Program control repertoire 72

4.3.4 Servo control repertoire 75

4.3.5 Interrupt routines 77

4.4 Examples of part programs 78

Page

5 POSITION CONTROL 83

5.1 Open loop dynamics 83

5.2 Continuous time transfer function 84

5.2.1 Simplified transfer function 86

5.3 Discrete time transfer function 88

5.4 Derivation of the state space model $G_\theta(s)$ 90

5.4.1 Derivation of the state space model from $G_\theta(z)$ 94

5.5 Evaluation of system parameter values 95

5.5.1 Evaluation of the mechanical time constants of the actuator systems 96

5.5.2 Sampling rate selection 101

5.6 Motion generation 102

5.6.1 Constant velocity motion 103

5.6.2 Parabolic motion 105

5.6.3 Simple harmonic motion 107

5.6.4 Cycloid Motion 109

5.6.5 Modified Cycloid motion 111

5.7 Robot joint control 113

5.7.1 Control strategies 114

5.7.2 Control law selection 114

5.7.3 The effect of velocity feedback 123

5.7.4 The implemented controller on the IC DD arm 126

5.8 Performance index 129

Page

6 FORCE CONTROL 130

6.1 Introduction 130

6.2 Compliance 131

6.2.1 Static servo stiffness 133

6.3 Force control model 136

6.3.1 Determination of the value of the static servo stiffness 138

6.4 Compliant peg-in-hole insertion 142

6.5 Automatic deburring 144

6.5.1 Precision deburring 144

6.5.2 Robot position uncertainties 146

6.5.3 Control strategy for deburring 146

6.6 Power dissipation 149

7 CONCLUSIONS AND FURTHER WORK 154

7.1 Conclusions 154

7.1.1 The direct-drive arm 154

7.2 Suggestions for further work 155

7.2.1 Third order model 155

7.2.2 The direct-drive arm 156

REFERENCES 160

APPENDIX 164

LIST OF ABBREVIATIONS

ADC	Analog to Digital Convertor
ASME	American Society of Mechanical Engineers
bit	Binary Digit
BLU	Basic Length Unit
ccw	counter-clockwise
CNC	Computer Numerical Control
CR	Carriage Return
cw	clockwise
DAC	Digital to Analog Convertor
d.o.f.	Degree of Freedom
IC	Imperial College
IEEE	Institute of Electrical &Electronic Engineers
MC	Motorola Corporation
M.I.T.	Massachusetts Institute of Technology
PIA	Peripheral Interface Adapter
PID	Proportional, Integral and Derivative Control
PPIL	Pick and Place Interpreter Language
RAM	Random Access Memory
ROM	Read Only Memory
SISO	Single Input, Single Output
SCARA	Selective Compliance Assembly Robot Arm
SWT	South West Technical Corporation
^C	Control-C

LIST OF FIGURES AND TABLES

		Page
1.1	Polar configuration	16
1.2	Arm and elbow configuration	17
1.3	Cylindrical configuration	17
1.4	Cartesian configuration	18
1.5	Gantry configuration	18
1.6	SCARA configuration	19
2.1	Two degree of freedom serial drive arm mechanism	24
2.2	Parallelogram arm mechanism	28
3.1	Two degree of freedom serial drive arm mechanism	38
3.2	Vector definitions between the base origin, the link origins and the centre of masses	41
3.3	Force and moment exerted on and from link i	42
3.4	Parallelogram arm mechanism	48
3.5	Variables used in recursive Newton-Euler equations	52
4.1	Photograph of the IC experimental arm, microprocessor controller, host computer and the terminal	57
4.2	Photograph of the IC experimental arm shown in detail	57
4.3	Diagram of the IC experimental robot arm	59
4.4	Dimensions and masses of the IC experimental arm arm links	60
4.5	Work space of the IC experimentsl arm	60
4.6	Layout of the computer hardware	64
4.7	Principles of decoding the encoder pulses	66
4.8	Memory map of the processor card	68
4.9	State diagram of the PPIL	70

		Page
5.1	Open loop block diagram of a single actuator system	85
5.2	Block diagram of the sampled data open loop system	89
5.3	Open loop block diagram of a single actuator system	96
5.4	Step response for joint 1	99
5.5	Step response for joint 2	100
5.6	Position, velocity and acceleration curves for constant velocity motion	104
5.7	Position, velocity and acceleration curves for the parabolic motion	106
5.8	Position, velocity and acceleration curves for simple harmonic motion	108
5.9	Position, velocity and acceleration curves for the cycloid motion	110
5.10	Position, velocity and acceleration curves for the modified cycloid motion	112
5.11	Block diagram of a single joint actuator system with velocity feed-forward	116
5.12	Block diagram of a single joint actuator control system incorporating velocity and acceleartion feed-forward	118
5.13	Profile response for joint 1 with proportional controller K_p=1	121
5.14	Profile response for joint 1 with proportional controller K_p=4	122
5.15	Block diagram of the positional controller with velocity feedback	124
5.16	Profile response for joint 1 with proportional controller and velocity feedback	125
5.17	Block diagram of the tracking controller with proportional and velocity feedback, velocity and acceleration feedforward	127
5.18	Profile response for joint 1 with proportional controller, velocity feedback, velocity and acceleration feedforward terms	128
6.1	Four bar parallel drive mechanism	135
6.2	Force control block diagram	138
6.3	Control system layout of a single joint	139

		Page
6.4	The graph of armature current vs. deflection (from a nominal position)	141
6.5	Geometric model of a burred workpiece	145
6.6	Deburring an edge	147
6.7	Mean power dissipation ratio for the parallelogram and serial drives	153
A.1	Motor specifications	166
A.2	Tachometer specifications	167
A.3	Amplifier specifications	169
A.4	Power supply board specifications	173

1 INTRODUCTION

Industry needs robots. There is abundant evidence that, in the right circumstances, industrial robots are highly cost-effective. They are capable of bringing benefits to workers and management alike. Workers can be relieved from tedious, difficult and dangerous jobs; and management can rely on a tireless, increasingly flexible mode of production. Robots do not merely make conventional industry more productive and more competitive; they also aid the development of new industrial processes and the emergence of new industries (Simons).

1.1 ASSEMBLY DIRECTION BREAKDOWN

An early study at the Massachusettes Institute of Technology by Kondoleon, analysed a small range of different non-robotically assembled products. It was discovered that in most cases, over half of the components were assembled from one dominant direction, about 20% more came from the opposite direction, and a further 10% arrived in a plane perpendicular to the two. Only the remaining 10% of the parts arrived from more complex directions, suggesting that traditionally products are predominantly stacks. It is worth noting that this often quoted study in fact involved ten products, including a refrigerator compressor, an electric saw, an electric motor, a toaster oven, a bicycle brake, and an automobile alternator.

As a result of work such as this, it is generally thought that robotic assembly should ideally be in a layer fashion from above along a vertical axis, or at least from as few directions as possible. In addition to the advantage that gravity assists the feeding and placing of parts, it has been claimed by researchers, particularly at the Charles Stark Draper

Laboratory in the early 1980's, that the approach allows the assembly to be accomp lished using fewer than six degrees-of-freedom that would usually be expected, and consequently that the robot required may be far cheaper than otherwise (Nevins and Whitney).

Although the concept of the assembly robot was first considered in research laboratories in the early seventies, it was not until mid-seventies that potential end-users such as General Motors started seriously to investigate the feasibility of performing assembly operations using programmable devices, spurred on by the need to increase mechanisation of mid-volume batch-type assembly for which changeover times were long and costly and labour expenses high. However, they soon realised that there were no commercially available robots to meet their requirements.

Consequently, a search began for the 'ideal' form for an assembly robot. It was in fact not at all clear what that form should be. For example, it was desired that the robot assembly should be fast, but if it was, it was realised it was likely to be expensive as well. Similarly, it was unclear what kind of motions and work envelope the robot should have, what payload it should be able to carry, and what repeatability it needed. Some of these questions have been answered empirically by conducting studies of existing (non-robotic) assembly systems. For example Drozda describes an early empirical result for payload requirements which came from an internal study by GM which produced the surprising statistics that 95% of all car components which they handled weigh less than three pounds. Interestingly, nearly all current assembly robots can only lift a few kilograms.

1.2 GENERAL ARM CONFIGURATIONS

In the design procedure, one of the first considerations is the actual configuration of the robot arm. There are essentially six such configurations which are commonly used, namely polar, anthropomorphic, cylindrical, cartesian, gantry and SCARA. These are shown in

Figures (1.1) to (1.6). From the diagrams, it should be noted that each configuration possesses three axes and hence three degrees of freedom (d.o.f.) so that the end effector can be located anywhere in three-dimensional space within the arm's workspace. The end effector itself will in the most general case also possess three degrees of freedom so that any orientation can be achieved. Hence a general robot will possess a total of six degrees of freedom which means that any position and any orientation can be reached in the workspace. Sometimes more d.o.f. are required if the workspace includes obstacles which must be negotiated. Similarly, not all tasks require the full six d.o.f. and so four or more commonly five are sufficient. So the choice of which configuration is to be used will depend on the arm specification and the task it will be required to perform. The following is a consideration of which configuration will be best suited to the use of robots in assembly automation.

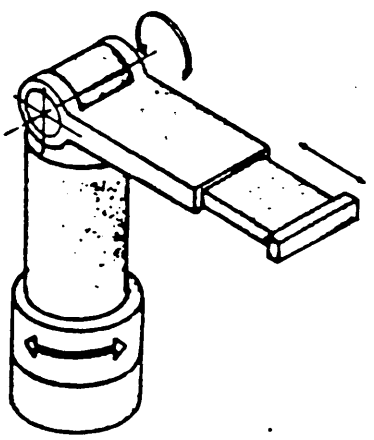


Fig. 1.1 Polar Configuration

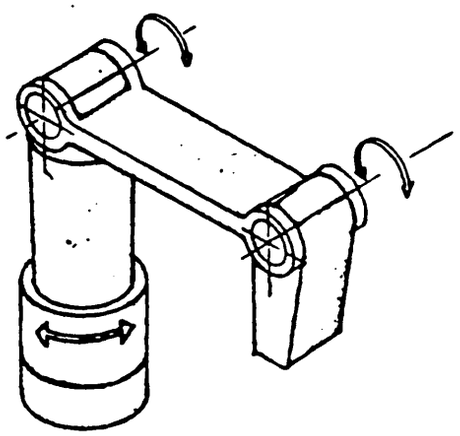


Fig. 1.2 Anthropomorphic Configuration

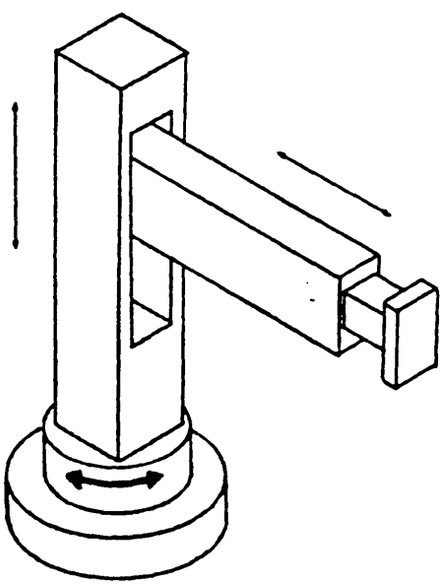


Fig. 1.3 Cylindrical Configuration

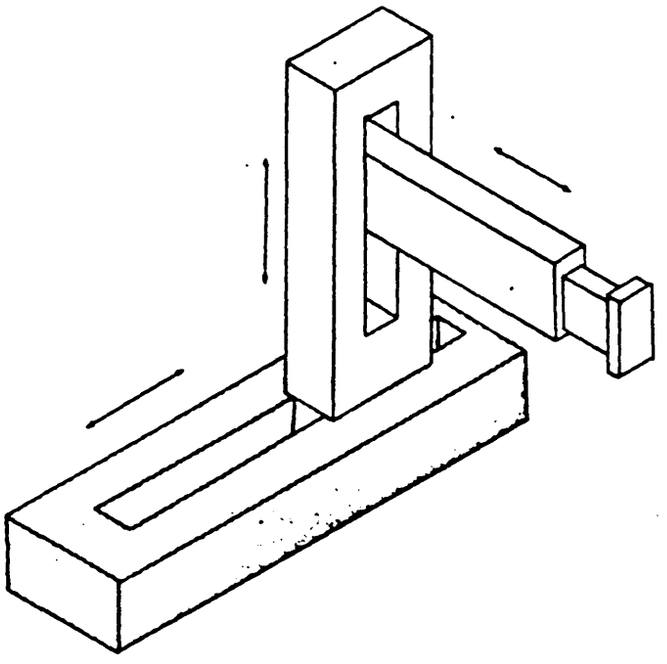


Fig. 1.4 Cartesian Configuration

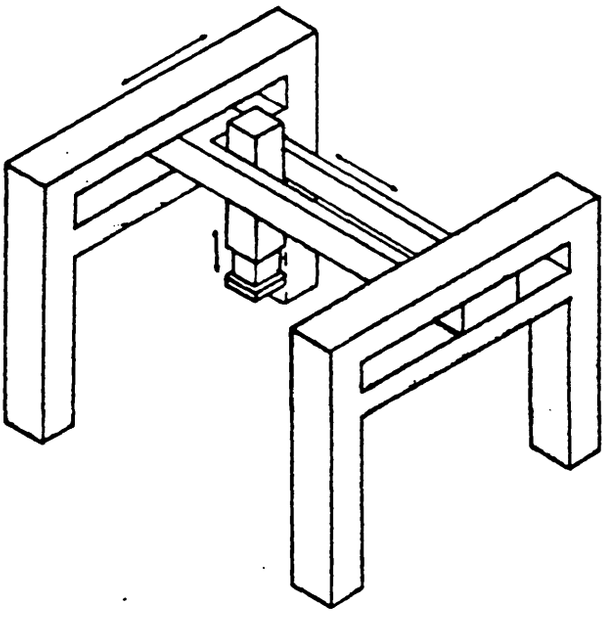


Fig. 1.5 Gantry Configuration

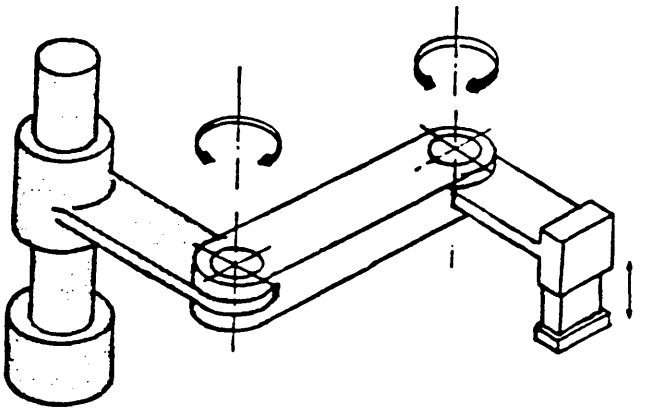


Fig. 1.6 SCARA Configuration

1.2.1 ARM CONFIGURATIONS FOR ASSEMBLY ROBOTS

Analysis of automatic assembly operations presented above, section 1.1, has shown that a single vertical movement is the most common . This can be interpreted as the simple insertion of a peg into a hole problem. The only other required motion is that found in basic pick and place operations. So the specification for an assembly robot is that it must possess general motion for positioning the component and a vertical motion for the actual assembly. All the possible configurations meet this requirement although in some cases e.g. polar and arm and elbow, the vertical motion is achieved by co-ordinated movement of more than one axis.

The cartesian configuration and hence the gantry configuration, which has identical motions, seems well suited to the assembly task as they can produce horizontal pick and

place motion together with the vertical motion. The horizontal motion is performed in straight lines due to the perpendicular axes which is an added benefit, particularly if the workstation is crowded. The SCARA configuration also seems suitable with its single axis vertical motion but the horizontal pick and place motion will be in curved arcs unless the joints are coordinated through performing kinematics calculation and interpolation is performed in world coordinate frame.

It is perhaps surprising given the above arguements to find that the most popular robot arm used for assembly work, the PUMA robot has an anthropomorphic configuration. The PUMA (Programmable Universal Manipulator for Assembly) grew out of research conducted at MIT and Stanford University. Two reasons have been cited for its popularity (Owen). Firstly the anthropomorphic configuration as the name suggests mimics the human arm so that the manual assembly workstations do not need re-designing with the introduction of the robot. Secondly, the arm has a 'front' and a 'back' which again avoids re-designing the workstation. The workpiece can be placed in front of the robot where the assembly will be performed and the components can be presented from the back.

With the increasing use of robots in industry, there is more scope to produce robots for a particular task rather than a general 6 d.o.f. multi-purpose robot. The interest shown in using robots in assembly has led to more and more SCARA robots being designed and this configuration is becoming standard for assembly robots. This can largely be attributed to the fact that cylindrical, cartesian and gantry robots require two or three prismatic joints compared to only one of SCARA robot. Hence there is more constructional complexity involved in converting the revolute motion of the motors to the required prismatic motion of the joints. The transmision mechanism used will increase the stiffness of such robots so that generally SCARA robots are best suited to light assembly work. Where the load is particularly heavy it is common to find gantry type robots which have extra supports over the cartesian robot as well as the stiffness of the prismatic joints.

When considering the possible configurations for a direct-drive robot, it is clear that all the joints must be revolute in order to avoid using some kind of transmission mechanism. This limits the choice to the arm and elbow configuration only. However, because of the advantages of the SCARA configuration, this can also be considered although the vertical motion cannot stictly be dirct-drive.

1.3 RESEARCH OBJECTIVES AND THE STRUCTURE OF THE THESIS

In recent years there has been considerable research into improving robot design and formulating new ones. As part of this radical idea, the direct-drive has been introduced which solves many of the problems of existing robots. A four-bar-link parallelogram SCARA type direct-drive robot arm has been built Imperial College of Science and Technology. The direct and inverse kinematics of the parallelogram arm and conventional serial arm are discu sed in chapter 2. A major consideration in control generally and in the control of robots in particular is the accuracy of the model used to describe the system dynamics. In chapter 3, robot dynamic analysis is discussed in detail. Chapter 4 deals with the implementation of the system and the robot language.

The purpose of the robot arm built here, is to address one of the problems following from the use of direct-drive, namely the control problem. In particular a direct-drive arm requires special attention to be paid to tracking control. Hence the research is directed to the design of a tracking controller and subsequent evaluation on the robot arm.This will be discussed in chapter 5.

Finally, the force control model, compliant motion and compliant assembly will be discussed in chapter 6, where the additional advantages of direct-drive, parallel drive mechanism become apparent.

2 KINEMATICS

2.1 OVERVIEW

In any mechanical design procedure, a kinematics analysis is essential before any dynamic analysis can be carried out. In a kinematic analysis, the position, velocity and acceleration of the manipulator tip is related to the joints' positions, velocities and accelerations.

This chapter deals with the kinematics of a two degree of freedom serial arm and also the kinematics of a four bar link parallelogram arm.

Statics is the relationship between the forces and torques that a manipulator is exerting on the environment and forces and torques at the links. This shall be discussed in chapter 6. Dynamics, however, is the relationship between the kinematics and the statics. In fact, dynamics refers to the forces and torques acting on a manipulator body that correspond to a particular kinematic state.

Direct kinematics is distinct from inverse kinematics. The direct kinematics deals with finding the position of the end of the manipulator hand given particular joint angles in polar co-ordinates. However, the inverse kinematics is set to find the joint angles for a given position of the end of the manipulator. From a computational point of view, the direct kinematic analysis is much simpler with a straight-forward solution for the unique end point of the manipulator corresponding to the given joint angles. An inverse kinematic analysis is very important to the robot control, design and trajectory planning. Direct and inverse

kinematics are related to each other since inverse kinematic velocity and acceleration relationships are usually found from their direct kinematic velocity and acceleration relationship by matrix inversions. However, the inverse kinematic solution for joint velocities and accelerations corresponding to hand velocities and accelerations is a simpler solution than the joint angle/hand position solution, even though a large matrix inversion may be required. This will be explained after the kinematic analyses are carried out.

2.1.1 DIRECT KINEMATIC ANALYSIS OF A TWO D.O.F. SERIAL ARM

The position of the end of the manipulator shown in Figure (2.1) is given by (x,y) in Cartesian space as :

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} l_1 \cos\theta_1 + & l_2 \cos(\theta_1 + \theta_2) \\ l_1 \sin\theta_1 + & l_2 \sin(\theta_1 + \theta_2) \end{bmatrix} \tag{2.1}$$

By differntiating equation (2.1), the relation between the joint velocities and the Cartesian velocities of the end effector can be written as:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} -l_1 \sin\theta_1 - & l_2 \sin(\theta_1 + \theta_2) & - l_2 \sin(\theta_1 + \theta_2) \\ l_1 \cos\theta_1 + & l_2 \cos(\theta_1 + \theta_2) & l_2 \cos(\theta_1 + \theta_2) \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \tag{2.2}$$

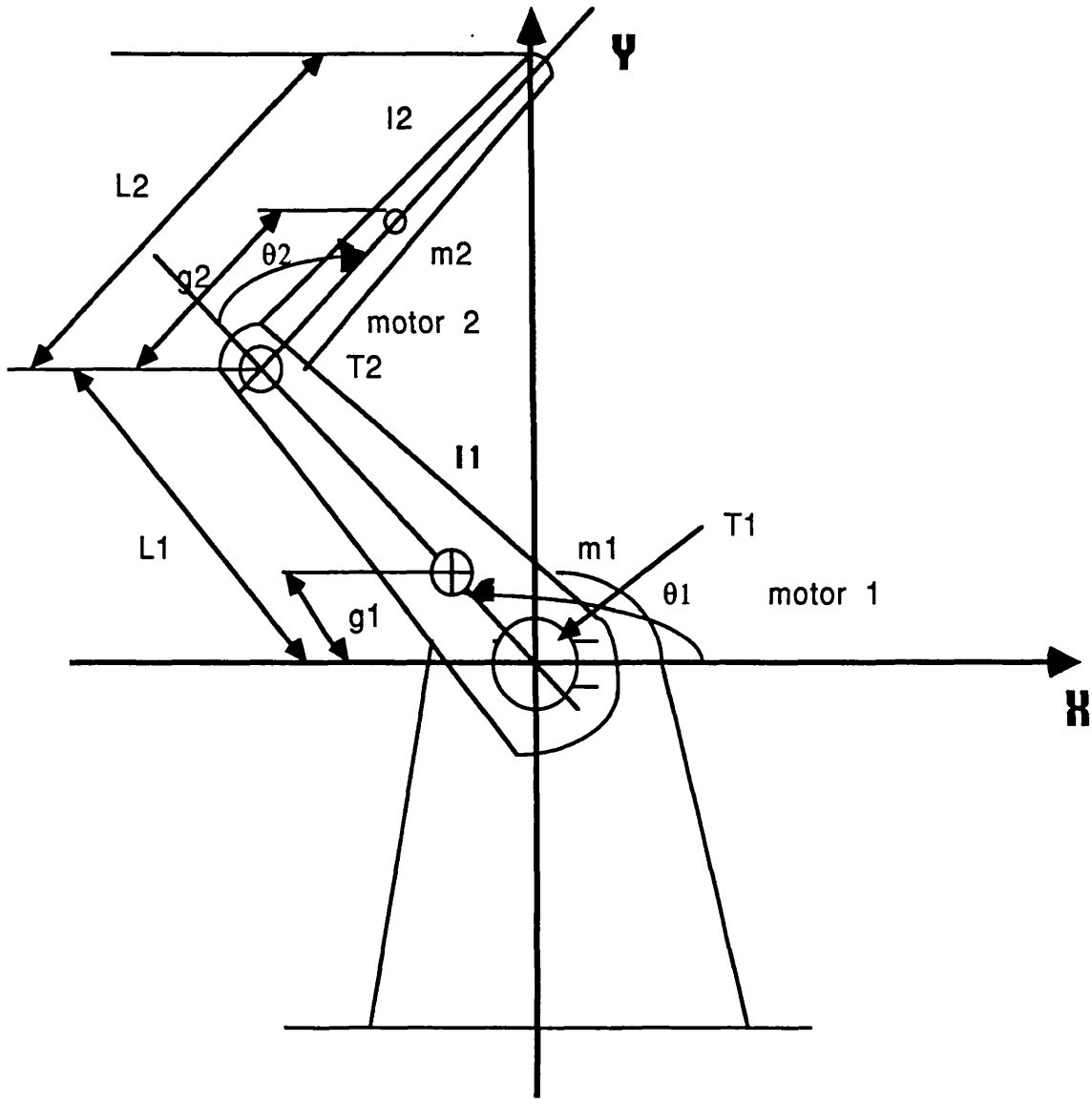


Fig. 2.1 Two Degree of Freedom Serial Drive Arm Mechanism

Again, differentiate equation (2.2) and express the acceleration equations in terms of the time derivative of θ_1 and $\theta_1+\theta_2$ to give:

$$\begin{bmatrix} \ddot{X} \\ \ddot{Y} \end{bmatrix} = \begin{bmatrix} -l_1 \sin\theta_1 & -l_2 \sin(\theta_1+\theta_2) \\ l_1 \cos\theta_1 & l_2 \cos(\theta_1+\theta_2) \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_1 + \ddot{\theta}_2 \end{bmatrix} - \begin{bmatrix} l_1 \cos\theta_1 & l_2 \cos(\theta_1+\theta_2) \\ -l_1 \sin\theta_1 & l_2 \sin(\theta_1+\theta_2) \end{bmatrix} \cdot \begin{bmatrix} \dot{\theta}_1^2 \\ (\dot{\theta}_1+\dot{\theta}_2)^2 \end{bmatrix} \quad (2.3)$$

2.1.2 INVERSE KINEMATICS OF A TWO D.O.F. SERIAL ARM

The inverse kinematics involves finding the joint angles for a given position of the end of the manipulator. In the following analysis, a solution of inverse kinematics is derived from a direct kinematic solution by matrix inversion.

The joint angles are derived from geometry of the arm as :

$$\theta_1 = \mathrm{TAN}^{-1}(Y/X) - \mathrm{TAN}^{-1}((l_2 \sin\theta_2) / ((l_1 + l_2 \cos\theta_2)) \quad (2.4)$$

$$\theta_2 = \mathrm{COS}^{-1}((X^2+Y^2-l_1^2-l_2^2)/(2l_1l_2)) \quad (2.5)$$

There are two solutions for θ_2. One solution corresponds to the 'elbow-down' configuration, the other to the 'elbow-up' configuration. By considering the quadrant in which the end of the manipulator lies, in other words the signs of X and Y, the value of θ_2 can be identified from the two solutions. However, by using TAN(a,b) trigonometrical functions, one can find a unique solution for θ_1.

By inverting the Jacobian matrix of equation (2.2) the solution of the joint velocities from the Cartesian velocities is obtained as :

$$\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} = \frac{1}{l_1 l_2 \sin\theta_2} \begin{bmatrix} l_2 \cos(\theta_1+\theta_2) & l_2 \sin(\theta_1+\theta_2) \\ -l_1\cos\theta_1 - l_2 \cos(\theta_1+\theta_2) & -l_1\sin\theta_1 - l_2 \sin(\theta_1+\theta_2) \end{bmatrix} \begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix}$$

(2.6)

The angular accelerations of the joints are derived from equation (2.3) by matrix inversion as follows:

$$\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_1 + \ddot{\theta}_2 \end{bmatrix} = \frac{1}{l_1 l_2 \sin\theta_2} \begin{bmatrix} l_2 \cos(\theta_1+\theta_2) & l_2 \sin(\theta_1+\theta_2) \\ -l_1 \cos\theta_1 & -l_1 \sin\theta_1 \end{bmatrix} \begin{bmatrix} \ddot{X} \\ \ddot{Y} \end{bmatrix}$$

$$+ \frac{1}{l_1 l_2 \sin\theta_2} \begin{bmatrix} l_1 l_2 \cos\theta_2 & l_2^2 \\ -l_1^2 & -l_1 l_2 \cos\theta_2 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1^2 \\ (\dot{\theta}_1 + \dot{\theta}_2)^2 \end{bmatrix} \tag{2.7}$$

2.2.1 DIRECT KINEMATICS OF THE TWO D.O.F. PARALLELOGRAM ARM

The position of the end of the manipulator shown in Figure (2.2) is given by (x,y) in Cartesian space as:

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} l_1 \cos\theta_1 - l_4 \cos\theta_2 \\ l_1 \sin\theta_1 - l_4 \sin\theta_2 \end{bmatrix} \tag{2.8}$$

By differentiating equation (2.8), the relation between the joint velocities and the Cartesian velocities of the end effector can be written as:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} -l_1 \sin\theta_1 & l_4 \sin\theta_2 \\ l_1 \cos\theta_1 & -l_4 \cos\theta_2 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \tag{2.9}$$

Again, differentiate equation (2.9) and express the acceleration equations in terms of the time derivative of θ_1 and θ_2 to give:

$$\begin{bmatrix} \ddot{X} \\ \ddot{Y} \end{bmatrix} = \begin{bmatrix} -l_1 \sin\theta_1 & l_4 \sin\theta_2 \\ l_1 \cos\theta_1 & -l_4 \cos\theta_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} -l_1 \cos\theta_1 & l_4 \cos\theta_2 \\ -l_1 \sin\theta_1 & l_4 \sin\theta_2 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1^2 \\ \dot{\theta}_2^2 \end{bmatrix} \quad (2.10)$$

2.2.2 INVERSE KINEMATICS OF THE TWO D.O.F. PARALLELOGRAM ARM

With the same parameters as shown in Figure (2.2), the inverse kinematic solutions are as follows :

$$\theta_1 = \text{TAN}^{-1}(Y/X) - \text{TAN}^{-1}((l_4 \sin(\theta_1 - \theta_2)) / ((l_1 - l_4 \cos(\theta_1 - \theta_2))) \quad (2.11)$$

$$\theta_2 = \pi + \theta_1 - \text{COS}^{-1}((X^2+Y^2-l_1^2-l_4^2)/(2l_1l_4)) \quad (2.12)$$

Alternatively,

$$\theta_1 = \text{COS}^{-1}((X^2+Y^2+l_1^2-l_4^2)/(2l_1(X^2+Y^2)^{1/2})) + \alpha \quad (2.13)$$

$$\theta_2 = \text{COS}^{-1}((l_1^2-X^2-Y^2 \quad -l_4^2)/(2l_4(X^2+Y^2)^{1/2})) + \alpha \quad (2.14)$$

where $\alpha = \text{TAN}^{-1}(Y/X)$ (2.15)

$$\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} = -\frac{1}{l_1 l_4 \sin(\theta_1 - \theta_2)} \begin{bmatrix} l_4 \cos\theta_2 & l_4 \sin\theta_2 \\ l_1 \cos\theta_1 & l_1 \sin\theta_1 \end{bmatrix} \begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} \quad (2.16)$$

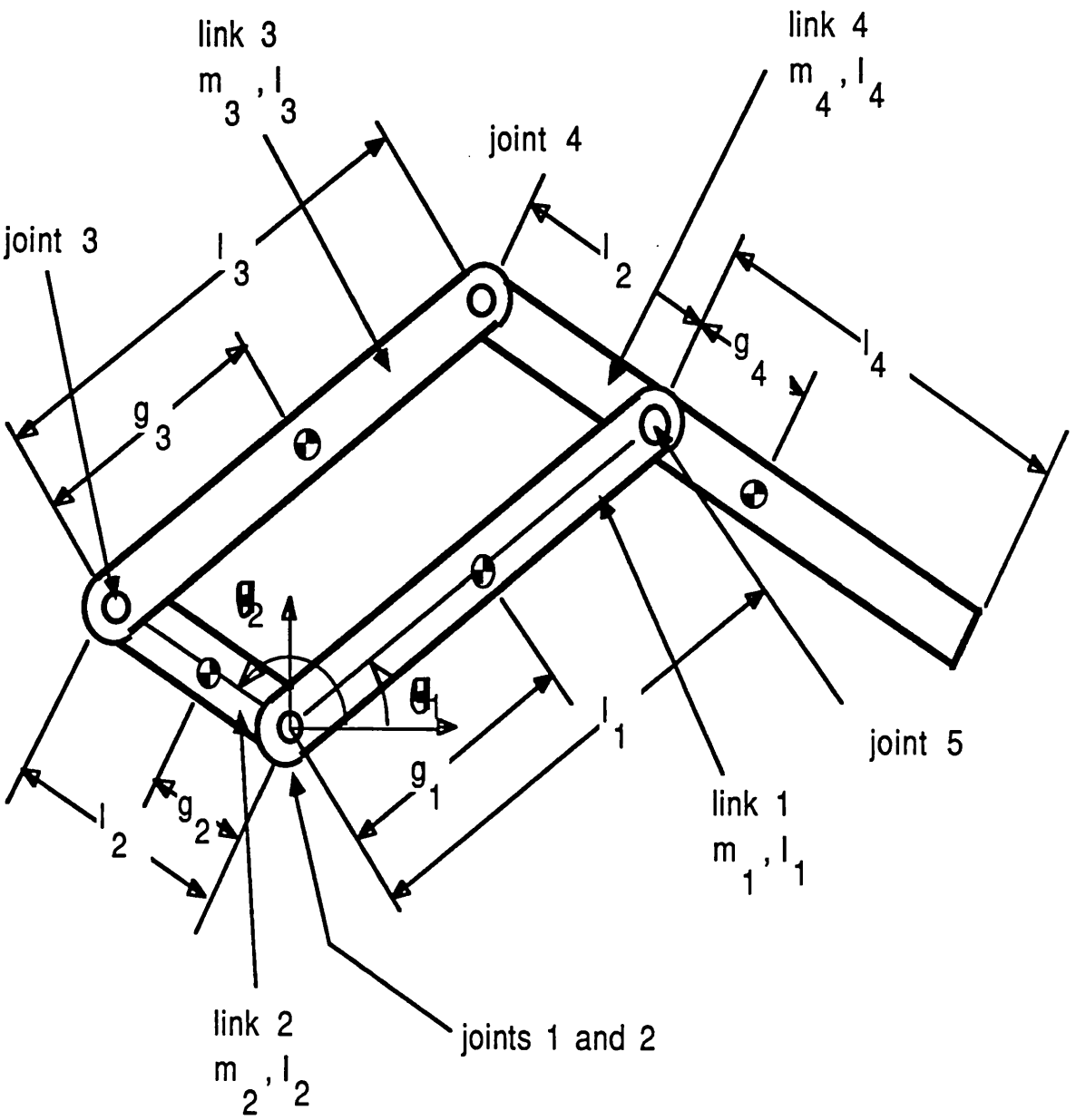


Fig. 2.2 Parallelogram Arm Mechanism

The angular accelerations of the joints are derived from equation (2.9) by matrix inversion :

$$\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = -\frac{1}{l_1 l_4 \sin(\theta_1+\theta_2)} \begin{bmatrix} l_4\cos\theta_2 & l_4\sin\theta_2 \\ l_1\cos\theta_1 & l_1\sin\theta_1 \end{bmatrix} \begin{bmatrix} \ddot{X} \\ \ddot{Y} \end{bmatrix}$$

$$+\frac{1}{l_1 l_4 \sin(\theta_1-\theta_2)} \begin{bmatrix} -l_1 l_4\cos(\theta_1-\theta_2) & l_4^{\,2} \\ -l_1^{\,2} & l_1 l_4\cos(\theta_1-\theta_2) \end{bmatrix} \begin{bmatrix} \dot{\theta}_1^{\,2} \\ \dot{\theta}_2^{\,2} \end{bmatrix} \quad (2.17)$$

The above solutions can be derived from equations (2.4) to (2.7) by replacing θ_2 with $(\theta_2-\theta_1-\pi)$, and $\dot{\theta}_2$ with $(\dot{\theta}_2 - \dot{\theta}_1)$ and $\ddot{\theta}_2$ with $(\ddot{\theta}_2 - \ddot{\theta}_1)$.

2.3 END-EFFECTOR TRAJECTORY

Straight line motion in Cartesian space is assumed in this section because many applications in robotics often involve trajectories in which the end effector is required to follow a straight-line path from the source to the destination (point A to point B).

In order to program the arm's end effector to follow a straight line, the line has to be divided into many small intervals, and at each interval, kinematic solutions are computed to fit the required constraints, i.e. maximum path velocity and acceleration. In the long move (long distance travel), a trapezium velocity profile is used and the maximum path velocity is reached. In the short move, the maximum path velocity is not reached and a triangular velocity profile is formed.

2.3.1 LONG MOVE

When the constrained path velocity is reached, the manipulator tip continues to move at a constant speed. The accelerating time, T_1, and deceleration time, T_3, are equal. They, together with the constant velocity time, T_2, form the total move-time. The path velocity vs time curve has a trapezoidal profile. Let V_{max} and A_{max} represent the constrained end effector velocity and acceleration, L represents the displacement from point A (X_a , Y_a) to point B (X_b , Y_b), θ_p represents the path gradient and X, $\dot{X}$, $\ddot{X}$, Y, $\dot{Y}$, $\ddot{Y}$ represent the kinematics of the hand in terms of cartesian space. Mathematically, they are defined as follows:

$$\theta_p = TAN^{-1}[(Y_B-Y_A)/(X_B-X_A)] \quad (2.18)$$

$$L=[(X_B-X_A)^2+(Y_B-Y_A)^2]^{1/2} \quad (2.19)$$

$$T_1=T_3= V_{max}/A_{max} \quad (2.20)$$

$$T_2= (L-V_{max}*T_1)/V_{max} \quad (2.21)$$

$$\text{Total move-time} = T_1+T_2+T_3 \quad (2.22)$$

During $0< t \leq T_1$

$$X = (A_{max}*\cos\theta_p)*t^2/2 + X_A \quad (2.23)$$

$$\dot{X} = (A_{max}*\cos\theta_p)*t \quad (2.24)$$

$$\ddot{X} = A_{max}*\cos\theta_p \quad (2.25)$$

$$Y = (A_{max}*\sin\theta_p)*t^2/2 + Y_A \quad (2.26)$$

$$\dot{Y} = (A_{max}*\sin\theta_p)*t \quad (2.27)$$

$$\ddot{Y} = A_{max}*\sin\theta_p \quad (2.28)$$

During $T_1 < t < T_1+T_2$

$$\ddot{X}=\ddot{Y}=0 \tag{2.29}$$

$$\dot{X} = (A_{max}*\cos\theta_p)*T_1 \tag{2.30}$$

$$X = (A_{max}*\cos\theta_p)*T_1^2/2 + X_A + (A_{max}*\cos\theta_p)*T_1*(t-T_1) \tag{2.31}$$

$$\dot{Y}= (A_{max}*\sin\theta_p)*T_1 \tag{2.32}$$

$$Y = (A_{max}*\sin\theta_p)*T_1^2/2 + Y_A + (A_{max}*\sin\theta_p)*T_1*(t-T_1) \tag{2.33}$$

During $T_1+T_2 < t < T_1+T_2+T_3$

$$\ddot{X} = - A_{max}*\cos\theta_p \tag{2.34}$$

$$\dot{X} = - (A_{max}*\cos\theta_p)*(t-T_1-T_2)+ (A_{max}*\cos\theta_p)*T_1 \tag{2.35}$$

$$X = - [(A_{max}*\cos\theta_p)*(t-T_1-T_2)^2]/2+[(A_{max}*\cos\theta_p)*(2t-T_1)]/2+X_a \tag{2.36}$$

$$\ddot{Y} = - A_{max}*\sin\theta_p \tag{2.37}$$

$$\dot{Y} = - (A_{max}*\sin\theta_p)*(t-T_1-T_2)+ (A_{max}*\sin\theta_p)*T_1 \tag{2.38}$$

$$Y = - [(A_{max}*\sin\theta_p)*(t-T_1-T_2)^2]/2+[(A_{max}*\sin\theta_p)*(2t-T_1)]/2+Y_a \tag{2.39}$$

2.3.2 SHORT MOVE

In the short move, the maximum path velocity is reached. Again, the accelerating time, T_1, and the decelerating time, T_3, are equal, but T_2 is now zero. Therefore, a triangular velocity profile is expected. Mathematically, the short move is defined as follows:

$$T_1=T_3=(L/A_{max})^{1/2} \tag{2.40}$$

$$T_1= 0 \tag{2.41}$$

$$\text{Total move time} = T_1 + T_3 \tag{2.42}$$

The kinematics of the arm in terms of cartesian coordinates ($X, \dot{X}, \ddot{X}$, etc.) are the same as equations (2.22) to (2.28) and (2.34) to (2.39) with t, T_1 and T_2 being substituted.

3 DYNAMICS

3.1 A COMPARISON OF DIRECT DRIVE ROBOTIC ARMS AND CONVENTIONAL ROBOTS

During the past and present decade, the choice of the robot arm configuration, the choice of different manipulator parameters, (mass, velocity, acceleration, dimensions, etc.), and the choice of the actuator sizes in the design of industrial manipulators has been freely speculated, frequently based on robot designers' experience but lacking any systematic approach. That is why so many of the industrial robots currently in use were designed with a high safety factor, especially on the actuator side. Such manipulators are not running at their best possible efficiency from the point of view of operating speed. At present, the design and control of manipulators is still much under research and development.

Dynamic analysis is also essential in designing efficient controllers for robots. The dynamic behaviour of robotic arms is highly nonlinear especially when the manipulator tip is required to move at high speed. The nonlinear behaviour occurs when the number of joints in simultaneous motion is more than one. Force interactions among the joints create couplings which complicate the control system. The couplings increase as the speed of the arm increases. Most present robots are designed to cope with movement at speeds of no more than 3 m/s. The built-in digital controller is not capable of solving full dynamic equations fast

enough in real time to send the compensating signals to reduce the steady state error which is worsened by disturbances such as frictional torques. To reduce the computational time, the dynamic equations are often simplified. In other words, the coriolis and centripetal components of the dynamic equations are neglected. However, these terms have been found to be comparable to the inertial components, especially at high speeds and accelerations. Due to this simplified method, the robotic arms move with large tracking errors. A more efficient and complete set of dynamic equations together with faster computers can improve the performance of industrial robots.

The theory of dynamic analysis can be applied to both design and control applications. In order to control the manipulator, the dynamic equations together with the control law will have to be pre-calculated to generate the control signal. In designing robots, the worst possible case of dynamics is usually considered so that a suitable arm with the appropriate sizes of actuators can be chosen. This type of analysis can be very useful not only to robot manufacturers but also to users to designate design specification for new robots or improve control algorithms using the existing robots. Due to a large number of design parameters and the complex interaction between them, computer-aided design methods are essential for all types of arm configurations. A good design formulation based on a complete dynamic model definitely helps to develop new generation robots with higher reliability, efficiency, accuracy and speed.

Apart from involving dynamics in design and control, it can also be used to plan the robot motion for minimizing the travel time without violating the joint constraints. The method of scaling down the velocities and acceleations of the fast moving joints is able to provide the optimum trajectory in which the manipulator arms should follow. The whole process is called 'trajectory planning'. The ability to plan robot motion is essential when the actuators are required to run at their full potential. In other words, a shortest possible time for a set task can be achieved. Leu, et al. have developed simulation programs which are capable of

estimating the cycle-time and planning robot motion. Again, the development of an efficient mathematical representation of manipulator dynamics is essential here.

In recent years, intensive research has been carried out to find more efficient formulations of dynamic equations governing the behaviour of robotic devices. Lagrange and Newton-Euler equations are most widely used at present in robotics. Nevertheles, these methods still suffer from certain disadvantages. The Lagrangian approach tends to lead to computational algorithms involving large number of arithmetic operations. When the manipulator arms form a closed-loop configuration, the Newton-Euler approach leads to unnecessary calculations associated with the elimination of certain forces and torques of interaction between robot links. There have been successful attempts to eliminate weaknesses of both methods so that they can be employed in real-time computations without worrying about the computing power. Recently, Kane's dynamic equations have shown that the computational time can be reduced even further by employing more 'explicit' formula expressions. However, dynamic equations for closed-loop configurations still require attention.

Most industrial robots require much higher torques and lower speeds in comparison with electrical motors' specifications. Therefore, mechanical reduction devices (or gears) are usually introduced between the motors and the links. Gears are commonly used in machines where backlash, wear, the need for lubrication and poor positioning accuracy can be tolerated. In positioning servomechanisms, however, these limitations degrade the control performance of robots. A manipulator with several axes of motion is more sensitive to disadvantages of gear reduction since multiple points of backlash will interact, resulting in an inaccurate positioning device. To minimize the error, precision anti-backlash gears must be used. Preloading of such gears leads to large frictions which can shorten their life due to wear. In addition, it is not possible to obtain accurate mathematical models for frictional torques. The mechanism must be adjusted and checked frequently to keep the backlash to a

minimum. Newly designed Harmonic Drives are now widely used in present robots. Although, much of the backlash has been eliminated in these drives, they have insufficient mechanical stiffness and significant fluctuations in torque transmission. In other words, conventional electromechanical robots with gearing have a limited performance under toque control due to friction and backlash. With the latest development of ultra high torque motors (Welburn) which are now available in the market, direct drive robots have become a reality.

A direct-drive robot has high torque motors directly coupled to the links without any gearings. Such a device has no backlash and low friction.
Therefore, the robot is capable of achieving fast and accurate positioning as well as torque control. However, there are some factors which hamper the use of this concept in robotics. Firstly, the robot is not permitted to exert a large torque or force for a long period of time without overheating the motor since the drive system has no means to amplify the motors' output torque mechanically. Secondly, in a serial-link configuration the weight of the motor is acting as a load for the next motor down the serial linkage. Careful design of the arm configuration is needed so that the mass of one of the motor does not couple to the other motor. Lastly, direct drive robots are very sensitive to load changes or disturbances. The control of direct drive robots becomes more complicated because the interactive and nonlinear torques are no longer negligible as the reduction ratio is now unity and any changes in inertia are directly reflected to the corresponding motor. However in conventional geared robots with a gear ratio of N to 1, the inetia as seen by the motor is reduced by a factor of $1/N^2$.

3.2 LAGRANGE AND NEWTON-EULER METHODS

Dynamic equations governing the behaviour of manipulator movement play a significant role in the design and control of industrial robots. In control, dynamic equations are used to

compute the forces and torques required to drive members of a manipulator according to the desired trajectories rapidly and repeatedly; hence it is essential to have the most efficient computational algorithms so that simplified or partial dynamic models are not necessary and pure compensating signals can be fed into the control loop in real time. That is why dynamic equations in robotics have been under investigation during the last decade.

Lagrange and Newton-Euler approaches have been the most frequently used methods of formulating the dynamics. Lagrange equations are established from energy considerations. They deal with velocities instead of accelerations and do not involve themselves with unknown reaction forces.Therefore, it is simple and straightforward to apply to systems with multiple degrees of freedom. A recursive set of Newton-Euler equations are found sequentially starting from the fixed base link. Then, force or torque balances at each succesive joint, starting at the manipulator tip, determine the actuator torques due to the inertial and applied loads. Both methods of dynamic formulation are applied in the following section.

The dynamic models presented in this thesis are all based on the assumption that the arm is a linkage comprising rigid bodies.

The motors' rotor inertias should not be neglected in the case where a high reduction ratio is introduced into the motors at the joints. However, in direct drive arms, the motors' rotor inertias are negligible compared to the inertias of the arm's links. Nevertheles, as is explained in chapter 5, the rotor inertia is taken into account in the case of the I.C. direct-drive arm dynamics.

3.2.1 DYNAMIC ANALYSIS OF SERIAL DRIVE MANIPULATORS (LAGRANGE APPROACH)

The Lagrange approach is a general technique for deriving the dynamics of a mechanical system. This approach is based on the energy stored in each element of the manipulator. The difference between the kinetic energy K and the potential energy P of the system is called the Lagrangian L.

$$L=K-P \tag{3.1}$$

Let F_i represent either a force or a torque depending whether q_i is a linear or an angular coordinate, q_i represents the coordinates in which the kinetic and potential energies are expressed, and $\dot{q}_i$ represent the corresponding velocity. The dynamic equations are obtained as:

$$F_i = \frac{d}{dt}\left[\frac{dL}{d\dot{q}_i}\right] - \frac{dL}{dq_i} \tag{3.2}$$

Using the same variables as shown in Figure (3.1) the kinetic energy of the upper arm can be derived directly as,

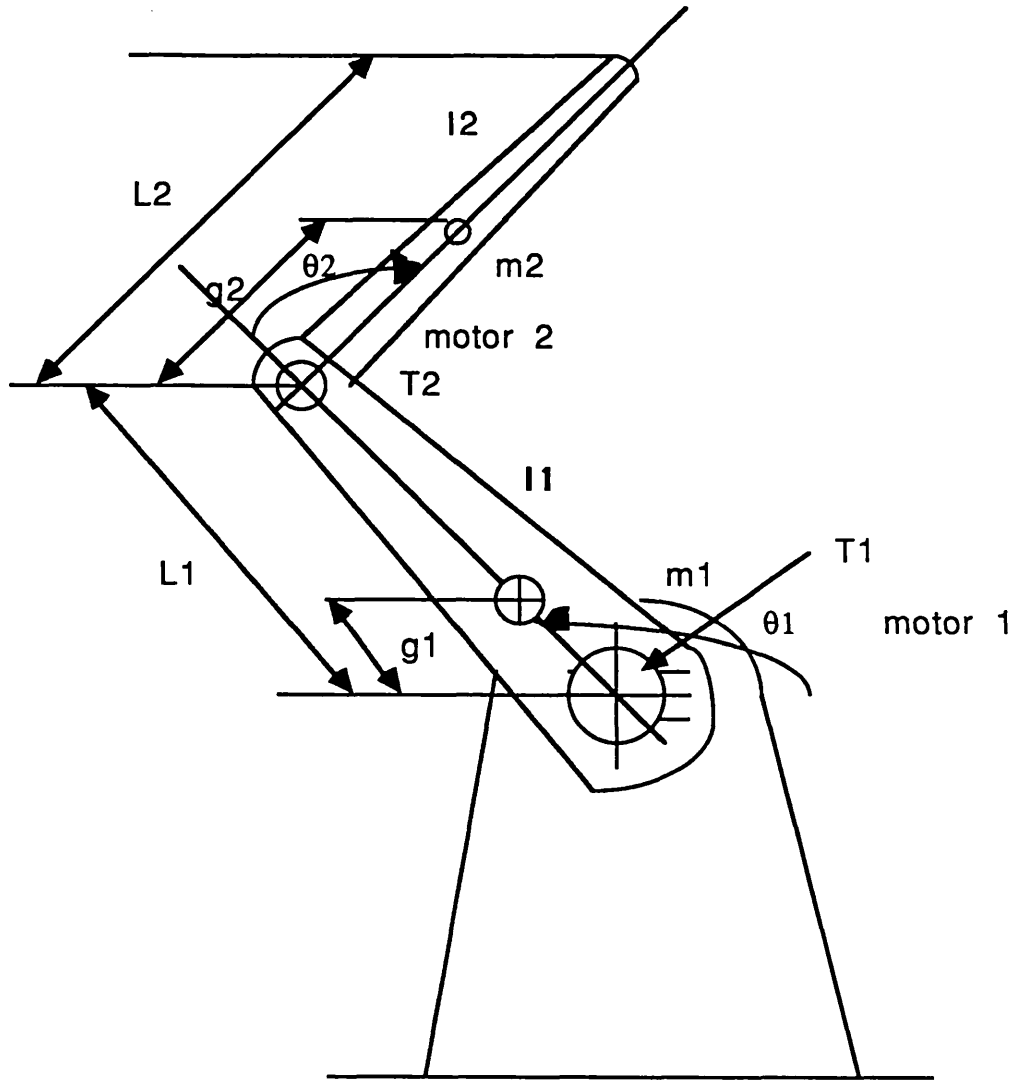


Fig. 3.1 Two d.o.f. Serial Drive Arm Mechanism

$$K_1 = \frac{1}{2} m_1 g_1^2 \dot{\theta}_1^2 + \frac{1}{2} I_1 \dot{\theta}_1^2 \qquad (3.3)$$

The potential energy, P_1, is constant as the manipulator can only move in the horizontal plane.

$$P_1=C_1 \tag{3.4}$$

The magnitude of the velocity squared of mass m_2 is:

$$V_2^2 = l_1^2 \dot{\theta}_1^2 + g_2^2 (\dot{\theta}_1^2 + 2 \dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_2^2) + 2 l_1 g_2 \cos(\theta_2) (\dot{\theta}_1^2 + \dot{\theta}_1 \dot{\theta}_2) \tag{3.5}$$

Hence the kinetic energy of the forearm is:

$$K_2 = \frac{1}{2} m_2 l_1^2 \dot{\theta}_1^2 + \frac{1}{2} m_2 g_2^2 (\dot{\theta}_1^2 + 2 \dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_2^2) +$$

$$m_2 l_1 g_2 \cos(\theta_2) (\dot{\theta}_1^2 + \dot{\theta}_1 \dot{\theta}_2) \tag{3.6}$$

Its potential energy is also constant.

$$P_2=C_2 \tag{3.7}$$

Substitute equation (3.3), (3.4), (3.6) and (3.7) into equation (3.1) to give:

$$\text{Lagrangian } L = \frac{1}{2}(m_1 g_1^2 + m_2 l_1^2 + I_1) \dot{\theta}_1^2 + \frac{1}{2} m_2 g_2^2 (\dot{\theta}_1^2 + 2 \dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_2^2)$$

$$m_2 l_1 g_2 \cos(\theta_2) (\dot{\theta}_1^2 + \dot{\theta}_1 \dot{\theta}_2) + \frac{1}{2} I_2 (\dot{\theta}_1 + \dot{\theta}_2)^2 + C_1 + C_2 \tag{3.8}$$

By differentiating the Lagrangian L according to equation (3.2):

$$\frac{dL}{d\dot{\theta}_1} = (m_1 g_1^2 + m_2 l_1^2 + I_1) \dot{\theta}_1 + m_2 g_2^2 \dot{\theta}_1 + m_2 g_2^2 \dot{\theta}_2 +$$

$$m_2 l_1 g_2 \cos(\theta_2) \dot{\theta}_2 + I_2 (\dot{\theta}_1 + \dot{\theta}_2) \tag{3.9}$$

and

$$\frac{d}{dt}\left[\frac{dL}{d\dot{\theta}_1}\right] = (I_1 + I_2 + m_1 g_1^2 + m_2 l_1^2 + 2m_2 l_1 g_2 \cos(\theta_2)) \ddot{\theta}_1 +$$

$$(I_2 + m_2 g_2^2 + m_2 l_1 g_2 \cos(\theta_2)) \ddot{\theta}_2 -$$

$$2m_2 l_1 g_2 \sin(\theta_2) \dot{\theta}_1 \dot{\theta}_2 - m_2 l_1 g_2 \sin(\theta_2) \dot{\theta}_2^2 \tag{3.10}$$

and

$$\frac{dL}{d\theta_1} = 0 \tag{3.11}$$

By substituting equations (3.11) and (3.10) into (3.2) the torque at joint 1 is given by:

$$\begin{aligned} T_1 = & (I_1 + I_2 + m_1 g_1^2 + m_2 l_1^2 + 2m_2 l_1 g_2 \cos(\theta_2))\ddot{\theta}_1 + \\ & (I_2 + m_2 g_2^2 + m_2 l_1 g_2 \cos(\theta_2))\ddot{\theta}_2 - \\ & 2m_2 l_1 g_2 \sin(\theta_2)\dot{\theta}_1\dot{\theta}_2 - m_2 l_1 g_2 \sin(\theta_2)\dot{\theta}_2^2 \end{aligned} \tag{3.12}$$

similarly,

$$\frac{dL}{d\theta_2} = m_2 g_2^2 \dot{\theta}_1 + m_2 g_2^2 \dot{\theta}_2 + m_2 l_1 g_2 \cos(\theta_2)\dot{\theta}_1 + I_2(\dot{\theta}_1 + \dot{\theta}_2) \tag{3.13}$$

and

$$\begin{aligned} \frac{d}{dt}\left[\frac{dL}{d\theta_2}\right] = & m_2 g_2^2 \ddot{\theta}_1 + m_2 g_2^2 \ddot{\theta}_2 + m_2 l_1 g_2 \cos(\theta_2)\ddot{\theta}_1 - \\ & m_2 l_1 g_2 \sin(\theta_2)\dot{\theta}_1\dot{\theta}_2 + I_2(\ddot{\theta}_1 + \ddot{\theta}_2) \end{aligned} \tag{3.14}$$

and

$$\frac{dL}{d\theta_2} = -m_2 l_1 g_2 \sin(\theta_2)(\dot{\theta}_1^2 + \dot{\theta}_1\dot{\theta}_2) \tag{3.15}$$

$$T_2 = \frac{d}{dt}\left[\frac{dL}{d\theta_2}\right] - \frac{dL}{d\theta_2} \tag{3.16}$$

The torque at joint 2 is then given by:

$$\begin{aligned} T_2 = & (I_2 + m_2 g_2^2 + m_2 l_1 g_2 \cos(\theta_2))\ddot{\theta}_1 + \\ & (I_2 + m_2 g_2^2)\ddot{\theta}_2 + m_2 l_1 g_2 \sin(\theta_2)\dot{\theta}_1^2 \end{aligned} \tag{3.17}$$

3.2.2 RECURSIVE NEWTON-EULER APPROACH (SERIAL DRIVE MECHANISM)

The Newton-Euler approach will be used to derive the dynamic equations for the serial arm mechanism. Firstly, vector definitions of variables used in the analysis have to be described. As shown in Figures (3.2) and (3.3) :

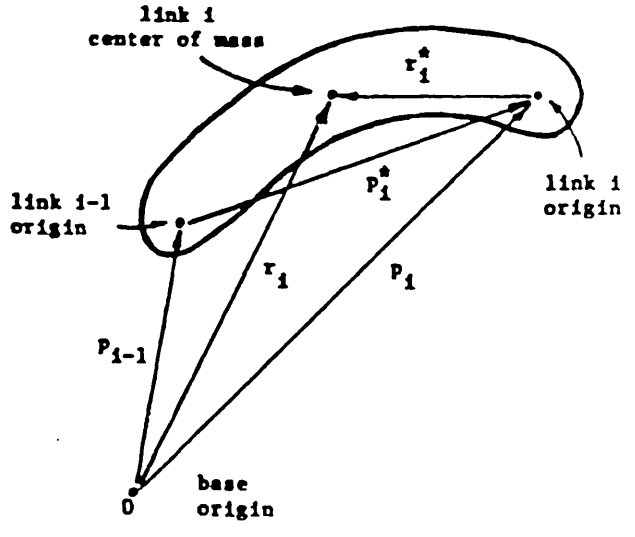


Fig.3.2 Vector definitions between the base origin, the link origins and centre of masses

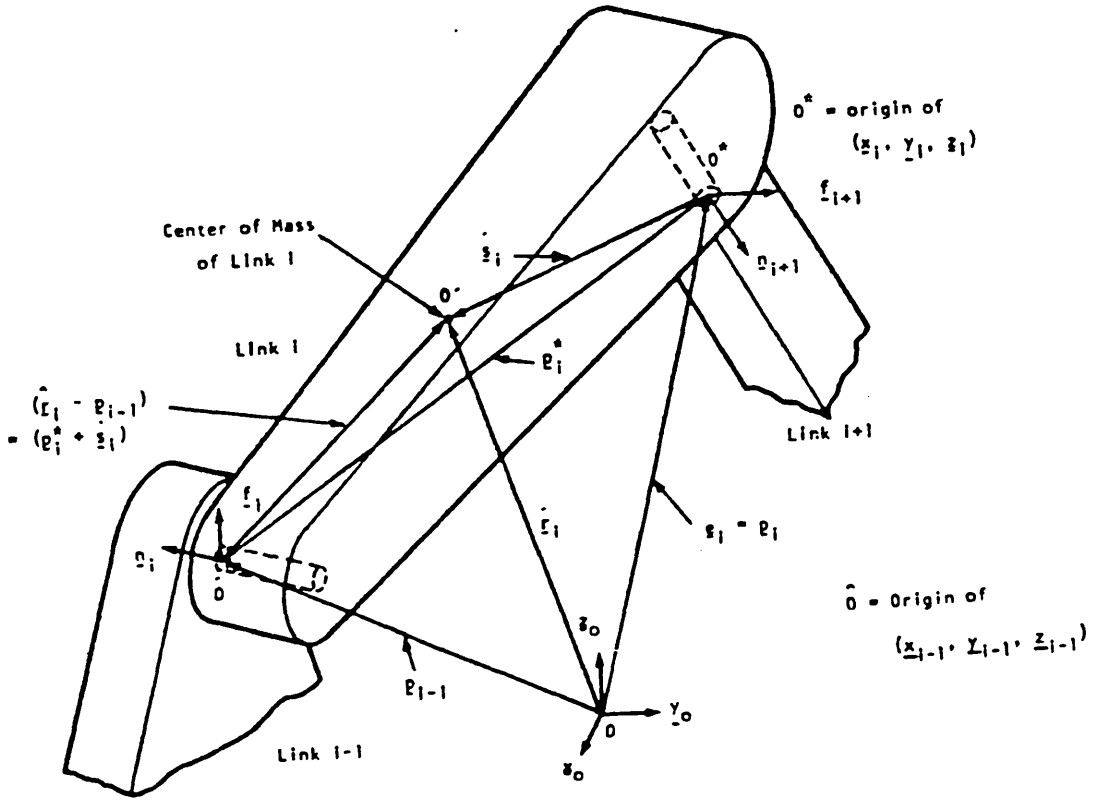


Fig. 3.3 Force and Moment Exerted on and from link i. (extracted from Hollerbach)

ω_i = Angular velocity of link i.

$\dot{\omega}_i$ = Angular acceleration of link i.

f_i = Total external force on link i.

$f_{i-1,\,i}$ = Force exerted on link i by link i-1.

n_i = Total external torque on link i.

$n_{i-1,\,i}$ = Moment exerted on link i by link i-1.

I_i = Inertia tensor of link i about its centre of mass.

T_i = Input torque at joint i.

The backward recursion propagates angular velocities, angular accelerations, linear accelerations, total link forces and total link torques from the base to the end link. For a rotational joint, the recursive equtions are :

$$\underline{\omega}_i = \underline{\omega}_{i-1} + \underline{z}_{i-1}\,\dot{\underline{q}}_i \tag{3.18}$$

$$\dot{\underline{\omega}}_i = \dot{\underline{\omega}}_{i-1} + \underline{z}_{i-1}\,\ddot{\underline{q}}_i + \underline{\omega}_{i-1} \wedge \underline{z}_{i-1}\,\dot{\underline{q}}_i \tag{3.19}$$

$$\ddot{\underline{p}}_i = \dot{\underline{\omega}}_i \wedge \underline{p}_i^{*} + \underline{\omega}_i \wedge (\underline{\omega}_i \wedge \underline{p}_i^{*}) + \ddot{\underline{p}}_{i-1} \tag{3.20}$$

$$\ddot{\underline{r}}_i = \underline{\omega}_i \wedge (\underline{\omega}_i \wedge \underline{r}_i^{*}) + \dot{\underline{\omega}}_i \wedge \underline{r}_i^{*} + \ddot{\underline{p}}_i \tag{3.21}$$

$$\underline{f}_i = m_i \ddot{\underline{r}}_i \tag{3.22}$$

$$\underline{n}_i = \underline{I}_i \dot{\underline{\omega}}_i + \underline{\omega}_i \wedge (\underline{I}_i \cdot \underline{\omega}_i) \tag{3.23}$$

The forward recursion propagates the forces and moments exerted on link i by link i-1 from the manipulator tip to the base.

$$\underline{f}_{i-1,\,i} = \underline{f}_i + \underline{f}_{i,\,i+1} \tag{3.24}$$

$$\underline{n}_{i-1,\,i} = \underline{n}_{i,\,i+1} + \underline{n}_i + (\underline{p}_i^{*} + \underline{r}_i^{*}) \wedge \underline{f}_i + \underline{p}_i^{*} \wedge \underline{f}_{i,\,i+1} \tag{3.25}$$

$$\underline{T}_i = \underline{z}_{i-1} \cdot \underline{n}_{i-1,\,i} \tag{3.26}$$

Applying the above equations to the serial link configuration,

$$\underline{\omega}_1 = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 \end{bmatrix} \tag{3.27}$$

$$\underline{\omega}_2 = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 + \dot{\theta}_2 \end{bmatrix} \tag{3.28}$$

$$\dot{\underline{\omega}}_1 = \begin{bmatrix} 0 \\ 0 \\ \ddot{\theta}_1 \end{bmatrix} \tag{3.29}$$

$$\dot{\underline{\omega}}_2 = \begin{bmatrix} 0 \\ 0 \\ \ddot{\theta}_1 + \ddot{\theta}_2 \end{bmatrix} \tag{3.30}$$

$$(\underline{p}_1^* + \underline{r}_1^*) = \begin{bmatrix} g_1 \cos\theta_1 \\ g_1 \sin\theta_1 \\ 0 \end{bmatrix} \tag{3.31}$$

$$(\underline{p}_2^* + \underline{r}_2^*) = \begin{bmatrix} (g_2 - l_2)\cos(\theta_1+\theta_2) \\ (g_2 - l_2)\sin(\theta_1+\theta_2) \\ 0 \end{bmatrix} \tag{3.32}$$

$$\underline{r}_1^* = \begin{bmatrix} (g_1 - l_1)\cos\theta_1 \\ (g_1 - l_1)\sin\theta_1 \\ 0 \end{bmatrix} \tag{3.33}$$

$$\underline{r}_2^* = \begin{bmatrix} g_2 - l_2 \\ 0 \\ 0 \end{bmatrix} \tag{3.34}$$

$$\underline{n}_1 = \begin{bmatrix} 0 \\ 0 \\ I_1 \ddot{\theta}_1 \end{bmatrix} \tag{3.35}$$

$$\underline{n}_2 = \begin{bmatrix} 0 \\ 0 \\ I_2 (\ddot{\theta}_1 + \ddot{\theta}_2) \end{bmatrix} \tag{3.36}$$

$$f_1 = m_1 \ddot{\underline{r}}_1 \tag{3.37}$$

$$f_1 = m_1 \cdot \begin{bmatrix} -g_1(\ddot{\theta}_1 \sin\theta_1 + \dot{\theta}_1^2 \cos\theta_1) \\ g_1(\ddot{\theta}_1 \cos\theta_1 - \dot{\theta}_1^2 \sin\theta_1) \\ 0 \end{bmatrix} \tag{3.38}$$

$$f_2 = m_2 \ddot{\underline{r}}_2 \tag{3.39}$$

$$f_2 = m_2 \cdot \begin{bmatrix} -l_1 \sin\theta_1 \ddot{\theta}_1 - l_1 \dot{\theta}_1^2 \cos\theta_1 - g_2 \sin(\theta_1+\theta_2)(\ddot{\theta}_1+\ddot{\theta}_2) \\ \quad - g_2 \cos(\theta_1+\theta_2)(\dot{\theta}_1+\dot{\theta}_2)^2 \\ \hline l_1 \cos\theta_1 \ddot{\theta}_1 + l_1 \dot{\theta}_1^2 \sin\theta_1 + g_2 \cos(\theta_1+\theta_2)(\ddot{\theta}_1+\ddot{\theta}_2) \\ \quad - g_2 \sin(\theta_1+\theta_2)(\dot{\theta}_1+\dot{\theta}_2)^2 \\ \hline 0 \end{bmatrix} \tag{3.40}$$

To derive T_2 :

$$f_{1,2} = f_2 \tag{3.41}$$

$$\underline{n}_{1,2} = \underline{n}_2 + (\underline{p}_2^* + \underline{r}_2^*) \wedge f_{1,2} \tag{3.42}$$

$$f_{2,3} = 0 \quad \text{and} \quad \underline{n}_{2,3} = 0$$

$$T_2 = [\underline{n}_{1,2}]$$

$$T_2 = [I_2 + m_2 l_1 g_2 \cos\theta_2 + m_2 g_2^2]\ddot{\theta}_1 + [I_2 + m_2 g_2^2]\ddot{\theta}_2 + m_2 l_1 g_2 \sin\theta_2 \dot{\theta}_1^2 \tag{3.43}$$

To derive T_1 :

$$\underline{n}_{0,1} = \underline{n}_1 + \underline{n}_{1,2} + (\underline{p}_1^* + \underline{r}_1^*) \wedge f_{0,1} - \underline{r}_1^* \wedge f_{1,2} \tag{3.44}$$

$$f_{0,1} = f_1 + f_{1,2} \tag{3.45}$$

$$T_1 = [\underline{n}_{0,1}]$$

$$T_1 = [\, I_1 + I_2 + m_1 g_1^2 + m_2 g_2^2 + m_2 l_1^2 + 2\, m_2 l_1 g_2 \cos\theta_2 \,]\, \ddot{\theta}_1$$
$$+ [\, I_2 + m_2 g_2^2 + m_2 l_1 g_2 \cos\theta_2 \,]\, \ddot{\theta}_2 - m_2 l_1 g_2 \sin\theta_2\, \dot{\theta}_2^2 - [2\, m_2 l_1 g_2 \sin\theta_2]\, \dot{\theta}_1\, \dot{\theta}_2 \qquad (3.46)$$

There are four terms of dynamic torque. The first term represents the inertial torque which is proportional to its own joint acceleration. The second term represents the interactive inertial torque which is proportional to the other joint's acceleration. The third term represents the centripetal toque which is proportional to the square of the other joint 's velocity. Finally, the fo rth term represents the coriolis torque which is proportional to the product of the joint velocities from two different links. The inertial torques arise from the normal action/reaction forces when accelerating each link. The centripetal torques arise when a body is constrained to rotate about a point. The coriolis torques arise from vertical forces deriving from the interaction of two rotating systems. The shoulder torque, equation (3.46), has the elbow and centripetal contributions from the elbow movement. Also, there is a coriolis torque acting at the shoulder, but none acts at the elbow.

3.3 PARALLEL DRIVE MECHANISM

A parallel drive mechanism is shown in Figure (3.4). It consists of a four bar link mechanism. Two motors are located at the base, i.e. joints 1 and 2 , and the distance between them is zero. In other words, the rotors of the two motors are colinear. The two pairs of opposite links are parallel. The weight of one motor is no longer a load on the other motor and the reaction torque of one motor does not act directly upon the other because the motors are both located at the base.

3.3.1 PARALLEL DRIVE DYNAMIC SOLUTIONS

Dynamic equations of the parallelogram arm mechanism can be derived via the Newton-Euler method of formulation. In order to find a solution, twelve simultaneous equations need to be solved. Therefore, it is not very efficient in terms of computational time.

Attempts have been made to search for a much more explicit dynamic equations. Asada, et al., has presented the equation of motion of a direct-drive arm as :

$$T_i = H_{ii}\,\ddot{\theta}_i + \Sigma_{j\neq i} H_{ij}\,\ddot{\theta}_j + \Sigma_j \Sigma_k \left(\frac{d H_{ij}}{d\theta_k} - \frac{d H_{jk}}{2 d\theta_i} \right) \dot{\theta}_j \dot{\theta}_k + T_{gi} \qquad (3.47)$$

where H_{ii} is the i-j element of the arm inertia tensor reflected to the motor,

T_{gi} is the gravity torque,

T_i is the joint torque.

The inertia tensor of the parallel-arm mechanism is then given by:

$$H = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix} \qquad (3.48)$$

where $H_{11} = I_1 + m_1 g_1^2 + I_3 + m_3 g_3^2 + m_4 l_1^2$ (3.49)

$H_{22} = I_4 + m_4 g_4^2 + I_2 + m_2 g_2^2 + m_3 l_2^2$ (3.50)

$H_{12} = H_{21} = (m_3 l_2 g_3 - m_4 l_1 g_4) \cos(\theta_2 - \theta_1)$ (3.51)

All variables are shown in Figure (3.4).

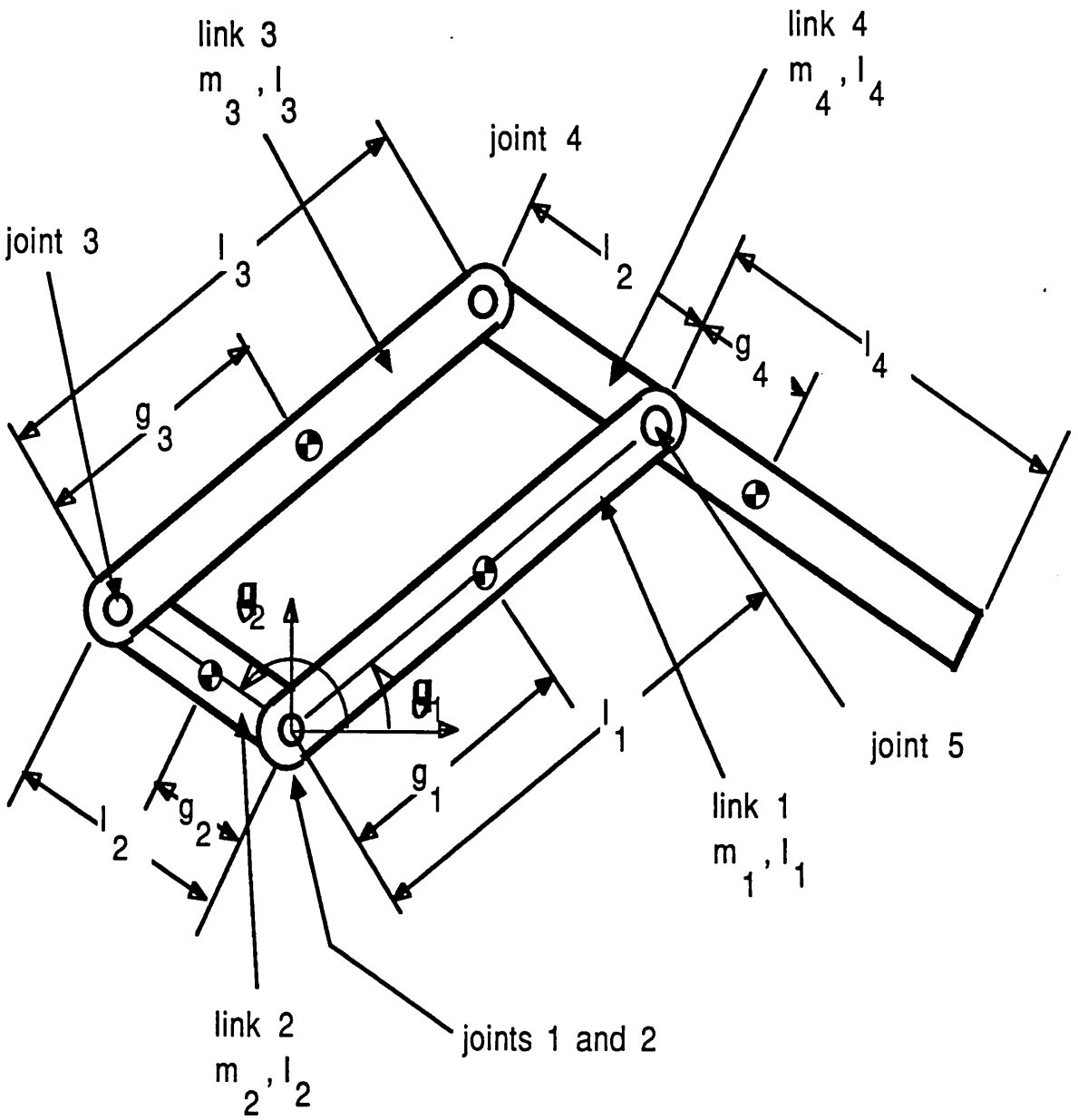


Fig. 3.4 Parallelogram Arm Mechanism

3.3.2 THE RECURSIVE NEWTON-EULER SOLUTIONS (PARALLELOGRAM MECHANISM)

Using the recursive relationship shown in equations (3.18) to (3.26), the derived equations are as follows:

$$\underline{\omega}_1 = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 \end{bmatrix} \quad (3.52) \qquad \underline{\omega}_2 = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_2 \end{bmatrix} \quad (3.53)$$

$$\dot{\underline{\omega}}_1 = \begin{bmatrix} 0 \\ 0 \\ \ddot{\theta}_1 \end{bmatrix} \quad (3.54) \qquad \dot{\underline{\omega}}_2 = \begin{bmatrix} 0 \\ 0 \\ \ddot{\theta}_2 \end{bmatrix} \quad (3.55)$$

$$\underline{\omega}_3 = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 \end{bmatrix} \quad (3.56) \qquad \underline{\omega}_4 = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_2 \end{bmatrix} \quad (3.57)$$

$$\underline{\omega}_4 = \begin{bmatrix} 0 \\ 0 \\ \ddot{\theta}_1 \end{bmatrix} \quad (3.58) \qquad \dot{\underline{\omega}}_4 = \begin{bmatrix} 0 \\ 0 \\ \ddot{\theta}_2 \end{bmatrix} \quad (3.59)$$

$$\underline{f}_4 = \underline{f}_{3,4} + \underline{f}_{1,4} = m_4 \ddot{\underline{r}}_4 \quad (3.60)$$

$$\underline{n}_4 = (\underline{p}_4^* + \underline{r}_4^*) \wedge (\underline{f}_4 - \underline{f}_{1,4}) + (\underline{p}_4^* + \underline{r}_4^*) \wedge \underline{f}_{1,4} = 0 \quad (3.61)$$

$$\underline{f}_3 = \underline{f}_{2,3} - \underline{f}_{3,4} = m_3 \ddot{\underline{r}}_3 \quad (3.62)$$

$$\underline{n}_3 = -(\underline{p}_3^* + \underline{r}_3^*) \wedge \underline{f}_{2,3} + \underline{r}_3^* \wedge \underline{f}_{3,4} \quad (3.63)$$

$$f_2 = f_{0,2} - f_{2,3} = m_2 \ddot{r}_2 \quad (3.64)$$

$$n_{0,2} = n_2 + (p_2^* + r_2^*) \wedge f_{0,2} - r_2^* \wedge f_{2,3} \quad (3.65)$$

$$f_1 = f_{0,1} - f_{1,4} = m_1 \ddot{r}_1 \quad (3.66)$$

$$n_{0,1} = n_1 + (p_1^* + r_1^*) \wedge f_{0,1} - r_1^* \wedge f_{1,4} \quad (3.67)$$

$$T_1 = [n_{0,1}] \quad (3.68)$$

$$T_2 = [n_{0,2}] \quad (3.69)$$

After considerable simplification, equations (3.60) t o (3.67) can be expressed as :

$$X_{3,4} + X_{1,4} = m_4 a_{i4} \quad (3.70)$$

$$Y_{3,4} + Y_{1,4} = m_4 a_{j4} \quad (3.71)$$

$$b_1 X_{1,4} - b_2 Y_{1,4} = b_3 + b_4 a_{j4} + b_5 a_{i4} \quad (3.72)$$

$$X_{2,3} - X_{3,4} = m_3 a_{i3} \quad (3.73)$$

$$Y_{2,3} - Y_{3,4} = m_3 a_{j3} \quad (3.74)$$

$$b_6 X_{3,4} + b_7 Y_{3,4} = b_8 \quad (3.75)$$

$$X_{0,2} - X_{2,3} = m_2 a_{i2} \quad (3.76)$$

$$Y_{0,2} - Y_{2,3} = m_2 a_{j2} \quad (3.77)$$

$$T_2 + l_2 \cos\alpha \, Y_{2,3} - l_2 \sin\alpha \, X_{2,3} = b_9 \quad (3.78)$$

$$X_{0,1} - X_{1,4} = m_1 a_{i1} \quad (3.79)$$

$$Y_{0,1} - Y_{1,4} = m_1 a_{j1} \quad (3.80)$$

$$T_1 - l_1 \cos\theta_1 \, Y_{1,4} + l \sin\theta_1 \, X_{1,4} = b_{10} \quad (3.81)$$

where $X_{i,j}$, $Y_{i,j}$ = Reactions between link i and link j, as shown in Figure (3.5).

T_1, T_2 = External joint torques at joint 1 and 2,

a_{in} , a_{jn} = Inertial acceleration of link n in cartesian coordinates i, j, k.

$\alpha = \theta_2 - \pi$

$$b_1 = l_2 \sin\alpha$$

$$b_2 = l_2 \cos\alpha$$

$$b_3 = - I_4 \ddot{\theta}_4$$

$$b_4 = - m_4 (l_2 \cos\alpha - (l_4 - g_4) \cos\alpha)$$

$$b_5 = m_4 (l_2 \sin\alpha - (l_4 - g_4) \cos\alpha)$$

$$b_6 = g_3 \sin\theta_1 + (l_3 - g_3) \sin\theta_1$$

$$b_7 = -g_3 \cos\theta_1 - (l_3 - g_3) \cos\theta_1$$

$$b_8 = I_3 \ddot{\theta}_3 + m_3 g_3 \cos\theta_1 a_{j3} - m_3 g_3 \sin\theta_1 a_{i3}$$

$$b_9 = I_2 \ddot{\theta}_2 - m_2 g_2 \cos\alpha a_{i2} + m_2 g_2 \sin\alpha a_{i2}$$

$$b_{10} = I_1 \ddot{\theta}_1 + m_1 g_1 \cos\theta_1 a_{j1} - m_1 g_1 \sin\theta_1 a_{i1}$$

$$a_{i1} = -g_1 \ddot{\theta}_1 \sin\theta_1 - g_1 \dot{\theta}_1^2 \cos\theta_1$$

$$a_{j1} = -g_1 \ddot{\theta}_1 \cos\theta_1 - g_1 \dot{\theta}_1^2 \sin\theta_1$$

$$a_{i2} = g_2 \ddot{\theta}_2 \sin\alpha + g_2 \dot{\theta}_2^2 \cos\alpha$$

$$a_{j2} = -g_2 \ddot{\theta}_2 \cos\alpha + g_2 \dot{\theta}_2^2 \sin\alpha$$

$$a_{i3} = l_2 \ddot{\theta}_2 \sin\alpha + l_2 \dot{\theta}_2^2 \cos\alpha - g_3 \ddot{\theta}_3 \sin\theta_1 - g_3 \dot{\theta}_3^2 \sin\theta_1$$

$$a_{j3} = -l_2 \ddot{\theta}_2 \cos\alpha + l_2 \dot{\theta}_2^2 \sin\alpha + g_3 \ddot{\theta}_3 \sin\theta_1 - g_3 \dot{\theta}_3^2 \sin\theta_1$$

$$a_{i4} = -l_1 \ddot{\theta}_1 \sin\theta_1 - l_1 \dot{\theta}_1^2 \cos\theta_1 - g_4 \ddot{\theta}_2 \sin\alpha - g_4 \dot{\theta}_2^2 \cos\alpha$$

$$a_{j4} = l_1 \ddot{\theta}_1 \cos\theta_1 - l_1 \dot{\theta}_1^2 \sin\theta_1 + g_4 \ddot{\theta}_2 \cos\alpha - g_4 \dot{\theta}_2^2 \sin\alpha$$

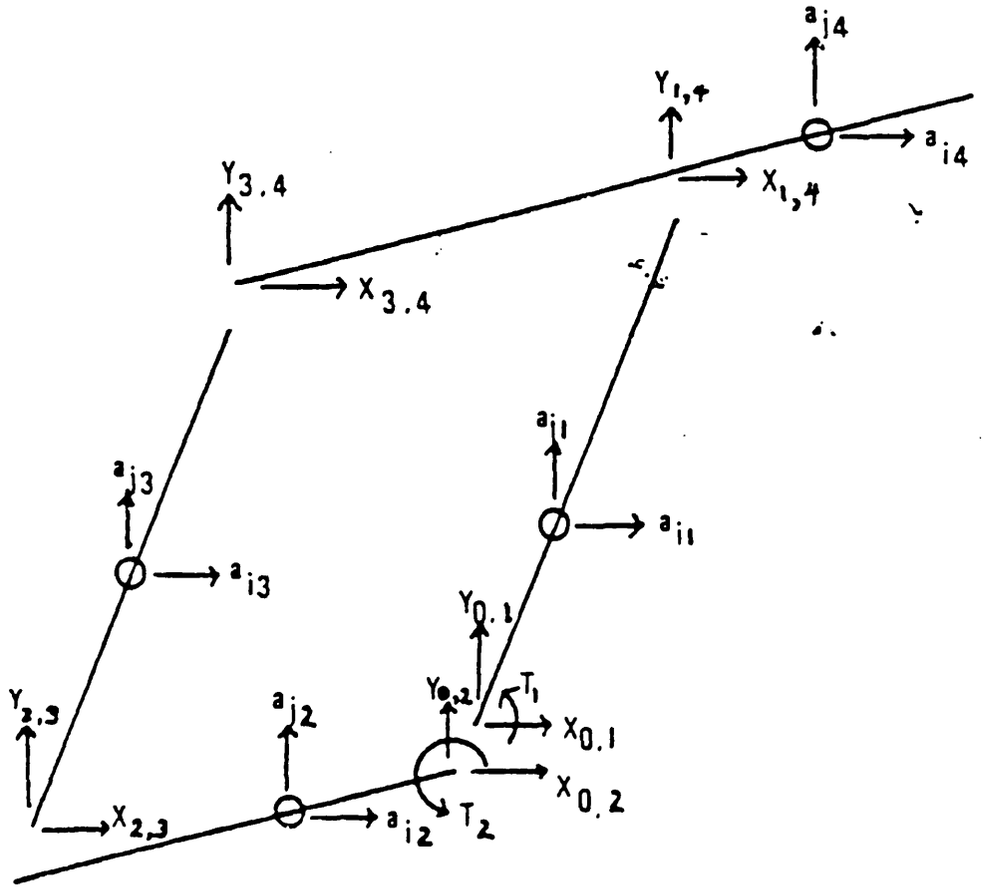


Fig. 3.5 Variables Used in Recurive Newton-Euler Equations

Solutions of reactions between links and the joint torques can be found by simultaneously solving equations (3.70) to (3.81). X, Y components of the action/reaction forces are then resolved for each link, giving vital information about the behaviour of forces at various joints. External joint torques, T_1 and T_2 , are supplied by the two motors situated at the base.

3.3.3 DYNAMICS OF THE PARALLEL DRIVE MECHANISM (LAGRANGE APPROACH)

By applying the Lagrange equations, the full dynamics equations of the parallelogram drive mechanism can be obtained as follows :

$$\begin{bmatrix} T_1 \\ T_2 \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} 0 & -X \\ X & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1^{\,2} \\ \dot{\theta}_2^{\,2} \end{bmatrix} \qquad (3.82)$$

where $H_{11} = I_1 + m_1 g_1^2 + I_3 + m_3 g_3^2 + m_4 l_1^2$ (3.83)

$H_{22} = I_4 + m_4 g_4^2 + I_2 + m_2 g_2^2 + m_3 l_2^2$ (3.84)

$H_{12} = H_{21} = (m_3 l_2 g_3 - m_4 l_1 g_4) \cos(\theta_2 - \theta_1)$ (3.85)

$X = (m_3 l_2 g_3 - m_4 l_1 g_4) \sin(\theta_2 - \theta_1)$ (3.86)

All variables are shown in Figure (3.4).

3.4 INVARIANT AND DECOUPLED ARM DYNAMICS

A general robot arm is a non-linear, highly coupled multivariable system which poses severe practical problems in terms of control (Waters, 1979). There are many theoretical methods for finding a solution that produce a range of acceptable results in simulation studies as well as in on-line control. Such methods include resolved motion rate control (Whitney, 1969), resolved acceleration control (Luh, Walker and Paul, 1980) and adaptive control (Koivo and Guo, 1981). Many of the recent control schemes are summarised in Luh (1983). In terms of implementation, the more sophisticated controllers require greater computational power which has its own disadvantages in performance and/or cost. Hence it is desirable to design as simple a controller as possible.

One area of control requiring more complex control system design is that of dealing with non-linearities. It is much easier to produce a controller with a linear system. In a robot arm, the non-linearities are manifested in the dependence of the arm kinematics for determining the dynamic equations. As a consequence, the design of the robot arms which possess no

coupling and which have constant inertias has been studied (Youcef-Toumi and Asada, 1985). A consideration of the equations of motion for the arm being studied, equations (3.82) to (3.86) show that only the interactive inertia torque, and centrifugal effect torque actually depend on the arm configuration. In other words, the direct inertia torque or the elements on the main diagonal in the inertia tensor are invariant and do not depend on the arms' position.

In a multivariable system, one of the aims is to always try and minimise the coupling or interaction. Robot arms are good examples of multivariable systems because in general, the torque required by a particular actuator is dependent on how the other links in the arm are moving. Hence the system is highly coupled. However the gearing in conventional robots tends to reduce this effect so that it is possible to treat each axis seperately with the coupling effects being considered as a disturbance, the independent joint control method. In a direct-drive arm the absence of any gearing may mean that these interaction torques will cause poor controller performance. Hence more complex controller design is required to compensate for the additional problems introduced by direct-drive.

If possible it would be desirable to find some other means of reducing or eliminating the effects of coupling. Again considering equations (3.82) to (3.86), it should be noted that the coupling term and the non-linear centrifugal term both contain the factor, $(m_3 l_2 g_3 - m_4 l_1 g_4)$ in each equation. If in the design stage, the mass and dimension properties are chosen so as to make this term zero then this will eliminate the problem. This corresponds to making the off-diagonal terms in the inertia tensor identically equal to zero. Asada and Youcef-Toumi (1984) proposed this as the best method to adopt in overcoming this drawback of direct-drive. The reference gives the condition for no interaction for all arm configurations as:

$$\frac{m_3 g_3}{m_4 g_4} = \frac{l_1}{l_2} \tag{3.87}$$

If this condition is satisfied then the dynamic equations for the two base motors reduce from equation (3.82) to:

$$T_1 = (I_1 + m_1 g_1^2 + I_3 + m_3 g_3^2 + m_4 l_1^2) \ddot{\theta}_1 \quad (3.88)$$

$$T_2 = (I_4 + m_4 g_4^2 + I_2 + m_2 g_2^2 + m_3 l_2^2) \ddot{\theta}_2 \quad (3.89)$$

The equations above show that the torque required at one of the motors depends only on the required angular acceleration. The load to the motor is constant and does not depend upon either the arm position or the motion of the other axis. Hence the arm inertia is said to be invariant and decoupled. The conclusion of this section is that it is possible to construct a direct-drive arm in a configuration shown in Figure (3.4), such that the two axes can be considered as two seperate sub-systems.

4 SYSTEM IMPLEMENTATION

4.1 INTRODUCTION

A direct-drive robot arm, the Imperial College Experimental Robot Arm, was constructed following the principles described in chapter 3. The objective of this chapter is to give an insight into the details of the computer numerical control of the arm's hardware and the associated control software. In addition, the instruction language for the operation of the arm will also be explained. This is a Basic-like language called the Pick and Place Interpreter Language (PPIL in short).

4.2 HARDWARE DETAILS

This section gives a detailed description of the hardware used to construct the IC Experimental Robot Arm. The arm can be thought of as consisting of several parts and these are each considered seperately. Firstly there is the arm itself, the design of which is based upon the specifications required to achieve the desirable properties described in chapter 3. Secondly the actuators are described, that is the motors incorporating tachometers and the amplifiers. Finally the microprocessor system used to control the arm is described together with the necessary additional microprocessor hardware.

A photograph of the IC Experimental Robot Arm is given in Figure (4.1) which illustrates the arm itself, the actuators and the microprocessor controller. The arm and the attached pneumatic actuator which moves the pneumatic gripper vertically up and down is shown in detail in the photograph in Figure (4.2).

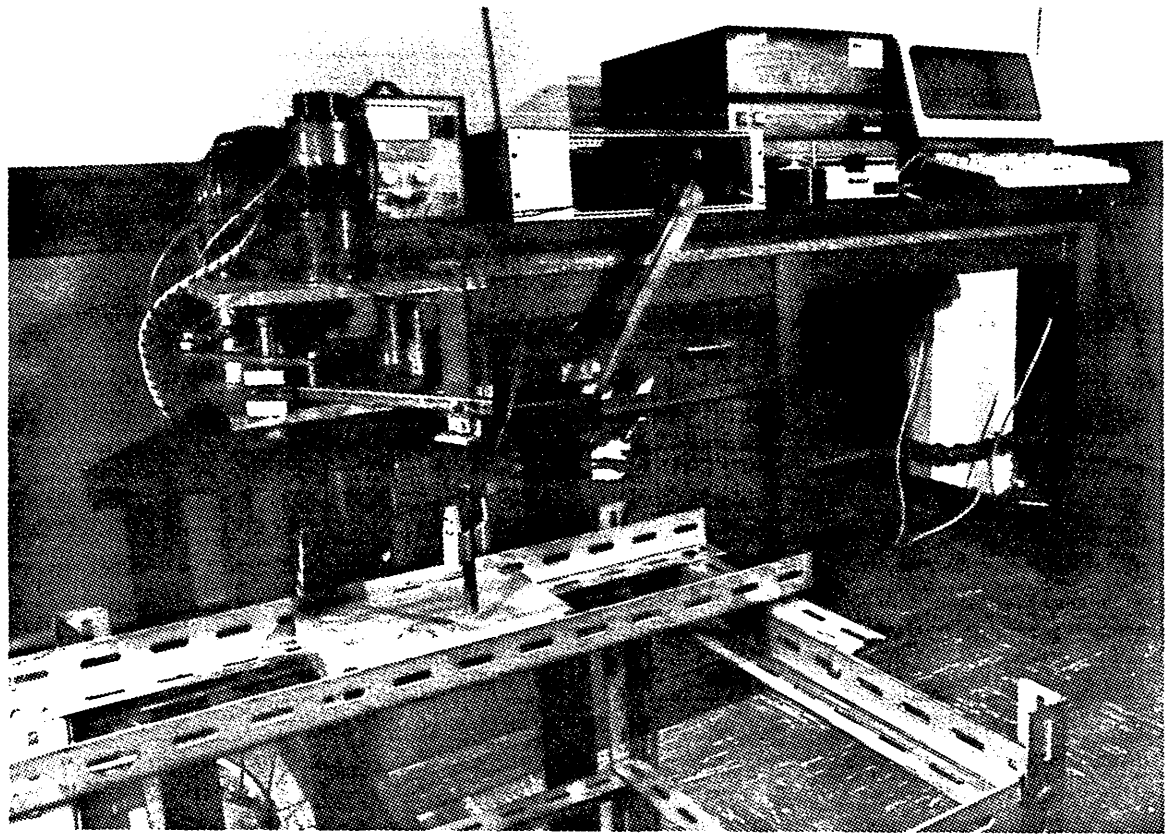


Fig. 4.1 Photograph of the IC Experimental Arm, Microproceesor Controller, Host Computer and the Terminal

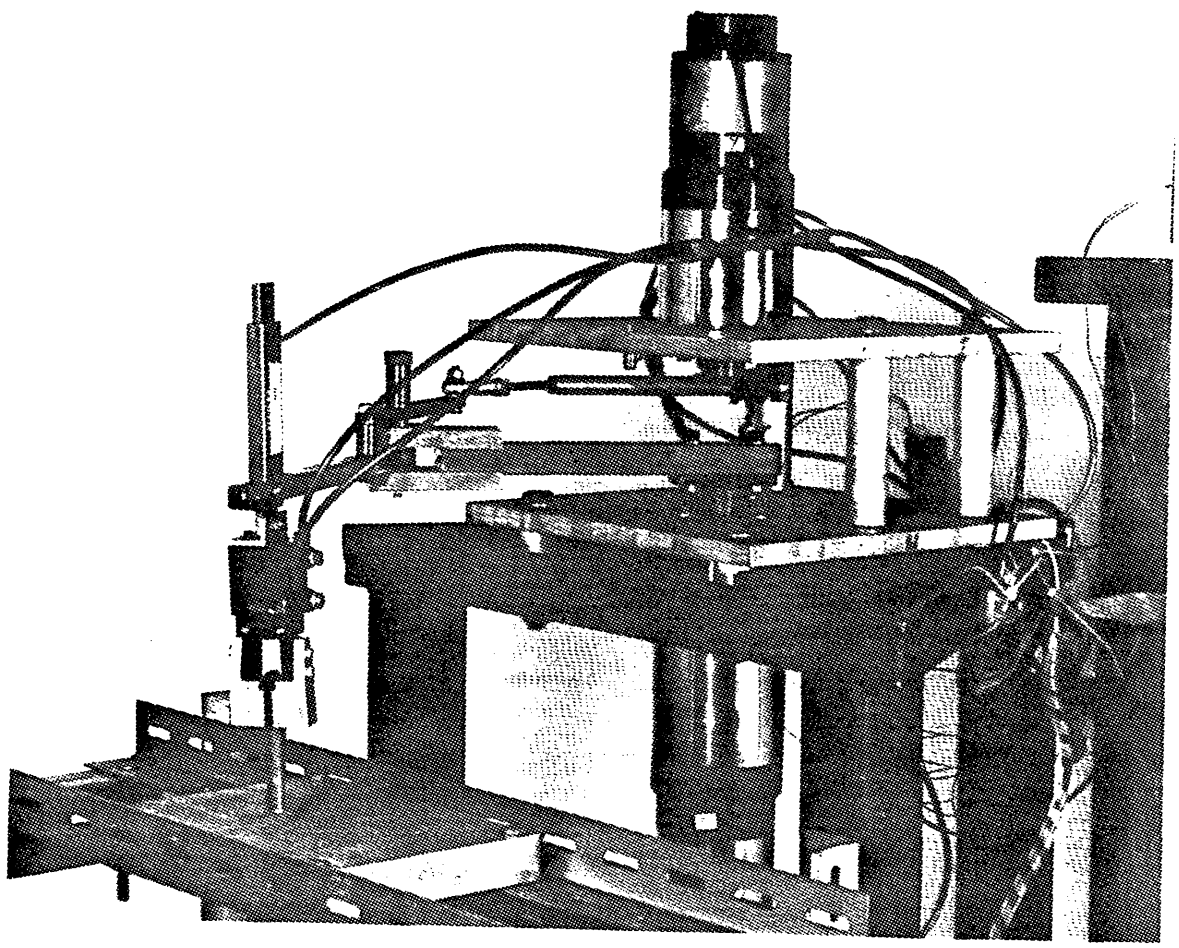


Fig. 4.2 Photograph of the IC Experimental Arm in Detail

4.2.1 DIRECT DRIVE ARM

The IC Experimental Arm is constructed with a four bar parallel drive mechanism as shown in Figure (4.3). As the motors are mounted at the base, the arm links can be of a light construction as they do not carry a large load. The basic configuration of the arm is illustrated which shows the critical dimensions and the values of these plus the mass values are given in Table (4.4). These are particularly important in evaluating the dynamic equations and achieving decoupling by satisfying the condition in equation (3.87).

Referring to Figure (4.3), link 1 can rotate from -27 ° to +187 ° and link 2 from -165 ° to +165 ° . This gives a nominal horizontal work envelope as shown in Figure (4.5) although there are obstructions i.e. the pillars supporting the top motor base plate, which produce cuts in the work envelope. One such, the largest is shown in the diagram.

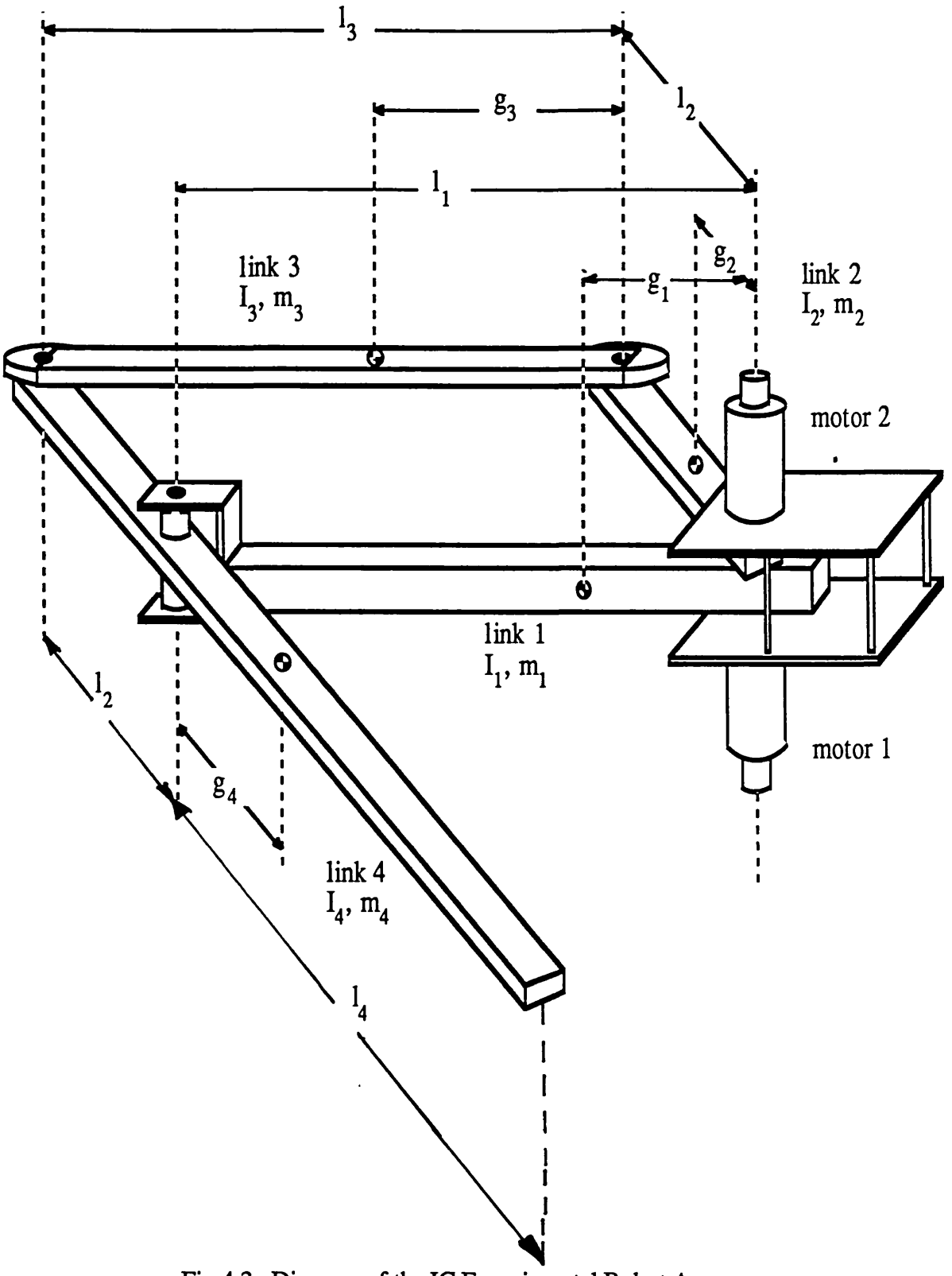


Fig.4.3 Diagram of the IC Experimental Robot Arm

Link Number, i	Length, l_i (m)	Mass, m_i (kg)	Distance, g_i (m)	Inertia, I_i ($kg.m^2$)
1	0.2	0.6	0.108	5.165×10^{-3}
2	0.1	0.22	0.029	3.851×10^{-4}
3	0.2	0.11	0.10	6.494×10^{-4}
4	0.2	0.28	0.028	1.112×10^{-3}

Table 4.4 Dimensions and Masses of the IC Experimental Robot Arm Links

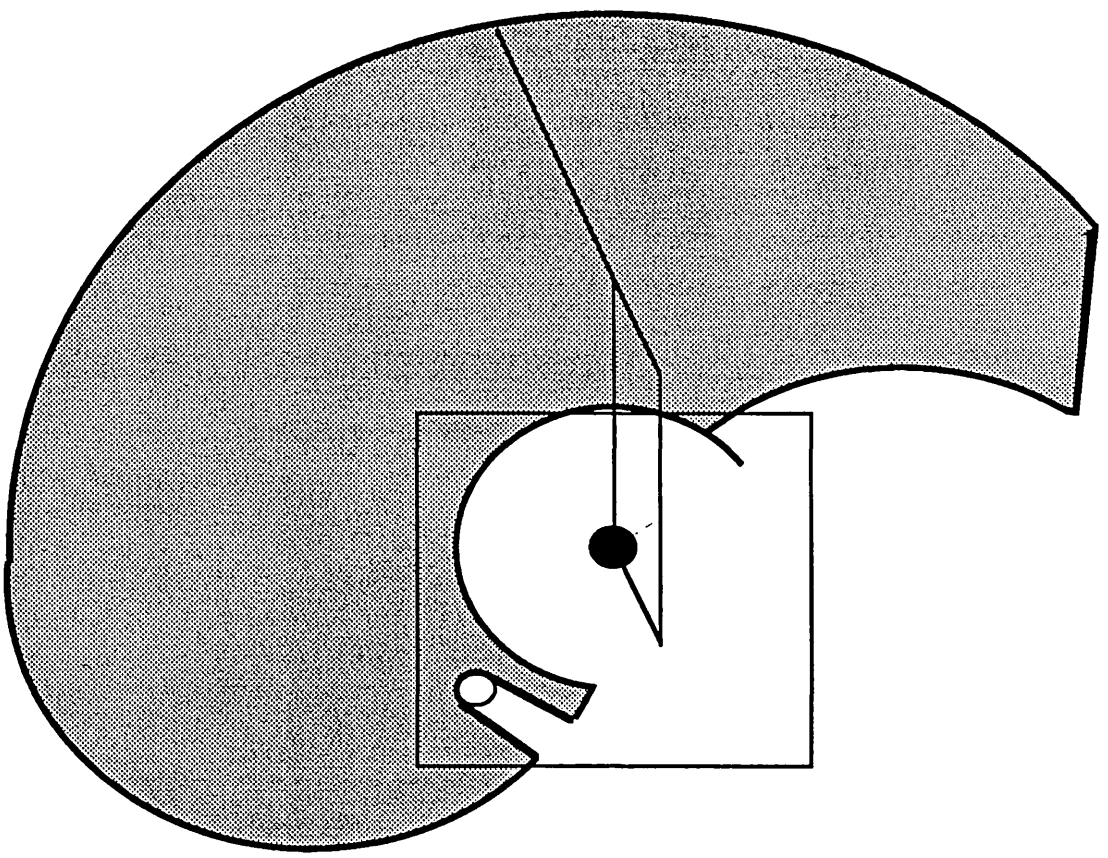


Fig. 4.5 Work Space of the IC Experimental Arm

An important part of the design are the bearings at the link joints. Apart from any friction at the rotor bearings in the motors which cannot be avoided when designing the arm, the only friction arises at these joints. One of the principles of direct-drive is the elimination of frictional torques and so the joint bearings must be of sufficient quality not to introduce a significant amount of friction. The IC Experimental Arm is only intended to demonstrate that the principle of a parallel drive, direct-drive mechanism is feasible and to investigate solutions to the position tracking and force control problem. Hence relatively simple and inexpensive bearings are used in the construction.

The final consideration in the construction of the arm is achieving the required mass distribution to obtain decoupling of the axes. The necessary condition is given in eq. (3.87) and using the appropriate figures from Table (4.4), it is found that :

$$\text{L.H.S.} = \frac{m_4}{m_3} \cdot \frac{g_4}{g_3} = \frac{0.28}{0.11} \cdot \frac{0.028}{0.1} = 0.713 \tag{4.1}$$

$$\text{R.H.S.} = \frac{l_2}{l_1} = \frac{0.1}{0.2} = 0.5 \tag{4.2}$$

Hence analytically it is found that decoupling is nearly achieved as the above equations show that the condition in eq. (3.87) is approximately met. In order to check whether decoupling is actually achieved it is now possible to calculate the full dynamic equations for each motor given in equations (3.82) to (3.86). If this is done then it is found that the coupling terms are two orders of magnitude smaller than the direct arm inertia term and so can be considered insignificant. In addition, experimentation on the arm where one axis is moved whilst the other is not servoed results in almost perfect decoupling given the resolution of the position measuring devices used. So it can be assumed that for all practical purposes there are no interaction torques between the two axes.

At this point it must be noted that the arm described so far does not include any end effector and hence cannot perform any meaningful operations. The purpose in constructing the arm is primarily to consider the position tracking control problem which only involves a pick and place type motion in the horizontal plane. In addition, the arm is also used for research into the use of force control and compliance in the assembly operation itself. As will be explained in chapter 6 a direct-drive arm is particularly well suited to force control applications as the force exerted can be measured by measuring the motor currents so obviating the use of complex and expensive force measuring devices.

Due to this further investigation, a simple end effector consisting of pneumatic drives to provide a vertical motion and gripper open/close control can be attached to the arm. Clearly this additional mass disturbs the required mass distribution so that the axes are no longer decoupled. This problem is considered later when the various control schemes are implemented but for now it can be noted that in the development of a commercial arm along these principles, the mass of the end effector can be taken into account and so decoupling can be achieved.

The question of disturbing the mass distribution also arises when considering actual assembly operations. The picking of a component for insertion will again result in an additional mass which will upset the decoupling. As the component is not a fixture to the arm, it cannot be taken into consideration as with the end effector. The problem can be solved by providing some form of movable counterbalance to negate the effect of the additional mass of the component. However this requires relatively complex mechanisms so introducing more problems. The IC Experimental Robot Arm is designed for light assembly work and consequently it is envisaged that the components will have a negligible mass compared to the arm itself. So in some applications, the mass of the component is something which must be considered and there are methods for overcoming the resultant loss of decoupling at the design stage. These are not deemed necessary here as the arm is proposed

to be used for light assembly work only where the mass of the component is negligible. In any case the aim of this study is primarily the position and force control problem and it is not productive to spend too much time considering ancillary problems.

4.2.2 SYSTEM'S ELECTRONICS HARDWARE

Figure (4.6) shows the layout of the electronics of the microprocessor system. The unit is made up of the following cards:

1- Power supply card

2- Microprocessor card

3- TIO1 card

4- MEM1 memory card

5- DAC/timer card

6- Encoder interface card

A brief description of each card is given in this section.

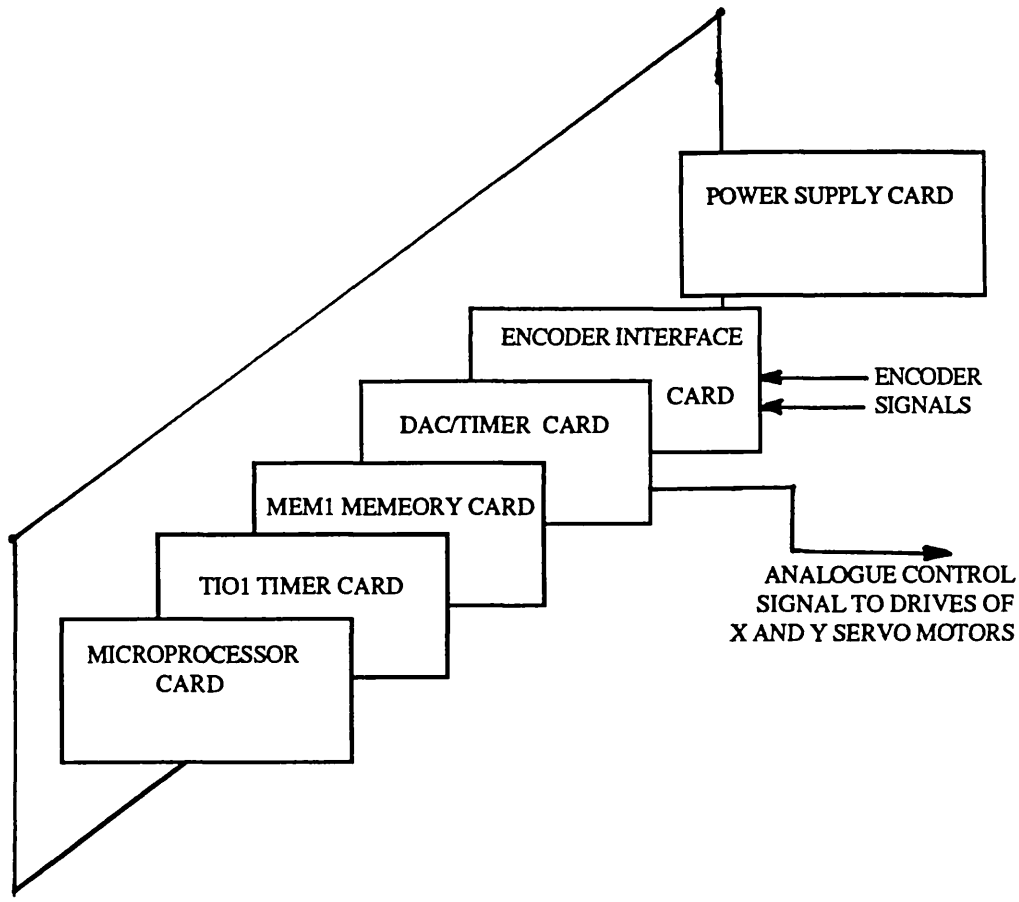


Fig. 4.6 Layout of the Computer Hardware

POWER SUPPLY CARD

A 30 watt switched-mode power supply designed for use in Eurocard systems is used. The card can deliver 3A current at 5V level and 1.2A current at 1 2V level. The present system's power requirement is well within the supply limits.

MICROPROCESSOR CARD

This card is a Syntel MP09 card which has the standard G64 bus on its backplane interface.It contains :

Motorola MC68B09 processor chip

Four sockets for up to 32 Kbytes of EPROM/RAM chips.

A programmable RS232 interface chip (MC68B50 ACIA chip)

A programmable timer/counter chip (MC68B40)

Both the position control prgram and the user defined program (instructions) are stored in this card. The control program and the user program are both stored in RAM.

DIGITAL I/O CARD

This is a Syntel TIO1 card which has a backplane interface to the standard G64 bus. The card contains 2 digital input/output chips which are fully programmable (MC68B20 PIAs). This card has also been used for interface to pneumatic on/off valves for the vertical pneumatic actuator and the attached pneumatic gripper.

DUAL DAC / TIMER CARD

This is a wirewrap card on the G64 bus which was developed at Imperial College. It contains :

- Two 12-bit digital to analogue convertors (DAC)
- Two programmable timer/counter chips (MC68B40)

This card is used to count position feedback pulses from the optical incremental encoders, and to provide precise timing for the motion generation. The two DACs provide the ana logue interface for the servo motor drive units.

ENCODER DECODER CARD

The purpose of this card is to obtain directional information and position information by decoding the encoder signals A and B. Figure (4.7) shows how this is done.

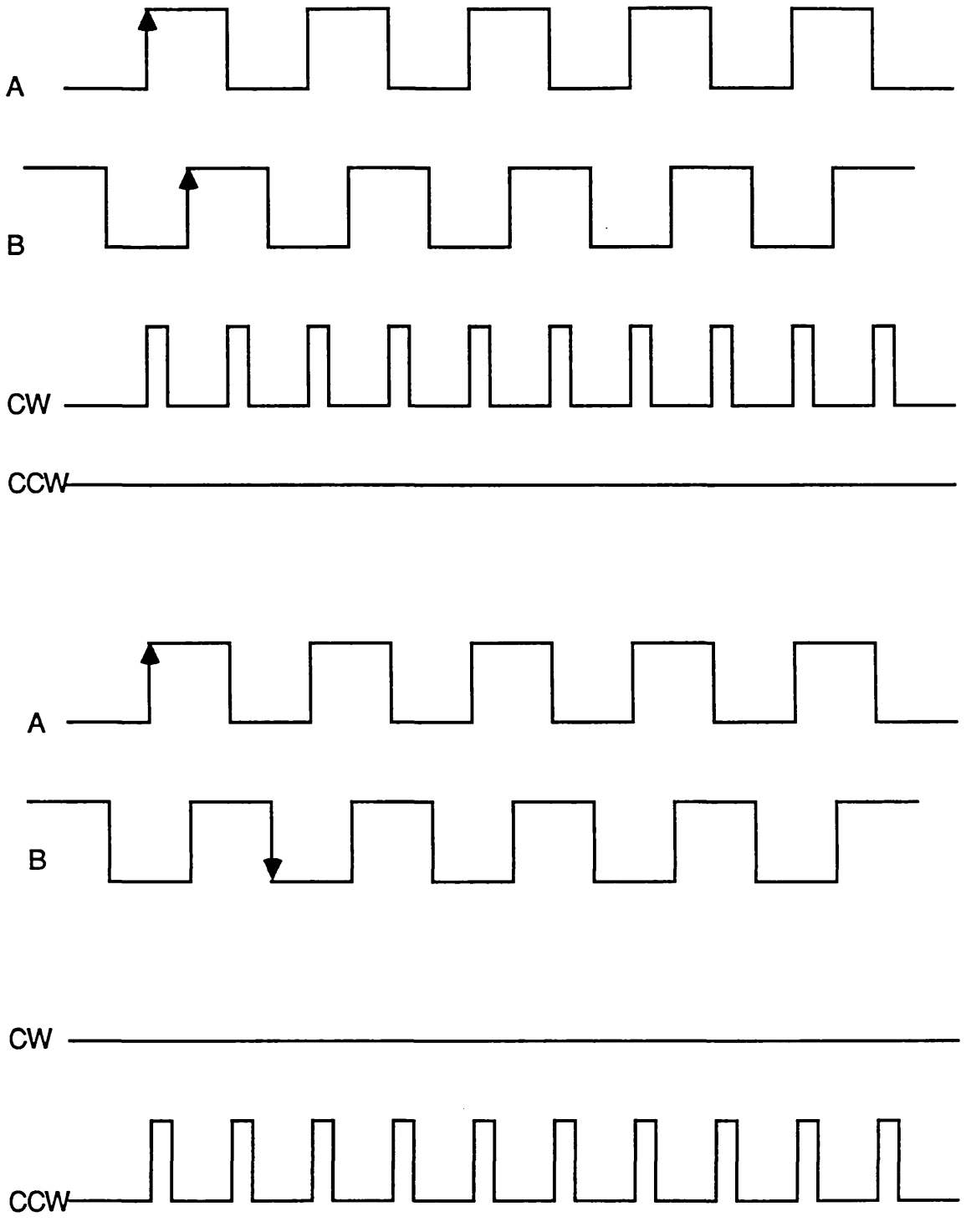


Fig. 4.7 Principles of Decoding the Encoder Pulses

As can be seen from the diagram above, the circuit effectively multiplies the encoder resolution by a factor of four. Hence, the 500 lines encoders can produce 2000 bits per revolution.

4.3 SYSTEM SOFTWARE

The system's executive software has been developed using the MC6809 assembly language. The choice of this low level language was deemed necessary due to the real time control needs of the servo actuators. The binary version of the program is stored on an 8 Kbytes RAM which is placed on the MP09 processor board explained earlier. The software contains a background program which executes the user interface via a Basic - like language called PPIL. The executive program also contains a timer driven interrupt routine for the purpose of motion generation and servo loop closures. The memory map of the MP09 processor board and the associated memory card is as follows:

$F7FE	Start of ACIA addresses
$F428	Start of PIA addresses
$F410	Start of timer on DAC board
$F400	Start of DAC addresses
$A000	Start of table storing SIN values
$9000	Start of table storing ATAN values
$8000	Start of table storing ACOS values
$7000	Start of the maths package
$4200	Start of force control program area (written in PL/9)
$41FF	End of force control variable storage area
$4000	Start of force control variable storage area

$2200	Start of PPIL program area (written using 6809 assembly language)
$1FFF	Start of stack
$1570	Bottom of program stack
$156F	End of RAM for user program
$1520	Start of RAM for user program
$151F	End of scratch RAM
$0100	Start of scratch RAM
$00FF	
$0000	G-BUG

Table 4.8 Memory Map of the Processor Card

4.3.1 SYNTAX OF PPIL

The general syntax of a line in PPIL is as follows :

XXXX (command type) (expression or argument), , CR

where XXXX represents a decimal four digit line number in the range 1-9999.

The (command type) may be one of the commands defined below. Only the first three characters of a command are required for recognition. Expressions are one argument or two arguments linked with an arithmetic operator. The valid arithmetic operators are :

+	addition	16 bit +16 bit ,	results 16 bits
-	subtraction	16 bit - 16 bit ,	results 16 bits
/	division	16 bit /16 bit ,	results 16 bits
*	multiplication	8 bit * 16 bit ,	results 16 bits

Arguments are decimal numbers in the range -32768 to 32767 or a symbol. Symbols are either single alphabetic character or an alphabetic character follwed by a single digit number. Hence the following are legal expressions :

0

1234

-12345

A+B

A1-B1

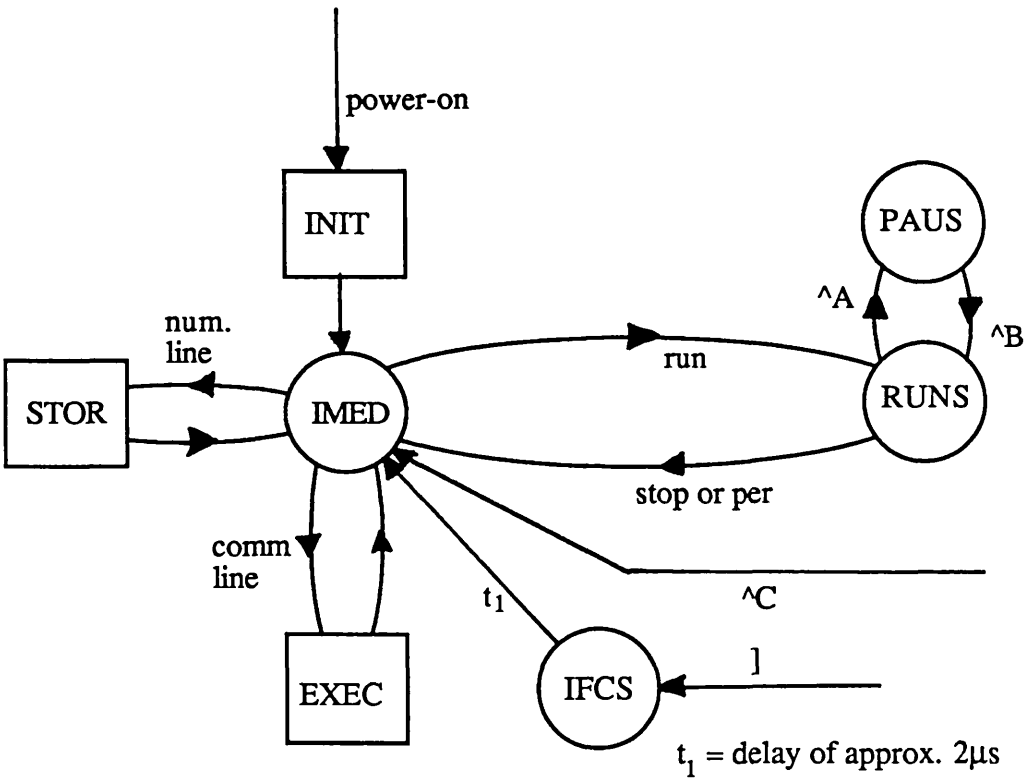
A*B

C/B1

PPIL has two modes of operations, RUN and IMMEDIATE modes. Refer to Figure (4.9) for the state diagram. Entry to the IMMEDIATE mode is signalled by the sequence:

READY CR

#



Mnenomic Tables

States

IMED	- Immediate
RUNS	- RUn state
PAUS	-Pause state
IFCs	-Motion stop state

Processes

INIT	-initialisation
STOR	-store line in memory
EXEC	- execute single line

Messages

run	-run command executed
pon	-power on
stop	-stop command executed
per	-programming error encountered
^C	-control-C character received
]	-right square bracket character received
^A	-control-A character received
^B	-control-B character received
num. line	- a numbered line
comm line	- a command line

Fig.4.9 State Diagram of PPIL

In the IMMEDIATE modes lines may be entered by entering numbers via the RS232 terminal interface. All the characters must be numeric or upper case alphabetic in ASCII code. Lines must always be terminated by a carriage return (CR). Lines are stored in RAM area sequentially regardless of the order in which they were entered. Lines may be edited by re-entering the desired line number followed by a new command. Lines can be deleted by entering the line number followed immediately by a CR. The completion of the store / edit procedure is signalled bythe sequence :

CR

#

Lines not preceded by a line number will be interpreted as commands and executed immediately after the CR. This feature applies to all commands including the MOVE command. The completion of a command in immediate mode is signalled by the sequence :

CR

READY CR

#

RUN mode is entered executing the RUN command in the IMMEDIATE mode. In RUN mode lines in the user program are executed in line number sequence (except where branching is implied), until the STOP command is encountered, an error occurs, a CONTROL C or a] (EMERGENCY STOP) character is received via the RS232 link. Program control then returns to the IMMEDIATE mode. The backspace character and the CONTROL X functions are included.

4.3.2 PROGRAM REPERTOIRE

Operation repertoire :

There are seven commands of this class :

MOVE

format : XXXX MOVE X Y T

description : Motion is generated and controleed with X and Y strokes in time T. X and Y are in encoder pulses, T is in milliseconds.

X, Y and T may be any previously defined expression or argument.

DWELL

format : XXXX DWL T

description : Motion delay of time T in milliseconds is generated.

T may be any previously defined expression or argument.

RAMP

format : XXXX RAM XINC YINC

description : Constant speed motion is generated using XINC and YINC as incremental length units per sampling period.

The variables may be any previously defined expression or argument. The motion is terminated by a "]" character via the RS232 interface.

4.3.3 PROGRAM CONTROL REPERTOIRE

These commands control the order of execution of the part program. They are similar in

action to commands found under the same names in other Basic interpreters.

RUN

format : RUN

description : This command transfers program control from IMMEDIATE mode to RUN mode in order to execute the user program in ascending line number sequence (except where branching is implied) until a STOP command is encountered, a CONTROL C or a] is received from the RS232 user interface.

STOP

format : XXXX STOP

description : Transfers program control to the IMMEDIATE mode. It also causes the text string "STOP AT LINE YYYY" to be output on the system console. YYYY shows the line number in which the STOP command was encountered.

GOTO

format : XXXX GOTO (expression)

description : Transfers program control to the line number defined in the expression. It may be used in the IMMEDIATE mode to start a program without clearing all the program variables.

GOSUB

format : XXXX GOSUB (expression)

description : Program control is transfered to the line number in the expression. The line number of the next line is stored in an eight deep push down stack for use when RETURNing from the sybroutine. Hence eight levels of subroutine nesting are permitted.

RETURN

format : XXXX RETURN

description : A line number is obtained from the push down stack and program control is transferred to that line number.

IF

format : XXXX IF (argument) test symbol (expression) THEN (line number)

description : Performs a comparison of argument and expression. The program control is transferredto the line number defined upon an affirmative result. Upon a non-affirmative result program control is transferred to the next sequential line number. The valid test symbols are :

= tests for arithmetic equality

> tests for argument greater than expression

< tests for argument less than the expression

LIST

format : LIST

description : It causes the listing of the user program on the system's RS232 console.

NEW

format : NEW

description : It erases all the contents of the user program and all of the arguments.

DATA HANDLING REPORTOIRE

There is one command which allows the creation and storage of arguments :

LET

format : XXXX LET (argument) = (expression)

description :

It evaluates the expression and transfers it into the argument store. Arguments may be re-defined, and are only deleted from the store by a NEW or a RUN command.

INPUT/OUTPUT REPORTOIRE

These commands allow program / user interaction :

INPUT

format : XXXX INPUT (argument), , ,

description : A ? is output to the console, expression entered by the user are then evaluated and stored in the arguments respectively.

PRINT

format : XXXX PRINT (expression), , , ,

description : The expression or an ASCII string between the " " symbols will be printed to the console.

4.3.4 SERVO CONTROL REPERTOIRE

These commands are used for setting up the servo gains. Note that by setting the gains to

zero the corresponding control actions are deactivated.

KPS

format : XXXX KPS (expression) (expression)

description : The error proportional gains for the X and the Y servos are set.

KVF

format : XXXX KVF (expression) (expression)

description : Two velocity feedforward gains (one for each axis) are set.

KAF

format : XXXX KAF (expression) (expression)

description : Two acceleration feedforward gains (one for each axis) are set.

KVB

format : XXXX KVB (expression) (expression)

description : Two velocity feedback gains (ne for each axis) are set.

Note that the expressions for the various gains are converted to two byte binary numbers. Each number is treated with its most significant byte as a binarey integer and its least significant byte as a binary fraction for the purpose of servo loop closure algorithms.

XDIS

formar : XDS (expression)

description : This command is used to display the X axis's trajectory information for the number of previous samples defined by the expression on the system console. The maximum

numerical value of the expression is 200.

YDIS

format : YDS (expression)

description : same as XDS.

STATUS

format : STA

description : This command displays the various gains currently set on the terminal for both actuators.

SAMPLING TIME

format : XXXX STI T

description : This command allows the alteration of the servo loop closure sampling time. The expression T is the required sampling time in microseconds.

4.3.5 INTERRUPT ROUTINES

The control program has two interrupt routines. The first source of interrupt is from the RS232 interface for the input of commands and the instructions one character at a time. This information is stored in a line buffer until a CR character is received. The CR informs the background routine idling in the IMMEDIATE mode to take appropriate action depending on the form of the instruction (i.e.numbered or unnumbered commands). The RS232 interrupt is also used for the emergency stop command "] " , and the stop at the end of the current move command " CONTROL C " .

The second source of continuous and synchronous interrupt is from a timer on the MP09

card for the purpose of the servo loop closures. By default, this interrupt occurs every 4 ms. Its frequency can be changed using the STI command described above. The servo loop closures interrupt routine carries out the following tasks sequentially.

I- Clears timer for the generation of the next interrupt.

II- Calculates incremental demand positions using the normalised table of a modified cycloid profile and a time base set by the MOVE command. Note that if the move command is inactive the stroke lengths are zero and thus no motion takes place.

III- Calculates the absolute demand positions.

IV- Samples the incremental actual positions and calculates their absolute values.

V- Calculates the error signal and multiplies them by their respective gains.

VI- Calculates the feedforward terms by the multiplication of the respective gains by the incremental demand position(s), or their rate of change in the case of acceleration feedforward.

VII- Calculates the velocity feedback terms by the multiplication of the respective incremental movement and the velocity feedback gains.

VIII- Sums up the control terms and converts the signals via the DACs for the drive system.

IX- Updates the various control cells for the next sample.

X- Returns the control of the program to the background routine.

4.4 EXAMPLES OF PART PROGRAM

Simple cyclic motion :

```
10 MOVE 200 200 400
20 DWL 300
```

```
30 MOVE -200 -200 400
40 DWL 300
50 GOTO 10
```

Description : Line 10 generates a profile motion 0f 200 encoder pulses in both X and Y actuator systems simultaneously. The motion will take place in 400 ms. Line 20 generates a dwell of period 300 ms. Line 30 is a reverse motion of the same length and taking the same time. This is followed in line 40 by a dwell period of 300 ms. In line 50 the GOTO command is used to transfer the control of the program back to line 10. Thus it is a continuous cyclic program.

Use of variables :

```
10 LET A=300
20 LET B=200
30 LET D=350
40 LET T=D+50
50 LET C=0
60 MOVE A B D
70 DWL T
80 MOV -A -B D
90 DWL T-50
100 LET C=C+1
110 IF C<99 THEN 60
120 PRINT " 100 CYCLES COMPLETED "
130 STOP
```

Description : Lines 10 to 50 set various arguments which will be used by the program. Lines 60 to 90 inclusive, are similar to the previous example. In line 100 a counter using the value of C is incremented. In line 110 the IF command is used to check whether the value of the counter has reached 100. If 100 cycles are carried out then the message in the PRINT command is sent to the console via the RS232 link. Finally the program will stop in line 130.

Use of the INPUT statement :

```
10 INPUT A
20 IF A=1 THEN 100
30 MOVE 250 0 600
40 MOVE 0 150 300
50 DWL 200
60 MOVE -250 -150 600
70 GOTO 10
100 MOVE 200 200 400
110 DWL 300
120 MOVE -200 -200 400
130 DWL 1000
140 GOTO 10
```

Description : The INPUT command in line 10 decides which cycle of motion should be carried out next. If the number entered is 1, then the motion cycle specified in lines 100 to 130 inclusive is carried out. Otherwise the motion cycle specified in lines 30 to 60 inclusive is executed. Note tha t this procedure could be extended using more IF commands. IN

addition, the GOSUB and RETURN s tatements may be used to make the part program more modular e.g.:

```
10 INPUT A
20 IF A=1 THEN 100
30 IF A=2 THEN 200
40 IF A=3 THEN 300
50 STOP
100 GOSUB 1000
110 GOTO 10
200 GOSUB 2000
210 GOTO 10
300  GOSUB 3000
310 GOTO 10
1000 MOVE 300 300 1500
1010 MOVE -300 -300 1500
1020 RETURN
ETC.
```

Thus the program has become more modular in its structure. Note that the use of INPUT command may be extended in the future to accept digital information from auxiliary devices such as sensors and limit switches or from another controller.

A second assembly language program, called Master, is run on the South West Tech. (SWT). This is a 6809-based microcomputer which provides facilities for listing, editing and assembling the assembly language program into machine code. The command available in the Master program. entered as a single letter, are :

D - Download

T - Transparent

S - Save data/program

L - Load program

M - Return to Monitor

F - Return to Flex

The first command, D, downloads the assembled PPIL program to the MC6809 controller. This is achieved by sending a preset number of bytes down the RS232 link between the SWT and the MC6809. The T command is a mode entered for running the PPIL program where the Master program appears 'transparent' i.e. the MC6809 is linked to the keyboard as if the SWT computer is not present. The S and L commands are used for storing and retrieving information on the SWT floppy disks. Data such as position and velocity for a move can be saved in data files, or a program of commands in the interpreter language of PPIL can be stored. Such a program can also be loaded into the PPIL software using the L command. The final two commands are both means of terminating the Master program. The M command returns the user to the monitor which contains simple commands to get the system running etc. whilst the F command returns the user to the Flex operating system on the SWT. Note that these commands do not affect the MC6809 controller which continues to run the PPIL program until reset or switched off.

5 POSITION CONTROL

5.1 OPEN LOOP DYNAMICS

In conventional geared robots, it is possible to consider the robot arm as a mechanically uncoupled system at low speeds. The control strategies adopted for such robots is often based on the above assumption. Consequently, the control signal for each joint is calculated without considering the motions of other linkages in the system. As was explained previously, direct drive arms are capable of moving at accelerations and speeds which are an order of magnitude higher than conventional robotic arms. This section deals with the open loop dynamics of I.C. direct-drive arm, leading to the formulation of the open loop transfer function.

In chapter 3 it was shown that when the parameters of the direct drive arm are chosen according to equation (3.87), the motors' torque equations simplify to:

$$T_i = H_{ii}\,\ddot{\theta}_i + T_{gi} \tag{5.1}$$

However, when the arm is mounted horizontally, the gravity torque T_{gi} is eliminated, thus:

$$T_i = H_{ii}\,\ddot{\theta}_i \tag{5.2}$$

Since there are no couplings between the links of the arm and the inertia is invariant, the whole system can be considered as two classical SISO (Single Input - Single Output) systems. As each actuator consists of the same components as the other one, it will be

sufficient to find the mathematical model for one axis only. It is clear that the real model of the actuator is infinitely complex; and certain assumptions have to be made in order to simplify the model for further analysis.

Figure (5.1) shows a block diagram model for the open loop dynamics of a single axis. The actuator consists of a permanent magnet d.c. servo motor driven via a standard servo amplifier which contains an analogue current feedback loop. The control computer sends a digital signal, calculated by the controller software, to the DAC (Digital to Analogue Convertor). The amplifier is then used to amplify the analogue signal from the DAC and output the resulting signal to the d.c. servo motor. The various terms in the expression H_{ii} and the inertia of the motor's rotor are collected together and called J. Since there are no gears between the motors and the links, frictional torques can be assumed to be negligible. If the arm is not completely balanced, then any interaction torquues can be treated as a disturbance.

The linear model assumes that the input signal $U_{d,}$ to the amplifier, never causes saturation of the amplifier's current or the motor's torque. Previous work has shown that this assumption is valid as long as the input signals which cause saturation are scaled (Hollerbach and Sahar). The motion generation software of the robot controller, will be explained in details in section (5.6). The modified cycloid motion with the appropriate parameters ensures the validity of the above assumption.

5.2 CONTINUOUS TIME TRANSFER FUNCTION

From the block digram in Figure (5.1), the open loop transfer function is derived as:

$$G_{\dot{\theta}}(s) = \dot{\theta}(s)/Ud(s) = \frac{K_c K_e K_a K_T}{JLs^2 + J(R+K_i)\, s + K_T(K_b+K_t)} \qquad (5.3)$$

K_a - Servo amplifier gain (V/V)

L - Motor armature inductance (H)

J -Motor and load inertia (Nm/rad.s^{-2})

K_e -Encoder gain (bit/rad)

K_c -Digital to analogue convertor (DAC) gain (V/bit)

K_t -Tachometer velocity feedback gain (V/rad.s^{-1})

K_T -Motor torque constant (Nm/A)

R - Motor armature resistance (Ω)

K_b -Motor emf constant (V/rad.s^{-1})

K_i - Amplifier current feedback gain (V/A)

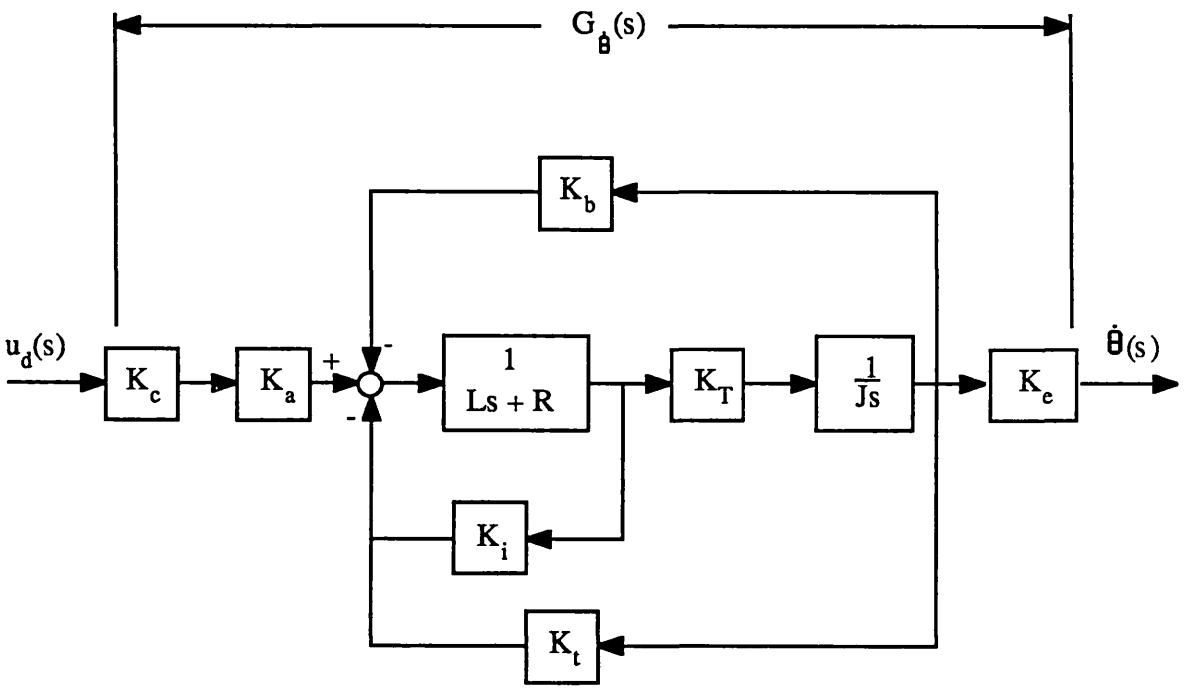


Fig. 5.1 Open loop block diagram of a single actuator system

The two poles of $G_{\dot{\theta}}(s)$ can be found by equating the denominator of equation (5.3) to zero.

$$JLS^2+J(R+K_i)S+K_T(K_b+K_t)=0 \tag{5.4}$$

$$P_{1,2} = \frac{-(R+K_i)}{2L} \cdot \left(1 \pm \sqrt{1 - \frac{4L\,K_T(K_b+K_t)}{J(R+K_i)^2}} \right) \tag{5.5}$$

5.2.1 SIMPLIFIED TRANSFER FUNCTION

If the value of the current feedback gain K_i, is adjusted so that :

$$J(R+K_i)^2 >> 4L\,K_T(K_b+K_t) \tag{5.6}$$

Then the two poles of $G_{\dot{\theta}}(s)$ will be given by:

$$P_1 = \frac{-K_T(K_b+K_t)}{J(R+K_i)} \tag{5.7}$$

$$P_2 \doteq \frac{-(R+K_i)}{L} \tag{5.8}$$

These two poles correspond to the time constants normally found in electrical servo systems, the mechanical and electrical time constants. The pole P_1 depends on the mechanical parameters of inertia and so is the mechanical time constant, τ_m whilst P_2 involves the electrical inductance and hence is the electrical time constant τ_e.

By substituting the above two time constants in equation (5.3), the following continuous time transfer function is obtained:

$$G_{\dot{\theta}}(s) = \frac{K_c K_e K_a / (K_b + K_t)}{(\tau_m s+1)(\tau_e s+1)} \tag{5.9}$$

where

$$\tau_m = \frac{J(R+K_i)}{K_T(K_b+K_t)} \tag{5.10}$$

and

$$\tau_e = \frac{L}{(R+K_i)} \tag{5.11}$$

In order to satisfy the condition in equation (5.6), K_i must be increased. This will decrease the value of the electrical time constant, but will increase the value of the mechanical time constant. At the same time, since there is no gearing between the motor and the link, the value of inertia J, is very large compared to that of the conventional robots. These effects combine to give

$$\tau_m >> \tau_e \tag{5.12}$$

This condition can be used to define a close first order approximation to equation (5.9),

$$G_{\dot{\theta}}(s) = \frac{k}{(\tau_m s+1)} \tag{5.13}$$

where

$$k = K_c K_e K_a / (K_b+K_t) \tag{5.14}$$

In the above equation K is defined as the steady-state open loop gain of the actuator. The open loop transfer function of the actuator has been approximated to a first order lag given in

equation (5.13). These assumptions have been made by considering the actuator in detail. It is evident that the real system has a higher order transfer function. However, it is always desirable to use the simplest possible transfer function which is adequate. The validity of the assumptions made can be further justified by experimental results.

5.3 DISCRETE TIME TRANSFER FUNCTION

In the previous section, the continuous time open loop transfer function of the actuator was derived. The transfer function is continuous time, since all the signals are analogue. A digital controller, MC6809, is employed to control the actuator. Cosequently, the whole system should now be treated as a sampled data system and hence the transfer function must be transformed from the s-plane to the z-plane.

A 12-bit digital to analogue convertor is on the input side of the open loop system given in Figure (5.1). The DAC is a zero-order hold which latches the last value it outputs. The continuous time transfer function of the zero-order hold is given by:

$$\frac{1- e^{-sTs}}{s} \tag{5.15}$$

where T_S is the sampling period.

Since the actuator is intended to be used as a position control device, an up-down counter is added on the output side of the open loop system after the incremental optical encoders. The up-down counter integrates the velocity to give position. Thus, by definition the continuous time transfer function of the up-down counter is :

1/s

Referring to the block diagram in Figure (5.1), it can be deduced that:

$$G_{\dot{\theta}}(s) = \frac{\dot{\theta}(s)}{U_d^*(s)} = \frac{1-e^{-sTs}}{s} \cdot \frac{k}{(\tau_m s+1)} \tag{5.16}$$

$$= k.\,(1-e^{-sTs}).\left(\frac{1}{s} - \frac{1}{s+1/\tau_m}\right) \tag{5.17}$$

Transforming into the z-plane, and re-writing e^{-T_s/τ_m} as e_m,

$$G_{\dot{\theta}}(z) = \frac{\dot{\theta}(z)}{U_d(z)} = k.\frac{z-1}{z} \cdot \left(\frac{z}{z-1} - \frac{z}{z-e_m}\right) = \frac{k.(1-e_m)}{z-e_m} \tag{5.18}$$

The above equation gives the transfer function from the input signal to the output velocity but the actuator is used as a position control device. Hence it is desired to find the transfer function to a position output and this is easily achieved in continuous-time by simply adding an integrator as shown in Figure (5.2). In order to determine the transfer function in discrete-time it is necessasry to return to continuous-time and transform to the z-plane rather than starting from the discrete-time velocity transfer function. So, in the s-plane, the position transfer function is :

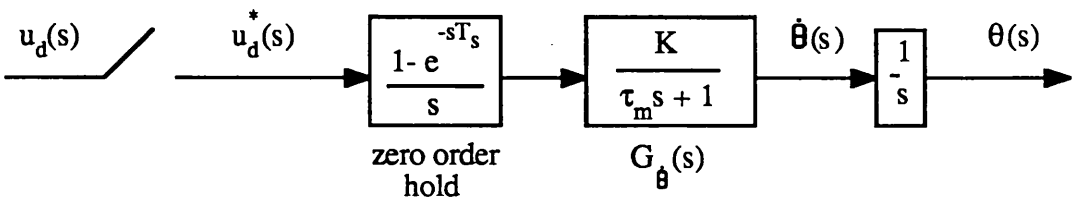


Fig. 5.2 Block Diagram of the Sampled Data Open Loop System

$$G_\theta(s) = \frac{\theta(s)}{U_d^*(s)} = \frac{1-e^{-sTs}}{s} \cdot \frac{k}{s(\tau_m s+1)} \tag{5.19}$$

$$= k.(1-e^{-sTs}).\left(\frac{1}{s^2} - \frac{\tau_m}{s} + \frac{\tau_m}{s+1/\tau_m}\right) \tag{5.20}$$

Now the above transfer function in equation (5.20) can be transformed into the z-plane as follows:

$$G_\theta(z) = \frac{\theta(z)}{U_d(z)} = k.\frac{z-1}{z} \cdot \left(\frac{T_s\, z}{(z-1)^2} - \frac{\tau_m\, z}{z-1} + \frac{\tau_m\, z}{z-e_m}\right) =$$

$$= \frac{k.\left[z.(T_s - \tau_m.(1-e_m)) + \tau_m.(1-e_m) - T_s.e_m\right]}{(z-1)\cdot(z-e_m)} \tag{5.21}$$

Hence the open loop position transfer function in discrete time has finally been derived and is stated in equation (5.21).

5.4 DERIVATION OF THE STATE SPACE MODEL $G_\theta(S)$

In classical control theory the starting point is nearly always the transfer function of the system to be controlled and this has already been derived in both continuous and discrete time. For the design of a controller using modern control theory however the system needs to be described in terms of a state space model, in discrete-time due to to the digital controller.

There are two methods of determining this model, one by describing the the continuous-time system by a state space model, also in continuous time and then discretising. This is the method followed in this section whilst the next section confirms the result by transforming the discrete-time transfer function into a state space model.

The block diagram in Figure (5.2) shows the open loop sampled data system. It is now necessary to consider just the continuous-time section so that the zero order hold is now ignored. The position transfer function can be easily derived from equation (5.13) by the addition of an integrator which corresponds to multiplying the denominator of $G_{\dot{\theta}}(s)$ by the Laplace transform operator, 's'. The result, given by equation (5.22) is a second order system of class 1 i.e. with a single free integrator.

$$G_{\theta}(s) = \frac{k}{s(\tau_m s+1)} \tag{5.22}$$

As the system is second order, two states are chosen and the obvious choices are position and velocity. Using equation (5.22) and the fact that velocity is the derivative of position, the state space model is,

$$d/dt \cdot \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -1/\tau_m \end{bmatrix} \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ k/\tau_m \end{bmatrix} .u_d(t) \tag{5.23}$$

Using the normal convention, the two matrices above are written as **A** and **B** and the two states are written as a single vector, $\underline{\theta}$, the state vector. To transform the state model to discrete-time it is necessary to solve the state transition equation for the continuous system over the interval $[kT_s\ ,\ (k+1)T_s]$.

$$\underline{\theta}[(k+1).T_s] = e^{A.Ts}.\underline{\theta}(k.T_s) + \int_{k.T_s}^{[(k+1).T_s} e^{A.[(k+1).Ts - t']}. .B. u_d(k.T_s)\ dt' \quad (5.24)$$

The first term on the right hand side involves the state transition matrix which can be calculated as follows :

$$e^{A.Ts} = L^{-1}\ [\text{adj}\ (sI - A)/\det\ (sI-A)] = \begin{bmatrix} 1 & \tau_m\ (1-e_m) \\ 0 & e_m \end{bmatrix} \quad (5.25)$$

The integral term on the right hand side of equation (5.24) can be evaluated using the derivation of the state transition matrix in equation (5.25) and by noting that the **B** matrix and the control signal, $u_d(k.T_s)$ are independent of time, t and so can be taken outside the integral sign.

$$\int_{k.T_s}^{(k+1).T_s} e^{A.[(k+1).Ts - t']}. .B. u_d(k.T_s)\ dt' =$$

$$\left[\int_{k.T_s}^{(k+1).T_s} \begin{bmatrix} 1 & \tau_m\ (1-e^{-[k+1).Ts\ -\ t']/\tau m}\ ) \\ 0 & e^{-[k+1).Ts\ -\ t']/\tau m} \end{bmatrix} dt' \right] .B.\ u_d\ (k.Ts) \quad (5.26)$$

Substituting equations (5.25) and (5.26) back into equation (5.24) and writing

$\underline{\theta}[(k+1).T_s]$ as $\underline{\theta}(k+1)$,

$\underline{\theta}(k.T_s]$ as $\underline{\theta}(k)$,

$u_d(k.T_s)$ as $u_d(k)$

the state equation becomes,

$$\underline{\theta}(k+1) = \mathbf{A}_p.\, \underline{\theta}(k) + \mathbf{B}_p.\, u_d(k) \tag{5.27}$$

where

$$\mathbf{A}_p = \begin{bmatrix} 1 & \tau_m\,(1-e_m) \\ 0 & e_m \end{bmatrix}$$

$$\mathbf{B}_p = \begin{bmatrix} k.T_s - k.\tau_m\,(1-e_m) \\ k.(1-e_m) \end{bmatrix}$$

and the corresponding output equation to complete the state space model is simply,

$$\theta(k) = \mathbf{C}_p.\, \underline{\theta}(k) \tag{5.28}$$

where $\mathbf{C}_p = [1 \quad 0]$

The two equations (5.27) and (5.28) are the final discrete-time, second order state space model for each of the two actuators. It is now possible to use the model to design a controller. It is important to note at this stage that the model developed so far is deterministic, that is no account has been made for noise disturbances. In practice all the signals will be noisy; the analog ones due to picking up stray electrical noise etc. and the digital ones due to truncation errors etc. However it is not considered necessary in this study to include stochastic models as this would have diverted the main thrust of the investigation.

5.4.1 DERIVATION OF THE STATE SPACE MODEL FROM $G_\theta(Z)$

Although the state space model has already been derived in the previous section, an alternative derivation is presented here to confirm the result. The analysis starts from the discrete-time transfer function, in fact both the velocity and position transfer functions in discrete time are required. From the velocity transfer function in equation (5.18), multiplying by the denominator of $G_{\dot\theta}(z)$ gives :

$$(z-e_m).\dot\theta (k)= k.(1-e_m).u_d(k) \tag{5.29}$$

and remembering that the z operator is the forward shift operator,

$$\dot\theta (k+1)=e_m.\dot\theta (k) + k.(1-e_m).u_d(k) \tag{5.30}$$

Following a similar analysis starting from equation (5.21) for the position gives,

$$(z-1).\theta(k)= \frac{k.(1-e_m).\tau_m.(1-e_m)}{z-e_m}.u_d(k) + k.\left[T_s - \tau_m.(1-e_m)\right].u_d(k) \tag{5.31}$$

but from equation (5.18),

$$\frac{k.(1-e_m)}{z-e_m}.u_d(k) =\dot\theta(k) \tag{5.32}$$

Substituting this into the first term on the right hand side of equation (5.31) and again using the z transform as the forward shift operator,

$$\theta(k+1)=\theta(k)+\tau_m.(1-e_m).\dot{\theta}(k)+k.\left[T_s-\tau_m.(1-e_m)\right].u_d(k) \qquad (5.33)$$

If equations (5.30) and (5.33) are combined into a matrix from then it is clear that the result is the same as that given in equation (5.27). Hence the above analysis has independently confirmed the derivation of the state space model.

5.5 EVALUATION OF SYSTEM PARAMETER VALUES

The mathematical model of each axis has been determined in both the continuous time and discrete time. In order to find a suitable controller for each axis, the parameters that appear in the models have to be determined first. Referring to Figure (5.3), the amplifier gain K_a and the tachometer feedback gain K_t are adjusted to give a steady state speed of 17.5 rad/sec for the maximum value of the control signal U_d, which is 10 volts.

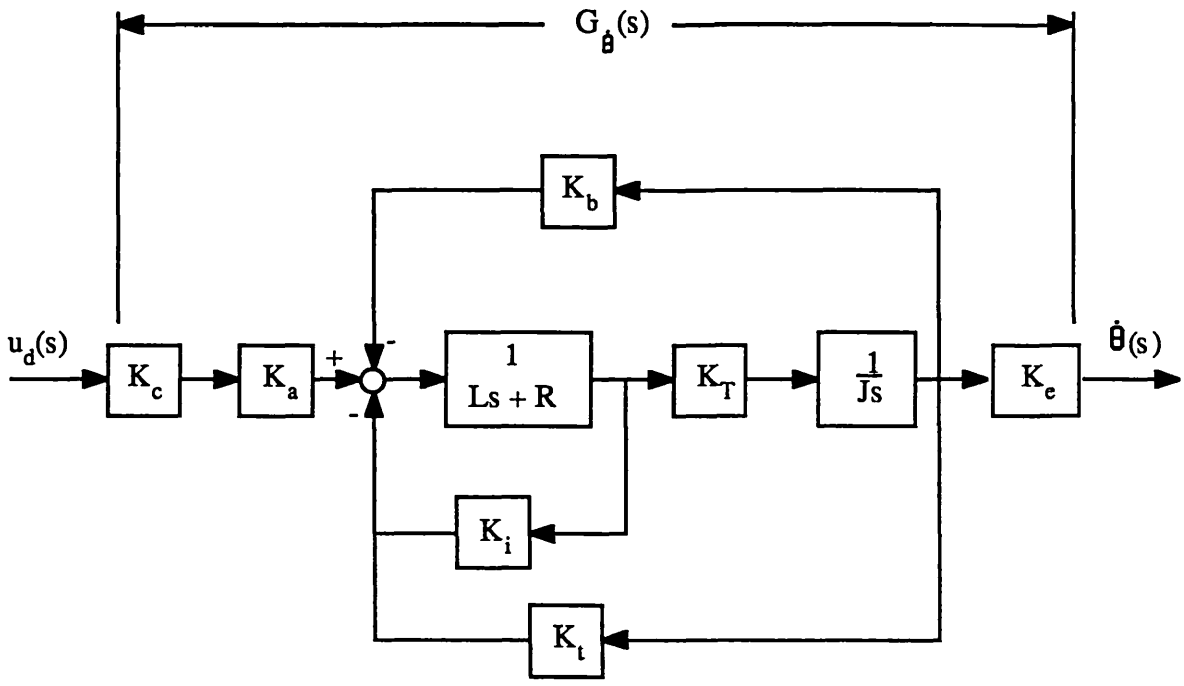


Fig. 5.3 Open loop block diagram of a single actuator system

The digital to analogue converter used has a resolution of 12 bits, producing a nominal ±10 V output to the motor amplifier. The 500 pulse incremental optical encoders together with the *4 multiplication gives the Basic Angle Unit (BAU) equal to 2π/2000 radians. Hence, from equation (5.14) the steady state open loop of the actuator system is given by:

$$k = \frac{K_e \cdot K_e \cdot K_a \cdot}{K_b + K_t} = \frac{10}{2^{11}} \cdot \frac{2000}{2\pi} \cdot \frac{17.5}{10} = 2.72\ s^{-1} \qquad (5.34)$$

5.5.1 EVALUATION OF THE MECHANICAL TIME CONSTANTS OF THE ACTUATOR SYSTEMS

It would be possible to calculate the value of the mechanical time constants for both

actuator systems from the expression in equation (5.10). However, the value of K_i is not known and is not easily determined. In addition, this would only result ia an approximate value. Consequently, it is necessary to use the more conventional, and more complicated method of frequency response tests to determine the values of τ_m for each axis.

For the frequency response tests, a sinusoidal input of variable frequency was input to the amplifiers and the velocity output was measured via the tachometer signal. For Joint 1, The mechanical time constant was found to be 23 ms. For joint 2, carrying considerably less inertia, the value of the mechanical time constant was found to be 8 ms. It is interesting to note that the mechanical time constants of the motors when disconnected from the arm are under 1 ms. Referring to equation (5.10), the substantial increases in the values of the mechanical time constants show the extent of the motion control problem due to the much larger reflected arm inertia in a direct drive arm. An alternative appraoch to finding the mechanical time constants is presented here which confirms the values obtained via the frequency response tests.

The continuous time open loop transfer function of the actuator system is given by:

$$\frac{k}{s\,(\tau_m s + 1)} \tag{5.35}$$

where k was found to be 2.72 s^{-1}.

The closed loop transfer function with a unity feedback gain and a microprocessor (propotional) gain of K_p can be derived from Figure (5.3) as:

$$\frac{K_p\,k}{K_p\,k + s\,(\tau_m s + 1)} \tag{5.36}$$

Let $K_p k = \mathbf{K}$

The above closed loop transfer function is a second order lag generally represented as:

$$\frac{\omega_n^2}{s^2 + 2\zeta\,\omega_n s + \omega_n^2} \tag{5.37}$$

Re-arranging equation (5.36) gives:

$$\frac{K/\tau_m}{s^2 + (1/\tau_m)s + K/\tau_m} \tag{5.38}$$

The objective is to find the value of the mechanical time constant τ_m. By comparing equations (5.37) and (5.38) it can be concluded that :

$$\omega_n^2 = \mathbf{K}/\tau_m \tag{5.39}$$

$$\omega_n = (\mathbf{K}/\tau_m)^{1/2} \tag{5.40}$$

$$2\zeta\,\omega_n = 1/\tau_m \tag{5.41}$$

$$2\zeta\,(\mathbf{K}/\tau_m)^{1/2} = 1/\tau_m \tag{5.42}$$

$$\zeta = 1/[2(\mathbf{K}*\tau_m)^{1/2}] \tag{5.43}$$

$$\tau_m = 1/4\,\zeta^2\,\mathbf{K} \tag{5.44}$$

Various step input tests were carried out in order to obtain the value of **k** which would give a critically damped closed loop response (i.e. no overshoots) $\zeta=1.0$.

Figures (5.4) and (5.5) show the step input tests for joints 1 and 2 respectively. As can be seen from the graphs, there are no overshoots in either responses. The corresponding values of **k** were found to be 10.88 s^{-1} and 31.28 s^{-1} for joints 1 and 2 respectively.

For jont 1 : $\tau_{m1} = 1/(4 * 1^2 * \mathbf{K})$

$\therefore \quad \tau_{m1} = 23$ ms

For joint 2 : $\tau_{m2} = 1/(4 * 1^2 * K)$

$\therefore \quad \tau_{m2} = 8$ ms

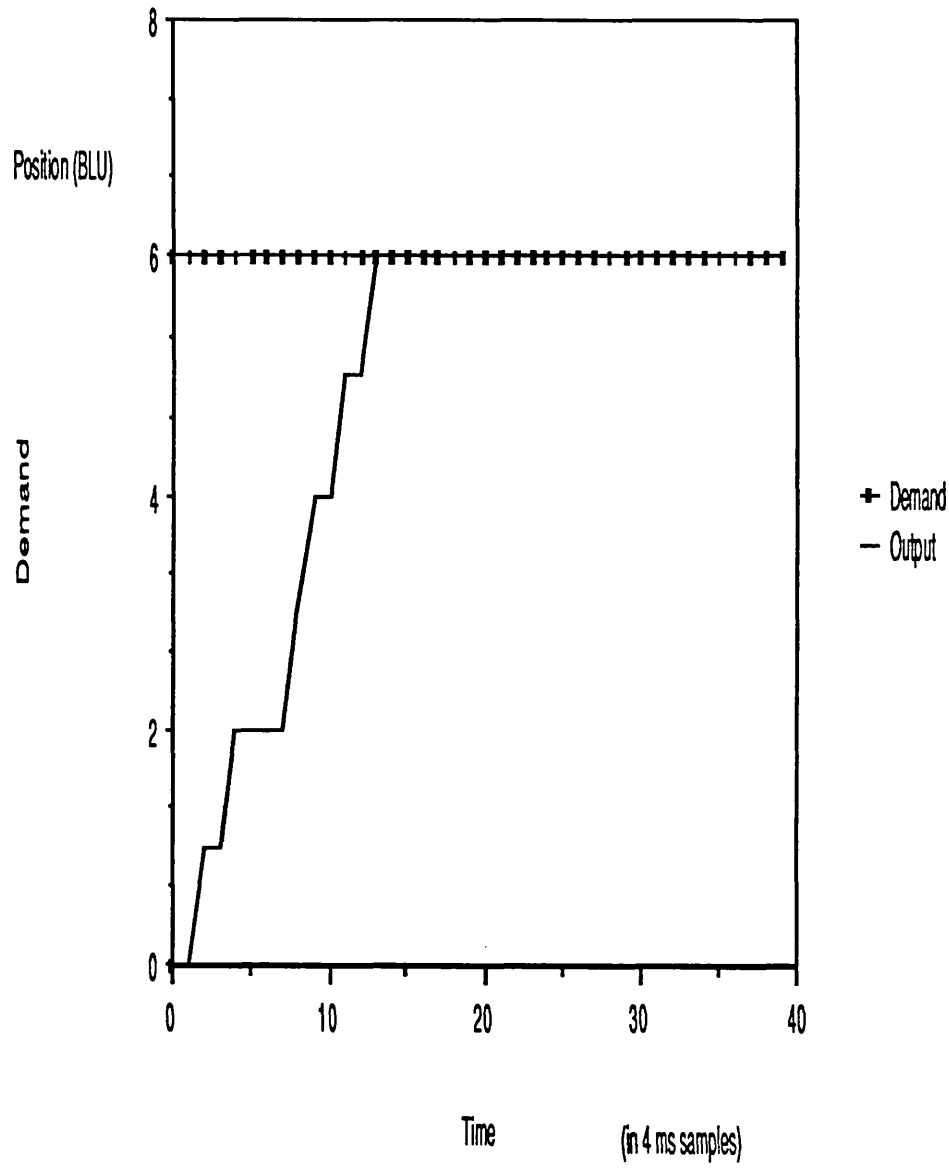


Fig. 5.4 Step Response for Joint 1

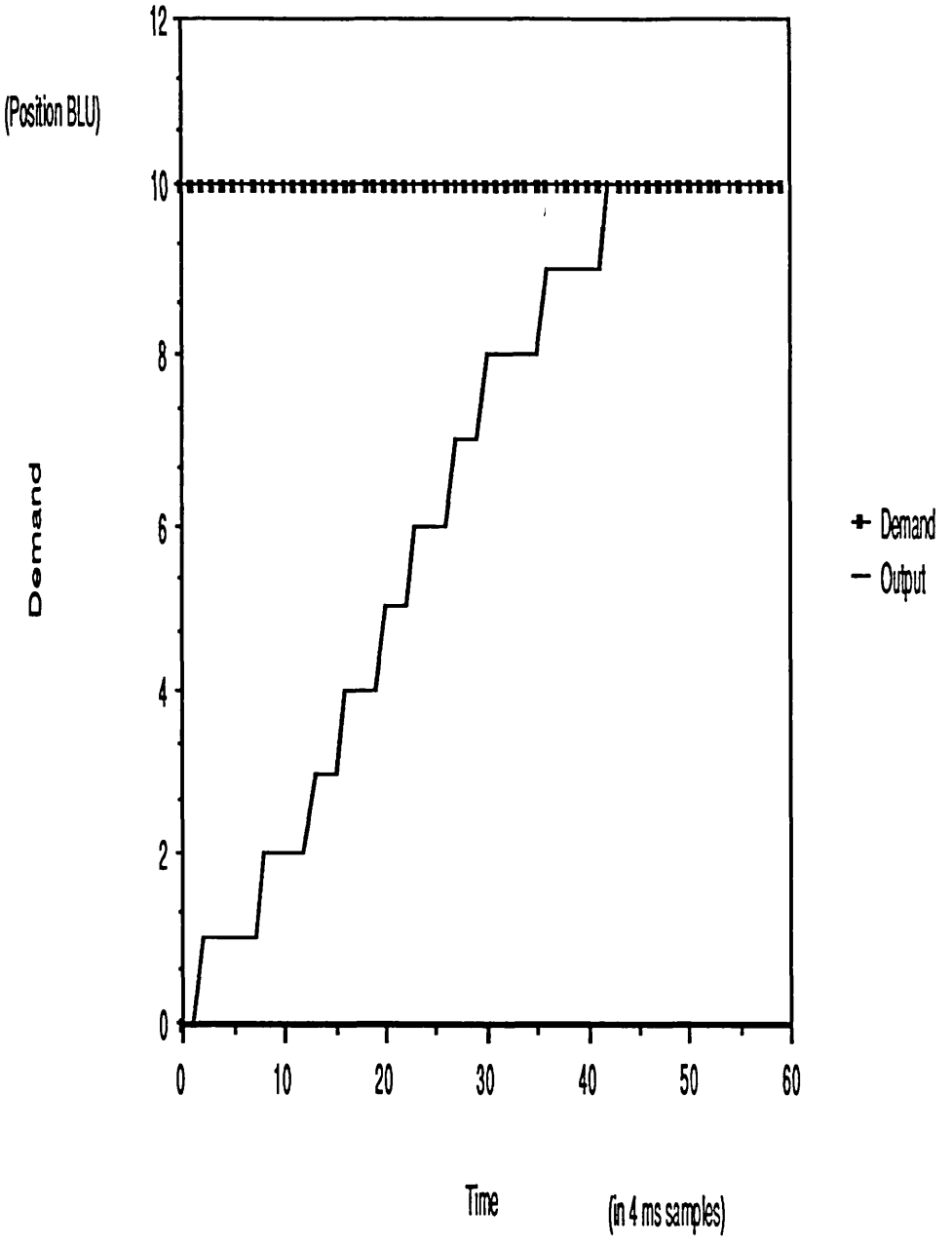


Fig. 5.5 Step Response for Joint 2

5.5.2 SAMPLING RATE SELECTION

It is known from the sampling theorem that the sampling frequency should be at least twice as fast as the highest frequency ω_o contained in the sampled signal (Koren). Therefore,

$$1/T_S = f_S \geq 2 . (\omega_o / 2 \pi) \tag{5.45}$$

The motor-amplifier-tacho combination behaves like a low pass filter with a critical frequency of $\omega_c=1/t$. Substituting $\omega_0 > \omega_c$ in equation (5.45) yeilds the necessary condition for selecting the sampling period:

$$T_S / \tau_m < \pi \tag{5.46}$$

The sampling period, T_S which although variable is generally set to be 4 ms. This is short enough to prevent aliasing and long enough to perform the necessary calculations.

In the case of the third order model for the actuator systems, the electrical time constant is anticipated to be of the order of 1 ms. Following the above argument would necessitate a sampling period less than or equal to 3 ms. Clearly this is not possible given the capabilities of the MC6809 computer. The third order model of the system is considered in chapter 7.

5.6 MOTION GENERATION

The motion trajectories along the path specified by the MOVE commands are genrated from a table of normalised data stored in memory as part of the PPIL software.

At every sample time, the next incremental demand is generated by the computer by taking an appropriate value from the table, the value being determined by the contents of a base speed demand generator timer whose value is set by the T parameter of the MOVE command. The value extracted from the table is then multiplied by the stroke length X and/or Y to provide the instantaneous demand position. The T parameter can be up to 5000 (5 s).

When the arm is to follow a planned trajectory in joint co-ordinates it is desirable for input profile to exhibit smooth velocity and acceleration characteristics. In addition, the input command must not cause amplifier current or motor torque saturation, otherwise the linear control model will become invalid. Therefore, the dynamic trajectories of the manipulator must be time-scaled. J. M. Hollerbach, has presented a method which allows modification of movement speed without complete dynamic recalculation. By exploiting this property, it can be determined whether a planned trajectory is dynamically realizable given actuator torque limits, and if not, how to modify the trajectory to bring it within dynamic and actuating constraints. The second method is based on the theory of cams and is of particular use in robot arms which are dynamically decoupled and have invariant inertia, as in I.C. direct drive arm. This method does not require any on-line inverse dynamic calculations, and hence is more advantageous. In the following sub-sections a variety of motion trajectories are discussed. The modified cycloid motion is the input profile chosen for implementation on the I.C. direct-drive arm. This profile exhibits a very smooth velocity and acceleration chracteristics, as can be seen in Figure (5.10).

5.6.1 CONSTANT VELOCITY MOTION

From Figure (5.6), it can be deduced that :

$$y = h\frac{t}{T} \tag{5.47}$$

$$V = \frac{dy}{dt} = \frac{h}{T} \tag{5.48}$$

$$\frac{d^2y}{dt^2} = a = 0 \tag{5.49}$$

where $0<t<T$

except at t=0 and t=T where the acceleration is theoretically infinite.

Theoretically, an instantaneous jump from zero velocity to any value of velocity results in an infinite acceleration and since inertia is involved in the system, this requires an infinitely large force/torque. In practice this is impossible and it will cause torque saturation.

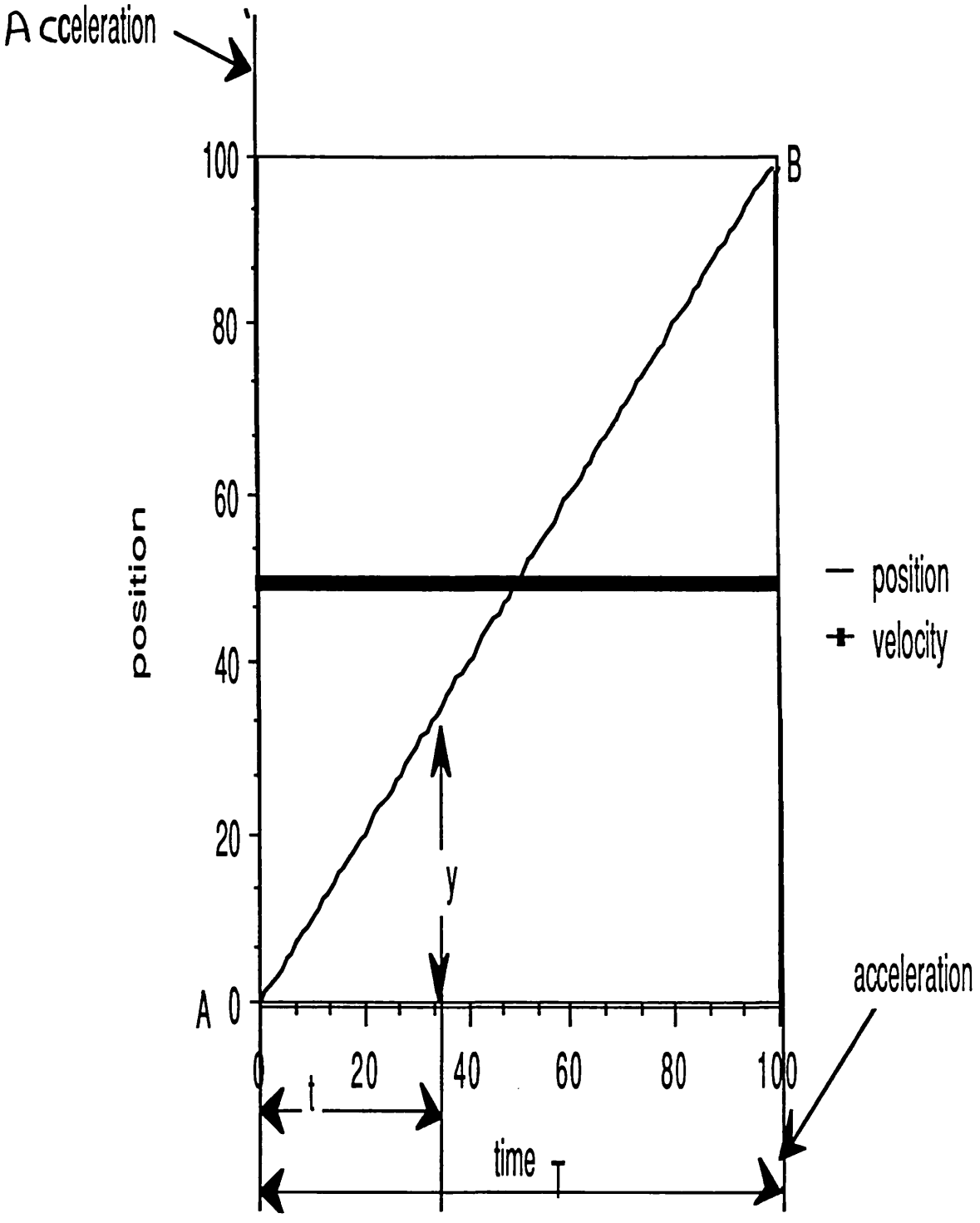


Fig. 5.6 Position, Velocity and Acceleration Curves For Constant Velocity Motion

5.6.2 PARABOLIC MOTION

From Figure (5.7), it can be deduced that :

$$y = 2h\frac{t^2}{T^2} \tag{5.50}$$

$$V = 4\frac{h}{T^2}.t \tag{5.51}$$

$$a = 4\frac{h}{T^2} \tag{5.52}$$

where $0 \leq t \leq T/2$

Examination of the above formula shows that the velocity is zero when t=0, and that the acceleration is constant. These formulae however apply only from t=0 to t=T/2. It is assumed that when this point is reached, y=h/2 the curve has to be inverted so that the velocity decreases as the rise continues and the velocity becomes zero when t=T and y=h. The most important advantage of this curve is that for a given h and T, it produces a relatively small acceleration. However, because of the sudden change in acceleration at the beginning, middle and the end of the motion, shocks are produced.

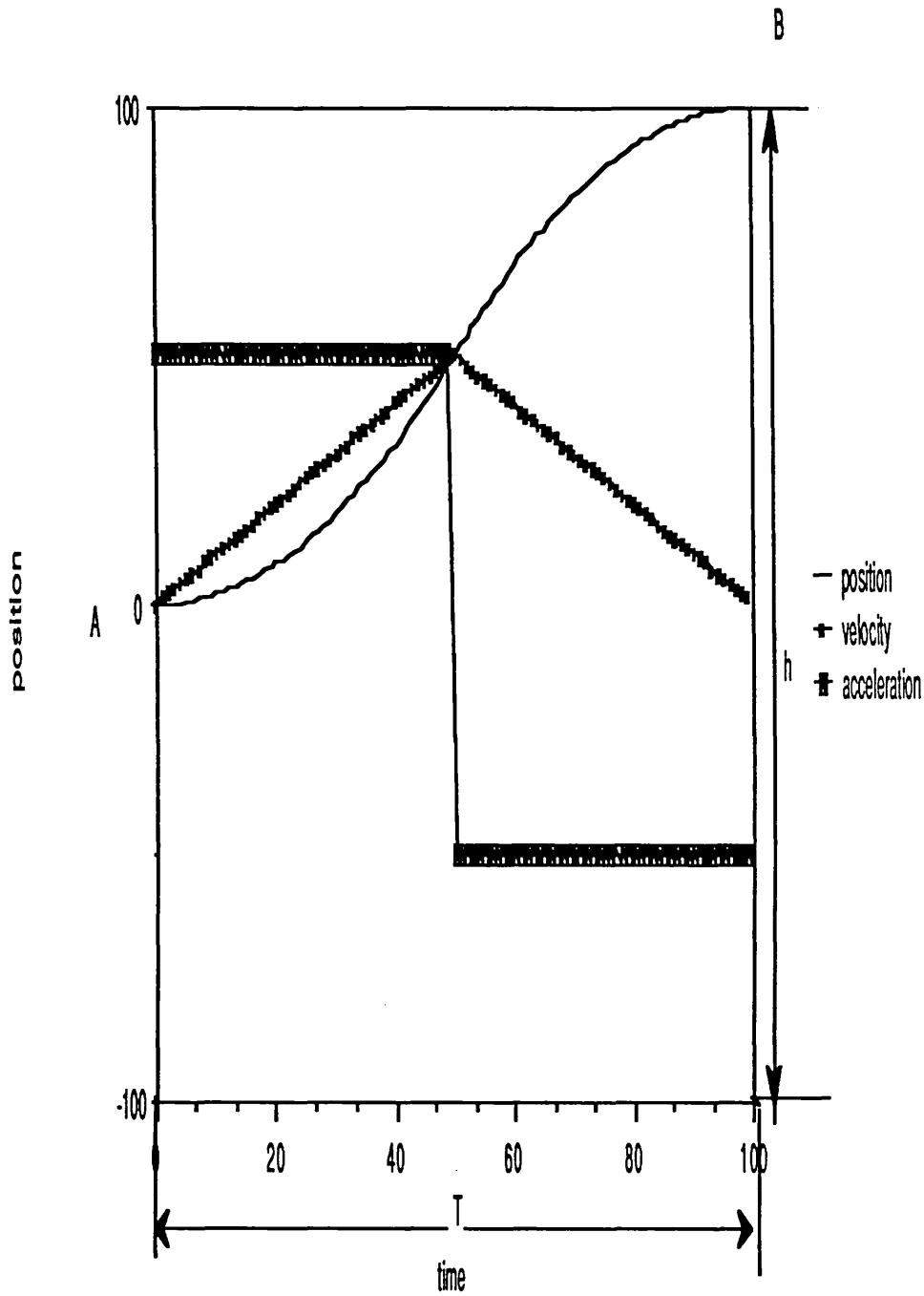


Fig. 5.7 Position, Velocity and Acceleration Curves for the Parabolic Motion

5.6.3 SIMPLE HARMONIC MOTION

Referring to Figure (5.8) :

$$y = \frac{h}{2}(1 - \cos(\pi t/T)) \tag{5.53}$$

$$V = \frac{h}{2T} \cdot \pi \cdot \sin(\pi t/T) \tag{5.54}$$

$$a = \frac{h}{2T^2} \cdot \pi^2 \cos(\pi t/T) \tag{5.55}$$

where $0 \leq t \leq T$

Smoothness in velocity and acceleration during the stroke is the advantage inherent in this curve. However, the step changes in acceleration at the beginning and the end of the motion tend to cause "jerky" motion.The maximum value of acceleration is $h\pi^2/2T^2$.

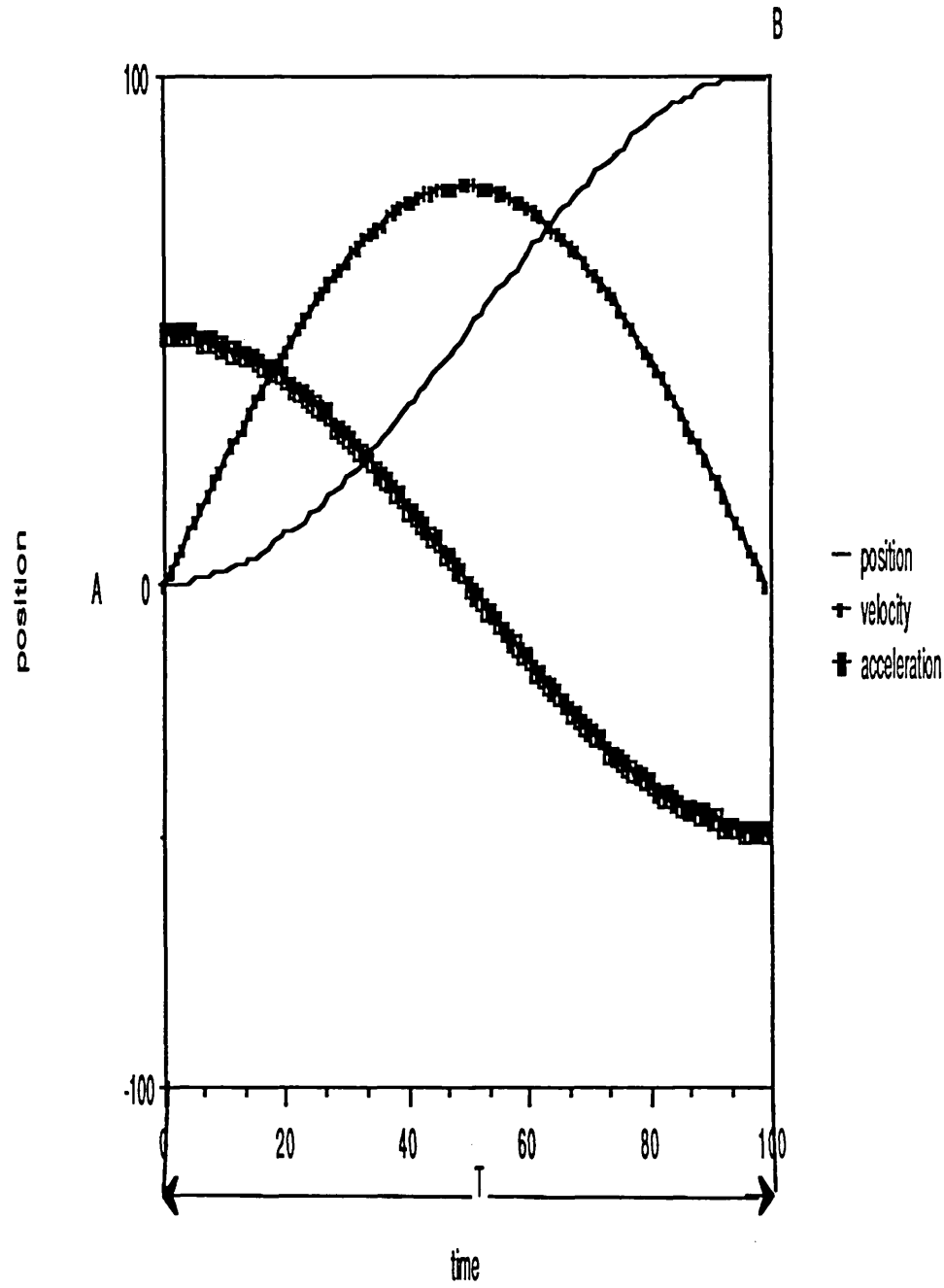


Fig. 5.8 Position, Velocity and acceleration Curves for Simple Harmonic Motion

5.6.4 CYCLOID MOTION

Referring to Figure (5.9):

$$y = h\left(\frac{t}{T} - \frac{1}{2\pi}\sin(2\pi t/T)\right) \tag{5.56}$$

$$V = \frac{h}{T}(1 - \cos(2\pi t/T)) \tag{5.57}$$

$$a = \frac{2\pi h}{T^2}.\sin(2\pi t/T) \tag{5.58}$$

where $0 \leq t \leq T$

This time displacement curve has excellent acceleration characteristics, i.e. there are no abrupt changes in its associated acceleration curve. However, the maximum value of the acceleration is $2\pi h/T^2$.

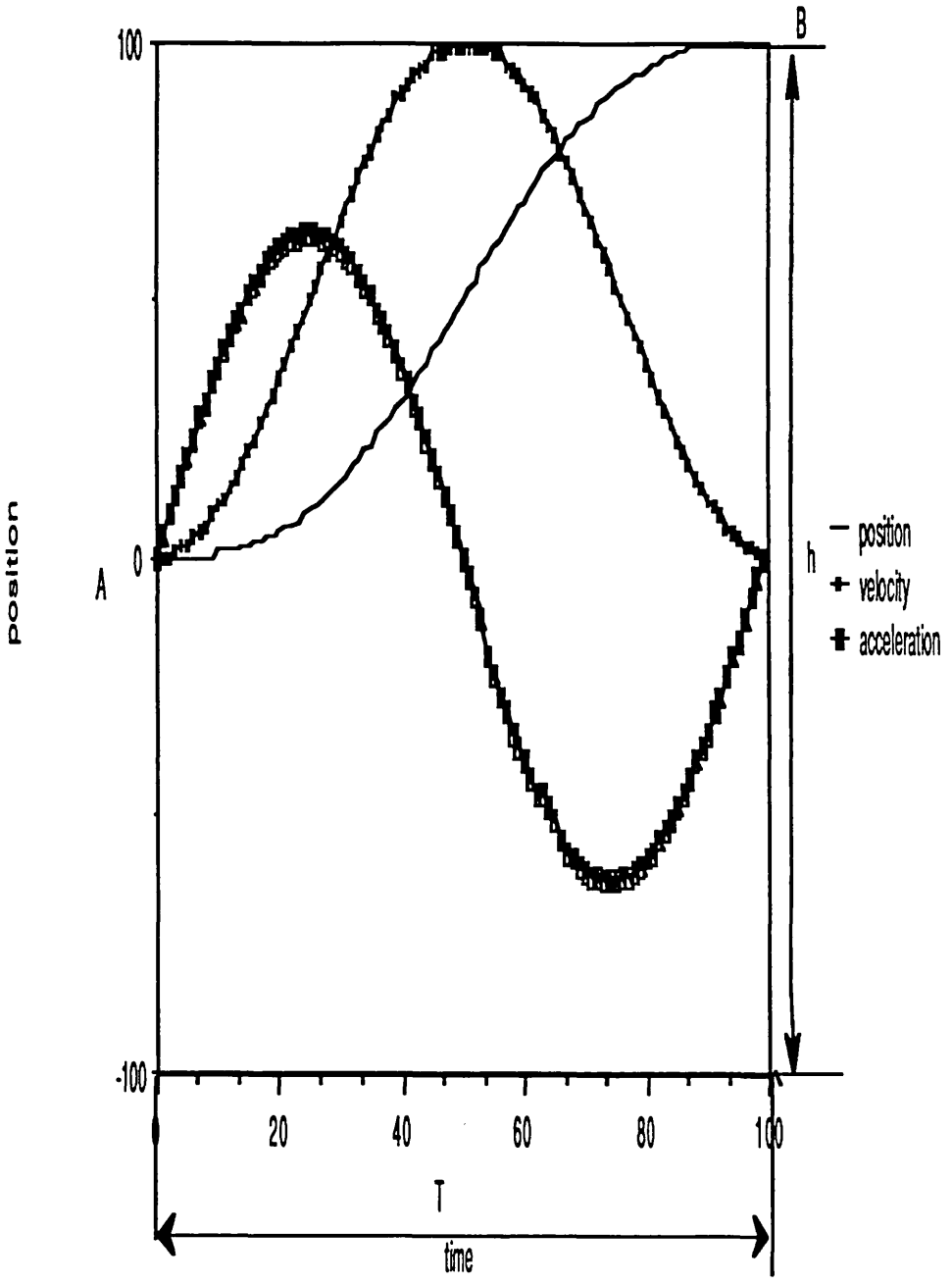


Fig. 5.9 Position, Velocity and acceleration Curves for the Cycloid Motion

5.6.5 MODIFIED CYCLOID MOTION

Referring to Figure (5.10), the equation for the modified cycloid is given by :

$$y = h[tm/T - (1/2\pi)*\sin(2\pi .tm/T)] \tag{5.59}$$

where $$tm = t + \frac{T}{2\pi} .x .\sin(2\pi.t/T)$$

$$x = \frac{\tan\delta}{\tan\gamma}$$

The velocity V is found from

$$V = \frac{1 - \cos(2\pi.t/T)}{1 - x.\cos(2\pi.t/T)} \cdot \frac{h}{T} \tag{5.60}$$

$$a = \frac{2\pi.(1 - x).\sin(2\pi.t/T)}{(1 - x.\cos(2\pi.t/T))^2} \cdot \frac{h}{T^2} \tag{5.61}$$

The acceleration a is obviously a function of x and t, and for a certain stroke h and time T, a certain maximum value of acceleration is obtained which is a function of x. For all these, there is a value of x=x optimum which will give the lowest possible maximum value of the acceleration, and it can be shown that (Jensen) :

$$x \text{ optimum} = 0.134$$

For this value of x, the maximum value of the acceleration becomes :

$$a_{max\ (optimum)} = 5.89\ h/T^2$$

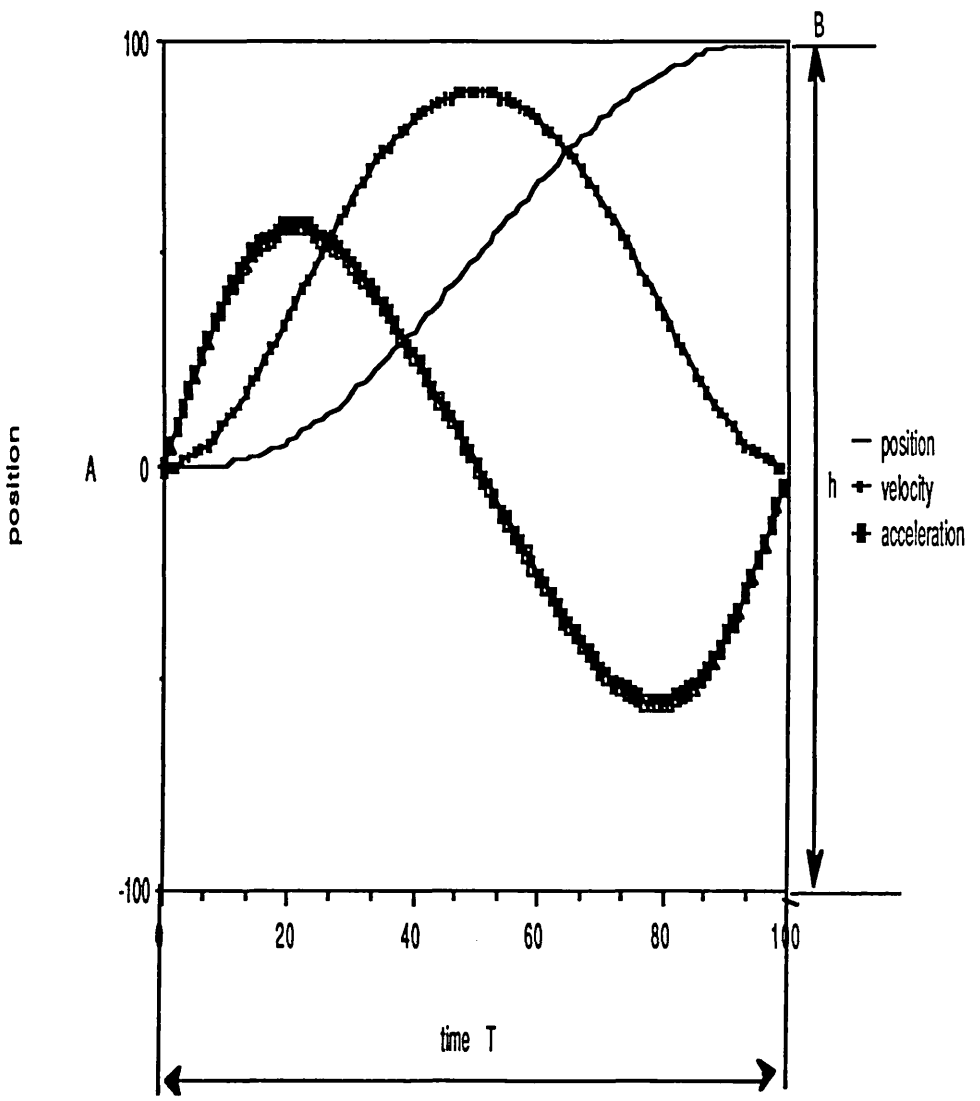


Fig. 5.10 Position, Velocity and Acceleration curves for the Modified Cycloid Motion

As was explained above, the modified cycloid profile is chosen for implementation on the I.C. direct-drive robot arm. The modified cycloid profile is superior to the cycloid profile for it enables the end effector to travel from point A to point B (Figure 5.10), in time T, with a maximum acceleration of $5.89h/T^2$ as opposed to $2\pi h/T^2$. Consequently, given a motor with a continuous stall torque of T_i and a mass with an invariant inertia of (H_{ii} + $I_{motor's\ rotor}$), the parameter T must be chosen such that :

$$T^2 \geq \frac{5.89 * h * (H_{ii} + I_{motor's\ rotor})}{T_i} \tag{5.62}$$

If the parameter T is chosen according to the above inequality, then the torque that the motor has to deliver never reaches saturation level and thus the control system does not become nonlinear. It is evident that the above statement would only hold when the end effector can precisely follow the reference input and that the motor's amplifier does not reach current saturation.

5.7 ROBOT JOINT CONTROL

In the previous sections, the direct-drive has been considered in detail and a model obtained for the system dynamics which has essentially defined the system as two single-input single-output joint actuator subsystems with invariant dynamics. Consequently, it is now possible to consider the problem of tracking control.

5.7.1 CONTROL STRATEGIES

The method by which specified input demands to the robot joints are converted to the output demands to the robot drives is termed the control strategy. The main types of control strategy relevant to robotics are as follows, (Brady et al 1982):

1- Open loop control :

This is the simplest form of control strategy. The joint input demands are converted directly to the output drive demands with no feedback from the mechanical system. This form of control is used typically with stepper motor drive systems.

2- Linear control :

Linear control incorporates a set of linear feedback relationship. A linear feedback law may have structure constraints in which only certain inputs and outputs are interconnected, as with the case when each joint is considered independently from any other joints.

3- Non-linear control :

This class of control law inculdes bang-bang control, global non-linear control, model reference control and self-tuning or adaptive control.

Most of the present industrial robots incorporate linear control, usually independent joint control. The common control law applied is of proportional-integral-derivative (PID) type.

5.7.2 CONTROL LAW SELECTION

The requirement for a minimum tracking error when the system is subjected to a constant acceleration/deceleration input leads to the use of the following control law :

$$U_d(k)=Kp[X_{ref}(k)-X(k)] + K'_{vff} * V_{ref}(k) + K'_{aff} * A_{ref}(k) \qquad (5.63)$$

The first term in the control law is in proportion to the instantaneous position error at the sampling intervals. The second term is proportional to the reference velocity and the third term is proportional to the reference acceleration/deceleration at the sampling intervals. In the following analysis, only joint 1 is considered in detail as joint 2 is identical to joint 1, except that the mechanical time constant of joint 2 is smaller than that of joint 1. However, the control design principles remain the same.

Determination of K_p

The proportional gain K_p was set so that the response of an equivalent continuous system had a damping factor of $\zeta=1.0$

Since

$$\zeta = \frac{1}{2*(k*K_p*\tau_m)^{1/2}} \tag{5.64}$$

Therefore,

$$K_p = \frac{1}{4*\zeta^2*K*\tau_m} = \frac{1}{4*(1.0)^2*2.72\ *0.023} = 4 \tag{5.65}$$

Determination of K_{vff}

When the arm is required to perform tracking at high speeds, the input signal is essentially a ramp which can be described as At, and that its Laplace transform is A/s^2. Referring to the block diagram given in Figure (5.11), the velocity feed-forward gain should be set to eliminate following errors due to a constant velocity demand.

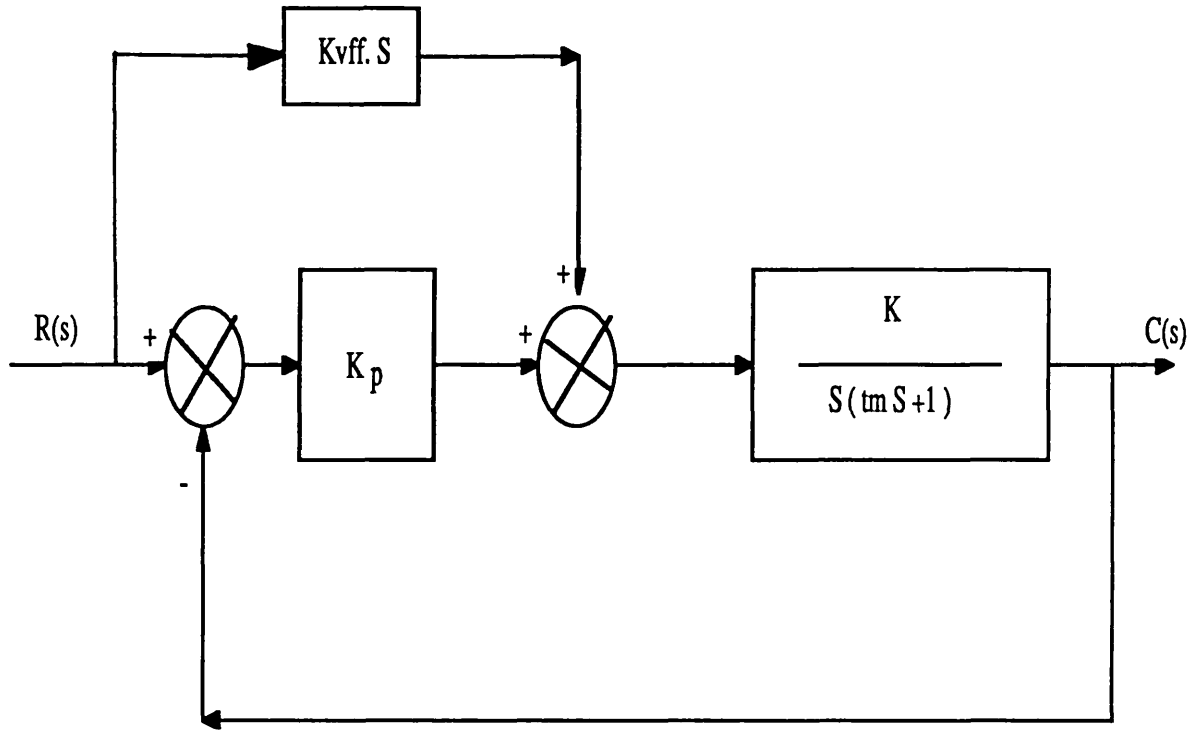


Fig. 5.11 Block Diagram of a Single Joint Actuator Control System With Velocity Feed-Forward

The positional error is given by :

$$E(s)= R(s) - C(s) \tag{5.66}$$

$$E(s)= R(s).\frac{s(\tau_m s +1) - k K_{vff}\ s}{s(\tau_m s +1) + k K_p} \tag{5.67}$$

where $R(s) = A/s^2$ for a ramp input.

The steady state following error e_{ss} is found from the final value theorem, stating that :

$$e_{ss} = \lim_{t \to \infty} e(t) = \lim_{s \to 0} s.E(s) \qquad (5.68)$$

Therefore, K_{vff} should be set equal to 1/k.

Since $V_{ref}(k)$ is calculated at every sample interval, the actual value of K'_{vff} used in equation (5.63) is given by :

$$K'_{vff} = \frac{1}{k * T_s} \qquad (5.69)$$

However $T_s = 4$ ms

$$K'_{vff} = \frac{1000}{2.72 * 4} = 92$$

Determination of K_{aff}

Velocity error is not sufficient to ensure accurate path tracking. When the robot joint is accelerating at constant rate to achieve certain velocity, then the positional demand is a parabolic signal $B.t^2/2$ with a Laplace transform of B/s^3. The block diagram of the proportinal controller with velocity and acceleration feed-forward can be seen in Figure (5.12).

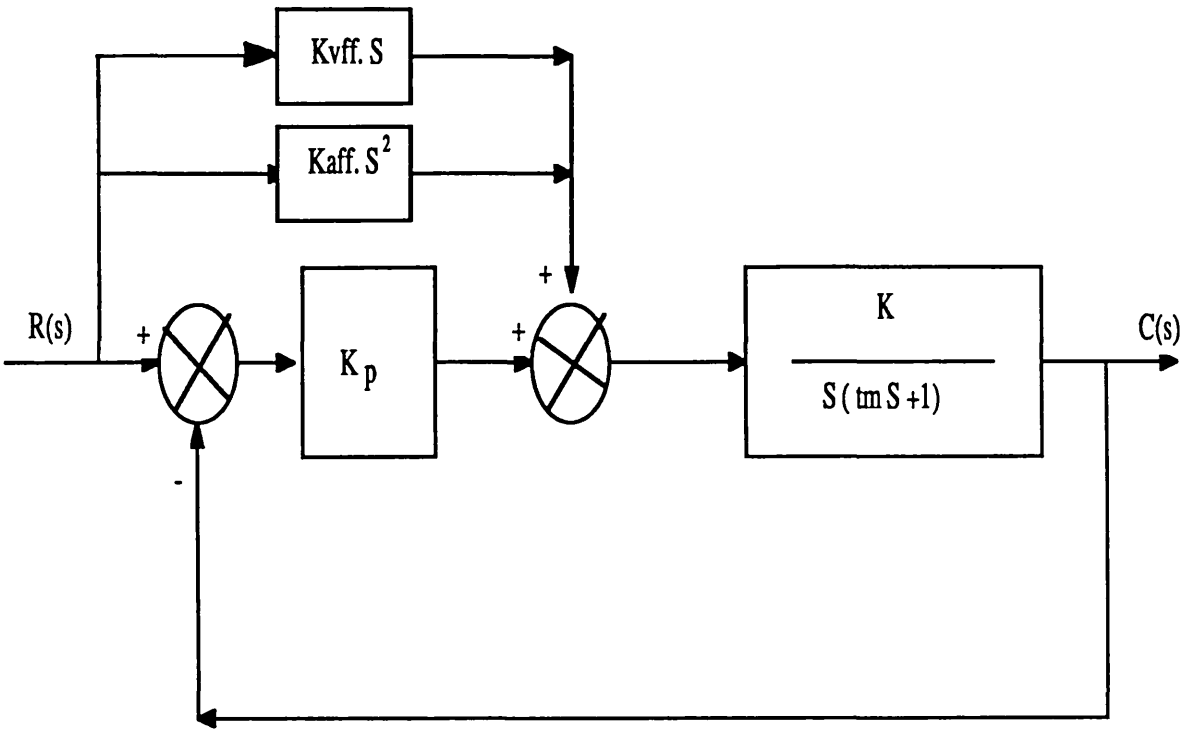


Fig. 5.12 Block Diagram of a Single Joint Actuator Control System Incorporating Velocity and Acceleration Feed-Forward

The positional error is given by :

$$E(s)= R(s) - C(s) \tag{5.70}$$

$$E(s)= R(s).\frac{s(\tau_m s +1) - k K_{vff}\ s - k K_{aff}\ s^2}{s(\tau_m s +1) + k K_p} \tag{5.71}$$

where $R(s) = B/s^3$ is the Laplace transform of the parabolic signal.

Therefore,

$$e_{ss}= s.\frac{B}{s^3} \cdot \frac{s(\tau_m s +1) - k K_{vff}\ s - k K_{aff}\ s^2}{s(\tau_m s +1) + k K_p} \tag{5.72}$$

It can be deduced from the above expression that if e_{ss} =0 then :

$$K_{aff} = \frac{\tau_m}{\kappa} \tag{5.73}$$

This ensures that the steady-state error due to acceleration input $At^2/2$ is zero, provided that K_{vff} = 1/k. Since A_{ref} (K) is calculated in BLU at every sample interval, the implemented value of K_{aff} used in equation (5.63) is given by :

$$K'_{aff} = \frac{1}{T_s^2} * \frac{\tau_m}{k} \tag{5.74}$$

Thus for a 4 ms sampling period,

$$K'_{aff} = \frac{10^6}{4*4} * \frac{0.023}{2.72} = 528$$

Determination of V_{ref}

V_{ref} can calculated using three different methods:

1- Backward Difference Method:

$$V_{ref1}(k)=X_{ref}(k)-X_{ref}(k-1) \tag{5.75}$$

2- Bilinear Transform Method:

$$V_{ref2}(k)=2[X_{ref}(k)-X_{ref}(k-1)] - [X_{ref}(k-1)-X_{ref}(k-2)] \tag{5.76}$$

This transformation predicts the velocity one step ahead provided the acceleration (deceleration) remains constant until the next sampling interval. It does however produce excessive overshoots at points where the velocity profile changes.

3- A Form Of Position Preview:

$$V_{ref3}(k)=0.5\,[V_{ref2}(k)+0.5\,(V_{ref1}(k+1) + V_{ref1}(k+2))] \tag{5.77}$$

By averaging the present velocity calculated using method 2 with the average of the next

two sampled velocities calculated using method 1, a better prediction is obtained particularly at points where the velocity profile changes.

Determination of A_{ref}

A_{ref} is determined using the backward difference method:

$$A_{ref}(k)=[X_{ref}(k)-X_{ref}(k-1)] - [X_{ref}(k-1)-X_{ref}(k-2)] \quad (5.78)$$

The following tests were carried out on joint 1 using the reference input described in section 5.6. Demand, output and error values were then plotted.

Test 1: Proportional Controller

$K_p=1$, $K_{vff}=0$, $K_{aff}=0$, $T_s=4$ ms

selecting

$$U_d(k)=K_p [X_{ref}(k)-X(k)]$$

resulted in very large following errors as expected. This can be seen in Figure (5.13).

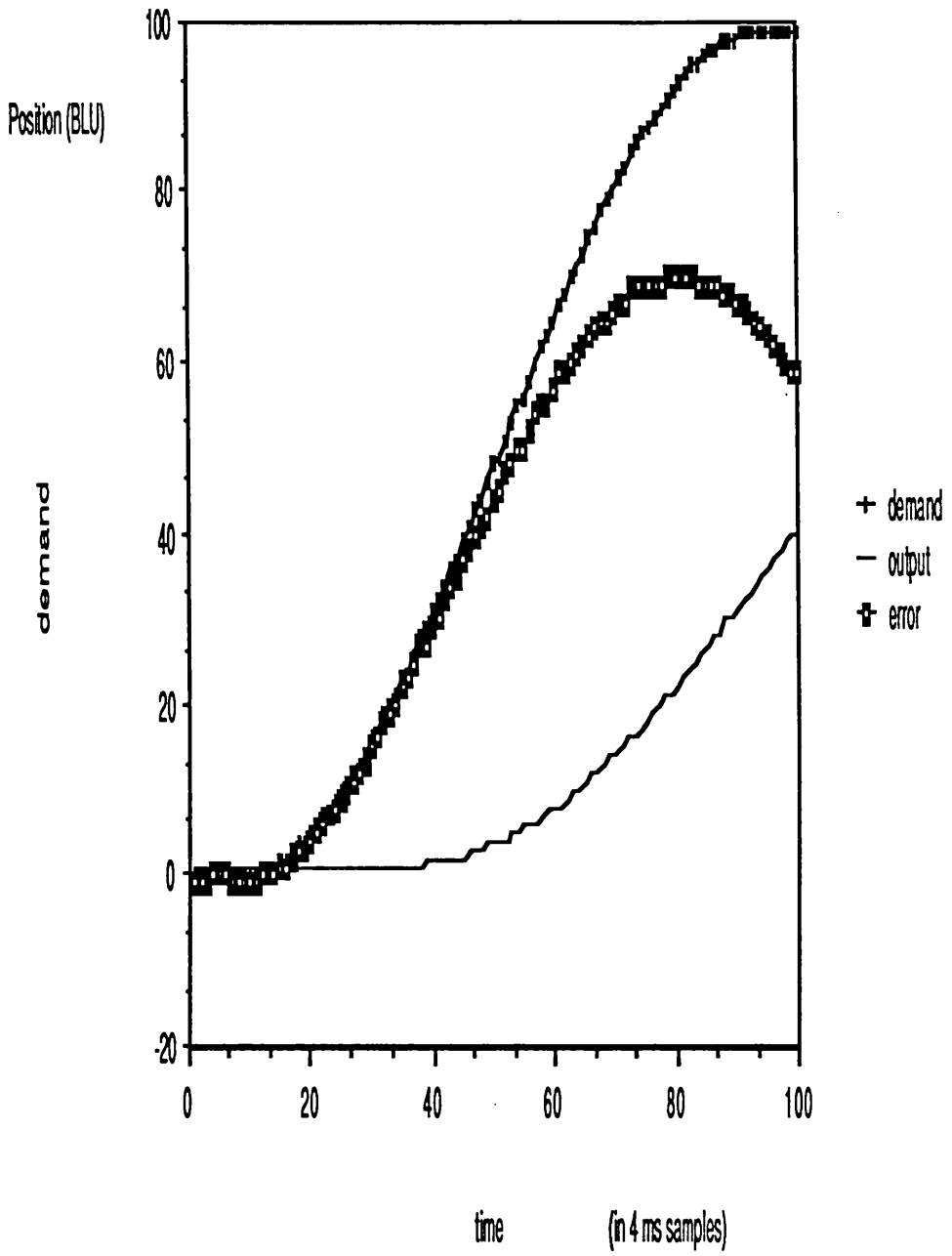


Fig. 5.13 Profile Response for Joint 1 with Proportional Controller $K_p=1$

Test 2: Critically Damped System

$K_p=4$, $K'_{vff}=0$, $K'_{aff}=0$, $T_s=4$ ms

selecting

$$U_d(k)=4*[X_{ref}(k)-X(k)]$$

reduced the error as expected. This can be seen in Figure (5.14). Note that for the system under consideration choosing $K_p=4$ leads to a damping factor of 1.0 for the equivalent continuous system.

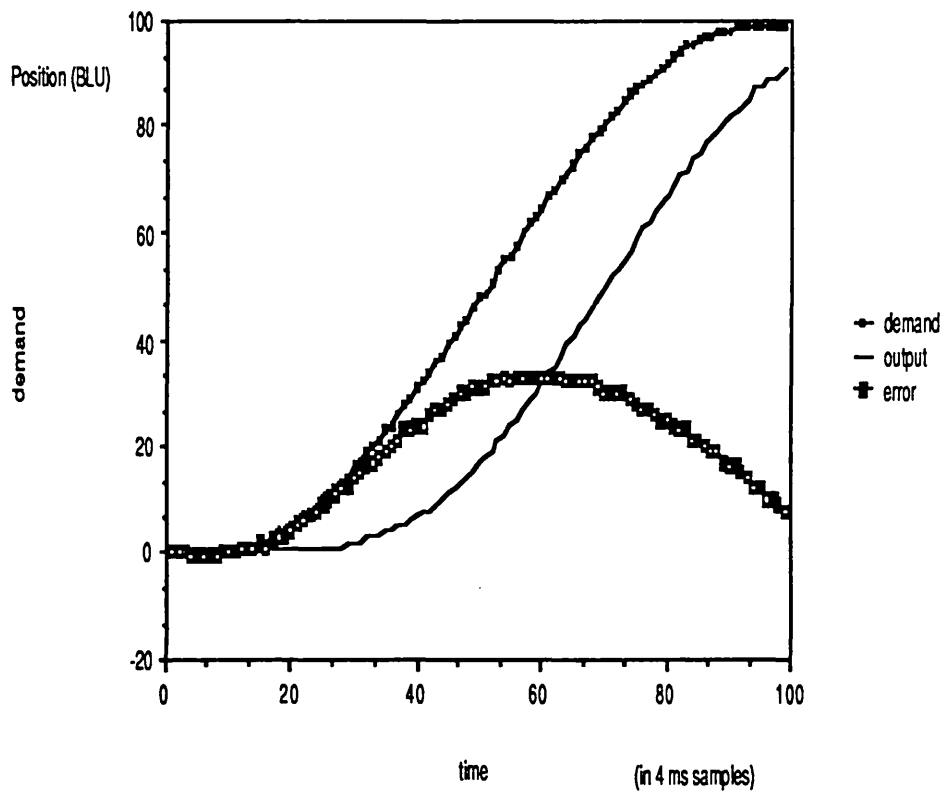


Fig. 5.14 Profile Response for Joint 1 with Proportional Controller $K_p=4$

5.7.3 THE EFFECT OF VELOCITY FEEDBACK

The first consideration of the design procedure is to provide analytically some means of measuring the performance of the tracking controller. The natural choices are the damping ratio, ζ and the natural frequency, ω_n which conventionally are used to determine the transient response of the closed loop system. The analysis must be performed in terms of a sampled data system as the controller is implemented by computer.

For a continuous time system it is generally known that the effect of a velocity feedbackk term is to decrease the time constant and the steady state gain. The latter effect can then be compensated for by increasing the proportional gain of the system. The exact relationships can easily be determined in such systems but with sampled data systems the relationships need to be considered further. The aim of this section is to determine these relationships and to use them to calculate the effect of the feedback when applied to the tracking controller.

The damping factor in a sampled-data system is given by (Koren):

$$\zeta = \frac{\alpha T_s}{[(\alpha T_s)^2 + (\omega T_s)^2]^{1/2}} \tag{5.79}$$

where

$$\alpha T_s = -\ln(a)$$

$$a^2 = X + K[\tau_m((1-X) - T_s X]$$

$$X = e^{-Ts/\tau m}$$

$$\omega T_s = \arccos\{[1+X-KT_s + K\tau_m(1-X)]/2a\}$$

Let $K.K_p = 348\ s^{-1}$, sampling period $T_s = 4$ ms and the mechanical time constant τ_m = 23 ms, then from equation (5.79) ζ is found to be 0.0518.

The very high proportional gain of the system results in a very fast response time but has

the adverse effect of making the system unstable. By introducing velocity feedback into the system the damping ratio can be increased and thus stability is maintained. The second advantage of a high proportional gain is that the static servo stiffness of actuator system increases. This is of particular significance in direct-drive robots, since any torque disturbance due to dynamic interactions among the links of the mechanism can be compensated for. Referring to Figure (5.15), the transfer function of the system is obtained as:

$$\frac{C(s)}{R(s)} = \frac{k.K_p}{\tau_m s^2 + (1+k K_{vb}) s + k K_p} \tag{5.80}$$

The aim of velocity feedback is to increase the damping ratio of the control system. In this case the velocity feedback gain is calculated so as to achieve critical damping (ζ=1). Thus, if K_p=128, then the value of K_{vb} = 1.71 for the equivalent continuous system. The implemented gain on the control computer for the velocity feedback is however,

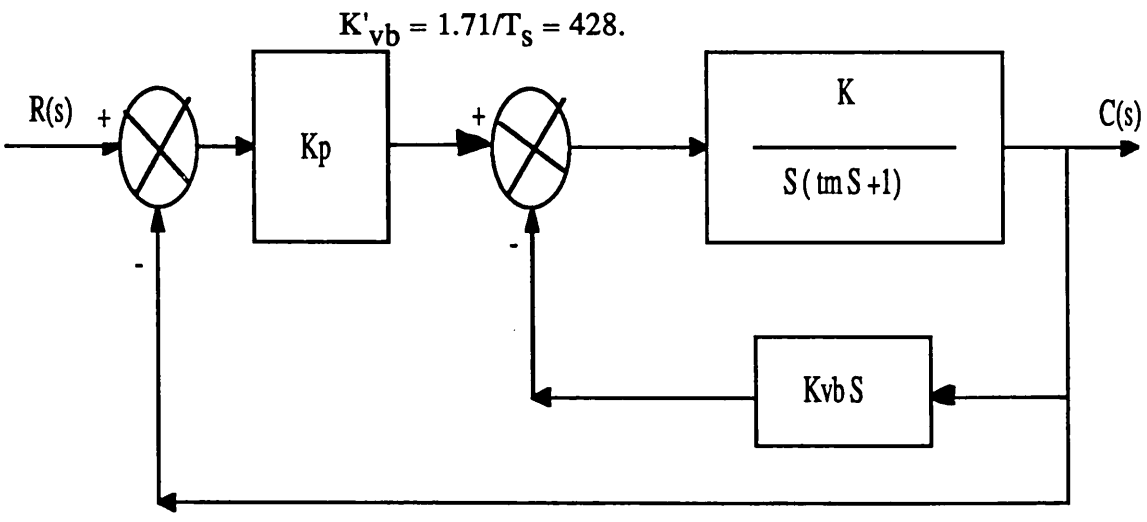


Fig. 5.15 Block Diagram of the Positional Controller with Velocity Feedback

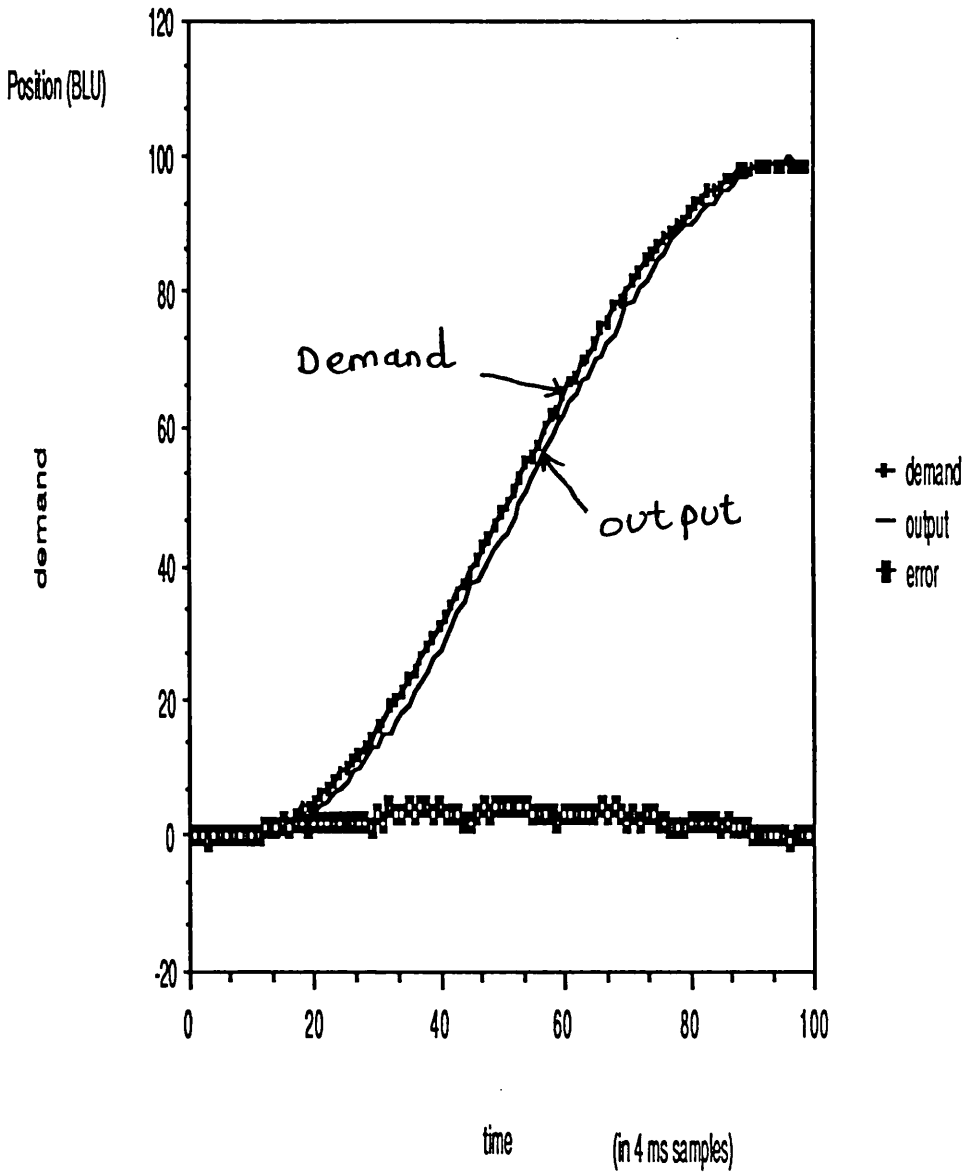


Fig. 5.16 Profile Response for Joint 1 with Proportional Controller K_p=128 and velocity feedback

5.7.4 THE CONTROLLER IMPLEMENTED ON I.C. DD ARM

Figure (5.16) shows how closely the output follows the reference input. However, the output profile lags behind the reference input. This is due to velocity and acceleration errors. The inclusion of the velocity and acceleration feed-forward terms, Figure (5.17), will eliminate such errors as can be seen in Figure (5.18).

For motor 1 :

K_p=128 , K'_{vff}=92 , K'_{aff}=528 , K'_{vb} = 428 , T_s=4 ms

For motor 2 :

K_p=128 , K'_{vff}=92 , K'_{aff}=184 , K'_{vb} = 280 , T_s=4 ms

selecting :

$$U_d(k) = K_p * [X_{ref}(k)-X(k)] + K'_{vff} * [X_{ref}(k)-X_{ref}(k-1)] + K'_{aff} *\{[X_{ref}(k)-X_{ref}(k-1)] - [X_{ref}(k-1)-X_{ref}(k-2)]\} - K'_{vb} *[X(k)-X(k-1)]$$

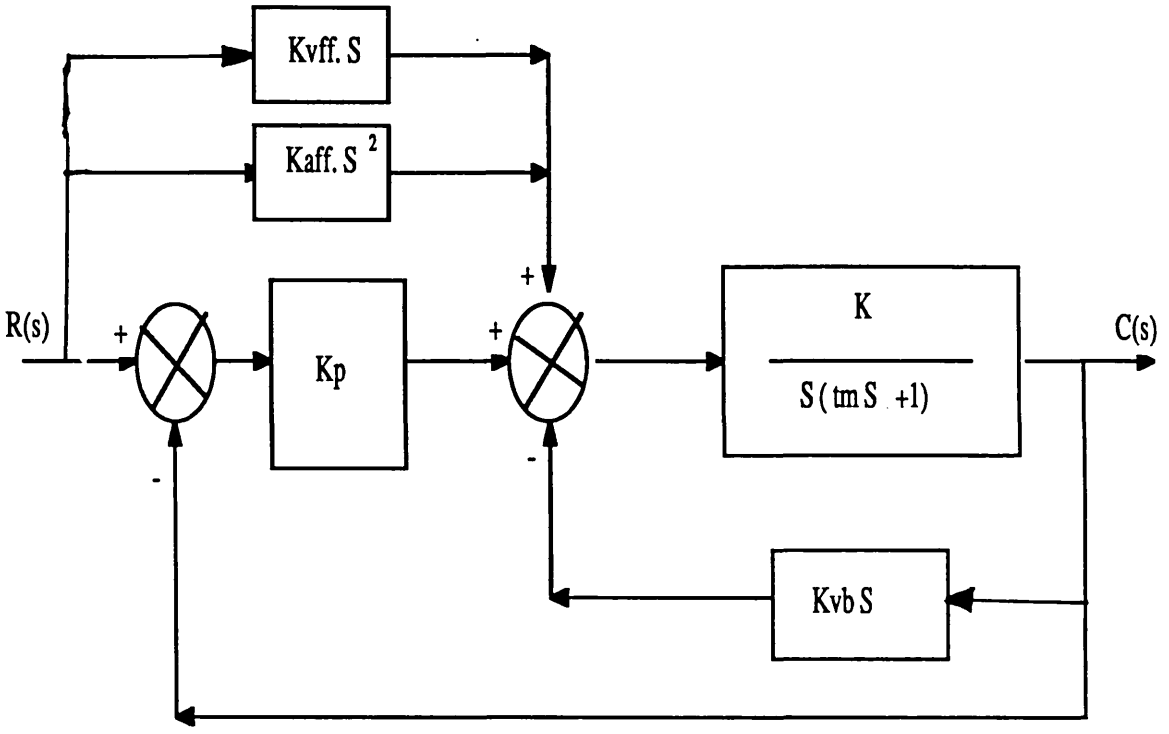


Fig. 5.17 Block Diagram of the Tracking Controller with Proportional and Velocity Feedback, velocity and acceleration feed-forward

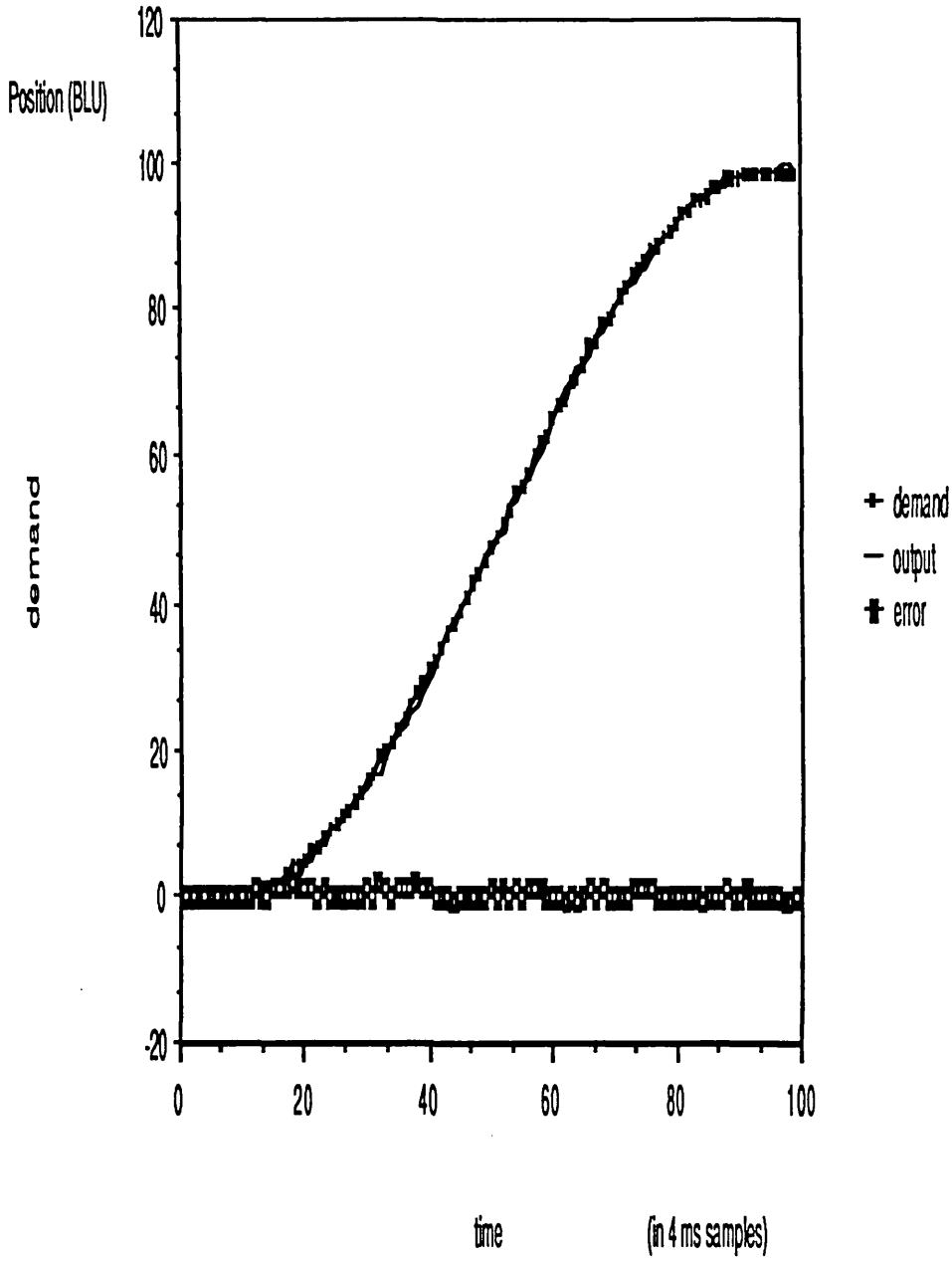


Fig. 5.19 Profile Response for Joint 1 with Proportional Controller, velocity feedback, velocity and acceleartion feed-forward terms

5.8 PERFORMANCE INDEX

An increased amount of emphasis on the mathematical formulation and measurement of control system performance can be found in the recent literature on automatic control. A performance index is a quantitative measure of the performance of a system and is chosen so that emphasis is given to the important system specifications. In designing a tracking controller for the direct-drive arm, one performance criterion proposed is the minimization of integral of the absolute-value of the error, which is defined as :

$$IAE = \int_0^T | e(t) | \ dt. \tag{5.81}$$

This criterion takes onto account the two contrary design requirements of servo systems, namely, small overshoots on the one hand and minimum steady-state position error on the other.

The system is considered an optimum control system when the system parameters are adjusted so that the index reaches an extremum value, in this case a minimum value. Consequently, as can be seen from Figure (5.18), the most satisfactory controller is the one described in section 5.7.4.

5.9 PD CONTROLLER

The closed-loop block diagram of the third order system is given in Figure 5.20. Figure 5.21 shows the root-locus of the system. As can be seen from the root-locus plot the pole at zero ensures that the steady-state error of a step input to the system is zero. However, the response of this system is oscillatory. The system will become unstable at high values of $K=k.k_p$ as can be seen from the root-locus plot. $k=2.72\ s^{-1}$, tm=23 ms and te=1 ms.

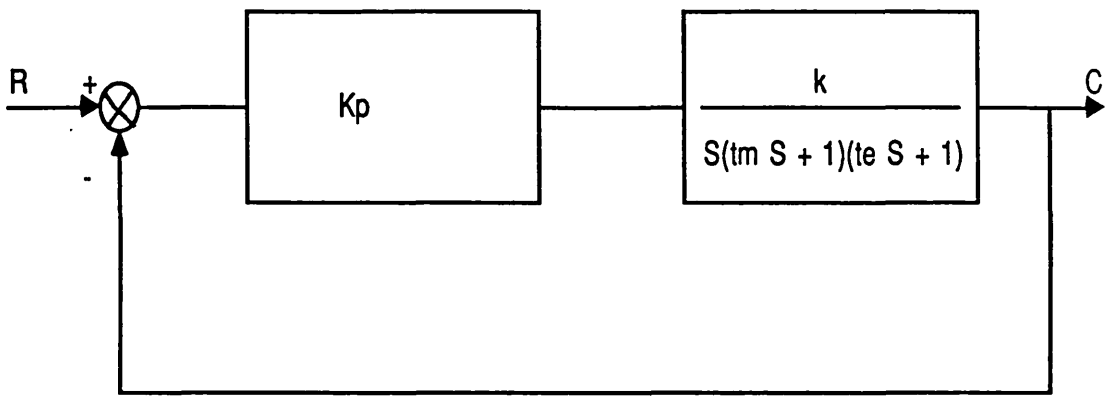


Fig. 5.20 Block Diagram of a Single Joint Actuator System with Proportional Controller

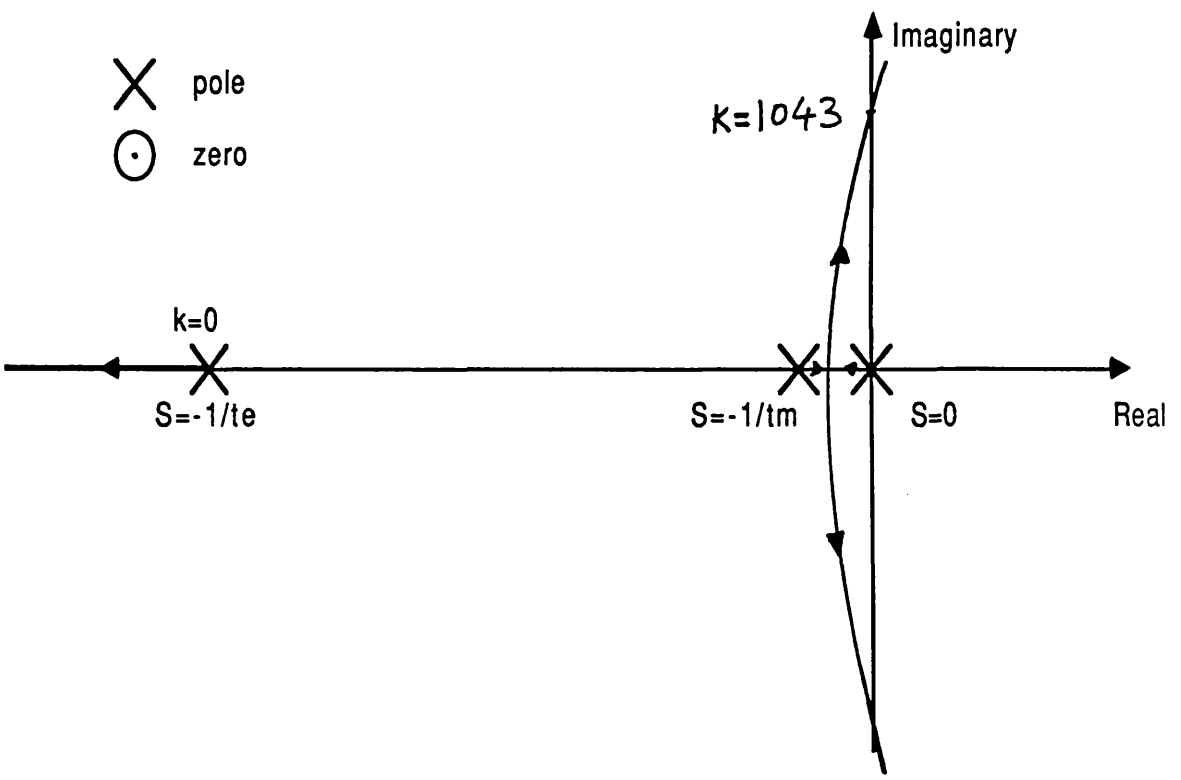


Fig. 5.21 Root-locus plot of the uncompensated system

The objective of the design is to achieve a fast response. The inclusion of a derivative of error term in the feed-forward loop improves the stability and the response of the system. The gains of the proportional and derivative terms are chosen so as to cancel the pole at $S=-1/t_m$. The root-locus plot of the compensated system is given in Figure 5.23. When $k_p=127.2$ and $k_d=2.93$ the pole due to t_m is cancelled and the system is underdamped i.e. $\zeta=0.85$. The respone of this system to a unity step input can be seen in Figure 5.24. Figure 5.25 shows the respose of the system to the modified-cycloid input.

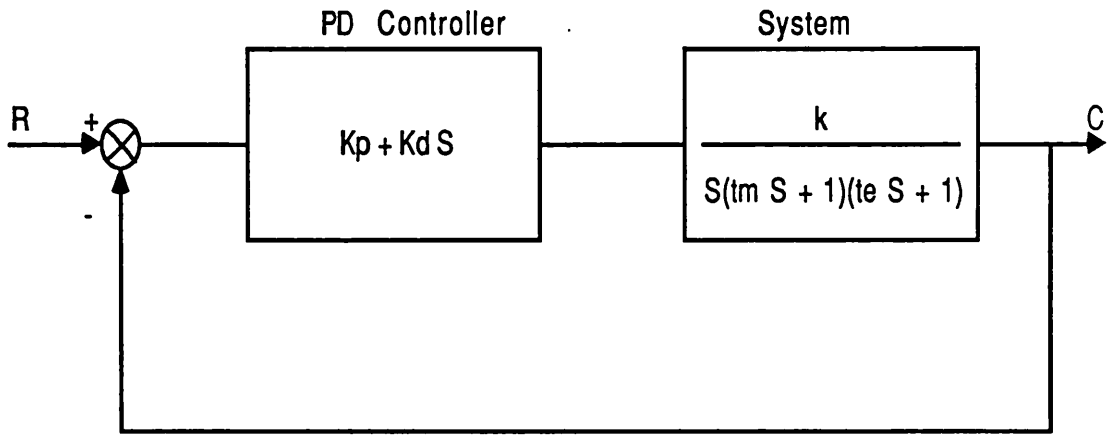


Fig. 5.22 Block Diagram of a Single Joint Actuator System with PD Controller

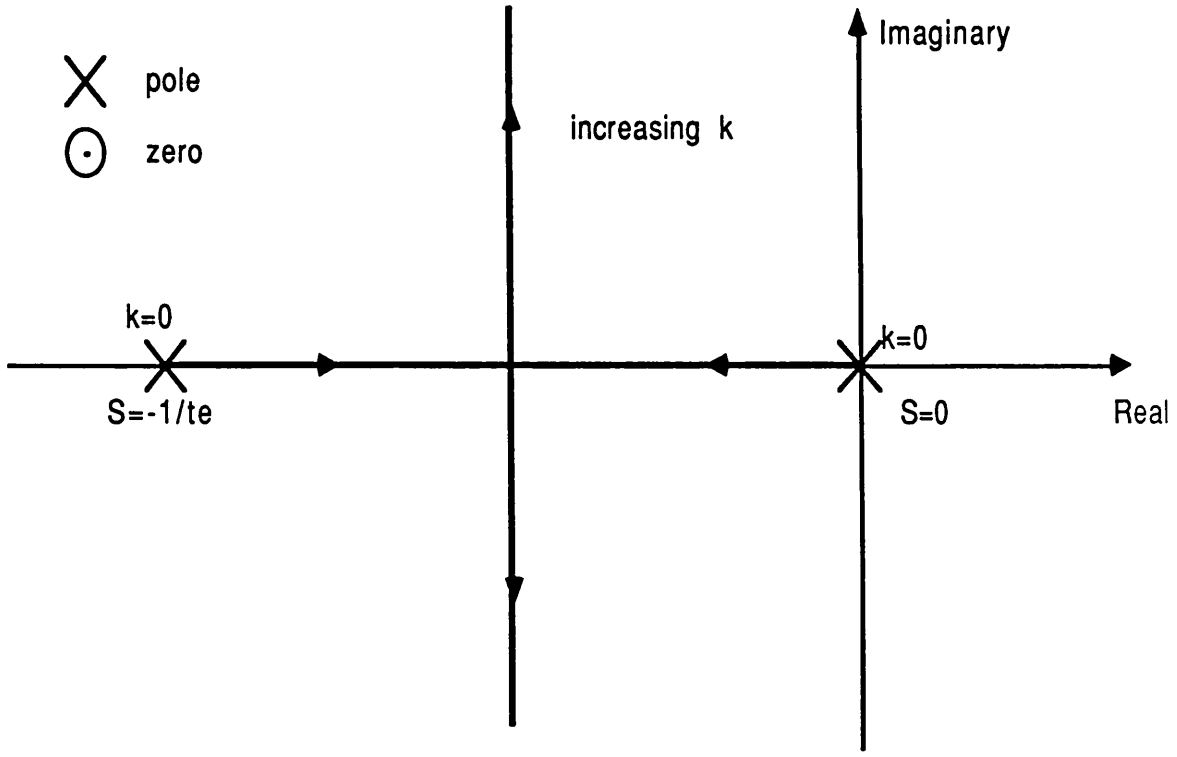


Fig. 5.23 Root-locus plot of the compensated system

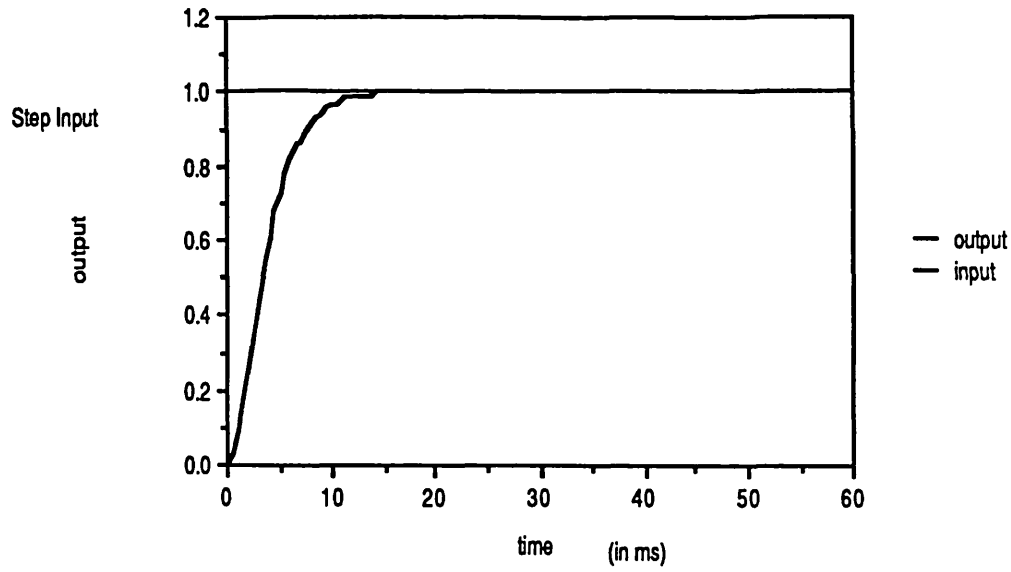


Fig 5.24 The Response of the Compensated system (with PD controller) to a step input of 1.0

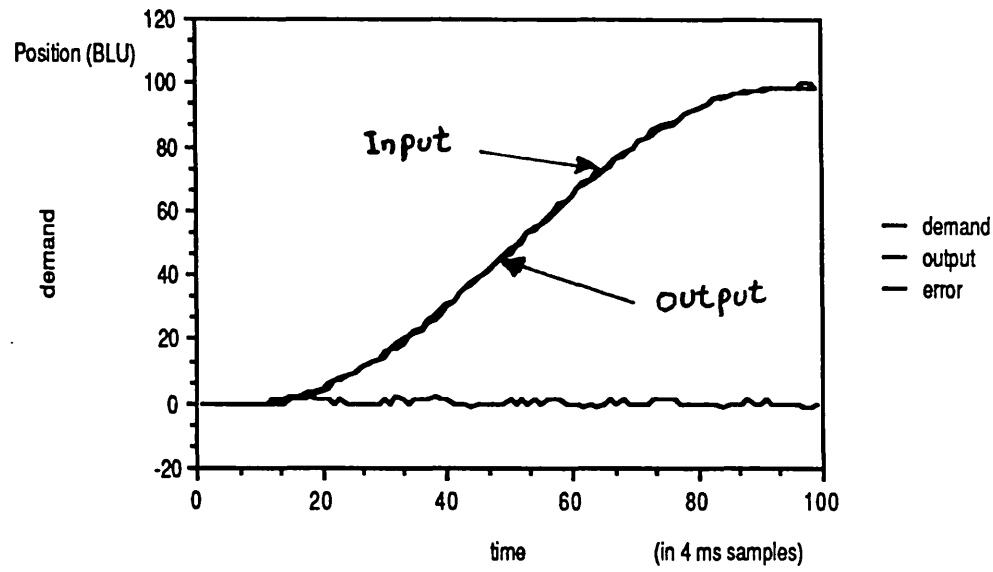


Fig 5.25 The Response of the Compensated system (with PD controller) to the Modified-Cycloid Input

6 FORCE CONTROL

6.1 INTRODUCTION

Industrial robots are computer-controlled mechanical manipulators, which are essentially positioning devices. It has been recognised that in design of teleoperators (i.e. remote manipulators) for speed, accuracy, dexterity, and load capacity, force reflecting capability must be included in the control system. An illustrative example involves fitting a part together with another part in assembly. As the tolerances of the mating parts become tight, it is impractical to require the manipulator and its controller to furnish such high positional accuracy. Alternatively, the controller must combine the positional control and the force control in compliance with the positional constraint imposed by the task geometry.

A number of force control schemes for robotic manipulators has appeared in the literature which require no direct measurement of forces by sensors. Whitney in 1969 developed a resolved-rate-motion scheme. It enabled one to generate the desired joint torques for the manipulator to reach the next desired position which was computed from differentially approximated motions. To generate the desired torques, the Jacobian matrix must be inverted to complete the closed-loop control system, which introduced computational difficulties. Luh, Walker and Paul extended Whitney's idea to include resolved accelerations for smooth transitions to different velocities. The disadvantage of inverting the Jacobian matrix still existed. Inoue's method of 1974 assumed that no information about the load was available

before the manipulator moved. His programming software allowed the user to choose some joints to have force control capabilities. For these joints, he assumed that the external torque was proportional to the settling time of the meas-ured positional error. With this information, the input torque was then adjusted to reduce the error. The disadvantage was that the user had to decide which joints were to be force controlled. Paul in 1972, suggested that the required force for each joint could be precalculated by Lagrangian equations. The computed torque value became the reference to which the actual joint torque had to be controlled. When the manipultor started to move, however, its geometrical configuration and, hence, the actually required torque for each joint changed. Since the precomputed torque requirement was not updated, errors could result. Nevins and Whitney in 1973 suggested the idea of using the force vector assembler. Their discussion was based on a sensing wrist with three extensional strain gauges and three strain gauge shear bridges. In 1976, Paul and Shimano developed a compliance scheme with the assumption that a sensing wrist was available so that the load forces and moments with respect to the axes of the Cartesian coordinates could be measured on line.

6.2 COMPLIANCE

Compliant motion of a manipulator occurs when the manipulator position is constrained by the task geometry. Compliant motion may be produced either by a passive mechanical compliance built into the manipulator, or by an active compliance implemented in the control servo loop. The second method, called force control is based on formal models of the manipulator and the task geometry. Passive compliance offers some performance advantages, but the force control method offers the advantage of programmability. This allows the manipulator to use the particular form of compliance necessary for the particular

application.

In active compliance motion, the manipulator's end effector is capable of moving along a predefined trajectory, while exerting a force at an angle to the path. This feature can be used for deburring of machined components. Hence, the development of the model for a general compliant manipulator can be simplified by considering the two simpler control modes :

I- Pure position/orientation control.

II- Pure force/torque control.

When the manipulator is in free space, pure position/orientation control can be achieved, since there are no objects against which the manipulator can exert force. On the other hand, when the manipulator is rigidly constrained by the environment to move, pure force/torque control can be achieved. The model for a general force control system will be intermediate between these two control systems.

Active force control refers to the use of force sensing elements (usually strain gauges) to sense and provide for correction of force errors. The sensors may be placed at the actuators, at the wrist, at the fingers or in the environment which the manipulator is contacting. However, one of the main advantages of the direct drive robot is the absence of gearing and hence frictional torques. Thus, it is possible to calculate the force exerted by the manipulator on the environment by directly relating the current in the armature to the torque exerted by the servo motor. By knowing the manipulator configuration (i.e. the transposed Jacobian matrix), the exerted force can be related to the measured torques. Therefore, the relationship between the armature current and the torque has to be established. In addition, the mathematical representation of the transposed Jacobian matrix for the parallelogram mechanism has to be deived, before one can discuss the control model and the control law and its implementation.

In the previous chapter, the dynamic model of the actuator system was derived. The pure position control law was also discussed in detail and the experimental results were presented. The implemented control law, section 5.7.5, enabled the position output to follow the demanded input by a tracking error of less than ± 1 BLU (Basic Length Unit).

This chapter deals first with pure force control and the combined position and force control system. Two practical experiments are then presented to demonstrate the performance of the compliant manipulator. The first experiment is the classical peg-in-hole insertion and the second one deals with the situation where the manipulator moves along a straight line (edge of a table) and exerts a force to the table at right angles to the direction of motion.

6.2.1 STATIC SERVO STIFFNESS

Static servo stiffness is the rate at which forces and torques on the arm increase as it is deflected from a nominal position. The basic stiffness formulation follows from a generalisation of the linear spring relationship,

$$f = K_{ii} .dx \tag{6.1}$$

To an n-dimensional matrix expression :

$$F = K.dX \tag{6.2}$$

where dX is a generalised displacement from a nominally commanded position, X_o, of the arm.

Using the differential transform or the so called Jacobian matrix, J, the Cartesian displacement from the joint angle displacement can be determined.Defining,

$$\delta\theta = \theta - \theta_o \tag{6.4}$$

as the difference between the joint angles and the nominally commanded joint angles, thus :

$$\delta X = J\,\delta\theta \tag{6.5}$$

Assuming the static and dynamic forced are compensated for or are small enough to be neglected, the joint torque, T, necessary to apply a force, F, at the end effector can be computed. The transposed Jacobian matrix establishes the following relationship

$$T = J^t F \tag{6.5}$$

Applying the above equation to the parallelogram mechanism shown in Figure (6.1), it can be deduced that :

$$\begin{bmatrix} T_1 \\ T_2 \end{bmatrix} = \begin{bmatrix} l_1 \sin(\theta_1) & -l_1 \cos(\theta_1) \\ -l_4 \sin(\theta_2) & l_4 \cos(\theta_2) \end{bmatrix} \cdot \begin{bmatrix} F_x \\ F_y \end{bmatrix} \tag{6.6}$$

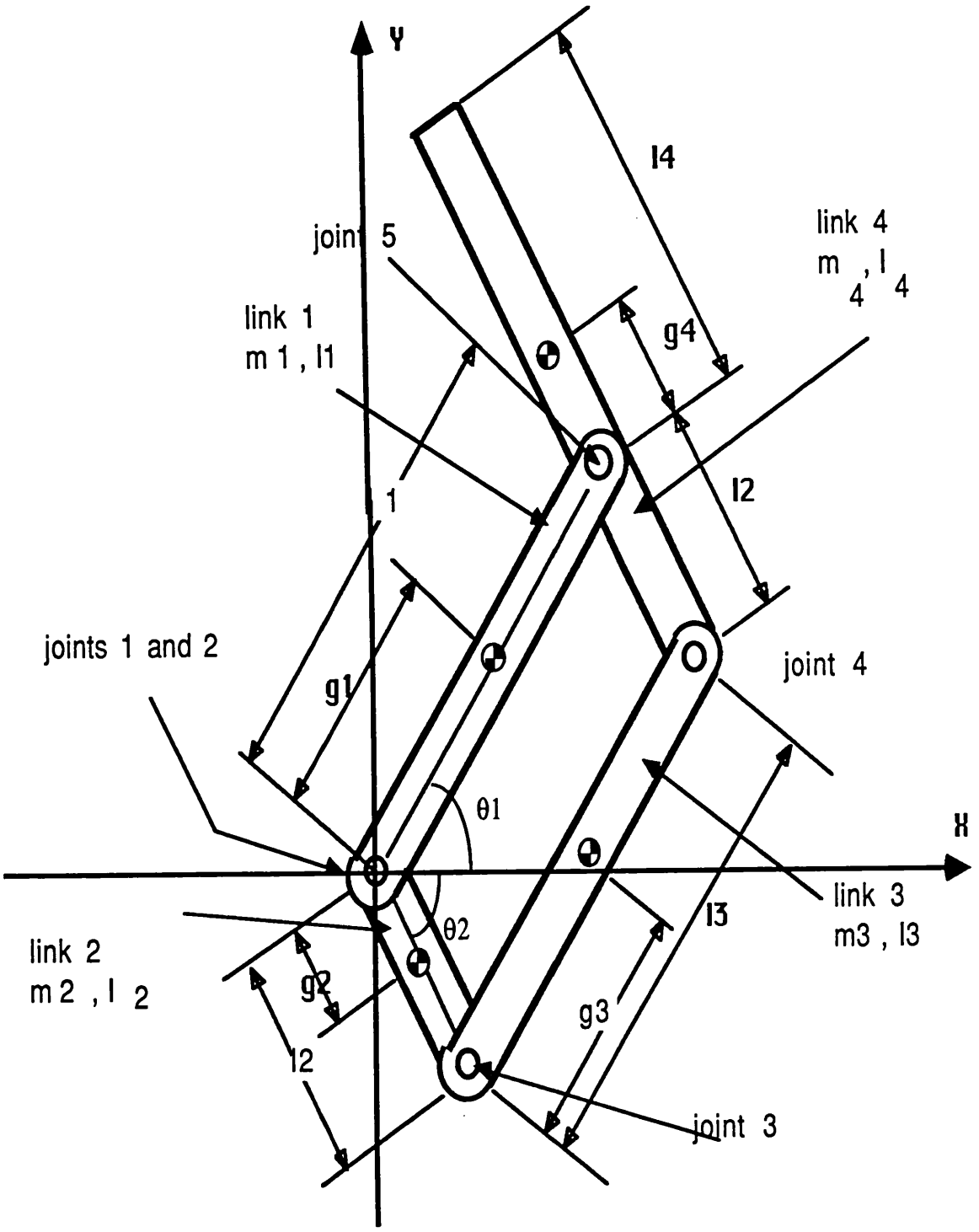


Fig. 6.1 Four Bar Parallelogram Drive Mechanism

Combining equations (6.2) to (6.5), the expression for the joint torques necessary to make the hand behave as a two-dimensional spring in cartesian space can be written as :

$$T = J^{t} K J \, \delta\theta \tag{6.7}$$

$K_\theta = J^{t} K J$ is called the stiffnes matrix of the arm. Thus, by varying the software gain of the controller, and therefore the stiffness matrix, K, the stiffness of the arm can be controlled in software.This formulation is central to the stiffness control method. It has the advantage that it requires only the computation of the jacobian matrix and not its inverse. The Jacobian itself is relatively simple to compute and can be evaluated from the knowledge of intermediate joint frame locations, as can be seen in equation (6.6).

The stiffness control method described here provides an intuitive format for simultaneous motion and force command. The method allows the programmer to think in terms of the stiffness of the arm in a particular direction. For example, if the arm is expected to meet some physical constraints, then the stiffness matrix is adjusted for low stiffness in the direction of the constraint, while at the same time the stiffness in all other directions can be kept high. In this way the contact forces can be controlled (Sahirad et. al.).

6.3 FORCE CONTROL MODEL

The torque applied at the i^{th} joint is given by the express ion :

$$T_i = T_{c,i} + dT_i + K_{v,i} \cdot J_{ii} \cdot d\dot{\theta}_i + V_{0,i} \cdot \mathrm{sgn}(\dot{\theta}_i) + T_g \tag{6.8}$$

where $T_{c,i}$=commanded torque, ith joint, given by

$$T_c=K_\theta.d\theta+T_a \tag{6.9}$$

$$T_a=J^T.F_a \tag{6.10}$$

F_a is the applied controlled force by the arm. dT_i is torque error in the i^{th} joint, given by:

$$dT_i=T_c-T_s \tag{6.11}$$

T_s is the sensed torque, given by:

$$T_s=K_f.d\theta_i \tag{6.12}$$

The last two terms of equation (6.8) are due to friction torque and gravity loading respectively. In the case of the direct drive arm mounted horizontally, both terms can be eliminated from equation (6.8).

There are situations when the arm is out of contact with the environment , the inclusion of the third term on the right hand side of equation (6.8), ensures that there is velocity dependent damping and hence large impact forces are avoided. A block diagram of the stiffness control system is given in Figure (6.2).

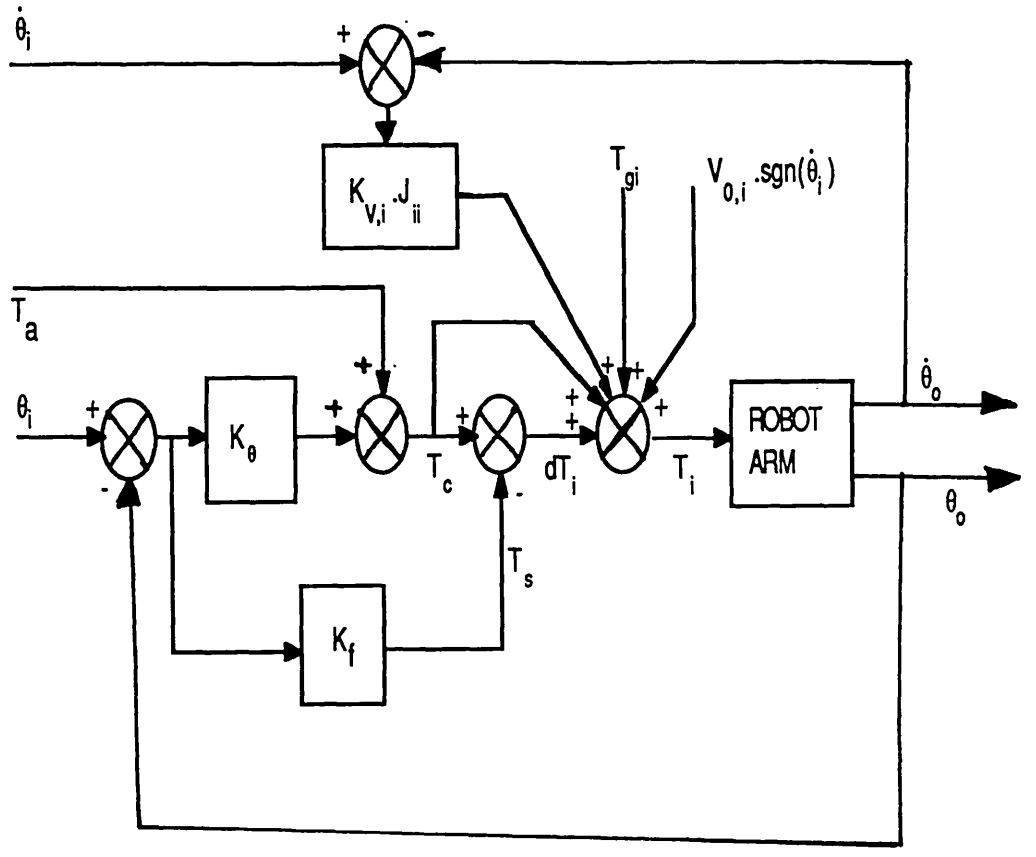


Fig. 6.2 Force Control Block Diagram

6.3.1 DETERMINATION OF THE VALUE OF THE STATIC SERVO STIFFNESS

As was explained previously, the static servo stiffness is given by:

$$K = \frac{\text{Torque, T}}{\text{Deflection from nominal position}} = \frac{T_o}{\delta\theta} \qquad \text{Nm/rad} \qquad (6.13)$$

where $\delta\theta$ is the positional error.

Referring to Figure (6.3), the transfer function of the block diagram governing the relationship between the exerted torque by the actuator and the positional error of the servo

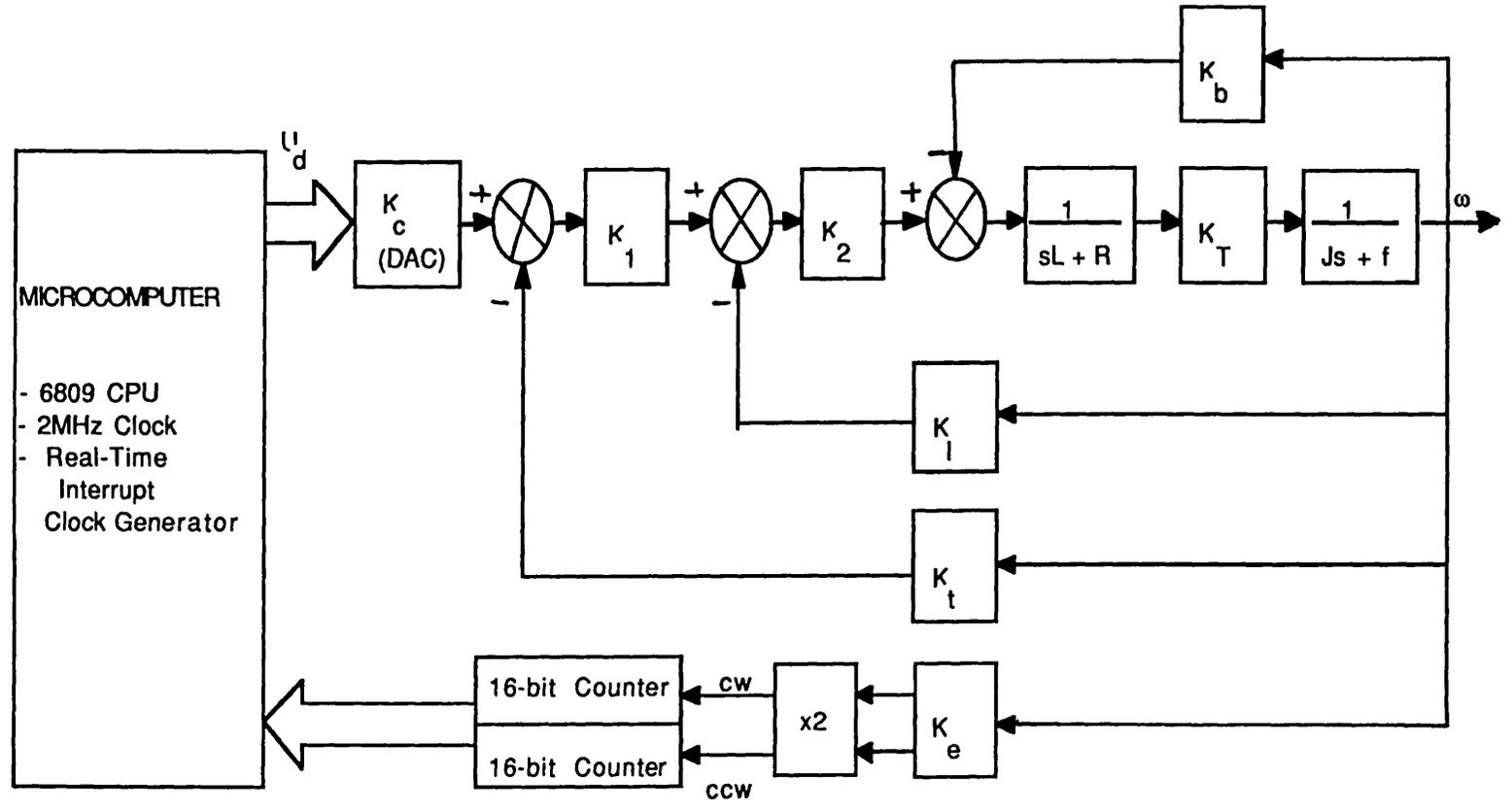


Fig. 6.3 Control System Layout of a Single Joint

K_c DAC Gain (V/BLU)
K_1 : Control Amp. Gain (V/V)
K_2 : Drive Amp. Gain (V/V)
K_T : Motor Torque Constant (Nm/A)
R^T : Motor Arm. Resistence (Ω)
L : Motor Arm. Inductance (H)

J : Motor + Load Inertia (Nm/rad/s^2)
K : Motor e.m.f. Constant (V/rad/s)
K^b Tacho gain (V/rad/s)
K^t Encoder Gain (BLU/rad)
K_e : Current Feedback Gain (V/A)
f_1 Viscous Friction Coefft. (Nm/rad/s)

system (i.e. deflection from a nominal position), in steady-state is given by :

$$\frac{T}{\delta\theta} = \frac{K_c \cdot K_1 . K_T . K_2}{K_2 . K_i + R} \qquad (6.14)$$

In order to find the numerical value of the static servo stiffness of each servo motor, K_c was set to 1 to simplify the calculations. The motor's torque constant, K_T is found from the manufacturer's specifications listed in the appendix. The value of

$$\frac{K_1 . K_2}{K_2 . K_i + R}$$

can be obtained experimentally as follows:

The armature current of motor 1 was monitored while the arm connected to motor 1 was deflected from a nominal position. The deflection was measured by monitoring the digital value fed to the digital to analogue convertor of the corresponding servo motor.

At this juncture, it should be emphasised that the current in the servo motor's armature is directly proportional to the digital number which is fed to the corresponding digital to analogue convertor by the micrprocessor. Consequently, instead of monitoring the current in the armature, which would require extra hardware in the form of analogue to digital convertors, it is adequate to monitor the value fed to the DAC instead.

The graph of armature current vs. deflection of motor 1 is plotted in Figure (6.4). The gradient of the line is 0.65 A/BAU, and K_c=1.

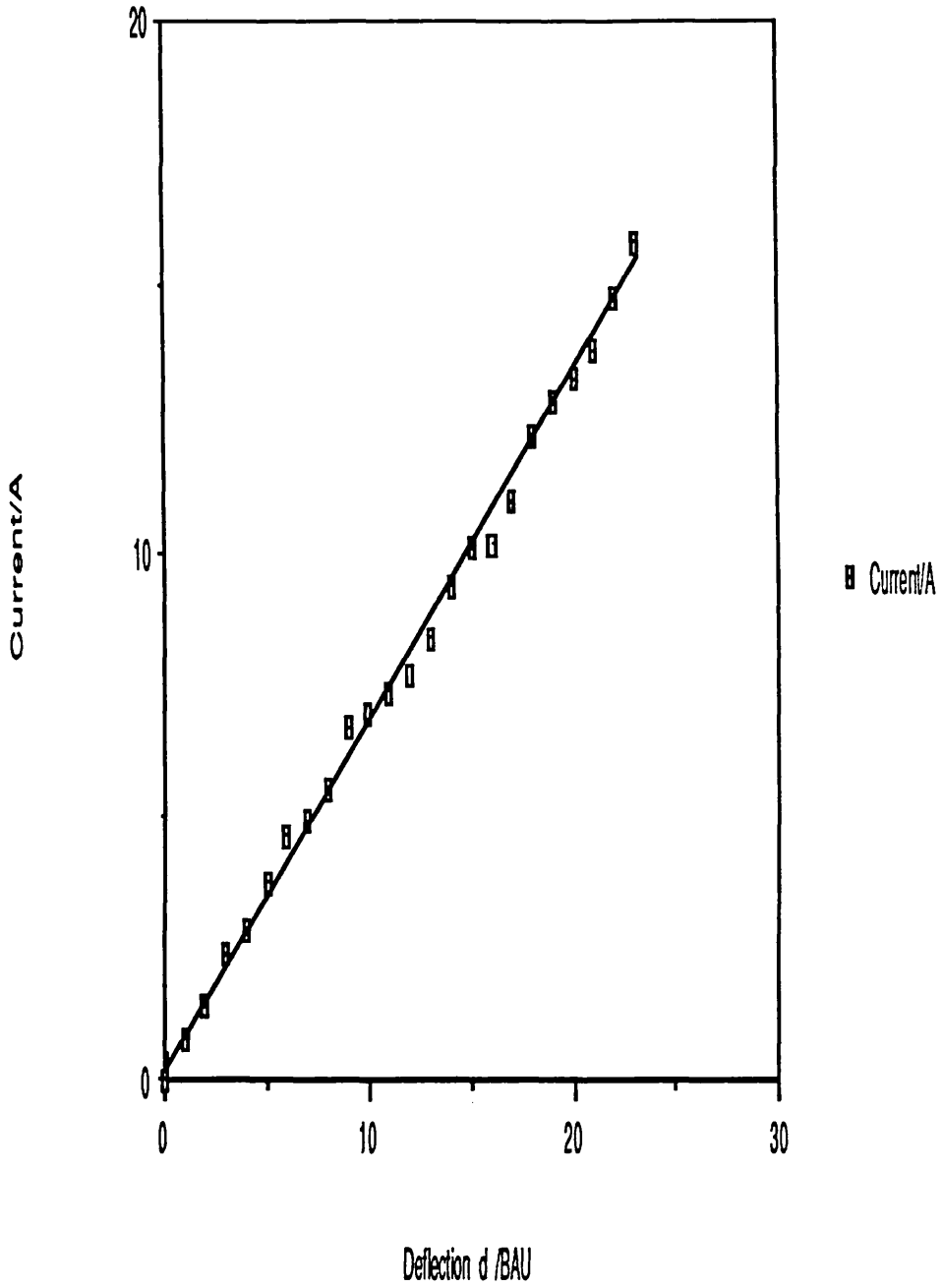


Fig. 6.4 The Graph of Armature Current vs. Deflection from Nominal Position

Therefore, the static servo stiffness of motor 1 can be calculated from equation (6.14) as follows:

K_1 Static servo stiffness = 1* 0.65 * 0.13 = 0.085 Nm/BAU

The same experiments were carried out for motor 2 and the corresponding value of the static servo stiffness was found to be 0.053 Nm/BAU.

6.4 COMPLIANT PEG-IN-HOLE INSERTION

The following program has been developed using PPIL commands to carry out the task of peg-in-hole insertion.

```
5    GOS   500
10   MOV   X     Y     T
20   DWL   T1
30   GOS   700
40   ACD
50   INS
60   DWL   T1
70   GRO
80   DWL   T1
90   ACU
100  DWL   T1
110  GRC
130  STOP
500  KPS   KP1   KP2
510  KVB   KVB1  KVB2
```

```
520  KVF   KVF1  KVF2
530  KAF   KAF1  KAF2
540  RET
700  KPS   1     1
710  KVB   0     0
720  KVF   0     0
730  KAF   0     0
740  RET
```

When the end effector is moving at high speed, the static servo stiffness of each servo motor is set at a high value because of the objective of the positional control law, which is the minimisation of the tracking error. The static servo stiffness of motor 1 in the I.C. direct drive arm is 3463 Nm/rad when the proporional gain of the controller is set to 128. At line 5 in the above program the control system's gains are set for tracking control and the peg which is held in the gripper, is moved above the hole chamfer. At line 30 the static servo stiffness of the manipulator is reduced to 27 Nm/rad for each joint, thus increasing the compliance of the manipulator. At line 40, the pneumatic actuator which moves the peg vertically up or down, pushes the peg downwards on the chamfer. Assuming that there is chamfered contact between the peg and the hole, a lateral force vector is produced, its direction being towards the centre of the hole. At line 50, the INS command transfers the program execution to the INSertion subroutine written in assembly language. The insertion subroutine monitors the torques and consequently the forces exerted on the peg, and guides the end effector and hence the peg towards the centre of the hole, by actuating the X and Y motors. Once the peg is inside the hole, the gripper is opened at line 70 and the pneumatic actuator is moved up via the command ACU (ACtuator Up). Finally, the gripper is closed at

line 110. This method eliminates the use of complicated and expensive force sensors which suffer from electrical noise. In addition since the torques are measured at the motor, there is no need for solving kinematic equations on-line and thus the speed of insertion can be increased. Note that the clearance of the assembly component must be greater than the resolution of the robot system.

6.5 AUTOMATIC DEBURRING

The deburring of machined parts is a major area of concern in improving manufacturing cost efficiency. Deburring costs for some cast parts can be as high as 35 percent of the total cost (Kazerooni). This is a major reason for the development of automated deburring operations. In most cases, burrs must be removed to allow the proper fitting of assembled parts and to ensure safe and proper functioning. On high-tem perature , high-speed rotating parts, deburring is further required in order to reduce turbulent gas flow, maintain dynamic balance and relieve localised stress. For these types of parts, the term precision deburring is used. The final geometry of a deburred edge must remain within a given set of tolerances. Additionally, the surface produced on the edge requires a high quality finish. Typically, manual deburring is the only deburring method available, and represents a time-consuming and expensive solution. This section examines an automated approach to deburring using the I.C. Experimental Direct-Drive Robot Arm.

6.5.1 PRECISION DEBURRING

The geometric model of a burred workpiece edge is shown in Figure (6.5).

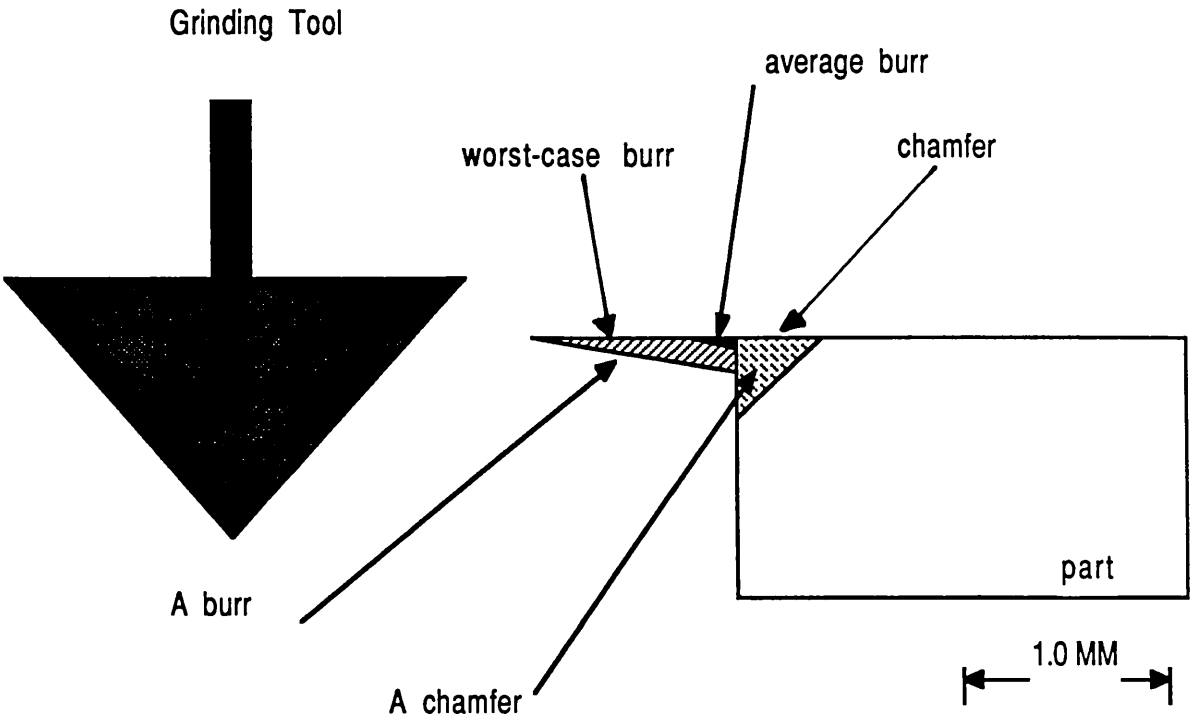


Fig. 6.5 Geometric Model of a Burred Workpiece

The material removal rate (MRR) of a deburring run is a function of the velocity of the tool bit along the edge and the cross sectional areas of both the chamfer and the burr. This relationship can be expressed as:

$$MRR = (A_{chamfer} + A_{burr}) * V_{tool}$$

A three dimensional geometry model of a burr, however, which includes the full geometry of the conic bit is more useful. It can be shown theoretically that the cutting force is largely a function of the average surface area of cut. This resultant cutting force, then, can be resolved with respect to both the part and the end-effector into two vector components of interest:

1- The tangential force (in the direction of the tool velocity).

2- The normal force.

6.5.2 ROBOT POSITION UNCERTAINTIES

While robots can meet the flexibility requirements for a deburring system, the positional accuracy of existing robots is generally poor. There are also some positional uncertainties in fixturing the part. A common solution to this problem has been the addition of a compliant element between the robot and the deburring tool. Considerable work has been done using compliant deburring end-effectors. The device features compliance in two orthogonal directions in the form of replacable springs and fluid dampers. The dynamic behaviour of such passive end-effectors in the normal direction to the part can be approximated by a second order dynamic equation.

However, in sections 6.2.1 and 6,3, it has been shown that the I.C. Experimental Robot Arm can be used as an active compliance device, which has the advantage of programmability and thus is more flexible than a passive device.

6.5.3 CONTROL STRATEGY FOR DEBURRING

There are two questions which may be raised:

1- What compliancy is needed in the normal direction and in the tangential direction in the deburring process?

2- Does the prescribed high compliancy for compensation of the robot position uncertainties conflict with the required compliancy for the deburring process?

Consider the deburring of a surface by a robot manipulator; the objective is to use an

end-effector to smooth the surface down to the comm anded trajectory shown in Figure (6.6).

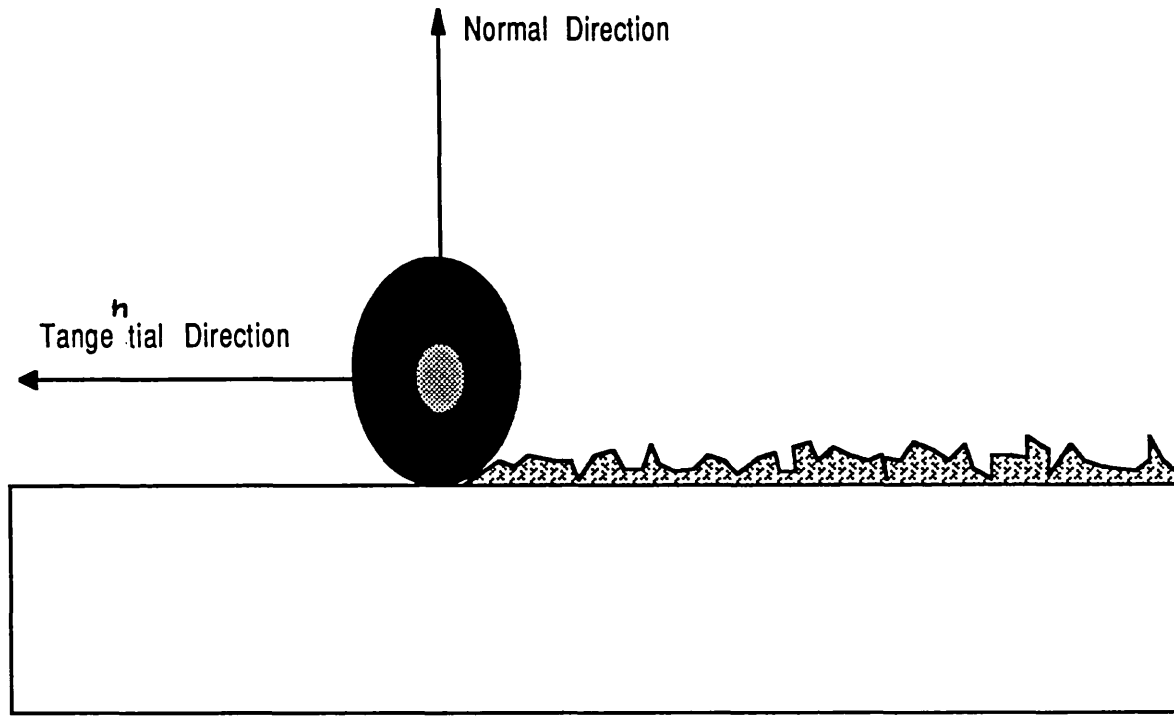


Fig. 6.6 Deburring an Edge

It is intuitive to set the gain of the motors so that the manipulator has a large impedance (small compliance) in the normal direction and a small impedance (large compliance) in the tangential direction. A large impedance in the normal direction causes the the end-point of the grinder to reject the interaction forces and stay very close to the commanded trajectory. The larger the impedance of the end-effector in the normal direction, the smoother the surface will be. Given the volume of metal to be removed, the desired tolerance in the normal direction prescribes an approximate value for impedance in the normal direction. The force necessary

to cut in the tangential direction at a constant traverse speed is approximately proportional to the volume of the metal removed. Therefore, the larger the burrs on the surface, the slower the manipulator must move in the tangential direction to maintain a relatively constant tangential force. This is necessary because a slower speed of the end-point along the surface implies a smaller volume of metal to be removed per unit of time, and consequently less force in the tangential direction. To remove the metal from the surface, the grinder should slow down in response to contact forces with large burrs.

The above explanation demonstrates that it is necessary for the end-effector to accommodate the interaction forces along the tangential direction, which directly implies a small impedance value in the tangential direction. If a designer does not accommodate the interaction forces by specifying a small stiffness value in the tangential direction, the large burrs on the surface will produce large contact forces in the tangential direction.

Two problems are associated with large contact forces in the tangential direction :

1- The cutting tool may stall (if it does not break).

2- A slight motion may develop in the end-point motion in the normal direction, which might exceed t h^edesired tolerances. A small value for the impedance in the tangential direction (relative to the impedance in the normal direction) guarantees the desired contact force in the tangential direction.

The following command moves the end-effector along a straight line defined by the two end points in Cartesian coordinates as (X_1, Y_1) and (X_n, Y_n). The intermediate points along the line are interpolated on-line. The corresponding joint coordinates θ_1 and θ_2 are then computed on-line according the inverse kinematic solutions derived in chapter 2 for the parallelogram arm mechanism. The values of the trigonometrical functions are stored in a look-up table in order to considerably reduce the computational time.

MVL X_1 Y_1 X_n Y_n ALPHA I F_x F_y

where I is the distance between the interpolated points on the line and ALPHA is the time taken to servo the manipulator between two consecutive interpolated points. If the straight line defined by the two points is, say edge of a table, then the end effector can exert a force F, where F_x and F_y are the components of the exerted force along the X and Y axes in Cartesian coordinates given in equation (6.6).

6.6 POWER DISSIPATION

As was explained earlier, another problem with the direct-drive is that the robot cannot exert a large force. Because of the direct coupling of the motors to their loads, the output torques of the motors are not amplified through reducers. In consequence, the motors have to bear the loads directly, and might overheat. In this section the power dissipated in the motors when a force is exerted at the tip of the arm is analysed. The four bar link parallel drive mechanism is compared with the serial drive mechanism, and another significant advantage of the parallel drive over serial drive regarding power dissipation is demonstrated.

An important specification of a torque motor is the motor constant, that is the ratio of output torque T to the square root of power dissipated in the motor P.

$$K_m = T/\sqrt{P} \tag{6.15}$$

The motor constant represents how efficiently the motor converts its input power into output torque. If K_m is small, the motor dissipates a large amount of power to exert torque.

When a robot bears a load at the arm tip, or is exerting a force F as in section 6.5, then the load is distributed to the multiple motors connected to the arm mechanism. In addition to

the individual motor characteristics, the efficiency of exerting a force at the arm tip also depends on the kinematic structure of the arm mechanism. Let F be the end point force exerted at the arm tip represented in Cartesian coordinates, and let T be the vector of equivalent torques of n motors, then one obtains

$$T=J^t F \tag{6.16}$$

where J is the Jacobian matrix associated with the coordinate transformation from joint coordinates to the Cartesian coordinates of the arm tip.

Let P be the total power dissipated in the n motors driving the arm mechanism and K_{mi} be the motor constant of the ith motor, then the total power dissipation P is given by the following quadratic form of the end point force F.

$$P=\sum_{i=1}^{n}\frac{T_i^2}{K_{mi}^2} = F^t L F \tag{6.17}$$

where L is given by

$$L=J \text{ diag } \left(\frac{1}{K^2_{mi}}, \ldots\ldots\ldots, \frac{1}{K^2_{mn}}\right) J^t$$

The matrix L is positive definite as long as the Jacobian matrix is nonsingular. Therefore, the matrix L has all real positive eigenvalues, $\lambda_i > 0$, and the quadratic form can be rewritten as:

$$P=\sum_{i=1}^{n} \lambda_i F_i^2 \tag{6.18}$$

where F_i is the component of the end point force along the eigenvector corresponding to λ_i . If the eigenvalue λ_i is large, a large amount of power is dissipated to exert force F_i in the direction of the corresponding eigenvector. for smaller eigenvalues, the arm can exert an end

point force more efficiently. To evaluate the overall efficiency of the arm-motor combination, the mean power dissipation ratio is defined, which is the mean of the eigenvalues and is given by :

$$\lambda_m = \text{trace}(L)/n \tag{6.19}$$

The units of λ_m are watts per square Newton.

For the four bar link parallel drive mechanism, matrix L is given by :

$$L = \begin{bmatrix} A.\sin^2(\theta_1)+B.\sin^2(\theta_2) & -A.\sin(\theta_1).\cos(\theta_1) - B.\sin(\theta_2).\cos(\theta_2) \\ -A.\sin(\theta_1).\cos(\theta_1) - B.\sin(\theta_2).\cos(\theta_2) & A.\cos^2(\theta_1)+B.\cos^2(\theta_2) \end{bmatrix} \tag{6.20}$$

where $A = \dfrac{l_1^2}{K^2_{m1}}$ and $B = \dfrac{l_4^2}{K^2_{m2}}$

The end point force F is represented in the coordinate system fixed at the base as shown in Figure (6.1). The mean power dissipation ratio is then given by :

$$\lambda_m = \frac{1}{2}\left(\frac{l_1^2}{K^2_{m1}} + \frac{l_4^2}{K^2_{m2}}\right) \tag{6.21}$$

Although matrix L varies depending on the arm configuration, the mean power dissipation λ_m is uniform for all the configurations. For shorter link lengths, l_1 and l_4, and larger motor constants, K_{m1} and K_{m2}, the arm-motor combination has a better efficiency having a small mean power dissipation ratio.

However, the mean power dissipation ratio of the serial drive mechanism is given by :

$$\lambda_m = \frac{1}{2}\left(\frac{l_1^2}{K^2_{m1}} + \frac{l_2^2}{K^2_{m2}} + \frac{l_2^2}{K^2_{m1}} \right) - \frac{l_1 l_2}{K^2_{m1}} . \cos(\theta_2) \qquad (6.22)$$

In this case, λ_m varies depending on the arm configuration. As θ_2 increases, λ_m becomes larger, namely, the efficiency becomes lower. Comparing equation (6.21) with equation (6.22), one finds that parallel drive has a smaller mean power dissipation ratio than the serial drive except when

$$| \theta_2 | \leq \cos^{-1} [l_4 / (2l_1)] \qquad (6.23)$$

The mean power curves are shown in Figure (6.7). In the case of I.C. direct-drive arm;

$$l_1 = l_4 = 0.20 \text{ m}$$

From equation (6.15)

$K_m = T/\sqrt{P}$

$\therefore$ $K_{m1} = 1.24/\sqrt{433} = 0.0596 \text{ Nm/W}^{1/2}$

and $K_{m2} = 0.6/\sqrt{179} = 0.0448 \text{ Nm/W}^{1/2}$

Substituting the above values into equations (6.21) and (6.22) gives :

$$\lambda_m \text{ (parallel drive)} = 15.59 \quad \text{W/N}^2$$

$$\lambda_m \text{ (serial drive)} = [\, 21.22 - 11.26 \cos(\theta_2) \,] \quad \text{W/N}^2$$

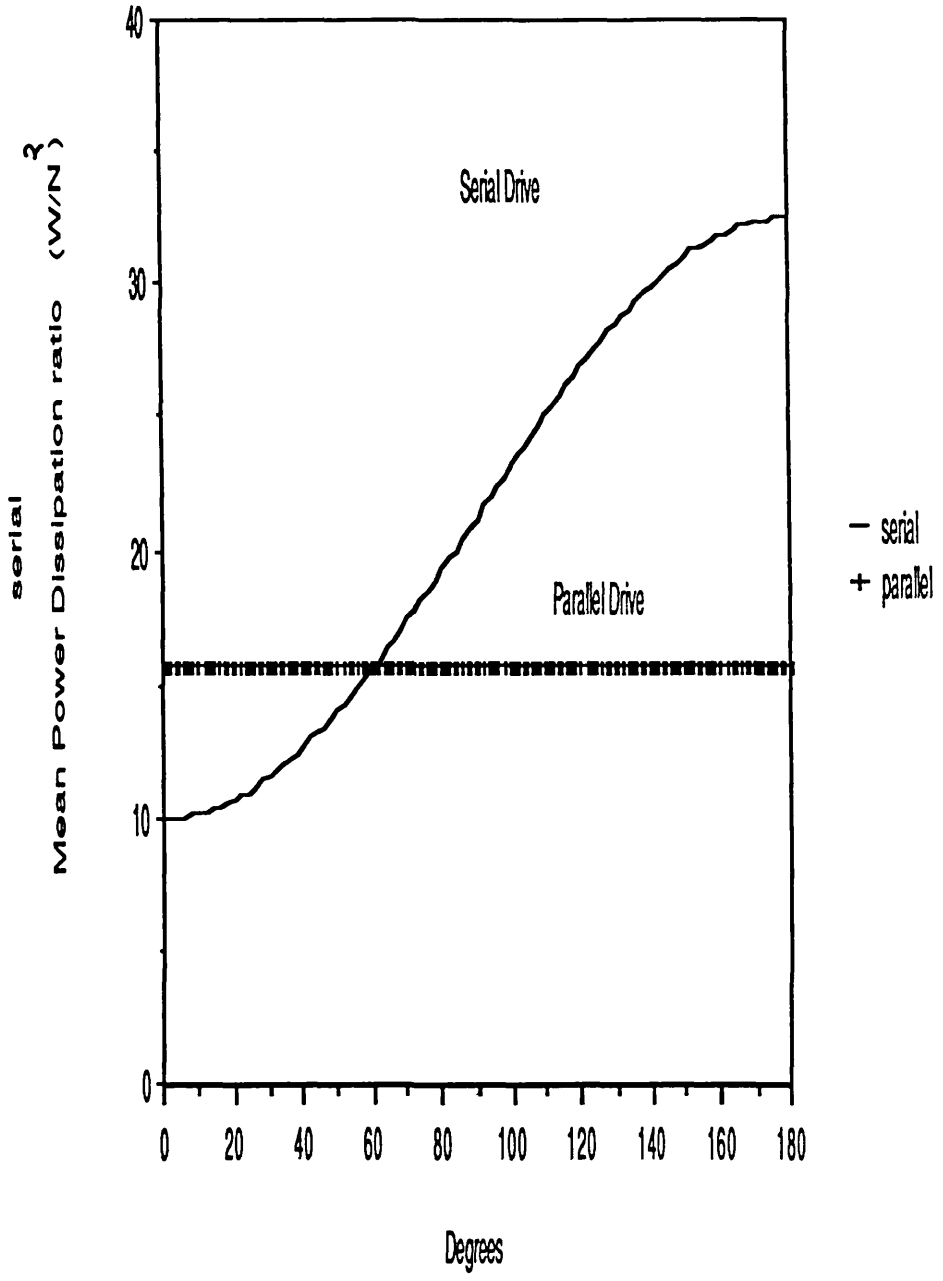


Fig. 6.7 Mean Power Dissipation ratio for the Parallel and Serial Drives

7 CONCLUSIONS AND FURTHER WORK

The final chapter is split into two sections, the first presents the conclusions of the work covered by this study and the second provides suggestions for further work.

7.1 CONCLUSIONS

7.1.1 THE DIRECT-DRIVE ARM

The IC Experimental Robot Arm has been designed and constructed according to the principles of a four bar, parallel drive mechanism enumerated by Asada and Youcef-Toumi. The decoupling of the two arm axes is found to be almost perfect. In addition the arm is mounted horizontally so that gravitational torques are eliminated and the arm dynamics are greatly simplified. In chapter 3, it was shown that the Lagrange and the Newton-Euler methods lead to the same result. As the parallelogram arm can be designed with a mass distribution which results in decoupling the arm, the dynamics of the arm do no need to be computed on-line. Hence, the dynamics needed for the control system design concentrate on the joint actuators rather than the arm itself; the normal situation for conventional robot arms.

Each joint actuator consisting of a motor-tacho-amplifier combination together with a DAC and encoder is modelled. Initially a third order position transfer function is derived but this is simplified by ignoring the electrical time constant. The resulting second order model is used for the design of the controller. In chapter 5, a detailed design procedure for the construction of the conventional positional controller which controls a single arm joint was

presented. Trajectory following errors of this simple controller were analysed and it was shown how velocity feedback and feedforward compensation methods are used to cancel out these errors.

Experimental results from the arm show that the velocity and acceleration values before saturation effects occur considerably lower than those for the M.I.T. direct-drive arm. This is not surprising as the motors used on the IC Experimental Arm only have maximum torques of 6.2 Nm and 3 Nm for joints 1 and 2 respectively, compared with 660 Nm and 230 Nm torque for the M.I.T. direct-drive arm.

The modified cycloid profile implemented as reference input into each actuator system exhibits very smooth velocity and acceleartion characteristics, and is highly recommended for any actuator system where fast point to point motion is required.

In chapter 6, it was shown that the stiffness control approach to force control in manipulators is a powerful tool. It provides effective means of controlling the force and compliance in assembly tasks.

7.2 SUGGESTIONS FOR FURTHER WORK

7.2.1 THIRD ORDER MODEL

In the design of all controllers, the first task is to determine an adequate model of the system dynamics. The dynamics of the actuators for the IC Experimental Arm are dealt with in chapter 5 in some detail. The initial analysis leads to a second order model for the velocity output which included two time constants, the mechanical and the electrical time constants. It was stated that $\tau_m >> \tau_e$ and on this basis the electrical time constant was ignored.. Hence a first order velocity transfer function resulted i.e. a second order position transfer function.

This section re-examines the validity of this assumptions and presents a case for including τ_e.

The basic reason for ignoring τ_e in the previous model is that its magnitude is much smaller than τ_m and that any delay introduced will be insignificant. This is a common assumption to make for DC servo motors and generally is true. In the case of a direct-drive robot however, there is a requirement on the motors to provide large torques at low speeds. Apart from the development of rare-earth magnets for use in such motors, this requirement is met in part by significantly more turns on the motor windings. The effect of this is to increase the inductance proportionally to the square of the number of turns. Hence the inductance in the motors used on the IC Experimental Arm is significantly higher than conventional DC servo motors. From the expression for the electrical time constant given in eq (5.11), it can be seen that τ_e increases in proportion with the inductance. As well as having a higher than normal value for τ_e as explained above, the effect of the velocity feedback in the control law is to decrease the mechanical time constant. With the reductions in the effective time constant of the system it is no longer safe to assume that τ_e is neligible.

As a final note, previous work on direct-drive arms (Asada and Youcef-Toumi 1983) found that the actuator dynamics had two poles at 3.65 Hz and 68 Hz. This corresponds to a mechanical time constant of about 44 ms and an electrical time constant of about 2.3 ms. Hence, it can be seen that to ignore the electrical time constant is to ignore some significant dynamic effects. Although the values will be different from the IC Experimental Arm, the basic principle has been illustrated.

7.2.2 THE DIRECT-DRIVE ARM

The limitations of the IC Experimental Arm have been found to restrict the investigation

of tracking controllers. Hence, it is necessary to improve the design of the arm. The following is the proposal for a completely new arm. The improvements on the new arm fall into two main categories.

Firstly, there is the actual hardware of the direcr-drive arm. It has already been found that the motors used cannot supply sufficient torque at low speeds to match the performance of the M.I.T. direct-drive arm. Hence, more powerful servo motors, particularly at low speeds are a prime requirement of any future, prototype direct-drive arm. Allied to this is the problem of position measurement. Encoders which are normally sufficient in conventional robots where their resolution is multiplied by the gear ratio, are found to be inadequate without gears. The recentl y available laser rotary encoders with 81000 pulses/revolution or resolvers which can have resolutions up to 2^{16} provide acceptable alternatives to the use of conventional low resolution rotary encoders.

If the new arm is to be used to investigate further the performance of direct-drive arms and in particular the application to assembly then more attention must be paid to the design of the arm links. In order to produce high performance, the weight of these links should be minimised and a possible way of doing this is with the use of composite matrerials. The design of the arm links in composites then opens up the whole new areas of materials, stress etc. which have not been considered so far. The problem of providing two (or three) extra degrees of freedom for vertical motion and orientation of the end effector can also be addressed. Whilst there are many solutions to this problem, none of them have the elegance and simplicity of the four bar parallel drive mechanism.

The second major limitation of the IC Experimental Arm is the MC6809 controller. The third order model of the actuator system could not be implemented on the arm as the shortest sampling period required to compute the control law is higher than the electrical time constant of the servo motors. In addition, it was found that acceleration feedforward signal of the

tracking controller suffered from digital noise due to truncation errors.This too is ultimately due to the lack of computing power of the controller. Upgrading the computer to something like the 32-bit MC68020 with floating point hardware will provide sufficient power to be able to increase the sampling frequency, which will improve the performance of the controller, and the third order model, discussed in section 7.2.1 can then be implememted. The inverse kinematics computations and the inverse Jacobian matrix could then be carried out more than 20 times faster than at present. In addition a more sophisticated robot programming language could be used so making the system more 'user friendly'.

The use of a powerful computer controller also introduces the opportunity to investigate more sophisticated control schemes. The new arm would experience some coupling in practical applications when for example a component for assembly is grasped. So the self-tuning control developed by Dr. Ristic (Ristic, 1986) is particularly well suited to this control problem. If coupling is considered, then the computed torque method could be applied which, for the four (or five) degree of freedom proposed arm would not impose too severe a computational burden.

Finally, if the proposed arm is made very light so that very high velocities and accelerations can be achieved then the arm links may possess some flexibility. In this case the control scheme developed by Cannon specially for flexible arms (Cannon and Schmitz 1984) could be applied. The scheme relies on vision sensing of the end point of the arm and the vision could also be used to help in assembly operation. Hence even if the arm's flexibility was small, it would be worth while to investigate the control scheme for the added benefit of being able to use a vision system.

In all the above control schemes though the availability of (assumed) unlimited computing power opposes the idea of keeping the system simple. The beauty of the IC Experimental Arm is its simplicity of design and construction. The implemented second order

tracking controller and the force controller, which does not require expensive and complex sensors, follow this principle and so on this level is prefer able to all the more sophisticated control schemes. This is not to say that these schemes should not be investigated but that their practical application to the four bar parallel drive mechanism of direct-drive arms is limited.

REFERENCES

Asada, H., and Kanade, T. (1983). Design of Direct-Drive Mechanical Arms, Transactions of the ASME, Journal of Vibration, Acoustics, Stress and reliability in Design, Vol. 105, July 1985.

Asada, H., and Kanade, T. and Takeyema, I.(1983) Control of Direct-Drive Arm, Transactions of the ASME, Journal of Dynamic Systems, Measurement and Control, Vol. 105, September 1983.

Asada, H., and Youcef - Toumi, K. (1983) Development of a Direct-Drive Arm Using high Torque Brushless Motors, Proceedings of The First International Symposium of Robotics Research, August 1983.

Asada, H., and Youcef - Toumi, K. (1984) Analysis and Design of a Direct-Drive Arm With a Five - Bar - Link Parallel Drive Mechanism, Transactions of the ASME, Journal of Dynamic Systems, Measurement and Control, Vol. 106, September 1984.

Brady, M. Hollerbach, J.M., Johnson T., Lozano-Perez T. and Mason, M. (1982). Robot Motion : Planning and Control, The MIT Press (1982).

Cannon, R.H. Jr. and Schmitz, E.(1984). Initial Experiments on the End - Point Control of a Flexible One-Link Robot, The International Journal of Robotics Research, Vol. 3, No. 3, Fall 1984.

Dorf, R.C. (1980) Modern Control Systems, Addison-Wesley Publishing Company.

Hollerbach, J. M. (1983) Dynamic Scaling of Manipulator Trajectories, AI Lab, MIT, AI memo no. 700.

Hollerbach, J. M. and Sahar, G.(1984) Dynamic Scaling of Manipulator Trajectories, Transactions of the ASME, Journal of Dynamic systems, Measurement and Control, Vol. 106, March 1984.

Inoue, H. (1974) Force Feedback in Precise Assembly tasks, MIT Artificial Intelligence Laboratory Memo 308 August 1974.

Jensen, P. W. (1963) Cam Design and Manufacture, The Industrial Press, New York.

Kane, T. R. and Levinson D.A. (1983).The Use of Kane's Dynamical Equations in Robotics, The International Journal of Robotics Research Vol. 2, No. 3

Kazerooni, H. Bausch, J.J. and Kramer B.M. (1986) An Approach to Automated Deburring by Robot Manipulators, Transactions of the ASME, Journal of Dynamic Systems, Measurement and Control, Vol. 108, December 1986.

Koivo A.J. and Guo T.R. (1981). Control of Robotic manipulator with Adaptive Controller, IEEE Proceedings of the 20th Conference on Decision and Control, December 1981.

Kondoleon, A.S. (1976). Applications of Technology-Economic Model of Assembly

Techniques and Programmable Assembly Machine Configurations, M.Sc. Thesis, Massachusettes Institute of Technology. 1976 (Source : Flexible Assembly Systems, Plenum Press, 1984).

Koren, Y. (1979) Design of Computer Control for Manufacturing Systems, Transactions of the ASME, Journal of Engineering for Industry, Vol. 101.

Kuo, B.C. (1980) Digital Control Systems, Holt-Saunders International Editions.

Luh, J. (1983). Conventional controller Design for Industrial Robots - A Tutorial, IEEE transactions on Systems, Man and Cybernetics, Vol. SMC-13, No. 3, May/June1983

Luh, J., Walker, M.W. and Paul R.P., (1980). Resolved Acceleration Control of Mechanical Manipulators, IEEE Transactions on Automatic Control, Vol. AC-25, No. 3, June 1980.

Nevins, J.L. and Whitney, D.E. (1978) Computer-controlled Assembly, Scientific American Vol. 238, No. 2.

Owen, T. (1985). Assembly with Robots, New Technology Modular Series, Kogan Page.

Pak, H. A. and Turner P. J. (1986) Optimal Tracking Controller Design for Invariant Dynamics Direct Drive Arms. Internal Report, Centre for Robotics and Automated Systems, Imperial College of Science and Technology, London.

Paul, R.P. (1972). Modelling, Trajectory Calculations and Servoing of a Computer Controlled Arm, Stanford Artificial Intelligence Laboratory Memo AIM-177 November 1972.

Paul, R.P. and Shimano, B. (1976) Compliance and Control, Proceedings of the Joint Automatic Control Conference, PP 694-699.

Ristic, M. (1986). Dynamic Analysis and Control af Anthropomorphic Robots, Ph.D. Thesis, Mechanical Engineering Department, Imperial College, May 1986.

Sahirad, M., Ristic, M. and Besant C.B. (1986). Implementation of Adaptive Positional Control and Active Compliance for Robots. UK Research in Advanced Manufacture. Proceedings of the Institution of Mechanical Engineers (1986).

Waters, R.C. (1979). Mechanical Arm Control, A.I. Memo AIM-549, Artificial Intelligence Laboratory, Massachusettes Institute of Technology, October 1979.

Welburn, R. (1984). Ultra High Torque motor system for direct drive robotics, Robots Conferences 8th vol. 2, Detroit, Michigan. pp 19.63 19.71

Whitney D.E. (1969). Resolved Motion Rate Control of Manipulators and Human Prothese, IEEE Transactions on Man - Machine Systems, Vol. MMS-10, June 1969.

Whitney D.E. (1977). Force Feedback Control Of Manipulator Fine Motions, Transactions of the ASME, Journal of Dynamic Systems, Measurement and Control, Vol. 99, June 1977.

APPENDIX

HARDWARE SPECIFICATION OF IMPERIAL COLLEGE EXPERIMENTAL DIRECT DRIVE ARM

The following appendix gives detailed specifications of the hardware used in the experimental arm. This includes complete performance characteristics of the actuators, comprising the motors, the tachometers and amplifiers. In addition, the specifications of the microprocessor cards, optical incremental encoders, the pneumatic valves used for the control of the pneumatic actuator and gripper are given.

1 ACTUATORS

1.1 D.C. SERVOMOTORS

The DC servomotors are manufactured by Electro-Craft Corporation, 1600 Second Street South, Hopkins, Minn. 55343 (612) 931-2700 .

The S-19 series servomotors are designed especially for depedable duty in rugged industrial environments, and are especially well suited for applications which demand high torque at low speeds, or high pulse torque performance. Their ability to produce high torque at low speed makes them ideal for use in direct-drive robots.

JOINT 1	JOINT 2
S-19-3A	S-19-1A

MOTOR RATINGS		
Continuous Output Power (W)	433	179
Continuous Stall Torque (Nm)	1.24	0.6
Peak Torque (Nm)	6.2	3
Maximum Terminal Voltage (V)	60	60
Stall Current (A)	9.5	7.3
MECHANICAL DATA		
Rotor Moment of Inertia (Kg m^2)	$2.6*10^{-4}$	$1.34*10^{-4}$
Theoretical Maximum Acceleration (rad/s^2)	$24*10^3$	$22*10^3$
Mechanical Time constant (ms)	12	12
Damping constant (Nm/rad.s^{-1})	$1.3*10^{-4}$	$6.7*10^{-5}$
Thermal Resistance (armature-to ambient) (C/W)	1.9	2.7
Maximum Armature Temperature (°C)	155	155
Maximum Friction Torque (Nm)	$5.6*10^{-2}$	$5.6*10^{-2}$
Maximum Shaft Radial Load (Kg)		
(1 in/25.4 mm from front bearing, contiuous)	13.6	13.6
Maximum Shaft Axial Load (Kg)	4.54	4.54
Motor Weight (including tachometer) (Kg)	4.54	2.72
WINDING DATA		
K_T Torque constant +-10% (Nm/A)	$1.3*10^{-1}$	$8.2*10^{-2}$
K_E Voltage Constant ±10% (V/rad.s^{-1})	$1.3*10^{-1}$	$8.2*10^{-2}$
Armature Resistance ±15% @ 25 °C (Ω)	0.4	0.4
Terminal Resistance ±15% @ 25 °C(Ω)	0.65	0.65
Electrical Time Constant (ms)	1.6	1.1
Maximum Operatingspeed (rad/s)	472	524

Maximum No-Load Speed (rad/s)	472	524
Maximum pulse Current (A)		
(before demagnetization)	63	77
Armature Inductance (mH)	0.64	0.42

Table A.1 motor specifications

1.2 TACHOMETERS

The tachometer produces a voltage proportional to the speed of the motor which is attached to it and as such is an integral part of the motor. The tachometers used are manufactured by Electo-Craft Corporation.The same model of tachometer, S496, is used for both motors. The specifications of the tachometer is given in thable A.2.

RATINGS	S496
Linearity (Maximum % Deviation)	0.2
Ripple (Maximum % Peak to Peak AC Component	
at 500 rpm)	2.0
Ripple Frequency (Cycles/Revolution)	21
Temperatue Coefficient (% / °C)	0.01
Insulation Resistance (M)	10
Torsional Resonant Frequency (kHz Minimum)	2

Rated Operational Life (Hours at 4000 rpm)	10 000
Operating Temperature Range (°C)	0 - 155
WINDING DATA	
Output Voltage Gradient (V/krpm ± 10%)	14
Armature Resistance (±20%)	110
Armature Inductance (mH Nominal)	8
Recommended Load Impedance (Ω)	10K

Table A.2 Tachometer Specifications

1.3 AMPLIFIERS

The amplifiers on each axis are manufactured by Electro-Craft Corporation and one of the models from the LA-5000 series of linear amplifiers is used on each of the two axes. They are designed to drive permanent magnet DC motors of both the iron core and moving magnet coil types. The amplifiers and motors are intended for use in a broad range of industrial servoapplications, including machine tool nd robotics applications, and are used in both velocity and position type servo systems.

Incorporated in the amplifier is overload circuitry which guards against over-loading either the amplifier or the motor. Two feedback loops are used; an inner loop feeding back a signal proportional to the motor current, and an outer loop feeding back a signal proportional to motor velocity. Since the outer loop is the dominant loop, the system is primarily a velocity sevo loop system. The specification of the LA-5400 model is given in the following table.

	LA-5400
PERFORMANCE	
Peak Current (A)	22
RMS (or Continuous) Output Current (A) at 50 °C	9
No-Load Output Voltage (V)	44
Output Voltage (V) @ 9A	15 - 35
Peak Output Power (W) @25 V	550
Contiuous Output Power (W) @35	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	±10 nominal; ±60 maximum
PHYSICAL CHARACTERISTICS	
Height (cm)	15.2
Width (cm)	16.5

Length (cm)	26.7
Weight (kg)	7.1
INPUT POWER REQUIREMENTS	115V AC, or 230V AC, 50/60 Hz, 1ϕ

Table A.3 Amplifier Specifications

MICROPROCESSOR AND THE ASSOCIATED SOFTWARE

MICROPROCESSOR BOARD

The MC6809 microprocessor board is used as the basis for the computer controller. This is supplied by Syntel MIcrosystems, Queens Mill Road, Huddersfield, HD1 3PG in the form of their MP09 board.

The SYNTEL MP09 is a high performance 6809 based processor module. The single-Eurocard module includes 6840 triple timer, RS232/422 interface signals and sockets for 32Kbyte prom-ram mix. The system is expandable via the standard G-64 bus interface and the memory map may be flexibly taylored with a 7640 prom.

FEATURES

* MC6809 PROCESSOR High performance 16-bit instructions.

* 32K BYTE PROM / RAM SOCKETS No external memory requirement for many systems.

*6840 COUNTER / TIMER Real-time applications.

* G-64 BUS COMPATIBLE Easily expandable.

* RS 422 / 232 INTERFACE Remote communications or debug channel.

* BIPOLAR MAPPING PROM Easily taylored memory map.

* TRUE VMA GENERATION Protection during SYNC or BUS request dead cycles.

* BURNED IN TESTES Reliable operation.

MEMORY EXPANSION CARD

The SYN MEM 1 is an eprom/ram module capable of supporting up to 32K-bytes in a flexible mix. Standard 28 pin sockets allow use with a wide range of CMOS and NMOS RAMS and EPROMS. The single -Eurocard Module when populated with cmos rams, is especially usefull in power- failure tolerant systems. Upon receipt of a mains power-fail input the module generates a Non-Maskable Interrupt prior to disabling the ram. This allows a responding processor to store critical data before the ram becomes protected from eratic processor operation as the 5 volt supply fails.

FEATURES

* ON-BOARD LITHIUM BATTERY Data retention in excess of 2 years.

* STANDARD EUROCARD FORMAT Compatible with off-the-shelf racking systems.

* PAGE - MODE INPUT For systems larger than 64k-byte.

* STANDARD 28 PIN MEMORY SOCKETS Compatible with 2716, 2732, 2764,

HM6116, MK4802, TC5517, TC5564, HM6264.

* STRAP SELECTABLE ADDRESS Easily taylored memory map.

* G-64 BUS COMPATIBLE Flexible and expandable.

TIMER INPUT / OUTPUT BOARD

The SYNTEL Timer-Input-Output module (TI01) is designed as a flexible digital interface from the microprocessor bus. The module provides 32 general purpose digital input / output lines and 8 special purpose interrupt / handshake lines from two MC6821 PIAs.

A further 9 I / O lines are associated with an MC6840 which provides three 16-bit timer / counter channels for event counting, pulse train generation, period measurement etc.

All outputs are buffered by (removable) darlington transistor arrays capable of driving 5 to 50V at 300mA.

All inputs may be optionally terminated.

FEATURES

* 49 I/O LINES Compact unit.

* 6840 TRIPLE COUNTER/TIMER Pulse-train generation, Event counting, Period measurement.

* G-64 BUS COMPATIBLE Expandability & compatibility.

* SWITCHABLE ADDRESS SELECTION Flexibility.

* OUTPUTS 5 TO 50V, 0.3A Direct relay drive etc.

* STANDARD EUROCARD FORMAT Compatible with off-the-shelf racking systems.

* BURNED- IN TESTED Reliable operation.

*LOW COST Wider market.

POWER SUPPLY CARD

A 30 watt switched-mode power supply designed for use in Eurocard systems is used. The card can deliver 3A current at 5V level and 1.2A current at 1.2V level. The present system's power requirement is well within the supply limits.This card is manufactured by the U.S Astec Electronics, 1101 Spacepark Drive, Santa Clara, California 95050, USA. It was obtained from Astec Europe Ltd., 16 Albury Close, Reading Berkshire RG3 1BD.

The unit features low magnetic radiation and is built to conform to International Safety specifications. The specifications of the power supply board is given in the following table.

PARAMETER	MIN	TYP.	MAX.	NOTES
Input Voltage (VAC)	90	115	135	
	170	230	265	
Inrush Current (A)	35			
Input Frequency (Hz)	47	50/60	63	
Outputs : VO1 (V)	4.95	5.0	5.05	+5V output
IO1(A)	0.75		3.0	
VO2(V)	11.7	12.3	13.0	+12V output
IO2(A)	0.3		1.2	
Efficiency (%)	60			at full load
Operating Temperature (C)	0		50	
Output Power (w)	7.5		30	
Output Ripple : VO1 (mV)		25	50	1 Hz to 300 Hz

	VO2 (mV)	60	120	
Load Regulation (%)				
	VO1(%)		1	IO1=50-100%, IO2=100%
	VO2(%)	2	4	IO2=25-100%, IO1=100%
Load Step Response Time (μs)		350	500	for 25% step
Hold-up Time (ms)	20			at max. load
MTBF		Greater than 6000 hours when operated at full load and ambient temperature of 35 C, at 60% confidence level.		

Table A.4 Power Supply Board Specifications