MOTOR SIMULATION
OF A
SHOCK ABSORBER

by

Brian Clancy, B. Sc. Eng

A thesis submitted to

Dublin City University

for the degree of

Master of Engineering

School of Electronic Engineering

Dublin City University

September 1996

Abstract

The use of modern control theory to produce an electrical motor simulation of a Formula 1, Grand Prix, passive motorcycle front suspension shock absorber is investigated It is shown, using a test-rig comprising two permanent magnet DC motors directly coupled, that desired shock absorber responses to load forces can be achieved using model reference control The controller feedback in this test rig is provided via a high resolution rotary position sensor A stochastic Kalman filter is used to produce estimates of the load(disturbance), force and velocity from this position information All states are then used in the controller

A mass, spring and damper model is chosen as a suitable representation of a shock absorber, and is assumed sufficiently complex to justify the control techniques used This linear model is translated using mathematical techniques into a rotary equivalent that is compatible for use in the controller This translation takes into account thermal effects, as well as kinematic requirements encountered by the motor on the basis of load-force data taken from the front suspension of a Formula 1 motorcycle in race conditions The parameters of the mass spring and damper model are found from simple static tests using the shock absorber removed from the bike

DECLARATION

I hereby declare that all the work presented in this thesis is my own, except where reference has been made. I also declare that no part of this thesis has been submitted for a degree at any other institution.

Signature..... Brian Clancy

Date

ACKNOWLEDGEMENTS

CONTENTS

Chapter 1 Introduction

Chapter 2 System Modelling

Chapter 3 Control Approach

Chapter 4 Coupling Sizing

Chapter 5 Control Design

Chapter 6 Implementation

Chapter 7 Conclusions and Recommendations

Appendix A
Motor Parameters
    A.1   Circuit Diagram Reduction

Appendix B   -
Encoder Interface Card Circuit Diagrams

Appendix C
Software Listing

Appendix D
Servo Motor Control Block Diagram

CHAPTER 1

INTRODUCTION

For the past 100 years, comfort in road vehicles has been linked to the introduction of suspension units consisting of passive components that isolate passenger compartments from mechanical vibration and reduce the effect of braking forces[1-1] Near optimal performance has been achieved by years of experimentation testing and careful selection of individual component characteristics, intended to minimise the design constraints of passive systems

In recent years, interest in computer controlled active suspension systems for both automobile and motor cycle has increased These suspensions can be categorised into two groups semi-active, and fully active systems[1-2] The semi-active type provides real time controlled dissipation of energy This is accomplished by an actuator called an active damper used in parallel with an ordinary spring The fully active system uses an actuator which creates the desired force in a suspension system

In general, suspension systems use only springs and dashpots, i e passive components, which can only react to relative positions and velocities in one manner Active suspensions seek to overcome this limitation by augmenting or replacing passive components with actuators that can be commanded to exert any desired force[1-3] Electric drives are considered as an alternative to the much researched hydraulic and electro-hydraulic actuators[1-4] typically used in this area

This thesis documents an initial step in an overall objective undertaken by Power Electronics Ireland[1] to research and develop an electrical fully active motor cycle front suspension The function of a motor cycle front suspension is to transmit forces, attenuate road vibrations, and to keep both wheels on the track The front suspension of a motor cycle will compress under cornering loads and in response to braking, and similarly it will extend during acceleration To some extent this functionality is acceptable, however it is highly desirable that it be capable of change through varying the main parameters of the suspension to suit different operating conditions

---

[1] An initiative sponsored by the Irish Government under its Programme of Advanced Technology (PAT)

The main section of a motor cycle passive front suspension is the shock absorber which consists of a mechanical spring and dashpot arrangement (see Fig 1 1(b)) The intention of the work carried out for this thesis was to design and implement an electric motor simulation of a passive mechanical shock absorber This being achieved the controller designed can be viewed as an active shock absorber controller (see Fig 1 1(a)) This was considered an initial step in designing an electrical active suspension

(a) Test rig (Motor and controller)   (b) Shock absorber
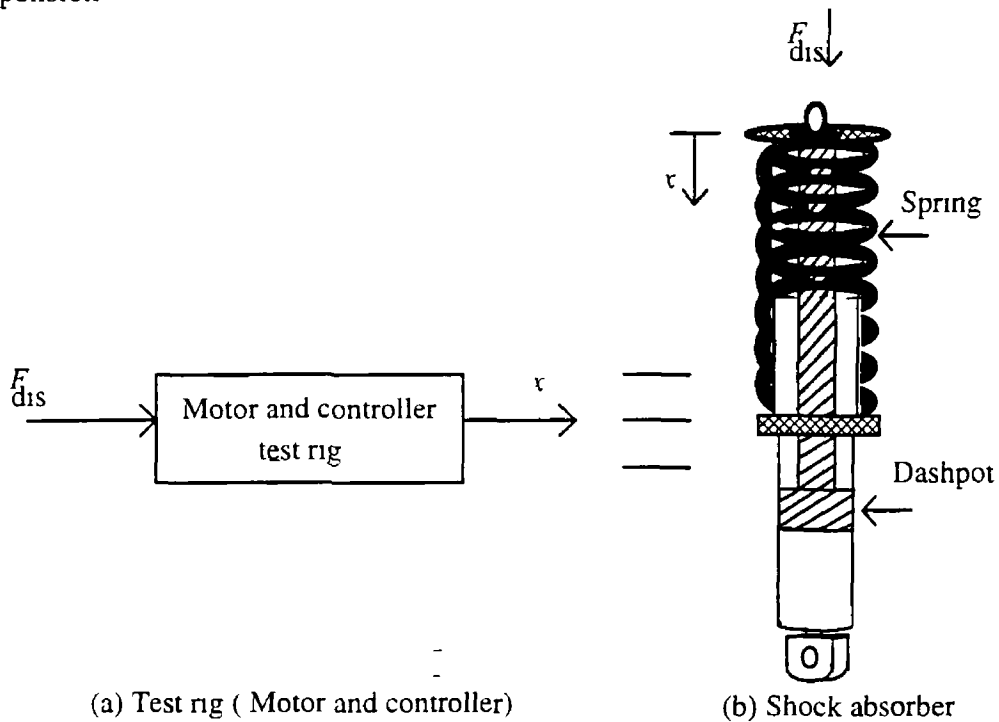
Figure 1 1 Shock absorber simulation test rig and simplified shock absorber

A Fomula 1 motor cycle front shock absorber was initially mathematically modelled and using a motor test rig the objective was to 'force' the motor to perform as a passive shock absorber In other words the electrical motor in the test rig responds to F_{dis} exactly as the mechanical passive shock absorber responds to F_{dis} This being achieved a test platform exists for further research and development in the electrical active suspension area

This shock absorber simulation problem is treated as a DC servo control problem With the introduction of digital signal processing (DSP) and faster microprocessors, more numerically intensive algorithms are being considered for the control of DC motors used in servo applications such as this[1-5] The control results achieved through digital control are often better than those achieved through analogue systems

2

Thesis Structure

The thesis is divided into seven chapters

The Introduction gives an overview of the both the intentions of and the approach to the thesis Chapter 2 introduces the shock absorber model and the test rig system being used The test rig consists of a motor and current controller around which the digital controller is placed It is shown in this chapter that the bandwidth of the current loop is large enough to assume that its dynamics can be modelled as a DC gain Chapter 3 describes the control problems posed by this shock absorber simulation project and introduces the required control approach which involves model reference and state feedback control along with stochastic estimation techniques All concepts are described and referenced throughout

Chapter 4 is an extension of the modelling described in chapter 2 A coupling sizing method is described for the general linear to rotary conversion problem and subsequently is used to complete the reference model of the shock absorber used in the digital controller Chapter 5 is concerned with the design of the controller for implementation All the control methods referred to in chapter 3 are now specifically adapted for the shock absorber simulation test rig Sample inputs are used to test the closed loop controller

Chapter 6 deals with the implementation of the controller on the motor test rig using a Loughborough Sound Images (LSI) PC System Board and the Texas Instruments TMS320C30 microprocessor Hardware requirements of the controller are introduced including the data input/output (I/O) and measurement sensor description The software approach is given in flow chart form Final implementation using the test rig is carried out and a comparison is made between the actual response and desired response to support the control approach chosen throughout Chapter 7 contains comments and conclusions regarding the overall thesis and implementation Recommendations for further work that may improve the system and advance the electrical active suspension concept are presented

CHAPTER 2

SYSTEM MODELLING

2 1 Introduction

The following chapter formulates a front suspension mechanical shock absorber model and hence the reference model used in the controller design The test rig used for the implementation of the motor controller consists of two permanent magnet DC motors coupled back to back The coupling between the motors is rigid The first motor is used as an active load that simulates external disturbances and the second is used to simulate a passive mechanical shock absorber

This section deals with the identification of a mathematical model for the test rig A mathematical model of the test rig is derived and is then simplified into an equivalent lower order transfer function For the purpose of design this overall complex electromechanical servo system, (i e the test rig consists of a motor and analogue current controller) can be simplified into an equivalent lower order transfer function by use of mathematical reduction and computer aided design (CAD) system identification techniques

2 2 Shock Absorber Models

Choosing and deriving a model of the front suspension shock absorber to be simulated is taken as the starting block in the control design problem being tackled in this project A shock absorber consists of many parameters, all of which have an effect on its static and dynamic response However, the most dominant parameters i e those that have most effect on the response of the shock absorber are modelled and the parameters that are less dominant are assumed to have negligible effect on the response, (these parameters are the spring stiffness and damping) This assumption is supported by research[2-1] The reason for this is to reduce the complexity of modelling and more specifically the complexity of the control required

4

2.2.1 An Ideal Shock Model

An ideal mechanical shock absorber model is the type chosen as it comprises the dominant parameters of the mechanical system i e spring stiffness and damping This analytical model is used as the starting block in the control design In preference to the sky-hook damping model proposed by Karnopp[2-2], a mass, spring and damper model is selected This model is chosen over the sky-hook model as it is a well established method of modelling a suspension system, with the advantage that it is directly compatible with state space control techniques

The shock absorber characteristics are modelled by the spring stiffness coefficient K_S, the damping rate B_S and the sprung mass M_S The displacement of the sprung mass is given by x Road disturbances and cornering forces are modelled by a linear force, F_{dis}

In this passive system_it is shown that the damping and spring stiffness must be chosen to obtain a compromise between both high frequency and low frequency disturbances i e the more damping in the passive system the better the low frequency control and the worse the high frequency performance becomes

Spring stiffness and damping are chosen in a motorcycle shock absorber to maintain optimal road holding in the straight along with optimal cornering performance There is quite clearly a compromise to be made when choosing both parameters As mentioned in the introduction, during cornering the front suspension shock absorber reacts to the forces due to braking and acceleration These forces are referred to as low frequency large disturbances, F_{dis}
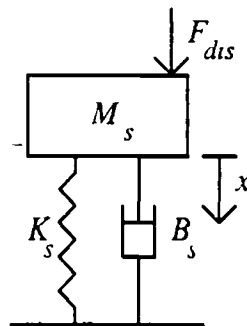
Figure 2.1 Ideal front suspension shock absorber model

$$\frac{x}{F_{dis}} = \frac{1}{M_s s^2 + B_s s + K_s}$$

(2 1)

State space description of the mechanical shock absorber is given below in Eqn(2 4)
This description is directly adaptable in this project as it is the model representaion
that will be used in the control design[2-3]

The reference suspension model is given by the linear state space equation

$$\dot{X}_M = A_M X + B_M F_{dis}$$ (2 2)

where

$$X = \begin{vmatrix} x \\ \dot{x} \end{vmatrix}$$ (2 3)

and

$$A_M = \begin{vmatrix} 0 & 1 \\ -\dfrac{K_s}{M_s} & -\dfrac{B_s}{M_s} \end{vmatrix} , \quad B_M = \begin{vmatrix} 0 \\ \dfrac{1}{M_s} \end{vmatrix}$$ (2 4)

A_M is the plant matrix and B_M is the input coupling matrix and both are selected by
specifying the parameters K_S, B_S and M_S Typical values of parameters are chosen
based on knowledge of the vehicle and the disturbances being experienced Static
tests are carried out on the Formula 1 motor cycle front suspension shock absorber
being simulated and a model is derived using the equations above

2.2.2 Rotary Equivalent

The shock absorber model just derived is that of a linear system In order that a rotary
motor may be used to simulate the dynamics of this, the motor controller requires a
rotary equivalent model of the one outlined above Therefore the resultant reference
model must take the coupling required to translate the linear model to a rotary The
effect of introducing a coupling ratio is simply given by the relationship
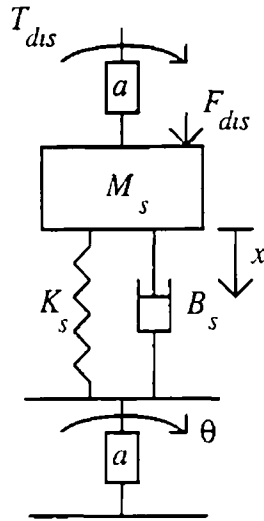
Fig 2 2 Rotary equivalent front suspension shock absorber model

$$x = a \times \theta \qquad (2\,5)$$

and

$$F_{dis} = \frac{T_{dis}}{a} \qquad (2\,6)$$

where

a is the coupling ratio in m/rad θ is equivalent to angular displacement in radians and T_{dis} is the equivalent torque experienced corresponding to the force F_{dis}

In this case the rotary equivalent to the linear model will be the shock model as given below

$$B_M = \left| \begin{array}{cc} 0 & 1 \\ -\dfrac{K_s}{M_s} & -\dfrac{B_s}{M_s} \end{array} \right|$$

$$\qquad (2\,7)$$

$$- \quad B_M = \left| \begin{array}{c} 0 \\ 1 \\ \overline{a^2 M_s} \end{array} \right|$$

7

2.3 The Test Rig

2 3.1 Description

The test rig being used for this shock absorber simulation is a set of two identical directly coupled permanent magnet (PM) motors One motor is controlled to produce the actuator torque T_c to simulate the reaction of the shock absorber and the other is a load motor to produce disturbance torques, T_{dis}, equivalent to the forces, F_{dis}, as described above See Fig 2 3
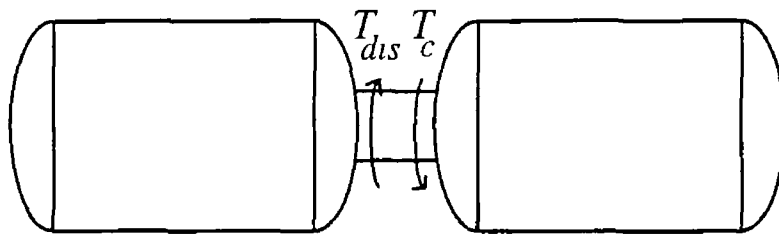
Permanent Magnet DC Motors

Figure 2 3 Simulation system

In DC motors the field flux ϕ is established by the stator either by means of permanent magnets or by means of a field winding In permanent magnet DC motors the field system can be more compact compared with an electromagnetic DC motor, and no current is needed to provide the field flux[2-4] The characteristics of the PM motor are identical to those found in a separately excited motor with a constant current in the field coil

The DC motors chosen for this rig are driven by A E G Pulse Width Modulation (PWM) drive systems The drive system consists of PWM inverters that convert mains AC supply from an isolating transformer into a smoothed DC "bus", to supply power to drive the motor[2-5] The PWM system is a four quadrant system as shown in Fig 2 4
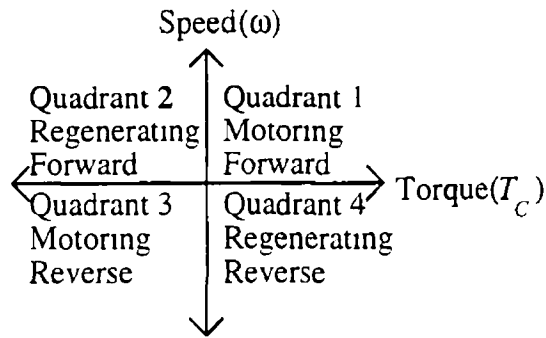
Figure 2 4 Motor Drive System

Motoring mode forward and reverse

In these modes, energy flows from the AC mains via the transformer and PWM to the controller

Regenerating mode forward and reverse

In these modes, energy flows from the motor via the controller, back to the DC bus and into the power supply regeneration circuit

The controller has a PWM full bridge topology which converts energy from the DC busbar (from the power supply module) to the current and voltage required to drive the motor over its entire speed range

2 3 2 Motor Simplification

The permanent magnet motors consist of rare earth (samarium cobalt) magnets on the stator bonded into a cylindrical tube to provide the working flux field The magnetic field set up by these magnets produces a high torque/weight ratio [2-6] The motor has the following general voltage equation

$$V = R_A I + L_A \frac{dI}{dt} + K_B \omega$$

(2 8)

where V is the motor terminal voltage

I is the motor armature current

R_A is the resistance of the armature windings

L_A is the inductance of the armature windings

ω is the angular velocity of the rotor

K_B is the back emf constant

and

$$T_C = J_M \frac{d\omega}{dt} + B_M \omega + T_{dis}$$

(2 9)

where J_M is the total system inertia

B_M is the viscous damping coefficient

T_{dis} is the disturbance torque

Eqns (2 8) & (2 9) are the standard DC machine equations

2 3.3 Analogue Controller Simplification

The drive selected is one of the AEG permanent magnet types [Refer to manual] It is a current controlled DC servo and consists of analogue current closed loop control A mathematical representation of this analogue system is given below in Fig 2 5 This block diagram represenatation is achieved from an analysis of the circuit given in appendix A The objective of this examination is to prove that the bandwidth of the current loop, being sufficiently large, can be simplified to be represented by a simple DC gain To prove that this is an acceptable assumption the current loop must be identified This is achieved using two well established techniques

- Mathematical Model Simplification
- System Identification

The current loop can be represented by the diagram below This figure represents the analogue drive broken down into its essential components Subsequently we will take a closer look at these components
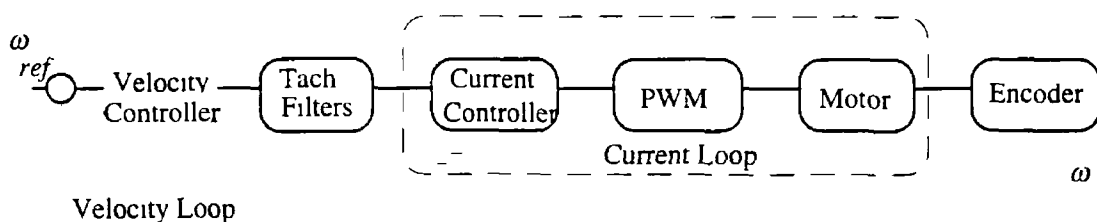
Figure 2.5 Block diagram of current servo system

Fig 2 6 below is derived in Appendix A 1
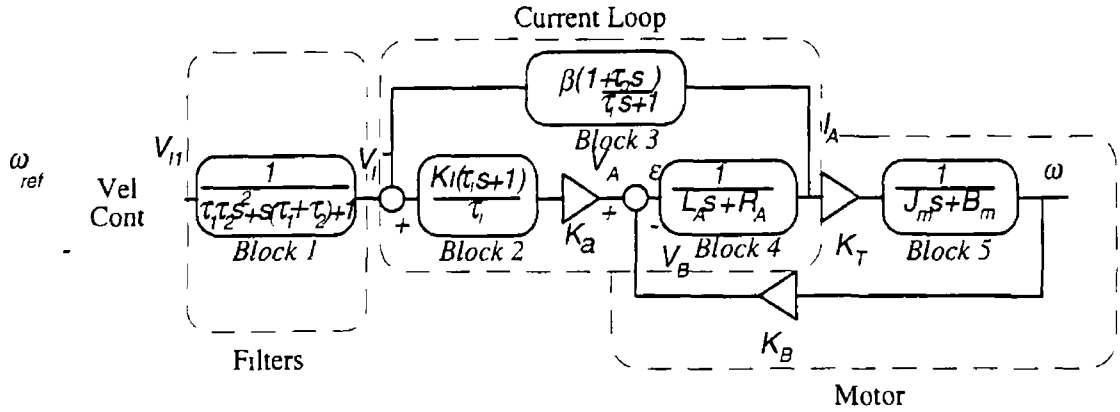
Figure 2.6 Mathematical block diagram representation of current servo system

Firstly simplify the current loop

$$\frac{\omega}{I_A} = \frac{K_T}{J_m s + B_m} \tag{2 10}$$

$$\frac{I_A}{\varepsilon} = \frac{1}{L_A s + R_A} \tag{2 11}$$

$$\varepsilon = V_A - V_B \tag{2 12}$$

$$\frac{I_A}{V_A} = \frac{J_m s + B_m}{(J_m s + B_m)(L_A s + R_A) + K_B K_T} \tag{2 13}$$

$$\Rightarrow \frac{I_A}{V_A} = \frac{\frac{J_m s}{B_m} + 1}{R_A(\frac{J_m s}{B_m} + 1)(\frac{L_A s}{R_A} + 1) + \frac{K_B K_T}{B_m}} \tag{2 14}$$

Using the motor parameters given in Appendix A,

$$\tau_e = \frac{L_A}{R_A} , \quad \tau_{em} = \frac{J_m R_A}{K_B K_T} , \quad \tau_m = \frac{J_m}{B_m} , \text{ Eqn (2 14) can be expressed as in Eqn (2 15)}$$

Where τ_m and τ_e are the mechanical time constants of the motor and τ_{em} is a cross constant used for reduction purposes

$$\frac{I_A}{V_A} = \frac{\tau_m s + 1}{R_1(\tau_m s + 1)(\tau_e s + 1) + \dfrac{\tau_m R_A}{\tau_{em}}}$$ (2 15)

$$\approx \frac{\tau_{em}(\tau_m s + 1)}{\tau_m R_A(\tau_{em} s + 1)}$$ (2 16)

$$\tau_m \gg \tau_e$$
$$\& \tau_m \gg \tau_{em}$$

This represents the motor transfer function in the current loop as shown below
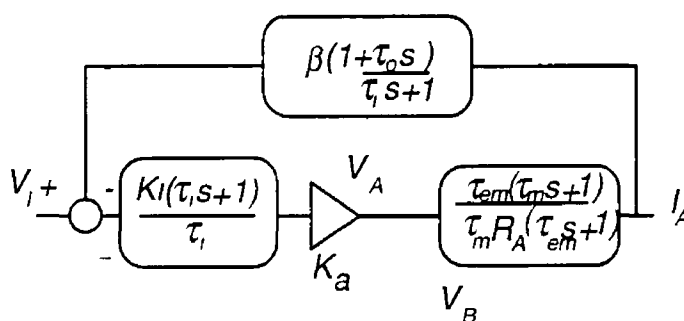
Figure 2.7 Reduced mathematical block diagram representation of current servo system

The simplified transfer function represents the current loop and the motor windings

$$\frac{I_A}{V_I} = \frac{\dfrac{K_I K_a \tau_{em}}{\tau_m R_A}(\tau_i s + 1)(\tau_m s + 1)}{\tau_i s(\tau_{em} s + 1) + \dfrac{K_I K_a \beta \tau_{em}}{\tau_m R_A}(1 + (\tau_o + \tau_i)s)(\tau_m s + 1)}$$ (2 17)

Substsituting $K = (K_I K_a \beta \tau_{em}) / (\tau_m R_A)$ into the equation 2 17 gives

$$\frac{I_A}{V_I} = \frac{\dfrac{K}{\beta}(\tau_i s + 1)(\tau_m s + 1)}{(\tau_i \tau_{em} + K\tau_o \tau_{em} + K\tau_m \tau_i)s^2 + (\tau_i + K\tau_o + K\tau_i + K\tau_m)s + K}$$ (2 18)

Because $\tau_i \tau_{em} \ll K(\tau_o \tau_m + \tau_i \tau_m)$ and $\tau_i \ll K(\tau_o + \tau_i + \tau_m)$, Eqn (2 18) can be approximated as

$$\frac{I_A}{V_I} = \frac{\dfrac{K}{\beta}(\tau_i s + 1)(\tau_m s + 1)}{K(\tau_o \tau_m + \tau_i \tau_m)s^2 + K(\tau_o + \tau_i + \tau_m)s + K}$$ (2 19)

12

$$\Rightarrow \frac{I_A}{V_I} = \frac{\frac{1}{\beta}(\tau_i s + 1)(\tau_m + 1)}{((\tau_u + \tau_i) + 1)(\tau_m + 1)} \qquad (2\ 20)$$

$$\Rightarrow \frac{I_A}{V_I} = \frac{\tau_i s + 1}{\beta((\tau_u + \tau_i)s + 1)} \qquad (2\ 21)$$

Using this model the current loop is reduced to a very manageable 1st order transfer function The time response of this simplified mathematical system was then analysed using time and frequency response techniques[2-7] The bandwidth of the current loop is approximately 350 rads/sec An approximation is sufficiently accurate in this case as it provides an upper limit for the closed loop digital controller that is designed and implemented in this application

2 3 4 Filter simplification

There are two tachogenerator filters present at the input to the analogue servo system that can be simplified and combined by the transfer function shown in Appendix A 1 The output of the tachogenerator filter (see Appendix D) feeds directly into the current command of the current controller

2.4 System Identification

System Identification is the subject of constructing mathematical models of dynamic systems based on observed data from the systems Although data sheets supplied with motors are usually accurate it is often advisable to ensure the models being used are correct and that there are no large discrepancies between the data sheets and the actual parameters

2 4.1 Identification Procedure

The construction of a model from data involves three basic entities
- The data

It is vital in identification that the correct input and output data is used to make the information gathered maximally informative
- A set of candidate models

13

When identifying a system a suitable model must be used to ensure accurate estimation of parameters This can be achieved by using functions in the CAD package, MATLAB® System Identification Toolbox[2-8]

- A rule by which candidate models can be assessed using the data

The accuracy of the model must be determined by attempting to reproduce the measured data using the model found

2.4 1 1 Model Type

For this system a linear-time invariant model is chosen The model is described by

$$y(t) = G(q)u(t) + H(q)e(t) \qquad (2\ 22)$$

where q^{-1} is the backshift operator, $e(t)$ is the noise term and

$$G(q) = \sum_{k=1}^{\infty} g(k)q^{-k}, \ H(q) = 1 + \sum_{k=1}^{\infty} h(k)q^{-k} \qquad (2\ 23)$$

For practical purposes G and H are each expressed in terms of a finite number of numerical values and these numbers determine the order of the model The way in which these values are calculated and combined determines the model structure The identification process now becomes a search for the best set of parameters Θ

$$y(t) = G(q,\Theta)u(t) + H(q,\Theta)e(t) \qquad (2\ 24)$$

The estimation procedure is used to select the parameter vector Θ

An ARX (auto regressive with exogenous input) model is represented by an input-output relationship The simplest input-output relationship is obtained by describing the present output in terms of the past inputs and outputs in a linear difference equation

$$y(t) + a_1 y(t-1) + \quad + a_{n_a} y(t - n_a)$$
$$= b_1 u(t-1) + \quad + b_{n_b} u(t - n_b) + e(t) \qquad (2\ 25)$$

where n_b is the total number of inputs and n_a is the total number of outputs

14

It is common to assume that *e(t)* is Gaussian. The white noise term here enters as a direct error in the difference equation and because of this the model above is often called an *equation error model*. The adjustable parameters are

$$\Theta = \left[a_1 a_2 \dots a_{n_a} \; b_1 \, b_2 \dots b_{n_b} \right]^T \tag{2.26}$$

Introducing,

$$A(q) = 1 + a_1 q^{-1} + \dots + a_{n_a} q^{-n_a} \tag{2.27}$$

$$B(q) = b_1 q^{-1} + \dots + b_{n_b} q^{-n_b} \tag{2.28}$$

i.e.

$$G(q,\Theta) = \frac{B(q)}{A(q)}, \; H(q,\Theta) = \frac{1}{A(q)} \tag{2.29}$$

The signal flow can be depicted as in Fig 2.8. It is obvious that this is not a complex model in that white noise is assumed to go through the denominator dynamics of the system before being added to the output. However this method will be used as an initial step in the identification procedure.
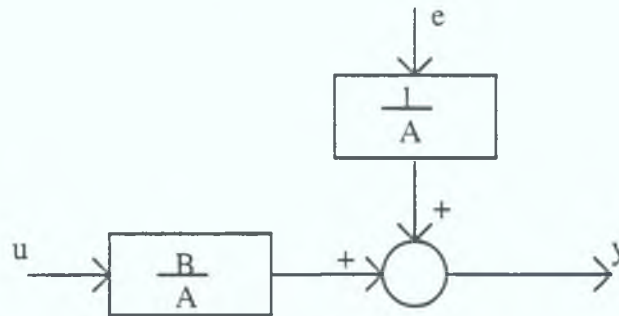
Figure 2.8 ARX signal flow diagram

The disadvantage of this model is that there is a lack of freedom in describing the properties of the noise term. From the diagram above Fig 2.8 the noise term is assumed to be filtered by the denominator dynamics. This is not as natural a case as it could be. This model is the initial attempt to produce an accurate identification; in the event that the results achieved are not sufficiently accurate we may require to use a different model structure[2-9].

15

2 4.2 Identification Implementation

Implementing identification is carried out following the identification steps outlined in the identification procedure described in the previous section The current loop is a SISO (single input/ single output) type system and the input and output measurements are voltage and current respectively A square wave of 8Hz and an amplitude of 1 V is used as the input test signal to the current loop The voltage and current are sampled at 1kHz using the TMS320C30 DSP (digital signal processing) card already referred to in the introduction This data is then imported to the CAD package MATLAB$^{\circledR}$, where it is used as the input and output data, respectively, in the ARX model as described above

The output data, i e the current measurement, is filtered using a 2^{nd} order Butterworth filter This filtering is available as a function in the MATLAB$^{\circledR}$ System Identification Toolbox If a good filtering algorithm were not available to the designer then alternative models, taking noise into account could be used These methods would include ARMAX (auto regressive moving average with exogeneous input) which allow the noise model to be manipulated to produce a more accurate identification

In this case the filtering produced relatively noise free data and the simpler ARX model is sufficiently accurate to produce a usable model
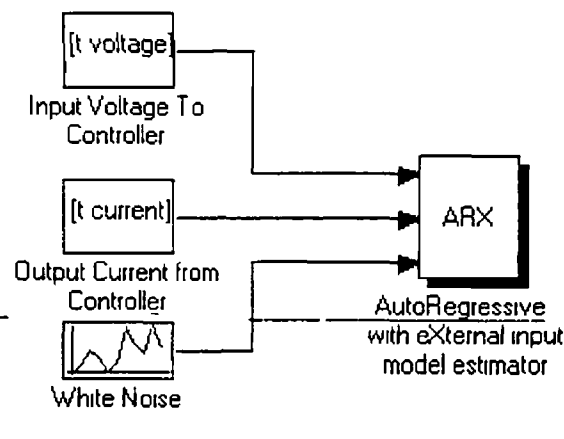
Figure 2 9 Implementation

Using MATLAB® the discrete model Eqn (2 30) is produced

$$\frac{V_i}{I_4} = \frac{3\ 3752q^{-1}}{1 - 0\ 6713q^{-1}}$$

This identified model produces a response to square wave shown below in Fig 2 10 This is compared with the actual measured response of the current loop to the same input As described above this is the assessment of a good model, i e how well the model can reproduce the measured data

Full order Mathematical model and reduced Identified model

Measured Data —————
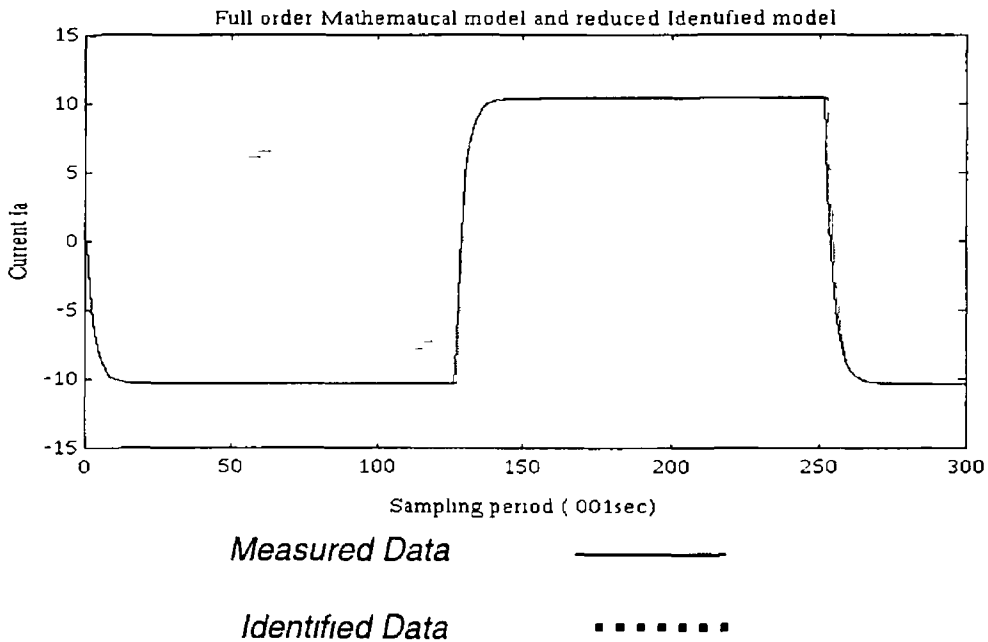
Identified Data ▪ ▪ ▪ ▪ ▪ ▪

Figure 2 10 Measured and identified data

The controller to be designed will be looped around the existing current controller as described above Therefore it is vital that the bandwidth of the current loop be accurately known The identification scheme used produced a first order transfer function representation of the loop The bode plot of this system using the MATLAB ® function (BODE) is shown below Fig 2 11
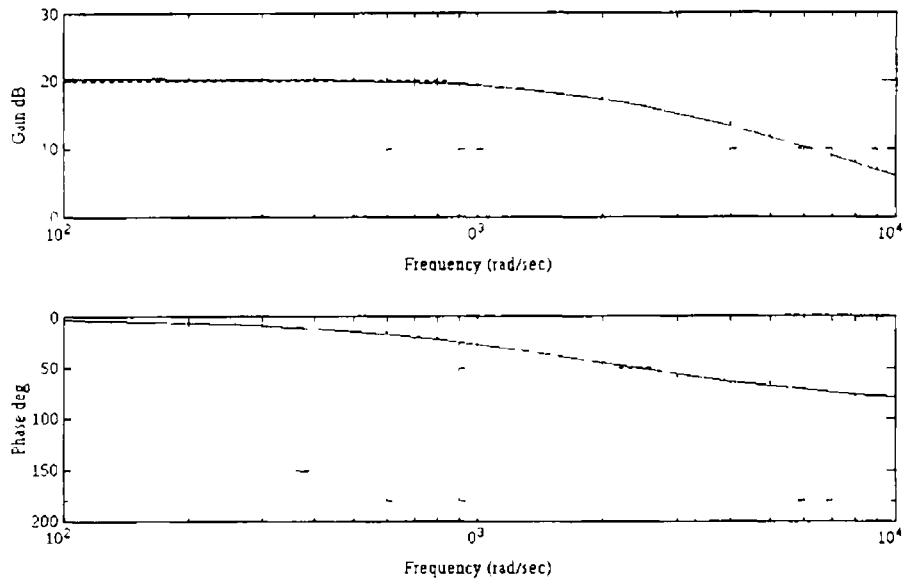
Figure 2 11 Frequency response of identified current loop

The bandwidth is the frequency from which the zero-frequency (steady state) gain is attenuated by more than *3dB* As can be seen from the plot above, the bandwidth is *350rad/sec* and will be the limiting factor in the design of the closed loop controller

Summary

This chapter introduces a simple mass, spring and damper model for a passive motorcycle front suspension shock absorber This model is used in the controller design later To simplify the control design, mathematical and identification techniques are implemented to accurately model the analogue servo system The identified model is tested for accuracy by comparing its output to that of the actual system From these tests it is found that the bandwidth of the current loop amplifier is large enough to assume that it can be represented by a DC gain in the controller

CHAPTER 3

CONTROL APPROACH

3.1 Introduction

The previous chapter described the simplification of the permanent magnet motor and the current control servo system With the development of the microprocessor, especially the D S P , the controller being designed around this inner analogue loop can now be implemented more precisely and flexibly using digital techniques rather than the analogue techniques widely used[3-1]

There are two main approaches that may be followed when designing a digital controller The first method is the use of well established continuous domain methods, called *s-plane*, design with the other being direct discrete time methods called *z-plane* design In the continuous approach all controller design is carried out in the s-plane requiring ultimately the transformation to the discrete domain when implemented on the digital platform This method is commonly referred to as the emulation method The other method is to discretise the system model at the outset and then perform the design entirely using the discrete representation The initial controller will be designed using the emulation method and subsequently discretised to design and analyse the estimator and controller combination

In this chapter the problem of simulating a shock absorber using a rotary permanent magnet DC motor is analysed The test rig described in chapter 2 is represented using state space techniques and a suitable control method is chosen to provide adequate control over the states selected

3.2 System Description
The transfer function of the system, as given in 2 10, is now transformed to a state space format This involves combining the equations (3 1 to 3 3) to give a matrix representation of the open loop transfer function

$$\frac{\omega}{T_c} = \frac{k_i}{J_m s + B_m}$$ (3.1)

$$J_m \ddot{\theta} + B_m \dot{\theta} = k_i T_c$$ (3.2)

$$\ddot{\theta} = \frac{k_i T_c - B_m \dot{\theta}}{J_m}$$ (3.3)

In the general case the state space description of the continuous system can be given by

$$\dot{x} = Ax + Bu$$ (3.4)

where u is the control input to the system.

The output can be described as a linear combination of the state vector x.

$$y = Cx$$ (3.5)

In the servo case as described in chapter 2 the state space description is given by

(3.6)

$$A = \begin{vmatrix} 0 & 1 \\ 0 & -\frac{B_m}{J_m} \end{vmatrix}, \quad B = \begin{vmatrix} 0 \\ \frac{k_i}{J_m} \end{vmatrix}, \quad C = |1 \quad 0|$$

3.3 Control Law

With the above representation and assuming that all states are available, by measurement or otherwise, a common control law is to feedback a linear combination of the states which results in the control input [3-2].

(3.7)

$$u = -Kx = |k_1 \quad k_2| \begin{vmatrix} x_1 \\ x_2 \end{vmatrix}$$

In this control law there are two feedback gains since it is a second order system, i.e. there are enough degrees of freedom to select arbitrarily any desired set of root locations by proper choice of $k's$. Therefore substituting this feedback law into the system Eqn(3.7) yields Eqn(3.8);
where
x_1 and x_2 correspond to θ and $\dot{\theta}$ respectively

20

$$\dot{x} = Ax - BKx \qquad (3\;8)$$

Therefore

$$\dot{x} = (A - BK)x \qquad (3\;9)$$

and the characteristic equation of the closed loop system is

$$\det |sI - (A - BK)| = 0 \qquad (3\;10)$$

Evaluating Eqn(3 10) yields a 2^{nd} order polynomial in s containing the gains k_1 and k_2 The control design then consists of picking the gains K in order that the roots of Eqn(3 10) are in the desired locations

3 3 1 Pole Selection

The design of a control system is an attempt to meet a set of specifications which define the overall performance of the system in terms of measurable quantities, however there are performance indices that can be established on the basis of the 'best' response for a particular system

The advantages of using a single performance index are that it is chosen as a function of the variable system, and its power to clearly distinguish between the optimum and non-optimum systems There are a number of performance indices used in practice A usable performance index is the IAE (Integral of the absolute error) Using this index there can be a large initial error and therefore it is desirable to reduce the weighting of the initial error and to weight more heavily the smaller later errors

The performance index chosen for this application is the ITAE (Integral time absolute error) index

$$ITAE = \int_0^T t|e(t)|dt$$

This index will be implemented in the next chapter as it produces smaller overshoots, and hence oscillations, than other indices (IAE, (Integral of absolute error) and ISE, (Integral of the square of the error) This is due to the fact that the ITAE criterion gives the best selectivity and is computationally straightforward[3-3]

3 3 2 Pole Placement

The pole locations selected, it is now necessary to calculate the gains that are required to make up the matrix K A suitable method has been derived, using Ackermanns formula to calculate this K using the CAD package Matlab and the function *Acker*

If we assume the selected poles are given by

$$\alpha(s) = s^2 + \alpha_1 s + \alpha_2 = 0 \qquad (3\ 11)$$

Then Ackermanns formula yields

$$K = |0 \quad 1||A \quad AB|^{-1} \alpha(A) \qquad (3\ 12)$$

where

$$\alpha(A) = A^2 + \alpha_1 A + \alpha_2 \qquad (3\ 13)$$

3 4 Controller Structure

The controller given in Eqn(3 4) is for a regulator design, in that the goal is to drive all states to zero, i e position and velocity This however is not the desired action in most controller cases and more specifically in this case The motor, whose states are position and velocity, clearly requires a different type of controller This controller must provide the necessary input to drive the motor to a desired position This involves the introduction of a reference input into the regulator structure

3 4 1 Reference Input

Introducing a reference input, r, (defined below in Eqn(3 5)), requires the use of a state command matrix, i e shock absorber model The state matrix defines the desired value of state x_{ref}
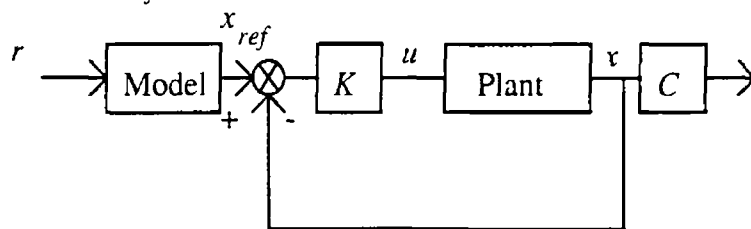

Figure 3.3 Full state feedback controller with reference input model

The model in question in Fig 3 3 refers to the model of a shock absorber being simulated and is designed to produce the reference input vector x_{ref}

If M_r is the Model

$$M_r r = x_{ref}$$ (3 14)

and

$$u = -K(x - x_{ref})$$ (3 15)

The servo system as described in Eqn(3 3) is a type 1 system so there is no steady state error and the final state

$$x(\infty) \cong x_{ss} = x_{ref}$$ (3 16)

where ss refers to the steady state value

3 5 Disturbance Rejection

A position and velocity fullstate feedback controller is the type being used in the test rig and therefore reference position and reference velocity are the only controlled inputs As described in the introduction to this chapter the input x_{ref} is obtained from the shock absorber model, however the input to this model r is the disturbance torque T_{dis} at the motor The effect of T_{dis} is unwanted other than as an input to the reference model and therefore must be rejected As a direct measurement of disturbance is not available a state estimator in the feedback path is used to estimate the value of disturbance If the disturbance can be found from the measured position it can also be used to cancel the actual disturbance and hence the system will behave as if no disturbance has occurred The control structure used to reject this disturbance is shown below in Fig 3 4
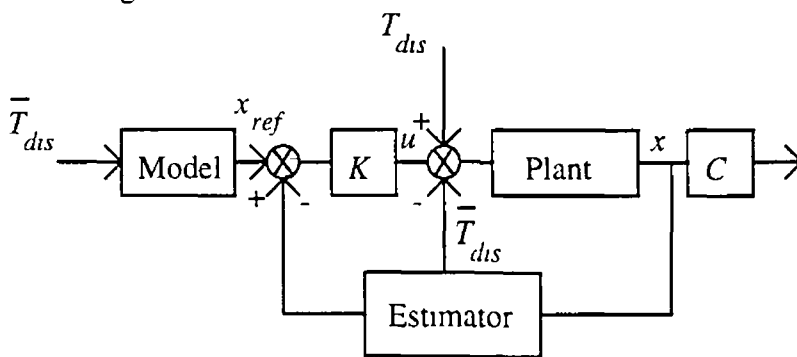

Figure 3.4 Overall controller structure

where $\overline{T}_{dis}$ is an estimate of torque distubance T_{dis}

3.5.1 Disturbance Analysis

The objectives of the controller structure in Fig 3 4 are to regulate the commanded position and velocity and to reject the effect of disturbance torque, T_{dis} To understand how to negate the effect of such disturbances, it is necessary to know how they affect the motor performance Taking the inner controller as the building block in this design, a simplified structure without the model will be used to analyse the effect of disturbances
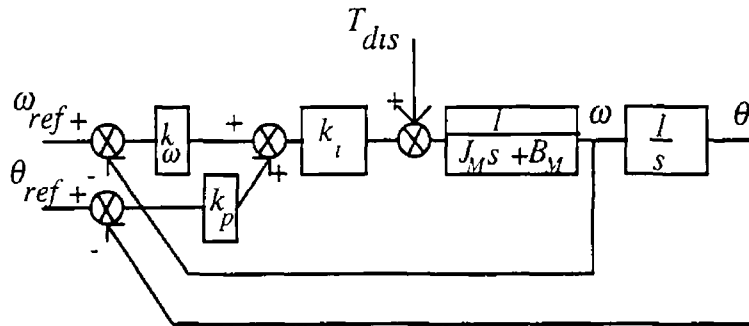
Figure 3.5 Full state feedback controller

3.6 Estimation

All estimation explanation and design will be carried out in discrete time All plant matrices will, for estmation analysis, be assumed to be discrete A, discretised, becomes Φ, B becomes Γ and C becomes H

The control structure presented thus far assumes that all state elements, position, velocity and disturbance are available directly for feedback Because typically and more specifically in this case, not all elements can be measured, the missing portion of the state vector must be reconstructed for use in the control law The methods required to obtain estimates of the entire state vector, given that a measurement of only one state element is available, will be analysed The missing state elements, and a smoothed value of the measured state, are obtained from estimators, designed using the discrete form of the system equations There are two basic type of estimates of the state vector $x(k)$, the predictor and the current estimate Using plant dynamics, Eqn(3 17), both types of estimates will be introduced

$$\bar{x}(k+1) = \Phi\bar{x}(k) + \Gamma u(k) \tag{3 17}$$

where k is a time step, $0,1,2 \quad n$

3 6.1 Prediction Estimator

Φ, Γ and *u(k)* are known and hence the estimator will work if the correct *x(0)* is chosen and $\bar{x}(0)$ is set to equal to it The block diagram representation of this open loop estimator is shown below Fig 3 6
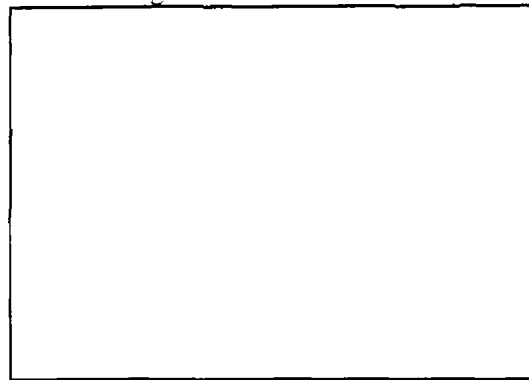
Figure 3 6 Open loop estimator

The error in the estimate is given by

$$\tilde{x}(k) \triangleq \bar{x}(k) - x(k) \tag{3 18}$$

If the initial value of $\bar{x}(0)$ is incorrect the dynamics of the estimate error are those of the plant For this system, which is asymptotically stable, the estimator running in open loop could produce bad estimates as it is not utilising any measurements from the system However, if the difference between the measured and estimated output is fed back and the model is constantly corrected with the error signal, the divergence will be minimised This produces the closed loop system as shown below in Fig 3 7
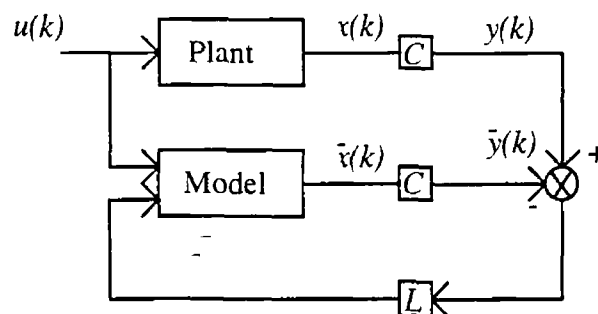
Figure 3 7 Closed loop estimator

The closed loop equation representing this estimator is given in Eqn(3 19)

$$\bar{x}(k+1) = A\bar{x}(k) + Bu(k) + L[y(k) - C\bar{x}(k)] \tag{3 19}$$

The type of estimator is the *prediction* type because a measurement at time k results in an estimate of the state valid at time $k+1$ The error dynamics (Assuming Φ and Γ are known) are described by

$$\tilde{x}(k+1) = (\Phi - LH)\tilde{x}(k) \qquad (3\ 20)$$

As in the case of pole placement, by defining the desired estimation pole locations, Ackermanns formula can be used to yield the L matrix

3 6 2 Current Estimator

For this project a current estimator is used as it has the very distinct advantage over its predictor counterpart in that the estimate is based on the most recent output measurement taken at time k The scheme for this estimator is given below in Fig 3 8
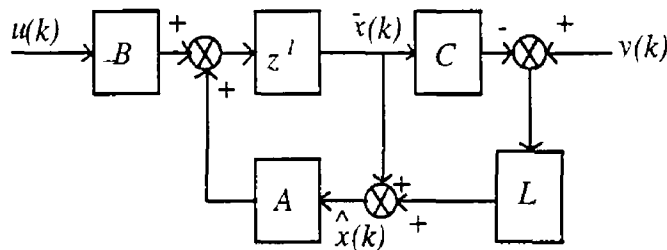
**Figure 3 8 Estimator block diagram**

$$\hat{x}(k) = \bar{x}(k) + L(y(k) - H\bar{x}(k)) \qquad (3\ 21)$$

where $\hat{x}(k)$ is the current estimate based on the current measurement $y(k)$ and $\bar{x}(k)$ is the predicted estimate based on the prediction technique described earlier i e

$$\bar{x}(k) = \Phi\hat{x}(k-1) + \Gamma u(k-1) \qquad (3\ 22)$$

Calculations that are not directly dependent on the $y(k)$ measurement can be performed before each sample instant to make this technique more efficient and accurate The estimator error equation for the current estimator is given below

$$\bar{x}(k+1) = [A - ALC]\bar{x}(k) \qquad (3\ 23)$$

3.7 Combined Control and Estimation

Although the controller and estimator roots are selected separately, the combination will have the same roots as if the two cases were designed together In other words

the complete system, as shown below in Fig 3 9, consists of the combination of the estimator and controller roots obtained assuming full state feedback
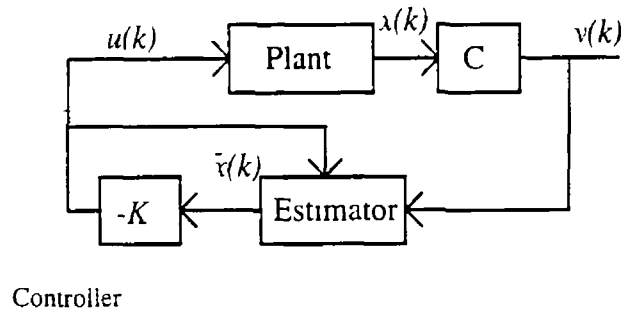
Controller

Figure 3 9 Combined estimator and controller

The combined Estimator Controller error equation can be represented by

$$\begin{vmatrix} \tilde{x}(k+1) \\ x(k+1) \end{vmatrix} = \begin{vmatrix} \Phi - \Phi LH & 0 \\ -\Gamma K & \Phi - \Gamma K \end{vmatrix} \begin{vmatrix} \tilde{x}(k) \\ x(k) \end{vmatrix} \tag{3 24}$$

The characteristic equation is given by

$$\begin{vmatrix} zI - \Phi + \Phi LH & 0 \\ 0 & zI - \Phi + \Gamma K \end{vmatrix} = 0 \tag{3 25}$$

3.7 1 Model Reference Controller

Disturbance Torque is estimated and the estimate if correct can be subtracted from the actual and the result is that the disturbance is rejected However as the disturbance torque T_{dis} is estimated it can be included in the control law as shown in Fig 3 10 Along with the estimate being used to reject it will also act as the input to the controller via a reference model The model subsequently produces desired positon and velocity inputs which are controlled by K See below in Fig 3 10
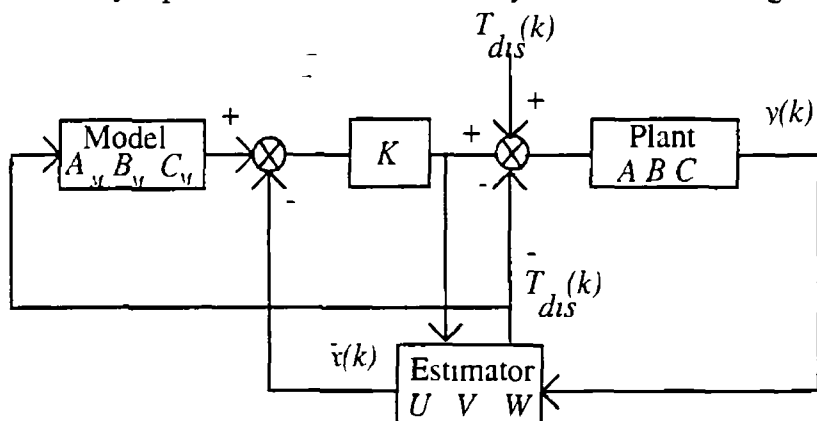
Figure 3.10 Model Reference controller

For this system the estimator model contains the plant and disturbance model To estimate the disturbance, we must provide a model of how it behaves If we assume that the disturbance is a bias or constant the model is

$$\begin{vmatrix} \bar{x}(k+1) \\ \overline{T}_{dis}(k+1) \end{vmatrix} = \begin{vmatrix} \Phi & \Gamma \\ 0 & 0 \end{vmatrix} \begin{vmatrix} \bar{x}(k) \\ \overline{T}_{dis}(k) \end{vmatrix} + \begin{vmatrix} \Gamma \\ 0 \end{vmatrix} u(k) \tag{3 26}$$

referring to Fig 3 10 where

$$U = \begin{vmatrix} \Phi & \Gamma \\ 0 & 0 \end{vmatrix}, \quad V = \begin{vmatrix} \Gamma \\ 0 \end{vmatrix} \text{ and } W = H \tag{3 27}$$

The estimation methods so far discussed can be used to reconstruct the state vector consisting of x and T_{dis} The only change is that the model is augmented This is referred to as *Bias estimation* This however does not affect the control gain K K is obtained using the un-augmented model as the system described by Eqn(3 26) is uncontrollable This is due to the fact that there is no control over the input T_{dis} However the disturbance will be used as an input to the reference model as shown in Fig 3 10

3 8 Discrete time Kalman Filter

As mentioned the current estimator is the more accurate of the two types of estimation techniques described Up until now the topics of measurement noise and model inaccuracies have been ignored These issues are very important in real time implementation and therefore must be catered for in the estimation procedure A well established stochastic method, taking noise and model errors into account, is Kalman estimation This method incorporates both state estimation and filtering to produce an accurate state matrix The Kalman filter is designed to solve the recursive estimation problem for discrete dynamical systems[3-4] where

$$x(k+1) = \Phi x(k) + \Gamma u(k) + Gw(k) \tag{3 28}$$

$$y(k) = Hx(k) + v(k) \tag{3 29}$$

where $w(k)$ is the process noise and $v(k)$ is the measurement noise

$$x(k) \in R^r, u(k) \in R^m, y(k) \in R^p, u(k) \in R^l \tag{3 30}$$

28

In the real world, observed random phenomena are not describable by independent random variables[3-5] The statistical dependence (correlation) between random systems at different times can usually be described by a dynamic system excited by a random source (Gaussian noise) For this reason statistically dependent or correlated data can be thought of as a dynamic system with an uncorrelated Gaussian noise as an input

Let $w(k)$ have a Gaussian distribution with a zero mean and covariance Q, and let the measurement noise $v(k)$ also be a Gaussian mean with covariance R

$$\text{i e } w=N(0,Q) \text{ and } v=N(0,R)$$

The linear dynamic system described above will preserve the Gaussian nature of the input signal The output of the linear dynamic system will have the same statistical properties of the noise source Thus the output of the linear dynamic system has a Gaussian distribution and a covariance not yet fully defined To find the statistical properties at the output of the system, the noise's statistical properties are propagated through the system The mean at the output of the system is the mean of the system equation given above Thus due to Gaussian distribution and zero mean this reduces to,

$$\bar{x}(k+1) = \Phi\bar{x}(k) + \Gamma\bar{u}(k) \tag{3 31}$$

The mean propagates exactly in accordance with the dynamics of the system

Note: For clarity here dependencies will be removed temporarily

i e

$$x(k) \to x, \quad \bar{x}(k) \to \bar{x} \text{ and } w(k) \to w$$

To find the propagated error covariance of the system, where the covariance P is defined as

$$P(k+1) = \overline{[(x(k+1) - \bar{x}(k+1))(x(k+1) - \bar{x}(k+1))]^T} \tag{3 32}$$

by substitution

$$K(k+1) = P(k)H^T(HP(k)H^T + R)^{-1} \tag{3 33}$$

$$P(k+1) = \overline{(\Phi(x-\bar{x})(x-\bar{x})^T\Phi^T + Gww^TG^T)} \tag{3 34}$$

but because

$$P(k) = \overline{((x-\bar{x})(x-\bar{x})^T)} \tag{3 35}$$

29

and w has a noise covariance of Q, Eqn(3 34) reduces to

$$P(k+1) = \Phi P(k) x \Phi^T + GQG^T \qquad (3\ 36)$$

This is known as Lyapunov's equation and represents the time update equation of the Kalman Filter Now reverting back to original notation
Measurement Update Equation

$$\bar{x}(k+1) = \bar{x}(k) + K(k+1)(y(k) - H\bar{x}(k)) \qquad (3\ 37)$$

Error Covariance

$$P(k+1) = (I - K(k+1)A)P(k) \qquad (3\ 38)$$

$$P(k) = (\Phi P(k)^{-1}\Phi^T + GQG^T) \qquad (3\ 39)$$

Kalman Gain

$$K(k+1) = P(k)H^T(HP(k)H^T + R)^{-1} \qquad (3\ 40)$$

3 8 1 Estimator Pole Location

Estimator root location selection is a similar type of problem to control root selection, however the design trade-off is somewhat different Fast roots in an estimator do not carry the penalties that they do in the control case as the signal is digital The only penalty associated with fast estimator roots is that they create an increased sensitivity between the sensor errors and estimation errors

The main aspect of estimator root selection is that estimator errors should be minimised with respect to the prevailing system disturbances and sensor noise It is also recommended to keep the estimator roots faster than the control roots in order that the total system response is dominated by the control roots Typically roots are selected that are 2 to 6 times faster than the control roots

In estimation the vital statistic is the ratio of the plant model errors to the sensor errors For an accurate plant model with small disturbances but large sensor errors, low esimator gains (slow roots) are used A system, however, with a plant model with

large disturbances but an accurate sensor achieves the best estimation by using large estimator gains (fast roots) in order to use the measured information to correct model errors as quickly as possible

3 9 Full State Space Description

The Kalman filter estimates position, velocity and disturbance torque The plant model used by the Kalman filter is the augmented model given in Eqn(3 24) By using this equation and the augmented control law

$$u(k) = [1\,K]\tilde{x}(k+1) \tag{3 41}$$

Using Eqn(3 26) and (3 39), the full state feedback controller, a state space description of the entire closed loop system can be derived

$$\begin{vmatrix} \tilde{x}(k+1) \\ x(k+1) \end{vmatrix} = \begin{vmatrix} \Phi - \Gamma[1\,K] - K_{ss}H\Phi + K_{ss}\Gamma[1\,K] & K_{ss}H \\ -\Gamma[1\,K] & \Phi \end{vmatrix} \begin{vmatrix} \tilde{x}(k) \\ x(k) \end{vmatrix} \tag{3 42}$$

Summary

This chapter has formalised the issues involved in the control of a permanent magnet DC motor to simulate a shock absorber Estimation techniques were introduced and a method of disturbance rejection using bias estimation was described with this application in mind The pole locations of both the controller and estimation were shown to be vital to the performance of the system ITAE is the criterion chosen for the pole location selection in the controller case and, due to the fact that a digital position sensor is used, the estimator poles can be placed in locations that will be faster than the controller poles This is desirable in most cases

CHAPTER 4

COUPLING SIZING

4 1 Introduction

In electrical motor applications, the system designer is increasingly challenged to improve system performance and reduce system cost. The main challenge is to optimise a servo system. The designer must have the ability to select an optimum electrical actuator for a specific application[4-1]

In the design of an electrical shock absorber the main problem consists in setting up the power of the single components of an electrical drive in order to obtain the desired performance of the shock absorber system without over-dimensioning the drive [4-2] A motor drive consists of an electrical motor, a power electronic converter and a position sensor. In this chapter criteria for optimum match between the mechanical load and drive components are discussed

The selection of the actuator for a particular application requires that the actuator's performance capabilities be as exactly defined as possible. In the type of servo system used in this application the figure of merits called Power Rate and Heat Rate will be used to represent the most important electrical and mechanical qualities of a motor. Both methods will also be used to investigate the requirements of an actuator used in an electrical shock absorber simulation set-up

The aim of this chapter is to derive the value of *a* (coupling ratio), modelled in chapter 2, Eqns(2 5 and 2 6) in order to complete the reference model used in the controller block diagram Fig 3 3

4.2 Motor and Load

Prior to selection of the motor drive components, the load requirements must be accurately available, i e load inertia maximum speed, speed range and direction of motion As mentioned in the introduction, cornering forces are the disturbances of interest in this thesis As an example of the forces experienced on a Formula 1 front suspension shock absorber, a sample set of cornering displacement profiles are selected from data taken from a set This set was taken from a Formula 1 Grand Prix track The measurements were made using a displacement sensor placed on the front suspension of a 500cc motor cycle shock absorber throughout a trial set of laps[4-3] This data or more specifically the section of this data that contains the cornering data, is analysed to produce a sample velocity profile as shown below
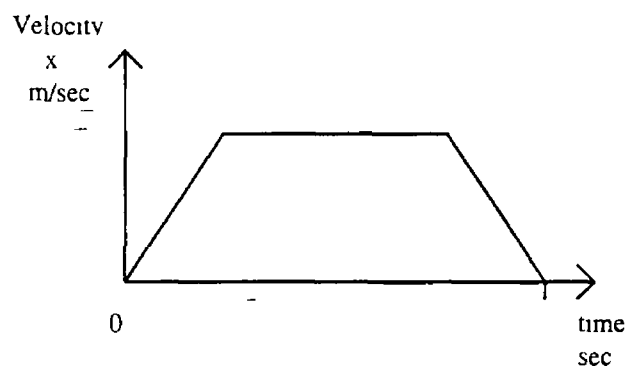
Figure 4 1 Velocity profile

Using the model described in chapter 2 this velocity profile can be differentiated to produce acceleration and load(disturbance)-force profiles, i e α and F_{dis} These profiles will be used with the Power Rate method to obtain a gearing ratio to match the test rig motor with the load force By analysing the position profile above, this sizing problem can be tackled as an incremental motion problem and thus the Power Rate method may be applied

4.3 Power Rate

Power Rate is defined as the rate of change of mechanical power with respect to time

$$Power\,Rate = \frac{dP_m}{dt} kW \,/\,\sec$$

(4 1)

where P_m is the output power of the actuator

$$\dot{P}_m = \frac{dP_m}{dt} = \frac{dT_\omega}{dt}$$

$$= T\frac{T}{J_m} = \frac{T^2}{J_m} kW / \sec$$

where T is the motor torque ω is the rotor velocity, and J_m is the rotor inertia

Power Rate is an actuator's ability to accept and transduce power with respect to time, to accept electrical input power and to produce usable mechanical power The larger the power rate of an actuator, the faster the usable mechanical power is available to provide energy for motion

There is a limit to the power rate available from any actuator, i e no motor has infinite power capability as shown in the diagram below
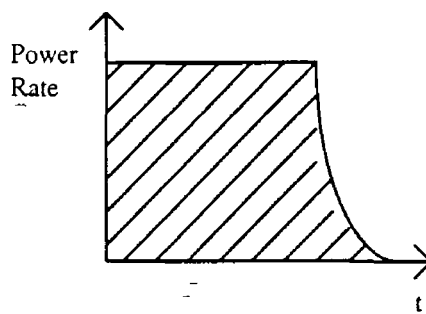
Figure 4 2 Power curve

The area under the curve is exactly equal to the quantity of power available to move the motor and load within a time t The curve tails off because no motor has infinite power capability When the maximum power of the device is reached, or the maximum power required by the application is achieved, the power acceleration stops

$$PowerRate = \dot{P}_m = \frac{T^2}{J_m} = \frac{dP_m}{dt} = \frac{d^2(Energy)}{dt^2}$$

Manipulating the above expression we can see that power rate contains some important motor parameters

$$\frac{T^2}{J_m} = \frac{K_T^2 I_A^2}{J_m} = \frac{I_A^2}{\dfrac{J_m}{K_T^2}} = \frac{I_A^2 R_A}{\dfrac{J_m R_A}{K_T^2}} = \frac{I_A^2 R_A}{\tau_m}$$

$$\frac{T^2}{J_m} = \frac{(I_A{}^2 R_A \text{ Losses Allowable})}{(\text{Mechanical Time Constant})} \tag{4 4}$$

where I_A equals the armature current, R_A equals the armature resistance, K_T is the motor torque constant and τ_m is the motor mechanical time constant This shows that a motor with a high thermal capacity (High Allowable I^2R losses) and low mechanical time constant will have a high power rate

4.3.1 Actuator Power Rate

It is necessary in this analysis to introduce a sample application of a motor drive to deduce a usable general mathematical relationship Concentrating on the acceleration and deceleration stages of the curve in Fig 4 1, the current, velocity and position profile for theses areas, are as shown in Fig 4 3

Figure 4 3 Non-constant velocity section of profile
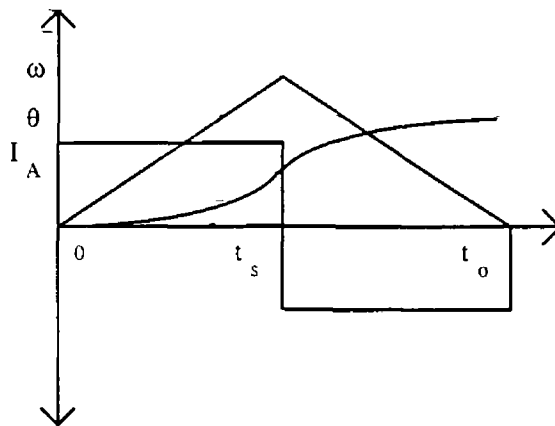
The friction of the load can be normalised by dividing it by the total torque available from the motor The normalised torque then equals the percentage of the total torque of which the load friction consists of Defining B as the normalised friction torque

$$B = \frac{T_f}{K_T I_A} \tag{4 5}$$

Integrating the acceleration calculates, t_s, the time to maximum velocity

35

$$\int_0^{t_r} \frac{(1-B)K_T I_A}{J_m + J_L} dt - \int_{t_r}^{t_o} \frac{(1+B)K_T I_A}{J_m + J_L} dt = 0 \tag{4.6}$$

$$\therefore t_s = (1+B)\frac{t_o}{2}$$

The total rotor displacement can be solved by knowing that displacement is 1/2 velocity multiplied by time

$$\theta = \frac{\omega_{max} t_o}{2}$$

and that velocity can be found by integrating the acceleration

$$\theta = \frac{t_o}{2} \int_0^{t_r} \frac{(1-B)K_T I_A}{J_m + J_L} dt = \frac{(1-B^2)K_T I_A t_o^2}{4(J_m + J_L)} \tag{4.7}$$

If $I^2 R_A$ is substituted by P

$$P = 16 \frac{R_A J_m}{K_T^2} \frac{(1+\frac{J_L}{J_m})^2}{\frac{J_L}{J_m}} \frac{J_L \theta^2}{\tau_o^4} \left| \frac{1}{1-B^2} \right|^2 \tag{4.8}$$

Rearranging terms a significant relationship is arrived at:

$$\text{Required Power Rate} = \frac{P_m}{\tau_m} = C J_L \alpha^2 K G \tag{4.9}$$

where

$$K = \frac{(1+\frac{J_L}{J_m})^2}{4\frac{J_L}{J_m}} \text{ and } G = \frac{1}{(1-B^2)^2} \tag{4.10}$$

The left side of the equation contains the $I^2 R_A$ losses divided by the mechanical time constant. As proven before from Eqn(4.4), this is equal to the Power Rate of the motor. In terms of this analysis it will be referred to as the motor Power Rate required by the application. The right side of the equation contains two parameters which directly effect the required motor Power Rate, K and G.

The parameter K is a function of inertia match between the motor and the reflected inertia of the load. The parameter G is a function of the normalised friction torque.

The plot below of K versus the inertia ratio reveals how the required motor Power Rate is affected as the inertias are mismatched
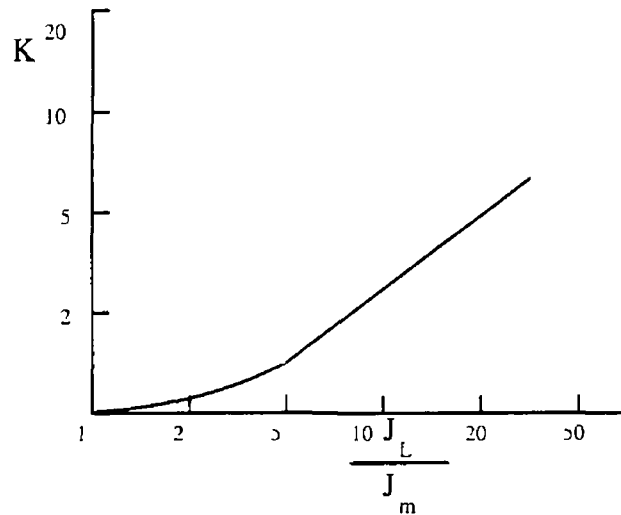
Figure 4 4 K v Inertia ratio

This plot shows that even if inertias of motor and load are mismatched to a factor of 5 the required motor Power Rate does not increase by a great deal However mismatching the inertias any more than this can have adverse effects on the Power Rate

4.3.2 Load Power Rate

So far the actuator power rate has been considered In this system the power rate of the motor will be matched to the power rate requirements of the application The power rate of the motor equals its torque capability squared, divided by the motor's own rotor inertia[4-4] Likewise the power rate which applies to the load consists of the load torque squared, divided by the load inertia

$$\dot{P}_L = (J_L \alpha_L + T_L)\alpha_L \qquad (4\ 11)$$

$$\dot{P}_L = \frac{T_L^2}{J_L} \qquad (4\ 12)$$

where α_L is the required load acceleration J_L is the load inertia and T_L is the load torque

There is a direct relationship between load power rate and motor power rate

37

$$\dot{P}_m = C\dot{P}_L \qquad\qquad (4\ 13)$$

$$\dot{P}_m = (\frac{1}{R} + R)^2 \dot{P}_L \qquad\qquad (4\ 14)$$

where R is the normalised gear ratio

$$R = \frac{R_g}{R_{go}} \qquad\qquad (4\ 15)$$

where R_g equals the ratio between actuator and load R_{go} equals

$$R_{go} = \sqrt{\frac{J_L \alpha_L + T_L}{J_m \alpha_L}} \qquad\qquad (4\ 16)$$

If inertias are matched, R equals 1 and any other value for R will increase the required motor power rate

Therefore the minimum relationship between motor power rate and load power rate is

$$\dot{P}_m = 4\dot{P}_L \qquad\qquad (4\ 17)$$

To sum up, the concept of power rate can be implemented as follows

- Compute the required load power rate

- Compute the required motor power

- Assume matched inertias and use Eqn(4 16) as the first iteration for computing the required motor power rate

- Using the required motor power rate and the required power, select a motor

- Solve for the required ratio that will match inertias

- Using this ratio, check that dynamical constraints are not exceeded

The above method of finding the coupling ratio, due to the fact that it ignores thermal aspects, sometimes fails to be accurate For this reason the concept of heat rate is

introduced to ensure that the motor avoids overheating during a predefined duty cycle.

4.4 Heat Rate

In DC motors the main contribution to heating is defined as "copper loss" or I^2R_A losses, L, related to rms torque[4-5]. In the previous sections, the match between motor and load was studied. This analysis established the peak torque and the maximum speed required from the motor. L however is calculated from the motor torque-profile[4-6].

$$L \equiv I^2 R = \frac{\tilde{T}^2}{K_T^2} R \qquad (4.18)$$

where ~ denotes rms value. Hence for the proper choice of motor, the condition

$$L_m \geq \frac{\tilde{T}^2}{K_T^2} R \qquad (4.19)$$

should be satisfied , where L_m is the maximum heat power that the motor can safely dissipate in steady state. L_m depends on the motor size and construction, ambient temperature and method of cooling. This parameter is specified by the motor manufacturer. See Appendix A.

For an ideal DC motor as shown in chapter 2, Power Rate and Heat Rate are given by the equations

$$\dot{P}_m \cong \frac{T_R^2}{J_m} = \frac{R I_p^2}{\tau_m} \qquad (4.20)$$

$$\dot{Q}_m \cong \frac{L_m}{\tau_m} = \frac{R \tilde{I}_{max}^2}{\tau_m} \qquad (4.21)$$

where I_p is the motor peak-current and I_{max} is the maximum (~) rms current permissible from heating considerations. Hence the constraint below must be adhered to

$$\dot{Q}_m > \dot{P}_m \qquad (4.22)$$

4.5 Sample Cornering Profile

The displacement profile below is chosen from the set of data referred to earlier in
this
chapter
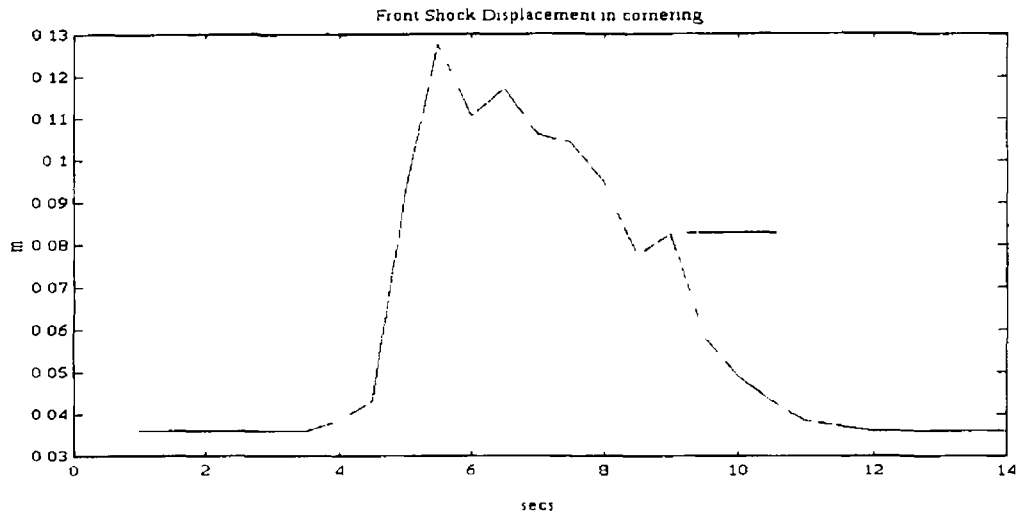
Figure 4 5 Cornering displacement profile

Displacement data can be used assuming an accurate model derived in chapter 2 for
the shock absorber Eqn(2 1) Acceleration, peak velocity and the magnitude and
direction of the force disturbance on the shock throughout this cornering can be
easily derived giving a max force (F_{dis}) of 1500N

The velocity profile below in Fig 4 6 is derived from the position (Displacement)
profile Fig 4 5 A worst case section of the profile is chosen i e the area of the curve
that produced the greatest acceleration on the suspension
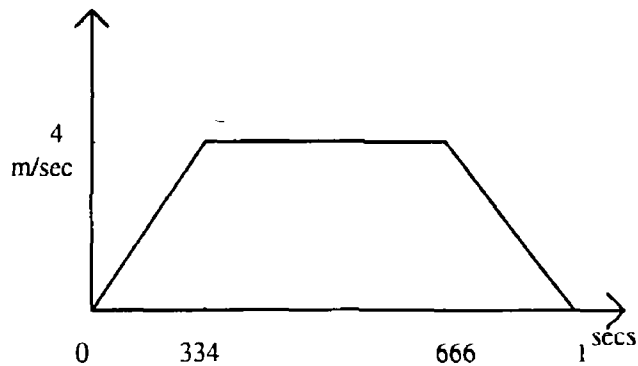
Figure 4 6 Velocity profile

1. Finding the load power rate

Total distance travelled

$$s = 0\ 2(0\ 334) + 0\ 4(0\ 332) = 0\ 13m$$

Load power rate

$$\dot{P}_L = (M_S\alpha + F_L)\alpha$$

$$\dot{P}_L = 900\,W/s$$

2 Solving for required motor power

Required motor power = power to drive load + power to drive rotor
Required load power = load power rate × time period

$$P_L = \dot{P}_L\,0\ 334$$

$$P_L = 300W$$

When the inertias are matched the motor power required to accelerate the motor's own inertia will equal the required load power

$$\text{Required motor power} = 600W$$

3. Motor power rate

Using Eqn(4 17)

$$\dot{P}_m = 4 \times \dot{P}_L$$

$$= 3600W/s$$

4 Motor Selection

The motors available and being used in the test rig, described in chapter 2 , are used in this calculation and the specifications are given in Appendix A

5. Gear Ratio

$$R_g = \frac{1}{\alpha}\sqrt{\frac{\dot{P_L}}{J_m}}$$

therefore

For inertia of $0\ 0059kg/m^2$

$$R_g = 652\ rad/m$$

coupling ratio $a=0\ 0015$ and pitch $p=9\ 4\ mm/rev$

Inertias are therefore matched if we use a lead screw with a pitch of $9\ 4\ mm/rev$

6 Check Velocity Parameter

Peak Velocity

$$\omega_p = \frac{\dot{x} \times R_g \times 60}{2\pi}rpm$$

$$=1245rpm$$

where x is linear distance travelled in meters

The peak velocity calculated above can be supplied by the motor in question and in simulation and tests a gear ratio of $a=0\ 0015m/rad$ will be used

Heat rate

The temperature constraints must be considered therefore the power loss must be calculated in order to find the thermal power loss Power losses are attributable to three major factors

1 PWM core loss

2 Losses proportional to ω^2

3 I^2R losses

The conventional simplifying assumption is that the motor thermal behaviour can be approximated by a first order differential Eqn(4 23)

$$\frac{1}{R_\theta}\Delta\theta + C\dot{\Delta\theta} = L \qquad\qquad (4\ 23)$$

42

where R_θ is the thermal resistance (from windings to ambience), C is the heat capacity, and $\Delta\theta$ is the winding-ambience temperature difference The equation can be written as Eqn (4 24)

$$\Delta\theta + \tau_\theta \, \dot{\Delta\theta} = R_\theta L \qquad (4\ 24)$$

where $\tau_\theta = CR_\theta$ is the thermal time constant of the motor In steady state $\Delta\theta = R_\theta L$ and at extreme permissible conditions,

$$\Delta\theta_{max} \approx R_\theta L_m \qquad (4\ 25)$$

Thus the motor heat rate $\dot{Q}_m$ can be expressed as

$$\dot{Q}_m \approx \frac{\Delta\theta_{max}}{R_\theta \tau_m} \qquad (4\ 26)$$

where

$$\tau_m = \frac{J_m R_A}{K_B^2}$$

and

$$R_\theta = \frac{100 - 40}{65} = 0\ 93^\circ C/W$$

where we assume the thermal resistance R_θ is a constant, insulation temperature is 100° and ambient temperature is 40°

Using Eqn(4 26)

$$\dot{Q}_m = 10\ 3 \times 10^3\ W/s$$

This condition satisfies Eqn (4 22)

Summary

The power rate procedure in this chapter enables the designer to take a methodical approach to finding the suitable coupling rate that will be used in the shock absorber model The power rate and heat rate techniques are suitable for a specific application A worst case cornering profile is used and the duty cycle is assumed to be low, i e less than 50% If the designer obtains different data at a later stage this method of choosing the actuator and coupling rate will remain valid

CHAPTER 5

CONTROL DESIGN

5 1 Introduction

In this chapter the control approach introduced in chapter 3 is studied in more practical terms, i e more specific to the design problem The controller is realised, suitable pole locations are found and a simulated step response of the closed loop ITAE controller is shown The disturbance rejection properties are also studied and simulated in continuous time

A sampling period is chosen using well established techniques A discrete simulation is then carried out to verify the design techniques chosen As estimation is a vital part of this control design the choice of gains is discussed and issues such as noise and errors are introduced as they effect the accuracy and speed of estimation Finally the full controller and estimator is simulated and the relevant responses are plotted

5 2 Full state Feedback Controller
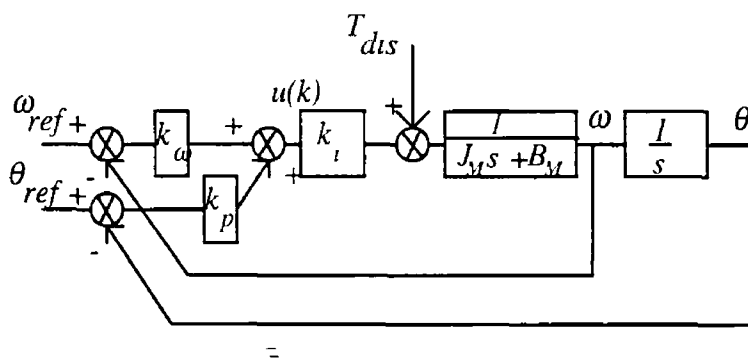

Fig.5 1 Full state feedback controller

Fig 5 1 is the controller set-up used for this simulation application The feedback controller has two reference inputs ω_{ref} and θ_{ref} Imposing the constraint on θ_{ref} namely $\theta_{ref}=\omega_{ref}/s$, the above controller has the same transfer function as a 'standard' proportional and integral (PI) controller

Analysing the system in Fig 5 1

$$u = (\omega_{ref} - \omega)k_{\omega} + (\theta_{ref} - \theta)k_p$$

(5 1)

i e

$$u = (k_{\omega}s + k_p)(\theta_{ref} - \theta)$$

(5 2)

therefore

$$u = \frac{\theta(J_m s^2 + B_m s)}{k_i}$$

(5 3)

The closed loop transfer function for the above system viewed as a position servo is given by

$$\frac{\theta}{\theta_{ref}} = \frac{k_i k_{\omega} s^2 + k_p k_i}{J_m s^2 + (B_m + k_{\omega} k_i)s + k_p k_i}$$

(5 4)

5 2 1 Controller Pole Locations

As described in chapter 3, the controller closed loop pole locations are calculated to achieve an optimum response to typical test inputs The overshoot, rise time and settling time are all to be optimised in relation to one another The method of pole location determination used is the ITAE method described in chapter 3 The standard optimum forms of ITAE transfer functions can be described by the closed loop transfer function described below

$$T(s) = \frac{a_{n-1}s + a_n}{s^n + a_1 s^{n-1} + \quad + a_{n-1}s + a_n}$$

(5 5)

This equation is a very close approximation to the closed loop transfer function of the system Eqn(5 4) From the paper-Graham and Lathrop[5-1], the optimal form of the characteristic equation for this second order system is given by

$$s^2 + 3 2\omega_n s + \omega_n^2 = 0$$

(5 6)

where ω_n is the desired closed loop natural frequency

A desired closed loop natural frequency of 200 rad/sec is chosen This bandwidth is chosen due to the fact that using the current loop, as described in chapter 2, an upper

limiting constraint on the speed of controller is in place With this in mind the denominator of the desired closed loop transfer function is

$$s^2 + 620s + 40000$$

This transfer function has closed loop poles at

$$s=-70\ 2\ rad/sec,$$
$$s=-569\ 8\ rad/sec$$

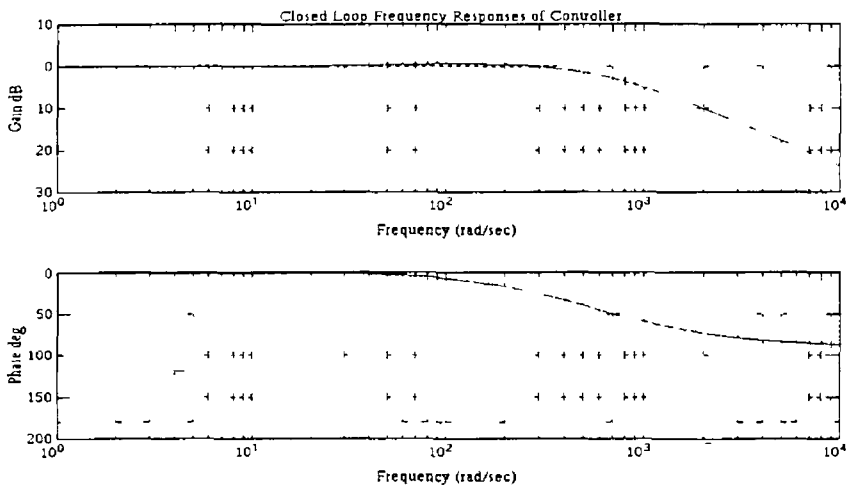Figure 5 2 Frequency response of ITAE closed loop controller

Using Ackermanns formula, or in this case simple direct comparison of the coefficients of the closed loop characteristic equations with the standard forms, the following gains are found

$$kp=4\ 62\ Nm/rad\ and\ k_\omega=\ 0707Nm/rad/sec$$

The simulated step response of the controller is shown below
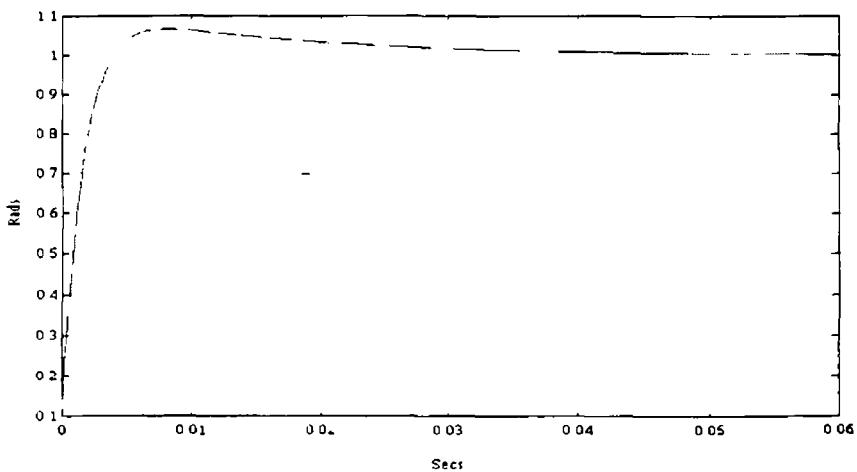
Figure 5.3 Step response of ITAE closed loop controller

5 3 Disturbance Rejection

The controller structure above regulates the commanded position and velocity No account for the disturbance input T_{dis} is made The relationship between disturbance torque and position is given by the transfer function below Eqn(5 7)

$$\frac{\theta}{T_{dis}} = \frac{1}{J_M s^2 + (B_M + k_\omega k_i)s + k_p k_i}$$ (5 7)

The relationship θ/T_{dis} is used as a figure of merit for the disturbance rejection properties of the controller As can be seen in the frequency response of the transfer function, θ/T_{dis}, as the disturbance increases so does this figure At low frequencies, i e inside the bandwidth of the controller this figure is constant It is given by kpk_i and is proportional to the square of the closed loop bandwidth of the controller

Figure 5 4 Frequency response of ITAE controller to disturbance

Figure 5 5 Step response of full state feedback controller

An integrator is an alternative method of providing this disturbance rejection action [5-2] The response of this integrator system given in Eqn(5 8) will be of third order

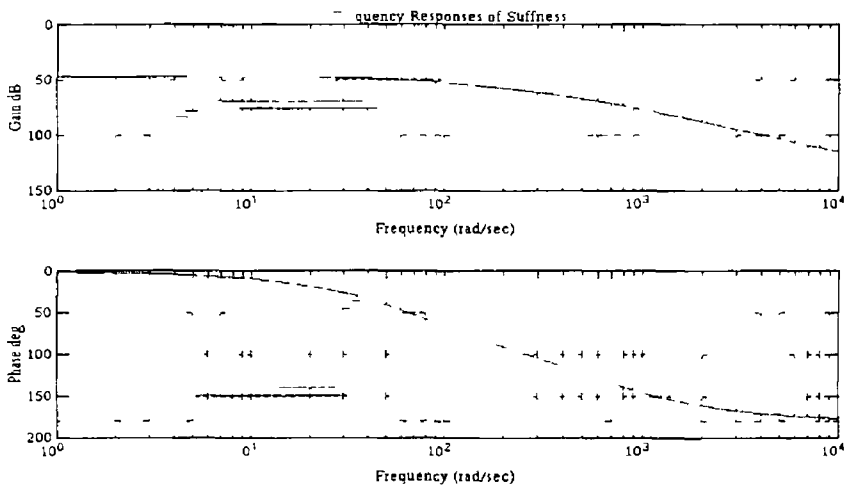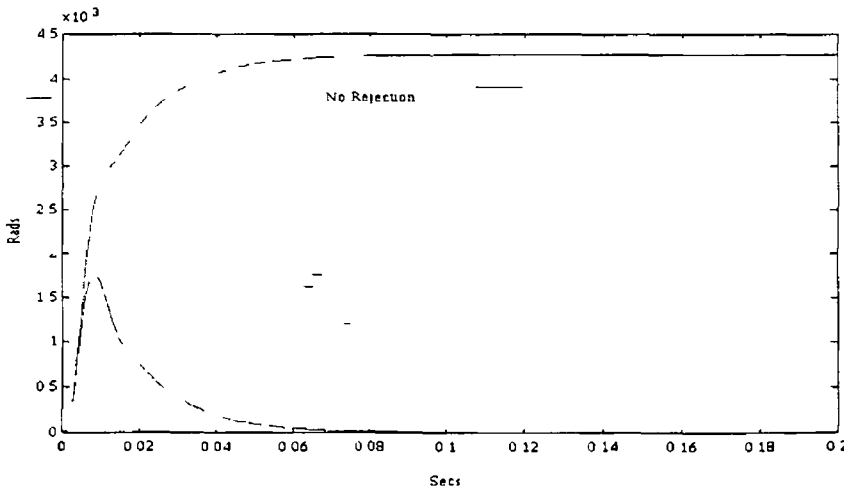$$\frac{\theta}{\theta_{ref}} = \frac{k_I k_\omega s^2 + k_p k_I s + k_I}{J_M s^3 + (B_M + k_\omega k_I) s^2 + k_p k_I s + k_I} \qquad (5\ 8)$$

where k_I is the integral gain

In order to maintain the same closed loop bandwidth the feedback gains for the integral controller must be increased This causes increased sensitivity in the estimator and can be avoided by use of disturbance rejection The full state feedback controller produces responses with less overshoot than the integrator controller while maintaining the desired disturbance rejection properties[5-3]

5 4 Selection of Controller Parameters

5 4 1 The Sampling Frequency

In digital control system analogue signals are sampled every sampling period T If the sample frequency is sufficiently high compared with the highest-frequency component involved in the continuous-time signal, the characteristics of the continuous-time signal may be preserved in the envelope of the sampled signal In order to reconstruct the digital signal from a sampled signal, there is a certain minimum frequency that the sampling operation must satisfy

$$\frac{f_s}{f_b} \geq 2 \qquad (5\ 9)$$

Where f_s is the sampling frequency and f_b is the closed loop bandwidth of the system

This provides a limit to a designer below which aliasing will occur and the system may become unstable or considerably slower than specified This relationship does not require that the sample frequency be twice as fast as all the open-loop poles of the plant However, it must be chosen to be at least twice as fast as the poles being controlled, i e the dominant poles In fact in design terms, the desired sampling multiple is Eqn(5 10), as suggested by Franklin and Powell[5-4]

$$6 \leq \frac{f_s}{f_b} \leq 40 \qquad (5\ 10)$$

This constraint will produce a smooth time response which is very desirable in this servo motor case[5-5] The degree of smoothness that is available is dependent on the speed of sampling, which is dependent on the controller being used In this case the

48

card being used (TMS320C30) has a fast processor that will allow up to 10kHz sampling rate The working closed loop bandwidth of the servo ≈ 200Hz, therefore a sample rate of 5kHz is chosen as representing the best sampling rate

For a system with random inputs such as this system it is vital that the controller is aware of the input without having a large delay In this case, where only low frequency disturbances are the considered inputs the sample rate selected is most certainly adequate

5 5 Digital Design

The simplest method of obtaining the digital controller from the analogue one designed above, is to transform the controller gains using the z-transform identity in Eqn(5 11)
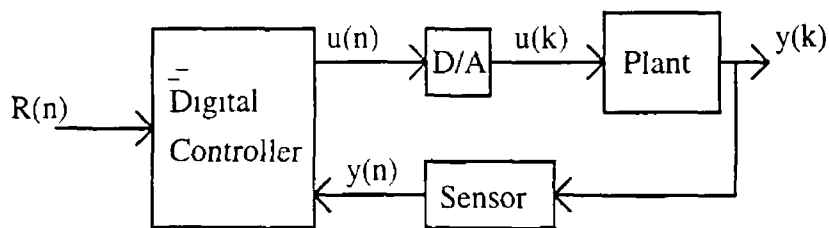
Figure 5 6 Digital control system

$$z = e^{sT} \qquad (5\ 11)$$

$$s = -70\ 2\ and\ s = -569\ 8$$

becomes

$$z = -0\ 9861\ and\ z = 0\ 8923$$

The step response for this discrete system is shown in Fig 5 7
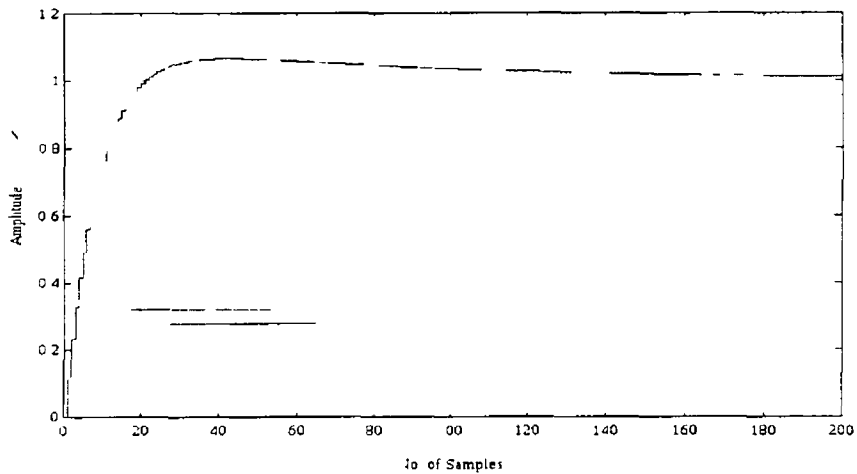
Figure 5 7 Discrete step response of ITAE closed loop controller

5 6 Estimator Design

The pole locations and hence the speed of the estimator must be chosen to obtain the best compromise between the accuracy and the speed of estimation of the state matrix As described in chapter 3, the ratio of R and Q determines the pole locations of the estimator
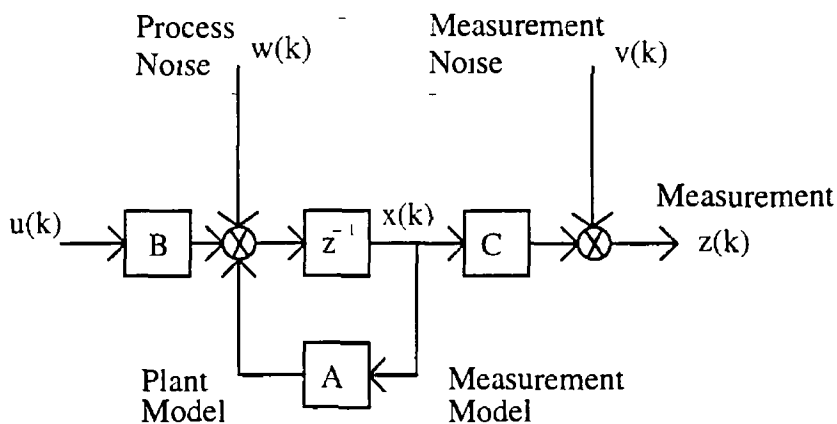
Figure 5.8 Plant and measurement with additive noise

To the plant is added process noise $w(k)$ and to the measurement noise $v(k)$ Using the discrete Kalman filter as described in chapter 3 the error in the estimate can be forced to zero even when there is noise present[5-6] The requirement of the estimator is that the gain K is chosen based on the stochastic considerations described in the previous chapter The resulting state estimator depends on the solution of the Ricatti equation

$$K(k+1) = P(k)H^T (HP(k)H^T + R)^{-1}$$

50

$P(k)$ is found using the time varying equations derived in chapter 3

5.6.1 Steady State Kalman Estimator

The estimator gains in the time varying estimator described in chapter 3 and represented by K above will eventually reach a steady state if enough time passes This is true due to the fact that P reaches a steady state value It is often desirable to determine the constant gain matrix to allow simplification of the controller without significant degradation in the performance of the estimate

The steady state Kalman estimator is identical in structure to the time varying one described in chapter 3 The only difference is that the gain, K, is determined so that the estimation errors are minimised *assuming* levels of process and measurement noise

$$K_{ss} = P_{ss}C^T(CP_{ss}C^T + R)^{-1} \tag{5 12}$$

The steady state Kalman equations can be written as

$$\bar{x}(k+1) = (A - K_{ss}C)\bar{x}(k) + K_{ss}y(k) \tag{5 13}$$

$$y(k) = C\bar{x}(k) \tag{5 14}$$

The main design problem associated with the estimator is to achieve a balance between accuracy, stability and speed of estimation In order to choose estimator pole locations as mentioned in chapter 3, a choice must be made as to how accurate the estimator must be, given the conditions under which it must run The process and measurement noise magnitudes are defined by the covariance matrices Q and R and the ratio of these two values determines the speed or pole locations of the estimator In the steady state case, it is necessary that good knowledge of the noise content contained in the measurement and the process are available to the designer

The vital statistic is the ratio of the plant model errors to the sensor errors For an accurate plant model with small disturbances but large sensor errors, low esimator gains (slow roots) are used A system. however, with a plant model with large disturbances but an accurate sensor achieves the best estimation by using large estimator gains (fast roots) in order to use the measured information to correct model errors as quickly as possible

5.6.2 Measurement Noise

In the case of measurement or sensor noise it is often possible to find the magnitude of the noise component from a manufacturer's specification sheets and mathematically this value can be squared to produce the value of R For this application, an accurate low noise digital encoder is used to provide the measurement information and therefore implies that there will be negligible noise effect from the sensor

Also in this case the sampling rate is high and the estimation accuracy improves as sample rate increases The digital encoder produces a signal with a resolution of 15 bits This high resolution greatly reduces the risk of high frequency noise

5.6 3 Process Noise

Plant inaccuracies are accounted for in the choice of Q which is more abstract and difficult to quantify than the process noise described above Plant inaccuracies and unknown disturbances in the plant model cause inaccuracies in the estimate of the state vector The selection of Q is also important as it must ensure that all elements of the state vector are sufficiently weighted throughout the sampling period

Q determines indirectly the pole locations of the closed loop estimator for the designer It is therefore a tradeoff selection as described above The estimator can be driven very quickly and therefore the constraint of having the estimator poles at least twice as fast as the controller closed loop poles can easily be met K_{SS} the steady state Kalman gain can be obtained by solving Eqn(5 12) This can be solved empirically through iteration, or by using numerically stable techniques[5-7] MATLAB® provides a method of solving for K_{SS} above in Eqn(5 2) using Linear Quadratic Estimation Techniques This method can be implemented in the discrete case directly and produces the Gain Matrix

$$Q = [0\ 0001\ 0\ 0,0\ 0\ 001\ 0,0\ 0\ 1000]$$

$$R = 1 \times 10^{-8}$$

and using the MATLAB® Identification Toolbox

$$K_{SS} = [0\ 2539,99\ 428,54\ 63]$$
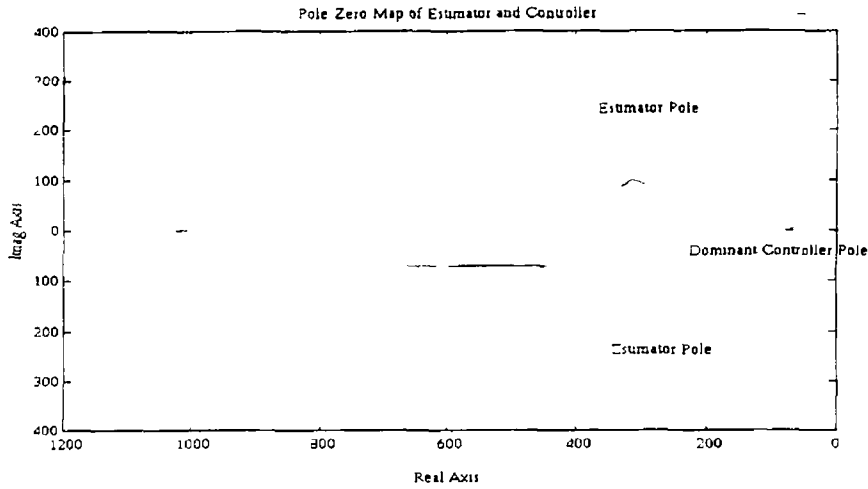
Figure 5 9 Full Controller Pole Locations

As can be seen in Fig 5 9 the dominant estimator poles are ≈ 6 times faster than the closed loop controller dominant pole The effectiveness of the estimator can be easily found in simulation where the states to be estimated are accessible to the designer The result of a *6 Nm* step in disturbance after a time *0 01sec* gives the following estimates

The states to be estimated are shown below using the estimator gain matrix of above
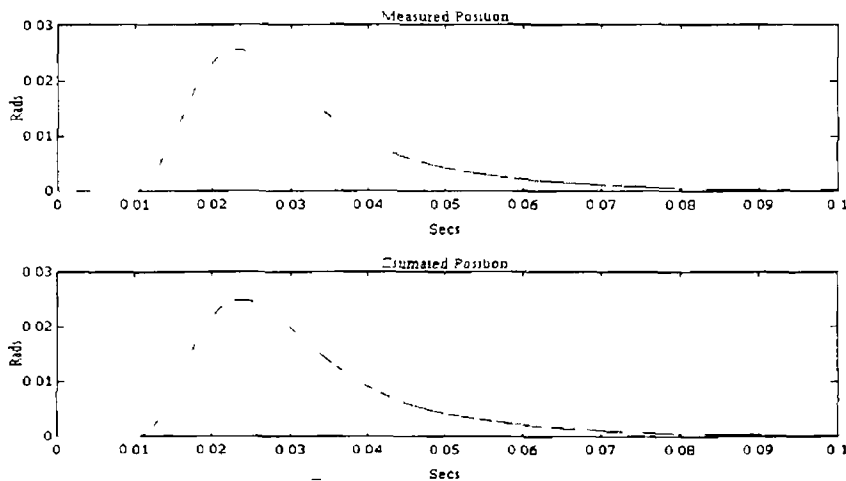
Figure 5 10 Measured and estimated position response to a 6 Nm step disturbance

53
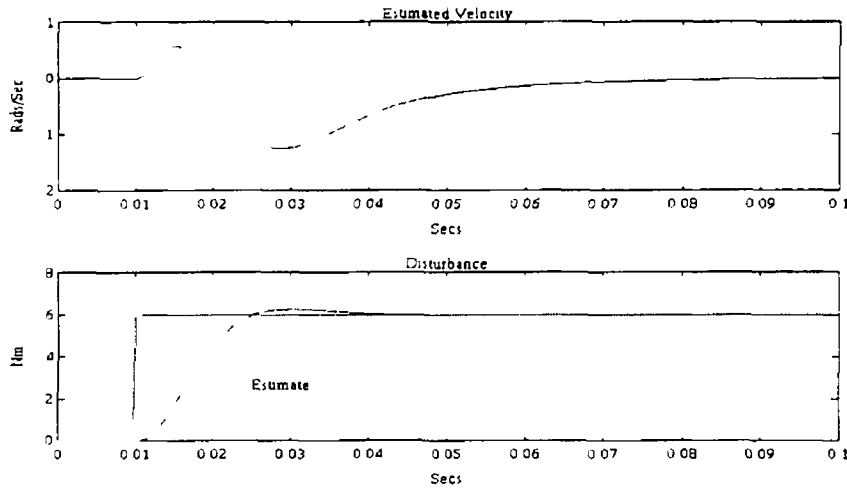
Figure 5 11 Estimated disturbance and velocity

5.7 Model Reference Controller Simulation

The final step in the controller design is to introduce the reference model as described in chapter 3 The front suspension shock absorber model chosen is as in Eqn(2 7) In order to derive values for the elements of this model, i e mass, spring stiffness, damping and coupling ratio a number of static tests were carried out on a Formula 1 racing motor cycle front suspension shock absorber The static test produced the figures below

$M_S = 2\ kg$
$B_S = 898\ Nm/s$
$K_S = 12250\ N/m$
$a = 0\ 0015\ m/rad$

Note The controller in Fig 3 10 is used in all simulation and implementation

Disturbance Inputs

In this test a disturbance step of 0 2 Nm is the input to the controller in Fig 3 10 This corresponds to a step force of 150 N on a shock absorber This represents a worst case step for the motor simulated shock absorber As can be seen from Fig 5 12 there is a delay due to the estimation present during the first number of cycles The computer simulation test results show that there is a delay of 25ms before the measured position converges with the desired (reference) position This is a worst

54

case delay deriving from the steady state nature of the estimator and is assumed negligible

Reference and Measured Position Response to 2Nm Step Disturbance
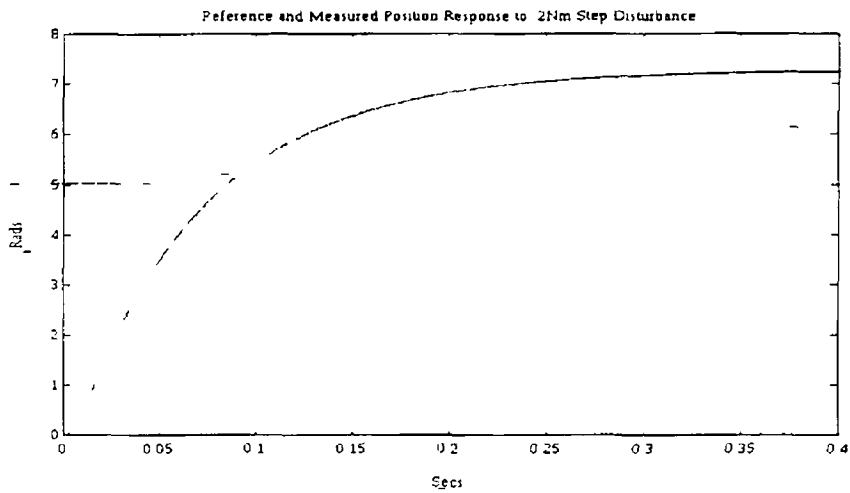
Figure 5 12 0 2 Nm step response of model reference controller

Reference (Desired) Position • ••
Measured Position ———

The controller is subsequenty tested using a sample ramp input to determine the tracking error inherent in the control design For a 1Nm ramp input Fig 5 13, concentrating on 50ms of the desired and measured, the accuracy is shown The percentage error is less than 1% This is acceptable and is judged to be negligible

Reference and Measured Position to 1Nm Ramp Disturbance
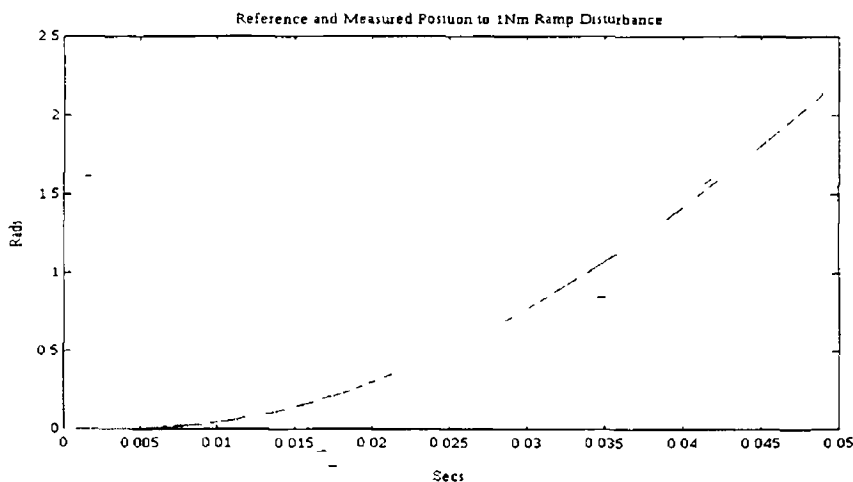
Figure 5 13 1 Nm ramp response of model reference controller

Suitable Sample inputs are chosen as below to test the system designed for this application These inputs are chosen as worst case samples taken from the data referred to in chapter 4 The object of the inputs is to fully test the control design chosen At a later date when more accurate load data becomes available this can be input in the place of this sample data

- Square Wave Inputs +/- 0 1Nm (≈150N)

- Ramp Inputs +/-1Nm (≈667N)

The input above is initially computer simulated, using Matlab on the shock absorber model to produce the desired response against which the computer simulation of the controller can be compared

In Fig 5 14 and Fig 5 15, the desired position responses is plotted in 'dashed' line and the controller position responses is plotted in an unbroken line There is a very slight delay in the controller response and only in the initial few cycles This is due to the use of the steady state Kalman filter, however as is mentioned in the estimation section of this thesis the effect of this delay is negligible compared with frequency of the disturbance input
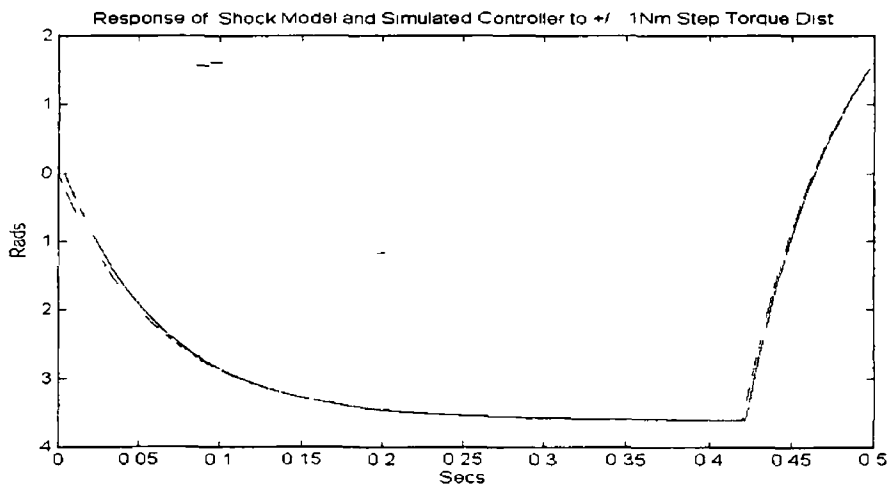
Figure 5 14 Sample Response of Simulated Model Reference Controller.
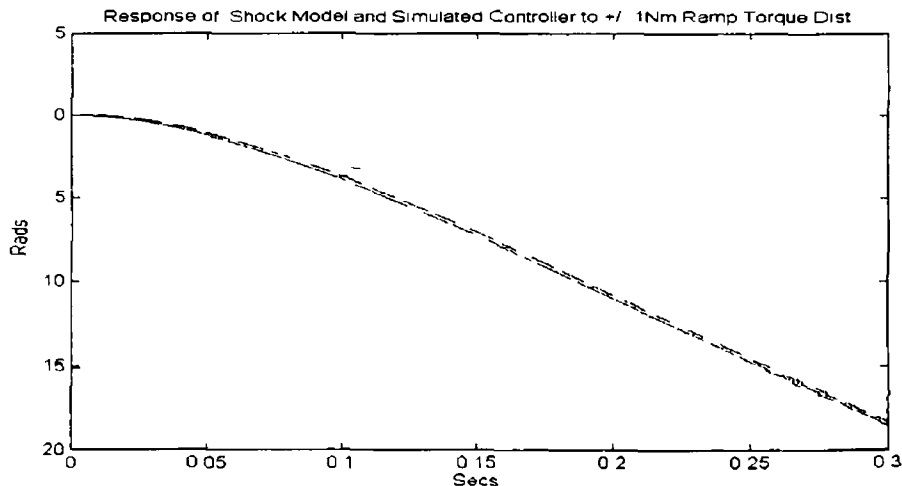
Figure 5 15 Sample Response of Simulated Model Reference Controller

Summary

The position and velocity ITAE controller was fully designed and the estimator and controller were tested using computer simulation to verify the design used In order to make the computer simulations as realistic as possible, measurement noise was assumed to be insignificant due to the fact that a digital transducer will be used to measure the position state in implementation A small portion of process noise was simulated throughout the simulated tests and had no significant effect on the accuracy of estimation The overall model reference controller responded as expected in computer simulation and proved that the shock absorber model can be simulated using the control design

CHAPTER 6

Implementation

6 1 Introduction

The design of the digital controller for this permanent magnet DC motor servo system was described previously The sample rate chosen enables realisable real-time control In this chapter the controller implementation, using the Texas Instruments DSP TMS320C30 chip and LSI card, is described The software tools required to adapt the control algorithms for this DSP platform are introduced and the hardware interface between the measurement sensor and the digital controller is described

Implementation is a description of how the theoretical control methods described previously are adapted to make them compatible with the platform on which they are to 'run' In this case flow charts describe the software required to handle the various tasks involved in implementing a motor controlled simulation of a shock absorber

The ITAE controller described in chapters 3 and 5 along with the model reference shock absorber simulation controller are implemented The actual and simulated results will be compared so as to validate the control approach used in this application

6 2 The TMS320C30

In control systems that involve a large amount of computation, i e estimation of a state vector every sample period, it is necessary to use a digital controller that has a fast response and high precision It can be difficult to implement a high precision digital servo for the following reasons

- If the signal is not sampled fast enough some information may be lost The processing since it takes a finite period of time, must be complete before the arrival of the next sample Most of the processors currently used in controllers are

not fast enough to process in real time as they rely upon lookup tables with pre computed results[6-1]

- Digital controllers use discrete steps to represent a signal The number of steps is limited to the wordlength of the processor This loss of resolution is referred to as quantisation error In addition, mathematical operations have to fit in a limited wordlength and this may cause a part of the operation to be lost This is referred to as truncation error Both of these errors cause conditions that can lead to instability

- Register overflow is another problem with digital controllers and is caused by successive mathematical operations and leads to sign changes and inaccuracies

All of the problems above will be shown to be overcome using the TMS320C30 DSP control card

6.2.1 Processor

The TMS320C30 controller is implemented on the TMS320C30 PC system board which consists of a TMS320C30 processor, expansion memories, a two channel 16-bit analogue interface, a PC interface a parallel expansion and a serial expansion It has a large memory space with 16 million 32-bit words and floating point arithmetic capabilities, (32-bit integer/40-bit floating point multiplier and ALU)[6-2], and provides high speed communication between the board and PC

The TMS320C30 DSP is a processor that has an execution time of 60nsecs It achieves this by implementing many functions in hardware which other processors implement in software or in micro code

The TMS320C30 can perform parallel multiply and ALU operations on integer or floating point data in a single cycle[6-3] The processor also possesses a general purpose register file, program cache dedicated auxiliary register arithmetic units and internal dual access memories

6.2.2 Memory Map

The total memory space of the TMS320C30 is 16M 32-bit words Programs, data, and I/O space are contained within this allowing tables, coefficients, program code,

or data to be stored in either RAM or ROM The card uses microprocessor mode memory maps as shown in Fig 6 1

0H 0BFH	Interrupt Locations and Reserved (192)	0H 0BFH	Interrupt Locations and Reserved (192)
0C0H 7FFFFFH	External $\overline{STRB}$ Active	0C0H 0FFFH	ROM(Internal)
800000H 801FFFH	Expansion Bus $\overline{MSTRB}$ Active (8K)	1000H 7FFFFFH	External $\overline{STRB}$ Active
802000H 803FFFH	Reserved (8K)	800000H 801FFFH	Expansion Bus $\overline{MSTRB}$ Active (8K)
		802000H 803FFFH	Reserved (8K)
804000H 805FFFH	Expansion Bus $\overline{IOSTRB}$ Active (8K)	804000H 805FFFH	Expansion Bus $\overline{IOSTRB}$ Active (8K)
806000H 807FFFH	Reserved (8K)	806000H 807FFFH	Reserved (8K)
808000H 8097FFH	Peripheral Bus Memory-Mapped Registers (6K)	808000H 8097FFH	Peripheral Bus Memory-Mapped Registers (6K)
809800H 809BFFH	RAM Block 0 (1K)Internal	809800H 809BFFH	RAM Block 0 (1K)Internal
809C00H 809FFFH	RAM Block 1 (1k)Internal	809C00H 809FFFH	RAM Block 1 (1k)Internal
80A000H 0FFFFFH	External $\overline{STRB}$ Active	80A000H 0FFFFFH	External $\overline{STRB}$ Active

Microprocessor Mode   Micro-Computer Mode

Fig6 1

The memory mapped peripheral registers are located starting at address 808000 hex The peripheral bus memory map is shown in Table 1 Each peripheral occupies a 16-word region in the memory map

The address for the Reset, Interrupt and Trap vectors are 0 hex through 3F hex, as shown in Table 2 The vectors stored in these locations are the addresses of the start of respective reset, interrupt and trap routines

The board provides two memory areas off-chip which are divided into areas A and B as shown in Fig 6 1 The external memory area A is divided into three areas again One 64k bank is populated on delivery with zero wait state devices External memory uses a hardware wait state generator to accommodate the different access times requirements of the various memories The board is equipped with 64kWords of one state memory in the B area at the address 30000 hex upwards This memory is used for the transfer of the data between PC and the TMS320C30

808000H	DMA Controller Registers
80800FH	(16)
808010H	Reserved
80801FH	(16)
808020H .	Timer 0 registers
80802FH	(16)
808030H	Timer 1 Registers
80803FH	(16)
808040H	Serial-Port 0 Registers
80804FH	(16)
808050H	Serial-Port 1 Registers
80805FH	(16)
808060H	Primary and Expansion Port
80806F	Resisters (16)
808070H	Reserved
80807FH	

Table 1 Peripheral bus memory map

00H	RESET
01H	INT0
02H	INT1
03H	INT2
04H	INT3
05H	XINT0
06H	RINT0
07H	XINT1
08H	RINT1
09H	TINT0
0AH	TINT1
0BH	DINT
0CH 1FH	RESERVED
20H -	TRAP 0
3CH	TRAP 28 (RESERVED)
3FH	TRAP 31 (RESERVED)

Table 2 Reset, Interrupt and Trap vector locations

6.2 3 Software Tools

Texas Instruments have a number of software tools for linking 'C' code programs and development of software

- Assembler

- Linker

- Archiver

- 'C' compiler

The Assembler generates object code in a 'Common Object File Format' (COFF) This format was developed to make modular programming easier Memory is allocated at link time when a command file directs the linker to where to place code, initialised and uninitialised data

6.2.3.1 Linking 'C' code

The 'C' compiler produces **asm** files that can be assembled like any other source file and can then be linked When linking 'C' code the -c option informs the linker to use special conventions defined by the 'C' environment All 'C' code must be linked with an object module called BOOT BJ This must be run first before the program starts running and performs the following tasks to set up the machine for 'C' code[6-4]

- It sets up the system stack
- It processes the runtime initialisation table and auto-initialises global variables (in ROM model)
- It disables interrupts and calls the function main

6.2 4 Analogue Interface

Despite the fact that the input data acquisition is digital, as will be discussed later in this chapter, the output torque command must be passed through an analogue interface as shown in Fig 5 6 The TMS320C30 PC System Board contains a complete analogue I/O subsystem There are two separate channels, each containing its own sample/hold amplifier, its own convertors, D/A and A/D, and analogue filters on the input and output

The A/Ds are Burr-Brown PCM78P devices which offer 16-bit precision with up to 200kHz sample rates The conversion is triggered by timer 1 (the 32-bit on-chip timer) which can be programmed by software to a resolution of 120nsec The analogue i/o subsystem is accessed through three 16-bit secondary I/O bus mapped registers

- Read/write channel A A/D & D/A 0804000 hex

- Read/write channel B A/D & D/A 0804001 hex

- Generate software conversion trigger 0804008 hex

Although each register consumes 32 bits of space, only the top 16 bits of each register are used the first two registers are used to access the A/D and D/A converters on the two analogue I/O channels All the registers are accessed with two memory wait states This results in an overall time of 180nsec to access each register

6 3 Sensor

Shaft encoders are digital transducers that are used for measuring angular displacements Applications of these devices include motion measurement in control of robotic manipulators machine tools and rotating motors The output of an incremental encoder is a pulse signal that is generated when the transducer disk rotates as a result of the motion being measured[6-5]

An incremental encoder, a Heidenhain Rotary Optical Encoder (ROD 450 1800), is coupled to the motor shaft[6-6] This encoder is chosen due to its high resolution high accuracy due to the noise immunity of digital signals superior construction and ease of adaptation in a digital control system, all of which are advantages over its analogue counterparts

6.3.1 Sensor Interface

An encoder interface card is required to interface the encoder with TTL compatible outputs to the Loughborough Sound Images (LSI) TMS320C30 DSP board The board communicates with the interface card via the LSI DSPlink bus

The DSPlink is the digital parallel system expansion of the micro controller TMS320C30 This system is provided as a memory-mapped peripheral area It has 16-bit width and transfers over the link using 2 wait-states to achieve a 180ns transfer cycle which is suitable for ribbon cable connection to the interface card DSPlink defines a minimum of 16 I/O ports in the I/O space The port being accessed is determined by a 4-bit address The entire address decoding is performed using the 74138 chip[6-7]

The interface card has been designed to operate with a maximum encoder line count of 10,000 lines at a maximum motor speed of 6,000 rpm The interface card has a number of features including non-slip counting, synchronous zero point crossing and software counter preloading for different line counts

The card is designed in three distinct blocks

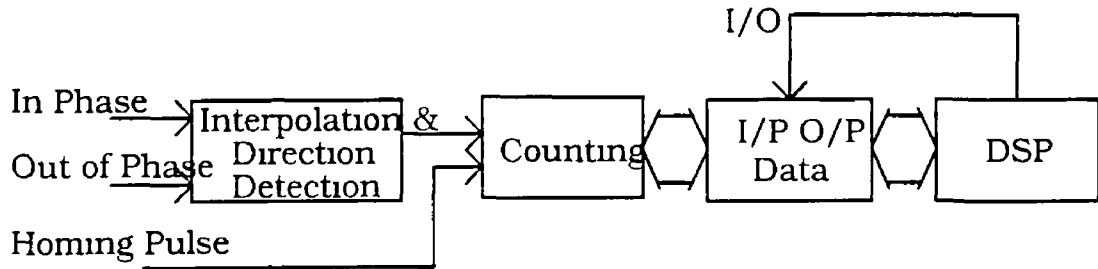- Interpolation and Direction Detection

- Counting

- Data Input/Output

Figure 6.2 Block diagram for encoder interface card

The interface card counts the edges of the incoming phase and quadrature out of phase signals It creates a count pulse on each of its rising and falling edges, using the finite state method This takes the previous and present states of the two signals and combines them logically to determine direction

A counter, consisting of four 4-bit counters, registers each count pulse and increments or decrements according to the direction signal A homing pulse is also available from the incremental encoder and is used to LOAD and CLEAR the counter If the homing pulse is present and the direction signal is counting down, a preset number is loaded into the counter If the direction signal is counting up, the counter's contents are cleared to zero

The card has been designed to capture and read the counter's contents without interference to the counting process and that data capture is avoided during counter transitions The preset number, as mentioned earlier, is loaded initially into a D-type latch where it is available for the LOAD signal The full circuit diagram of the encoder card is shown in Appendix B

The incremental encoder requires initialisation before an effective measurement of position is available to the controller To achieve this the motor shaft must be forced to pass through the homing pulse This is achieved using a simple torque ramp command as shown in the code and in the flow chart Fig 6 4

6 4 Software Design

The TMS320C30 is well supported by a full set of software development tools which include assembly-language, 'C' language or a mixture of both In this project the code will be written in 'C' as it is more user friendly and can be adapted and commented easily

6.4 1 Initialisation

The initialisation routine consists of page pointer, stack pointer, external bus operation, timer and wait states The initialisation program flow chart is shown in Fig 6 3

- External Bus Operations

Two external interfaces are provided on the TMS320C30, namely the primary bus and the expansion bus The processor uses these buses to access data between memory and external peripheral devices The primary bus consists of a 32-bit data bus and a 24-bit address bus The expansion bus consists of a 32-bit data bus and a 13-bit address bus Both the primary bus and the expansion bus have associated control registers which are memory-mapped

- Timer set up

The TMS320C30 proves two internal timers, timer 0 and timer 1 The system board users timer 1 to signal the external A/D to start conversion with an internal clock Three memory mapped registers are used by the timer which are global control register, period register and counter register The timer global control register is a 32-bit register that contains the global and port control bits for the timer module in which the bits 3 to 0 are the port control bits and the bits 11 to16 are the timer global control bits The global control register determines the timer operating mode, monitors the timer status and controls the function of the I/O pm of the timer The 32-bit timer period register specifies the timer's signalling frequency

- Interrupt Control

After the timer starts an A/D conversion the A/D performs the conversion then outputs an end of conversion signal which triggers the INTL interrupt request

Interrupts are synchronised internally by three flip-flops Once synchronised, the interrupt will set the corresponding interrupt flag register bit if the interrupt is active

When a particular interrupt is processed by the CPU, the corresponding interrupt flag bit is cleared by the internal interrupt acknowledge signal for one cycle and then set to '1' again When the TMS320C30 is reset zero is written into the interrupt flag register, thereby clearing the pending interrupt

To enable the interrupt a '1' must be written to bit 1 of the TMS320C30's IE register and a '1' to bit 13 of the store register The CPU global interrupt enables bit GIE, located in the CPU store register and controls all CPU interrupts
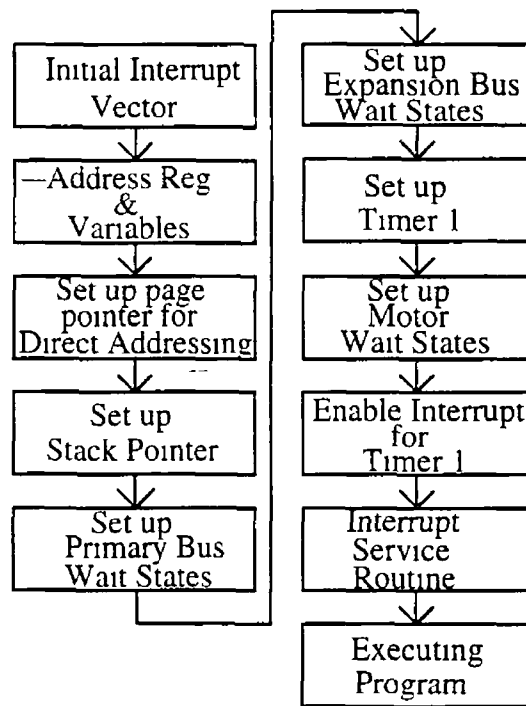

Figure 6 3 Initialisation flow chart

6 5 Program Flow Chart

The control routine executes the interrupt service routine and the main control loop as shown in Fig 6 4 The interrupt service routine samples signal data from the DSPlink, performs the control techniques as described in chapter 3 and 5 and outputs the control output to the D/A converter

67

The controller software offers two modes of operation a Display Mode and a Command mode

- The Display mode scrolls any parameters or states on the PC screen giving a visual interface to the user

- The Command Mode allows the user vary specified parameters on-line without having to recompile the program See Appendix C
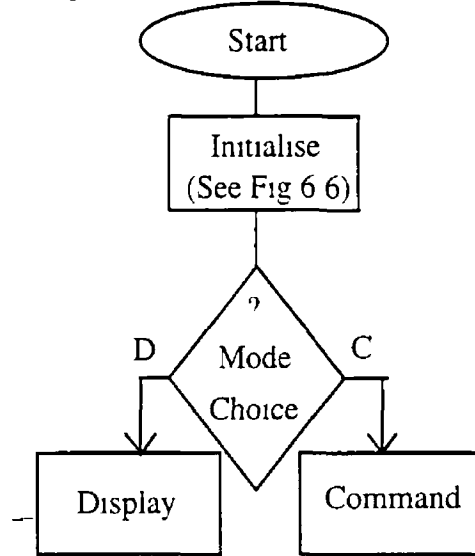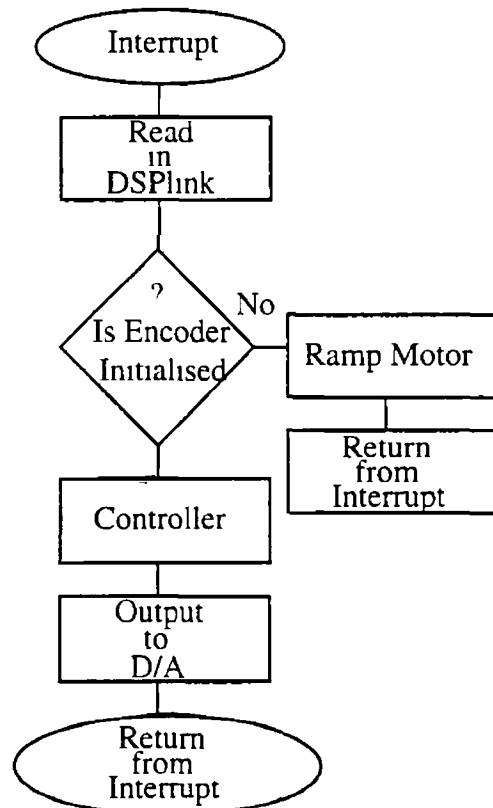

Figure 6 4 Main routine flow chart

Figure 6.5 Interrupt routine flow chart

6 5.1 Data Output

The analogue output ranges between -3 0 volts and 3 0 volts The required input to the current loop, described in chapter 2, is between -10 volts and 10 volts A simple inverting amplifier is used between the output of the D/A and the input to the current loop Floating point numbers of -1 000 and 1 000 correspond to max torque negative and positive respectively

6 5 2 Serial Sampling

For this Real Time Application the sampling is carried out at 5kHz This does not allow the time needed to write the required data to a file and remain sampling A high speed serial link is provided on the TMS320C30 The serial port can transfer up to 32 bits of data in both directions simultaneously

A second board is used as the receive side of the link This link facilitates the transfer of all states at a sample rate of 1kHz to a file that can be saved and used to test and debug the software and control design
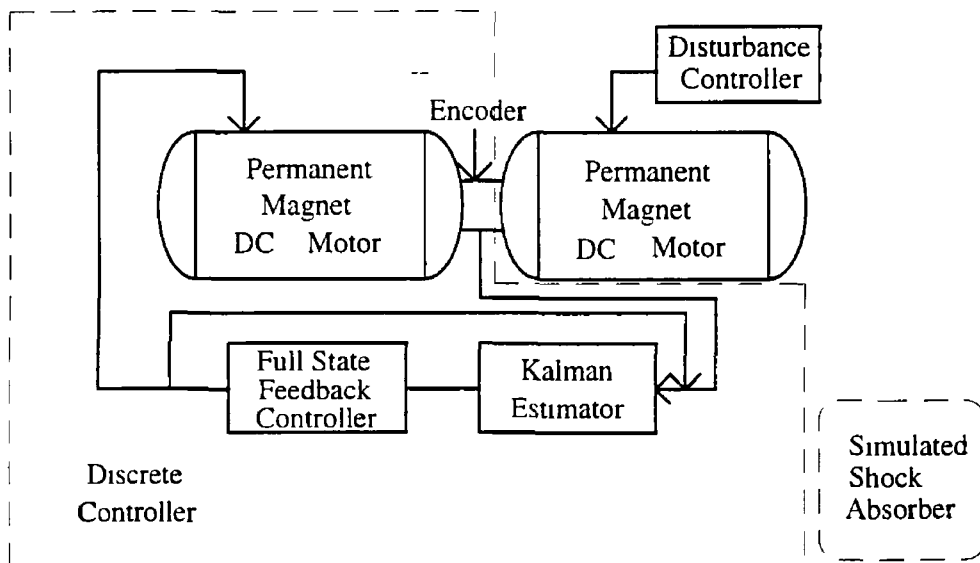

Figure 6 6 Implementation hardware

6.6 Final Simulation and Tests

The implementation will be carried out using the system shown above in Fig 6.6.
All control simulation is carried out for the final implementation using MATLAB®
and simulab which is a program for simulating dynamic systems [6-8].

6.6.1 Controller and Disturbance Rejection Results

The controller is implemented using the methods described in both chapter 3 and 5.
A step command of *0.1rads* is required at the controller. The disturbance rejection is
tested by producing a *6 Nm* step disturbance to the controller after *0.3sec*.
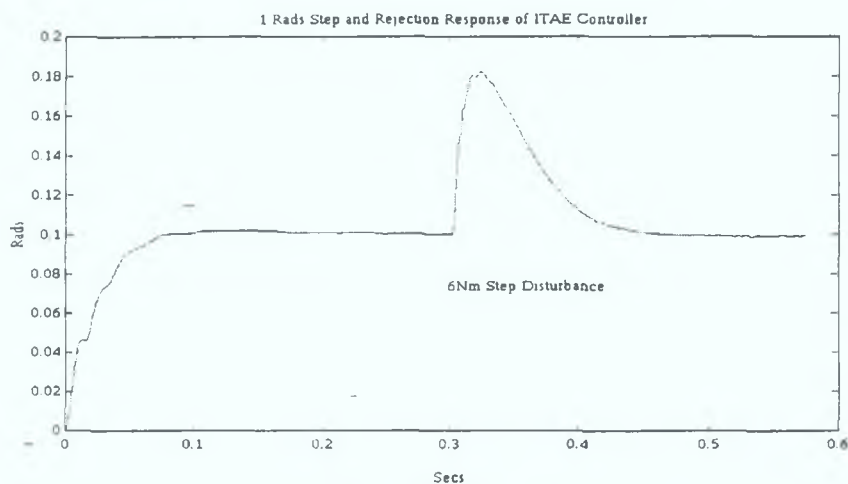
**Figure 6.7 Position response of implemented state feedback controller with
disturbance rejection**

The implemented step is marginally slower than the simulated, however the marginal
difference is acceptable and is negligible. The rejection properties are seen to be as
designed and this is a vital aspect of the model reference controller as described in
chapters 3 and 5.

Identical inputs to those described in chapter 5 are now used as inputs to the
controlled system as shown in Fig 6.6. Simulated Responses Figs 5.14 and 5.15 to
the above inputs will be compared to the implemented responses.

The desired controller responses are plotted in a 'dashed' line and the actual
implemented resplonse are plotted in an unbroken line.
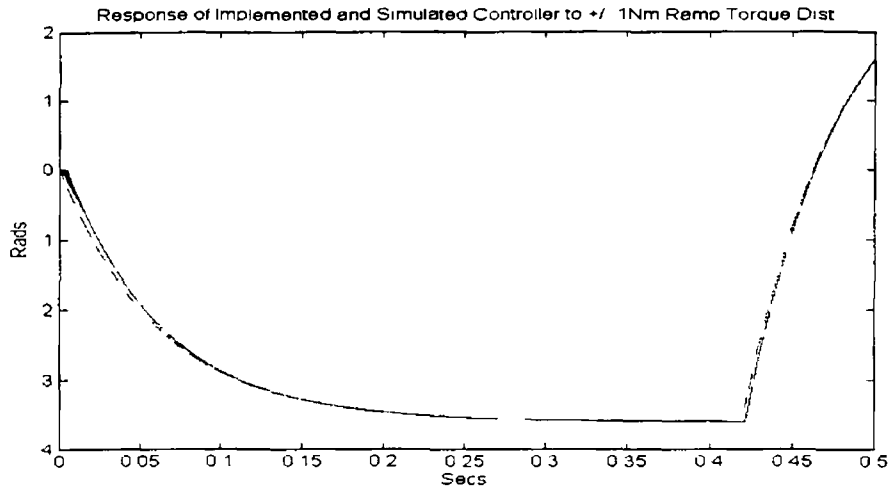
Figure 6 8 Implemented model reference controller
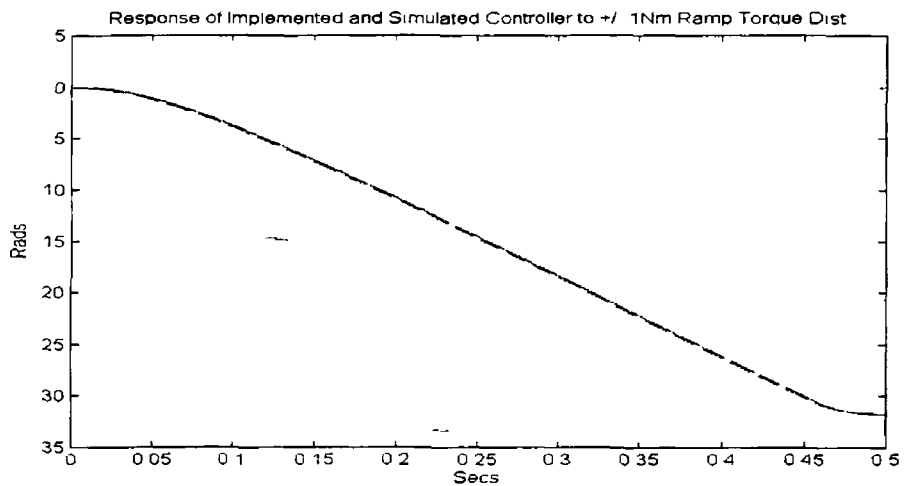
Figure 6 9 Implemented model reference controller

It can be seen from the above responses that there is an almost perfect match between the simulated and actual There is a slight delay in both responses over the first few cycles This may be attributable to slight modelling errors and an initial delay in the estimation of the state vector As stated earlier these errors are negligible for this application

Summary

This chapter has described the implementation for the motor simulated shock absorber The hardware was designed to produce accurate and 'clean' measurement information that is then available to be used in the the discrete controller implemented on a real time DSP platform The software user interface capability of the controller was a very significant aspect of this system and allowed the user access to all the parameters, such as controller and estimator gains, so that the designer and user were

71

was a very significant aspect of this system and allowed the user access to all the parameters, such as controller and estimator gains, so that the designer and user were able to tune the controller on-line without having to re-compile the program after each parameter change

CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS

7.1 CONCLUSIONS

This thesis outlines the design methodology for the control of a motor simulated shock absorber To be more specific a Formula 1 motor-cycle front suspension shock absorber was simulated using a Permanent Magnet DC motor and digital controller The design applied throughout the thesis was presented in the following order

- Shock Absorber Modelling
- System Description
- Control Approach
- Control Design
- System Implementation

There were two main problems posed by this servo application The first of these was the fact that a *rotary* motor was required to simulate a *linear* mechanical system It was necessary therefore to transform the linear system model into a rotary one by means of a suitable coupling ratio In chapter 4 a methodical approach to choosing this ratio was introduced and sample calculations were made for the permanent magnet motor used

The second problem that had to be dealt with was the fact that the disturbance torque T_{dis} was not measured directly and therefore had to be calculated from the position information produced by a digital rotary encoder This process required state estimation techniques using a Kalman Filter The steady state version was chosen over its time-varying counterpart, because it was numerically less intensive and allowed the allocation of more processing time to the user interface tasks These tasks included display and on-line parameter changes which are vital aspects of any experimental rig such as this one

The rotary encoder was chosen in this application for the reasons described in chapter 6 The most important of these being the fact that the digital nature of the

measurement signal was almost totally immune to noise A resolver can also be used with slight variation to the control algorithms however, the estimator poles will be slower than the ones designed and documented in this thesis

An ideal mechanical shock absorber model was chosen as the reference model in this design It was assumed to be linear and of 2nd order which was regarded as the simplest, while still effective representation The choice of model however, was not considered to be of paramount importance as the control techniques can be varied with ease to take more complex models into account

As described in chapter 1, the intention of the work carried out was to introduce and implement a technique for the control of an electrical actuator to simulate the passive shock absorber section of a Fromula 1 motorbike front suspension It was also the intention to produce an experimental test-rig that can be used as part of further research into the area of electrical suspensions The objective, described in the introduction of producing a test rig that can be used as a motobike front suspension shock absorber simulation rig and an active load was realised In chapters 5 and 6, it was seen that, both in simulation and implementation, the control techniques chosen were valid and produced desired responses to randomly chosen disturbances

7.2 RECOMMENDATIONS

As mentioned above the initial research specifications set out in the introduction were met and a motor simulation of a shock absorber has been implemented There are however steps that should be taken to improve the accuracy of simulation and also to bring this research further towards realising the electrical active suspension concept Recommendations made by this author are as follows

- It is vital that a more accurate shock absorber model is available to the control designer The model used for this thesis is a simplification assuming a linear and time invariant representation. Research is currently being carried out into this area using system identification techniques[7-1] and it is probable that the actual shock absorber model is time varying

- The model reference controller structure chosen can easily be evolved from a simple linear feedback scheme into an adaptive approach that can vary shock absorber parameters when required Model reference adaptive control of a motor

simulated shock absorber is a logical step towards an electrical active suspension and is recommended by this author as the control approach to be considered next

- More accurate load data (disturbance torque) must be made available to produce more realistic test data For this thesis random sample inputs, chosen as suitable test signals, are taken from cornering data as described in chapter 4 However, this only verifies the control theory used

Bibliography

CHAPTER 1

1-1 P G Wright, "The Application of Active Suspension to High Performance Road Vehicles", *Proc of Int Congress Transport Election*, Deaton, MI, USA, pp 333-338, Oct 1986

1-2 Lane R Miller "Tuning passive, semi-active and fully active suspension systems" Proceedings of the 27th Conference on Decision and Control, Texas 1988

1-3 A B Dunwoody "Digital Active Suspension" Car Supension Systems and Vehicle Dynamics 1991 pp33-43

1-4 Satoh S, Fukushima N "An active suspension employing an electrohydraulic pressure control system" Proceedings of the 27th Conference on Decision and Control, Hawaii, 1990

1-5 M R Stojic, "Design of microprocessor-based digital system for DC motor speed control", IEEE Trans on Industrial Electronics , Vol IE-31, no 3, pp243-248, 1984

CHAPTER 2

2-1 Lane R Miller "Tuning passive, semi-active, and fully-active suspension systems" Proceedings of the 27th Conference on Decision and Control, Austin, Texas Dec 1988

2-2 D C Karnopp, M J Crosby and R A Harwood, "Vibration control using semi-active force generators" ASME Paper 73-DET-122,1973

2-3 Myoungho Sunwoo, Ka C Cheok, and J Huang "Model reference Adaptive control for the vehicle active suspension systems" IEEE Trans on Industrial Electronics, Vol 38, No 3, 1991

2-4 Tak Kenjo "Electric motors and their Controls" Oxford Press 1991

2-5 AEG PWM/K "Installations and Service Manual"

2-6 Tak Kenjo "Permanent Magnet and Brushless DC Motors" Oxford Press 1985

2-7 I J Nagrath M Gopal "Control Systems Engineering", 2nd Ed , Wiley & Sons 1982

2-8 "Identification Toolbox Users Guide", Mathworks Inc 1991

2-9 Lennart Ljung "System Identification" Prentice Hall 1987

CHAPTER 3

3-1 G Franklin, J Powell, M Workman, "Digital Control of Dynamic systems", Addison Wesley, 1990

3-2 G F Franklin, J Powell, A Emami-Naeini "Feedback Control of Dynamic systems" Addison-Wesley, 1991

3-3 Nagrath & Gopal "Systems Modelling and Analysis" McGraw-Hill 1982

3-4 Frank L Lewis "Optimal Estimation" Wiley & Sons, 1986

3-5 D E Catlin, "Estimation, Control, and the Discrete Kalman Filter" Springer-Verlag, 1989

CHAPTER 4

4-1 J Floresta, "Power Rate" Kollmorgen Corporation New York, 1984

4-2 R Balli, "Dimensioning the active suspension system of a wheeled vehicle" Electrische Bahnen eb 88, 1990

4-3 P V Knight "Operations Manual 20 Channel CardCorder Unit", Cranfield Impact Centre, 1990

4-4 J Tal, "The Optimal Design of Incremental Motion Servo Systems", Chap 15 of B C Kuo and J Tal, DC Motors and Control Systems, SRL Publishing Company 1978

4-5 Mohan, N, Undeland, T M, Robbins, W P "Power Electronics" Wiley & Sons, 1989

4-6 N A Shneydor, "A Procedure for the design of motor and coupling for DC servo Applications" Israel Institute of Technology, 1989 IEEE

CHAPTER 5

5-1 D Graham & R C Lathrop, "The Synthesis of Optimum Transient Reponse Criteria and Standard forms", AIEE Trans Vol 72, part 2, pp 273-288, 1953

5-2 P Kettle & A Murray, "Disturbance Rejection and DC motor control" Irish DSP and Control Colloquium, 1993

5-3 Yoichi Hori "Disturbance Suppression on an acceleration control type DC servo system", PESC '88 pp 222-229 1988

5-4 G Franklin, J Powell, M Workman. "Digital Control of Dynamic systems", Addison Wesley, 1990

5-5 H F Van landingham, "Introduction to Digital Control Systems", MacMillan, 1985

5-6 F L Lewis "Optimal Estimation with an Introduction to Stochastic Control Theory" Wiley & Sons, 1976

5-7 Lennart L, "Theory and Practice of Recursive Identification", MIT Press, 1983

CHAPTER 6

6-1 Panos E Papamichalis, "Digital Signal Processing Applications (volume 2)" Prentice Hall, 1991

6-2 "Third generation TMS320C30 User's Guide", Texas Instruments, 1988

6-3 TMS320C30 Assembly Language Tools Users's Guide, Texas Instruments, 1988

6-4 D Doyle, Design and developement of real-time multitasking executive for embedded systems Meng 1994

6-5 Norton H N, "Sensor and Analyser Handbook" Prentice Hall, 1982

6-6 Heidenhain "Incremental Rotary Encoders" May 1991

6-7 Loughborough Sound Images Ltd, "TMS320C30 PC System Board User Guide" Version 1 01 Sept 1990

6-8 "Simulink User's Guide" Mathworks 1992

CHAPTER 7

7-1 J-Ledwidge, System Identification and Parameter Estimation of a Motorcycle Suspension System, A Masters Thesis, Dublin City University, 1995

APPENDIX A

Motor Parameters

Parameter	Symbol	Units	Rating
Power	P	W	746
Max Operating Speed	ω_m	rpm	2000
Peak Torque	T_{max}	Nm	33 9
Continuous Torque	T_R	Nm	4 07
Peak Current	I_{max}	A	90
Continuous Current	I_R	A	8 04
Max Terminal Voltage	V_{max}	V	115
Torque Sensitivity	K_T	Nm/A	506
Back EMF constant	K_B	V/(rad/sec)	506
DC resistance @ 25^{o}C	R_A	Ω	1
Inductance @ 1000 Hz	L_A	mH	3 2
Mech Time Constant	τ_m	sec	1 9
Elec Time Constant	τ_e	msec	3 2
Cross Constant	τ_{em}	msec	5 5
Rotor Inertia	J_M	kgm^2	0014
Damping	B_M	Nm/(rads/sec)	075

APPENDIX A.1

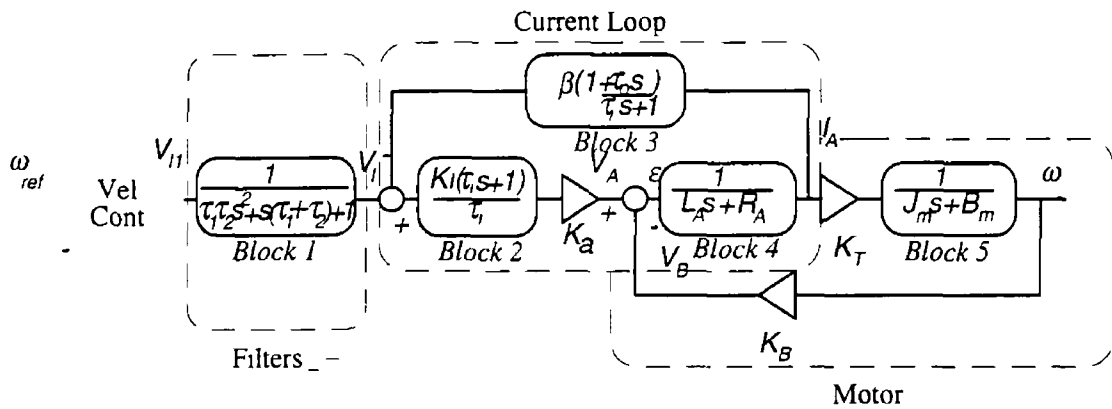## Circuit Diagram Reduction

Fig A 1 Block diagram of current servo system

The circuit below is taken from the circuit diagram in Appendix D
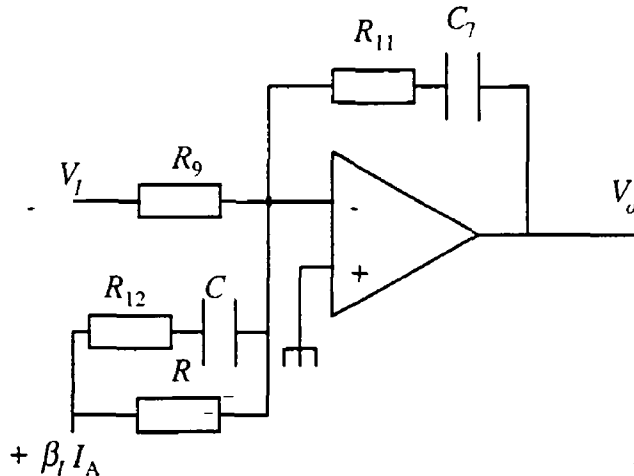
Current Loop Block

Fig A 2 The operational amplifier circuit of Blocks 3&4

$$\frac{V_o}{R_{11} + \dfrac{1}{sC_7}} = \frac{V_I}{R} - \beta_I \left(\frac{1}{R} + \frac{1}{R_{12} + \dfrac{1}{sC_9}} \right) I_A$$

where $R_{11}=100k\Omega$, $R_{12}=100k\Omega$, $R=43\ 2k\Omega$, $C_7=022\mu F$, $C_9=022\mu F$ and V_o is the output of the current loop

$$V_o = \frac{R_{11}}{R}\frac{R_{11}C_7+1}{R_{11}C_7 s}(V_I - \beta_I I_A(1+\frac{RC_9 s}{R_{12}C_9+1}))$$

$$V_o = K_I \frac{\tau_i s+1}{\tau_i s}(V_I - \beta_I I_A(1+\frac{\tau_o s}{\tau_i+1}))$$

where τ_i, $=R_{11}C_7$

$\tau_o=RC_9$ and β_I is the current sensitivity of the motor

In the torque-speed characteristic for this motor available in the data sheets supplied with the motor, the maximum torque is *34Nm* The maximum current therefore is $T_{max}/K_T=34/\ 506=67\ 2A$ The maximum voltage of the controller is *10V* Hence

$$\beta=10V/(67\ 2/10)=1\ 4$$

Block 1 Input Filters

The filters are D1 & D2 in Figure D1
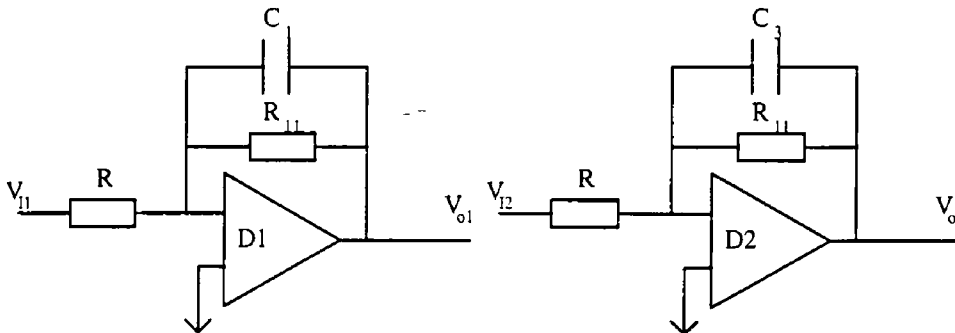
Figure A 3 Operational Ampliers Block 1

The transfer function of the circuit above is given below

$$\frac{V_o}{V_{I1}} = \frac{1}{RC_1 s+1}\times\frac{1}{RC_3+1}$$

$C_1=0\ 01\mu F$

$C_3=001\mu F$

$R=20k\Omega$

PWM Block

K_a is the gain of the PWM generator and inverter This can be approximated to a gain The input to the PWM is +/- *10V* and the DC output of the inverter is *140V* Therefore $K_a=14$

APPENDIX B

Encoder Interface Card Circuit Diagrams
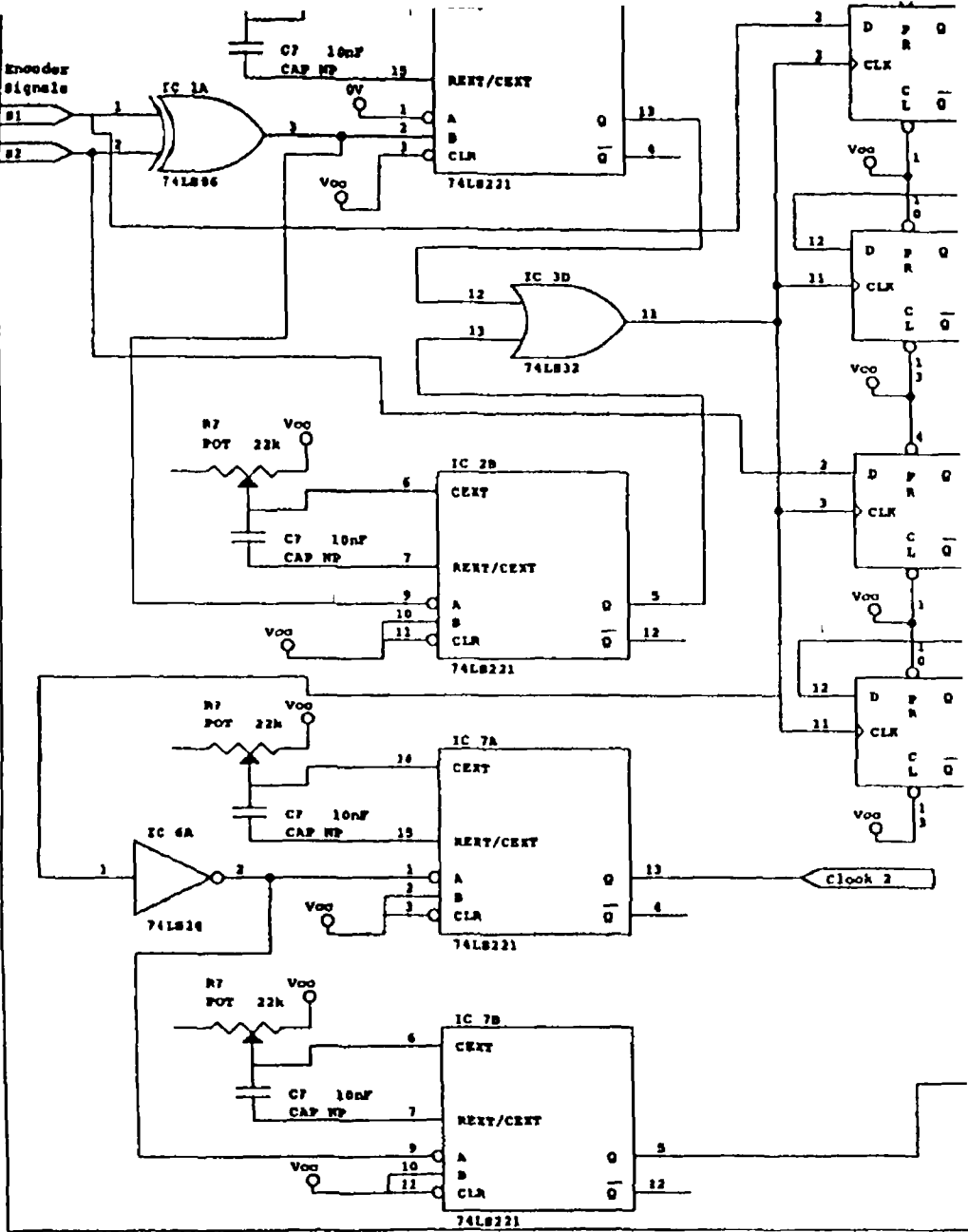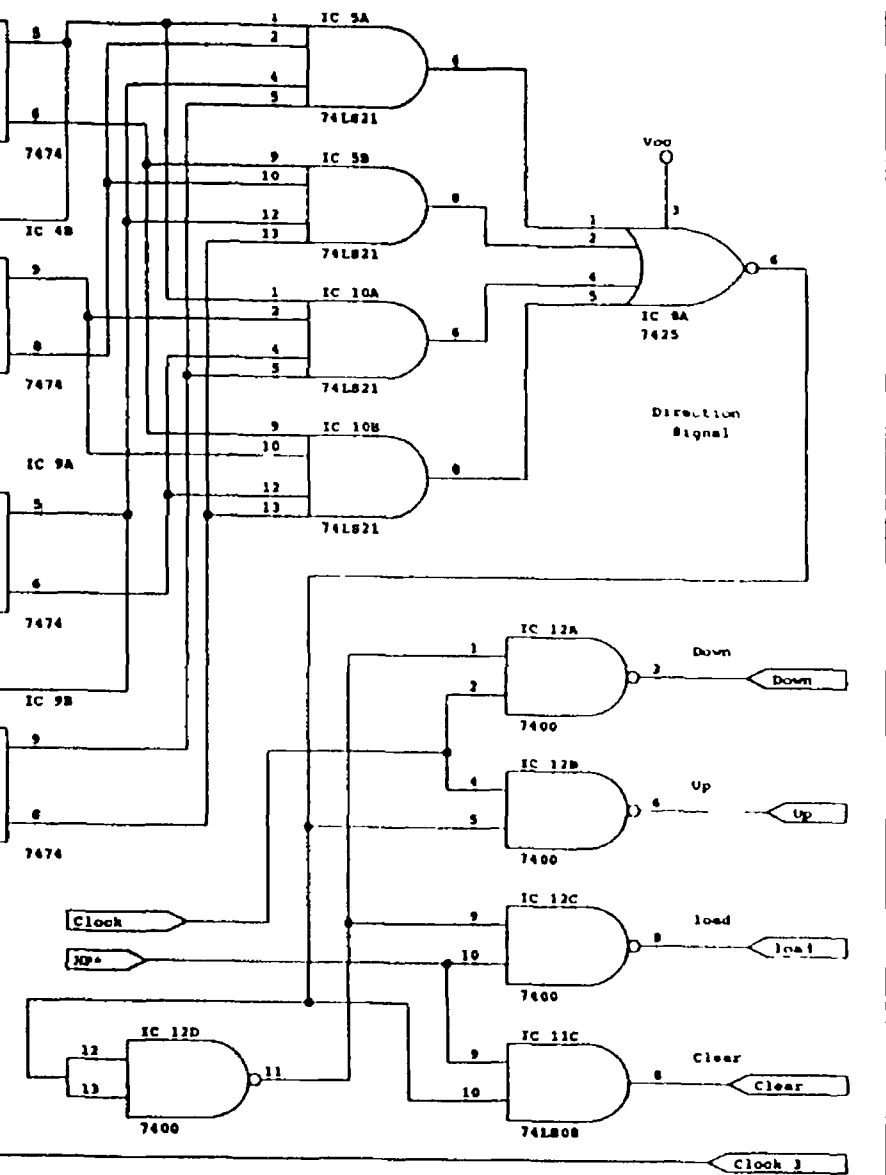
Encoder Interface Card Schematic

Size	Document Number		REV
A	Sheet 1		
Date: June 4, 1995	Sheet 1 of 3		
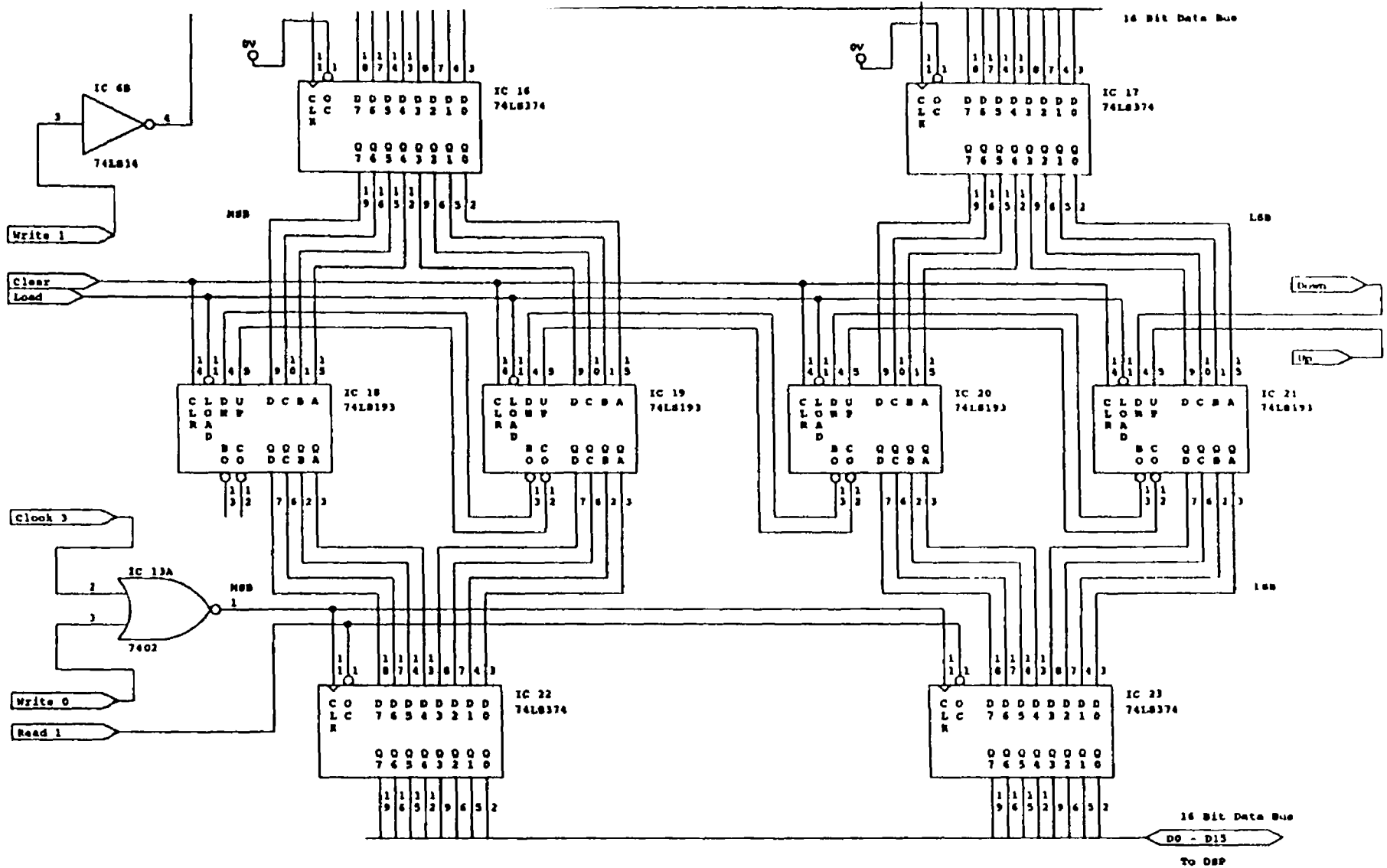
Encoder Interface Card, Schematic

Size	Document Number		REV
A			
	Sheet 2		
Date:	June 6, 1995	Sheet	2 of 3
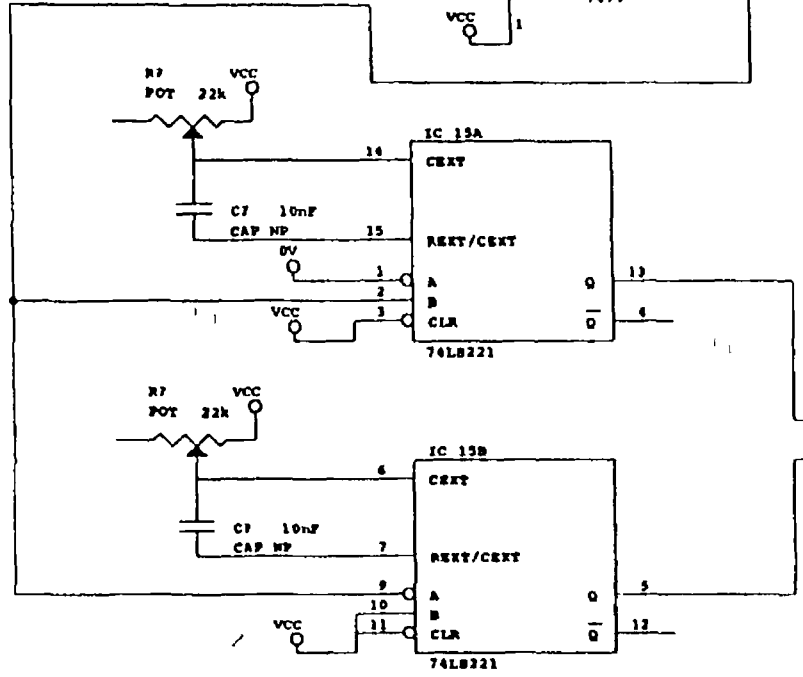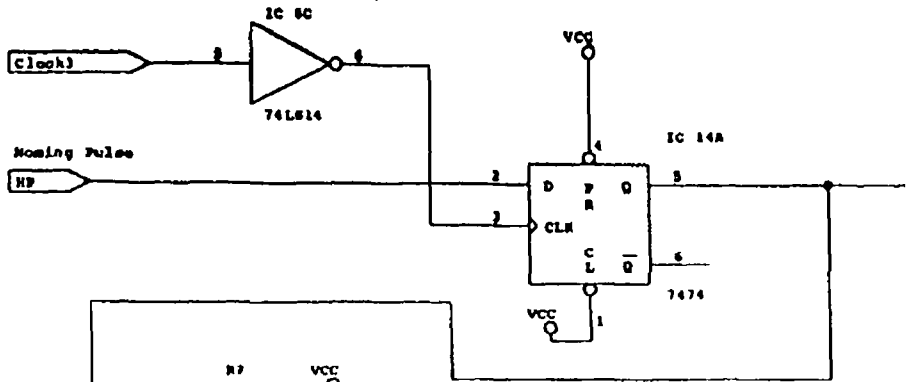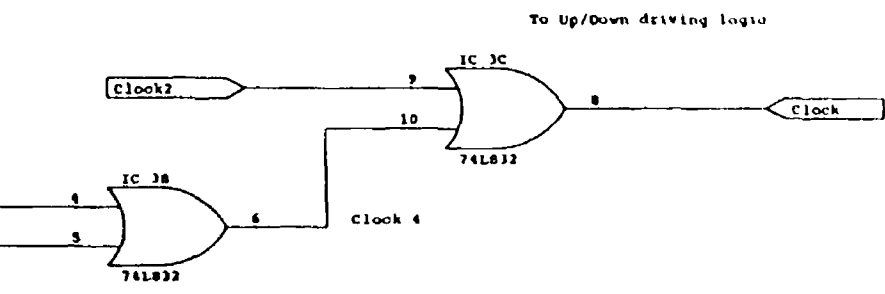
To all MP inputs

MP

To Up/Down driving logic

Clock2

IC 3C

9

10

74LS32

8

Clock

IC 3B

4

5

6

74LS32

Clock 4

APPENDIX C

Program listing

```
#include "dsp_link h"
#include "std_lib h"
#include<stdio h>
#include<math h>
#include<stdlib h>
#define SERIAL
#define IR_INT01 0x2
#define DEF 5
#define CON 6
#define FREQUENCY              5000 //Sampling Frequency//
#define CONTROL_FREQ           5000
#define SERIAL_SAMPLE          5
#define T (double) 1/FREQUENCY
#define BITS 256
#define saturate(a,b) ((a)>(b))?(b) ((a)<-(b))?(-b) (a) /Saturate output floating point-
/+1//
void Init_ISR(unsigned int),
void c_int01(void),
void setvect(),
int getvect(int num),
void signal(),
extern display(),
extern confine(),
void GIE(int),
```

// Model Reference is described here A_M and B_M//

```
float Am[2][2]={
            { 9999, 0002},
            {-1 1716, 9140}
            },
float Bm[2][1]={
            {0},
            {4 33}
            },
```

/* Augmented Plant Matrix */

```
float X[3][3]={
            { 1, 0002,0 0000},
            {0, 999,0 0342},
```

```c
                    {0,0,1}
                    },
float Y[3][1]={
                    {0 0003},
                    {1 7092},
                    {0}
                    },

//Kalman Gain Matrix //

float K_SS[3][1]={
                    {9492},
                    {100 994},
                    {2 632}
                    },

int echo=DEF,
double interval,
float Pref=0,Vref=0,mes_pos=0 0,pos=0 0,vel=0 0,pos1=0,vel1=0 01,vel2=0 t=0 0,
float Tm=0 0,Td=0 0,U=0 0,kw= 0734,kp=4 76 ki=50,
float X_[3],flagie=1,e=0 0,
char command[256],
char *ptr,
double Step=1,

int IN_ISR=0,
int stop=0,
unsigned long real_timer=0,

#if defined(SERIAL)
extern serial_command,
#endif

extern init_serial(),
extern send_states(),

int sleep(int),


smain(int argc,char *argv[])
        {
        int i,
        double vtmp=0 0,ptmp=0 0,

        for(i=1,i<argc,i++)
                {
                if(*(argv[i]++)=='/')
                        {
```

```c
switch(*(argv[1]++)) //Varying Controller and Estimator
                      Parameters at execlsi command//
    {
    case 'p' kp=atof(argv[1]),      break,
    case 'w' kw=atof(argv[1]),      break,
    case 'K'
    Kss[0][0]=atof(argv[1++]),
    Kss[1][0]=atof(argv[1++]),
    Kss[2][0]=atof(argv[1++]),
    break,
    case 'E'                break,
    case 'C' echo=CON   break,
    default             break,
    }
  }
}

#if defined(SERIAL) //Serial Link //
init_serial(),
#endif

Init_ISR(FREQUENCY),

while(1)
    {
    #if defined (SERIAL)
    if(serial_command==START)
        {
        sleep(SERIAL_SAMPLE),
        send_states(),
        }
        else
        #endif
            {
            display_mam(),
            }
        }
    }
    sleep(int c_ticks)
        {
        static int last_tick=0,
        while(real_timer<(last_tick+c_ticks)),
        last_tick=real_timer,
        }
      }
    }
  }
void Init_ISR(unsigned int freq)
    {
```

```c
        int * temp,
        temp=PRIMCTL,
        *temp=0x800
        temp=EXPCTL,
        *temp=0x00,
        setvect(IR_INT01,c_int01),
        set_timer(freq),
        enable(IR_INT01),
        }
void c_int01()
        {
        unsigned long int *reg,*cntl,
        static long the_count,
        float res=36000,
        real_timer++,
        reg=ENCODE_REG,
        cntl=ENCODE_CTL,
        *reg=PERIOD<<16,
        *cntl=READ,
        mes_pos=((float)((*reg>>16))*2*M_PI)/res,
        if(int0flag()==0)                         //Encoder Initialisation//
                {                                 //If flag is not set //
                outport(A_CHAN,-FULL*vel1),       //Encoder is initialised //
                vel1=vel1+001,
                return,
                }
        controller(mes_pos),
        outport(A_CHAN,-FULL*Tm),                 //Output
Commanded            Torque T_c//
        return,
        }
// Control routine //
void controller(float mes_pos)
        {
        static double ipart
        X_[0]=A[0][0]*pos+A[0][1]*vel+Bkal[0][0]*Tm,
        X_[1]=A[1][1]*vel+A[1][2]*Td+Bkal[1][0]*Tm,
        X_[2]=A[2][2]*Td,
        e=asin(sin(mes_pos)*cos(X_[0])-cos(mes_pos)*sin(X_[0])),
        pos=X_[0]+L[0][0]*e,
        vel=X_[1]+L[1][0]*e,
        Td=X_[2]+L[2][0]*e,
        Pref=Am[0][0]*Pref+Am[0][1]*Vref
        Vref=Am[1][0]*Pref+Am[1][1]*Vref+Bm[1][0]*Td,
        pos1=(Pref-pos)*kp,
        vel2=(Vref-vel)*kw,
        U=pos1+vel2-(Td/ki),
        Tm=saturate(U,1),
        return,
```

```
        }
display_main()
        {
        double tmp,
        char c,
        if(echo==DEF) display(),
        if(echo==CON)
        {
                printf("Command>>"),
                gets(command),
                ptr=command
                switch(*ptr)// Varying Controller and Estimator
                        Parameters on command line //
                {
                case 'D' echo=DEF,              break,
                case 'p' kp=atof(++ptr),        break,
                case 'w' kw=atof(++ptr),        break,
                }
                case 'K' switch(*(++ptr))
                {
                -       ++ptr,
                        case '1' Kss[0][0]=atof(++ptr),    break,
                        case '2' Kss[1][0]=atof(++ptr),    break,
                        case '3' Kss[2][0]=atof(++ptr),    break,
                }
                break,          -
        }
        }
```

APPENDIX D

Servo Motor Control Block Diagram

BHT MOTOR CONTROL
BLOCK DIAGRAM
FIG D1   188 02