CONTROL OF A LEADSCREW DRIVEN FLEXIBLE ROBOT ARM

by
Nabil Georges Chalhoub

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Mechanical Engineering)
in The University of Michigan
1986

Doctoral Committee:

Assistant Professor A. Galip Ulsoy, Chairman
Professor Milton Chace
Professor N. Harris McClamroch
Professor Richard Scott

ensm
UMR1114

ABSTRACT

CONTROL OF A LEADSCREW DRIVEN FLEXIBLE ROBOT ARM

by

Nabil Georges Chalhoub

Chairman: A. Galip Ulsoy

High performance requirements in robotics have led to the consideration of structural flexibility in robot arms. Here the dynamics of a spherical coordinate robot arm, whose last link is very flexible, is studied for the purpose of control. The assumed modes method is employed to represent the deflection of any point on the flexible link, and the Lagrangian approach is used to obtain the unconstrained equations of motion. The robot arm, considered in this work, has two joints driven by a leadscrew transmission mechanisms. The kinematic constraints associated with the nonbackdrivable leadscrews are also considered.

An integral plus state feedback controller is derived based on a linearized version of the rigid body model of the robot arm. The controller is then implemented on the rigid and flexible model. The rationale is to simulate the controllers currently used in industrial robots and to assess the interrelationships

between the robot arm structural flexibility and the controller design. The simulation results illustrate the differences between leadscrew driven and unconstrained axes of the robot; they indicate the trade-off between speed and accuracy; and show potential instability mechanisms due to the interaction between the controller and the robot structural flexibility.

The integral plus state feedback controller, derived for the rigid body model of the robot arm, is extended to include the flexible motion. The objective is to introduce additional damping into the flexible motion. This is done by using additional sensors to measure the compliant link vibrations and feed them back to the controller. The effect of control and observation spillover is examined, and found not to present a serious practical problem. The simulation results show that additional damping in the flexible motion can be achieved by including the flexible motion in the control action. Experimental evaluation showed excellent agreement with the results of the digital simulations.

To my parents

ACKNOWLEDGMENTS

I would like to thank all the members of my dissertation committee for their interest in my work and their helpful comments. I would also like to extend my appreciation and gratitude to my advisor, A. Galip Ulsoy, for his sincere and excellent guidance.

A special thanks goes to the Robotics division of the Center for Research in Integrated Manufacturing (CRIM) at the University of Michigan for providing me with financial support throughout the duration of my thesis work. The help of their technical staff, John Caldwell, Jerry Turney and Rob Giles, is very much appreciated. I would also like to extend my thanks to Mary Ann Pruder and Marianne Moylan for typing my dissertation and all the related technical reports.

Finally, a special thanks goes to my parents for their encouragement and their unfailing support.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF APPENDICES

Appendix

CHAPTER 1

INTRODUCTION

This chapter provides an overview of the research in robot arm modeling, dynamics, and control. The motivation for high precision and high speed robot operation is discussed and some of the problems are identified. The current state-of-the art in robot arm dynamic modeling and control is reviewed, including studies of robot arm flexibility. Finally, the objectives of this study are set forth, and the organization of the dissertation is summarized.

1.1. Motivation

Until the 1980's, the use of robots was primarily limited to industrial and research environments which are hazardous or unpleasant for human workers. In recent years, the evolution of the mechanical arm from teleoperator and crane to present day industrial and space robots has increased their implementation in almost every industrial area. However, the poor endpoint positional accuracy of current robots have restricted their applications to tasks that are error tolerant. Many tasks, such as assembly of high precision parts require positional tolerances on the order of 0.0254mm (0.001"). Existing manipulators can perform tasks with a repeatable position on the order of 0.254mm (0.01").

The operation of high-precision robots is severely limited by the dynamic deflection which persists for a period of time after a move is completed. The settling time required for this residual vibration delays subsequent operations, thus conflicting with the demand for increased productivity. These conflicting requirements have rendered the robotic assembly task to be a challenging research problem.

The performance problems in robotics are related to the transmission mechanisms (i.e. backlash, friction and compliance), the structural deformation (both static and dynamic), and the massive design of the robot arm.

1.2. Previous Work in the Dynamic Modeling of Robots

The automation of assembly tasks will be greatly enhanced if robots can operate at higher speeds with greater positioning accuracy. These goals cannot be achieved with the existing massive robot designs, which make them slow and heavy. Many robot arms are made to be massive for increased rigidity. To improve their structural performance, payloads and operating speeds are kept fairly low. In other words, the rigid body assumption is justified by conceeding low performance.

For higher operating speeds, mechanisms should be made light-weight to reduce the driving torque requirements and enable the robot to respond faster. However, lighter members are likely to elastically deform, thus making it a necessity to take into consideration the dynamic effects of the distributed link flexibility. This is because high speed operation leads to high inertial forces which in turn cause vibration and deteriorate accuracy. This difficulty may be eased by fabricating the moving members of manipulators in fiber reinforced composite materials which can result in high structural stiffness and strength with low mass [1]; however the basic problem remains.

In an attempt to improve the overall performance of manipulators, some researchers have treated the aforementioned problems separately. A direct drive arm is used by Asada [2] to solve the backlash and friction problems. The compliance problem in the drive mechanism is discussed by Kuntze and Jacubusch [3]. Zalucky and Hardt [4] used a micromanipulator to correct for the link static deflection.

The first step in improving the performance of robots is to obtain a reasonably accurate dynamic model. The general practice has been based on rigid body analysis where all the robot links are assumed to be rigid [5-9]. This is understandable since even the rigid body model is complex, nonstationary and highly nonlinear. However, when the robot is operated at high speed, the vibration induced by the large inertial and external forces can no longer be neglected and the rigid body assumption becomes invalid.

Until recently, the inclusion of the link's flexibility in the dynamic model of the robot arm is done by solving for the rigid body motion and the flexible motion separately [10]. Generally the flexible motion is small, and its effect on the rigid body motion is neglected. Therefore, only the effect of the rigid body motion on the flexible motion is considered. This is done by solving the rigid body equations for inertial forces which in turn are introduced as excitation sources to the elastic problem. This approach is adequate for modeling fairly rigid structures. However, an accurate dynamic model for a very flexible structure would require all the coupling terms between the flexible and the rigid body motions to be retained. This is done by using coupled reference position and elastic deformation models. Some researchers have used finite element techniques to describe the elastic deformation [11-16]. Others have used global methods such as the assumed modes method, Rayleigh-Ritz method, etc. [17-18]. These approximate techniques can be used to yield a set of equations which represent the combined rigid and flexible motion.

1.3. Overview of Robot Controller Design

Robots are inherently very complex structures. Their rigid body dynamic models exhibit a nonlinear and nonstationary behavior. In today's robots, the desired end-effector position is converted by real-time kinematic computation into the equivalent angles that each joint must assume [19]. The joints are then simultaneously driven to their assigned angles by separate conventional servo loop controllers. However, the implementation of conventional linear control techniques have led to poor performance because of both the inherent geometric nonlinearities of these systems, and the dependence of the system dynamics on the characteristic of the manipulated objects. Therefore, a sophisticated controller design is needed to ensure the robot desired performance.

In the control of rigid robots, adaptive, nonlinear feedback, and optimal control techniques have been investigated. Adaptive control theory [20-25] has been proposed as a promising solution to the nonlinearity and nonstationarity problems. The main task of the adaptive controller is to adjust the feedback gains of the manipulator so that its closed loop performance characteristics closely match the desired ones. However, the large required computation time has restricted their applications to simulation studies.

Nonlinear feedback control, or the "computed torque" method [26], has led to better performance over conventional control techniques in computer simulations. The controller is based on an idealized model of the manipulator. This is a severe drawback because when the "idealized" control law is used with an actual plant, the validity of the scheme is subject to doubt and the robustness of the controller becomes of great importance.

A time-optimal control strategy is studied in [27] to enable faster movement of the robot arm. Kahn and Roth [28] showed that this technique results in a two-point boundary value problem. As a consequence, the computation must be repeated for each new set of initial and final conditions used in the numerical solution. In addition, the numerical algorithm yields an optimal solution that is a function of time and does not account for any unexpected disturbances which may act on the system. These drawbacks have rendered the implementation of the time-optimal control technique difficult, if not impossible, for real time application in feedback control of robots. However, this technique has turned out to be a powerful tool for off-line trajectory planning.

In an attempt to relax these difficulties, a suboptimal control strategy is often used in digital simulations. In this technique, linearized version of the dynamic model of the manipulator is obtained for which an analytical optimal control solution can be found. Many of these sophisticated control techniques suffer from excessive computation time requirements which make them unattractive for real time applications using current microprocessor technology [29].

Besides the difficult issues encountered in rigid robots, a new problem is created when robot compliance is included in the dynamic model. It emanates from the distributed nature of the link mass and elasticity. An infinite number of degrees of freedom are required to specify the position of every point on the elastic link. However, due to physical limitations, only a finite number of sensors and actuators can be mounted on the flexible body. Thus resulting in the problem of controlling a large dimensional system with a smaller dimensional controller [30-31].

Meckl and Seering [32-33] proposed two types of forcing functions to achieve fast response with negligible residual vibrations. The first type allows vibration to occur during the move but would stop the motion in such a way as to eliminate any residual vibration. This consists of a bang-bang action with multiple switching points. The second type avoids the excitation of the resonant modes of the structure as it moves. This is done by using a forcing function with no frequency components matching the system natural frequencies. This work is still confined to research laboratories; it can be used off-line to generate the trajectory of the robot arm that satisfies a criterion considering both the response time and the residual vibration.

By and large, the research in the closed loop control of flexible manipulators can be divided into two categories. The first uses additional sensors to measure the flexible motion (i.e. assumes all state variables to be available) [34-37]. This enables the inclusion of the flexible motion in the control action, thus achieving better positional accuracy with the existing joint motors. The second category employs a micromanipulator along with additional sensors to compensate for static and dynamic structural deflections [4], [38-39]. This concept gives the control system designer more capabilities to improve the robot arm performance at additional hardware cost.

1.4. Dissertation Overview

The purpose of this work is to improve the end-effector positional accuracy by measuring and feeding back its transverse vibration. The spherical coordinate robot arm considered in this study is schematically illustrated in Fig. 1-1.

Two types of controllers are implemented. The first is a conventional linear controller based on the rigid body model. The rationale is to simulate the controllers currently used in industrial robots and to assess the interrelationships between the robot arm flexibility and the controller design. The second control system uses the measured flexible motion to introduce additional damping in the flexible structure.

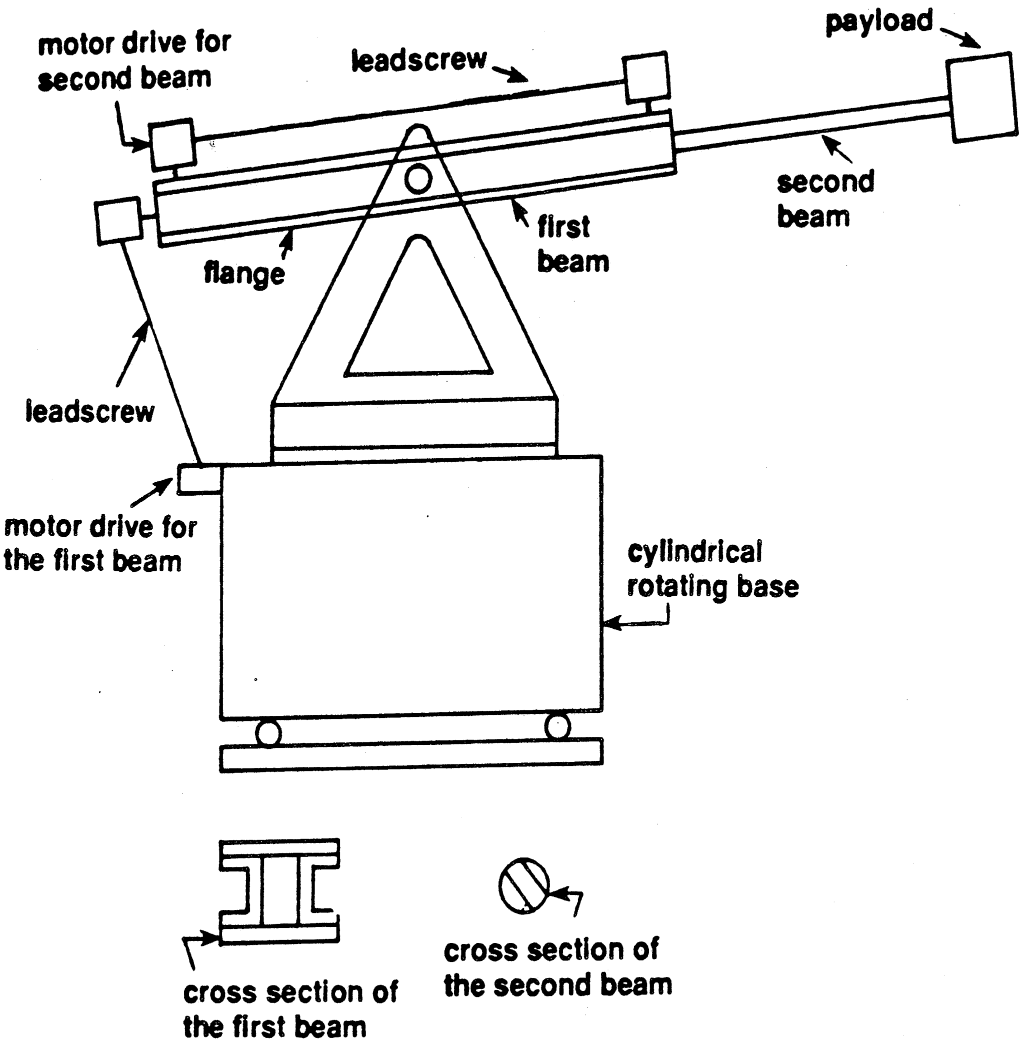


Figure 1 -1. A schematic of the physical system.

In the next chapter, the dynamic modeling of the robot arm including the transverse elastic displacements of the last link is presented in detail. The derivation of the constraint equations imposed by the leadscrews are derived in Chapter 3.

Chapter 4 includes the design of the rigid body controller and the results of the investigation of the interrelationships between the structural flexibility and the controller design.

In Chapter 5, the rigid and flexible motion controller is designed. The separate and combined effects of control and observation spillover are shown.

Chapter 6 presents the experimental apparatus along with the corresponding results. A comparison and discussion of the theoretical and experimental results is also made.

Finally, Chapter 7 summarizes the work done, presents the main conclusions, and states some prospective research topics.

CHAPTER 2

DYNAMIC MODELING OF THE ROBOT ARM

Typically, the dynamic modeling of a flexible robot arm considers the flexible motion to be small and neglects its effect on the rigid body motion. This technique gives an accurate model for fairly rigid structures. However, in the case of very flexible manipulators, all the coupling terms between the rigid and the flexible motions need to be retained. This is done by using coupled reference position and elastic deformation models.

In this chapter, the dynamics of a spherical coordinate robot arm, whose last link is flexible, is studied for the purpose of control. The assumed modes method is employed to represent the deflection of any point on the flexible link. Lagrange's method represents a very systematic approach to obtain the equations of motion of such a complicated system in the simplest manner possible.

2.1. Modeling

The physical system considered in this work is a spherical coordinate robot, which has two revolute joints and one prismatic joint (see Fig. 1-1). It consists of an arm connected to a gear driven rotating base. The arm is made of two beams such that the second beam can move axially into the first. The entire arm is free to rotate around the horizontal axis passing though the pivot point 0 and parallel to the unit vector $\underset{\sim}{K}$ (see Fig. 2-1). The r and θ coordinates have leadscrew transmission mechanisms, whereas power is transmitted to the ϕ coordinate through a gear train. All axes are driven by dc servo motors.

The second beam consists of a thin, lightweight and very flexible aluminum rod. The rationale is to exaggerate the flexibility problem which leads to a better understanding of the interaction between the structural flexibility and the controller design. This will also narrow the gap between the natural frequencies of the elastic modes and the servo loop frequencies, thus reducing the numerical computation problem associated with stiff systems and the problem of sampling rapidly enough to characterize the high frequencies of the flexible arm.

The first and second beams of the robot arm have length L_1^t and L_2 respectively as shown in Fig. 2-1. A reference inertial frame $(\underset{\sim}{I}, \underset{\sim}{J}, \underset{\sim}{K})$ fixed at point 0, is shown along with a non-inertial, body fixed, rotating reference frame $(\underset{\sim}{i}, \underset{\sim}{j}, \underset{\sim}{k})$. The equations are derived in terms of the latter to keep the mass moment of inertia constant throughout the rigid body motion of the robot.

The robot arm can rotate with angular velocity $\dot{\phi}$ around the vertical axis passing through $\underset{\sim}{J}$ and $\dot{\theta}$ around the horizontal axis along $\underset{\sim}{k}$. The second beam has one additional degree of freedom r which allows it to move axially into the first beam. The payload at the end-of-arm is modeled by a concentrated mass m_p at the end of the second beam.

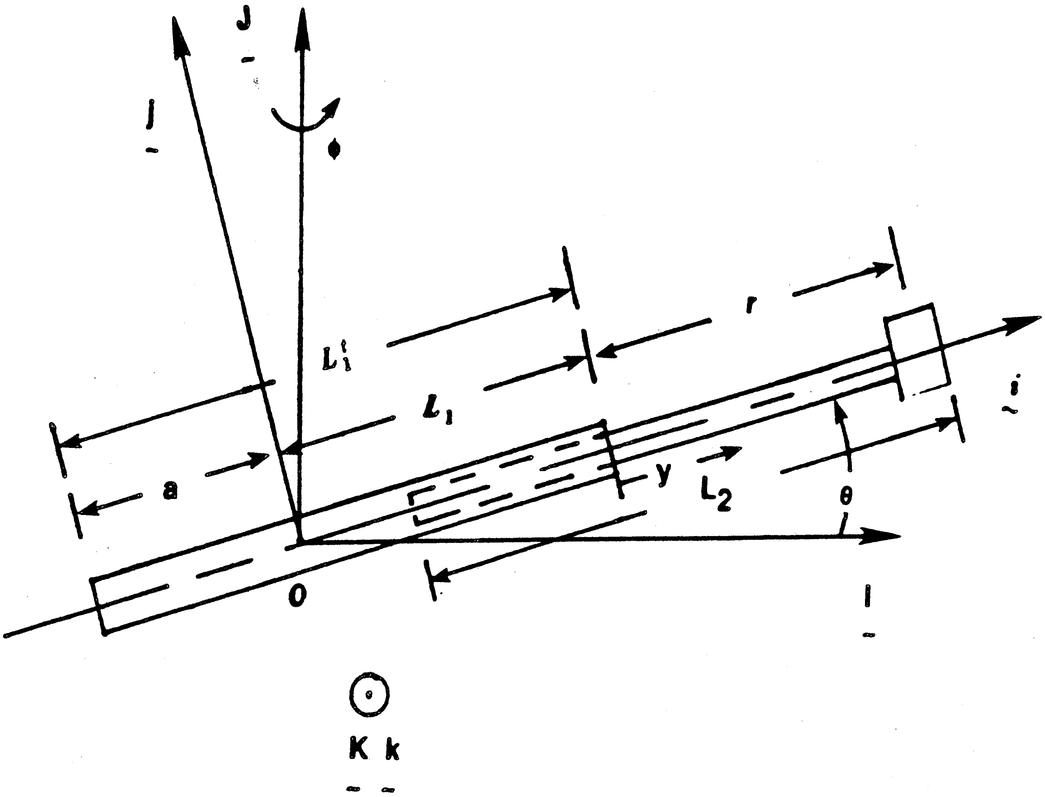


Figure 2-1. Arm geometry and coordinates.

The general modeling problem for articulated flexible manipulators has been studied by others [11-16]. The constraint equations for a slot joint which allows for a combined translational and rotational motions is discussed in [40]. In this study, the prismatic joint constraint is handled by considering the portion of the second beam protruding from the first beam to have a flexible cantilever beam like behavior, whereas the first beam and the part of the second beam located inside the first beam are considered to undergo rigid body motion only. Referring to Fig. 1-1, this modeling assumption is justified by the structural properties of the first beam which is stiff in flexure. The protruding portion of the second beam is relatively flexible. It is more rigid in compression than in flexure. Therefore, the longitudinal vibrations of the flexible portion of the second beam are neglected and only transverse vibrations are considered in addition to the rigid body motion. Since the elastic deformation is small, the variations of the mass moment of inertia due to the flexible motion are assumed to be negligible.

2.2. Equations of Motion

Lagrange's equation [41-42] is implemented to obtain the dynamic model of this complicated system. This method considers the system as a whole rather than its individual components, thus eliminating the need to calculate interacting forces. It formulates the problem in terms of two scalar functions, namely, work and kinetic energy.

The position vector of the mass center for the rigid portion of the robot arm becomes important in the derivation of its kinetic energy expression. For the first beam, one has,

$$\underset{\sim}{R}^{*}_{1} = \left(\frac{L_1^t}{2} - a \right) \underset{\sim}{i} \tag{2-1}$$

where the star superscript indicates a value related to the center of mass. L_1^t is the total length of the first beam. a is the length of the portion of the first beam lying to the left of the pivot point 0 (see Fig. 2-1). The position vector of the mass center of the part of the second beam located inside the first beam is

$$\underset{\sim}{R}^{*}_{2} = x'' \underset{\sim}{i} \tag{2-2}$$

where $x'' = L_1 + \frac{r}{2} - \frac{L_2}{2}$, L_1 is the length of the portion of the first beam lying to the right of the pivot point 0, L_2 is the total length of the second beam, and r is the length of the part of the second beam protruding from the first beam. The position vector of any point on the protruding part of the second beam is,

$$\underset{\sim}{R}_{3} = y' \underset{\sim}{i} + V\underset{\sim}{j} + W\underset{\sim}{k} \tag{2-3}$$

where $y' = L_1 + y$ and $\dot{y}' = \dot{r} = \dot{y}$. The assumed modes method [41] is used to obtain the expressions for the transverse vibrations V and W. They are written as a linear combination of admissible functions, $\Phi_i(y)$, of spatial coordinates, multiplied by time dependent generalized coordinates $q_{ij}(t)$.

$$V = \sum_{i=1}^{n} \Phi_i(y) q_{1i}(t) \tag{2-4}$$

$$W = \sum_{j=1}^{n} \Phi_j(y) q_{2j}(t)$$

An approximation to the dynamics of infinite dimensional systems is often based on the fact that low frequencies alone are adequate to describe the flexible behavior of the system. Therefore, n is selected to be 2, and the admissible functions are chosen to be the first two eigenfunctions of a clamped-free beam. Referring to Fig. 1-1, these boundary conditions are chosen because the second beam slides inside the first beam and is supported by a teflon sleeve which only permits axial motion. The clamped-free beam eigenfunctions Φ_1 and Φ_2 have the following general form which is described in [43], [44].

$$\Phi_i = \cos h\left(\frac{\epsilon_i y}{r}\right) - \cos\left(\frac{\epsilon_i y}{r}\right) - \alpha_i \left[\sin h\left(\frac{\epsilon_i y}{r}\right) - \sin\left(\frac{\epsilon_i y}{r}\right)\right] \tag{2-5}$$

where the values of ϵ_i and α_i for each mode are given in [43]. Note that an additional approximation has been introduced in that the solution is not separable in space and time when $r(t)$ is not constant. However, this forms a useful approximation as long as $r(t)$ satisfies the Euler-Bernoulli beam assumptions. That is r must always be much greater than the beam cross sectional dimensions, the deflection terms V and W don't exceed one tenth of r, and rotary inertia and shear deformation effects are ignored. Substituting (2-4) and (2-5) into (2-1) to (2-3) gives the complete form for the position vectors. These results are then used to obtain the velocity terms from:

$$\dot{\underset{\sim}{R}}_i = \frac{d\underset{\sim}{R}_i}{dt} + \underset{\sim}{\Omega} \times \underset{\sim}{R}_i \tag{2-6}$$

where $\underset{\sim}{\Omega}$ is the rotation vector of the $(\underset{\sim}{i}, \underset{\sim}{j}, \underset{\sim}{k})$ basis and $\underset{\sim}{R}_i$ is the position vector of any particle along link i.

The velocity terms are then used to develop the kinetic energy expressions for each component of the robot-arm including the payload, m_p. The latter is considered to be a point mass, and does not include rotation around its own mass center. The total kinetic energy will be

$$T_t = \sum_{i=1}^{m} T_i + T_p = \sum_{i=1}^{m} \frac{1}{2} \int_{m_i} (\dot{\underset{\sim}{R}}_i \cdot \dot{\underset{\sim}{R}}_i) dm_i + \frac{1}{2} m_p (\dot{\underset{\sim}{R}}_p \cdot \dot{\underset{\sim}{R}}_p) \tag{2-7}$$

where m is the number of links in the robot arm, T_i, is the kinetic energy of link i and T_p is the kinetic energy of the payload.

The total potential energy consists of the strain energy of the second beam [45] and the potential energy associated with the rigid body motion. The strain energy includes terms due to flexural rigidity.

$$\frac{1}{2} \int_0^r EI(y) \left[\frac{\partial^2 V(y,t)}{\partial y^2} \right]^2 dy + \frac{1}{2} \int_0^r EI(y) \left[\frac{\partial^2 W(y,t)}{\partial y^2} \right]^2 dy \tag{2-8}$$

and terms due to the axial force:

$$\frac{1}{2} \int_0^r T(y,t) \left[\frac{\partial V(y,t)}{\partial y} \right]^2 dy + \frac{1}{2} \int_0^r T(y,t) \left[\frac{\partial W(y,t)}{\partial y} \right]^2 dy \tag{2-9}$$

where $T(y,t)$ is the axial force in the second beam which arises due to gravity and inertial forces. The latter consist of centripetal acceleration and flexible motion effects. However, the variation of the axial force due to the flexible motion is neglected and one obtains,

$$T(y,t) = \rho A_2(\dot{\phi}^2\cos^2\theta + \dot{\theta}^2)(L_1 r + \frac{r^2}{2}) \left[1 - \frac{L_1 y + \frac{y^2}{2}}{L_1 r + \frac{r^2}{2}}\right]$$

$$+ m_p(L_1 + r)\left[\dot{\phi}^2\cos^2\theta + \dot{\theta}^2\right] + \left[m_p + \rho A_2(r - y)\right] g\sin\theta \tag{2-10}$$

A compressive $T(y,t)$ would tend to decrease the transverse stiffness of the beam while a tensile $T(y,t)$ would have the opposite effect. The total potential energy will be:

$$V_t = \left\{ m_1\left(\frac{L_1^t}{2} - a\right) + m_{l1}d_{l1} + m_{H1}(L_1 + r - L_2) + m_2\left[L_1 + r - \frac{L_2}{2}\right]\right.$$

$$\left. + m_p(L_1 + r)\right\} g\sin\theta + \rho A_2 g\cos\theta \int_0^r V(y)dy + m_p V \mid_{L_1+r} g\cos\theta$$

$$+ \frac{1}{2}\int_0^r EI(y)\left[\frac{\partial^2 V(y,t)}{\partial y^2}\right]^2 dy + \frac{1}{2}\int_0^r EI(y)\left[\frac{\partial^2 W(y,t)}{\partial y^2}\right]^2 dy$$

$$+ \frac{1}{2}\int_0^r T(y,t)\left[\frac{\partial V(y,t)}{\partial y}\right]^2 dy + \frac{1}{2}\int_0^r T(y,t)\left[\frac{\partial W(y,t)}{\partial y}\right]^2 dy \tag{2-11}$$

where m_{l1} and d_{l1} are respectively the mass and the distance from the $(\underset{\sim}{i}, \underset{\sim}{j}, \underset{\sim}{k})$ origin to the mass center of the leadscrew that drives the second beam and the m_{H1} is its housing's mass.

The coefficients in the kinetic and potential energy expressions, given in equations (2-7) and (2-11), involve the integration of complicated functions with respect to the spatial coordinate y. Some of the terms encountered in the kinetic energy expression have the following form,

$$\int_0^r \Phi_i(y)dy \qquad \int_0^r y\Phi_i(y)dy$$

$$\int_0^r \Phi_i(y)\cdot\Phi_j(y)dy \qquad \int_0^r \Phi_i(y)\dot{\Phi}_j(y)dy$$

$$\int_0^r \dot{\Phi}_i(y)dy \qquad \int_0^r y\dot{\Phi}_i(y)dy$$

where $i=1,2$ and $j=1,2$. The integration terms appearing in the potential energy expression can be written in the following general form,

$$\int_0^r \left[\frac{d^2\Phi_i(y)}{dy^2}\right]\cdot\left[\frac{d^2\Phi_j(y)}{dy^2}\right]dy \quad for\begin{cases} i=1,2 \\ j=1,2 \end{cases}$$

where $\Phi_i(y)$ is the eigenfunction previously defined in equation (2-5). These complex functions are integrated numerically using Gaussian quadrature of high order (40 points) [46].

The virtual work principle is implemented to obtain an expression for the generalized forces. If the forces exerted on the robot arm are divided into conservative forces, which are directly related to the potential energy, V_t, and nonconservative forces which are not, then the virtual work expression could be written as, [41]

$$\delta W = \delta W_c + \delta W_{nc} \tag{2-12}$$

where the subscripts c and nc denote terms due to the conservative and nonconservative forces respectively. δW_c is related to the potential energy V_t as follows:

$$\delta W_c = -\delta V_t = -\sum_{i=1}^{n} \frac{\partial V_t}{\partial x_i} \delta x_i \tag{2-13}$$

where x_i is the i^{th} generalized coordinate that can be either one of the rigid body degrees of freedom r, θ and ϕ or one of the flexible motion coordinates q_{11}, q_{12}, q_{21} and q_{22}. The free body diagram of the robot arm, which shows the nonconservative forces, is depicted in Fig. 2-2. The expression for δW_{nc} can be easily obtained.

$$\begin{aligned} \delta W_{nc} &= \sum_{i=1}^{n} Q_{inc}\,\delta x_i = \underset{\sim}{R} \cdot \delta \underset{\sim}{R}{}^{0}_{1} \big|_{x_1=0} + F_c \underset{\sim}{i} \cdot \delta \underset{\sim}{R_2} \big|_{L_1+r-L_2} \\ &+ T_3 \underset{\sim}{J} \cdot \delta \underset{\sim}{\chi_1} + F_{ax} \underset{\sim}{j} \cdot \delta \underset{\sim}{R}{}^{L}_{1} \big|_{-a} \\ &= F_c\,\delta r - aF_{ax}\,\delta\theta + T_3\delta\phi \end{aligned} \tag{2-14}$$

where Q_{inc} is the i^{th} nonconservative generalized force. $\delta \underset{\sim}{\chi_1}$ is the virtual rotation vector of $\underset{\sim}{\Omega}$. $\underset{\sim}{R}$ is the reaction force at the pivot point. F_c, F_{ax} and T_3 are the

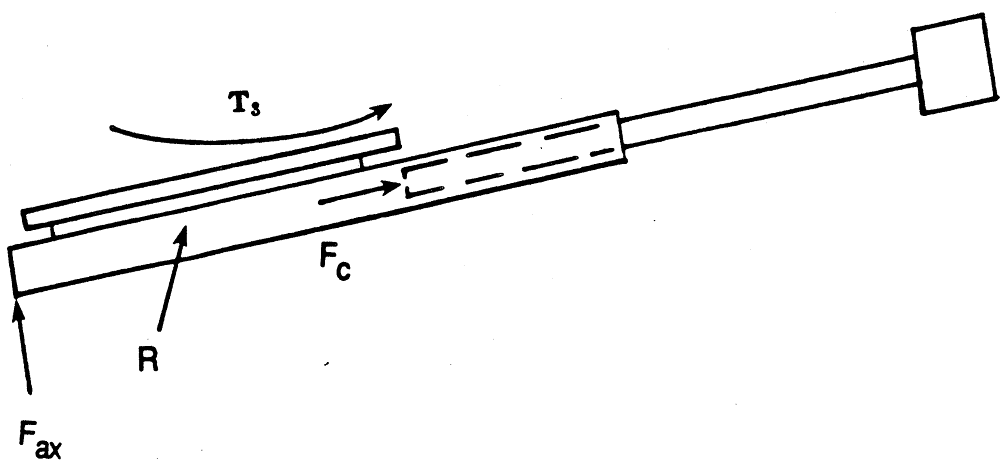


Figure 2-2. Free body diagram of the robot arm.

joint driving force and torques. Now that the kinetic energy, potential energy and the virtual work expressions are derived, the equations of motion can be readily obtained by implementing the following form of Lagrange's equation

$$\frac{d}{dt}\left[\frac{\partial T_t}{\partial \dot{x}_i}\right]-\frac{\partial T_t}{\partial x_i}+\frac{\partial V_t}{\partial x_i}=Q_{inc} \tag{2-15}$$

The resulting equations of motion are seven highly nonlinear, coupled, second order ordinary differential equations. The reader is referred to Appendix A for the listing of the unconstrained equations.

2.3. Summary

A brief description of the physical system is given along with the definition of the coordinate axes and the generalized coordinates. The assumed modes method is implemented to model the flexible motion of the last link. Expressions for the virtual work, kinetic and potential energies are derived. Lagrange's equation is employed to obtain the unconstrained equations of motion.

Some of these equations can be greatly simplified once the role played by the leadscrews and the effect of the rigid body motion on the flexible motion and vice-versa, are understood. These leadscrew constraints are considered in the next chapter.

CHAPTER 3

LEADSCREW DYNAMIC MODELING AND CONSTRAINTS

The robot arm considered in this work has two joints driven by leadscrew transmission mechanisms which are used to transmit power by converting angular motion to linear motion. In this chapter, the kinematic constraints associated with a leadscrew are introduced to investigate the behavior of a leadscrew driven flexible robot arm in the presence of coulomb friction and the self locking condition (i.e., the nonbackdrivability of the leadscrew).

3.1. Overview of the Static Model of a Leadscrew

A detailed derivation of the static model of a leadscrew can be found in [47]. A brief summary of the modeling procedure is included to provide the required background for the next section.

A free body diagram of an unrolled single thread of a leadscrew is shown in Fig. 3-1. F_{ax} is the resultant of all the forces exerted on the leadscrew in the axial direction. N is the normal force and P is the force required to raise or lower the axial load, F_{ax}, on the thread surface. The friction force, f_f, is considered to be of the simplest form of dry coulomb friction,

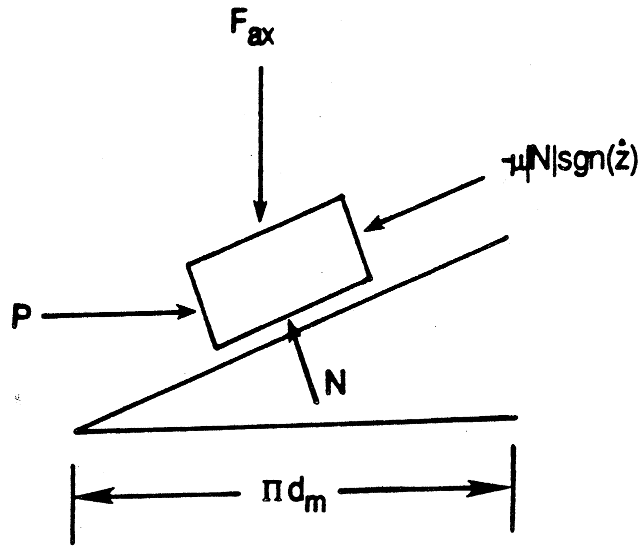


Figure 3-1. Free body diagram of an unrolled thread.

$$f_f = -\mu \mid N \mid sgn(\dot{z}) \tag{3-1}$$

where μ is the coefficient of friction and $\dot{z}$ represents the velocity along the rough surface of the leadscrew thread. The system is in equilibrium under the action of these forces. Therefore, the minimum force, P, required to raise or lower the axial load, F_{ax}, can be obtained from the summation of forces in the vertical and horizontal directions. The expression for P to raise the load has the following form,

$$P = \frac{F_{ax}\left[\dfrac{l}{\pi d_m} + \mu_s\right]}{1 - \left(\dfrac{\mu_s l}{\pi d_m}\right)} \tag{3-2a}$$

where μ_s is the static coefficient of friction. To lower the load, P becomes

$$P = \frac{F_{as}\left[\mu_s - \left(\frac{l}{\pi d_m}\right)\right]}{1 + \left(\frac{\mu_s l}{\pi d_m}\right)} \qquad \textbf{(3-2b)}$$

where l is the lead of the leadscrew and d_m is its mean diameter. The expression given in (3-2) represent the force required to overcome static friction and the geometric effect. This force, P, can be directly related to the torque, T, applied on the leadscrew by,

$$T = \frac{Pd_m}{2} \qquad (3\text{-}3)$$

In the case where $\mu_s < \left(\frac{l}{\pi d_m}\right)$ in (3-2b), the force required to lower the axial load becomes negative. This causes the screw to spin without external effort. However, most leadscrews are nonbackdrivable in the absence of the applied force, P. This results in satisfaction of the following inequality,

$$\mu_s > \tan \psi_1 = \frac{l}{\pi d_m} \qquad (3\text{-}4)$$

where ψ_1 is the thread helix angle. The inequality in (3-4) is often referred to as the self-locking condition.

3.2. Generalization of the Static Model

The static model of the leadscrew derived in [47] is useful for determining the torque required to overcome static friction and geometric effects for the load configuration given in Fig. 3-1. This model needs to be modified if different configurations are considered.

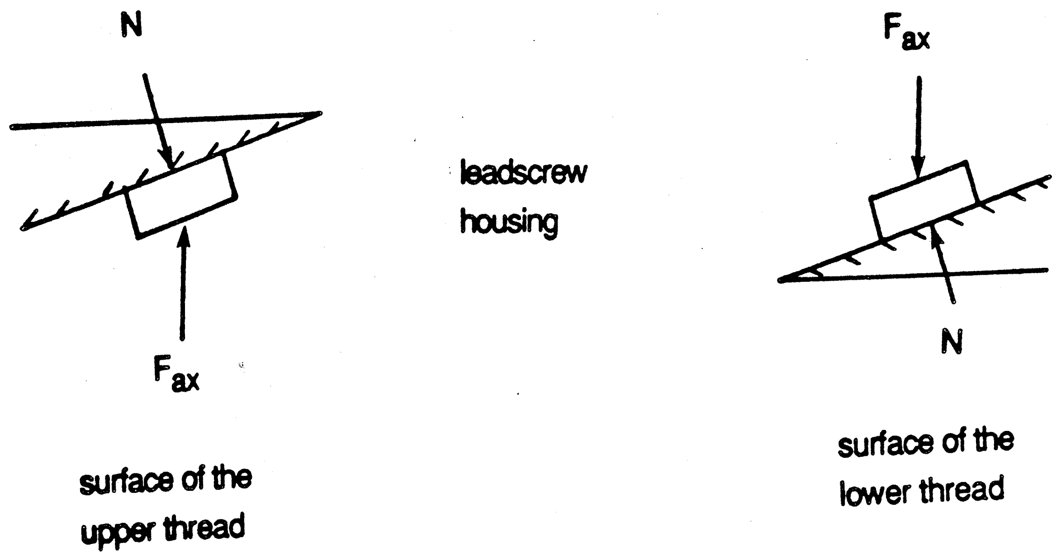


Figure 3-2. Transition of the leadscrew housing motion from lower to upper thread depending on the sign of the normal force, N.

In this work, the leadscrew is assumed to satisfy the self-locking condition. The friction force has the same form as in (3-1) and the axial force, F_{ax}, can be either tensile or compressive. As a consequence, the normal force, N, can now be positive or negative. This sign change in N can be physically interpreted as having the leadscrew housing sliding on the upper thread's surface if it was originally on the lower surface or vice versa. This is illustrated in Fig. 3-2. The equations of motion for both the upper and lower thread surfaces can be combined in the following general form,

$$-F_{ax}\sin\psi_1 - \mu\ |N|\ sgn(\dot{z}) + P\cos\psi_1 = m_f\ \ddot{z} \tag{3-5}$$

where $m_f = 0$ is a fictitious mass, and z is the coordinate along the thread surface. It can be directly related to the rigid body degree of freedom driven by the corresponding leadscrew through simple geometric relations. The expression for the normal force, N, is obtained from the summation of forces in the vertical direction,

$$N = F_{az} \cos \psi_1 + P \sin \psi_1 \tag{3-6}$$

In this work, the axial load, F_{az}, consists of inertial and gravitational terms. It can be written as

$$F_{az} = F_{az}(\underset{\sim}{z}, \dot{\underset{\sim}{z}}, \ddot{\underset{\sim}{z}}, g) \tag{3-7}$$

where $\underset{\sim}{z}^T = [r, \theta, \phi, q_{11}, q_{12}, q_{21}, q_{22}]$ is the generalized coordinate vector and g is the gravitational acceleration. Substituting (3-6) and (3-7) into (3-5), the general form of the equation of motion becomes

$$\ddot{z}_i = f_i\,[\underset{\sim}{z}, \dot{\underset{\sim}{z}}, g, abs(\underset{\sim}{z}, \dot{\underset{\sim}{z}}, \ddot{\underset{\sim}{z}}, g)] \tag{3-8}$$

where abs $(\underset{\sim}{z}, \dot{\underset{\sim}{z}}, \ddot{\underset{\sim}{z}}, g)$ function represents the absolute value of the normal force N in (3-5). The term $\ddot{z}_i$ can be either $\ddot{r}$ or $\ddot{\theta}$ since only two of the rigid body degrees of freedom r and θ are driven by leadscrew transmission mechanisms. The f_i's are very complex, highly nonlinear functions which do not have closed form solutions. Instead, they are solved numerically on a digital computer. However, most numerical algorithms require that the equations be written in either of the following forms,

$$\ddot{\underset{\sim}{z}} = f(\dot{\underset{\sim}{z}}, \underset{\sim}{z}, t) \tag{3-9}$$

or reduced to the first order form

$$\dot{\underset{\sim}{z}} = f(\underset{\sim}{z}, t) \tag{3-10}$$

where now $\underset{\sim}{z}^T = [r, \theta, \phi, q_{11}, q_{12}, q_{21}, q_{22}, \dot{r}, \dot{\theta}, \dot{\phi}, \dot{q}_{11}, \dot{q}_{12}, \dot{q}_{21}, \dot{q}_{22}]$. However, (3-8) cannot be written in either of these two forms due to the presence of the acceleration vector $\ddot{\underset{\sim}{z}}$ on the right hand side. This difficulty would not exist had it not been for the absolute value term of the normal force, N, in equation (3-5). To overcome this problem, separate equations are written to describe the motion on the upper and lower thread surfaces of the leadscrew. By writing the force balance equations from the appropriate free body diagram in Fig. 3-2, the lower thread equation of motion can be obtained,

$$m_f \ddot{z} = -F_{az}\left[\sin\psi_1 + \mu\cos\psi_1 sgn(\dot{z})\right] + P\left[\cos\psi_1 - \mu\sin\psi_1 sgn(\dot{z})\right] \tag{3-11}$$

Similar procedures are followed to derive the equation of motion on the upper thread of the leadscrew,

$$m_f \ddot{z} = F_{az}\left[\sin\psi_1 - \mu\cos\psi_1 sgn(\dot{z})\right] + P\left[\cos\psi_1 + \mu\sin\psi_1 sgn(\dot{z})\right] \tag{3-12}$$

This results in having two equations of motion for each degree of freedom driven by a leadscrew mechanism.

3.3. Coulomb Friction Effect

The effect of friction can be clearly identified at the beginning of the motion or when the response of the degree of freedom driven by a leadscrew is oscillatory. If the leadscrew is to rotate as soon as the input torque is applied, then the latter must be large enough to overcome the combined effects of static friction and geometry. Otherwise, the leadscrew would remain at a standstill, thus causing a delay in the system response.

In the case of an oscillatory response, the time at which the maximum overshoot occurs corresponds to the position where the system comes to a complete halt (i.e. velocity is zero). This results in a sudden jump in the required driving torque caused by the sharp increase in the magnitude of the friction force due to the transition from a dynamic to a static coefficient of friction. This may reshape the entire response if the leadscrew is driven by a controller. The latter, being unaware of this sudden increase in the required driving torque, would supply an input torque that is not large enough to overcome the resistive torque of the static friction. As a result, the system would remain at rest waiting for the control effort to build up. Under these conditions, the system is said to fall in the "stopping region." This is illustrated in Fig. 3-3. For some special cases, analytical expressions for the upper and lower bound of that region are developed in [48].

3.4. Self-Locking Condition Effect

In this work, a self-locking condition is assumed; that is, once the leadscrew is subjected to an axial load, it will not rotate unless the applied input torque is large enough to overcome a certain resistive torque, T_R. In the case of compressive axial force, the latter would have the following form,

$$T_R = \frac{F_{ax}\, d_m}{2} \; \frac{(\mu_s + \tan \psi_1)}{(1 - \mu_s \tan \psi_1)} \tag{3-13a}$$

for raising the load, and

$$T_R = \frac{F_{ax}\, d_m}{2} \frac{(\mu_s - \tan\psi_1)}{(1 + \mu_s \tan\psi_1)} \tag{3-13b}$$

for lowering the load. The self-locking condition plays an important role in a multi degree of freedom system where the axial force exerted on the leadscrew is partially due to coupling terms between the degree of freedom driven by the leadscrew and the rest of the system degrees of freedom. In the absence of an input torque, the leadscrew which is nonbackdrivable, will not rotate regardless of the magnitude of the axial force. This renders the degree of freedom driven by the leadscrew transmission mechanism completely decoupled from the rest of the system degrees of freedom. However, this nice property is lost with the application of an input torque. Since the latter is directly related to the axial force which in turn depends on the coupling terms.

To recapitulate, the self-locking assumption of the leadscrew transmission mechanism leads to the following,

1. The degree of freedom driven by the leadscrew is completely decoupled from the rest of the system degrees of freedom in the absence of the input torque.
2. Once the input torque is applied, the coupling terms between the system degrees of freedom are retained through the axial force.

3.5. Constrained Equations of Motion

The gear train that drives the ϕ motion or the base joint is assumed to be ideal (i.e. no backlash or friction). Therefore, the physical system does not impose any constraint on ϕ. In this work, only the leadscrew constraints are con-

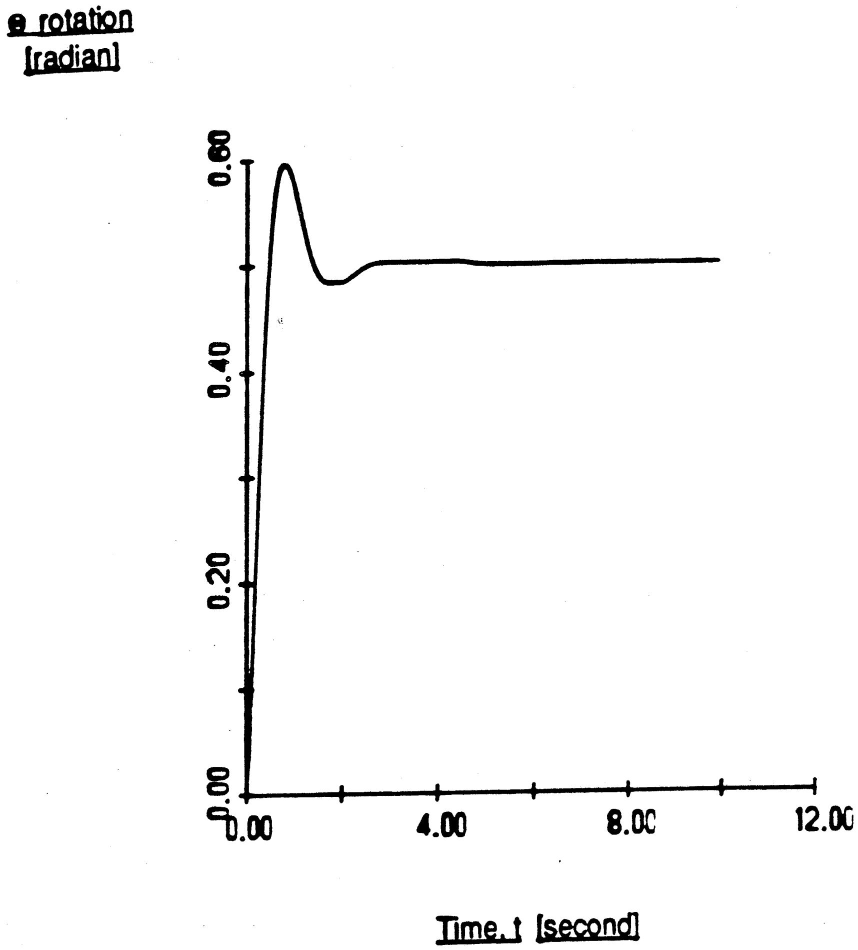


Figure 3-3. Effect of coulomb friction on an oscillatory response

sidered. As a result, the equations of motion for r and θ, derived in Chapter 2 and listed in Appendix A, need to be modified. Newton's method is employed to compute the axial forces exerted on the leadscrews that drive r and θ. The latter, which consist of inertial and gravitational terms, have the general form given in equation (3-7). The constrained r and θ equations of motion are obtained by substituting F_{az} by its value in equations (3-11) and (3-12). The interested reader is referred to Appendix B for the listing of the constrained equations.

In an attempt to check the consistency of these equations, a dynamic model based on the rigid body assumption of the robot arm is derived in appendix C. It shows that the rigid body model can be obtained as a special case from the general equations of motion presented in appendices A and B. This provides a partial confirmation of the validity of the modeling procedure.

3.6. Summary

A brief description of the static model of a leadscrew has been presented along with some modifications to allow for the handling of the dynamic case. Effects of coulomb friction and the self-locking condition on the behavior of a leadscrew driven rigid body degree of freedom are discussed. Finally, the constrained equations of motion for r and θ are derived.

Now that the derivation of the dynamic model of the flexible robot arm is complete, the interrelationships between the structural flexibility and the controller design will be investigated in the next chapter.

CHAPTER 4

RIGID BODY CONTROLLER

This chapter presents the derivation of an integral plus state feedback controller based on a linearized version of the rigid body model of the robot arm. The controller is then implemented on the rigid and flexible model derived in Chapters 2 and 3. The rationale is to simulate the controllers currently used in industrial robots and to assess the interrelationships between the robot arm structural flexibility and the controller design.

4.1. Design Of The Integral Plus State Feedback Controller

In most existing industrial robots, the nonlinear terms in the equations of motion are not considered in the design of the control system. This design practice is valid as long as the robot arm is restricted to slow motion. However, when manipulators are operated at high speed, the effect of geometric nonlinearities and the dependence of system dynamics on the characteristic of the manipulated objects become significant, thus leading to a degradation of the overall robot arm performance.

A conventional linear controller is implemented in this work to simulate the controllers currently used in industrial robots. It is designed based on a linearized version of the rigid body equations of motion derived in Appendix C. Throughout this work, the payload is kept constant and the rigid body degrees of freedom are restricted to the vicinity of the equilibrium state. This is done to reduce the effects of the variations in the manipulated objects and the nonlinear terms on the linear controller. Thus leading to better assessment of the interrelationships between the robot arm structural flexibility and the controller design.

The first step in designing the controller is to express the rigid body equations of motion in terms of state variables which are defined as follows:

$$\begin{array}{lll} y_1 = r & y_2 = \theta & y_3 = \phi \\ y_4 = \dot{r} & y_5 = \dot{\theta} & y_6 = \dot{\phi} \end{array} \qquad (4\text{-}1)$$

The general form of the nonlinear state equations, based on the rigid body equations of motion given in Appendix C, can be written as,

$$\dot{\underset{\sim}{y}} = f(\underset{\sim}{y}, \underset{\sim}{u}) \qquad (4\text{-}2)$$

where $\underset{\sim}{y}$ is the state vector and $\underset{\sim}{u}$ is the control vector. The function f is continuously differentiable in the y_i's. It is expanded in a Taylor series about the equilibrium state $\underset{\sim}{y}_e^T = [r_e, \theta_e, \phi_e, 0,0,0]$ to give,

$$f_i(\underset{\sim}{y}, \underset{\sim}{u}) = f(\underset{\sim}{y}_e, \underset{\sim}{u}_e) + \sum_{j=1}^{n} \left(\frac{\partial f_i}{\partial y_j} \right)_{y_{je}} (y_j - y_{je}) + \sum_{k=1}^{m} \left(\frac{\partial f_i}{\partial u_k} \right)_{u_{ke}} (u_k - u_{ke}) + [\text{ higher order terms }] \qquad (4\text{-}3)$$

Define $\delta u = u_k - u_{ke}$ and $\delta y = y_j - y_{je}$. Substitute (4-3) into (4-2) and retain the first order terms only, to obtain,

$$\delta \dot{\underset{\sim}{y}} = A \delta \underset{\sim}{y} + B \delta \underset{\sim}{u} \tag{4-4}$$

Equation (4-4) represents a first order linear approximation of (4-2). The matrices A and B can be obtained from,

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial y_1} & - & - & \frac{\partial f_1}{\partial y_6} \\ - & & & - \\ - & & & - \\ - & & & - \\ \frac{\partial f_6}{\partial y_1} & - & - & \frac{\partial f_6}{\partial y_6} \end{bmatrix}_{y_e, u_e} \quad \text{and } B = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & - & - & \frac{\partial f_1}{\partial u_3} \\ - & & & - \\ - & & & - \\ - & & & - \\ \frac{\partial f_6}{\partial u_1} & - & - & \frac{\partial f_6}{\partial u_3} \end{bmatrix}_{y_e, u_e} \tag{4-5}$$

Note that each rigid body degree of freedom driven by a leadscrew has two equations representing its motion on the lower and upper thread. This results in two sets of linear equations. The A and B matrices are listed in Appendix D .

Next the design of a linear controller for the rigid body axes r, θ, and ϕ is considered. In modeling such a controller, the dynamics associated with joint sensors and actuators are neglected. It is assumed that r, θ and ϕ and their time derivatives are available, and that the control torques T_1, T_2, and T_3 can be applied to drive each axis. A simple linear controller is used. It is a multiple input - multiple output integral plus state feedback controller whose control vector, $\underset{\sim}{u}$, can be written as,

$$\underset{\sim}{u} = -K^S \underset{\sim}{y} + K^I \int (\underset{\sim}{R} - C\underset{\sim}{y}) dt \tag{4-6}$$

where K^S is the 3×6 state feedback gain matrix, K^I is the 3×3 integral gain matrix and $\underset{\sim}{R}^T = [\, R_1, R_2, R_3, 0, 0, 0 \,]$ is the desired reference input. The term $C\underset{\sim}{y}$

represents the system output where the C matrix has the following form

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

By defining three additional state variables as,

$$y_7 = \int (y_1 - R_1)dt \qquad y_8 = \int (y_2 - R_2)dt \qquad y_9 = \int (y_3 - R_3)dt \tag{4-7}$$

the control vector, $\underset{\sim}{u}$ becomes

$$\underset{\sim}{u} = -K\underset{\sim}{Z} \tag{4-8}$$

where $\underset{\sim}{Z}^T = [y_1, y_2, y_3, y_4, y_5, y_6, y_7, y_8, y_9]$. The 3×9 gain matrix K now has the following form:

$$K = \begin{bmatrix} K^S & \vdots & K^I \end{bmatrix} \tag{4-9}$$

The gains are chosen based on the linear model described in (4-4) and (4-5). The selected gain matrix is of the following form,

$$K = \begin{bmatrix} k^S_{11} & k^S_{12} & 0 & k^S_{14} & 0 & 0 & k^I_{11} & 0 & 0 \\ k^S_{21} & k^S_{22} & 0 & 0 & k^S_{25} & 0 & 0 & k^I_{22} & 0 \\ 0 & 0 & k^S_{33} & 0 & 0 & k^S_{36} & 0 & 0 & k^I_{33} \end{bmatrix}$$

This controller decouples the linearized system, thus allowing us to carry out a design procedure for each axis independently to achieve a damping ratio of $\xi = 1$ and desired servo-loop frequencies of ω_r, ω_θ, and ω_ϕ for the r, θ, and ϕ axes respectively [49]. The formulations needed to evaluate the gain matrices of the lower and upper thread motions are given in Appendix D. This simple controller is then

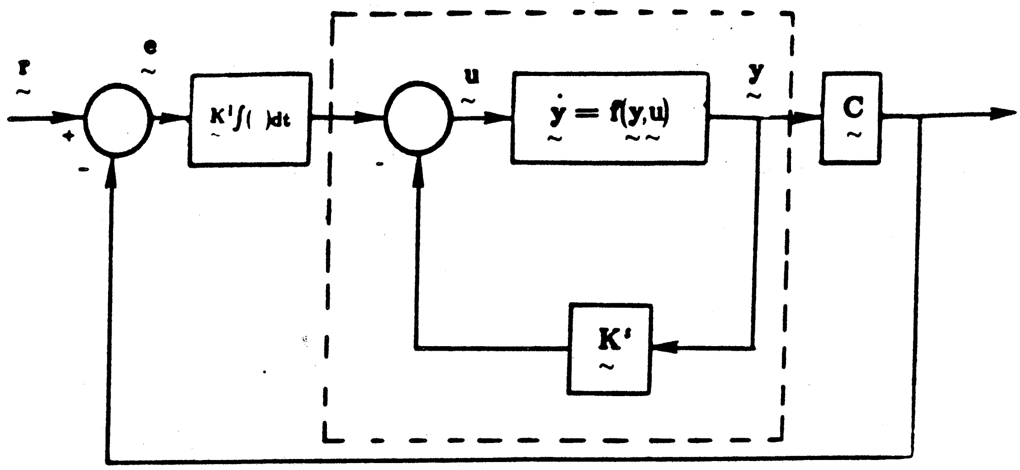


Figure 4-1. Block diagram of the integral plus state feedback controller

applied to the robot arm as modeled by both the rigid body and the combined rigid body and flexible equations of motion. This is illustrated in the block diagram in Fig. 4-1. The computer code written for the digital simulation is listed in Appendix F. The simulation results are presented in the next section.

4.2. Results and Discussion

The dynamic equations of the flexible robot arm along with the equations obtained from the controller lead to a set of seventeen complex, coupled, highly nonlinear stiff equations. The difficulty in handling stiff problems is that most conventional numerical algorithms for solving ordinary differential equations require a time increment equal in measure to the minimum time constant of the system while the problem range which is the difference between initial and final value of time is equal in measure to the maximum system time contant. As a result, the problem cannot be run to completion in a reasonable number of steps. Gear's method [50], used here, is well suited to handle stiff systems since it automatically changes the step size depending upon the region where the solution is relatively active.

The purpose of the simulation studies is to investigate the inter-relationships between the robot structural flexibility and the controller design. These include the effect of constraints due to transmission mechanisms, forced excitation due to inertial forces, and potential instability mechanisms such as resonance. The standard set of physical system parameters used in the computer program are listed in Table 1.

The results obtained for the rigid and flexible motion are shown in Fig. 4-2 to 4-13. The first three plots show the critically damped behavior of the rigid body degrees of freedom r, θ, and ϕ. This illustrates the good performance of the linear controller in the vicinity of the equilibrium point. The plot in Fig. 4-5 represents the motion of the flexible coordinate $q_{11}(t)$. In the transient response, one notes the following:

(1) The excitation of the structural vibration is due to the effect of the rigid body motion. This excitation decreases with time due to the damping introduced by the rigid body controller, and to the diminishing effect of the inertial forces associated with the rigid body motion as it approaches the steady state.

TABLE 1

Standard Set of Physical System Parameters	**VALUE**
Mass of the first beam (m_1)	0.698 Kg
Mass of the second beam (m_2)	0.0429 Kg
Mass of the Payload (m_p)	0.05 Kg
Cross sectional area of the second beam (A_2)	0.00003167 m^2
Length of the first beam (L_1)	0.358 m
Length of the second beam (L_2)	0.5 m
Gravitational acceleration (g)	9.81 m/sec^2
Aluminum density (ρ)	2707 Kg/m^3
Flexural rigidity (EI)	5.67 Pa
Reference position for r	0.4 m
Reference position for θ	0 rad
Reference position for ϕ	0 rad
Desired reference position for r	0.5 m
Desired reference position for θ	0.5 rad
Desired reference position for ϕ	0.5 rad
Servo natural frequency for r (ω_{nr})	4 rad/sec
Servo natural frequency for θ ($\omega_{n\theta}$)	4 rad/sec
Servo natural frequency for ϕ ($\omega_{n\phi}$)	8 rad/sec

(2) The natural frequency of the flexible mode decreases with time due to the increase of the span of the flexible beam and to the vanishing effect of the rigid body inertial forces applied to the robot arm as modeled by both the rigid body and the combined rigid body and flexible equations of motion.

(3) The steady state response is a small amplitude sustained oscillation around a negative value. The latter represents the static deflection due to gravity while the sustained oscillation shows the decoupled response of the elastic motion from the rigid body motion at steady state due to the effect of the self locking condition of the leadscrew.

The motion of $q_{12}(t)$ is illustrated in Fig. 4-6. It has one additional feature. The magnitude of the vibratory motion is on the order of $10^{-3}m$ while the one for $q_{11}(t)$ is on the order of $10^{-1}m$. This is due to the large amount of energy required to excite the higher modes. Coupling with the first mode is also evident.

Figures 4-7 and 4-8, which illustrate the flexible motion represented by $q_{21}(t)$ and $q_{22}(t)$, indicate a very important aspect of this particular robot design. The transient responses show the large excitation induced by the rigid body motion, and then die out with time. These interesting results can be interpreted as follows:

(1) Gravity doesn't affect the flexible motion in the $q_{21}(t)$ and $q_{22}(t)$ directions. Therefore, their motions are expected to either die out to zero with time or oscillate around zero (i.e. no static deflection).

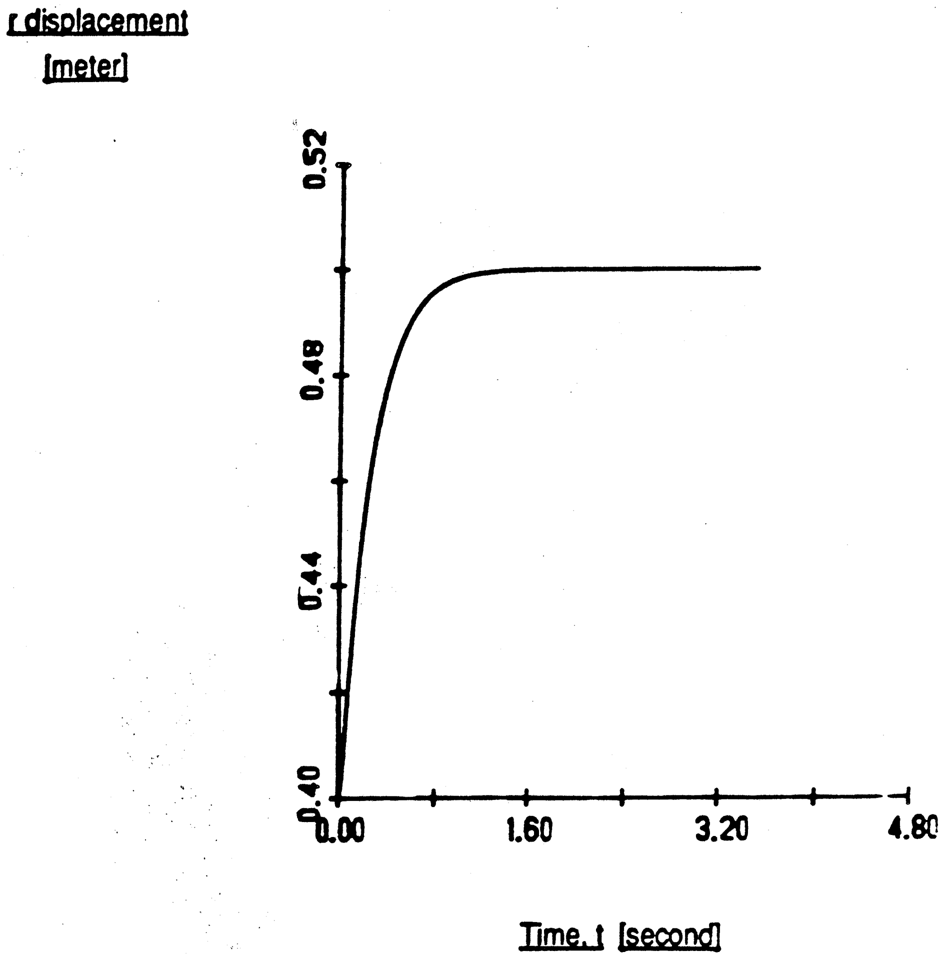


Figure 4-2. r response obtained from the rigid body controller in the base run.

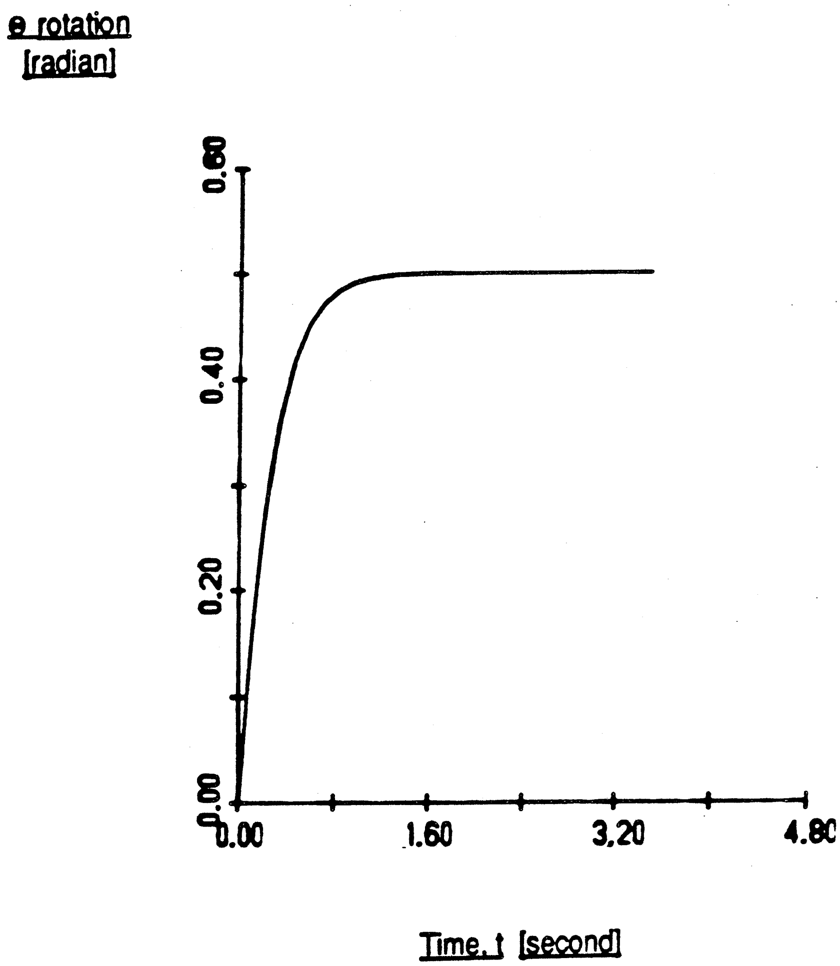


Figure 4-3. θ response obtained from the rigid body controller in the base run.

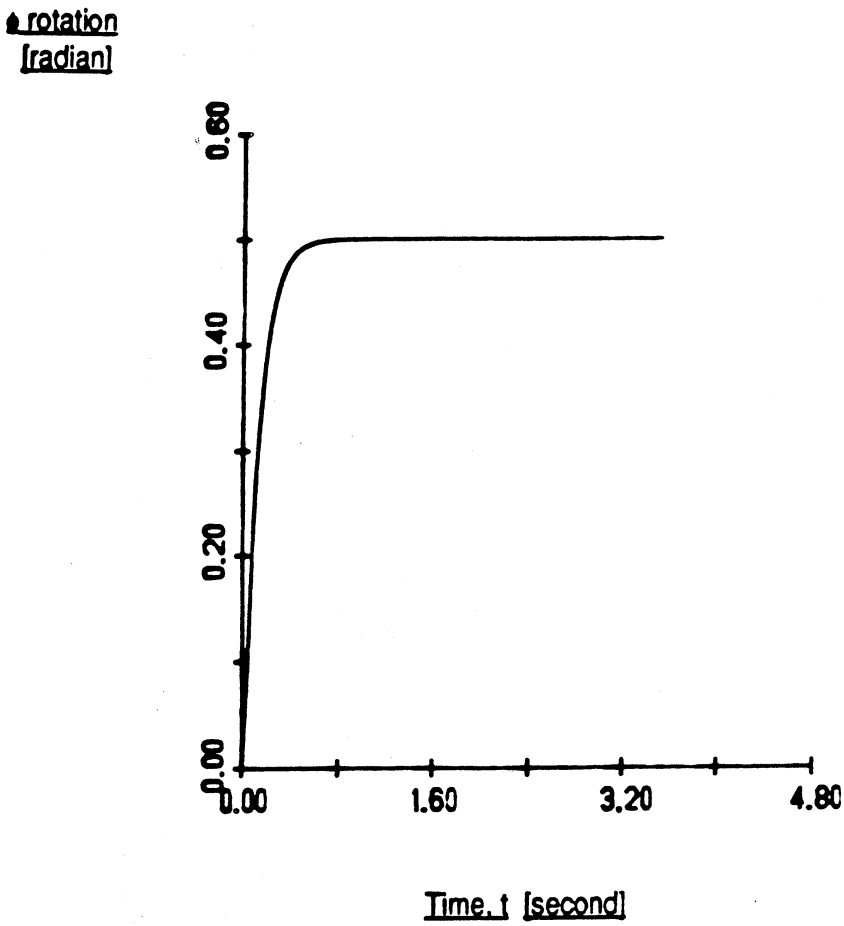


Figure 4-4. ϕ response obtained from the rigid body controller in the base run.

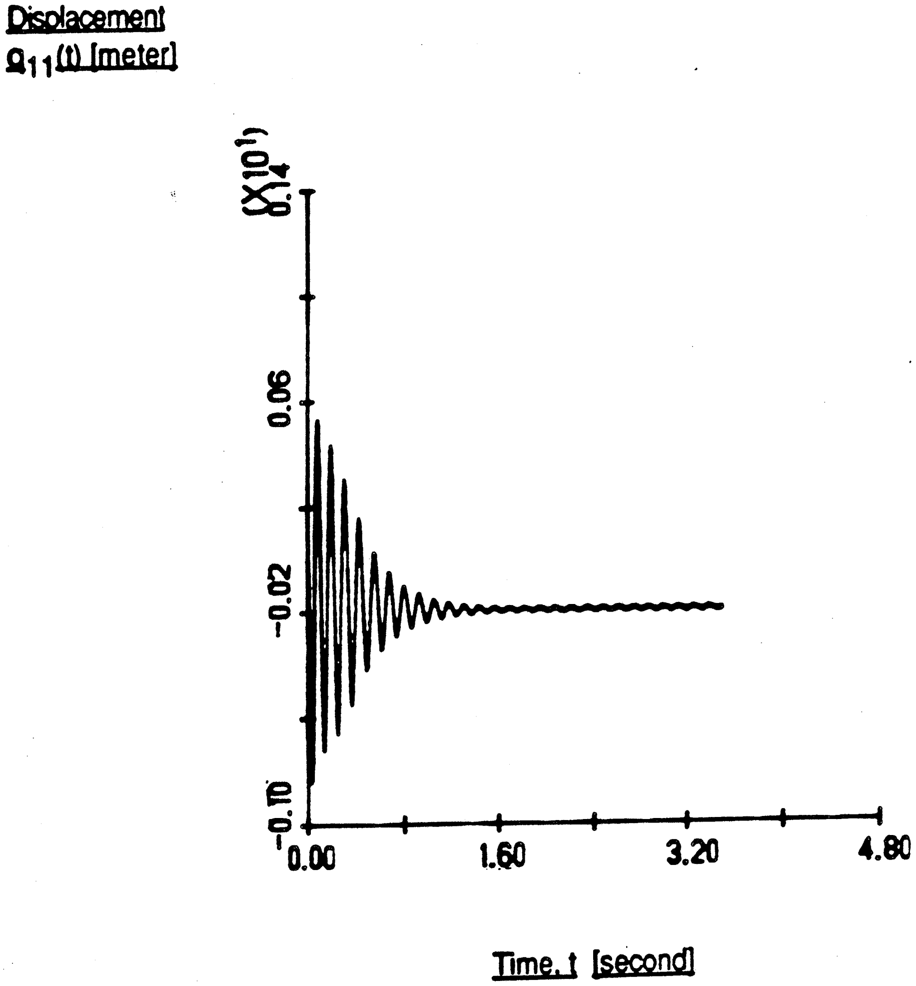


Figure 4-5. Flexible motion coordinate $q_{11}(t)$ in response to the rigid body controller in the base run.

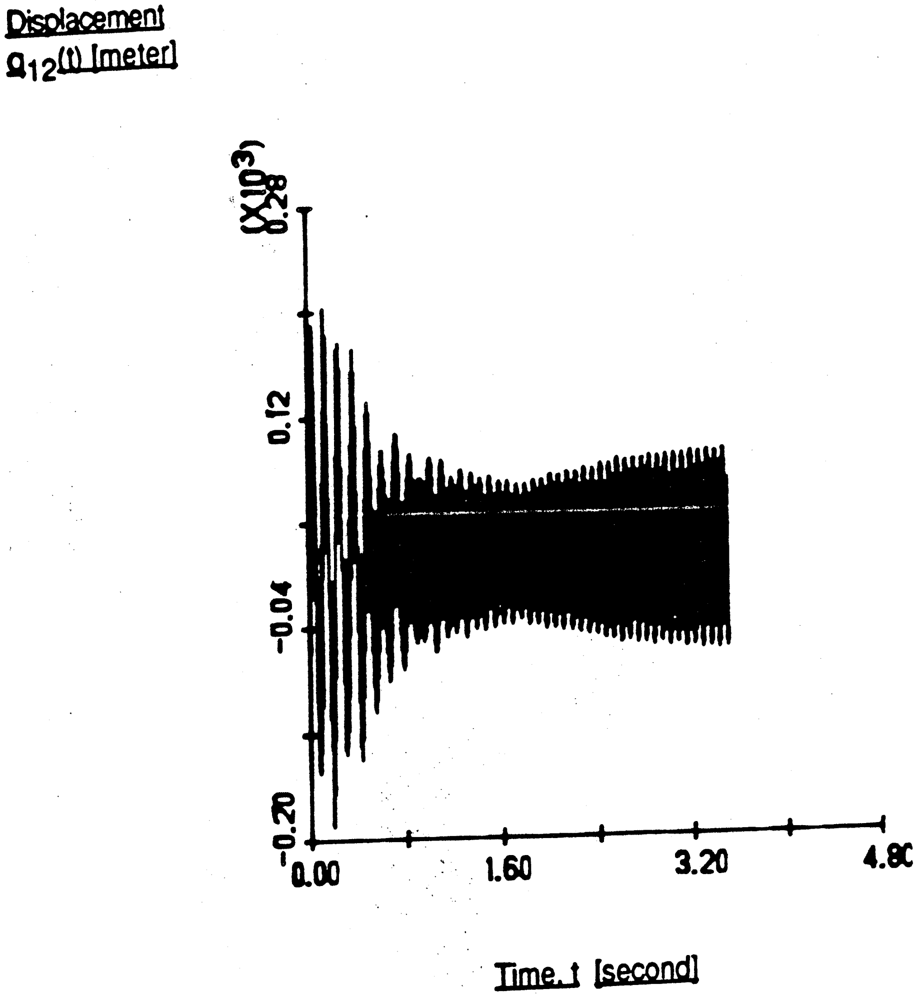


Figure 4-6. Flexible motion coordinate $q_{12}(t)$ in response to the rigid body controller in the base run.

(2) Since there is no constraint on the rigid body motion in the ϕ direction, the flexible motion in the $q_{21}(t)$ and $q_{22}(t)$ directions act as a disturbance source on the rigid body motion in the ϕ direction. These disturbances are compensated for by the rigid body controller.

Figures 4-9 to 4-11 represent the control torques T_1, T_2, and T_3. The oscillations in the control signals for the base and second joints show the effect of the flexible motion on the rigid body motion. The control torques T_1 and T_2 for the r and θ axes respectively are set to zero once the steady state is reached. This is due to the nonbackdrivable characteristic of the leadscrew transmission mechanism which satisfies the self locking conditions. The total deflection of the end effector in the vertical and the horizontal directions (i.e. V and W in equation (2-4)) are shown in Figs. 4-12 and 4-13 and found to be dominated by the first mode and nearly identical in shape to Figs. 4-5 and 4-7.

Additional runs are made with modifications to the standard set of parameters to study the behavior of the system in the following areas:

(1) *Instability mechanisms:* The flexural rigidity, EI, is reduced to $0.44Pa$. This can be achieved by constructing the flexible beam of polymers. As a result, the natural frequencies of the flexible beam are lowered to values that are close to the servo loop frequencies. The main results are shown in Fig. 4-14 to 4-18. The important features of this run are:

(a) In Figs. 4-14 to 4-16, the ϕ response has small overshoot. The control torques for the base and second joints are very oscillatory. This illus-

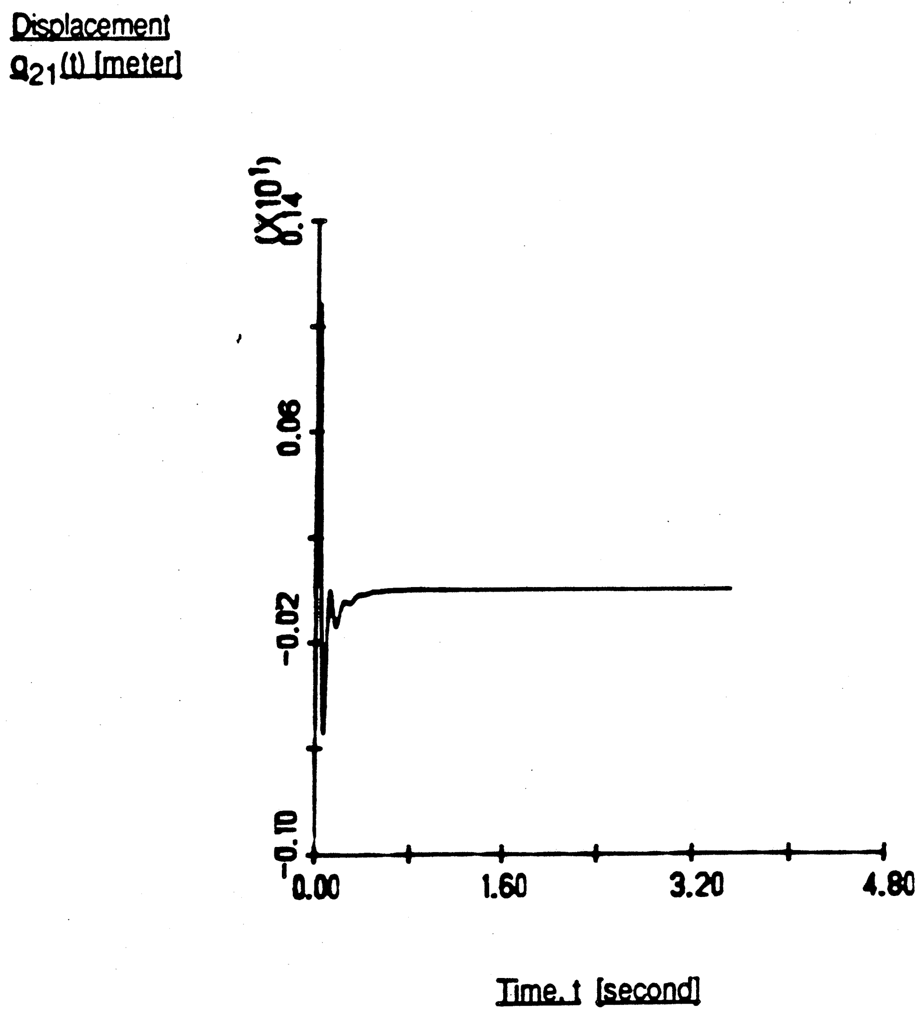


Figure 4-7. Flexible motion coordinate $q_{21}(t)$ in response to the rigid body controller in the base run.

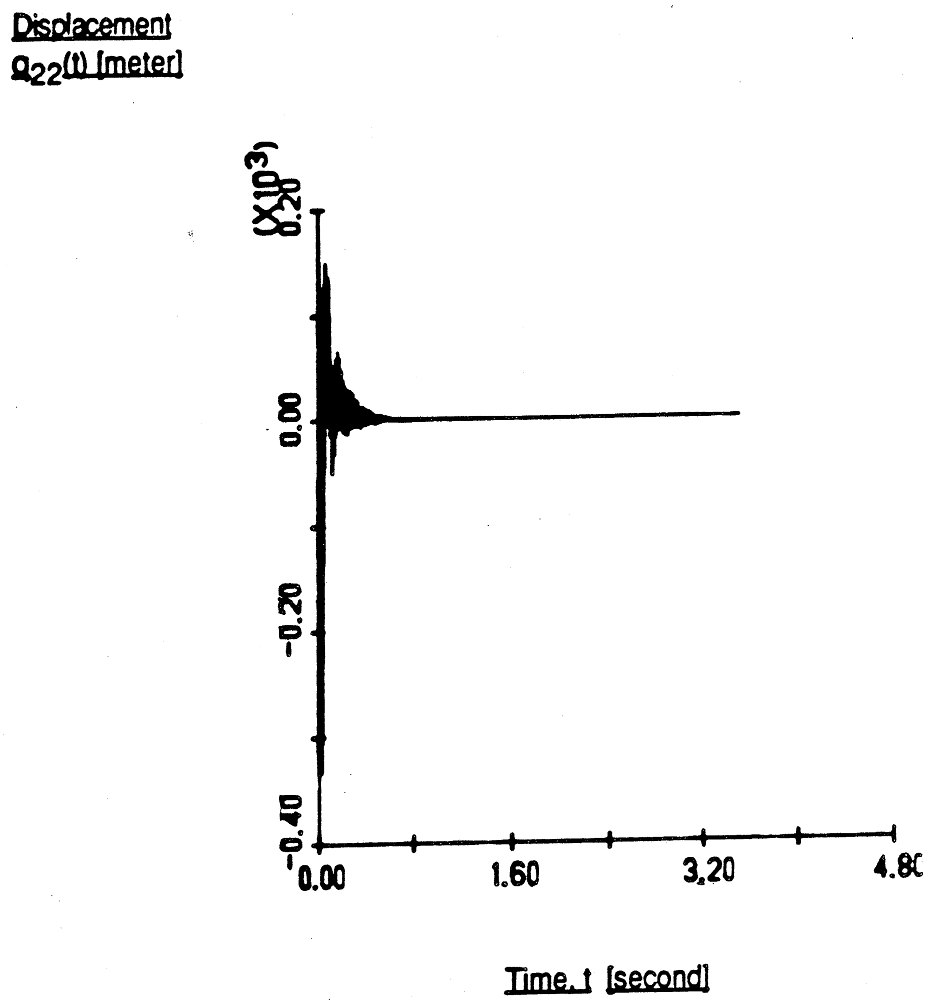


Figure 4-8. Flexible motion coordinate $q_{22}(t)$ in response to the rigid body controller in the base run.

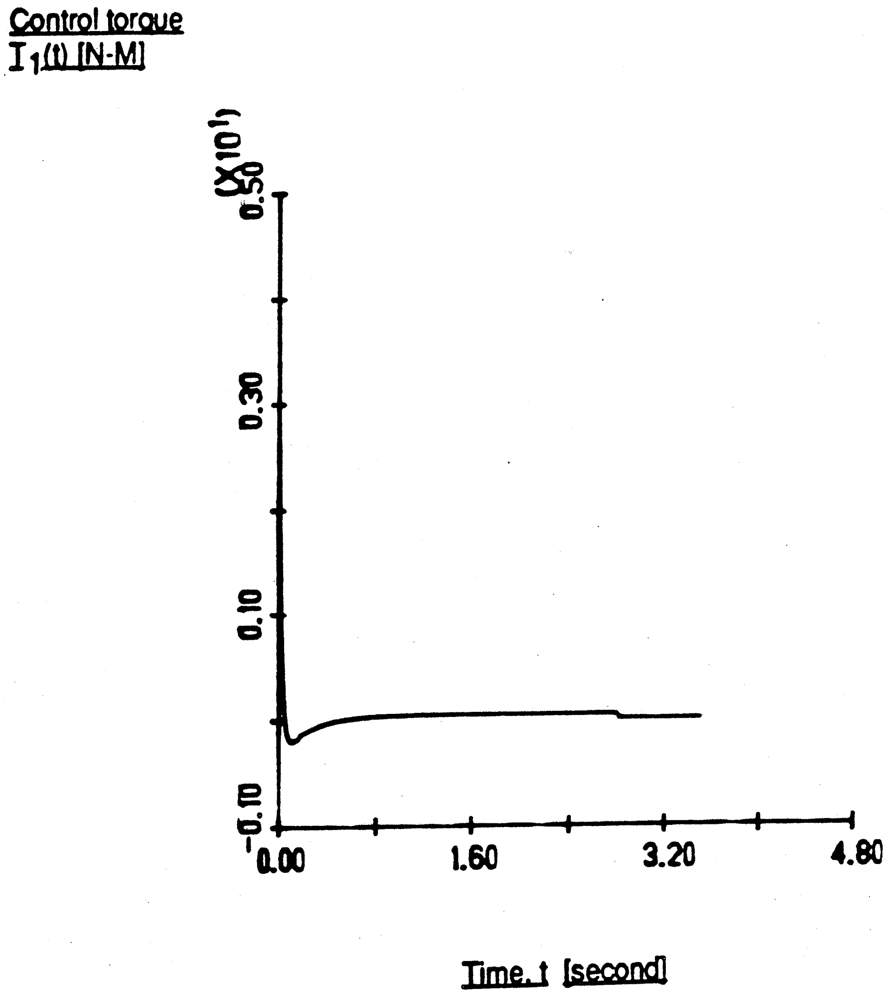


Figure 4-9. Control signal for the prismatic joint obtained from the rigid body controller in the base run.

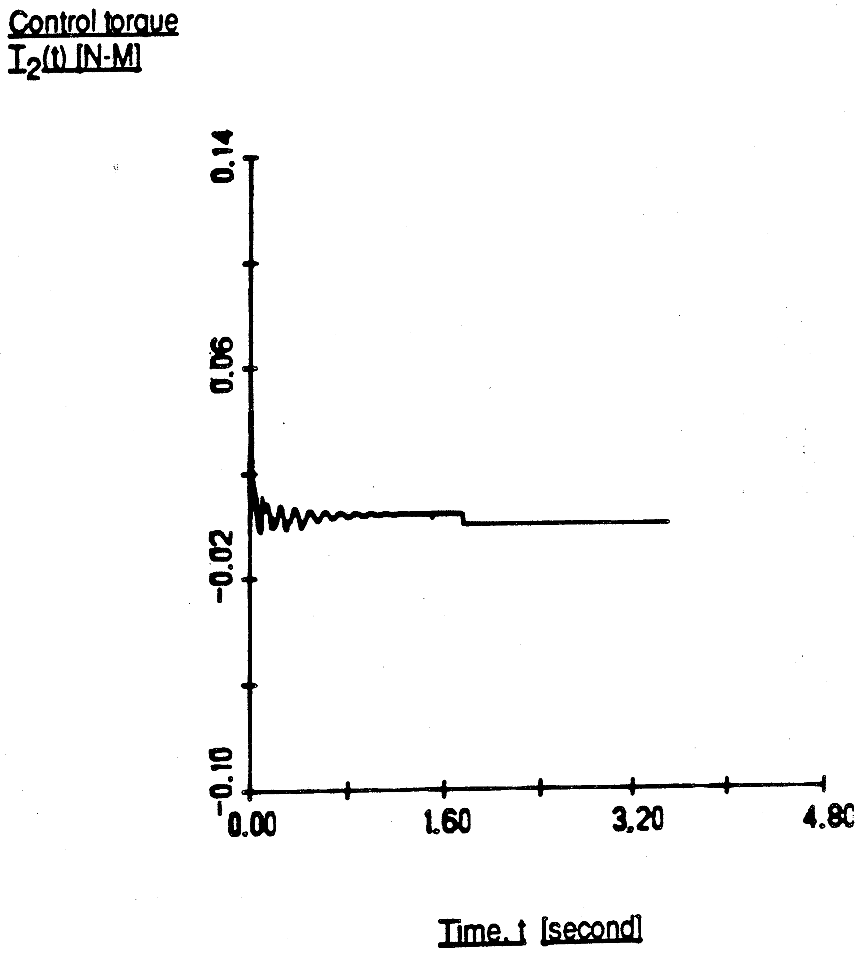


Figure 4-10. Control signal for the second joint obtained from the rigid body controller in the base run.

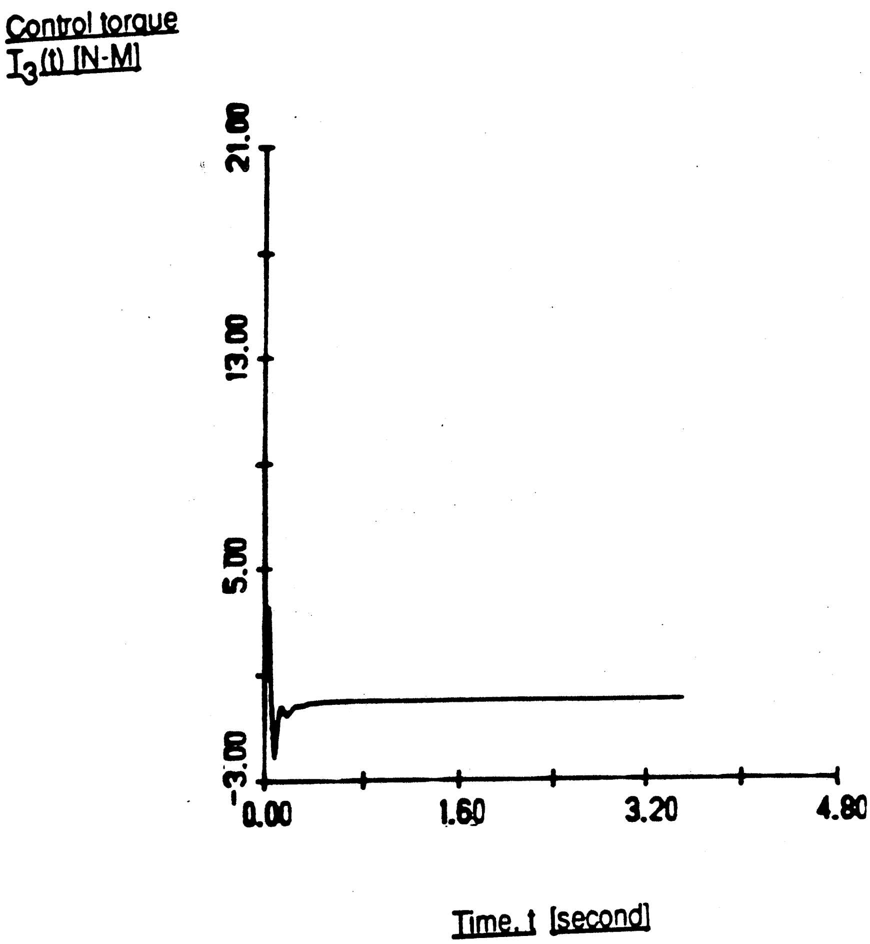


Figure 4-11. Control signal for the base joint obtained from the rigid body controller in the base run.

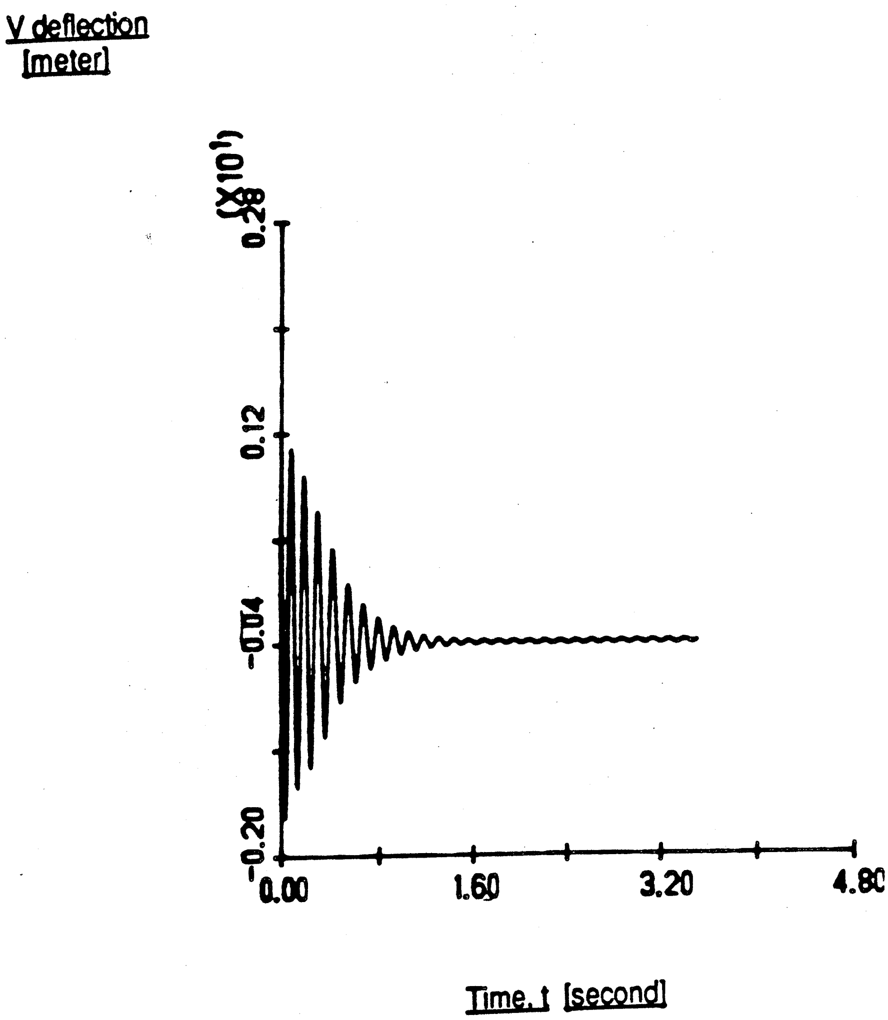


Figure 4-12. Total vertical deflection, v, obtained from the rigid body controller in the base run.

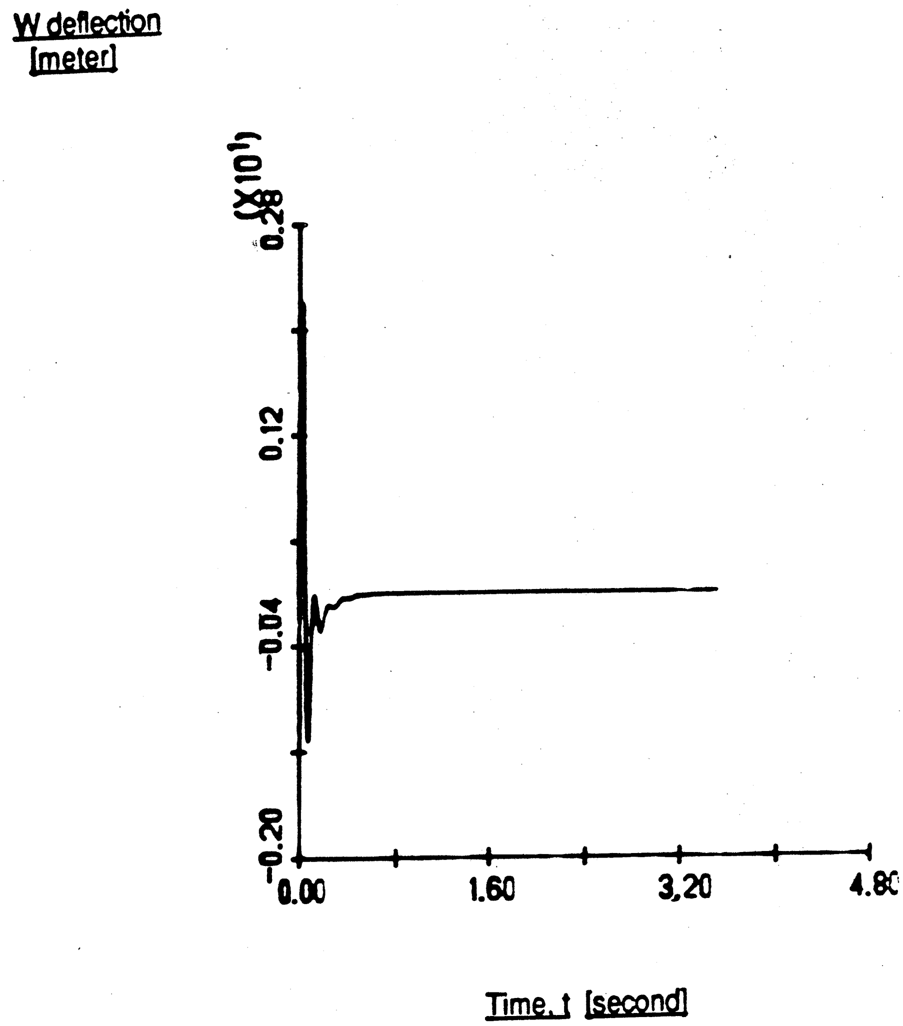


Figure 4-13. Total horizontal deflection, w, obtained from the rigid body controller in the base run.

trates the effect of the flexible motion on the rigid body motion.

(b) The flexible motion frequency is reduced and the magnitude of the oscillation in both the vertical and the horizontal directions increased. (see Fig. 4-17 and 4-18).

(c) Due to the small value of EI and to the proximity of the servo-loop frequency ω_ϕ to the fundamental beam natural frequency, the amplitude of these oscillations grow with time. This is clearly shown in Fig. 4-18. Such proximity is not likely to occur in practice, but would lead to very poor performance when it does.

Note that the integration is interrupted at 1.24 seconds since this case could not be run to completion with a reasonable computation time due to the divergence of the solution.

(2) *Speed accuracy trade off:* The servo loop natural frequencies for r, θ, and ϕ, are raised from $\omega_r = 4, \omega_\theta = 4$ and $\omega_\phi = 8$ rad/sec in the base run, to $\omega_r = 8, \omega_\theta = 8$, and $\omega_\phi = 16$ rad/sec. Comparing the numerical results obtained with those of the base run, an increase on the order of 2 is observed in the amplitudes of the flexible motion coordinates. This is illustrated in Fig. 4-19 and 4-20 where the response of the first elastic mode in the horizontal and vertical directions are given. Unlike the base run, the responses of θ and ϕ (see Fig. 4-21 and 4-22) show the effects of the flexible motion on the rigid body motion. This is also reflected in Fig. 4-23 and 4-24 where the control torques for the base and second joints become more oscillatory then

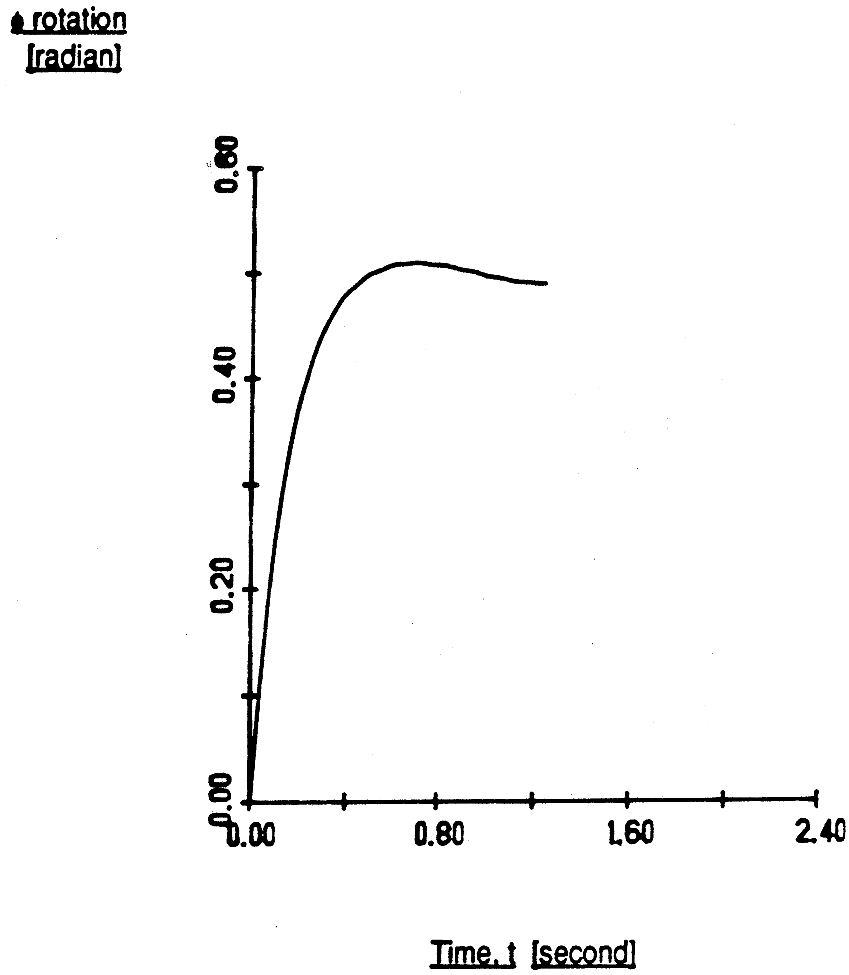


Figure 4-14. ϕ response obtained from the rigid body controller in the instability run.

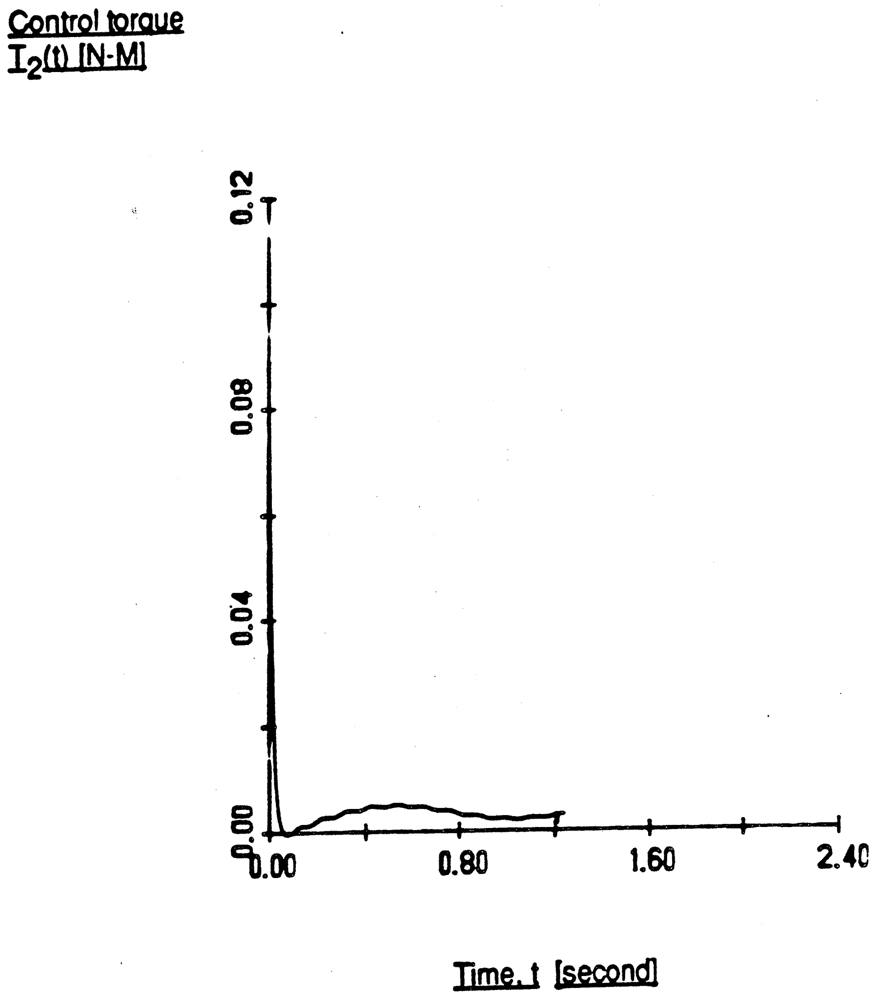


Figure 4-15. Control signal for the second joint obtained from the rigid body controller in the instability run.

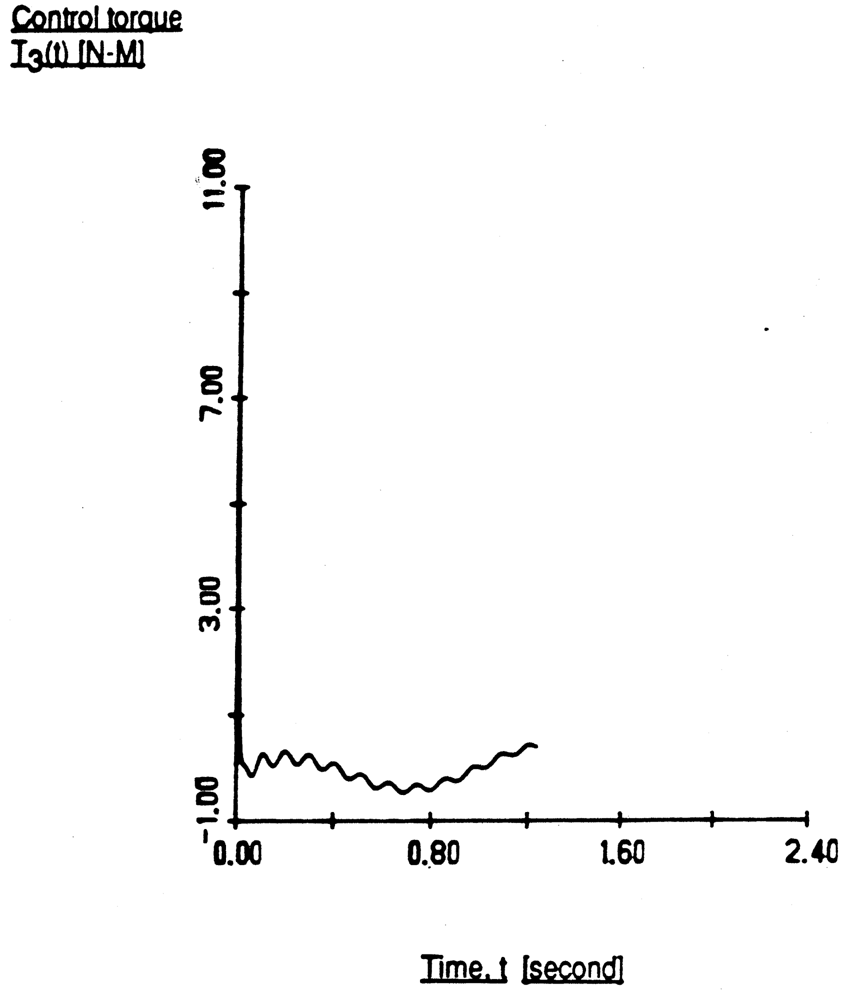


Figure 4-16. Control signal for the base joint obtained from the rigid body controller in the instability run.

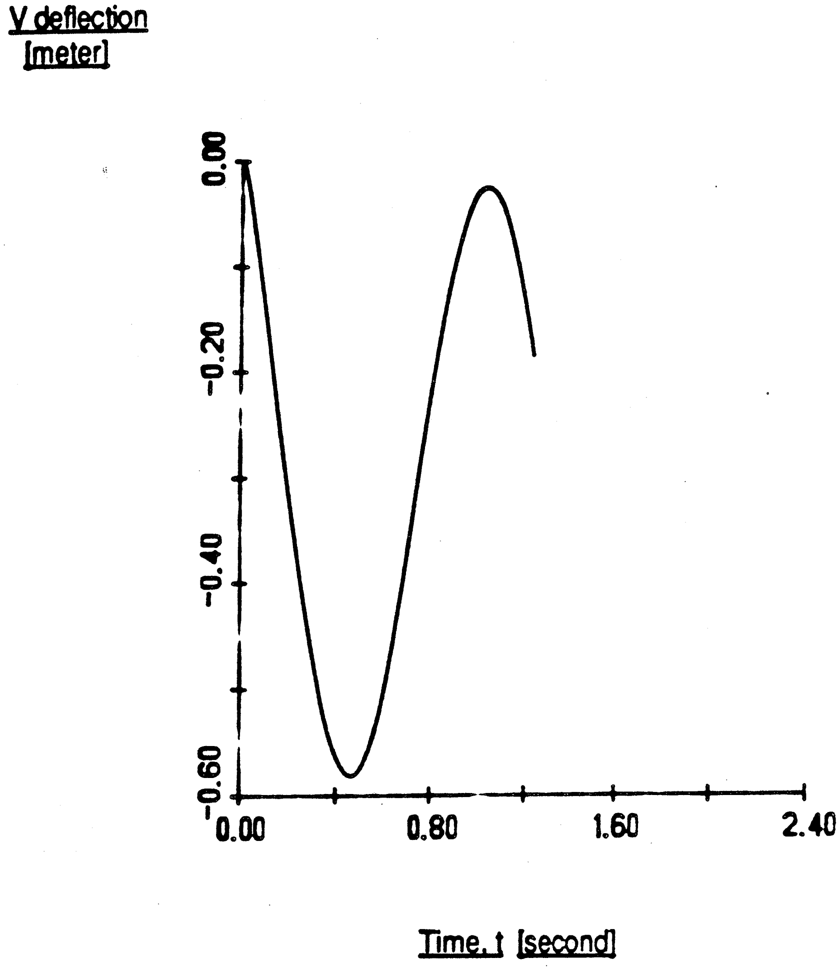


Figure 4-17. Total vertical deflection, v, obtained from the rigid body controller in the instability run.

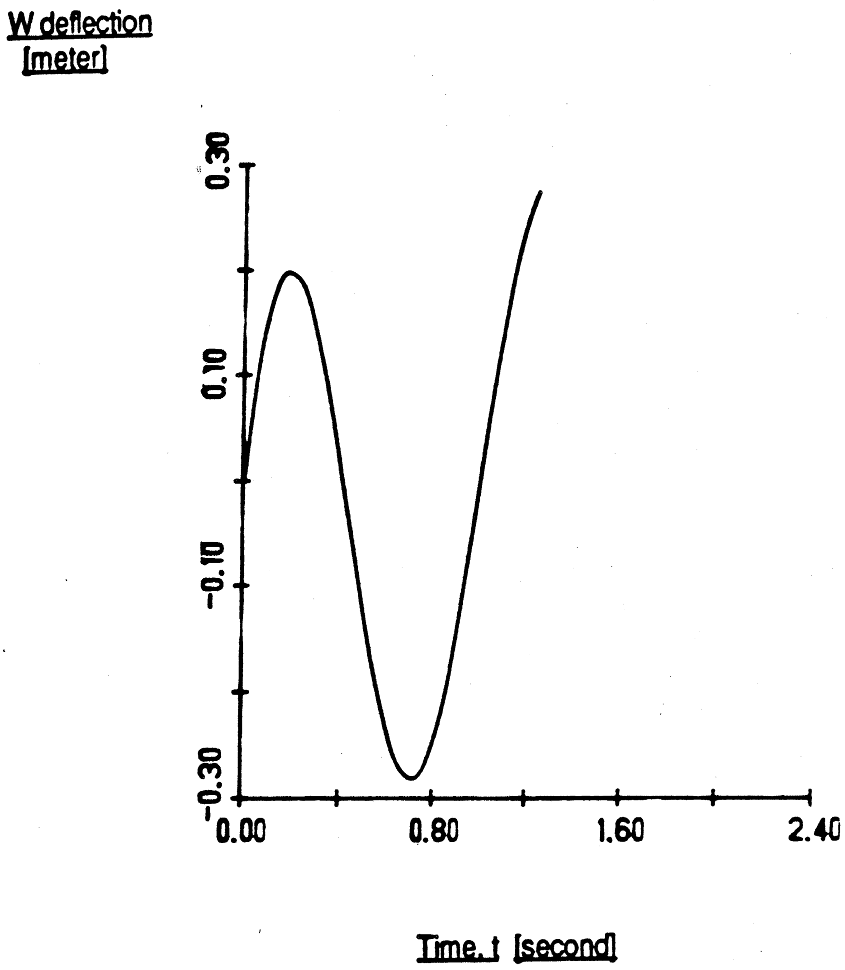


Figure 4-18. Total horizontal deflection, w, obtained from the rigid body controller in the instability run.

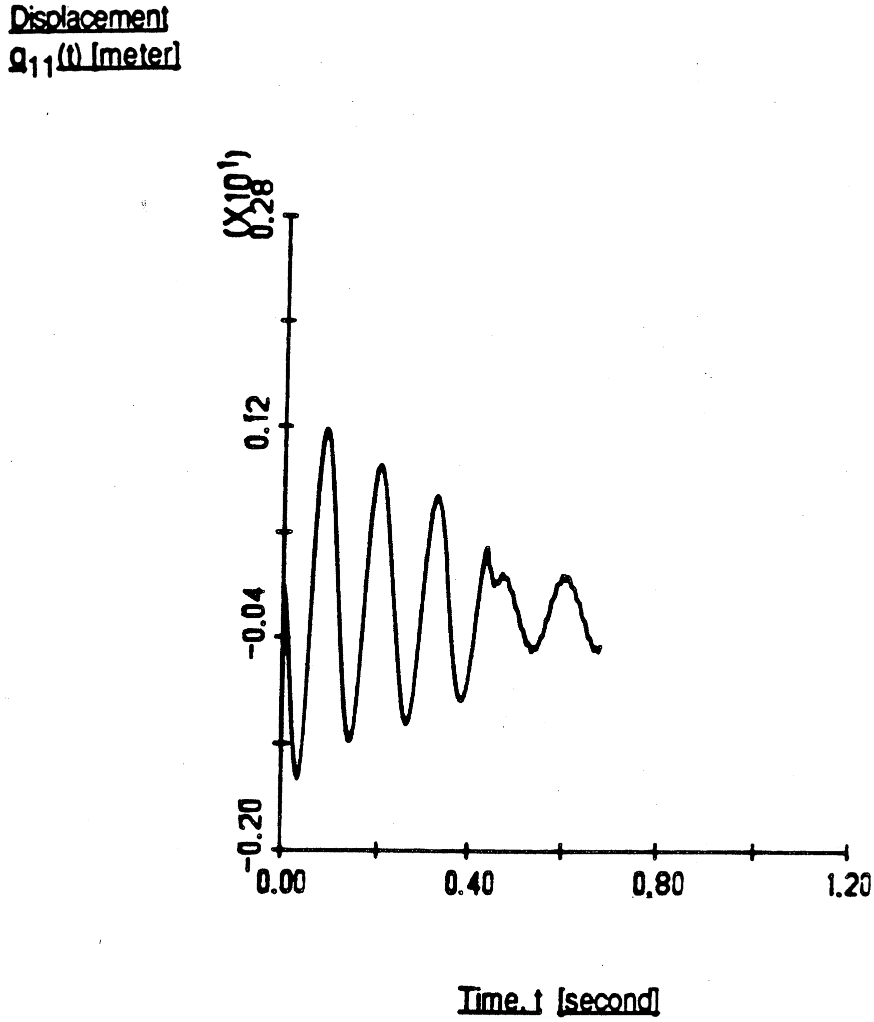


Figure 4-19. Flexible motion coordinate $q_{11}(t)$ in response to the rigid body controller in the speed-accuracy trade-off.

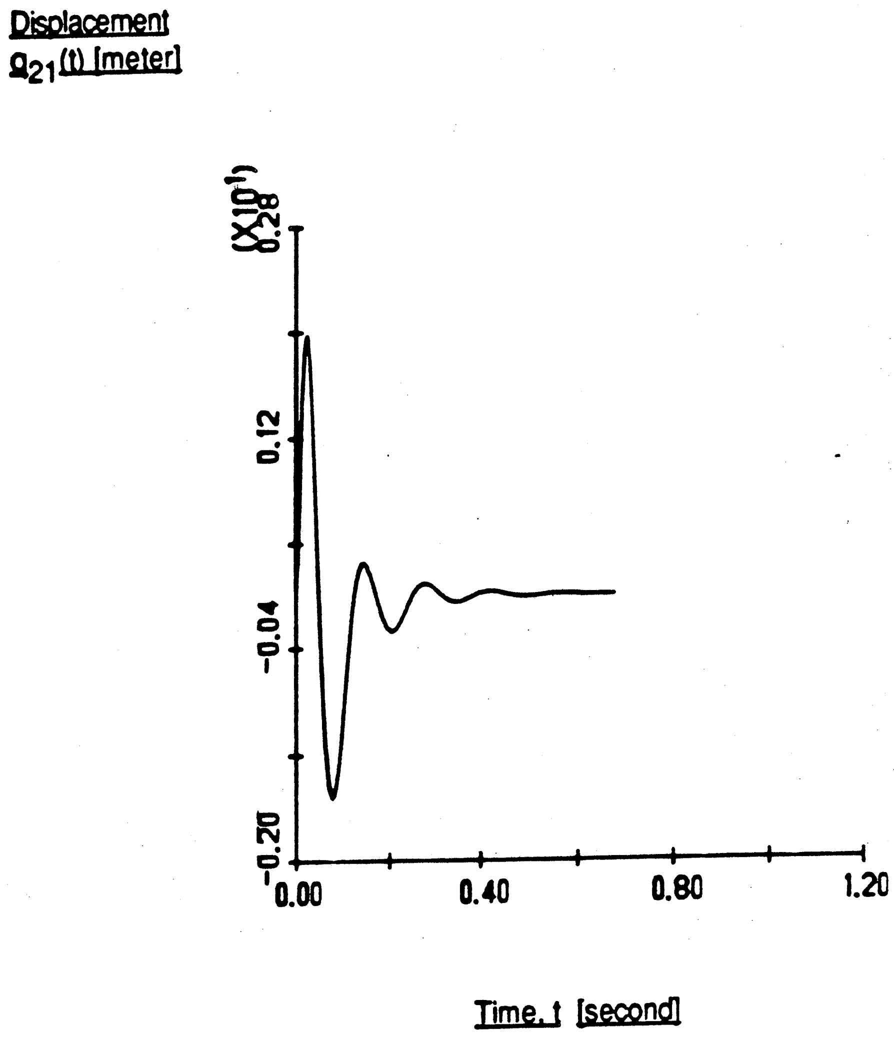


Figure 4-20. Flexible motion coordinate $q_{21}(t)$ in response to the rigid body controller in the speed-accuracy trade-off.

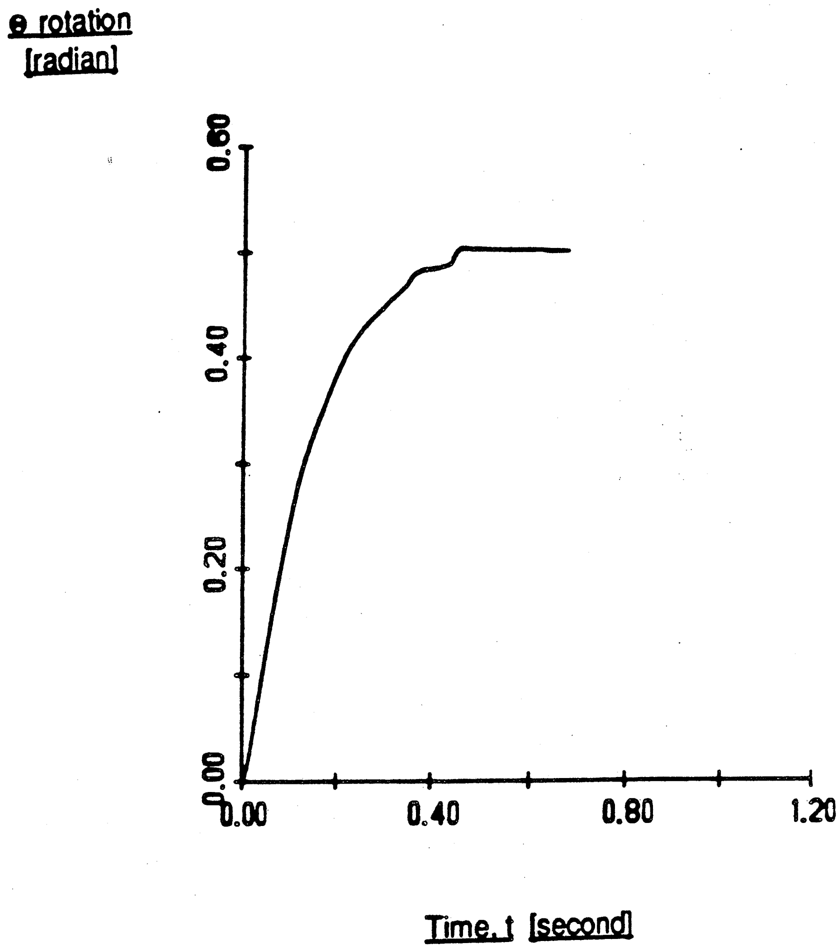


Figure 4-21. θ response obtained from the rigid body controller in the speed-accuracy trade-off.

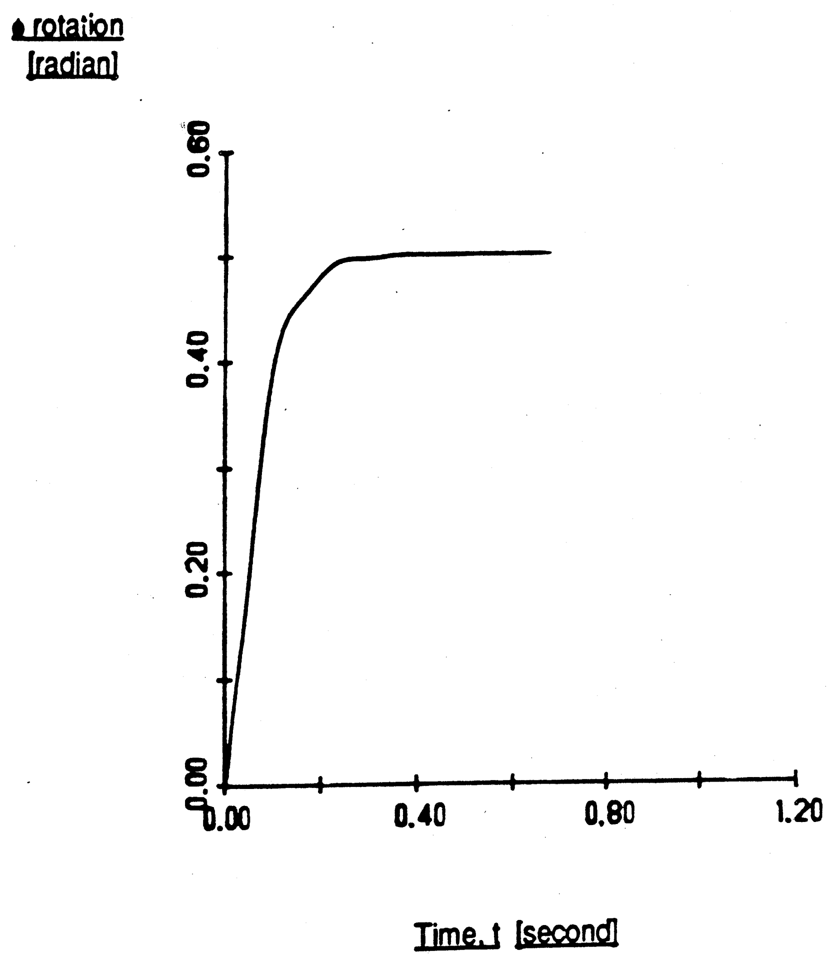


Figure 4-22. ϕ response obtained from the rigid body controller in the speed-accuracy trade-off.

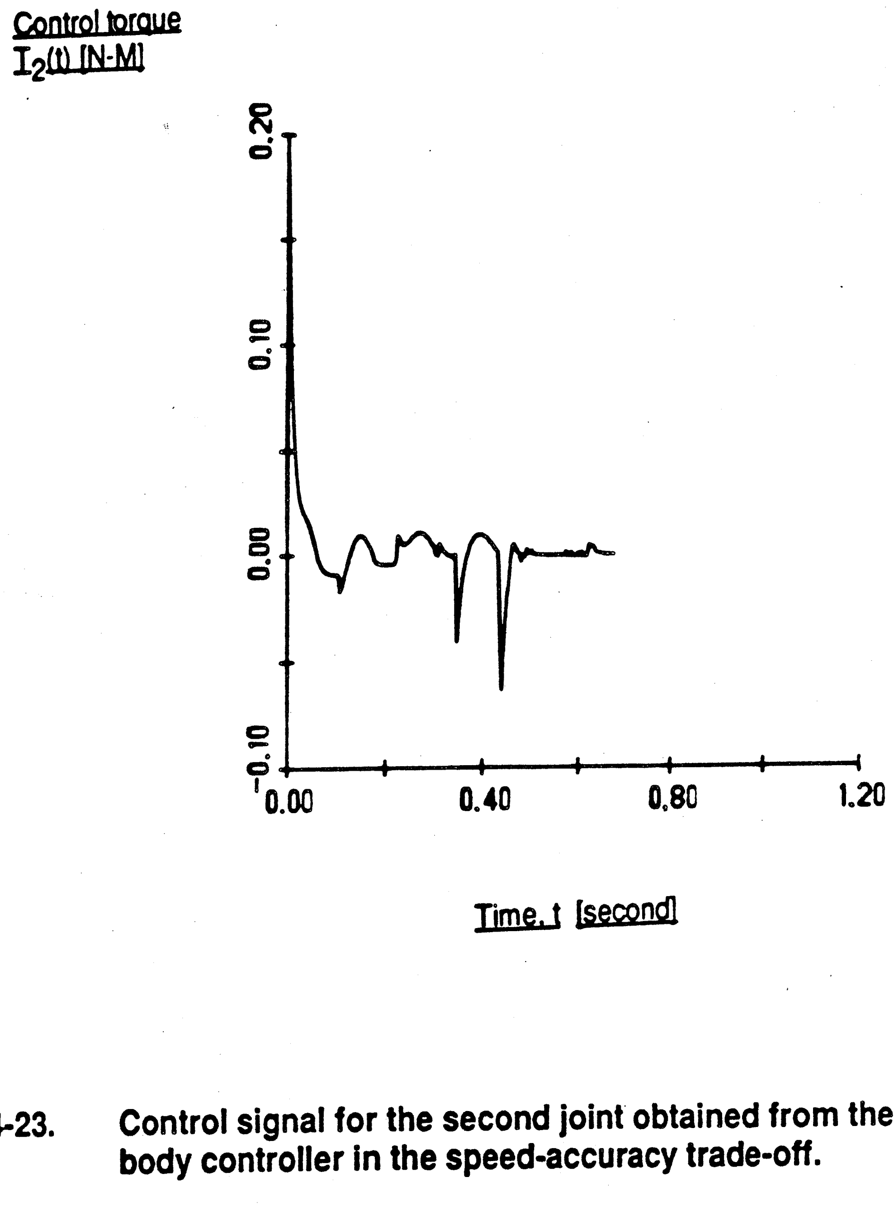


Figure 4-23. Control signal for the second joint obtained from the rigid body controller in the speed-accuracy trade-off.

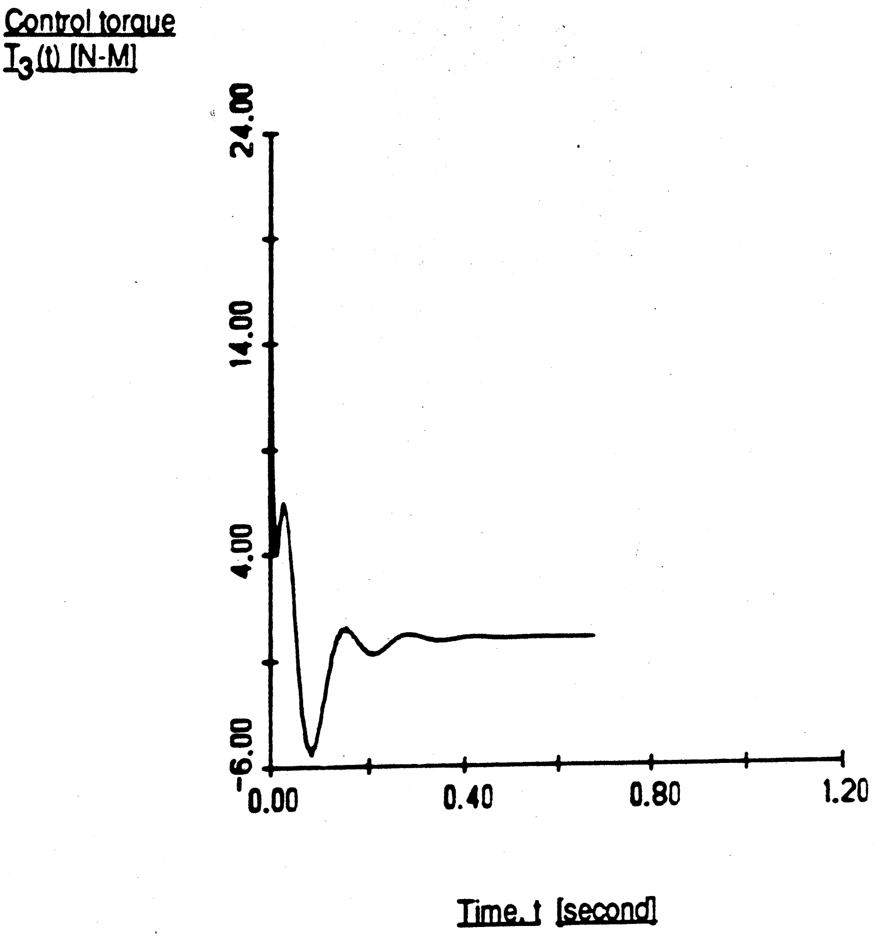


Figure 4-24. Control signal for the base joint obtained from the rigid body controller in the speed-accuracy trade-off.

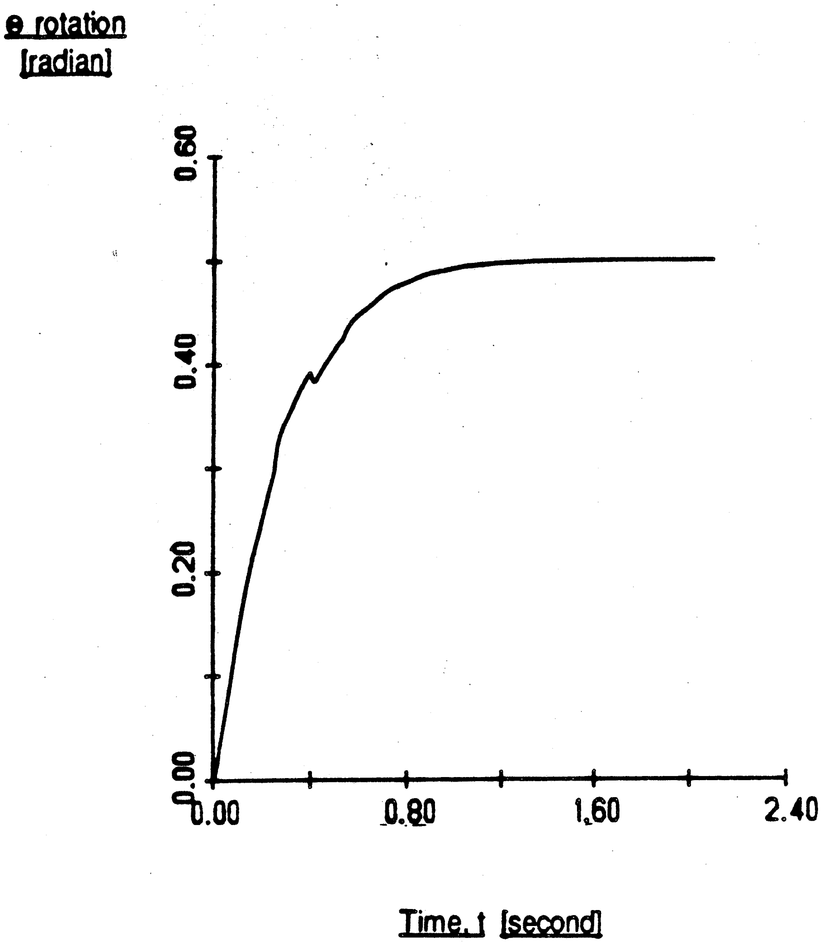


Figure 4-25. θ response obtained from the rigid body controller with the payload increased to 78.5 g.

their counterparts in the base run.

(3) *Effect of leadscrew on the controller design:* The payload mass is increased from 50g in the base run to 78.5g and the gains of the rigid body controller are retuned to compensate for this new payload mass. The latter increases the flexible motion inertial terms, causing large fluctuations in the magnitude of the normal force, N, and even in some cases changing its sign. Recall from the leadscrew constraint derivation that changes in the sign of N result in switching of the motion from the lower to the upper thread or vice versa. This will, depending on the type of motion, either increase or decrease sharply the required torque to rotate the leadscrew which may be under tension or compression. Thus causing a deterioration in the controller performance in the vicinity of the transition period. This is clearly shown in Fig. 4-25 where the notch in the θ response is caused during the transition of the motion from one surface to another.

4.3. Summary and Conclusions

An integral plus state variable feedback controller, based on the linearized version of the rigid body model of the robot arm, is derived. This simple controller design is considered to be representative of the robot controllers currently in use. The dynamic model and the controller design are used as the basis for the simulation studies. The latter demonstrates the potential mechanisms by which the robot structural dynamics and controller design can interact.

The effects of constraints, due to the leadscrew transmission mechanisms, give the robot arm a cantilever beam-like behavior in the r and θ directions which leads to some sustained vibratory motion of $q_{11}(t)$ and $q_{12}(t)$ at steady state. The absence of the constraint in the ϕ direction allows the strain energy to be dissipated by the rigid body controller. The simulation results demonstrate the following additional features: (i) There is a potential for instability if the servo-loop frequencies approach the natural frequency of the robot arm, and (ii) High speed operation deteriorates the accuracy of the system, and results in a trade-off between speed and accuracy.

These results are very useful for designing and evaluating potential approaches to the flexible robot arm control problem. The latter will be discussed and treated in the next chapter.

CHAPTER 5

DESIGN AND IMPLEMENTATION OF THE RIGID AND FLEXIBLE MOTION CONTROLLER

The interrelationships between the structural flexibility and the controller design, obtained in Chapter 4, provide guidance in developing the multiple-input multiple-output controller for the rigid and flexible motion of the robot arm. Towards achieving this goal, a simple case, involving the controller design of a three lumped mass system, is treated. The latter simulates the control problem of a compliant beam whose flexible motion is approximated by two elastic modes. The rationale is to start with the control of a relatively simple system as a preliminary step towards the general control problem of the flexible robot arm.

5.1. Description of the Three Lumped Mass System

In an attempt to gain insight into the control problem of the flexible robot arm, a simple system consisting of three lumped masses connected by massless springs is considered (see Fig. 5-1). This system can be viewed as a lumped model of an articulated compliant beam. The rationale is to scale down the complexity of the dynamic model of the robot arm, to get better insight into the controller

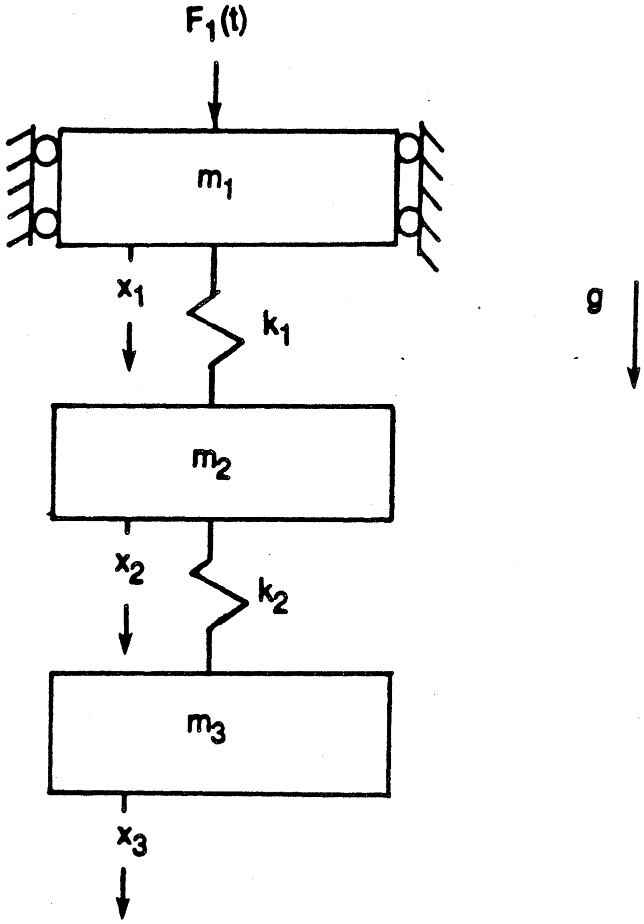


Figure 5-1. A schematic of the three lumped mass model.

performance, while working with a simple linear model, and to reduce the cost of computer runs.

The first mass reflects the rotational rigid body degree of freedom of the joint whereas the second and third masses represent the first and second elastic modes of transverse vibration. No structural damping is considered in the discrete model and the springs represent the structural stiffness of the beam. The three lumped masses are supported in the vertical direction to include the effect of gravity. The control force is applied to the first mass only.

Denoting the displacements of the first, second and third masses by x_1, x_2 and x_3 respectively. The equations of motion can be written as

$$m_1\ddot{x}_1 = k_1(x_2 - x_1) + m_1 g + F_1 \tag{5-1a}$$

$$m_2\ddot{x}_2 = -k_1(x_2 - x_1) + k_2(x_3 - x_2) + m_2 g \tag{5-1b}$$

$$m_3\ddot{x}_3 = -k_2(x_3 - x_2) + m_3 g \tag{5-1c}$$

where k_1 and k_2 are the structural stiffness terms and g is the gravitational acceleration.

5.2. Controller Design for the Lumped Model

Two types of controllers are used for the three lumped mass model. The first is based solely on the motion of the first mass. This gives a good indication of the effect of the rigid body controller on the flexible system which is represented by the second and third masses. The second controller is designed based on the motions of the first and second masses, thus, including the flexible motion into the control action. The third mass, representing the second elastic mode, is intentionally not considered in the controller design to study the effect of control spillover.

In both controller designs, an integral plus state feedback controller is implemented. The gains of the rigid body controller are computed based on the equation of motion for the first mass,

$$m_1\ddot{x}_1 = F_1 + m_1 g \tag{5-2}$$

This equation is identical to (5-1a) with the flexibility term $k_1(x_2 - x_1)$ set to zero. The control force, F_1, has the following form,

$$F_1 = -\begin{bmatrix} k_1^s & k_2^s \end{bmatrix}\begin{Bmatrix} x_1 \\ \dot{x}_1 \end{Bmatrix} - k_1^I \int (x_1 - R_1)dt \tag{5-3}$$

By substituting (5-3) into (5-2), the gains are computed to achieve a damping ratio of $\xi = 0.53$ and a desired natural frequency for the first mass. The controller is then applied to the three lumped mass model.

The rigid and flexible motion controller is designed based on the linear equations of motion of the first and second masses,

$$m_1\ddot{x}_1 = F_1 + m_1 g + k_1(x_2 - x_1) \tag{5-4a}$$

$$m_2\ddot{x}_2 = m_2 g - k_1(x_2 - x_1) \tag{5-4b}$$

These equations are similar to (5-1a) and (5-1b) with the coupling term $k_2(x_3 - x_2)$ set to zero. Define the state variables to be,

$$y_1 = x_1 \quad y_2 = x_2 \quad y_3 = \dot{x}_1 \quad y_4 = \dot{x}_2 \quad y_5 = \int (x_1 - R_1)dt$$

the equations of motion become,

$$\begin{Bmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \\ \dot{y}_4 \\ \dot{y}_5 \end{Bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ -\dfrac{k_1}{m_1} & \dfrac{k_1}{m_1} & 0 & 0 & 0 \\ \dfrac{k_1}{m_2} & -\dfrac{k_1}{m_2} & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{Bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dfrac{1}{m_1} \\ 0 \\ 0 \end{bmatrix} F_1 + \begin{Bmatrix} 0 \\ 0 \\ g \\ g \\ -R_1 \end{Bmatrix} \tag{5-5}$$

in matrix notation, this can be written as,

$$\dot{\underset{\sim}{y}} = A\underset{\sim}{y} + BF_1 + \underset{\sim}{W} \tag{5-6}$$

where the vector, $\underset{\sim}{W}$, shows the effect of gravity, Note that with this notation, the integral action becomes embedded in the state equations. Thus, the integral plus state variable feedback controller can simply be written as,

$$F_1 = -K\underset{\sim}{y} \tag{5-7}$$

where the gain matrix $K^T = [k_1^s\ k_2^s\ k_3^s\ k_4^s\ k_1^I]$. Substitute (5-7) into (5-6) to get,

$$\dot{\underset{\sim}{y}} = A\underset{\sim}{y} + BF_1 + \underset{\sim}{W} = (A - BK)\underset{\sim}{y} + \underset{\sim}{W} \tag{5-8}$$

The eigenvalues of $(A - BK)$ matrix can be arbitrarily assigned by the state feedback $F_1 = -K\underset{\sim}{y}$, as long as the system given by (5-6) is controllable. First, the state equations are converted to the controllable canonical form. This is done by transforming the state vector $\underset{\sim}{y}$ to a new state vector $\underset{\sim}{z}$ using a constant transformation matrix T so that

$$\underset{\sim}{y} = T\underset{\sim}{Z} \tag{5-9}$$

The differentiation of this equation yields,

$$\underset{\sim}{\dot{y}} = T\underset{\sim}{\dot{Z}} \tag{5-10}$$

Substituting these values into (5-6) to get

$$\begin{aligned} \underset{\sim}{\dot{Z}} &= T^{-1}AT\underset{\sim}{Z} + T^{-1}BF_1 + T^{-1}\underset{\sim}{W} \\ &= A'\underset{\sim}{Z} + B'F_1 + T^{-1}\underset{\sim}{W} \end{aligned} \tag{5-11}$$

where $A' = T^{-1}AT$ and $B' = T^{-1}B$. Let t_i be the i^{th} column of T, i.e.

$$T = \left[t_1, t_2, \ldots, t_n\right] \tag{5-12a}$$

Use the formulation given in [51], [52], [53], [54] to compute the t_i's

$$\begin{aligned} t_n &= B \\ At_n &= t_{n-1} - a_1 t_n \\ &\;\;. \\ &\;\;. \\ &\;\;. \\ At_2 &= t_1 - a_{n-1} t_n \\ At_1 &= -a_n t_n \end{aligned} \tag{5-12b}$$

where the a_i's are the coefficients of the characteristic polynomial of A. This choice of T leads, to the following general form of the matrices A' and B',

$$A' = T^{-1}AT = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ -a_5 & -a_4 & -a_3 & -a_2 & -a_1 \end{bmatrix} \text{ and } B' = T^{-1}B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \tag{5-13}$$

In the new state equations, the control signal F_1 becomes

$$F_1 = -K^{*}\underset{\sim}{Z} = -K^{*}T^{-1}\underset{\sim}{y} = -K\underset{\sim}{y} \tag{5-14}$$

where $K^{*T} = [k_1^{*}, k_2^{*}, k_3^{*}, k_4^{*}, k_5^{*}]$. Use (5-14) in (5-11),

$$\dot{\underset{\sim}{Z}} = (A^{*} - B^{*}K^{*})\underset{\sim}{Z} + T^{-1}\underset{\sim}{W} \tag{5-15}$$

the closed loop matrix can be written as,

$$A^{*} - B^{*}K^{*} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ (-a_5 - k_1^{*}) & (-a_4 - k_2^{*}) & (-a_3 - k_3^{*}) & (-a_2 - k_4^{*}) & (-a_1 - k_5^{*}) \end{bmatrix} \tag{5-16}$$

The gains are computed by matching the entries of $(A^{*} - B^{*}K^{*})$ with those of the desired matrix A^{*}_d. The latter has the following form,

$$A^{*}_d = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ -a_{d5} & -a_{d4} & -a_{d3} & -a_{d2} & -a_{d1} \end{bmatrix} \tag{5-17}$$

where a_{di}'s are the i^{th} coefficient of the desired characteristic polynomial. The desired eigenvalues are assigned according to settling time and percent overshoot considerations.

The gain matrix, K^{*}, of the new state vector is directly related to the gain matrix, K, of the original state vector through equation (5-14). This leads to,

$$K = K^{*}T^{-1} \tag{5-18}$$

The merits of the two controllers, derived in this section, are compared by applying them on the system of three lumped masses.

5.3. Results and Discussion of the System with Three Lumped Masses

The gains for the rigid body controller are computed to achieve a damping ratio $\xi = 0.53$ and a natural frequency, $w_n = 15$ rad /sec . This controller is applied to the three lumped masses and the results are illustrated in Fig. 5-2 to 5-4. The response of the first mass, x_1, is very oscillatory due to the effect of coupling with the second and third masses. Note that the three lumped mass model (see Fig. 5-1) represents a conservative system. That is, once the system is excited, it should oscillate indefinitely. However, the decay in the x_2 and x_3 responses, in Fig. 5-3 and 5-4, is caused by the damping induced by the rigid body controller.

For best evaluation of the rigid and flexible motion controller, a reduced model, consisting of the first and second mass, is initially used. The exclusion of the third mass eliminates the effect of control spillover. Figures 5-5 and 5-6 represent the responses of the first and second mass respectively. The overshoot is due to the assignment of the desired damping ratio $\xi = 0.53$. The oscillation seen in Fig. 5-2 and 5-3 is completely eliminated due to the additional damping introduced by the rigid and flexible motion controller.

Finally, to show the effect of control spillover, the rigid and flexible motion controller is applied on the three lumped masses. This is illustrated in Fig. 5-7 to 5-9. The responses of x_1 and x_2, which represent the displacements of the first and second mass, become more oscillatory then in Fig. 5-5 and 5-6. This deterioration in the response is expected due to the effect of control spillover. However, the performance of this controller, even in the presence of control

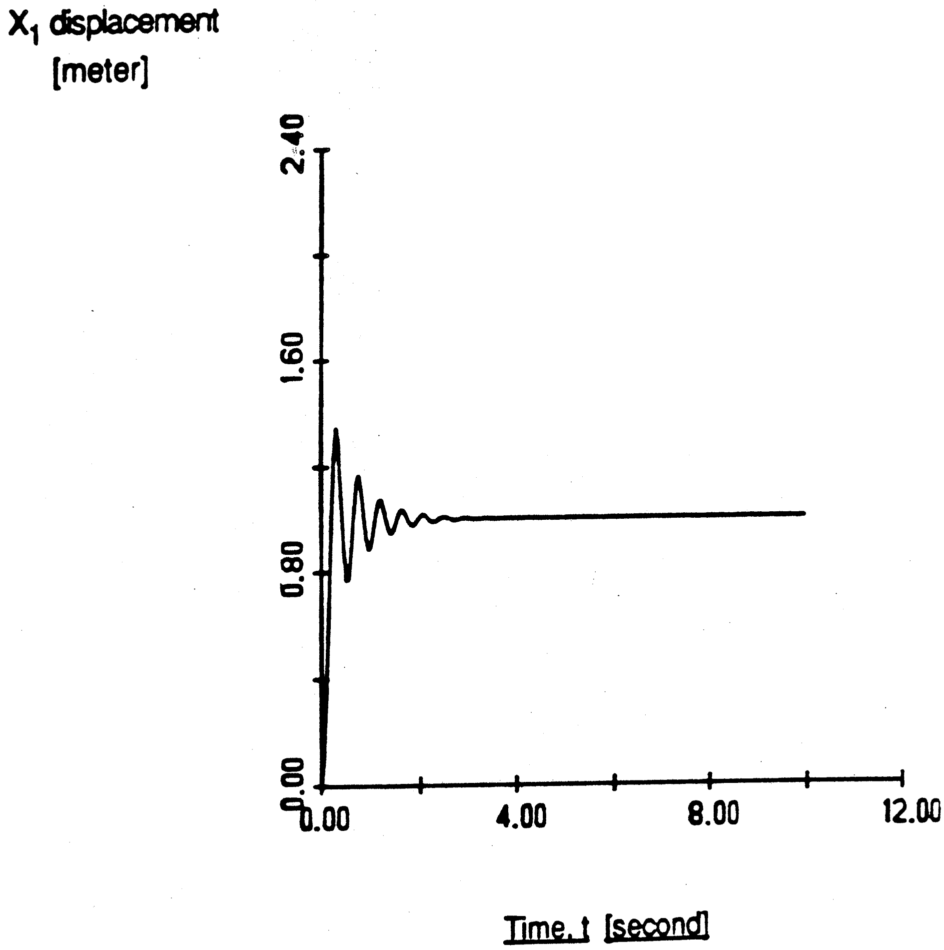


Figure 5-2. First mass response obtained from the rigid body controller.

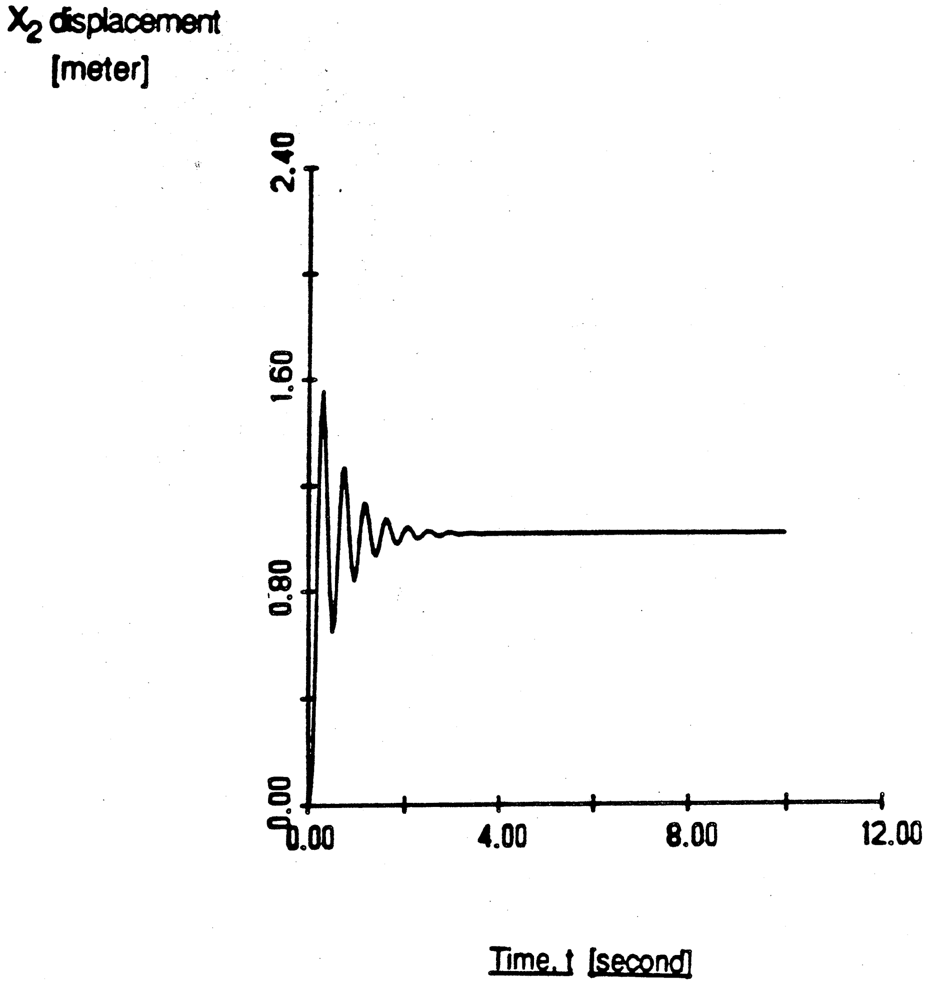


Figure 5-3. Second mass response obtained from the rigid body controller.

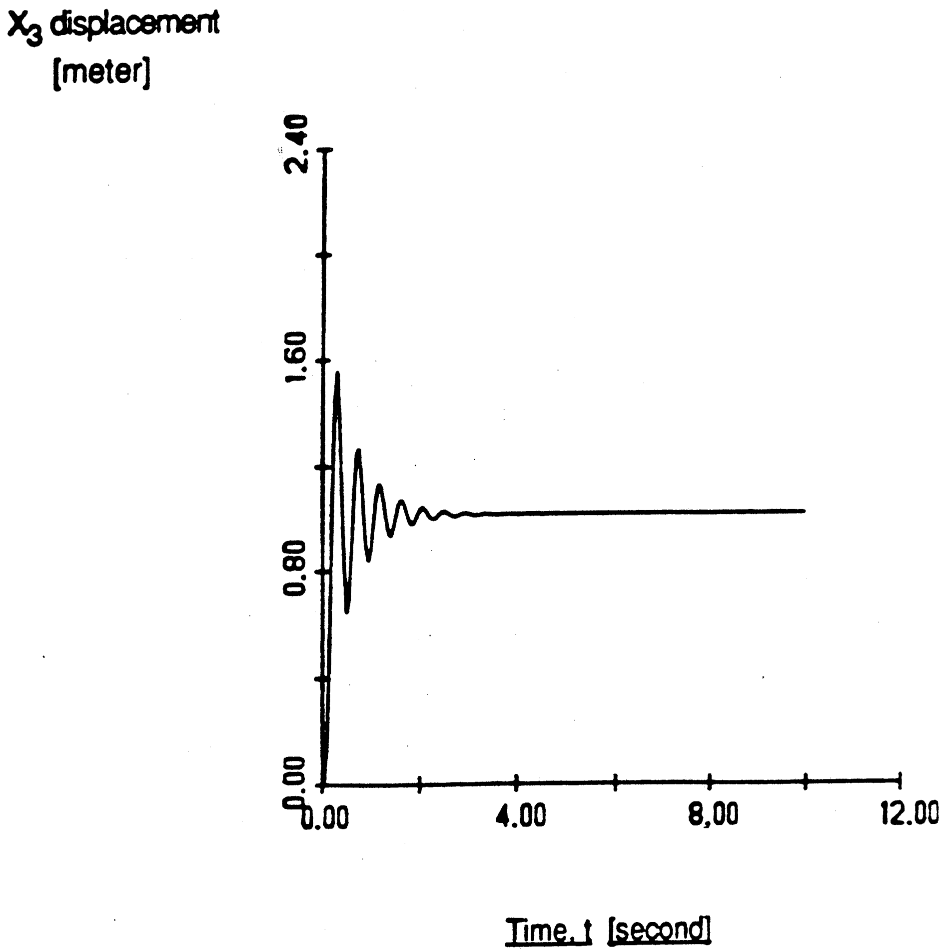


Figure 5-4. Third mass response obtained from the rigid body controller.

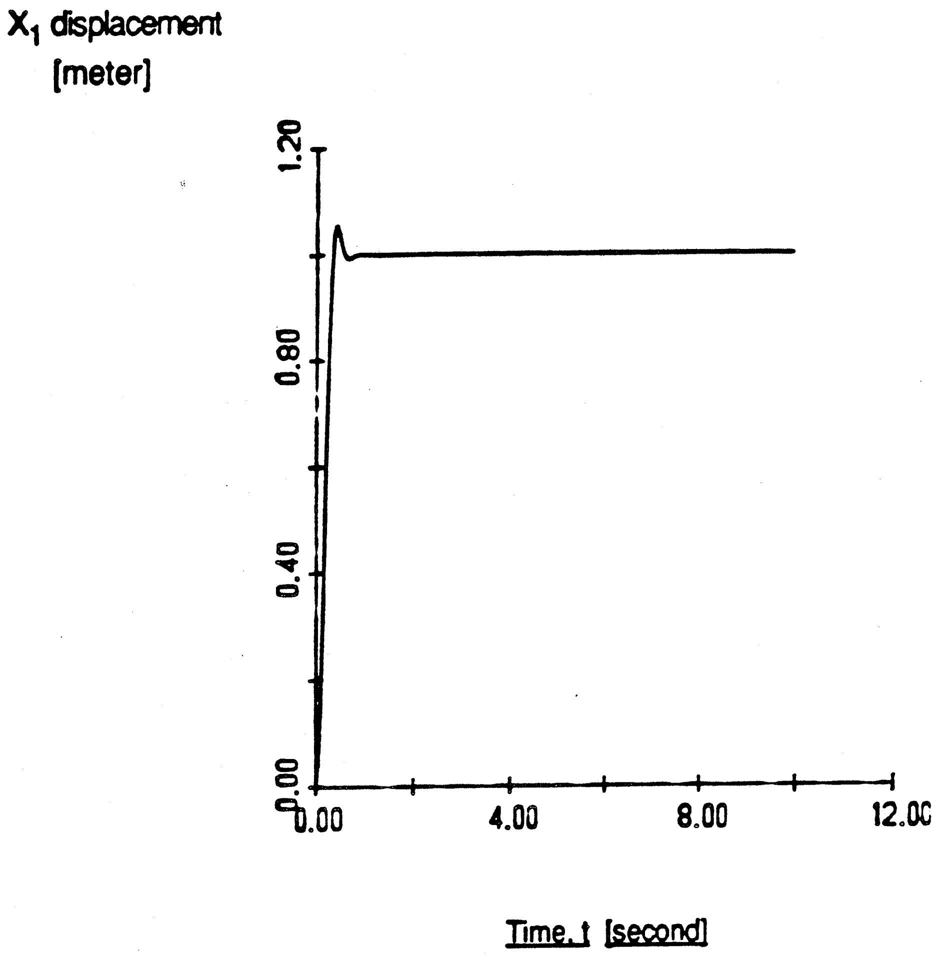


Figure 5-5. First mass response obtained from the rigid and flexible motion controller in the reduced order model case.

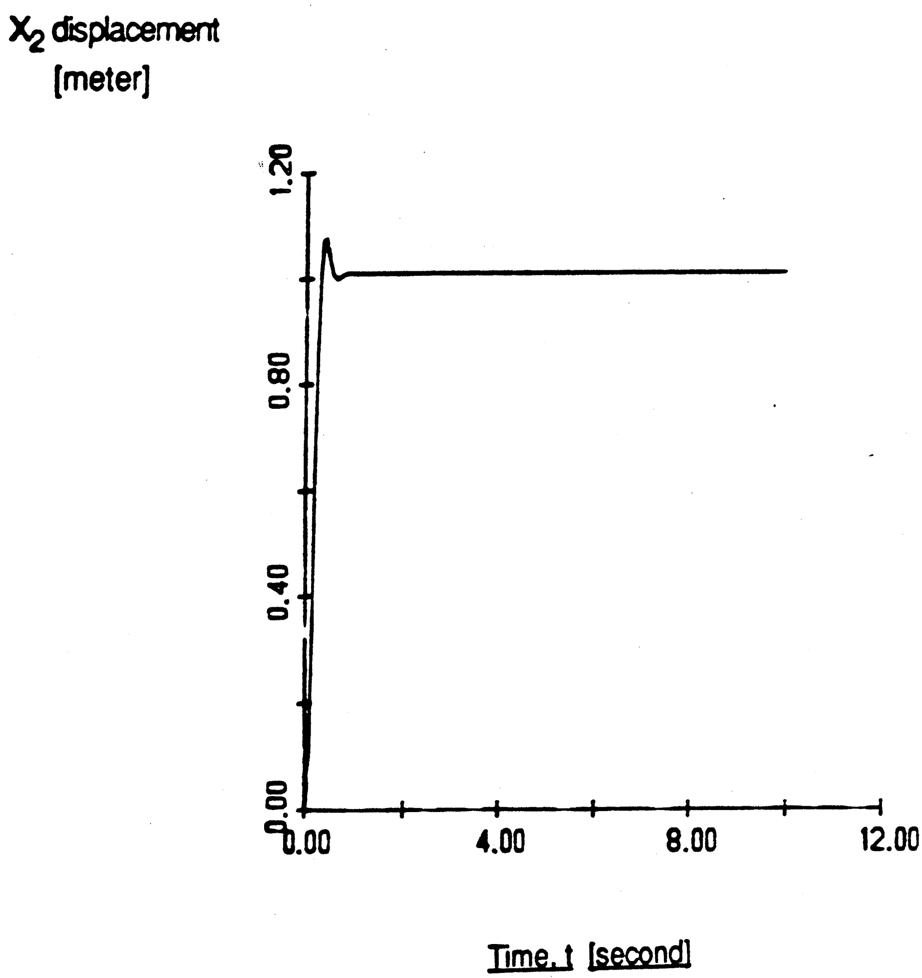


Figure 5-6. Second mass response obtained from the rigid and flexible motion controller in the reduced order model case.

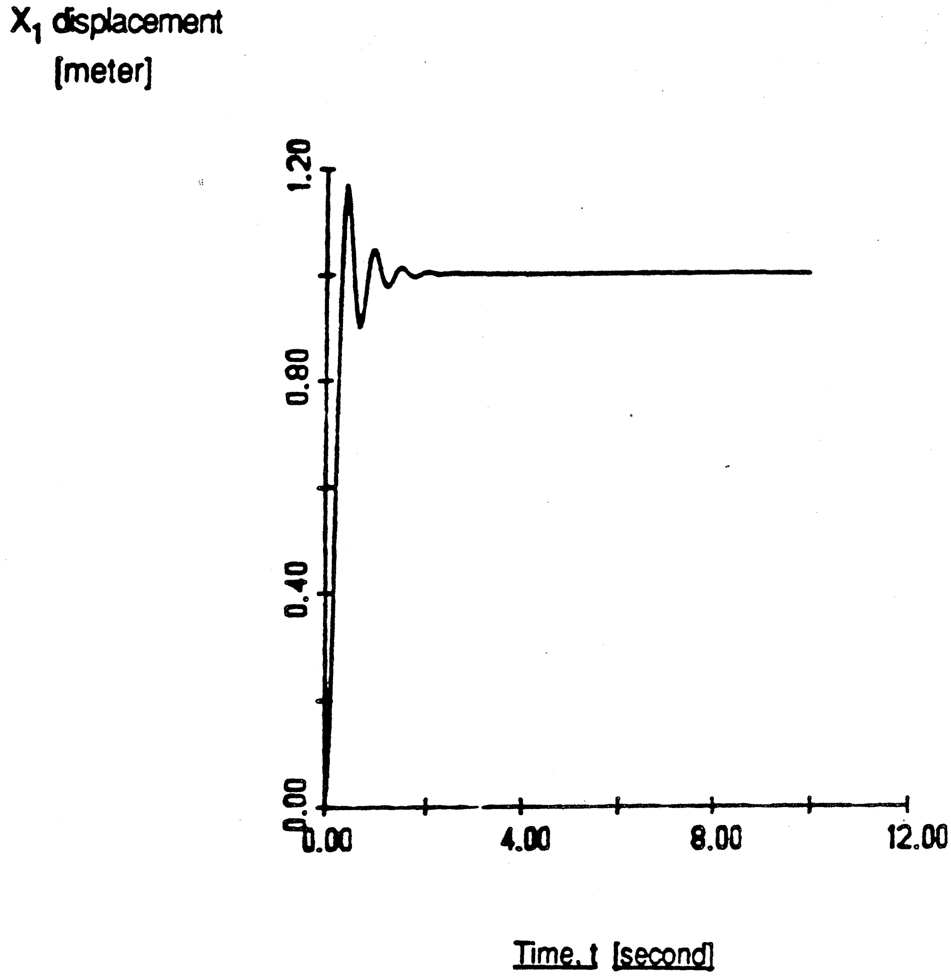


Figure 5-7. First mass response obtained from the rigid and flexible motion controller in the control spillover case.

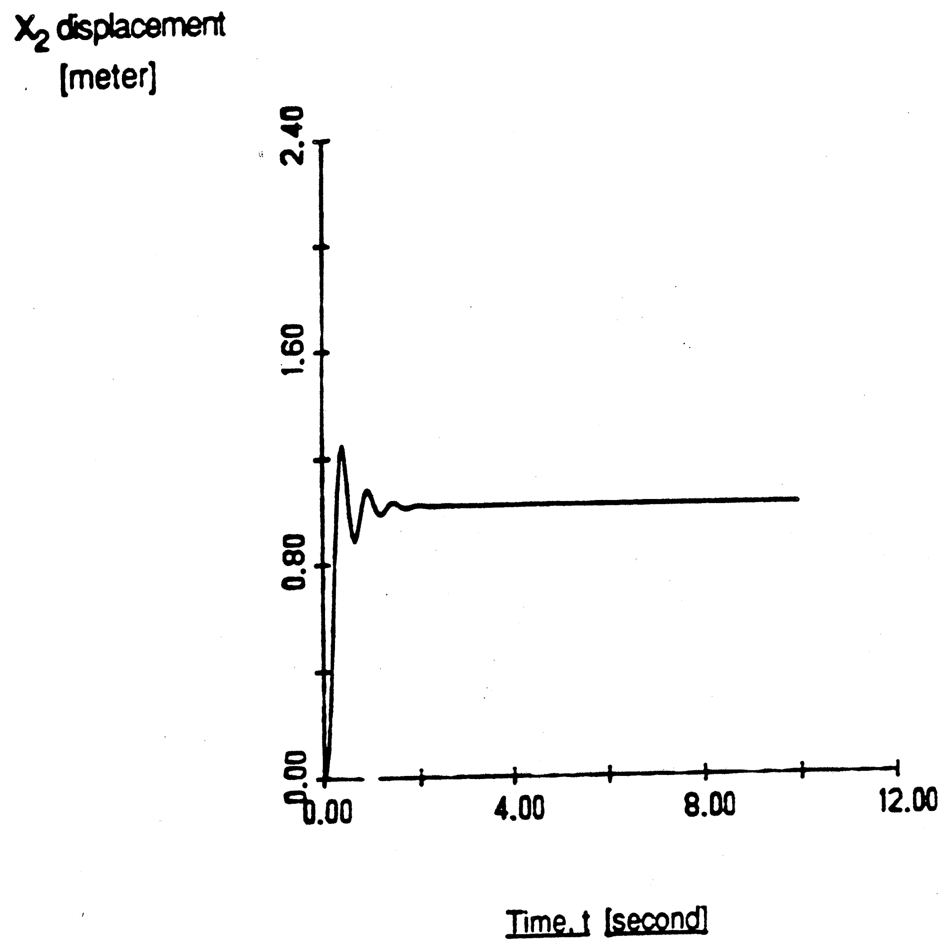


Figure 5-8. Second mass response obtained from the rigid and flexible motion controller in the control spillover case.

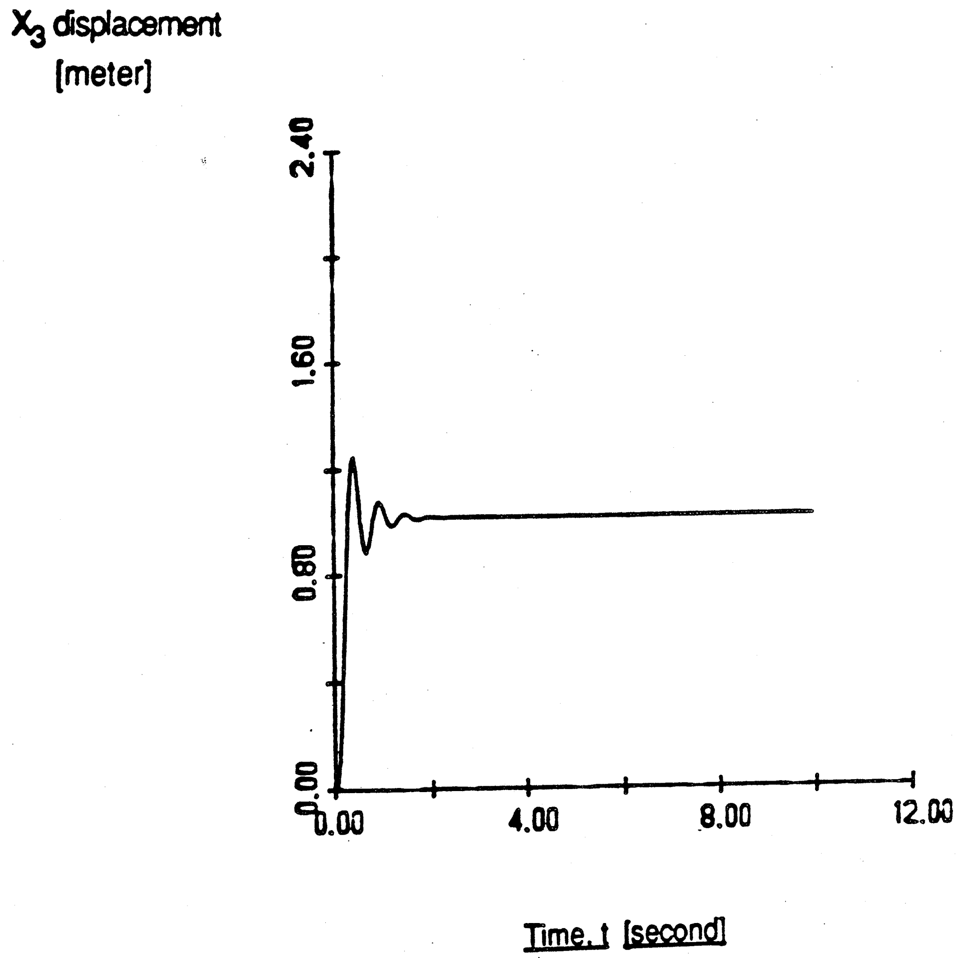


Figure 5-9. Third mass response obtained from the rigid and flexible motion controller in the control spillover case.

spillover, is still superior to the rigid body controller (see Fig. 5-2 to 5-4). The oscillatory motion is approximately reduced to half.

At steady state, the deviation of the second and third mass displacements from the desired value is caused by the static deflection of the springs.

5.4. Linearization of the Rigid and Flexible Equations of Motion

The first step in designing the rigid and flexible motion controller for the robot, is to obtain a linearized version of the equations describing the combined rigid and flexible motions. The latter have the following general form,

$$M(\underset{\sim}{x})\ddot{\underset{\sim}{x}} + F(\underset{\sim}{x},\dot{\underset{\sim}{x}}) = F'\ (\underset{\sim}{T}) \tag{5-19}$$

where $x^T = [r,\theta,\phi,q_{11},q_{12},q_{21},q_{22}]$ and $\underset{\sim}{T}^T = [T_1,T_2,T_3]$. The inertia matrix $M(\underset{\sim}{x})$ is not diagonal. Therefore, the linearization procedure, outlined in section 4-1, cannot be applied directly. An intermediate step, involving the algebraic computation of the inverse inertia matrix, is required to transform (5-19) to the form given in equation (4-2). To avoid this difficulty, another procedure, which is also based on Taylor series expansion, is implemented.

Suppose $\underset{\sim}{T}_n(t),\underset{\sim}{x}_n(t),\dot{\underset{\sim}{x}}_n(t)$ and $\ddot{\underset{\sim}{x}}_n(t)$ satisfies the second order differential equation given in (5-19) for $t \geq 0$. Define

$$\begin{aligned} \underset{\sim}{x}(t) &= \underset{\sim}{x}_n(t) + \delta \underset{\sim}{x}(t) \\ \underset{\sim}{T}(t) &= \underset{\sim}{T}_n(t) + \delta \underset{\sim}{T}(t) \end{aligned} \tag{5-20}$$

Substitute (5-20) into (5-19), to get

$$M(\underset{\sim}{x}_n + \delta \underset{\sim}{x})\left[\ddot{\underset{\sim}{x}}_n + \delta \ddot{\underset{\sim}{x}}\right] + F(\underset{\sim}{x}_n + \delta \underset{\sim}{x},\ \dot{\underset{\sim}{x}}_n + \delta \dot{\underset{\sim}{x}}) = F'\ (\underset{\sim}{T}_n + \delta \underset{\sim}{T}) \tag{5-21}$$

Expand by Taylor series the nonlinear terms around $\underset{\sim}{x}_n$ and $\underset{\sim}{T}_n$. This yields,

$$\left[M(\underset{\sim}{x}_n) + \frac{\partial M(\underset{\sim}{x}_n)}{\partial x}\underset{\sim}{\delta x}\right]\left[\underset{\sim}{\ddot{x}}_n + \underset{\sim}{\delta \ddot{x}}\right] + F(\underset{\sim}{x}_n, \underset{\sim}{\dot{x}}_n) + \frac{\partial F(\underset{\sim}{x}_n, \underset{\sim}{\dot{x}}_n)}{\partial x}\underset{\sim}{\delta x} +$$

$$\frac{\partial F(\underset{\sim}{x}_n, \underset{\sim}{\dot{x}}_n)}{\partial \dot{x}}\underset{\sim}{\delta \dot{x}} = F'(\underset{\sim}{T}_n) + \frac{\partial F'(\underset{\sim}{T}_n)}{\partial T}\underset{\sim}{\delta T} + \text{higher order terms} \tag{5-22}$$

Rearranging the terms,

$$\left\{M(\underset{\sim}{x}_n)\underset{\sim}{\ddot{x}}_n + F(\underset{\sim}{x}_n, \underset{\sim}{\dot{x}}_n) - F'(\underset{\sim}{T}_n)\right\} + M(\underset{\sim}{x}_n)\underset{\sim}{\delta \ddot{x}} + \frac{\partial M(\underset{\sim}{x}_n)}{\partial x}\underset{\sim}{\ddot{x}}_n\underset{\sim}{\delta x} + \frac{\partial M(\underset{\sim}{x}_n)}{\partial x}\underset{\sim}{\delta x}\underset{\sim}{\delta \ddot{x}}$$

$$+ \frac{\partial F(\underset{\sim}{x}_n, \underset{\sim}{\dot{x}}_n)}{\partial x}\underset{\sim}{\delta x} + \frac{\partial F(\underset{\sim}{x}_n, \underset{\sim}{\dot{x}}_n)}{\partial \dot{x}}\underset{\sim}{\delta \dot{x}} = \frac{\partial F'(\underset{\sim}{T}_n)}{\partial T}\underset{\sim}{\delta T} + \text{higher order terms} \tag{5-23}$$

Assuming that the solution, $\underset{\sim}{x}_n$, is at equilibrium (i.e. $\dot{x}_n = \ddot{x}_n = 0$) and retaining only first order terms, equation (5-23) becomes

$$M(\underset{\sim}{x}_n)\underset{\sim}{\delta \ddot{x}} + \frac{\partial F(\underset{\sim}{x}_n, 0)}{\partial x}\underset{\sim}{\delta x} + \frac{\partial F(\underset{\sim}{x}_n, 0)}{\partial \dot{x}}\underset{\sim}{\delta \dot{x}} = \frac{\partial F'(\underset{\sim}{T}_n)}{\partial T}\underset{\sim}{\delta T} \tag{5-24}$$

which can be rearranged to

$$\underset{\sim}{\delta \ddot{x}} = -M^{-1}(\underset{\sim}{x}_n)\frac{\partial F(\underset{\sim}{x}_n, 0)}{\partial x}\underset{\sim}{\delta x} - M^{-1}(\underset{\sim}{x}_n)\frac{\partial F(\underset{\sim}{x}_n, 0)}{\partial \dot{x}}\underset{\sim}{\delta \dot{x}} + M^{-1}(\underset{\sim}{x}_n)\frac{\partial F'(\underset{\sim}{T}_n)}{\partial T}\underset{\sim}{\delta T} \tag{5-25}$$

where $M^{-1}(\underset{\sim}{x}_n)$ is the inverse of the inertia matrix $M(\underset{\sim}{x})$ evaluated at $\underset{\sim}{x}_n$. In matrix form, equation (5-25) can be written as,

$$\begin{Bmatrix} \underset{\sim}{\delta \dot{x}} \\ \underset{\sim}{\delta \ddot{x}} \end{Bmatrix} = \begin{bmatrix} O_{7\times 7} & I_{7\times 7} \\ -M^{-1}(\underset{\sim}{x}_n)\dfrac{\partial F(\underset{\sim}{x}_n, 0)}{\partial x} & -M^{-1}(\underset{\sim}{x}_n)\dfrac{\partial F(\underset{\sim}{x}_n, 0)}{\partial \dot{x}} \end{bmatrix}\begin{Bmatrix} \underset{\sim}{\delta x} \\ \underset{\sim}{\delta \dot{x}} \end{Bmatrix} + \begin{bmatrix} O_{7\times 3} \\ M^{-1}(\underset{\sim}{x}_n)\dfrac{\partial F'}{\partial T} \end{bmatrix}\underset{\sim}{\delta T} \tag{5-26}$$

The linearized form of the equations of motion for the rigid and flexible motion can be readily obtained by evaluating equation (5-26).

5.5. Derivation of the Rigid and Flexible Motion Controller

This section, is a very important one, in that it contains a detailed discussion of how the flexible motion is treated in the design of the rigid and flexible motion controller.

Any compliant link would undergo static and dynamic deflections. There are two approaches with which the static deflection problem can be handled. In the first approach, the static deflection is computed theoretically and then taken into consideration in specifying the desired end effector final position to the controller. In the second approach, additional sensors and actuators are used. The static deflection is measured and corrected for by the additional actuator referred to as a "straightness servo" in [4]. In this work, no attempt is made to compensate for the static deflection, as this can be treated as a separate problem.

The control objective is to introduce additional damping into the flexible motion. This is done by using additional sensors to measure the compliant link dynamic vibrations and feed them back to the controller. Theoretically, any flexible system has an infinite number of elastic modes. Due to physical limitations, a limited number of sensors and actuators can be applied, thus restricting the controller design to the few critical modes. Note that the outputs of the sensors would contain information about the unmodeled as well as the modeled modes. This is referred to as observation spillover. Similarly, the control action would affect both the modeled and unmodeled modes leading to the control spillover phenomenon. Based on the work done by Balas in [30], it is expected that the higher unmodeled modes can cause detrimental effect on the system response. It can even lead to instability. To examine the effect of observation and control spillover, only the first elastic mode in the vertical and horizontal directions are considered in the controller design. The second mode is considered to be a representative of the higher unmodeled modes.

The rigid and flexible motion controller is designed based on the linearized equations of motion given in (5-26). An integral plus state feedback controller is implemented. The integral action is applied on the joint angles to insure that they reach their desired values with zero steady state error. Define the state vector, $\underset{\sim}{y}$, and the control vector, $\underset{\sim}{u}$, to be,

$$\underset{\sim}{y} = \left\{ \frac{\delta x}{\delta \dot{x}} \right\}_{\sim} \qquad \underset{\sim}{u} = \underset{\sim}{\delta T} \tag{5-27}$$

Then equation (5-26) can be written in the following general form,

$$\dot{\underset{\sim}{y}} = A\underset{\sim}{y} + B\underset{\sim}{u} \tag{5-28}$$

This represents a linear, time invariant, multi-input multi-output system. Three additional state variables are introduced to facilitate the implementation of the integral plus state feedback controller,

$$y_{11} = \int (r - R_1)dt \qquad y_{12} = \int (\theta - R_2)dt \qquad y_{13} = \int (\phi - R_3)dt$$

This embeds the integral action into the state equations. Thus allowing the control action to be expressed as,

$$\underset{\sim}{u} = -K^F \underset{\sim}{y} \tag{5-29}$$

where K^F is the feedback gain matrix of the rigid and flexible motion controller.

Substitute (5-29) into (5-28), to get

$$\dot{\underset{\sim}{y}} = (A - BK^F)\underset{\sim}{y} \tag{5-30}$$

The eigenvalues of the closed loop matrix, $(A - BK^F)$, can be arbitrarily assigned by the controller, defined in (5-29), as long as the system given by (5-28) is controllable [55]. Note that for a given set of eigenvalues, the solution for the gain matrix is not unique.

The eigenvalues determine the stability and the speed of the response where the closed loop eigenvectors reshape the transient response. The non-uniqueness of the state feedback gain matrix, K^F, in pole assignment provides the designer the freedom to partially select the closed loop eigenvectors [56], [57].

Standard procedures employ a transformation matrix to convert the linearized system to controllable canonical form for arbitrary eigenvalues assignment. Note that the set of eigenvectors are assigned by the particular choice of the transformation matrix. One such systematic method [53] is used to compute the gain matrix K^F.

First, a state feedback is introduced to make the system controllable by a single component of the $\underset{\sim}{u}$ vector. This converts the multi-input system to single input one. Second, the method established for single-input systems, which was used earlier in the three lumped mass case, is employed to assign the desired closed loop eigenvalues. Note that the gain matrix is not unique for a given set of eigenvalues. The methodology, described here, does not make any attempt to use the freedom given by the nonuniqueness of the state feedback gain to partially assign the closed loop eigenvectors. The latter, which determines the shape of the transient response, ends up being arbitrarily assigned; thus, leading, in most cases, to poor transient response.

This method is employed on the system given by (5-28). Although, the steady state response followed exactly the reference input, the transient response was very oscillatory. Most of the control effort was carried out by a single component of the control vector, $\underset{\sim}{u}$, thus rendering this particular control design to be completely unacceptable from a practical point of view.

Being aware of these difficulties, the control problem is now approached from a physical point of view. No transformation matrix is used. Therefore, all the state variables have physical interpretations. The main task of the controller is to move the robot arm from one position to another with the least amount of vibration possible. From the results of the rigid body controller (see Fig. 4-2 to 4-13), the rigid body degrees of freedom r, θ and ϕ exhibit a critically damped behavior with zero steady state error. However, the transverse deflection in both the vertical and the horizontal directions are oscillatory. In light of this, the rigid and flexible motion controller is obtained by expanding the rigid body controller with the emphasis on introducing additional damping into the flexible system. To do this, the terms in equation (5-26) which represents the open loop case, have to be physically interpreted. $M^{-1}(\underset{\sim}{x}_n)\dfrac{\partial F(\underset{\sim}{x}_n,0)}{\partial x}$ consists of all the stiffness terms whereas $M^{-1}(\underset{\sim}{x}_n)\dfrac{\partial F(\underset{\sim}{x}_n,0)}{\partial \dot{x}}$ contains all the damping terms. By evaluating equation (5-28) and including the integral action one obtains,

$$
\begin{Bmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \\ \dot{y}_4 \\ \dot{y}_5 \\ \dot{y}_6 \\ \dot{y}_7 \\ \dot{y}_8 \\ \dot{y}_9 \\ \dot{y}_{10} \\ \dot{y}_{11} \\ \dot{y}_{12} \\ \dot{y}_{13} \end{Bmatrix}
=
\left|\begin{array}{ccccc|ccccc|ccc}
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
\hline
A_{61} & A_{62} & 0 & A_{64} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
A_{71} & A_{72} & 0 & A_{74} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & A_{85} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
A_{91} & A_{92} & 0 & A_{94} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & A_{10,5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right|
\begin{Bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \\ y_{11} \\ y_{12} \\ y_{13} \end{Bmatrix}
$$

$$
+ \left[\begin{array}{ccc}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
\hline
B_{61} & B_{62} & 0 \\
B_{71} & B_{72} & 0 \\
0 & 0 & B_{83} \\
B_{91} & B_{92} & 0 \\
0 & 0 & B_{10,3} \\
\hline
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0
\end{array}\right]
\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix}
\tag{5-31}
$$

where the A_{ij} and B_{ij} terms represent the open loop terms. This yields $M^{-1}(\underset{\sim}{x}_n)\dfrac{\partial F(\underset{\sim}{x}_n,0)}{\partial \dot{x}} = 0$. That is no structural damping is considered. This is consistent with the robot arm dynamic modeling assumption. Applying the rigid body controller,

$$\underset{\sim}{u} = -K^r \underset{\sim}{y}$$

where K^r represents the feedback gain matrix of the rigid body controller.

$$K^r = \begin{bmatrix} k^r_{11} & k^r_{12} & 0 & 0 & 0 & k^r_{16} & 0 & 0 & 0 & 0 & k^f_{1,11} & 0 & 0 \\ k^r_{21} & k^r_{22} & 0 & 0 & 0 & 0 & k^r_{27} & 0 & 0 & 0 & 0 & k^f_{2,12} & 0 \\ 0 & 0 & k^r_{33} & 0 & 0 & 0 & 0 & k^r_{38} & 0 & 0 & 0 & 0 & k^f_{3,13} \end{bmatrix} \tag{5-32}$$

the closed loop matrix $(A - BK^r)$ becomes,

$$(A - BK^r) = \left| \begin{array}{ccccc|ccccc|ccc} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline A_{61} - \alpha_1 & A_{62} - \alpha_1 & 0 & A_{64} & 0 & -\alpha_1 & -\alpha_1 & 0 & 0 & 0 & -\alpha_1 & -\alpha_1 & 0 \\ A_{71} - \alpha_1 & A_{72} - \alpha_1 & 0 & A_{74} & 0 & -\alpha_1 & -\alpha_1 & 0 & 0 & 0 & -\alpha_1 & -\alpha_1 & 0 \\ 0 & 0 & -\alpha_1 & 0 & A_{85} & 0 & 0 & -\alpha_1 & 0 & 0 & 0 & 0 & -\alpha_1 \\ A_{91} - \alpha_1 & A_{92} - \alpha_1 & 0 & A_{94} & 0 & -\alpha_1 & -\alpha_1 & 0 & 0 & 0 & -\alpha_1 & -\alpha_1 & 0 \\ 0 & 0 & -\alpha_1 & 0 & A_{10,5} & 0 & 0 & -\alpha_1 & 0 & 0 & 0 & 0 & -\alpha_1 \\ \hline 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right|$$

$$\tag{5-33}$$

where α_1 terms represent the entries altered or created by the rigid body controller. The portion of the closed loop matrix that corresponds to $M^{-1}(\underset{\sim}{x}_n)\dfrac{\partial F(\underset{\sim}{x}_n, 0)}{\partial \underset{\sim}{x}}$

has nonzero terms. The latter represent the damping introduced by the rigid body controller in both the rigid and flexible motions. This is consistent with the decaying oscillation obtained in the transverse deflection for both the vertical and horizontal directions (see Fig. 4-5 to 4-8).

Since the main emphasis of the rigid and flexible motion controller is to introduce additional damping into the flexible system, then the gain matrix of the rigid body controller is modified to give

$$K^F = \begin{bmatrix} k^r_{11} & k^r_{12} & 0 & 0 & 0 & k^r_{16} & 0 & 0 & k^F_{19} & 0 & k^f_{1,11} & 0 & 0 \\ k^r_{21} & k^r_{22} & 0 & 0 & 0 & 0 & k^r_{27} & 0 & k^F_{29} & 0 & 0 & k^f_{2,12} & 0 \\ 0 & 0 & k^r_{33} & 0 & 0 & 0 & 0 & k^r_{38} & 0 & k^F_{3,10} & 0 & 0 & k^f_{3,13} \end{bmatrix} \tag{5-34}$$

The corresponding closed loop matrix can be written as follows,

$$(A - BK^F) = \left|\begin{array}{ccccc|ccccc|ccc} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hdashline A_{61}-\alpha_1 & A_{62}-\alpha_1 & 0 & A_{64} & 0 & -\alpha_1 & -\alpha_1 & 0 & -\alpha_2 & 0 & -\alpha_1 & -\alpha_1 & 0 \\ A_{71}-\alpha & A_{72}-\alpha_1 & 0 & A_{74} & 0 & -\alpha_1 & -\alpha_1 & 0 & -\alpha_2 & 0 & -\alpha_1 & -\alpha_1 & 0 \\ 0 & 0 & -\alpha_1 & 0 & A_{85} & 0 & 0 & -\alpha_1 & 0 & -\alpha_2 & 0 & 0 & -\alpha_1 \\ A_{91}-\alpha_1 & A_{92}-\alpha_1 & 0 & A_{94} & 0 & -\alpha_1 & -\alpha_1 & 0 & -\alpha_2 & 0 & -\alpha_1 & -\alpha_1 & 0 \\ 0 & 0 & -\alpha_1 & 0 & A_{10,5} & 0 & 0 & -\alpha_1 & 0 & -\alpha_2 & 0 & 0 & -\alpha_1 \\ \hdashline 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right| \tag{5-35}$$

where $A_{i,j}$'s terms are obtained from the linearized version of the rigid and flexible motion equations. α_1 terms represent the entries altered or created by the rigid body controller and α_2 terms represent the entries created to induce more damping into the flexible system. Note that the selection of the gains k_{19}^F, k_{29}^F and $k_{3,10}^F$ should be done very carefully, since they simultaneously introduce new terms in the rigid and flexible motion equations. Thus, the larger these gains are, the more damping of the transverse deflections can be achieved and the greater is the effect of the flexible motion on the rigid body motion. This causes the latter to become oscillatory. Therefore a compromise should be made in introducing as much damping as possible while keeping the effect of the flexible motion on the rigid body motion to a minimum.

5.6. Results and Discussion of the Rigid and Flexible Motion Controller

The rigid and flexible motion controller is first applied on a reduced order model where only one elastic mode is considered to represent the flexible motion in each the horizontal and the vertical directions. The rationale is to eliminate the effect of observation and control spillover. The results are illustrated in Fig. 5-10 to 5-17. The first three plots show the expected critically damped response of the rigid body degrees of freedom r,θ and ϕ. They are identical to their counterparts in the base run of the rigid body controller (see Fig.4-2 to 4-4). The first mode in the vertical and the horizontal directions are presented in Fig. 5-13 and 5-14. The oscillations are due to the inertial forces emanating from the fast and

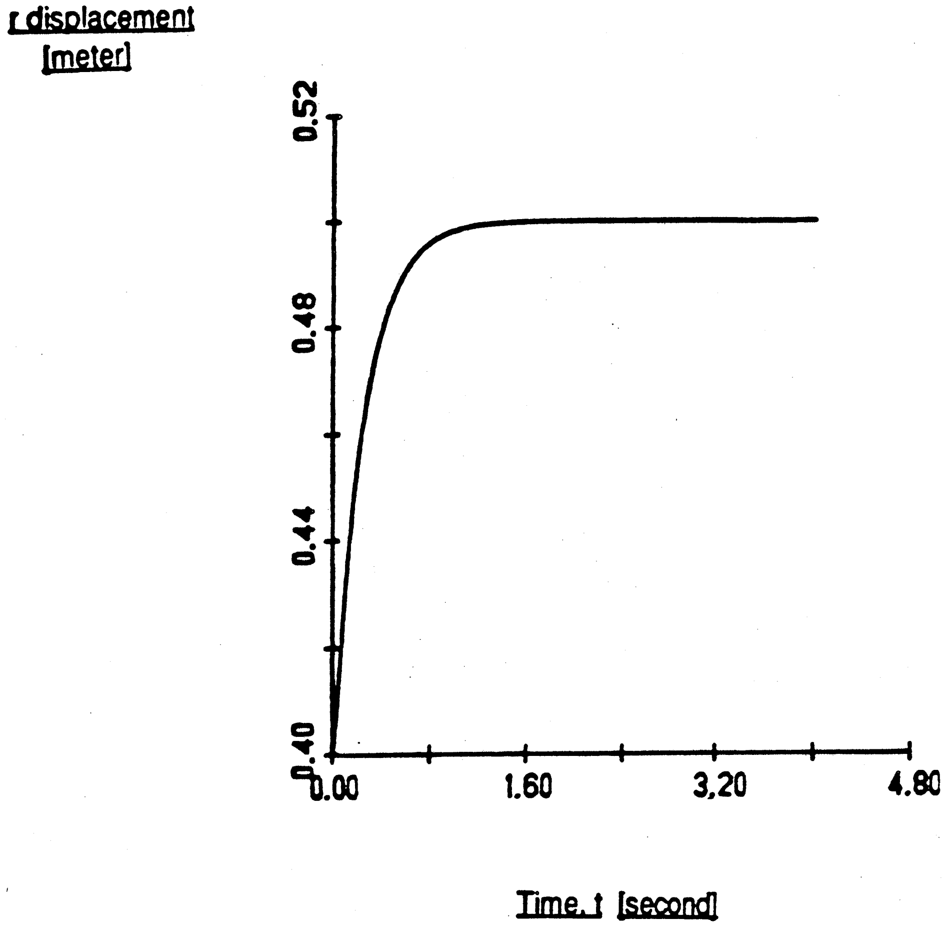


Figure 5-10. r response obtained from the rigid and flexible motion controller in the reduced order model case.

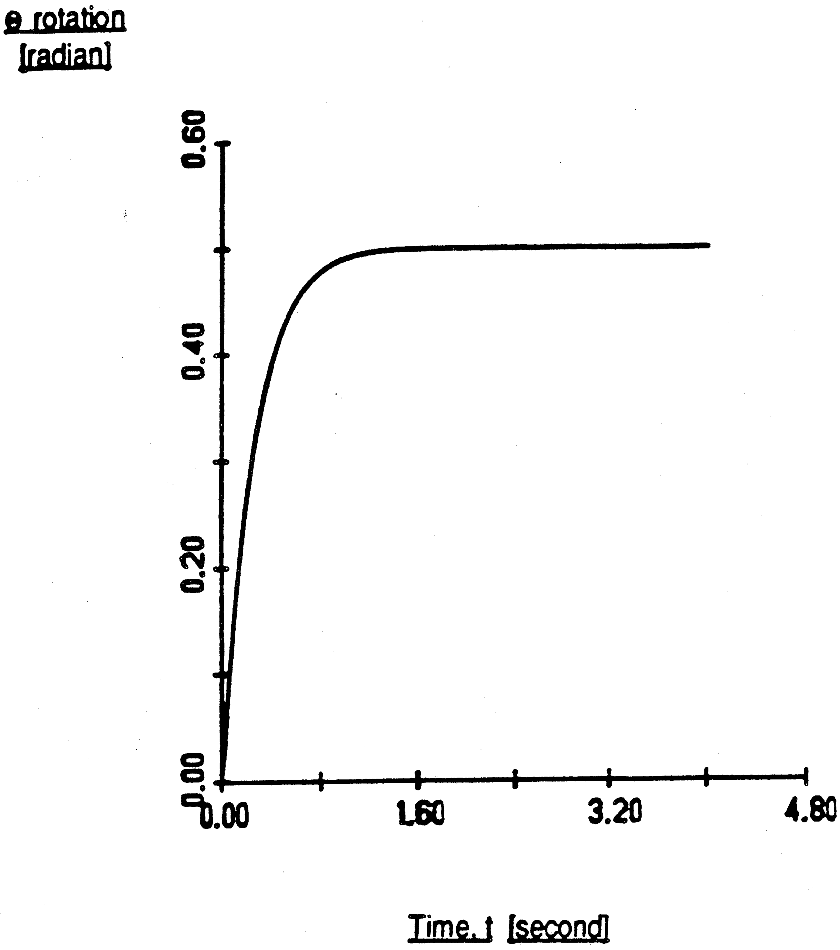


Figure 5-11. θ response obtained from the rigid and flexible motion controller in the reduced order model case.

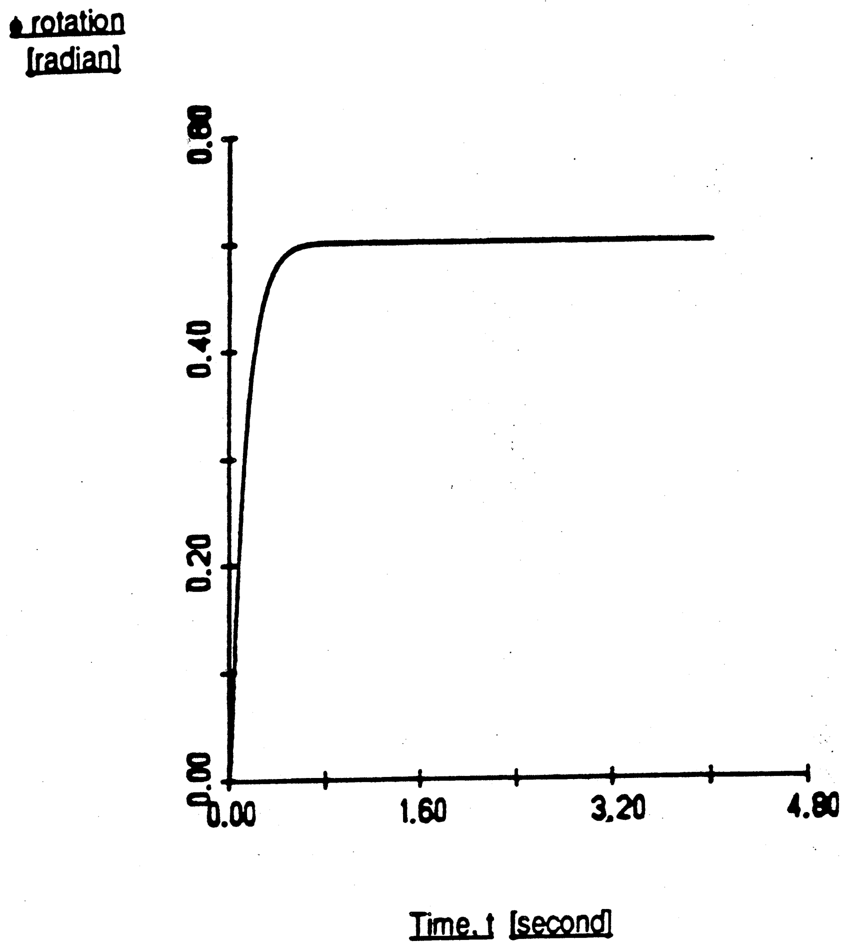


Figure 5-12. φ response obtained from the rigid and flexible motion controller in the reduced order model case.

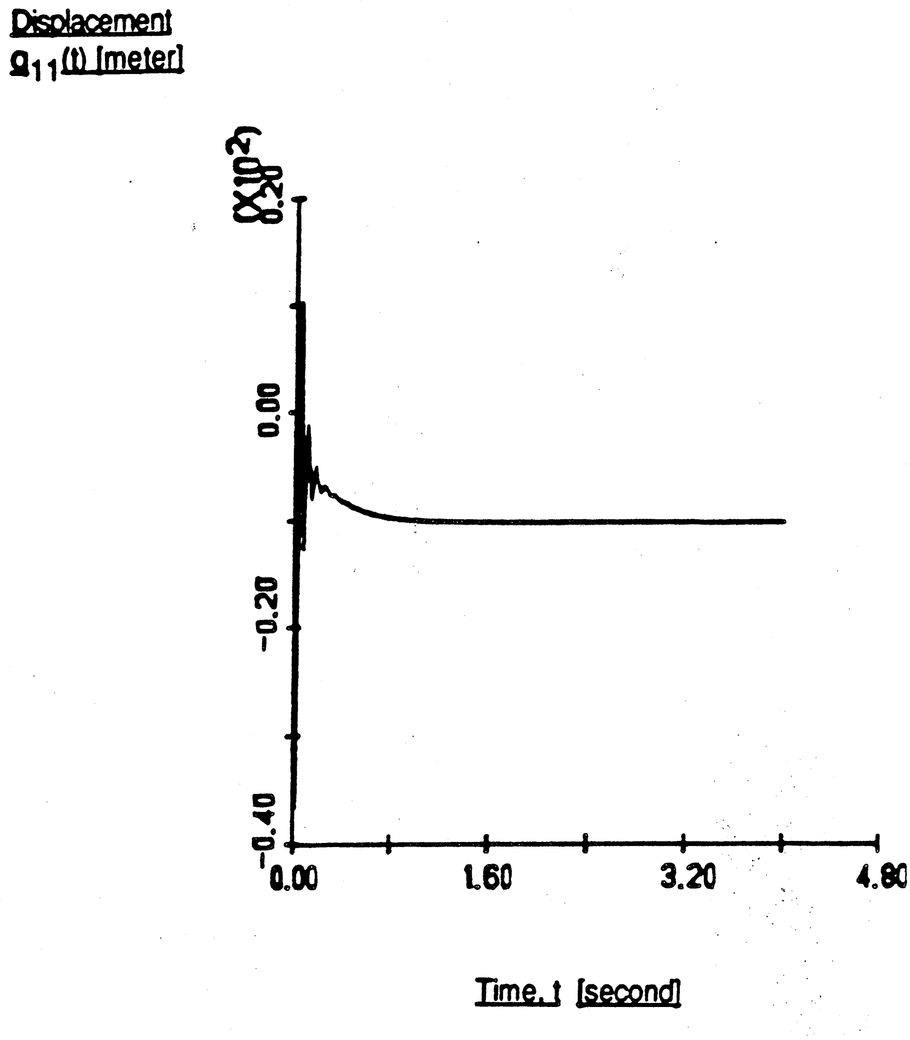


Figure 5-13. Flexible motion coordinate $q_{11}(t)$ in response to the rigid and flexible motion controller in the reduced order model case.

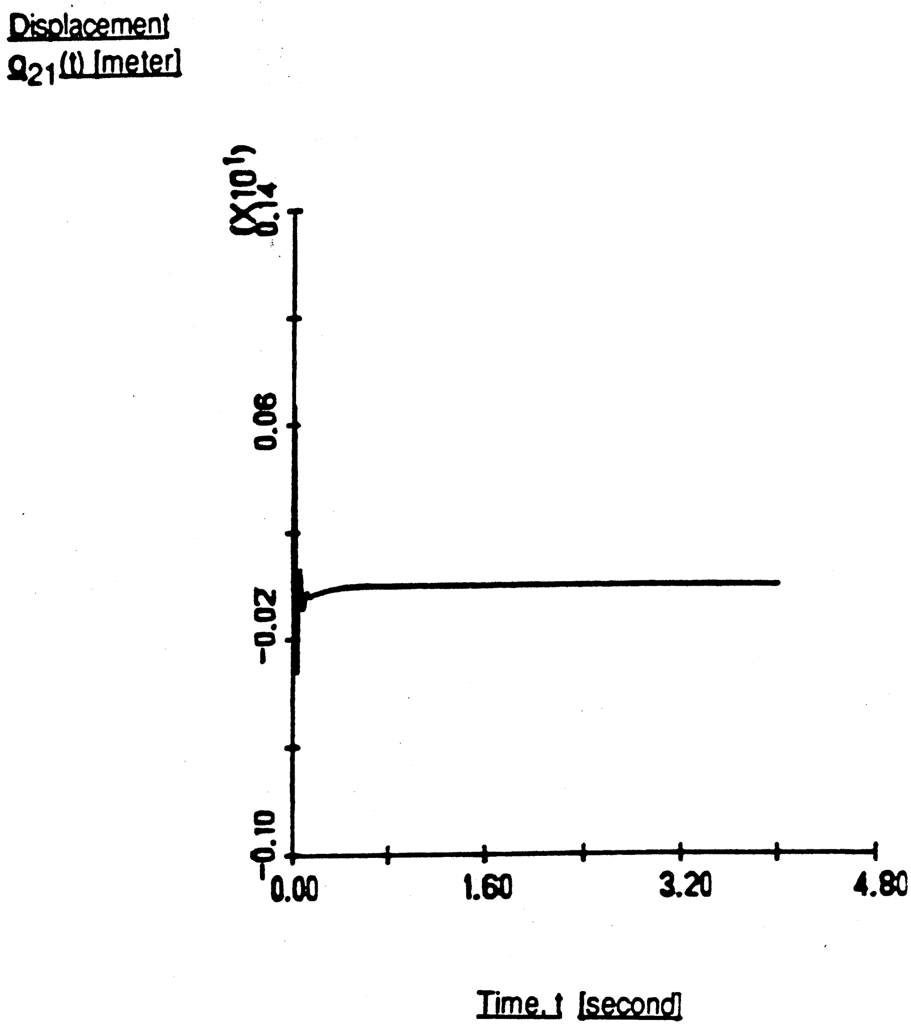


Figure 5-14. Flexible motion coordinate $q_{21}(t)$ in response to the rigid and flexible motion controller in the reduced order model case.

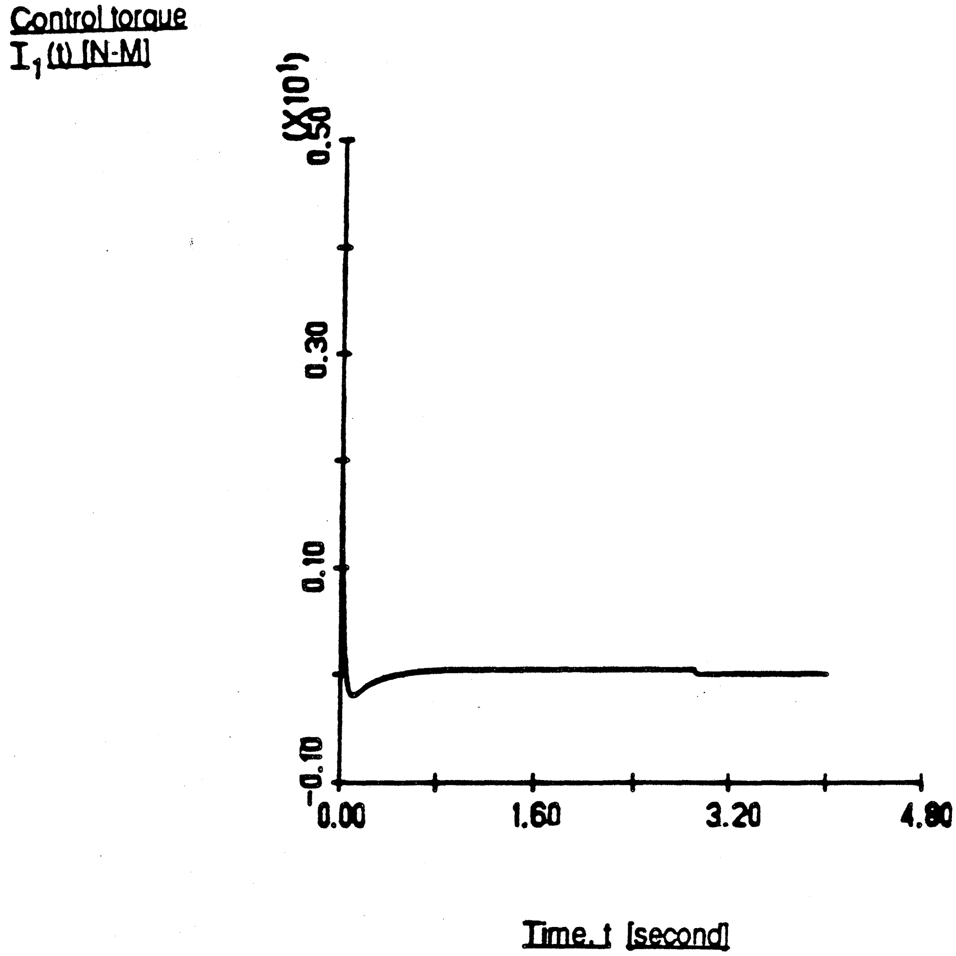


Figure 5-15. Control signal for the prismatic joint obtained from the rigid and flexible motion controller in the reduced order model case.

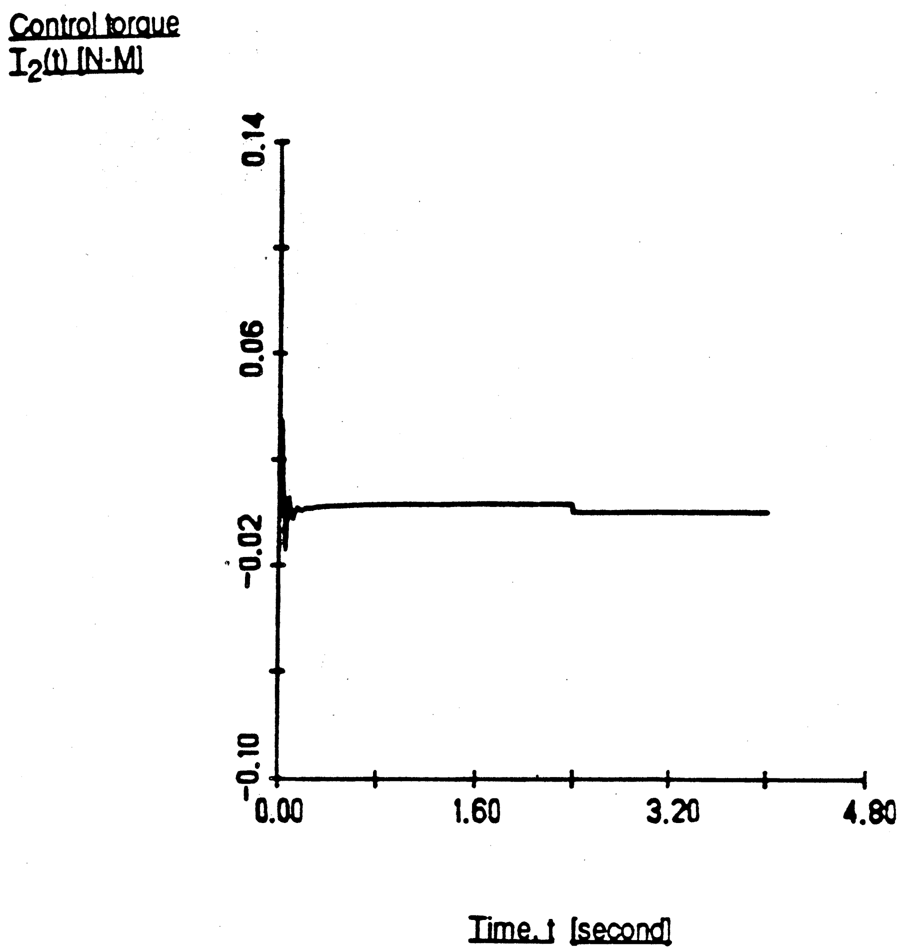


Figure 5-16. Control signal for the second joint obtained from the rigid and flexible motion controller in the reduced order model case.

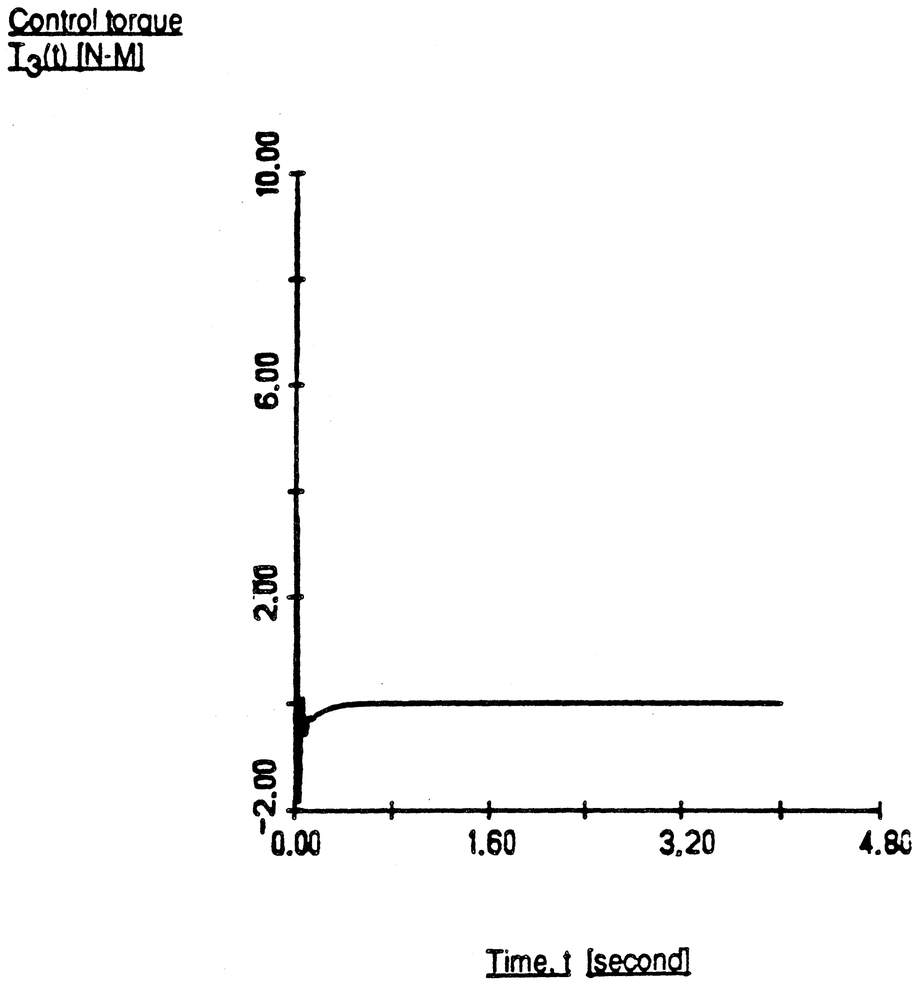


Figure 5-17. Control signal for the base joint obtained from the rigid and flexible motion controller in the reduced order model case.

sudden rigid body motions. However, these vibrations have rapidly died out, thus reflecting the additional amount of damping introduced by including the flexible motion into the control action. Finally, Fig. 5-15 to 5-17 show the applied control torques T_1, T_2 and T_3. The oscillations in the control signals show the effect of the flexible motion on the rigid body motion.

Additional runs are made to study the following effects:

(1) *Control spillover:* The rigid and flexible motion controller is now applied on the full rigid and flexible motion model. (i.e. two elastic modes are used to represent the transverse vibration in both the vertical and horizontal directions). This is done to assess the effect of control spillover on the second uncontrolled elastic mode. The main results are shown in Fig. 5-18 to 5-23. The first four plots represent the first and second elastic modes in the vertical and horizontal directions respectively. Even though the responses corresponding to the first mode become more oscillatory then the previous case, the overall performance of the flexible motion coordinates show a reduction of approximately 50% in magnitude when compared with their counterparts in the base run of the rigid body controller. The large oscillation in the control torques T_2 and T_3, shown in Fig. 5-22 and 5-23, reflects the greater effect of the flexible motion on the rigid body motion.

(2) *Control and observation spillover:* This run simulates an actual situation where both control and observation spillover can be present. In the experimental set up, accelerometers mounted at the end effector, are implemented to measure the tip acceleration. The latter is then integrated twice to yield

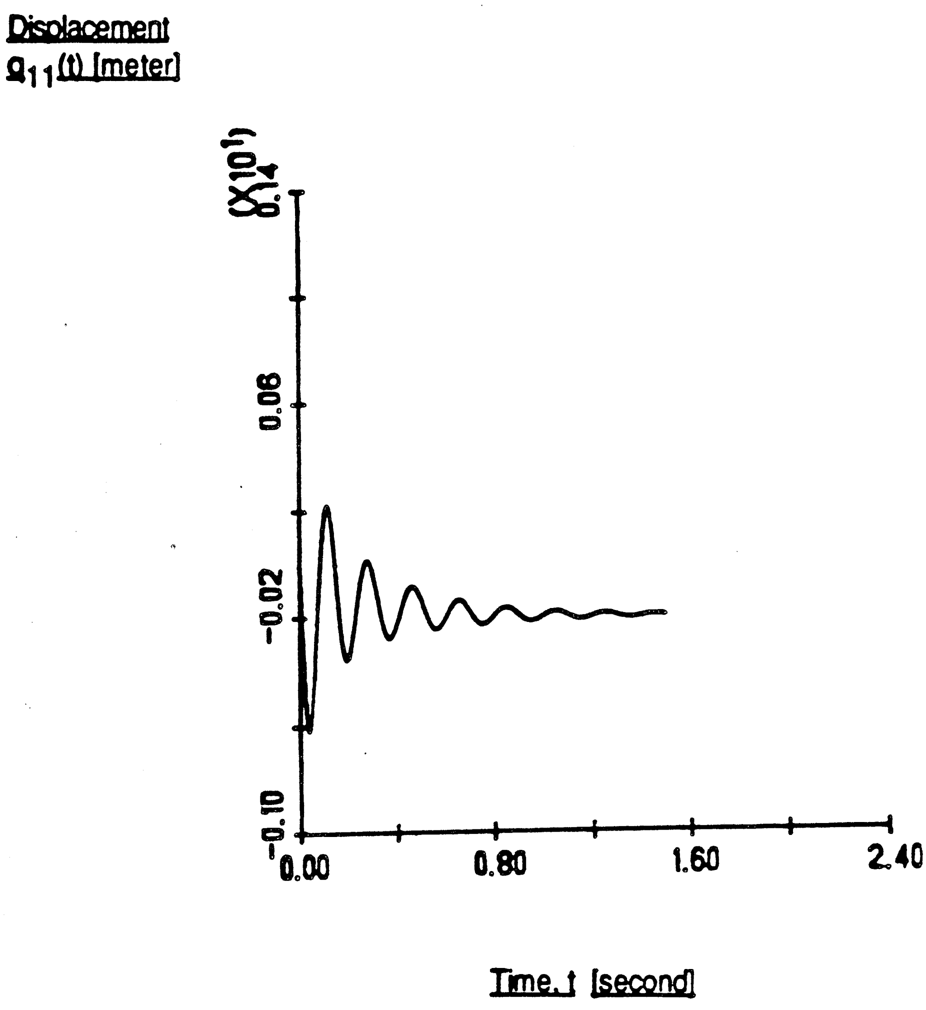


Figure 5-18. Flexible motion coordinate $q_{11}(t)$ in response to the rigid and flexible motion controller in the control spillover case.

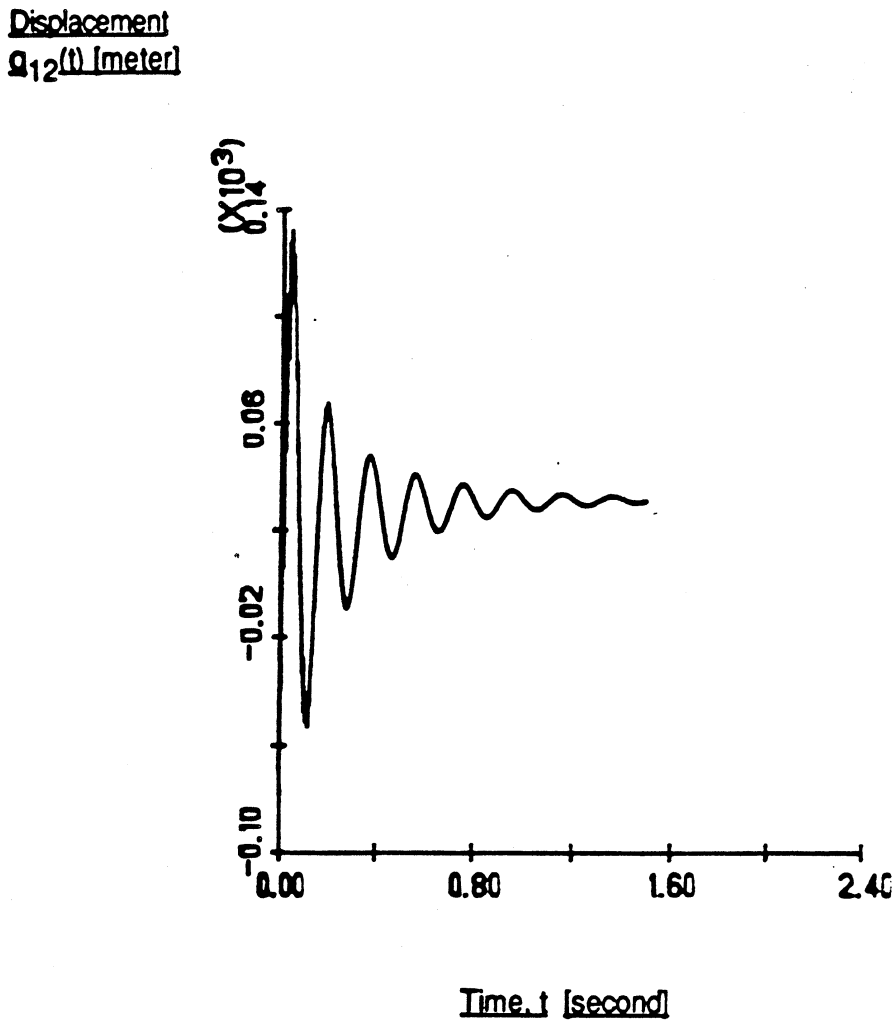


Figure 5-19. Flexible motion coordinate $q_{12}(t)$ in response to the rigid and flexible motion controller in the control spillover case.

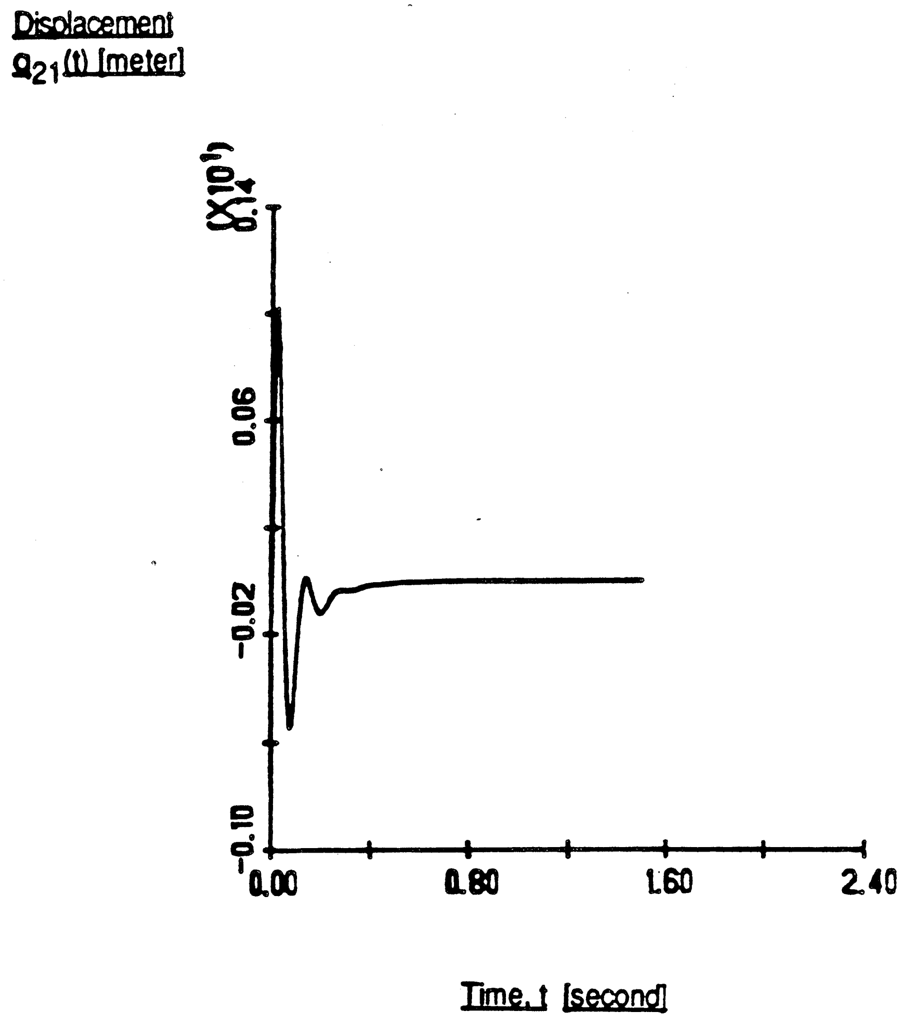


Figure 5-20. Flexible motion coordinate $q_{21}(t)$ in response to the rigid and flexible motion controller in the control spillover case.

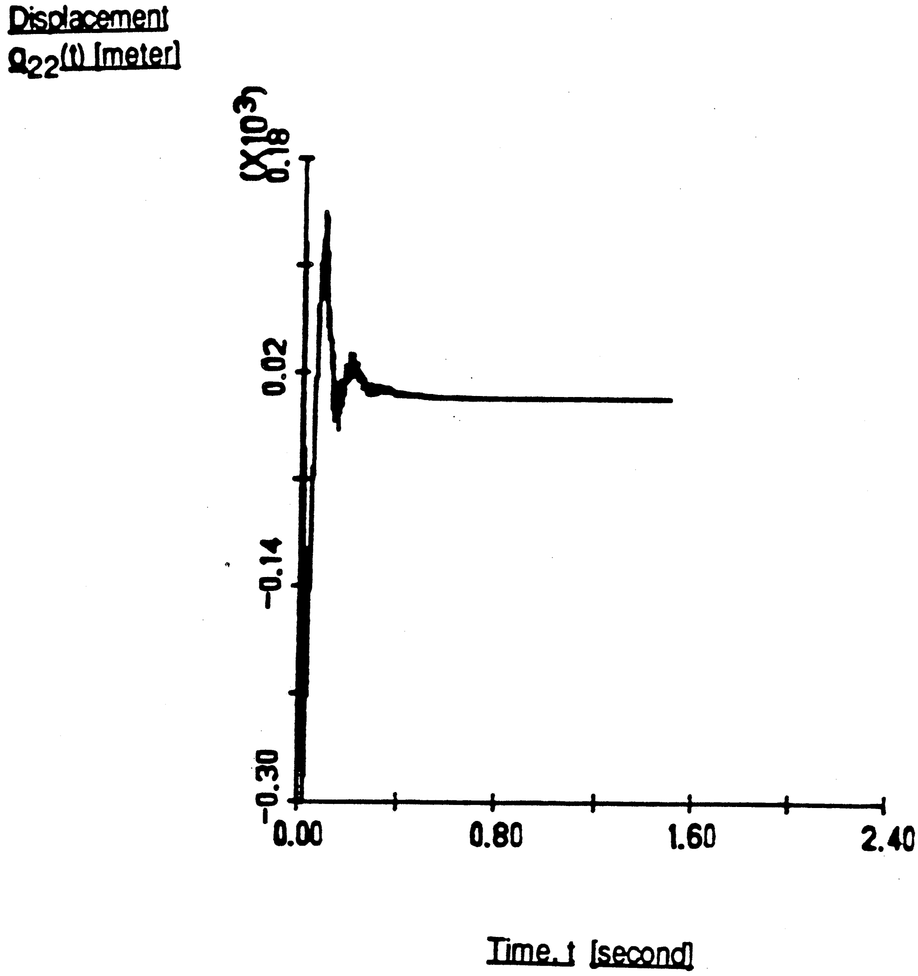


Figure 5-21. Flexible motion coordinate $q_{22}(t)$ in response to the rigid and flexible motion controller in the control spillover case.

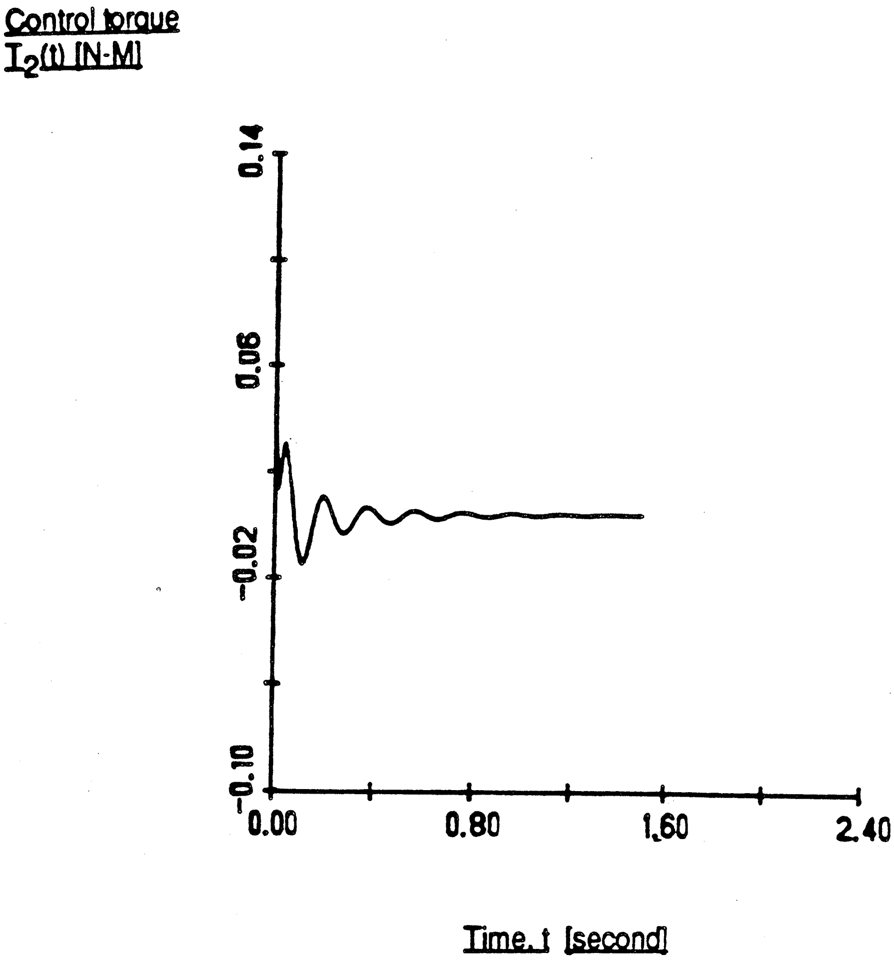


Figure 5-22. Control signal for the second joint obtained from the rigid and flexible motion controller in the control spillover case.

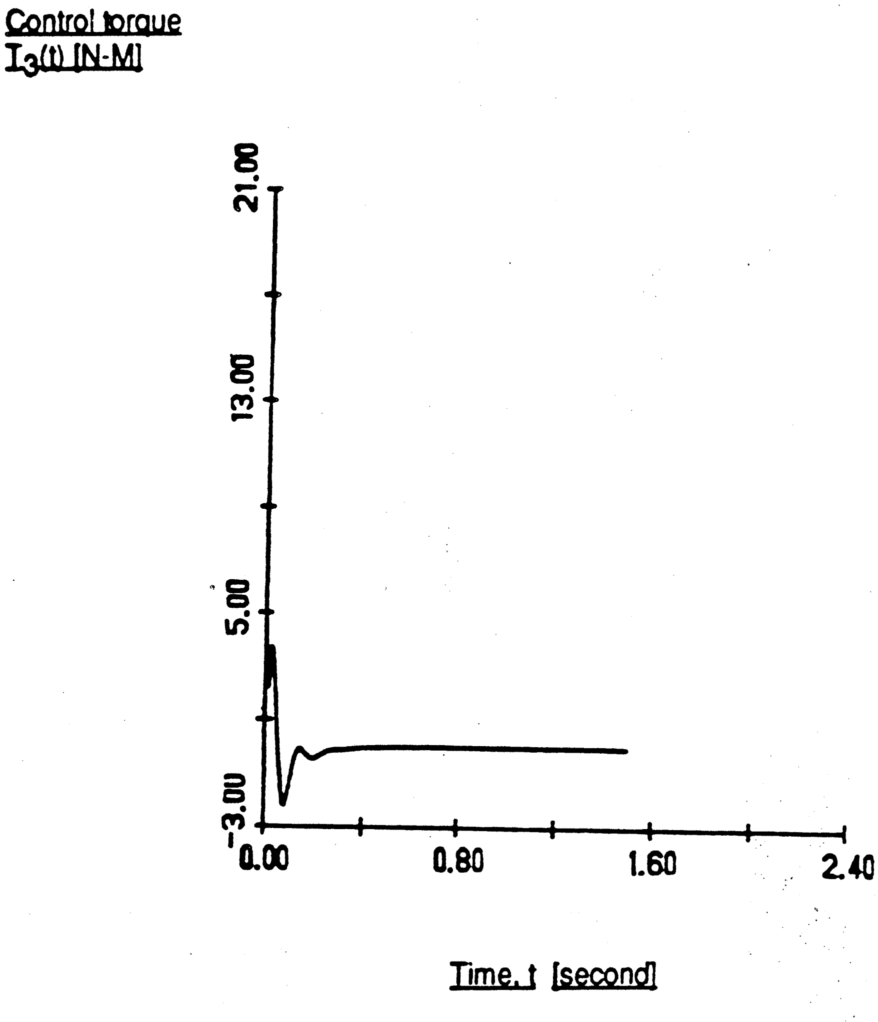


Figure 5-23. Control signal for the base joint obtained from the rigid and flexible motion controller in the control spillover case.

the transverse deflections V and W and their time derivatives $\dot{V}$ and $\dot{W}$. Assuming that the flexible motion is entirely due to the first mode, then V and W can be written as

$$V = \Phi_1 q_{11}(t) \qquad W = \Phi_1 q_{21}(t) \tag{5-36}$$

The flexible motion coordinates $q_{11}(t)$ and $q_{21}(t)$ become,

$$q_{11}(t) = \frac{V}{\Phi_1} \quad \text{and} \quad q_{21}(t) = \frac{W}{\Phi_1} \tag{5-37}$$

where Φ_1 is the eigenfunction of a clamped free beam, previously defined in equation (2-5), evaluated at the end effector. In digital simulation, the observation spillover is introduced as follows,

$$q_{11}(t) = \frac{\Phi_1(r)q_{11}(t) + \Phi_2(r)q_{12}(t)}{\Phi_1(r)} = \frac{V(r)}{\Phi_1(r)} \tag{5-38}$$

$$q_{21}(t) = \frac{\Phi_1(r)q_{21}(t) + \Phi_2(r)q_{22}(t)}{\Phi_1(r)} = \frac{W(r)}{\Phi_1(r)} \tag{5-39}$$

Thus the $q_{11}(t)$ and $q_{21}(t)$ measurements, that are fed back to the controller, are no longer representatives of the first mode only. Instead, they contain informations about both the first and second elastic modes. Figures 5-24 to 5-27 present the main results of this run. In Fig. 5-24 and 5-26, the responses of the first mode in the vertical and horizontal directions are almost identical to those in the control spillover case. However, the uncontrolled second mode exhibits more oscillatory behavior in the horizontal direction while it becomes unstable in the vertical direction. This instability is obtained due to the combined effect of control and observation spillover on the second mode.

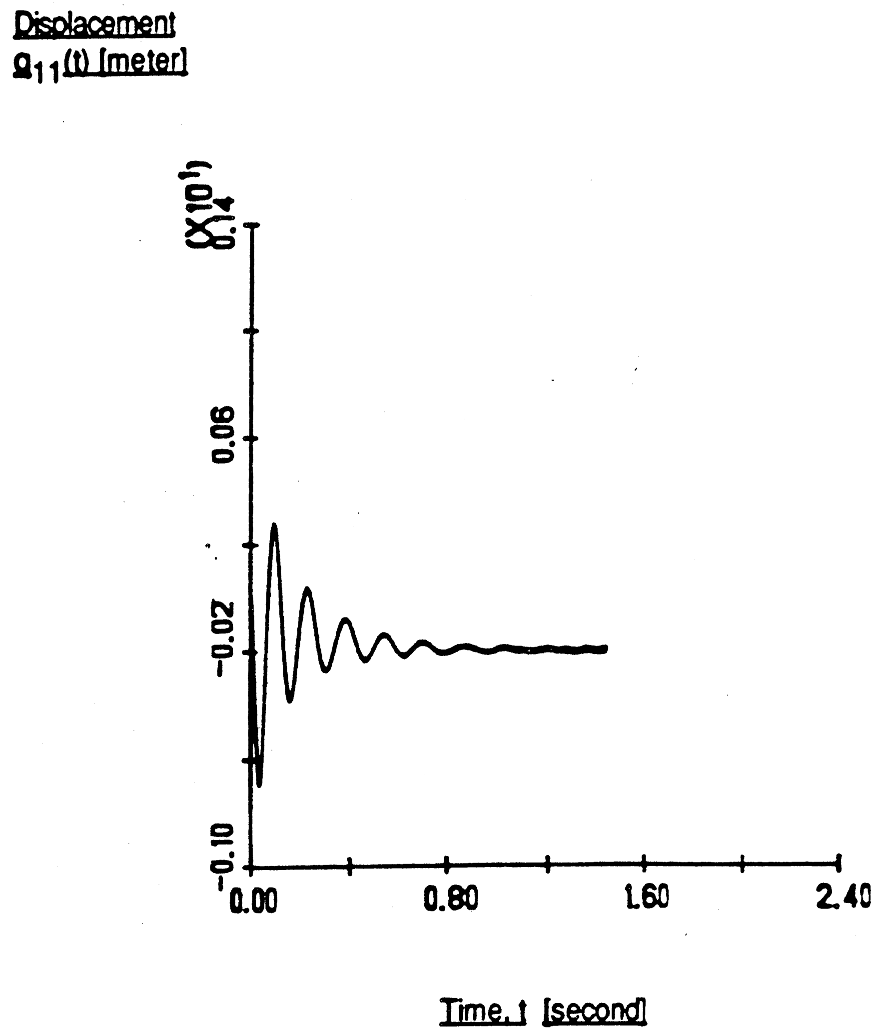


Figure 5-24. Flexible motion coordinate $q_{11}(t)$ in response to the rigid and flexible motion controller in the control and observation spillover.

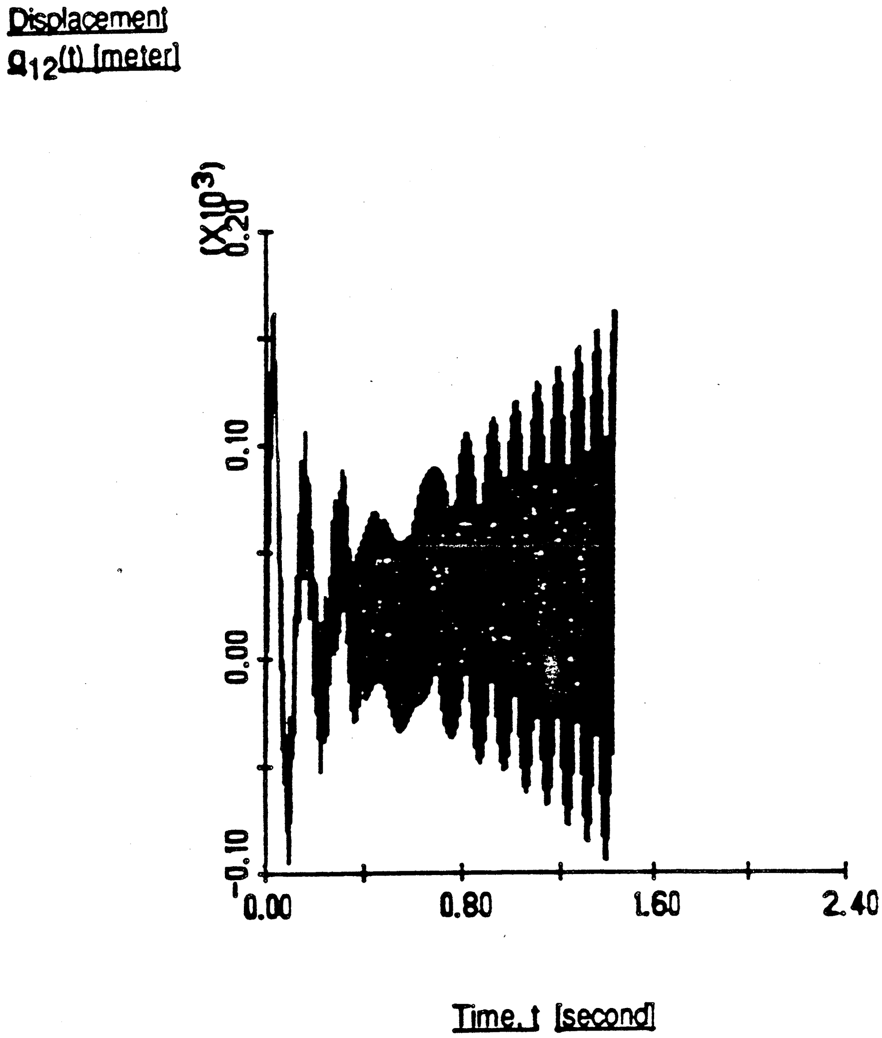


Figure 5-25. Flexible motion coordinate $q_{12}(t)$ in response to the rigid and flexible motion controller in the control and observation spillover.

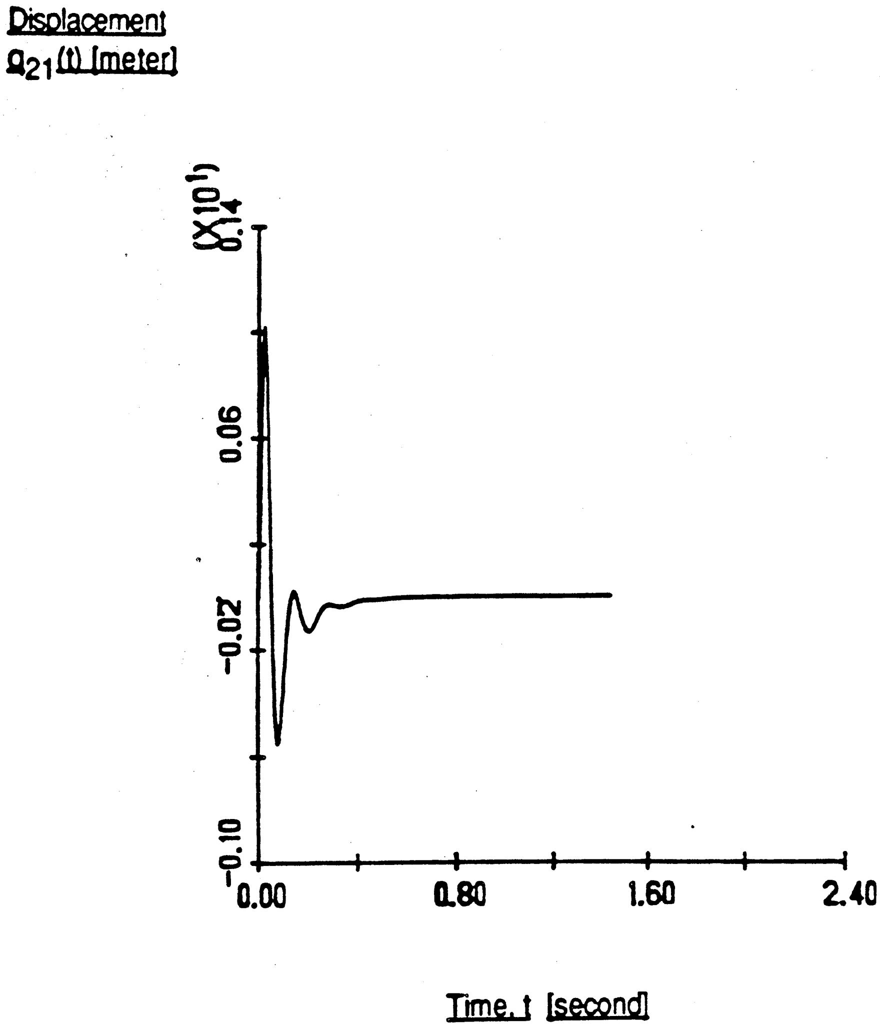


Figure 5-26. Flexible motion coordinate $q_{21}(t)$ in response to the rigid and flexible motion controller in the control and observation spillover.

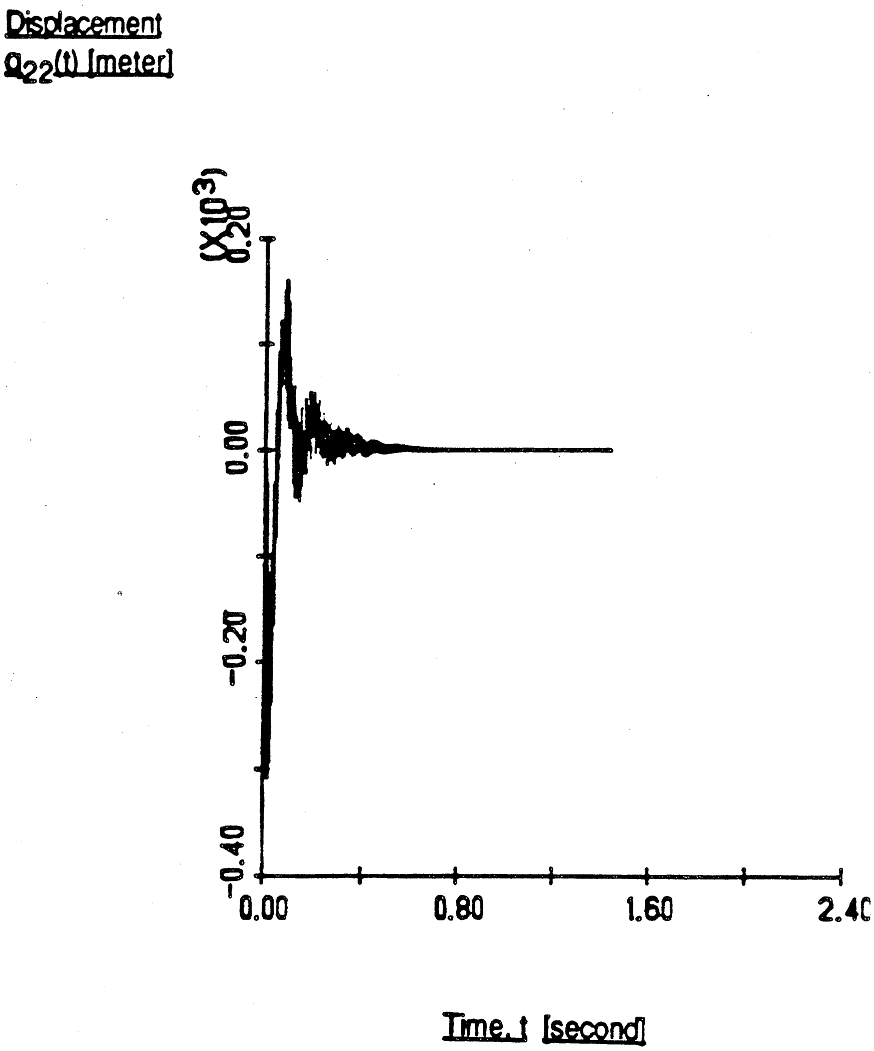


Figure 5-27. Flexible motion coordinate $q_{22}(t)$ in response to the rigid and flexible motion controller in the control and observation spillover.

Balas discusses the stability issue under these conditions in [30].

(3) *Control and Observation spillover in the presence of structural* damping. Any physical system possesses a certain amount of structural damping. The latter is introduced into the dynamic model of the flexible robot arm by using Rayleigh's dissipation function, F, which can be written in the following general form,

$$F = \frac{1}{2} \sum_{r=1}^{n} \sum_{s=1}^{n} c_{rs}\, \dot{q}_r\, \dot{q}_s \tag{5-40}$$

where c_{rs} is the damping coefficient. Considering only diagonal terms (i.e. $C_{rs} = 0$ for $r \neq s$) equation (5-40) becomes,

$$F = \frac{1}{2} \left(c_1 \dot{q}_{11}^2 + c_2 \dot{q}_{12}^2 + c_3 \dot{q}_{21}^2 + c_4 \dot{q}_{22}^2 \right) \tag{5-41}$$

The values of C_{ss}'s are determined experimentally and found to be 0.103 kg rad/sec which corresponds to a damping ratio, $\xi = 0.0145$. Viscous damping forces can be derived from Rayleigh's dissipation function, F, using the following formulation,

$$Q_i = - \frac{\partial F}{\partial \dot{x}_i} \tag{5-42}$$

where x_i is the i^{th} generalized coordinate. Dividing the nonconservative forces Q_{inc} in equation (2-15) into those dissipative type and those of impressed upon the system by external forces, Lagrange's equation become

$$\frac{d}{dt}\left(\frac{\partial T_t}{\partial \dot{x}_i} \right) - \frac{\partial T_t}{\partial x_i} + \frac{\partial V_t}{\partial x_i} + \frac{\partial F}{\partial \dot{x}_i} = Q_{incd} \tag{5-43}$$

where Q_{incD} denotes generalized forces due only to external forces exerted on the system. The rigid and flexible motion controller is then applied to the modified robot arm model described in equation (5-43). The results are shown in Fig. 5-28 to 5-31. These plots are identical to their counterparts in the undamped model with control and observation spillover case except for the response of the second mode in the vertical deflection, $q_{12}(t)$, which becomes stable.

5.7. Summary

This chapter provides the design of the rigid and flexible motion controller for the robot arm. First the control of a three lumped mass system, which represents a discrete model for a single compliant link, is developed. The rationale is to start with the control of relatively simple system and progress gradually towards the general nonlinear problem.

Two types of controllers are employed in the lumped model study. The first, simulating the rigid body controller, is based entirely on the motion of the first mass. In the design of the second controller, the dynamics of both the first and second masses are considered. The third mass is deliberately left out to study the effect of control spillover. The main results obtained, can be stated as follows,

(1) The rigid body controller provides some damping of the flexible motion.

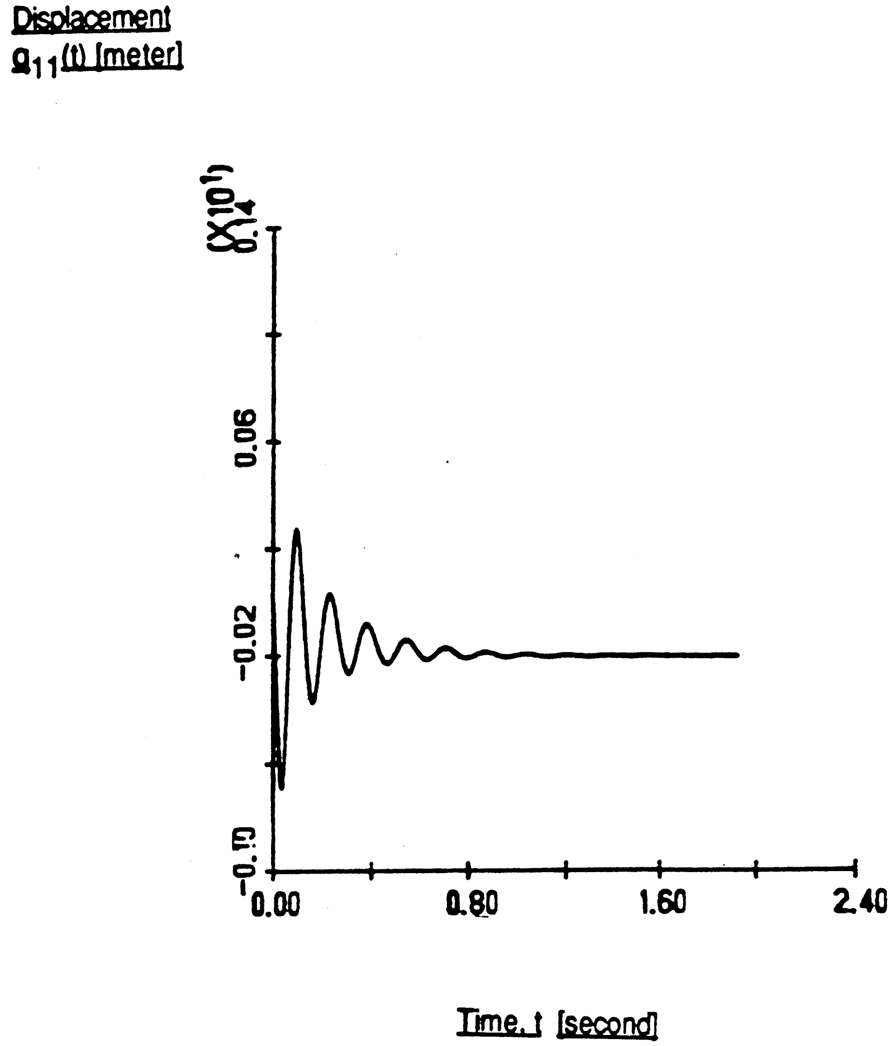


Figure 5-28. Flexible motion coordinate $q_{11}(t)$ in response to the rigid and flexible motion controller in the control and observation spillover with structural damping included.

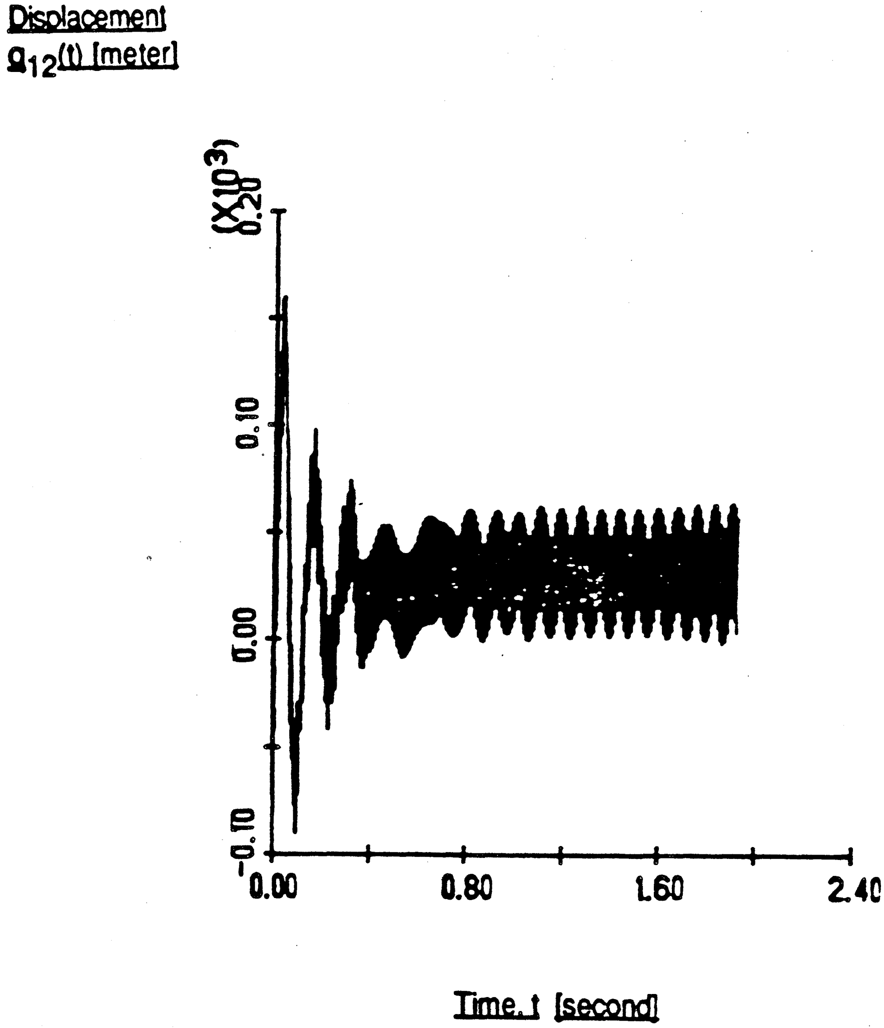


Figure 5-29. Flexible motion coordinate $q_{12}(t)$ in response to the rigid and flexible motion controller in the control and observation spillover with structural damping included.

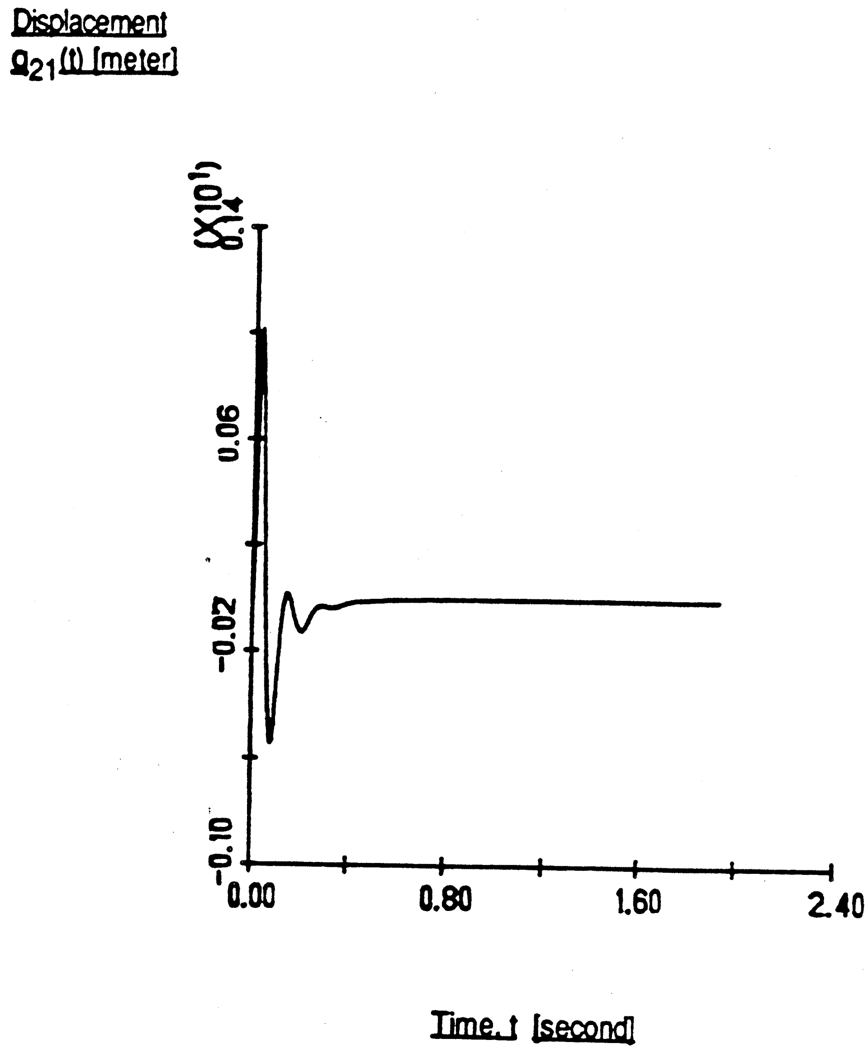


Figure 5-30. Flexible motion coordinate $q_{21}(t)$ in response to the rigid and flexible motion controller in the control and observation spillover with structural damping included.

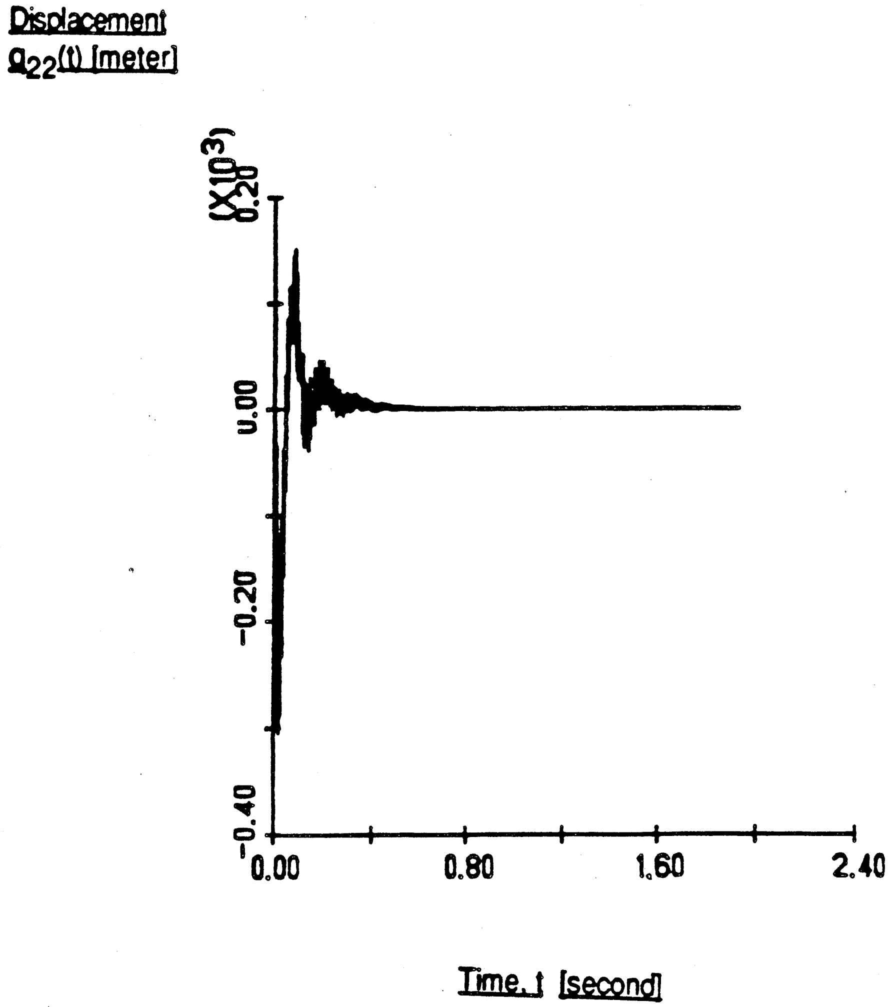


Figure 5-31. Flexible motion coordinate $q_{22}(t)$ in response to the rigid and flexible motion controller in the control and observation spillover with structural damping included.

(2) In the absence of observation and control spillover, the rigid and flexible motion controller completely eliminates the vibratory motion.

(3) In the presence of control spillover, the rigid and flexible motion controller reduces the oscillation observed in the rigid body controller case by almost half.

The rigid and flexible motion controller is designed based on a linearized version of the general equations of motion. Only the first mode of the transverse deflection in both the vertical and horizontal directions are considered in the control action. Higher modes are left out to study the effect of control and observation spillover. The controller objective is to add more damping to the flexible motion while maintaining a good rigid body motion performance. The following conclusions can be drawn:

(1) In the ideal case where there is no observation or control spillover, the flexible motion recovers rapidly from the sudden excitation induced by the rigid body motion inertial forces.

(2) In the presence of control spillover, the overall performance of the flexible motion show a reduction of approximately 50% in magnitude when compared with their counterparts in the base run of the rigid body controller.

(3) In the presence of observation and control spillover, identical results are obtained for the flexible motion as in the control spillover case except for the response of the uncontrolled second mode in the verti-

cal direction which becomes unstable.

(4) In the presence of structural damping with observation and control spillover; the response of the uncontrolled second elastic mode in the vertical direction becomes stable.

A comment is in order here. One should note that even in the presence of observation and control spillover, the rigid and flexible motion controller provides an additional damping capable of reducing by approximately 50% the magnitude of the flexible motion oscillation detected in the rigid body controller.

There is a need for experimental validation of the dynamic model of the robot arm, as well as evaluation of the rigid body controller versus rigid and flexible motion controller. This will be the subject of the next chapter.

CHAPTER 6

EXPERIMENTAL RESULTS AND COMPARISON TO SIMULATION RESULTS

The flexible robot arm control problem has been investigated using analytical methods and digital simulation. There is a need for experimental evaluation of both the dynamic model of the robot arm and the controller design. This chapter experimentally compares the rigid and flexible motion controller with the rigid body controller to evaluate the merit of measuring and feeding back the flexible motion.

6.1. Experimental Set Up

The three degree of freedom spherical coordinate laboratory robot, described in section 2.1, is used for the experimental work. It is interfaced to an IBM/PC microcomputer. A Tecmar Lab Tender interface board which has built-in 8 bit analog to digital and digital to analog converters is used. The schematic of the experimental set up is given in Fig. 6-1. Three DC servo motors, driven by power amplifiers, are used as actuators. The control torques are transmitted to the links by leadscrews for the r and θ axes and a gear train for the ϕ axis. One

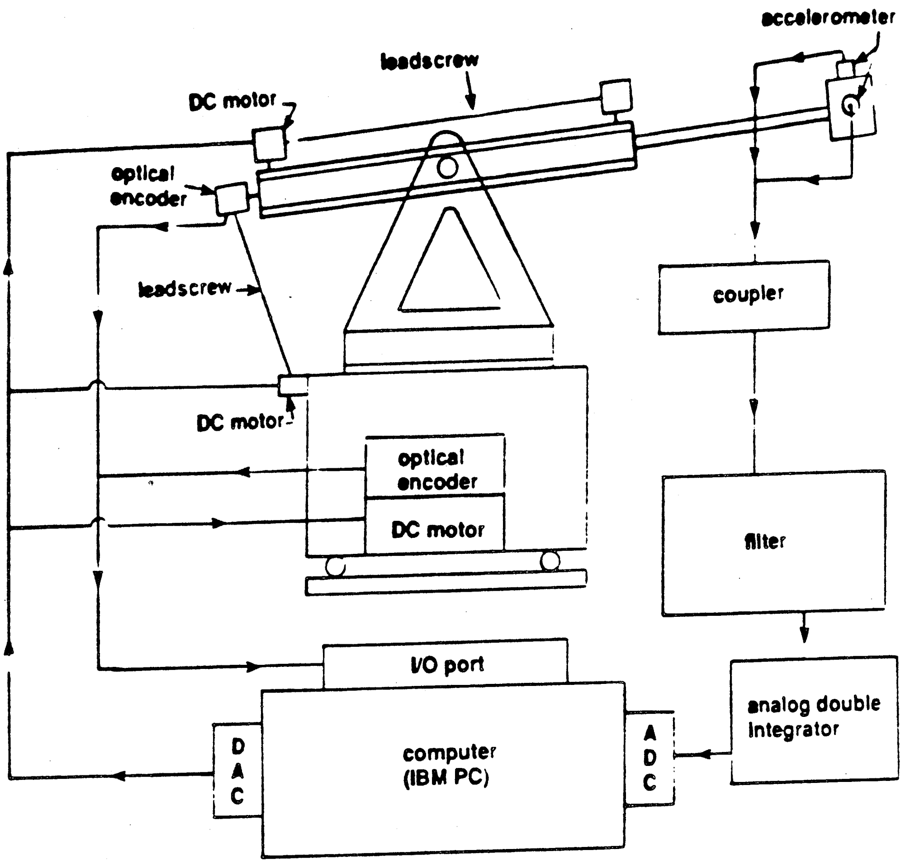


Figure 6-1. A schematic of the experimental set up.

tachometer generator and one optical encoder [58], [59] is mounted on each joint to measure the position and velocity of its corresponding rigid body degree of freedom. The optical encoder signals are read by the computer through digital counters. Each joint has a different size of counter associated with it according to their range of operation. A 20 bit counter is used to record the number of counts generated by the optical encoder monitoring the ϕ rotation whereas 12 bit and 20 bit counters are used to record the θ and r motions respectively. The contents of any counter can be strobed into a tri-state buffer. The latter can be accessed by the computer through differential drivers. Differential receivers are also used to reduce the noise on the control lines originating with the microcomputer.

The part of the second beam protruding from the first beam, which is considered to be flexible in the digital simulation, is made of a thin aluminum rod. The payload is kept constant. It consists of a mounting stud and two Kistler piezotron accelerometers (model 8606A100). The latter are used to measure the end effector transverse vibration in both the vertical and the horizontal directions. The stability conditions for such systems where the sensors are not colocated with the actuators are discussed in [60]. The acceleration signals are passed through a Kistler piezotron coupler (model 5120). The latter has a built in Kistler low pass filter (model 5318A22) with a break frequency set to 60 HZ. This attenuates the higher mode signals, thus reducing the effect of observation spillover. The dc component of the coupler's output which represents the end effector acceleration due to the rigid body motion needs to be filtered out. This is done by passing the acceleration signals through two cascaded high pass Krohn-Hite filters (model 3322) with a break frequency set to 3 HZ. Finally, the total deflection of the end effector and its time rate are obtained by integrating twice the accelerometer signals using analog double integrators [61], [62]. These integrators are used to devote most of the sampling period to the computation of the control action. The schematic drawing of the circuit board is given in Appendix E. The flexible motion signals are digitized by the 8 bit analog to digital converter whose resolutions are 0.2176 mm/count for the total deflection, V, and 16.314 mm/sec/count for $\dot{V}$.

The prismatic joint poses a very challenging problem. As the second beam slides inside or outside the first beam, the length of the flexible part varies. Thus, causing the beam natural frequencies to change significantly. This problem is not considered in this study. Therefore, the length of the part of the second beam protruding from the first beam, r, is kept constant. Due to hardware difficulties, the base joint governing the ϕ rotation was not operational, thus restricting the experimental work to the second joint only. That is only the rigid body coordinate θ and the flexible motion in the vertical direction, V, can be controlled. The horizontal deflection, W, would also be affected by the controller due to its coupling with V. The θ rotation is measured by an optical encoder with 625 counts per revolution (Optisyn model 77-4-003-625AA). The latter is enhanced by connecting the encoder shaft to a gear train with gear ratio of 8:1 to yield a resolution of 31.25 counts per degree of θ.

In designing the second beam, the following constraints are taken into consideration:

(1) The capability of the small dc servo motor mounted on the second joint.

(2) The computation time required for the control algorithm.

(3) The speed of the microcomputer used.

To satisfy the first constraint, the second beam should be lightweight. The control algorithm or the speed of the microcomputer impose a very stringent constraint on the design of the second beam. In the experimental work, the flexible motion is considered to be dominated by the first elastic mode. In order to avoid aliasing [63], the sampling theorem must be satisfied. That is the sampling period must be at most half the harmonic motion period. Thus, the fundamental frequency, which is fixed once a certain design for the second beam is adopted, would set the maximum sampling period. However, for high fundamental frequency, the period, which is inversely proportional to the natural frequency, becomes very small. Depending on the speed of the microcomputer or the control algorithm used, the upper bound of the sampling period imposed by the design of the second beam, may or may not be large enough to perform the control loop. Therefore, in this work, the second beam is designed to have a low fundamental frequency. This would give ample time to try different control algorithms. For the experimental set up employed here, the length of the second beam is chosen to be 0.5m and its diameter is 0.635cm so that its fundamental frequency would be approximately 6 HZ. The control algorithm used is an integral plus state feedback controller whose control loop requires a small amount of computation. The sampling period used is 0.024 sec. and is well below the upper bound imposed by the sampling theorem.

6.2. Experimental Evaluation of Model Parameters

In contrast to the digital simulation, the dynamics of the actuators and sensors are considered in the experimental work. The robot arm is treated as a pure inertia loading on the motor shaft. The armature controlled DC motor used in the experimental set up is very common in servo control systems. The derivation of its dynamic model is well documented in the literature [49]. The corresponding transfer function can be written as follows,

$$\frac{\dot{\theta}}{u} = \frac{K}{\tau s + 1} \tag{6-1}$$

where u is the input voltage to the motor, τ is the time constant, and K is the overall gain of the motor, the power amplifier and the tachometer generator. The parameters K and τ are determined experimentally from open loop runs. For a constant step input $u(t) = A$ volt and initial condition $\dot{\theta}\,(t = 0) = 0$, the response of the system described by equation (6-1) can be given by

$$\theta(t) = KA(1 - e^{-t/\tau}) \tag{6-2}$$

At steady state, θ_{ss} is equal to KA. The gain, K, can now be easily determined since θ_{ss} corresponds to the steady state value of the open loop response of the system to a step input. The time constant, τ, corresponds to the time at which the open loop response reaches $KA(1 - e^{-1})$. Several open loop runs are made, where different step inputs are given to the system and the sensors values are stored in the computer. The average values for K and τ are found to be 2.284 degree/sec/volt and 0.15 sec respectively.

6.3. Design and Performance of the Rigid Body Controller

The model described by equation (6-1) is first expressed in terms of state variables which are defined as follows:

$$\begin{aligned} y_1 &= \theta \\ y_2 &= \dot{\theta} \\ y_3 &= \int(\theta - R_2)dt \end{aligned} \tag{6-3}$$

where R_2 is the desired position of θ. The state variable y_3 is introduced to imbed the error signal into the state equations. The latter would have the following form,

$$\begin{Bmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \end{Bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\frac{1}{\tau} & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{Bmatrix} y_1 \\ y_2 \\ y_3 \end{Bmatrix} + \begin{bmatrix} 0 \\ \frac{K}{\tau} \\ 0 \end{bmatrix} u + \begin{Bmatrix} 0 \\ 0 \\ -R_2 \end{Bmatrix} \tag{6-4}$$

In matrix form, equation (6-4) can be expressed as

$$\dot{\underset{\sim}{y}} = A\underset{\sim}{y} + Bu + \underset{\sim}{R} \tag{6-5}$$

Using the procedure outlined in [64], the state equations are converted to difference equations which can be written as,

$$\underset{\sim}{y}(t_{k+1}) = P\underset{\sim}{y}(t_k) + Qu(t_k)$$
$$t_k = k.T \tag{6-6}$$

where T is the sampling period. P and Q are obtained from

$$P = e^{AT} = I + A\left\{IT + \frac{AT^2}{2!} + \frac{A^2T^3}{3!} + ... + \frac{A^i\,T^{i+1}}{(i+1)!} + ...\right\}$$
$$Q = \int_0^T e^{A\xi}d\xi B = \left\{IT + \frac{AT^2}{2!} + \frac{A^2T^3}{3!} + ... + \frac{A^i\,T^{i+1}}{(i+1)!} + ...\right\}B \tag{6-7}$$

Two different values of $i = 100$ and $i = 300$, are considered in the series. No significant differences in the results are observed, thus, for $i = 100$ the series has converged.

Next the design of the rigid body controller is considered. An integral plus state feedback controller is used. The control signal $u(t_k)$ becomes,

$$u(t_k) = -\begin{bmatrix} k_1 & k_2 & k_3 \end{bmatrix}\begin{Bmatrix} y_1(t_k) \\ y_2(t_k) \\ y_3(t_k) \end{Bmatrix} = -K^T\underset{\sim}{y}(t_k) \tag{6-8}$$

where K^T is the transpose of the controller gain matrix. The closed loop difference equations are obtained by substituting (6-8) into (6-6)

$$\underset{\sim}{y}(t_{k+1}) = (P - QK^T)\underset{\sim}{y}(t_k) + R \tag{6-9}$$

The gains are selected to yield a damping ratio $\xi = 1$ and desired servo loop frequency ω_f. The controller software used in the experimental studies is listed in Appendix G.

The rigid body controller is applied on the robot arm to rotate it from its initial position $\theta(0) = 0$, which corresponds to the horizontal position, to a certain desired position, R_2. Several runs are made. It is found that for R_2 greater that 20 degrees the effect of nonlinearity becomes significant and the response is oscillatory. Typical results of the rigid body controller are shown in Fig. 6-2 to 6-4 for $R_2 = -20$ degrees. The first plot shows the θ response. It has a small overshoot which persists for a while due to the effect of friction. Figures 6-3 and 6-4 represent the control voltage applied to the motor and the total vertical deflection of the end effector. The maximum deflection observed is approximately $\pm$ 3.5mm. The control signal shows some saturation. This is desirable since it serves the purpose of driving the robot arm at its highest speed. The pulse like pattern observed in Fig. 6-3 and 6-4 illustrate the poor resolution obtained from the 8 bit analog to digital and digital to analog converters.

An additional run is performed to study the effect of friction. This is done by driving the robot arm beyond its linear range and the response becomes oscillatory with a large amount of overshoot. This is illustrated in Fig. 6-5. At the peak of the oscillation, $\dot{\theta}$ is zero. This results in a sudden increase in the required driving torque due to the change from dynamic to static coefficient of friction. The control signal, illustrated in Fig. 6-6, is unable to respond quickly to the sharp variation in the resistive torque would cause the robot arm to come to a complete halt waiting for the control effort to build up. This is consistent with what is observed in the simulation result in Fig. 3-3.

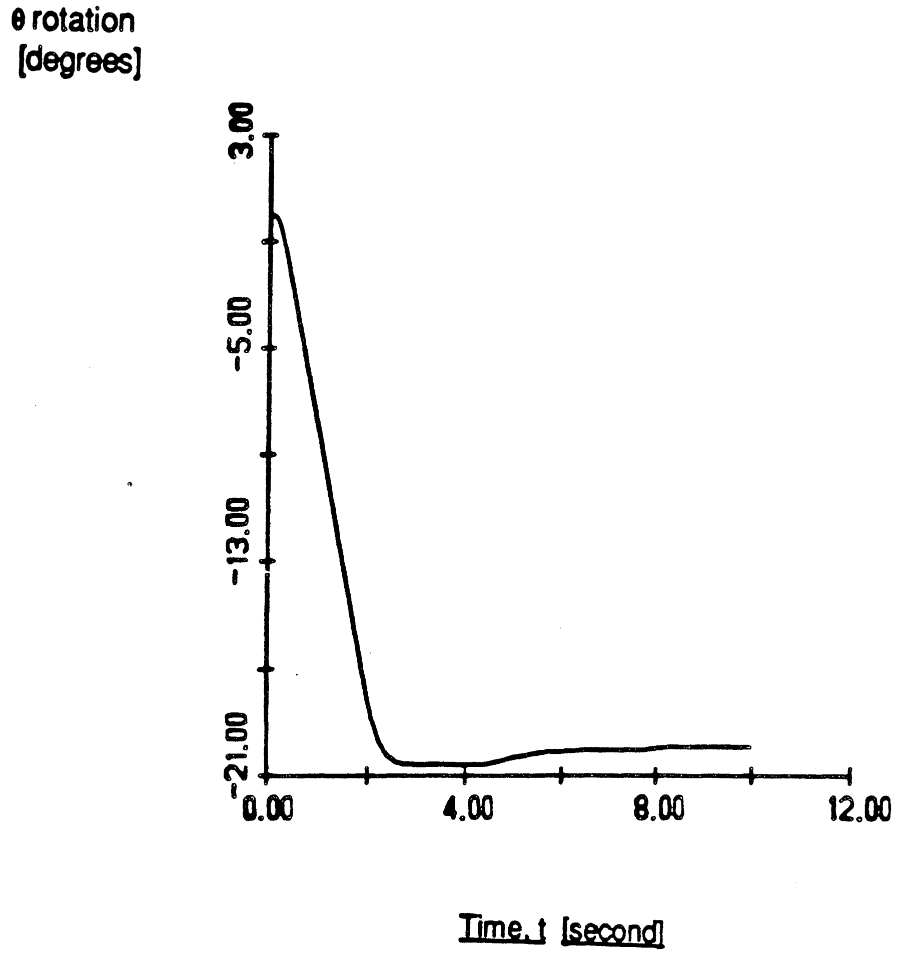


Figure 6-2. θ response obtained from the rigid body controller in the experimental work.

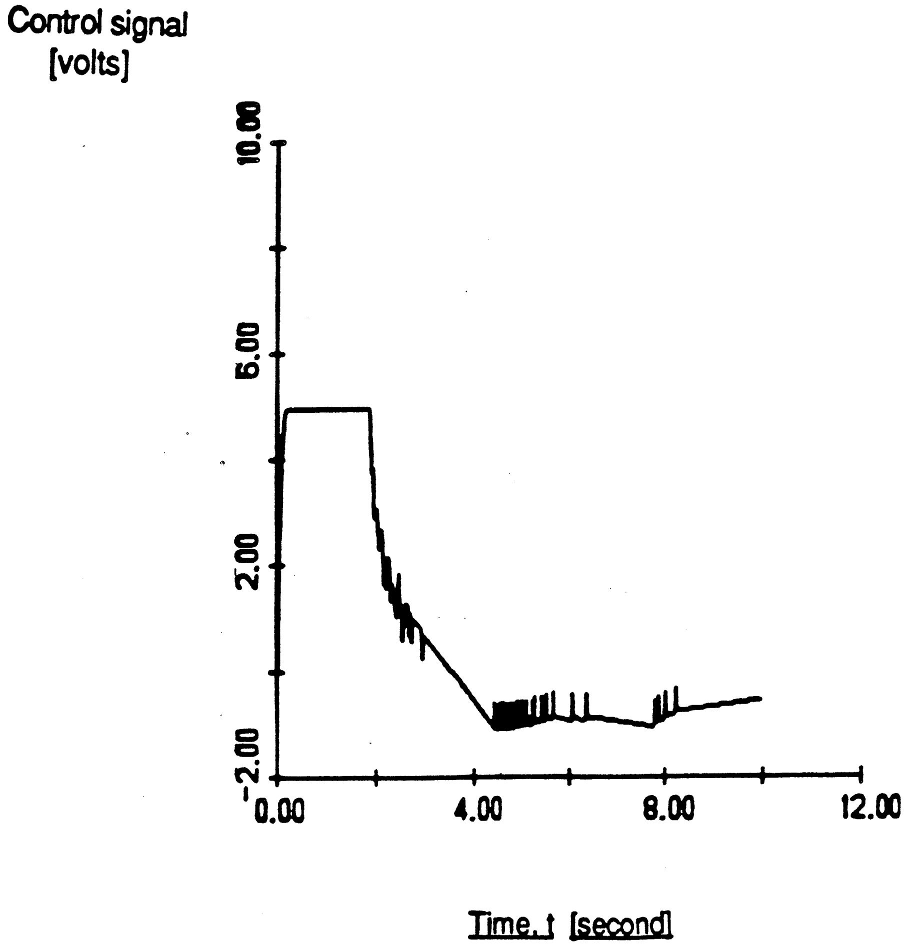


Figure 6-3. Control signal for the second joint obtained from the rigid body controller in the experimental work.

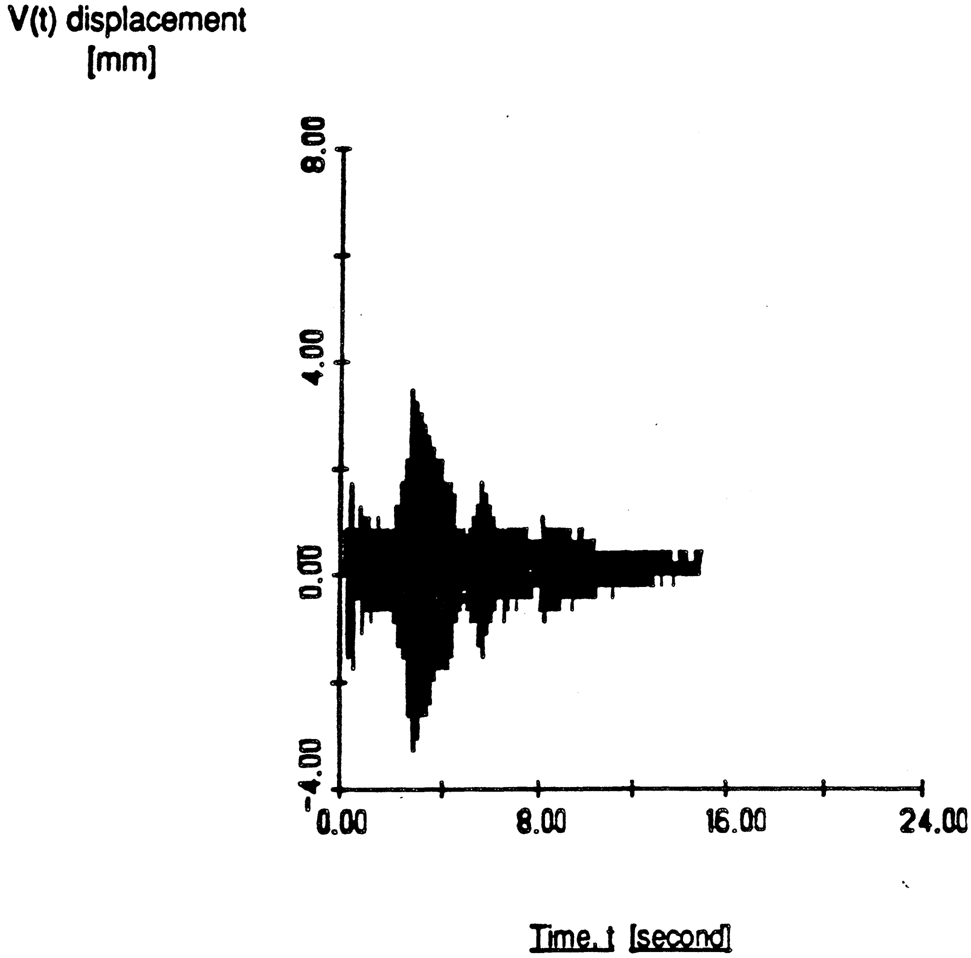


Figure 6-4. **Total vertical deflection in response to the rigid body controller in the experimental work.**

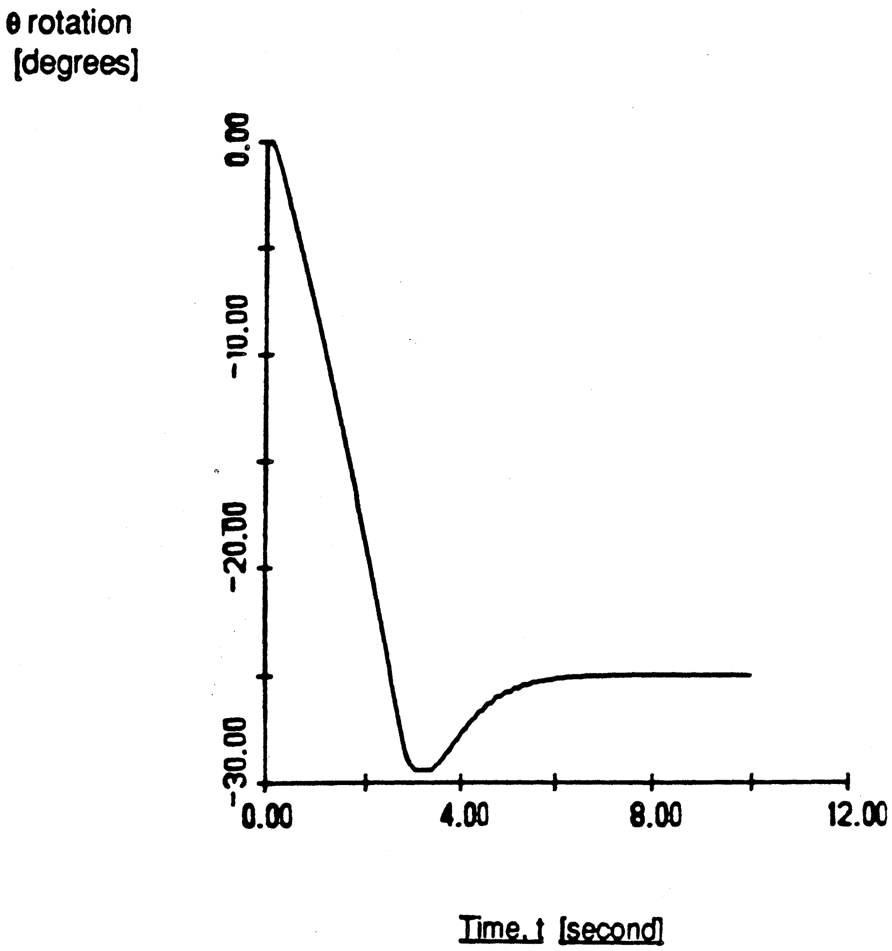


Figure 6-5. θ response showing the effect of friction in the experimental work.

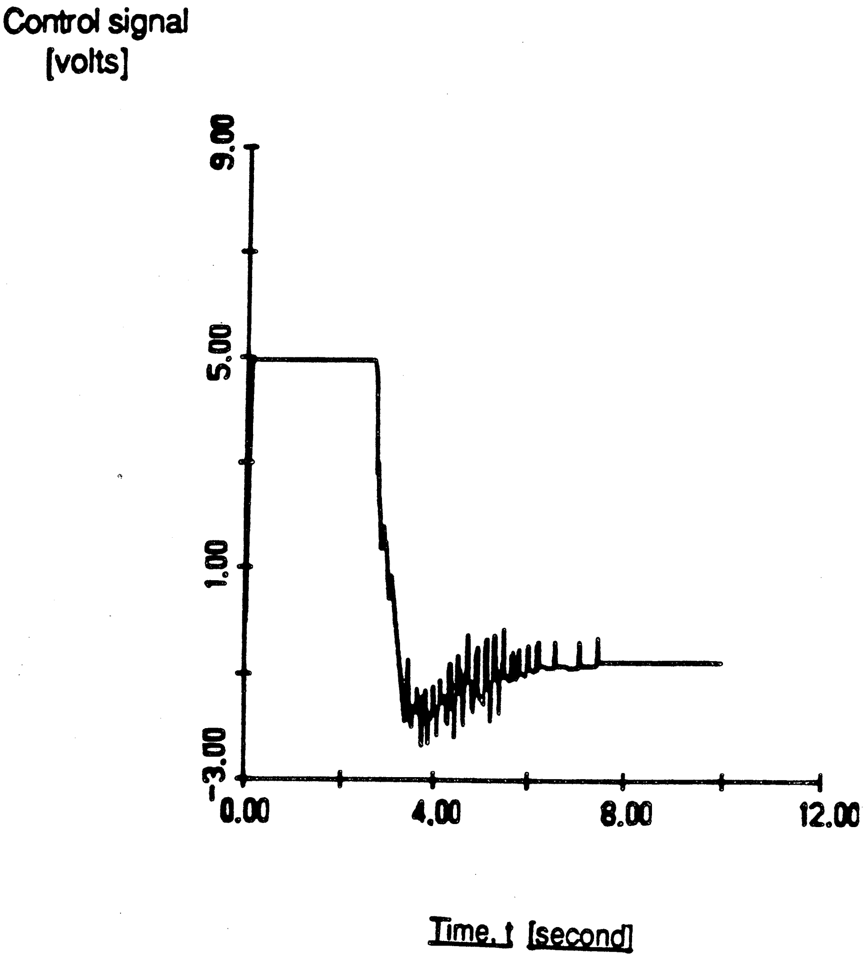


Figure 6-6. Control signal for the second joint showing the effect of friction in the experimental work.

6.4. Design and Results of the Rigid and Flexible Motion Controller

In the experimental work, the observation and control spillover are always present. The control action affects all elastic modes whereas the accelerometer signals contain information from all modes. A low pass filter with a break frequency set to 60 HZ attenuates the effect of higher modes. However, it does not eliminate the observation spillover problem entirely. Fortunately, the physical system exhibits some light structural damping. As seen in Chapter 5, this provides a margin of stability for the uncontrolled modes.

The objective of the rigid and flexible motion controller is to introduce more damping into the flexible motion. This is done by following the same controller design approach used in Chapter 5. That is expanding the rigid body controller to include the feedback signal of the derivative of the flexible motion coordinate, $\dot{q}_{11}(t)$. The control signal becomes

$$u(t_k) = -\begin{bmatrix} k_1 \; k_2 \; k_3 \; k_4 \end{bmatrix} \begin{Bmatrix} y_1(t_k) \\ y_2(t_k) \\ y_3(t_k) \\ y_4(t_k) \end{Bmatrix} = -K^T \underset{\sim}{y} \tag{6-10}$$

where the $y_4(t_k)$ in the augmented state vector $\underset{\sim}{y}$ represents $\dot{q}_{11}(t)$. The latter is obtained by assuming that the flexible motion is dominated by the first elastic mode. Therefore, from the assumed modes method, the total vertical deflection, V, can now be written as

$$V = \Phi_1(y) q_{11}(t) \tag{6-11}$$

its time derivative $\dot{V}$, is

$$\dot{V} = \dot{\Phi}_1(y)q_{11}(t) + \Phi_1(y)\dot{q}_{11}(t) \tag{6-12}$$

In the experimental study, the length of the flexible beam is kept constant. This yields

$$\dot{\Phi}_1(y) = 0 \tag{6-13}$$

Thus leading to

$$\dot{V} = \Phi_1(y)\dot{q}_{11}(t) \tag{6-14}$$

Since the accelerometers are located at the end effector, then

$$\dot{q}_{11}(t) = \frac{\dot{V}}{\Phi_1(r)} \tag{6-15}$$

where $\Phi_1(r)$ is the value of the assumed mode shape evaluated at the end effector. $\dot{V}$ is the time rate of the total deflection. It is obtained by integrating the accelerometer signal once using the hardware integrator. Equation (6-15) provides a means to compute the newly introduced state variable $y_4 = \dot{q}_{11}(t)$ required for the rigid and flexible motion controller. Only the gain k_4 needs to be selected since the rest of the gains k_1,k_2 and k_3 retain the same values that they hold in the rigid body controller. The gain k_4 is selected by trial and error. A large value for k_4 would increase the effect of the flexible motion on the rigid body motion. This would have an adverse effect on the response of the rigid body coordinate θ. Therefore a compromise is reached where the gain k_4 is selected to be large enough to damp out the flexible motion within an acceptable tolerance while keeping the effect of the flexible motion on the rigid body motion to minimum. The rigid and flexible motion controller is applied to the robot arm and the

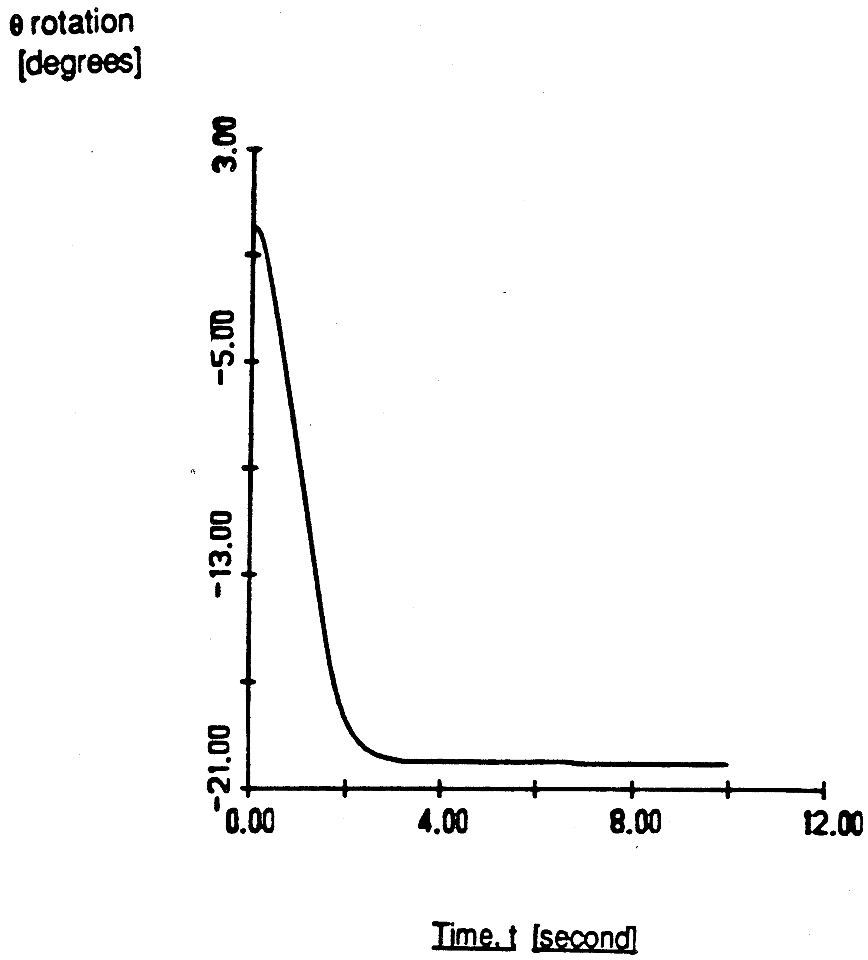


Figure 6-7. θ response obtained from the rigid and flexible motion controller in the experimental work.

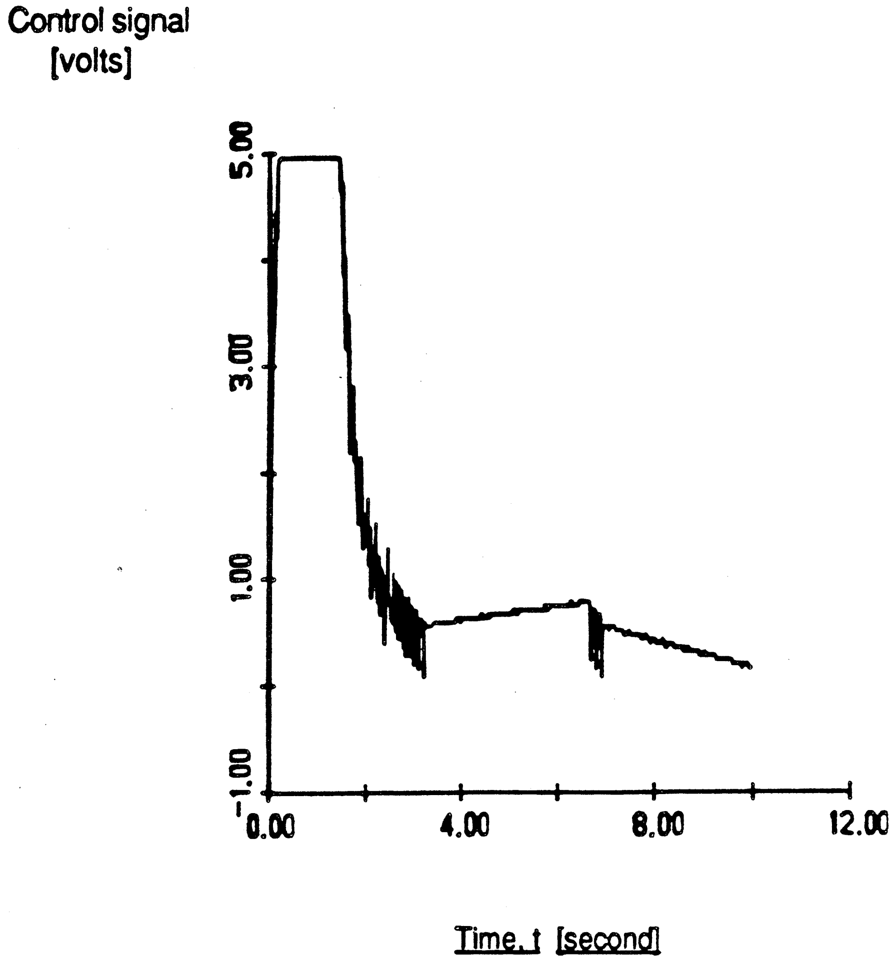


Figure 6-8. Control signal for the second joint obtained from the rigid and flexible motion controller in the experimental work.

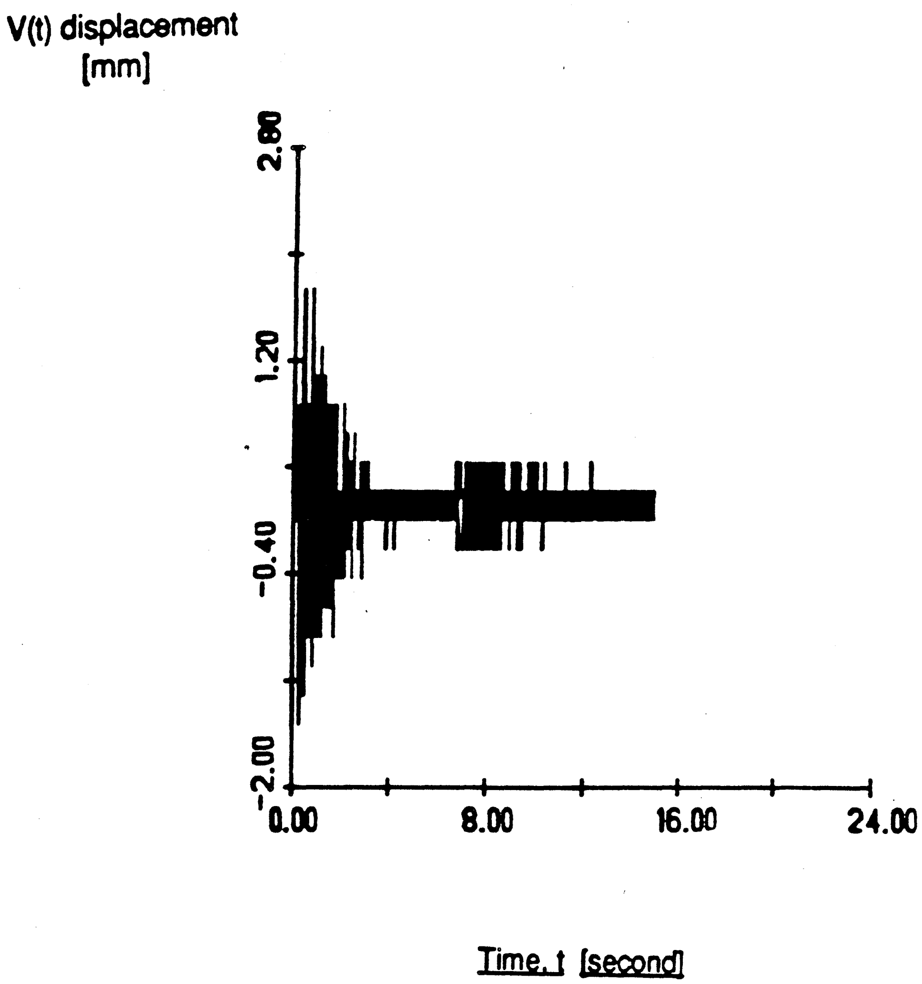


Figure 6-9. Total vertical deflection in response to the rigid and flexible motion controller in the experimental work.

results are shown in Fig. 6-7 to 6-9. The first plot represents the overdamped θ response. The applied control voltage, shown in Fig. 6-8, has two important features:

(1) The control signal fluctuates sharply during the period where θ approaches its desired value. In response to these oscillations, the DC motor introduces a series of pulses into the system. This results in reduction of the amplitude of the vertical deflection. Thus leading to additional damping in the flexible motion.

(2) A small position error exists after $t = 3$ sec. The control effort is unable to overcome the resistive torque, induced by the friction and geometry effects of the leadscrew, and brings the θ response to a standstill. The integral action, which tries to correct for the position error, starts building up the control action. This phenomenon is clearly observed in the control signal plot (Fig. 6-8).

Finally, Fig. 6-9 shows a reduction in the maximum deflection of the end effector of approximately 75% as compared to the rigid body controller results shown in Fig. 6-4.

6.5. Conclusions

In experiments, the rigid and flexible motion controller reduced the transverse deflection in the vertical direction by approximately 75% over the rigid body controller even in the presence of observation and control spillover.

The effect of friction, observed in Figs. 6-5 and 6-6, is very consistent with the results obtained in the digital simulation.

Note that in the digital simulation, an approximate reduction of 50% in the vertical deflection is obtained versus an approximate 75% reduction in the experimental work. This may in part be due to the fact that in the experimental work only the second joint is used while in the digital simulation all three joints are moved simultaneously.

6.6. Summary

This chapter provides a description of the experimental set up. A set of difference equations, which represents a simple linear model of the servo positioner, are obtained. Several open-loop runs are performed to measure the open loop characteristics of the motor, the power amplifier and the tachometer generator. Both a rigid body controller and rigid and flexible motion controller are implemented. A comparison between their performances is made. Also the effect of friction is experimentally demonstrated.

CHAPTER 7

SUMMARY AND CONCLUSIONS

This chapter provides a brief summary of the dissertation content. The main conclusions are outlined and some prospective research topics, from unsolved issues encountered in the present research, are stated.

7.1. Summary

An overview of the research done in robot arm modeling, dynamics, and control is presented. In this work, the dynamics of a spherical coordinate robot arm, whose last link is very flexible, is studied for the purpose of control. The assumed modes method is employed to represent the deflection of any point on the flexible link. All the coupling terms between the rigid and the flexible motions are retained. This is done by using coupled reference position and elastic deformation models. The kinetic and potential energy expressions are developed. Finally, Lagrange's method is implemented to obtain the unconstrained equations of motion.

The robot arm considered in this work has two joints driven by leadscrew transmission mechanisms. The kinematic constraints associated with a leadscrew are introduced to investigate the behavior of a leadscrew driven flexible robot arm in the presence of coulomb friction and the self locking condition (i.e. the leadscrew is nonbackdrivable).

An integral plus state feedback controller is derived based on a linearized version of the rigid body model of the robot arm. The controller is then implemented on the rigid and flexible model. The rationale is to simulate the controllers currently used in industrial robots and to assess the interrelationships between the robot arm structural flexibility and the controller design.

In an attempt to gain insight into the control problem of the flexible robot arm, a simple case, involving the controller design of a three lumped mass system is considered. The latter simulates the control problem of a compliant beam whose flexible motion is approximated by two elastic modes. This allows the study of the controller performance while working with a simple linear model.

The objective in designing the rigid and flexible motion controller is to introduce additional damping into the flexible motion. The integral plus state feedback controller, derived for the rigid body model of the robot arm, is extended to include the flexible motion. This is done by using additional sensors to measure the compliant link dynamic vibrations and feed them back to the controller. The effect of control and observation spillover is examined by considering one elastic mode in both the vertical and horizontal directions in the controller design. The second mode in each direction is considered to be representative of the higher unmodeled modes.

Finally, an experimental evaluation is performed on both the rigid and flexible motion controller and the rigid body controller to assess the merit of measuring and feeding back the flexible motion.

7.2. Major Contributions

The major contributions of this work are:

(1) The kinematic constraints associated with a leadscrew are developed and the behavior of a leadscrew driven flexible robot arm in the presence of coulomb friction and the self locking condition is investigated.

(2) The investigation of the interrelationships between the robot arm structural flexibility and the controller design has led to the following conclusions:

  (a) Drive characteristics influence coupling between flexible and rigid body motions.

  (b) Rigid body controllers can provide some damping into the flexible motion, and the reduction of arm vibration depends on drive characteristics.

  (c) Interactions between structural design and controller design should be considered.

(3) The inclusion of the flexible motion in the control action results in additional damping into the flexible motion and the important issue of observation and control spillover is found to present no serious practical problem.

(4) The dynamic model of the leadscrew driven flexible robot arm and controller design are evaluated experimentally and found to be in good agreement with the digital simulation results.

7.3. Prospective Research Topics

The following issues, which were encountered in this work, need to be addressed in future research:

1 - The problem of developing a general procedure to obtain a dynamic model of a prismatic joint in a flexible robot arm should be addressed.

2 - Using additional sensors to measure the flexible motion, the merits of introducing additional damping into the flexible motion by different types of controllers need to be examined.

3 - The issue of using additional sensors and actuators should be studied.

APPENDICES

APPENDIX A

THE UNCONSTRAINED EQUATIONS OF MOTION

The resulting seven nonlinear, coupled, second order ordinary differential equations of the unconstrained motion are listed in this appendix. The equation of motion for r is,

$$\dot r = \ddot r = 0 \quad \text{for} \quad F_c = 0$$

and

$$\begin{aligned}
&-\rho A_2\dot r^2 + \rho A_2(L_2 - r)\ddot r + \rho A_2\Bigl\{\ddot r r + \dot r^2 + (\dot r\dot\phi\cos\theta + r\ddot\phi\cos\theta - r\dot\phi\dot\theta\sin\theta)\Bigl[.78q_{21} + .43q_{22}\Bigr] \\
&+ r\dot\phi\cos\theta(.78\dot q_{21} + .43\dot q_{22}) - (\dot r\dot\theta + r\ddot\theta)\Bigl[.78q_{11} + .43q_{12}\Bigr] - r\dot\theta(.78\dot q_{11} + .43\dot q_{12}) \\
&+ 0.61(\ddot r q_{11}^2 + 2\dot r q_{11}\dot q_{11}) + 0.67(\ddot r q_{11}q_{12} + \dot r\dot q_{11}q_{12} + \dot r q_{11}\dot q_{12}) + 0.5(\dot r q_{11}\dot q_{11} + r\dot q_{11}^2 + rq_{11}\ddot q_{11}) \\
&+ 0.65(\dot r q_{11}\dot q_{12} + r\dot q_{11}\dot q_{12} + rq_{11}\ddot q_{12}) + 0.18(\ddot r q_{12}^2 + 2\dot r q_{12}\dot q_{12}) - 0.65(\dot r q_{12}\dot q_{11} + r\dot q_{12}\dot q_{11} + rq_{12}\ddot q_{11}) \\
&+ 0.5(\dot r q_{12}\dot q_{12} + r\dot q_{12}\dot q_{12} + r\ddot q_{12}q_{12}) + \ddot\theta L_1(.78q_{11} + .43q_{12}) + \dot\theta L_1(.78\dot q_{11} + .43\dot q_{12}) \\
&+ (\ddot\theta r + \dot\theta\dot r)(.35q_{11} - .25q_{12}) + \dot\theta r(.35\dot q_{11} - .25\dot q_{12}) \\
&- (\ddot\phi r\sin\theta + \dot\phi\dot r\sin\theta + \dot\phi r\dot\theta\cos\theta)(.5q_{11}q_{21} + .65q_{11}q_{22} - .65q_{12}q_{21} + .5q_{12}q_{22}) \\
&- \dot\phi r\sin\theta\Bigl[.5(\dot q_{11}q_{21} + q_{11}\dot q_{21}) + .65(\dot q_{11}q_{22} + q_{11}\dot q_{22}) - .65(\dot q_{12}q_{21} + q_{12}\dot q_{21}) \\
&+ .5(\dot q_{12}q_{22} + q_{12}\dot q_{22})\Bigr] + 0.61(\ddot r q_{21}^2 + 2\dot r q_{21}\dot q_{21}) + 0.67(\ddot r q_{21}q_{22} + \dot r\dot q_{21}q_{22} + \dot r q_{21}\dot q_{22}) \\
&+ 0.18(\ddot r q_{22}^2 + 2\dot r q_{22}\dot q_{22}) + (\ddot\phi r\sin\theta + \dot\phi\dot r\sin\theta + \dot\phi\dot\theta r\cos\theta)\Bigl[.5q_{21}q_{11} + .65q_{21}q_{12} \\
&- .65q_{22}q_{11} + .5q_{22}q_{12}\Bigr] + \dot\phi r\sin\theta\Bigl[0.5(\dot q_{21}q_{11} + q_{21}\dot q_{11}) + 0.65(\dot q_{21}q_{12} + q_{21}\dot q_{12}) \\
&- 0.65(\dot q_{22}q_{11} + q_{22}\dot q_{11}) + 0.5(\dot q_{22}q_{12} + q_{22}\dot q_{12})\Bigr] - L_1(\ddot\phi\cos\theta - \dot\phi\dot\theta\sin\theta)\Bigl[.78q_{21} + .43q_{22}\Bigr] \\
&- L_1\dot\phi\cos\theta(.78\dot q_{21} + .43\dot q_{22}) - (\ddot\phi\cos\theta - \dot\phi\dot\theta\sin\theta)\Bigl[.35rq_{21} - .25rq_{22}\Bigr]
\end{aligned}$$

$$- \dot{\phi}\cos\theta\Big[.35(\dot{r}q_{21} + r\dot{q}_{21}) - .25(\dot{r}q_{22} + r\dot{q}_{22})\Big] + 0.5(\dot{r}q_{21}\dot{q}_{21} + r\dot{q}_{21}^{2} + rq_{21}\ddot{q}_{21})$$

$$+ 0.65(\dot{r}q_{21}\dot{q}_{22} + r\dot{q}_{21}\dot{q}_{22} + rq_{21}\ddot{q}_{22}) - 0.65(\dot{r}q_{22}\dot{q}_{21} + r\dot{q}_{22}\dot{q}_{21} + rq_{22}\ddot{q}_{21})$$

$$+ 0.5(\dot{r}q_{22}\dot{q}_{22} + r\dot{q}_{22}^{2} + rq_{22}\ddot{q}_{22})\Big\} + m_p\Big\{\ddot{r} + 2(\ddot{\phi}\cos\theta - \dot{\phi}\dot{\theta}\sin\theta)(q_{21} - q_{22})$$

$$+ 2\dot{\phi}\cos\theta(\dot{q}_{21} - \dot{q}_{22}) - 2\ddot{\theta}(q_{11} - q_{12}) - 2\dot{\theta}(\dot{q}_{11} - \dot{q}_{12})\Big\} + \frac{\rho A_2}{2}\Bigg[\dot{r}^2 + \dot{\theta}^2\left(L_1 - \frac{L_2}{2} + \frac{r}{2}\right)^2$$

$$+ \dot{\phi}^2\left(L_1 - \frac{L_2}{2} + \frac{r}{2}\right)^2\cos^2\theta\Bigg] - \frac{\rho A_2}{2}(L_2 - r)\Bigg[\dot{\theta}^2\left(L_1 - \frac{L_2}{2} + \frac{r}{2}\right)$$

$$+ \dot{\phi}^2\left(L_1 - \frac{L_2}{2} + \frac{r}{2}\right)\cos^2\theta\Bigg] + \frac{\rho A_2(L_2 - r)^2}{8}(\dot{\phi}^2\cos^2\theta + \dot{\theta}^2)$$

$$- \frac{\rho A_2}{2}\Big\{\dot{r}^2 + \dot{\phi}^2\cos^2\theta(q_{21}^2 + q_{22}^2) + \dot{\theta}^2(q_{11}^2 + q_{12}^2) + 2\dot{r}\dot{\phi}\cos\theta(.78q_{21} + .43q_{22})$$

$$- 2\dot{r}\dot{\theta}(.78q_{11} + .43q_{12}) - 2\dot{\phi}\dot{\theta}\cos\theta(q_{11}q_{21} + q_{12}q_{22}) + \dot{r}q_{11}\dot{q}_{11} + 1.3\dot{r}q_{11}\dot{q}_{12} - 1.3\dot{r}q_{12}\dot{q}_{11}$$

$$+ \dot{r}q_{12}\dot{q}_{12} + \dot{q}_{11}^2 + \dot{q}_{12}^2 + 2\dot{\theta}L_1\Big[.78\dot{q}_{11} + .43\dot{q}_{12}\Big] + 2\dot{\theta}(.35\dot{r}q_{11} - .25\dot{r}q_{12} + 1.14r\dot{q}_{11} + 0.182r\dot{q}_{12})$$

$$- 2\dot{\phi}\sin\theta(.5\dot{r}q_{11}q_{21} + .65\dot{r}q_{11}q_{22} - .65\dot{r}q_{12}q_{21} + .5\dot{r}q_{12}q_{22} + \dot{q}_{11}q_{21} + \dot{q}_{12}q_{22})$$

$$+ \dot{\theta}^2(L_1^2 + 2L_1r + r^2) - 2\dot{\theta}L_1\dot{\phi}\sin\theta(.78q_{21} + .43q_{22}) - 4\dot{\theta}\dot{\phi}\, r\sin\theta(.57q_{21} + 0.091q_{22})$$

$$+ \dot{\phi}^2\sin^2\theta(q_{21}^2 + q_{22}^2) + \dot{r}q_{21}\dot{q}_{21} + 1.3\dot{r}q_{21}\dot{q}_{22} - 1.3\dot{r}q_{22}\dot{q}_{21} + \dot{r}q_{22}\dot{q}_{22} + \dot{q}_{21}^2 + \dot{q}_{22}^2$$

$$+ 2\dot{\phi}\sin\theta(.5\dot{r}q_{21}q_{11} + .65\dot{r}q_{21}q_{12} - .65\dot{r}q_{22}q_{11} + .5\dot{r}q_{22}q_{12} + \dot{q}_{21}q_{11} + \dot{q}_{22}q_{12})$$

$$- 2L_1\dot{\phi}\cos\theta(.78\dot{q}_{21} + 43\dot{q}_{22}) - 2\dot{\phi}\cos\theta(.35\dot{r}q_{21} - .25\dot{r}q_{22} + 1.14r\dot{q}_{21} + 0.182r\dot{q}_{22})$$

$$+ \dot{\phi}^2\sin^2\theta(q_{11}^2 + q_{12}^2) - 2\dot{\phi}^2\sin\theta\cos\theta L_1(.78q_{11} + .43q_{12}) - 4\dot{\phi}^2 r\sin\theta\cos\theta(.57q_{11} + .091q_{12})$$

$$+ (L_1^2 + 2L_1r + r^2)\dot{\phi}^2\cos^2\theta\Big\} - \frac{m_p}{2}\Big\{4\dot{\theta}(\dot{q}_{11} - \dot{q}_{12}) + 2\dot{\theta}^2(L_1 + r) - 4\dot{\theta}\dot{\phi}\sin\theta(q_{21} - q_{22})$$

$$- 4\dot{\phi}\cos\theta(\dot{q}_{21} - \dot{q}_{22}) - 4\dot{\phi}^2\sin\theta\cos\theta(q_{11} - q_{12}) + 2(L_1 + r)\dot{\phi}^2\cos^2\theta\Big\} + m_{H_1}\ddot{r}$$

$$- (\dot{\phi}^2\cos^2\theta + \dot{\theta}^2)m_{H_1}(L_1 + r - L_2) = F_c - (m_2 + m_p + m_{H_1})g\sin\theta - \rho A_2 g\cos\theta(0.78q_{11} + 0.43q_{12})$$

$$+ \frac{EI}{r^4}\Big[18.5(q_{11}^2 + q_{21}^2) + 728.28(q_{12}^2 + q_{22}^2)\Big] \qquad \textit{for} \quad F_c \neq 0$$

The equation of motion for θ is,

$$\ddot{\theta} = \dot{\theta} = 0 \quad \text{for} \quad F_{as} = 0$$

and

$$\left[\frac{m_1 L_1'^2}{12} + m_1\left(\frac{L_1'}{2} - a\right)^2 + c + m_{H_1}(L_1 + r - L_2)^2\right](\ddot{\theta} + \dot{\phi}^2\sin\theta\cos\theta)$$

$$+ \rho A_2(L_2 - r)\dot{\theta}\left(L_1 - \frac{L_2}{2} + \frac{r}{2}\right)\dot{r} + \rho A_2(L_2 - r)\ddot{\theta}\left(L_1 - \frac{L_2}{2} + \frac{r}{2}\right)^2$$

$$- \dot{r}\rho A_2\dot{\theta}\left(L_1 - \frac{L_2}{2} + \frac{r}{2}\right)^2 - \frac{\rho A_2(L_2 - r)^2\dot{\theta}\dot{r}}{4} + \frac{\rho A_2(L_2 - r)^3\ddot{\theta}}{12} + 2\dot{r}\dot{\theta} m_{H_1}(L_1 + r - L_2)$$

$$+ \frac{\rho A_2}{2}\Big\{2(\ddot{\theta} r + \dot{r}\dot{\theta})(q_{11}^2 + q_{12}^2) + 2\dot{\theta} r(2q_{11}\dot{q}_{11} + 2q_{12}\dot{q}_{12}) - 2(\ddot{r} r + \dot{r}^2)(.78q_{11} + .43q_{12})$$

$$- 2\dot{r}r(.78\dot{q}_{11} + .43\dot{q}_{12}) - 2\left[\ddot{\phi}\, r\cos\theta + \dot{\phi}\,\dot{r}\cos\theta - \dot{\phi}\,\dot{\theta} r\sin\theta\right](q_{11}q_{21} + q_{12}q_{22})$$

$$- 2\dot{\phi}\, r\cos\theta(\dot{q}_{11}q_{21} + q_{11}\dot{q}_{21} + \dot{q}_{12}q_{22} + q_{12}\dot{q}_{22}) + 2L_1\Big[.78\ddot{r}\, q_{11} + .78\dot{r}\dot{q}_{11} + .43\ddot{r}\, q_{12} + .43\dot{r}\dot{q}_{12}$$

$$+ .78\dot{r}\dot{q}_{11} + .78r\ddot{q}_{11} + .43\dot{r}\dot{q}_{12} + .43r\ddot{q}_{12}\Big] + 2(\ddot{r} r + \dot{r}^2)(.35q_{11} - .25q_{12}) + 2r\dot{r}(.35\dot{q}_{11} - .25\dot{q}_{12})$$

$$+ 4r\dot{r}(.57\dot{q}_{11} + .09\dot{q}_{12}) + 2r^2(.57\ddot{q}_{11} + .09\ddot{q}_{12})$$

$$+ 2\ddot{\theta}\left(L_1^2 r + L_1 r^2 + \frac{r^3}{3}\right) + 2\dot{\theta}(L_1^2\dot{r} + 2L_1 r\dot{r} + r^2\dot{r})$$

$$- 2L_1(\ddot{\phi}\sin\theta + \dot{\phi}\,\dot{\theta}\cos\theta)(.78rq_{21} + .43rq_{22}) + 2L_1\dot{\phi}\sin\theta\left[.78(\dot{r}q_{21} + r\dot{q}_{21}) + .43(\dot{r}q_{22} + r\dot{q}_{22})\right]$$

$$- 2(\ddot{\phi}\, r^2\sin\theta + \dot{\phi}\, r^2\dot{\theta}\cos\theta + 2r\dot{r}\dot{\phi}\sin\theta)(.57q_{21} + .09q_{22}) - 2\dot{\phi}\, r^2\sin\theta(.57\dot{q}_{21} + 0.09\dot{q}_{22})\Big\}$$

$$+ \frac{m_p}{2}\Big\{8\ddot{\theta}(q_{11}^2 + q_{12}^2 - 2q_{11}q_{12}) + 8\dot{\theta}(2q_{11}\dot{q}_{11} + 2q_{12}\dot{q}_{12} - 2\dot{q}_{11}q_{12} - 2q_{11}\dot{q}_{12})$$

$$- 4\ddot{r}(q_{11} - q_{12}) - 4\dot{r}(\dot{q}_{11} - \dot{q}_{12}) - 8(\ddot{\phi}\cos\theta - \dot{\phi}\,\dot{\theta}\sin\theta)\left[q_{11}q_{21} - q_{11}q_{22} - q_{12}q_{21} + q_{12}q_{22}\right]$$

$$- 8\dot{\phi}\cos\theta\left[\dot{q}_{11}q_{21} + q_{11}\dot{q}_{21} - \dot{q}_{11}q_{22} - q_{11}\dot{q}_{22} - \dot{q}_{12}q_{21} - q_{12}\dot{q}_{21} + \dot{q}_{12}q_{22} + q_{12}\dot{q}_{22}\right]$$

$$+ 4\dot{r}(\dot{q}_{11} - \dot{q}_{12}) + 4(L_1 + r)(\ddot{q}_{11} - \ddot{q}_{12}) + 2\ddot{\theta}(L_1 + r)^2 + 4\dot{\theta}(L_1 + r)\dot{r}$$

$$- 4\left[\dot{r}\dot{\phi}\sin\theta + (L_1 + r)\ddot{\phi}\sin\theta + (L_1 + r)\dot{\phi}\,\dot{\theta}\cos\theta\right](q_{21} - q_{22}) - 4(L_1 + r)\dot{\phi}\sin\theta(\dot{q}_{21} - \dot{q}_{22})\Big\}$$

$$+ \rho A_2(L_2 - r)\dot{\phi}^2\left(L_1 - \frac{L_2}{2} + \frac{r}{2}\right)^2\cos\theta\sin\theta + \frac{\rho A_2(L_2 - r)^3}{12}\dot{\phi}^2\cos\theta\sin\theta$$

$$-\frac{\rho A_2}{2}\Big\{-2\dot{\phi}^2\cos\theta\sin\theta(rq_{21}^2 + rq_{22}^2) - 2\dot{r}\dot{\phi}\, r\sin\theta(.78q_{21} + .43q_{22}) + 2\dot{\phi}\, r\dot{\theta}\sin\theta(q_{11}q_{21} + q_{12}q_{22})$$

$$- 2\dot{\phi}\, r\dot{r}\cos\theta(.5q_{11}q_{21} + .65q_{11}q_{22} - .65q_{12}q_{21} + .5q_{12}q_{22}) - 2\dot{\phi}\, r\cos\theta(\dot{q}_{11}q_{21} + \dot{q}_{12}q_{22})$$

$$- 2\dot{\theta}rL_1\dot{\phi}\cos\theta(.78q_{21} + .43q_{22}) - 2\dot{\theta}\dot{\phi}\, r^2\cos\theta(.57q_{21} + .09q_{22}) + 2\dot{\phi}^2 r\sin\theta\cos\theta(q_{21}^2 + q_{22}^2)$$

$$+ 2r\dot{r}\dot{\phi}\cos\theta(.5q_{21}q_{11} + .65q_{21}q_{12} - .65q_{22}q_{11} + .5q_{22}q_{12}) + 2\dot{\phi}\, r\cos\theta(\dot{q}_{21}q_{11} + \dot{q}_{22}q_{12})$$

$$+ 2L_1\dot{r}\dot{\phi}\sin\theta(.78q_{21} + .43q_{22}) + 2L_1\dot{\phi}\, r\sin\theta(.78\dot{q}_{21} + .43\dot{q}_{22}) + 2r\dot{r}\dot{\phi}\sin\theta(.35q_{21} - .25q_{22})$$

$$+ 2\dot{\phi}\, r^2\sin\theta(.57\dot{q}_{21} + .09\dot{q}_{22}) + 2\dot{\phi}^2 r\sin\theta\cos\theta(q_{11}^2 + q_{12}^2) - 2\dot{\phi}^2 L_1 r(\cos^2\theta - \sin^2\theta)(.78q_{11}$$

$$+ .43q_{12}) - 2\dot{\phi}^2 r^2(\cos^2\theta - \sin^2\theta)(.57q_{11} + .09q_{12}) - 2\cos\theta\sin\theta\dot{\phi}^2\left(L_1^2 r + L_1 r^2 + \frac{r^3}{3}\right)\Big\}$$

$$-\frac{m_p}{2}\Big\{ - 8\dot{\phi}^2\cos\theta\sin\theta(q_{21}^2 + q_{22}^2 - 2q_{21}q_{22}) - 4\dot{r}\dot{\phi}\sin\theta(q_{21} - q_{22})$$

$$+ 8\dot{\phi}\,\dot{\theta}\sin\theta(q_{11}q_{21} - q_{11}q_{22} - q_{12}q_{21} + q_{12}q_{22}) - 8\dot{\phi}\cos\theta(\dot{q}_{11}q_{21} - \dot{q}_{11}q_{22} - \dot{q}_{12}q_{21} + \dot{q}_{12}q_{22})$$

$$- 4\dot{\theta}(L_1 + r)\dot{\phi}\cos\theta(q_{21} - q_{22}) + 8\dot{\phi}^2\sin\theta\cos\theta(q_{21}^2 + q_{22}^2 - 2q_{21}q_{22})$$

$$+ 8\dot{\phi}\cos\theta(\dot{q}_{21}q_{11} - \dot{q}_{21}q_{12} - \dot{q}_{22}q_{11} + \dot{q}_{22}q_{12}) + 4(L_1 + r)\dot{\phi}\sin\theta(\dot{q}_{21} - \dot{q}_{22})$$

$$+ 8\dot{\phi}^2\sin\theta\cos\theta(q_{11}^2 + q_{12}^2 - 2q_{11}q_{12}) - 4\dot{\phi}^2(L_1 + r)(\cos^2\theta - \sin^2\theta)(q_{11} - q_{12})$$

$$- 2(L_1 + r)^2\dot{\phi}^2\cos\theta\sin\theta\Big\} = -aF_{ax} - \Big\{m_1\left(\frac{L_1^t}{2} - a\right) + m_{l_1}d_{l_1} + m_{H_1}(L_1 + r - L_2)$$

$$+ m_2\left(L_1 + r - \frac{L_2}{2}\right) + m_p(L_1 + r)\Big\} g\cos\theta - \rho A_2 g\sin\theta\left[0.78rq_{11} + 0.43rq_{12}\right]$$

$$- m_p g\sin\theta\left[2q_{11} - 2q_{12}\right] \qquad \textit{for} \quad F_{ax} \neq 0$$

The equation of motion for ϕ is,

$$
\begin{aligned}
&\left[m_1\left(\frac{L_1'}{2} - a\right)^2 + \frac{m_1 L_1'^2}{12} + c + m_{H_1}(L_1 + r - L_2)^2\right]\left(\ddot{\phi}\cos^2\theta - 2\dot{\phi}\dot{\theta}\cos\theta\sin\theta\right) \\
&- \rho A_2 \dot{r}\dot{\phi}\cos^2\theta\left(L_1 - \frac{L_2}{2} + \frac{r}{2}\right)^2 + \rho A_2 \ddot{\phi}\cos^2\theta(L_2 - r)\left(L_1 - \frac{L_2}{2} + \frac{r}{2}\right)^2 \\
&+ \rho A_2 \dot{\phi}\dot{r}\cos^2\theta(L_2 - r)\left(L_1 - \frac{L_2}{2} + \frac{r}{2}\right) - 2\rho A_2 \dot{\phi}\dot{\theta}\sin\theta\cos\theta(L_2 - r)\left(L_1 - \frac{L_2}{2} + \frac{r}{2}\right)^2 \\
&- \frac{\rho A_2}{4}\dot{r}\dot{\phi}\cos^2\theta(L_2 - r)^2 + \frac{\rho A_2}{12}\ddot{\phi}\cos^2\theta(L_2 - r)^3 \\
&- \frac{\rho A_2}{6}\dot{\phi}\dot{\theta}\sin\theta\cos\theta(L_2 - r)^3 + 2\dot{r}\dot{\phi}\cos^2\theta\, m_{H_1}(L_1 + r - L_2) \\
&+ \rho A_2\Big\{\left[\ddot{\phi} r\cos^2\theta + \dot{\phi}\dot{r}\cos^2\theta - 2\dot{\phi}\dot{\theta} r\cos\theta\sin\theta\right](q_{21}^2 + q_{22}^2) + \dot{\phi} r\cos^2\theta(2q_{21}\dot{q}_{21} + 2q_{22}\dot{q}_{22}) \\
&+ \left[\ddot{r} r\cos\theta + \dot{r}^2\cos\theta - \dot{r} r\dot{\theta}\sin\theta\right](0.78q_{21} + 0.43q_{22}) + \dot{r} r\cos\theta(0.78\dot{q}_{21} + 0.43\dot{q}_{22}) \\
&- \left[\ddot{\theta} r\cos\theta + \dot{\theta}\dot{r}\cos\theta - \dot{\theta}^2 r\sin\theta\right](q_{11}q_{21} + q_{12}q_{22}) - r\dot{\theta}\cos\theta(\dot{q}_{11}q_{21} + q_{11}\dot{q}_{21} + \dot{q}_{12}q_{22} + q_{12}\dot{q}_{22}) \\
&- \left[\dot{r}^2\sin\theta + \ddot{r} r\sin\theta + r\dot{r}\dot{\theta}\cos\theta\right](0.5q_{11}q_{21} + 0.65q_{11}q_{22} - 0.65q_{12}q_{21} + 0.5q_{12}q_{22}) \\
&- r\dot{r}\sin\theta\Big[0.5(\dot{q}_{11}q_{21} + q_{11}\dot{q}_{21}) + 0.65(\dot{q}_{11}q_{22} + q_{11}\dot{q}_{22}) - 0.65(\dot{q}_{12}q_{21} + q_{12}\dot{q}_{21}) \\
&+ 0.5(\dot{q}_{12}q_{22} + q_{12}\dot{q}_{22})\Big] - (\dot{r}\sin\theta + r\dot{\theta}\cos\theta)\left[\dot{q}_{11}q_{21} + \dot{q}_{12}q_{22}\right] - r\sin\theta\Big[\ddot{q}_{11}q_{21} + \dot{q}_{11}\dot{q}_{21} \\
&+ \ddot{q}_{12}q_{22} + \dot{q}_{12}\dot{q}_{22}\Big] - L_1\left[\ddot{\theta} r\sin\theta + \dot{\theta}\dot{r}\sin\theta + r\dot{\theta}^2\cos\theta\right](0.78q_{21} + 0.43q_{22}) \\
&- \dot{\theta} r L_1\sin\theta(0.78\dot{q}_{21} + 0.43\dot{q}_{22}) - (\ddot{\theta} r^2\sin\theta + 2r\dot{r}\dot{\theta}\sin\theta + r^2\dot{\theta}^2\cos\theta)\left[0.57q_{21} + 0.09q_{22}\right] \\
&- r^2\dot{\theta}\sin\theta(0.57\dot{q}_{21} + 0.09\dot{q}_{22}) + \left[\ddot{\phi} r\sin^2\theta + \dot{\phi}\dot{r}\sin^2\theta + 2\dot{\phi}\dot{\theta} r\sin\theta\cos\theta\right](q_{21}^2 + q_{22}^2) \\
&+ \dot{\phi} r\sin^2\theta(2q_{21}\dot{q}_{21} + 2q_{22}\dot{q}_{22}) + \left[\dot{r}^2\sin\theta + r\ddot{r}\sin\theta + r\dot{r}\dot{\theta}\cos\theta\right](0.5q_{21}q_{11} + 0.65q_{21}q_{12} \\
&- 0.65q_{22}q_{11} + 0.5q_{22}q_{12}) + r\dot{r}\sin\theta\Big[0.5(\dot{q}_{21}q_{11} + q_{21}\dot{q}_{11}) + 0.65(\dot{q}_{21}q_{12} + q_{21}\dot{q}_{12}) \\
&- 0.65(\dot{q}_{22}q_{11} + q_{22}\dot{q}_{11}) + 0.5(\dot{q}_{22}q_{12} + q_{22}\dot{q}_{12})\Big] + r\sin\theta(\ddot{q}_{21}q_{11} + \dot{q}_{21}\dot{q}_{11} + \ddot{q}_{22}q_{12} + \dot{q}_{22}\dot{q}_{12}) \\
&+ \left[\dot{r}\sin\theta + r\dot{\theta}\cos\theta\right](\dot{q}_{21}q_{11} + \dot{q}_{22}q_{12}) + L_1\dot{\theta}\sin\theta(0.78\dot{r}q_{21} + 0.43\dot{r}q_{22} + 0.78r\dot{q}_{21} + 0.43r\dot{q}_{22}) \\
&- L_1\cos\theta\left[0.78(\ddot{r}q_{21} + \dot{r}\dot{q}_{21}) + 0.43(\ddot{r}q_{22} + \dot{r}\dot{q}_{22}) + 0.78(\dot{r}\dot{q}_{21} + r\ddot{q}_{21}) + 0.43(\dot{r}\dot{q}_{22} + r\ddot{q}_{22})\right] \\
&- \left[\dot{r}^2\cos\theta + r\ddot{r}\cos\theta - r\dot{r}\dot{\theta}\sin\theta\right](0.35q_{21} - 0.25q_{22}) - r\dot{r}\cos\theta(0.35\dot{q}_{21} - 0.25\dot{q}_{22})
\end{aligned}
$$

$$- \left[2r\dot{r}\cos\theta - r^2\dot{\theta}\sin\theta\right](0.57\dot{q}_{21} + 0.09\dot{q}_{22}) - r^2\cos\theta(0.57\ddot{q}_{21} + 0.09\ddot{q}_{22})$$

$$+ (\ddot{\phi}\ r\sin^2\theta + \dot{\phi}\ \dot{r}\sin^2\theta + 2\dot{\phi}\ \dot{\theta}r\sin\theta\cos\theta)(q_{11}^2 + q_{12}^2) + 2\dot{\phi}\ r\sin^2\theta(q_{11}\dot{q}_{11} + q_{12}\dot{q}_{12})$$

$$- 2L_1\left[\ddot{\phi}\ r\sin\theta\cos\theta + \dot{\phi}\ \dot{\theta}r\cos^2\theta - \dot{\phi}\ \dot{\theta}r\sin^2\theta + \dot{\phi}\ \dot{r}\sin\theta\cos\theta\right](0.78q_{11} + 0.43q_{12})$$

$$- 2\dot{\phi}\ rL_1\sin\theta\cos\theta(0.78\dot{q}_{11} + 0.43\dot{q}_{12}) - 2\Big[\ddot{\phi}\ r^2\sin\theta\cos\theta + 2r\dot{r}\dot{\phi}\sin\theta\cos\theta$$

$$+ \dot{\phi}\ r^2\dot{\theta}\cos^2\theta - \dot{\phi}\ r^2\dot{\theta}\sin^2\theta\Big](0.57q_{11} + 0.09q_{12}) - 2\dot{\phi}\ r^2\sin\theta\cos\theta(0.57\dot{q}_{11} + 0.09\dot{q}_{12})$$

$$+ (L_1^2\dot{r} + 2L_1r\dot{r} + r^2\dot{r})\dot{\phi}\cos^2\theta + \left[L_1^2r + L_1r^2 + \frac{r^3}{3}\right](\ddot{\phi}\ \cos^2\theta - 2\dot{\phi}\ \dot{\theta}\sin\theta\cos\theta)\Big\}$$

$$+ m_p\Big\{4\left[\ddot{\phi}\ \cos^2\theta - 2\dot{\phi}\ \dot{\theta}\cos\theta\sin\theta\right](q_{21}^2 + q_{22}^2 - 2q_{21}q_{22})$$

$$+ 4\dot{\phi}\cos^2\theta(2q_{21}\dot{q}_{21} + 2q_{22}\dot{q}_{22} - 2\dot{q}_{21}q_{22} - 2q_{21}\dot{q}_{22}) + 2\left[\ddot{r}\cos\theta - \dot{r}\dot{\theta}\sin\theta\right](q_{21} - q_{22})$$

$$+ 2\dot{r}\cos\theta(\dot{q}_{21} - \dot{q}_{22}) - 4\left[\ddot{\theta}\cos\theta - \dot{\theta}^2\sin\theta\right](q_{11}q_{21} - q_{11}q_{22} - q_{12}q_{21} + q_{12}q_{22})$$

$$- 4\dot{\theta}\cos\theta(\dot{q}_{11}q_{21} + q_{11}\dot{q}_{21} - \dot{q}_{11}q_{22} - q_{11}\dot{q}_{22} - \dot{q}_{12}q_{21} - q_{12}\dot{q}_{21} + \dot{q}_{12}q_{22} + q_{12}\dot{q}_{22})$$

$$- 4\dot{\theta}\cos\theta(\dot{q}_{11}q_{21} - \dot{q}_{11}q_{22} - \dot{q}_{12}q_{21} + \dot{q}_{12}q_{22})$$

$$- 4\sin\theta(\ddot{q}_{11}q_{21} + \dot{q}_{11}\dot{q}_{21} - \ddot{q}_{11}q_{22} - \dot{q}_{11}\dot{q}_{22} - \ddot{q}_{12}q_{21} - \dot{q}_{12}\dot{q}_{21} + \ddot{q}_{12}q_{22} + \dot{q}_{12}\dot{q}_{22})$$

$$- 2\left[\ddot{\theta}\sin\theta(L_1 + r) + \dot{\theta}\dot{r}\sin\theta + \dot{\theta}^2\cos\theta(L_1 + r)\right](q_{21} - q_{22}) - 2\dot{\theta}\sin\theta(L_1 + r)\left[\dot{q}_{21} - \dot{q}_{22}\right]$$

$$+ 4(\ddot{\phi}\ \sin^2\theta + 2\dot{\phi}\ \dot{\theta}\sin\theta\cos\theta)\left[q_{21}^2 + q_{22}^2 - 2q_{21}q_{22}\right] + 8\dot{\phi}\sin^2\theta(q_{21}\dot{q}_{21} + q_{22}\dot{q}_{22} - \dot{q}_{21}q_{22} - q_{21}\dot{q}_{22})$$

$$+ 4\dot{\theta}\cos\theta(\dot{q}_{21}q_{11} - \dot{q}_{21}q_{12} - \dot{q}_{22}q_{11} + \dot{q}_{22}q_{12}) + 4\sin\theta(\ddot{q}_{21}q_{11} + \dot{q}_{21}\dot{q}_{11} - \ddot{q}_{21}q_{12}$$

$$- \dot{q}_{21}\dot{q}_{12} - \ddot{q}_{22}q_{11} - \dot{q}_{22}\dot{q}_{11} + \ddot{q}_{22}q_{12} + \dot{q}_{22}\dot{q}_{12}) - 2\left[\dot{r}\cos\theta - (L_1 + r)\dot{\theta}\sin\theta\right](\dot{q}_{21} - \dot{q}_{22})$$

$$- 2(L_1 + r)\cos\theta(\ddot{q}_{21} - \ddot{q}_{22}) + 4(\ddot{\phi}\ \sin^2\theta + 2\dot{\phi}\ \dot{\theta}\cos\theta\sin\theta)(q_{11}^2 + q_{12}^2 - 2q_{11}q_{12})$$

$$+ 8\dot{\phi}\sin^2\theta(q_{11}\dot{q}_{11} + q_{12}\dot{q}_{12} - \dot{q}_{11}q_{12} - q_{11}\dot{q}_{12}) - 4\Big[\ddot{\phi}\ \sin\theta\cos\theta(L_1 + r) + \dot{\phi}\ \dot{\theta}\cos^2\theta(L_1 + r)$$

$$- \dot{\phi}\ \dot{\theta}\sin^2\theta(L_1 + r) + \dot{\phi}\ \dot{r}\sin\theta\cos\theta\Big](q_{11} - q_{12}) - 4\dot{\phi}\sin\theta\cos\theta(L_1 + r)(\dot{q}_{11} - \dot{q}_{12})$$

$$+ \ddot{\phi}\ (L_1 + r)^2\cos^2\theta + 2(L_1 + r)\dot{\phi}\ \dot{r}\cos^2\theta - 2\dot{\phi}\ \dot{\theta}(L_1 + r)^2\cos\theta\sin\theta\Big\} + \ddot{\phi}(m_{H_2}a^2 + I_{l_2}) = T_{c\phi}$$

The equation of motion for $q_{11}(t)$ is,

$$\frac{\rho A_2}{2}\Big\{(\dot{r}^2 + r\ddot{r})(q_{11} - 1.3q_{12}) + r\dot{r}(\dot{q}_{11} - 1.3\dot{q}_{12}) + 2\dot{r}\dot{q}_{11} + 2r\ddot{q}_{11} + 1.56L_1(\ddot{\theta} r + \dot{\theta}\dot{r})$$

$$+ 1.14(\ddot{\theta} r^2 + 2\dot{\theta}\dot{r}r) - 2(\ddot{\phi}\ r\sin\theta q_{21} + \dot{\phi}\,\dot{\theta}r\cos\theta q_{21} + \dot{\phi}\ \dot{r}\sin\theta q_{21} + \dot{\phi}\ r\sin\theta\dot{q}_{21}) - 2\dot{\theta}^2 r q_{11}$$

$$+ 1.56r\dot{r}\dot{\theta} + 2\dot{\phi}\ \dot{\theta}r\cos\theta q_{21} - 1.52\dot{r}^2 q_{11} - 1.06\dot{r}^2 q_{12} - r\dot{r}\dot{q}_{11} - 1.3r\dot{r}\dot{q}_{12} - 1.56\dot{r}\dot{\theta}L_1 - 0.7\dot{\theta}\dot{r}r$$

$$+ 2.6\dot{\phi}\ \dot{r}r\sin\theta q_{22} - 2\dot{\phi}\ r\sin\theta\dot{q}_{21} - 2\dot{\phi}^2 r\sin^2\theta q_{11} + 1.56\dot{\phi}^2 L_1 r\sin\theta\cos\theta + 1.14\dot{\phi}^2 r^2\sin\theta\cos\theta\Big\}$$

$$+ \frac{m_T}{2}\Big\{8(\ddot{q}_{11} - \ddot{q}_{12}) + 4\ddot{\theta}(L_1 + r) + 4\dot{\theta}\dot{r} - 8(\ddot{\phi}\ \sin\theta + \dot{\phi}\ \dot{\theta}\cos\theta)\Big[q_{21} - q_{22}\Big]$$

$$- 8\dot{\phi}\sin\theta(\dot{q}_{21} - \dot{q}_{22}) - 8\dot{\theta}^2(q_{11} - q_{12}) + 4\dot{r}\dot{\theta} + 8\dot{\phi}\ \dot{\theta}\cos\theta(q_{21} - q_{22}) - 8(\dot{q}_{21} - \dot{q}_{22})\dot{\phi}\sin\theta$$

$$- 8\dot{\phi}^2\sin^2\theta(q_{11} - q_{12}) + 4\dot{\phi}^2\sin\theta\cos\theta(L_1 + r)\Big\} = -0.68m_2'' g\cos\theta - 2m_p g\cos\theta - \frac{12.362}{r^3}EIq_{11}$$

$$- \rho A_2(\dot{\phi}^2\cos^2\theta + \dot{\theta}^2)\left(L_1 r + \frac{r^2}{2}\right)\left\{\frac{4.65}{r}q_{11} - \frac{7.38}{r}q_{12}\right\}$$

$$+ \rho A_2(\dot{\phi}^2\cos^2\theta + \dot{\theta}^2)\Big[(3.08L_1 + 1.13r)q_{11} - (6.96L_1 + 3r)q_{12}\Big]$$

$$- \Big[m_T(L_1 + r)(\dot{\phi}^2\cos^2\theta + \dot{\theta}^2) + (m_p + \rho A_2 r)g\sin\theta\Big]\left[\frac{4.65}{r}q_{11} - \frac{7.38}{r}q_{12}\right]$$

$$+ \rho A_2 g\sin\theta\Big[3.08q_{11} - 6.96q_{12}\Big]$$

The equation of motion for $q_{12}(t)$ is,

$$
\begin{aligned}
&\rho A_2\Big\{(\dot r^2 + r\ddot r)(0.65q_{11} + 0.5q_{12}) + r\dot r(0.65\dot q_{11} + 0.5\dot q_{12}) + \ddot r q_{12} + r\ddot q_{12} + 0.43L_1(\ddot\theta r + \dot\theta\dot r) \\
&+ 0.09(2\dot\theta r\dot r + \ddot\theta r^2) - (\ddot\phi\ r\sin\theta q_{22} + \dot\phi\dot\theta r\cos\theta q_{22} + \dot\phi\dot r\sin\theta q_{22} + \dot\phi\ r\sin\theta\dot q_{22}) - \dot\theta^2 r q_{12} + 0.43r\dot r\dot\theta \\
&+ r\dot\phi\dot\theta\cos\theta q_{22} - 0.53\dot r^2 q_{11} - 4.34\dot r^2 q_{12} + 0.65r\dot r\dot q_{11} - 0.5r\dot r\dot q_{12} - 0.43\dot r\dot\theta L_1 + 0.25r\dot r\dot\theta \\
&- 1.3\dot\phi\ r\dot r\sin\theta q_{21} - \dot\phi\ r\sin\theta\dot q_{22} - r\dot\phi^2\sin^2\theta q_{12} + 0.43r\dot\phi^2\sin\theta\cos\theta L_1 + 0.09r^2\dot\phi^2\sin\theta\cos\theta\Big\} \\
&+ m_r\Big\{4(\ddot q_{12} - \ddot q_{11}) - 2\Big[\ddot\theta(L_1 + r) + \dot\theta\dot r\Big] - 4(\ddot\phi\ \sin\theta + \dot\phi\ \dot\theta\cos\theta)(q_{22} - q_{21}) - 4\dot\phi\ \sin\theta(\dot q_{22} - \dot q_{21}) \\
&- 4\dot\theta^2(q_{12} - q_{11}) - 2\dot r\dot\theta + 4\dot\phi\ \dot\theta\cos\theta(q_{22} - q_{21}) - 4\dot\phi\ \sin\theta(\dot q_{22} - \dot q_{21}) - 4\dot\phi^2\sin^2\theta(q_{12} - q_{11}) \\
&- 2\dot\phi^2\sin\theta\cos\theta(L_1 + r)\Big\} = -1.43m_2''\ g\cos\theta + 2m_r\,g\cos\theta - \frac{485.519}{r^3}\ EIq_{12} \\
&- \rho A_2(\dot\phi^2\cos^2\theta + \dot\theta^2)\left(L_1 r + \frac{r^2}{2}\right)\left[-\frac{7.38}{r}\ q_{11} + \frac{32.42}{r}\ q_{12}\right] \\
&+ \rho A_2(\dot\phi^2\cos^2\theta + \dot\theta^2)\Big[-(6.96L_1 + 3r)q_{11} + (23.77L_1 + 9.73r)q_{12}\Big] \\
&- \Big[m_r(L_1 + r)(\dot\phi^2\cos^2\theta + \dot\theta^2) + (m_p + \rho A_2 r)g\sin\theta\Big]\left\{-\frac{7.38}{r}\ q_{11} + \frac{32.42}{r}\ q_{12}\right\} \\
&+ \rho A_2 g\sin\theta\Big[-6.96q_{11} + 23.77q_{12}\Big]
\end{aligned}
$$

The equation of motion for $q_{21}(t)$ is,

$$
\begin{aligned}
&\rho A_2\Big\{0.5(\dot r^2 q_{21} + r\ddot r\, q_{21} + r\dot r\dot q_{21}) - 0.65(\dot r^2 q_{22} + r\ddot r\, q_{22} + r\dot r\dot q_{22}) + \dot r\dot q_{21} + r\ddot q_{21} \\
&+ \ddot\phi\, r\sin\theta q_{11} + \dot\phi\,\dot\theta r\cos\theta q_{11} + \dot\phi\,\dot r\sin\theta q_{11} + \dot\phi\, r\sin\theta\dot q_{11} \\
&- 0.78\Big[\dot r\dot\phi\, L_1\cos\theta + r\ddot\phi\; L_1\cos\theta - r\dot\phi\,\dot\theta L_1\sin\theta\Big] - 0.57\Big[2r\dot r\dot\phi\cos\theta + r^2\ddot\phi\;\cos\theta - r^2\dot\phi\,\dot\theta\sin\theta\Big] \\
&- \dot\phi^2 r\cos^2\theta q_{21} - 0.78r\dot r\dot\phi\cos\theta + r\dot\phi\,\dot\theta\cos\theta q_{11} + \dot\phi\sin\theta(0.5r\dot r q_{11} - 0.65r\dot r q_{12} + r\dot q_{11}) \\
&+ 0.78r\dot\theta L_1\dot\phi\sin\theta + 0.57r^2\dot\theta\dot\phi\sin\theta - r\dot\phi^2\sin^2\theta q_{21} - 0.76\dot r^2 q_{21} - 0.53\dot r^2 q_{22} - 0.5r\ddot r q_{21} - 0.65r\ddot r q_{22} \\
&- \dot\phi\sin\theta(0.5r\dot r q_{11} + 0.65r\dot r q_{12}) + 0.78\dot r L_1\dot\phi\cos\theta + 0.35r\dot r\dot\phi\cos\theta\Big\} + m_p\Big\{4(\ddot q_{21} - \ddot q_{22}) \\
&+ 4(\ddot\phi\;\sin\theta + \dot\phi\,\dot\theta\cos\theta)\Big[q_{11} - q_{12}\Big] + 4\dot\phi\sin\theta(\dot q_{11} - \dot q_{12}) - 2\Big[\dot r\dot\phi\cos\theta + (L_1 + r)\ddot\phi\;\cos\theta \\
&- (L_1 + r)\dot\theta\dot\phi\sin\theta\Big] - 4\dot\phi^2\cos^2\theta(q_{21} - q_{22}) - 2\dot r\dot\phi\cos\theta + 4\dot\phi\,\dot\theta\cos\theta(q_{11} - q_{12}) + 4\dot\phi\sin\theta(\dot q_{11} - \dot q_{12}) \\
&+ 2\dot\theta(L_1 + r)\dot\phi\sin\theta - 4\dot\phi^2\sin^2\theta(q_{21} - q_{22})\Big\} = \frac{-12.362}{r^3}\,EIq_{21} - \rho A_2(\dot\phi^2\cos^2\theta + \dot\theta^2)\left(L_1 r + \frac{r^2}{2}\right) \\
&\times\left[\frac{4.65}{r}\,q_{21} - \frac{7.38}{r}\,q_{22}\right] + \rho A_2(\dot\phi^2\cos^2\theta + \dot\theta^2)\Big[(3.08L_1 + 1.13r)q_{21} - (6.96L_1 + 3r)q_{22}\Big] \\
&- \Big[m_p(L_1 + r)(\dot\phi^2\cos^2\theta + \dot\theta^2) + (m_p + \rho A_2 r)g\sin\theta\Big]\left\{\frac{4.65}{r}\,q_{21} - \frac{7.38}{r}\,q_{22}\right\} \\
&+ \rho A_2 g\sin\theta(3.08q_{21} - 6.96q_{22})
\end{aligned}
$$

The equation of motion for $q_{22}(t)$ is,

$$
\begin{aligned}
&\rho A_2\Big\{0.65(\dot r^2 q_{21} + r\ddot r q_{21} + r\dot r\dot q_{21}) + 0.5(r\ddot r q_{22} + \dot r^2 q_{22} + r\dot r\dot q_{22}) \\
&+ \dot r\dot q_{22} + r\ddot q_{22} + \ddot\phi\, r\sin\theta q_{12} + \dot\phi\dot\theta r\cos\theta q_{12} + \dot\phi\,\dot r\sin\theta q_{12} + \dot\phi\, r\sin\theta\dot q_{12} \\
&- 0.43L_1(\dot r\dot\phi\cos\theta + r\ddot\phi\cos\theta - r\dot\phi\dot\theta\sin\theta) - 0.09(2r\dot r\dot\phi\cos\theta + r^2\ddot\phi\cos\theta - r^2\dot\phi\dot\theta\sin\theta) - r\dot\phi^2\cos^2\theta q_{22} \\
&- 0.43r\dot r\dot\phi\cos\theta + r\dot\phi\dot\theta\cos\theta q_{12} + 0.43r\dot\theta L_1\dot\phi\sin\theta + 0.09r^2\dot\theta\dot\phi\sin\theta - r\dot\phi^2\sin^2\theta q_{22} - 0.53\dot r^2 q_{21} \\
&- 4.34\dot r^2 q_{22} + 0.65r\dot r\dot q_{21} - 0.5r\dot r\dot q_{22} + \dot\phi\sin\theta(1.3r\dot r q_{11} + r\dot q_{12}) + 0.43\dot r L_1\dot\phi\cos\theta - 0.25r\dot r\dot\phi\cos\theta\Big\} \\
&+ m_p\Big\{4(\ddot q_{22} - \ddot q_{21}) + 4(\ddot\phi\sin\theta + \dot\phi\dot\theta\cos\theta)(q_{12} - q_{11}) + 4\dot\phi\sin\theta(\dot q_{12} - \dot q_{11}) \\
&+ 2\Big[\dot r\dot\phi\cos\theta + (L_1 + r)\ddot\phi\cos\theta - (L_1 + r)\dot\phi\dot\theta\sin\theta\Big] - 4\dot\phi^2\cos^2\theta(q_{22} - q_{21}) \\
&+ 2\dot r\dot\phi\cos\theta + 4\dot\phi\dot\theta\cos\theta(q_{12} - q_{11}) + 4\dot\phi\sin\theta(\dot q_{12} - \dot q_{11}) - 2\dot\theta(L_1 + r)\dot\phi\sin\theta - 4\dot\phi^2\sin^2\theta(q_{22} - q_{21})\Big\} \\
&= -\frac{485.519}{r^3}EIq_{22} - \rho A_2(\dot\phi^2\cos^2\theta + \dot\theta^2)\left[L_1 r + \frac{r^2}{2}\right]\left\{-\frac{7.38}{r}q_{21} + \frac{32.42}{r}q_{22}\right\} \\
&+ \rho A_2(\dot\phi^2\cos^2\theta + \dot\theta^2)\Big[-(6.96L_1 + 3r)q_{21} + (23.77L_1 + 9.73r)q_{22}\Big] \\
&- \Big[m_p(L_1 + r)(\dot\phi^2\cos^2\theta + \dot\theta^2) + (m_p + \rho A_2 r)g\sin\theta\Big]\left\{-\frac{7.38}{r}q_{21} + \frac{32.42}{r}q_{22}\right\} \\
&+ \rho A_2 g\sin\theta\Big[-6.96q_{21} + 23.77q_{22}\Big]
\end{aligned}
$$

APPENDIX B

THE CONSTRAINED EQUATIONS OF MOTION FOR θ AND r

Only the listing of the affected equations is shown here (i.e. r and θ equations). The constrained equation of motion for r on the lower thread is

$$\Bigg\{(m_2 + m_p + m_{H_1})g\sin\theta - \Bigg[m_2\Big(L_1 + r - \frac{L_2}{2}\Big) + m_p(L_1 + r)$$

$$+ m_{H_1}(L_1 + r - L_2)\Bigg](\dot{\theta}^2 + \dot{\phi}^2\cos^2\theta) + 2\dot{\phi}\cos\theta\Big[(0.78\dot{q}_{21} + 0.43\dot{q}_{22})m_2'' + 2m_p(\dot{q}_{21} - \dot{q}_{22})\Big]$$

$$- 2\dot{\theta}\Big[m_2''(0.78\dot{q}_{11} + 0.43\dot{q}_{12}) + 2m_p(\dot{q}_{11} - \dot{q}_{12})\Big] + \ddot{\phi}\cos\theta\Big[m_2''(0.78q_{21} + 0.43q_{22})$$

$$+ 2m_p(q_{21} - q_{22})\Big] + (\dot{\phi}^2\sin\theta\cos\theta - \ddot{\theta})\Big[m_2''(0.78q_{11} + 0.43q_{12})$$

$$+ 2m_p(q_{11} - q_{12})\Big]\Bigg\}(\tan\psi_1 + \mu sgn(\dot{r})) + \Bigg\{\frac{4\pi}{ld_m}J_{l_1}(1 - \mu\tan\psi_1)$$

$$+ (m_2 + m_p + m_{H_1})(\mu + \tan\psi_1)\Bigg\}\ddot{r} + \Bigg\{2\rho A_2\dot{r}\dot{\phi}\cos\theta(0.78q_{21} + 0.43q_{22})$$

$$- 2\dot{r}\dot{\theta}\rho A_2(0.78q_{11} + 0.43q_{12})\Bigg\}(\tan\psi_1 + \mu) = \frac{2T_1}{d_m}(1 - \mu\tan\psi_1 sgn(\dot{r}))$$

where m_2'' is the mass of the part of the second beam protruding from the first beam and μ is the dynamic coefficient of friction.

The constrained equation of motion for r on the upper thread is,

$$
\begin{aligned}
&\Bigg\{(m_2 + m_p + m_{H_1})g\sin\theta - \Bigg[m_2\left(L_1 + r - \frac{L_2}{2}\right) + m_p(L_1 + r) \\
&+ m_{H_1}(L_1 + r - L_2)\Bigg](\dot\phi^2\cos^2\theta + \dot\theta^2) + 2\dot\phi\cos\theta\Big[(0.78\dot q_{21} + 0.43\dot q_{22})m_2'' + 2m_p(\dot q_{21} - \dot q_{22})\Big] \\
&- 2\dot\theta\Big[m_2''(0.78\dot q_{11} + 0.43\dot q_{12}) + 2m_p(\dot q_{11} - \dot q_{12})\Big] + \ddot\phi\cos\theta\Big[m_2''(0.78 q_{21} + 0.43 q_{22}) \\
&+ 2m_p(q_{21} - q_{22})\Big] + (\dot\phi^2\sin\theta\cos\theta - \ddot\theta)\Big[m_2''(0.78 q_{11} + 0.43 q_{12}) \\
&+ 2m_p(q_{11} - q_{12})\Big]\Bigg\}(\tan\psi_1 - \mu sgn(\dot r)) + \Bigg\{\frac{4\pi}{ld_m} J_{l_1}(1 - \mu\tan\psi_1) \\
&+ (m_2 + m_p + m_{H_1})(\mu + \tan\psi_1)\Bigg\}\ddot r + \Bigg\{2\rho A_2\dot r\dot\phi\cos\theta(0.78 q_{21} + 0.43 q_{22}) \\
&- 2\dot r\dot\theta\rho A_2(0.78 q_{11} + 0.43 q_{12})\Bigg\}(\mu + \tan\psi_1) = \frac{2T_1}{d_m}(1 + \mu\tan\psi_1 sgn(\dot r))
\end{aligned}
$$

An intermediate result for the constrained equation of motion for θ is presented here to define the quantity $R_{n\ell}$,

$$
\begin{aligned}
R_{n\ell} = {} & m_{H_2}g - \Bigg[-\frac{a}{2}m_1^L + m_1^R\frac{L_1}{2} + m_{l_1}d_{l_1} + m_2\left(L_1 + r - \frac{L_2}{2}\right) + m_{H_1}(L_1 + r - L_2) \\
&+ m_p(L_1 + r)\Bigg]\frac{g\cos\theta}{a} - \frac{\dot\phi^2\sin\theta\cos\theta}{a}\Bigg[\frac{m_1 L_1^{t\,2}}{12} + c + m_2\left(L_1 + r - \frac{L_2}{2}\right)^2 \\
&+ \frac{m_2 L_2^2}{12} + m_1\left(\frac{L_1^t}{2} - a\right)^2 + m_{H_1}(L_1 + r - L_2)^2 + m_p(L_1 + r)^2\Bigg] \\
&- \frac{\rho A_2}{a}\Big\{0.78\ddot r L_1 q_{11} + 1.56 L_1\dot r\dot q_{11} + 0.43\ddot r L_1 q_{12} + 0.86\dot r L_1\dot q_{12} + 0.78 r L_1\ddot q_{11} \\
&+ 0.43 r L_1\ddot q_{12} - 0.43(\dot r^2 q_{11} + r\ddot r q_{11} + r\dot r\dot q_{11}) - 0.68(\dot r^2 q_{12} + r\ddot r q_{12} + r\dot r\dot q_{12}) \\
&+ 0.57(2r\dot r\dot q_{11} + r^2\ddot q_{11}) + 0.09(2r\dot r\dot q_{12} + r^2\ddot q_{12})\Big\} - \frac{2m_p}{a}\Big\{(r + L_1)(\ddot q_{11} - \ddot q_{12}) - \ddot r(q_{11} - q_{12})\Big\} \\
&+ \frac{2\dot\phi\rho A_2\sin\theta}{a}\Big[(0.78 r L_1 + 0.57 r^2)\dot q_{21} + (0.43 r L_1 + 0.09 r^2)\dot q_{22} + (0.78\dot r L_1 + 0.35 r\dot r)q_{21} \\
&+ (0.43 L_1\dot r - 0.25 r\dot r)q_{22}\Big] + \frac{4\dot\phi m_p\sin\theta}{a}(L_1 + r)(\dot q_{21} - \dot q_{22})
\end{aligned}
$$

$$+ \frac{\rho A_2 \sin\theta \ddot{\phi}}{a} \left[(0.78rL_1 + 0.57r^2)q_{21} + (0.43rL_1 + 0.09r^2)q_{22}\right]$$

$$+ \frac{2m_p \ddot{\phi} \sin\theta}{a} \left[(q_{21} - q_{22})(L_1 + r)\right] + \frac{\rho A_2}{a} \dot{\phi}^2 \sin^2\theta \left[(0.78rL_1 + 0.57r^2)q_{11}\right.$$

$$\left. + (0.43rL_1 + 0.09r^2)q_{12}\right] + \frac{2m_p}{a} \dot{\phi}^2 \sin^2\theta (q_{11} - q_{12})(L_1 + r)$$

$$- \frac{\rho A_2}{a} \dot{\phi}^2 \cos^2\theta \left[(0.78rL_1 + 0.57r^2)q_{11} + (0.43rL_1 + 0.09r^2)q_{12}\right]$$

$$- \frac{2m_p}{a} \dot{\phi}^2 \cos^2\theta (L_1 + r)(q_{11} - q_{12}) + \frac{2\dot{\phi}\rho A_2 \cos\theta}{a} \left[r(q_{11}\dot{q}_{21} + q_{12}\dot{q}_{22}) + \dot{r}(0.5q_{11}q_{21} + 0.5q_{12}q_{22}\right.$$

$$\left. + 0.65q_{12}q_{21} - 0.65q_{11}q_{22})\right] + \frac{8m_p}{a} \dot{\phi}\cos\theta \left[q_{11}\dot{q}_{21} - q_{11}\dot{q}_{22} - q_{12}\dot{q}_{21} + q_{12}\dot{q}_{22}\right]$$

$$+ \frac{\ddot{\phi} \rho A_2 \cos\theta}{a} r(q_{11}q_{21} + q_{12}q_{22}) + \frac{4m_p \ddot{\phi} \cos\theta}{a} (q_{11}q_{21} - q_{12}q_{21} - q_{11}q_{22} + q_{12}q_{22})$$

$$+ \frac{m_2^{\#}}{a} \dot{\phi}^2 \sin\theta\cos\theta (q_{11}^2 + q_{12}^2) + \frac{4m_p}{a} \dot{\phi}^2 \sin\theta\cos\theta \left[q_{11}^2 + q_{12}^2 - 2q_{11}q_{12}\right]$$

where m_1^L and m_1^R are the masses of the parts of the first beam lying to the left and to the right of the pivot point 0 respectively.

The constrained equation of motion for θ on the upper thread is,

$$R_{nt}(\tan\psi_1 + \mu sgn\dot{\theta}) + \left\{ \frac{-2\dot{r}\dot{\theta}}{a} \left[m_{H_1}(L_1 + r - L_2) + m_2\left(L_1 + r - \frac{L_2}{2}\right) \right.\right.$$

$$\left. + m_p(L_1 + r)\right] - \frac{8m_p \dot{\theta}}{a} \left[q_{11}\dot{q}_{11} - q_{11}\dot{q}_{12} + q_{12}\dot{q}_{12} - q_{12}\dot{q}_{11}\right] - \frac{2\rho A_2 \dot{\theta}}{a} \left[0.5\dot{r}(q_{11}^2 + q_{12}^2)\right.$$

$$\left.\left. + r(q_{11}\dot{q}_{11} + q_{12}\dot{q}_{12})\right]\right\}(\tan\psi_1 + \mu) + \frac{2T_2}{d_m}(1 - \mu sgn(\dot{\theta})\tan\psi_1)$$

$$= \left\{ \frac{4\pi a}{l d_m} J_{l_2}(1 - \mu\tan\psi_1) + \frac{m_{H_2} a}{\sin\psi_1\cos\psi_1} + \left(\frac{1}{a} \left[m_1\left(\frac{L_1^t}{2} - a\right)^2 + \frac{m_1 L_1^{t\,2}}{12} + c \right.\right.\right.$$

$$\left. + m_2\left(L_1 + r - \frac{L_2}{2}\right)^2 + \frac{m_2 L_2^2}{12} + m_{H_1}(L_1 + r - L_2)^2 + m_p(L_1 + r)^2\right]$$

$$\left.\left. + \frac{m_2^{\#}}{a}(q_{11}^2 + q_{12}^2) + \frac{4m_p}{a}(q_{11}^2 + q_{12}^2 - 2q_{11}q_{12})\right)(\tan\psi_1 + \mu)\right\}\ddot{\theta}$$

where J_{l_2} is the mass moment of inertia of the second joint leadscrew around its axis.

The constrained equation of motion for θ on the lower thread is,

$$
\begin{aligned}
& R_{nf}(\tan\psi_1 - \mu sgn(\dot{\theta})) + \Bigg\{ \frac{-2\dot{r}\dot{\theta}}{a} \left[m_{H_1}(L_1 + r - L_2) + m_2\left(L_1 + r - \frac{L_2}{2}\right) + m_p(L_1 + r) \right] \\
& - \frac{8m_p\dot{\theta}}{a} \left[q_{11}\dot{q}_{11} - q_{11}\dot{q}_{12} + q_{12}\dot{q}_{12} - q_{12}\dot{q}_{11} \right] - \frac{2\rho A_2\dot{\theta}}{a} \Big[0.5\dot{r}(q_{11}^2 + q_{12}^2) \\
& + r(q_{11}\dot{q}_{11} + q_{12}\dot{q}_{12}) \Big] \Bigg\} (\mu + \tan\psi_1) + \frac{2T_2}{d_m} (1 + \mu\tan\psi_1 sgn(\dot{\theta})) \\
& = \Bigg\{ \frac{m_{H_2}a}{\sin\psi_1\cos\psi_1} + \Bigg[\frac{1}{a} \left(\frac{m_1 L_1^{t\,2}}{12} + c + m_2\left(L_1 + r - \frac{L_2}{2}\right)^2 + m_1\left(\frac{L_1^t}{2} - a\right)^2 \right. \\
& \left. + \frac{m_2 L_2^2}{12} + m_{H_1}(L_1 + r - L_2)^2 + m_p(L_1 + r)^2 \right) + \frac{m_2''}{a}(q_{11}^2 + q_{12}^2) \\
& + \frac{4m_p}{a}(q_{11}^2 + q_{12}^2 - 2q_{11}q_{12}) \Bigg] (\mu + \tan\psi_1) + \frac{4\pi a}{ld_m} J_{l2}(1 - \mu\tan\psi_1) \Bigg\} \ddot{\theta}
\end{aligned}
$$

APPENDIX C

THE RIGID BODY EQUATIONS OF MOTION OF THE ROBOT ARM

In an attempt to check the work done for the general case, the equations of the rigid body motion of the robot arm are derived. The system is now greatly simplified since it has only three degrees of freedom (i.e. r, θ and ϕ). The robot arm is treated as an assembly of rigid bodies whose mass centers become important in the derivation of their kinetic energies (see Fig. 2-1). For the first beam, one has,

$$\underset{\sim}{R}^{*}_{1} = \left(\frac{L_1^t}{2} - a \right) \underset{\sim}{i} \qquad \text{(C-1)}$$

where the star superscript indicates a value related to the center of mass. L_1^t is the total length of the first beam and a is the length of the portion of the first beam lying to the left of the pivot point 0. The position vectors of the mass center of the second beam and the payload are,

$$\underset{\sim}{R}^{*}_{2} = \left(L_1 + r - \frac{L_2}{2} \right) \underset{\sim}{i} \qquad \text{(C-2)}$$

$$\underset{\sim}{R_p} = (L_1 + r)\, \underset{\sim}{i} \qquad \text{(C-3)}$$

where L_1 is the length of the portion of the first beam lying to the right of the pivot point 0. r is the length of the part of the second beam protruding from the first beam and L_2 is the total length of the second beam. The position vectors of the mass center of the leadscrew and its housing that produce the translational motion of the second beam are,

$$\underset{\sim}{R}^{*}_{l_1} = d_{l_1}\underset{\sim}{i} \tag{C-4}$$

$$\underset{\sim}{R}_{H_1} = (L_1 + r - L_2)\underset{\sim}{i} \tag{C-5}$$

where d_{e_1} is the distance from the origin of $(\underset{\sim}{i}, \underset{\sim}{j}, \underset{\sim}{k})$ to the mass center of the leadscrew. The leadscrew housing is assumed to be a point mass located at the beginning of the second beam.

KINETIC ENERGY

The velocity terms can be derived from,

$$\dot{\underset{\sim}{R}}^{*}_{i} = \frac{d\underset{\sim}{R}^{*}_{i}}{dt} + \underset{\sim}{\Omega} \times \underset{\sim}{R}^{*}_{i}$$

$$\underset{\sim}{\Omega} = \dot{\phi}\,(\sin\theta\ \underset{\sim}{i} + \cos\theta\ \underset{\sim}{j}) + \dot{\theta}\ \underset{\sim}{k} \tag{C-6}$$

where $\underset{\sim}{\Omega}$ is the rotation vector of the $(\underset{\sim}{i}, \underset{\sim}{j}, \underset{\sim}{k})$ basis and $\underset{\sim}{R}^{*}_{i}$ is the position vector of the mass center for the i^{th} body. The velocity terms are then used to develop the kinetic energy for each part of the robot arm including the payload, m_p. The latter is considered to be a point mass, and does not include rotation around its own mass center. Since the first beam and the leadscrew that drives the second beam undergo fixed point rotation, then their kinetic energies can be defined by:

$$T_1 = \frac{1}{2} \underset{\sim}{\Omega} \cdot (\underset{\sim}{H_o} + \underset{\sim}{H_{l_1}}) \tag{C-7}$$

where $\underset{\sim}{H_o}$ and $\underset{\sim}{H_{l_1}}$ are the angular moments of momentum of the first beam and the leadscrew respectively around point O. The kinetic energy of the second beam has rotational and translational parts,

$$T_2 = \frac{1}{2} m_2 \dot{\underset{\sim}{R}}'_2 \cdot \dot{\underset{\sim}{R}}'_2 + \frac{1}{2} \underset{\sim}{\Omega} \cdot \underset{\sim}{H}'_2 \tag{C-8}$$

where $\underset{\sim}{H}'_2$ is the angular moment of momentum around the mass center of the second beam. Since the payload and the leadscrew housing are considered to be point masses and do not undergo rotation about their own mass centers, then their kinetic energies would simply be written as,

$$T_3 = \frac{1}{2} m_p \dot{\underset{\sim}{R}}_p \cdot \dot{\underset{\sim}{R}}_p + \frac{1}{2} m_{H_1} \dot{\underset{\sim}{R}}_{H_1} \cdot \dot{\underset{\sim}{R}}_{H_1} \tag{C-9}$$

where m_{H_1} and m_p are the masses of the leadscrew housing and the payload respectively. The kinetic energy of the leadscrew and its housing that are used to rotate the first beam around the pivot point 0 can be obtained from

$$T_4 = \frac{1}{2} \underset{\sim}{\Omega_1} \cdot \underset{\sim}{H}^o_{l_2} + \frac{1}{2} \underset{\sim}{\Omega_1} \cdot \underset{\sim}{H}^o_{H_2} \tag{C-10}$$

where $\underset{\sim}{H}^o_{l2}$, $\underset{\sim}{H}^o_{H_2}$ are the angular moments of momentum of the leadscrew and its housing respectively around point O. The expression for Ω_1 can be written as

$$\underset{\sim}{\Omega_1} = \dot{\phi} \underset{\sim}{J} \tag{C-11}$$

The total kinetic energy would then be

$$T_t = \sum_{i=1}^{4} T_i \qquad \text{(C-12)}$$

POTENTIAL ENERGY

The datum line is chosen to be aligned with the $\underset{\sim}{I}$ unit vector. It is horizontal and passes through the point O. The potential energy associated with the rigid body motion arises from the gravitational acceleration, g, and is:

$$\begin{aligned} V_t = \Bigg\{ & m_1\left(\frac{L_1^t}{2} - a\right) + m_{l_1}d_{l_1} + m_{H_1}(L_1 + r - L_2) \\ & + m_2\left(L_1 + r - \frac{L_2}{2}\right) + m_p\,(L_1 + r)\Bigg\}\, g\, \sin\theta \end{aligned} \qquad \text{(C-13)}$$

VIRTUAL WORK PRINCIPLE

The virtual work principle is employed to obtain the generalized forces. If the forces exerted on the robot arm can be divided into conservative forces, which are directly related to the potential energy expression V_t, and nonconservative forces which are not, then the virtual work term can be written as [42]:

$$\delta W = \delta W_c + \delta W_{nc} \qquad \text{(C-14)}$$

where the subscripts c and nc denote terms due to conservative and nonconservative forces respectively. δW_c is related to V_t as follows:

$$\delta W_c = -\delta V_t = -\sum_{i=1}^{n} \frac{\partial V_t}{\partial x_i}\, \delta x_i \qquad \text{(C-15)}$$

where x_i is the i^{th} generalized coordinate which, in this case, can be any of the rigid body degrees of freedom r, θ and ϕ. δW_{nc} can be written in the following form,

$$\delta W_{nc} = \sum_{i=1}^{n} Q_{inc}\, \delta z_i \tag{C-16}$$

where Q_{inc} represent nonconservative generalized forces. δW_{nc} expression can be easily derived from the free body diagram depicted in Fig. 2-2.

$$\begin{aligned} \delta W_{nc} &= \underset{\sim}{R} \cdot \delta \underset{\sim}{R}'_1 + F_c \underset{\sim}{i} \cdot \delta \underset{\sim}{R_2}\Big|_{(L_1 + r - L_2)} + T_3 \underset{\sim}{J} \cdot \delta \underset{\sim}{\chi_1} + F_{az} \underset{\sim}{j} \cdot \delta R_1\Big|_{-a} \\ &= F_c\, \delta r - a F_{az}\, \delta\theta + T_3 \delta\phi \end{aligned} \tag{C-17}$$

where $\underset{\sim}{\delta\chi_1}$ is the virtual rotation vector of $\underset{\sim}{\Omega}$. Comparison of (C-16) and (C-17) yields,

$$\begin{aligned} Q_{rnc} &= F_c \\ Q_{\theta nc} &= -aF_{az} \\ Q_{\phi nc} &= T_3 \end{aligned} \tag{C-18}$$

LAGRANGE'S EQUATIONS

Now that the expressions for the nonconservative generalized forces, the total kinetic and potential energies are derived; the equations of motion can be obtained using the following form of Lagrange's equation,

$$\frac{d}{dt}\left(\frac{\partial T_t}{\partial \dot{x}_i}\right) - \frac{\partial T_t}{\partial x_i} + \frac{\partial V_t}{\partial x_i} = Q_{inc} \tag{C-19}$$

Substitute (C-12), (C-13) and (C-18) into (C-19) to get the following:

$$\begin{aligned} &(m_{H_1} + m_2 + m_p)\ddot{r} - (\dot{\phi}^2 \cos^2\theta + \dot{\theta}^2)\Big[m_{H_1}(L_1 + r - L_2) \\ &+ m_2\left(L_1 + r - \frac{L_2}{2}\right) + m_p (L_1 + r)\Big] = F_c - g\sin\theta(m_2 + m_p + m_{H_1}) \end{aligned} \tag{C-20}$$

The unconstrained equation of motion for θ is,

$$
\begin{aligned}
&\Bigg\{ \frac{m_1 L_1'^2}{12} + m_1\left(\frac{L_1'}{2} - a\right)^2 + c + m_{H_1}(L_1 + r - L_2)^2 + m_2\left(L_1 + r - \frac{L_2}{2}\right)^2 + \frac{m_2 L_2^2}{12} \\
&+ m_p (L_1 + r)^2 \Bigg\} \ddot{\theta} + 2\dot{r}\dot{\theta}\left\{ m_{H_1}(L_1 + r - L_2) + m_2\left(L_1 + r - \frac{L_2}{2}\right) + m_p (L_1 + r)\right\} \\
&+ \dot{\phi}^2 \sin\theta\cos\theta \Bigg\{ \frac{m_1 L_1'^2}{12} + m_1\left(\frac{L_1'}{2} - a\right)^2 + c + m_{H_1}(L_1 + r - L_2)^2 \\
&+ m_2\left(L_1 + r - \frac{L_2}{2}\right)^2 + \frac{m_2 L_2^2}{12} + m_p (L_1 + r)^2 \Bigg\} = - aF_{az} - g\cos\theta\Bigg[m_1\left(\frac{L_1'}{2} - a\right) \\
&+ m_{l_1} d_{l_1} + m_{H_1}(L_1 + r - L_2) + m_2\left(L_1 + r - \frac{L_2}{2}\right) + m_p (L_1 + r)\Bigg]
\end{aligned}
\tag{C-21}
$$

where c is the mass moment of inertia of the prismatic joint leadscrew with respect to the $\underset{\sim}{j}$ and $\underset{\sim}{k}$ directions. Finally the unconstrained equation of motion for ϕ can be written as follows,

$$
\begin{aligned}
&\ddot{\phi}\Bigg\{ \Bigg[\frac{m_1 L_1'^2}{12} + m_1\left(\frac{L_1'}{2} - a\right)^2 + c + m_{H_1}(L_1 + r - L_2)^2 + m_2\left(L_1 + r - \frac{L_2}{2}\right)^2 \\
&+ \frac{m_2 L_2^2}{12} + m_p (L_1 + r)^2 \Bigg] \cos^2\theta + \left(m_{H_2} a^2 + I_{l_2}\right)\Bigg\} - 2\dot{\phi}\,\dot{\theta}\cos\theta\,\sin\theta\Bigg\{ \frac{m_1 L_1'^2}{12} \\
&+ m_1\left(\frac{L_1'}{2} - a\right)^2 + m_{H_1}(L_1 + r - L_2)^2 + c + m_2\left(L_1 + r - \frac{L_2}{2}\right)^2 \\
&+ \frac{m_2 L_2^2}{12} + m_p (L_1 + r)^2 \Bigg\} + 2\dot{r}\dot{\phi}\cos^2\theta\Bigg\{ m_{H_1}(L_1 + r - L_2) \\
&+ m_2\left(L_1 + r - \frac{L_2}{2}\right) + m_p (L_1 + r)\Bigg\} = T_3
\end{aligned}
\tag{C-22}
$$

CONSTRAINED EQUATIONS OF MOTION

The only constraints, that are considered in this study, are the ones imposed by the leadscrew transmission mechanisms. Therefore, only the r and θ equations will be affected. The constrained equations of motion can be obtained by applying the formulation derived in Chapter 3 on (C-20) and (C-21). The modified r equation would have the following form for upper thread motion

$$\Big\{-(m_2 + m_p + m_{H_1})g\sin\theta + (\dot{\phi}^2 \cos^2\theta + \dot{\theta}^2)\Big[m_p(L_1 + r) + m_2\Big(L_1 + r - \frac{L_2}{2}\Big)$$
$$+ m_{H_1}(L_1 + r - L_2)\Big]\Big\}(\tan\psi_1 - \mu sgn(\dot{r})) + \frac{2T_1}{d_m}(1 + \mu\tan\psi_1 sgn(\dot{r}))$$
$$= \Big\{(m_2 + m_p + m_{H_1})(\tan\psi_1 + \mu) + \frac{4\pi}{ld_m} J_{l_1}(1 - \mu\tan\psi_1)\Big\}\ddot{r}$$

(C-23)

and for lower thread motion

$$\Big\{-(m_2 + m_p + m_{H_1})g\sin\theta + (\dot{\phi}^2 \cos^2\theta + \dot{\theta}^2)\Big[m_p(L_1 + r) + m_2\Big(L_1 + r - \frac{L_2}{2}\Big)$$
$$+ m_{H_1}(L_1 + r - L_2)\Big]\Big\}(\tan\psi + \mu sgn(\dot{r})) + \frac{2T_1}{d_m}(1 - \mu\tan\psi_1 sgn(\dot{r}))$$
$$= \Big\{(m_2 + m_p + m_{H_1})(\tan\psi_1 + \mu) + \frac{4\pi}{ld_m} J_{l_1}(1 - \mu\tan\psi_1)\Big\}\ddot{r}$$

(C-24)

where ψ_1 is the helical angle of the thread, μ is the dynamic coefficient of friction, T_1 is the applied control torque, d_m and l are the mean diameter and the lead of the leadscrew respectively. J_{l_1} is the mass moment of inertia of the prismatic joint leadscrew around its axis.

The constrained equation of motion for θ on the lower thread is

$$
\Bigg\{ m_{H_2} g - \frac{g\cos\theta}{a}\left[-\frac{a}{2}\, m_1^L + m_1^R \frac{L_1}{2} + m_{l_1} d_{l_1} + m_{H_1}(L_1 + r - L_2) + m_2\left(L_1 + r - \frac{L_2}{2} \right) \right.
$$

$$
\left. + m_p (L_1 + r) \right] - \frac{\dot{\phi}^2 \sin\theta\cos\theta}{a}\left[c + \frac{m_1 L_1^{t\,2}}{12} + m_1\left(\frac{L_1^t}{2} - a \right)^2 + m_{H_1}(L_1 + r - L_2)^2 \right.
$$

$$
\left. + \frac{m_2 L_2^2}{12} + m_2\left(L_1 + r - \frac{L_2}{2} \right)^2 + m_p (L_1 + r)^2 \right] \Bigg\} (\tan\psi_1 - \mu sgn(\dot{\theta}))
$$

$$
- \frac{2\dot{r}\dot{\theta}}{a}\left[m_{H_1}(L_1 + r - L_2) + m_2\left(L_1 + r - \frac{L_2}{2} \right) + m_p (L_1 + r) \right] (\tan\psi_1 + \mu)
$$

$$
+ \frac{2T_2}{d_m}(1 + \mu\tan\psi_1 sgn(\dot{\theta})) = \Bigg\{ \frac{m_{H_1} a}{\sin\psi_1\cos\psi_1} + \frac{1}{a}\left[c + \frac{m_1 L_1^{t\,2}}{12} + m_1\left(\frac{L_1^t}{2} - a \right)^2 \right.
$$

$$
\left. + m_{H_1}(L_1 + r - L_2)^2 + \frac{m_2 L_2^2}{12} + m_2\left(L_1 + r - \frac{L_2}{2} \right)^2 + m_p (L_1 + r)^2 \right] (\tan\psi_1 + \mu)
$$

$$
+ \frac{4\pi a}{l d_m} J_{l_2}(1 - \mu\tan\psi_1) \Bigg\} \ddot{\theta}
$$

(C-25)

The constrained equation of motion for θ on the upper thread is:

$$
\Bigg\{ m_{H_2} g - \frac{g\cos\theta}{a}\left[-\frac{a}{2}\, m_1^L + m_1^R \frac{L_1}{2} + m_{l_1} d_{l_1} + m_{H_1}(L_1 + r - L_2) + m_2\left(L_1 + r - \frac{L_2}{2} \right) \right.
$$

$$
\left. + m_p (L_1 + r) \right] - \frac{\dot{\phi}^2 \sin\theta\cos\theta}{a}\left[c + \frac{m_1 L_1^{t\,2}}{12} + m_1\left(\frac{L_1^t}{2} - c \right) \right.
$$

$$
\left. + \frac{m_2 L_2^2}{12} + m_2\left(L_1 + r - \frac{L_2}{2} \right)^2 + m_{H_1}(L_1 + r - L_2)^2 + m_p (L_1 + r)^2 \right] \Bigg\} (\tan\psi_1 + \mu sgn(\dot{\theta}))
$$

$$
- \frac{2\dot{r}\dot{\theta}}{a}\left[m_{H_1}(L_1 + r - L_2) + m_2\left(L_1 + r - \frac{L_2}{2} \right) + m_p (L_1 + r) \right] (\tan\psi_1 + \mu)
$$

$$
+ \frac{2T_2}{d_m}(1 - \mu\tan\psi_1 sgn(\dot{\theta})) = \Bigg\{ \frac{m_{H_2} a}{\sin\psi_1\cos\psi_1} + \frac{1}{a}\left[c + \frac{m_1 L_1^{t\,2}}{12} + \frac{m_2 L_2^2}{12} \right.
$$

$$+ m_1\left(\frac{L_1^l}{2} - a\right)^2 + m_2\left(L_1 + r - \frac{L_2}{2}\right)^2 + m_{H_1}(L_1 + r - L_2)^2 + m_p(L_1 + r)^2\Bigg](\tan\psi_1 + \mu)$$

$$+ \frac{4\pi a}{l d_m} J_{l_2}(1 - \mu\tan\psi_1)\Bigg\}\ddot{\theta}$$

(C-26)

where m_1^L and m_1^R are the masses of the parts of the first beam lying to the left and to the right of the pivot point 0 respectively. J_{l_2} is the mass moment of inertia of the second joint leadscrew around its axis.

APPENDIX D

RIGID BODY CONTROLLER GAIN MATRICES

Evaluation of equation (4-5) leads to the following general form for the A and B matrices,

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & A_{42} & 0 & 0 & 0 & 0 \\ A_{51} & A_{52} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ B_{41} & 0 & 0 \\ 0 & B_{52} & 0 \\ 0 & 0 & B_{63} \end{bmatrix}$$

Upper thread motion for r and θ yields the following entries for A and B matrices,

$$A_{42} = \frac{-(m_2 + m_p + m_{H_1})g\cos\theta_e(\tan\psi_1 - \mu sgn(\dot{r}))}{(m_2 + m_p + m_{H_1})(\tan\psi_1 + \mu) + \frac{4\pi}{ld_m} J_{l_1}(1 - \mu\tan\psi_1)}$$

$$A_{51} = \left\{ -\frac{g\cos\theta_e}{a}(m_{H_1} + m_2 + m_p)(\tan\psi_1 + \mu sgn(\dot{\theta}))\alpha_{2e} - \frac{2}{a}\left[m_{H_1}(L_1 + r_e - L_2) \right.\right.$$

$$\left. + m_2\left(L_1 + r_e - \frac{L_2}{2}\right) + m_p(L_1 + r_e)\right]\left\{ m_{H_2}g - \frac{g\cos\theta_e}{a}\left[\frac{m_1^R L_1}{2} - \frac{a}{2}m_1^L + m_{l_1}d_{l_1}\right.\right.$$

$$\left.\left. + m_{H_1}(L_1 + r_e - L_2) + m_2\left(L_1 + r_e - \frac{L_2}{2}\right) + m_p(L_1 + r_e)\right]\right\}$$

$$\left. \times (\tan\psi_1 + \mu)(\tan\psi_1 + \mu sgn(\dot{\theta}))\right\} / \alpha_{2e}^2$$

$$A_{62} = \left\{ \frac{g\sin\theta_e}{a} \left[\frac{m_1^R L_1}{2} + m_{l_1} d_{l_1} + m_{H_1}(L_1 + r_e - L_2) + m_2\left(L_1 + r_e - \frac{L_2}{2}\right) \right.\right.$$

$$\left.\left. - \frac{a}{2} m_1^L + m_p (L_1 + r_e)\right] (\tan\psi_1 + \mu sgn(\dot{\theta}))\right\} / \alpha_{2e}$$

$$B_{41} = \frac{2(1 + \mu\tan\psi_1 sgn(\dot{r}))}{d_m \left[(m_2 + m_p + m_{H_1})(\tan\psi_1 + \mu) + \frac{4\pi}{l d_m} J_{l_1}(1 - \mu\tan\psi_1) \right]}$$

$$B_{62} = \frac{2(1 - \mu\tan\psi_1 sgn(\dot{\theta}))}{\alpha_{2e}\, d_m}$$

where

$$\alpha_{2e} = \frac{m_{H_2} a}{\sin\psi_1\cos\psi_1} + \frac{1}{a}\left[c + \frac{m_1 L_1^{t\,2}}{12} + m_{H_1}(L_1 + r_e - L_2)^2 + \frac{m_2 L_2^2}{12} + m_1(\frac{L_1^t}{2} - a)^2 \right.$$

$$\left. + m_2\left(L_1 + r_e - \frac{L_2}{2}\right)^2 + m_p (L_1 + r_e)^2 \right] (\tan\psi_1 + \mu) + \frac{4\pi a}{l d_m} J_{l_2}(1 - \mu\tan\psi_1)$$

Any variable with a subscript e should be evaluated at the equilibrium point. Lower thread motion for r and θ gives the following entries for A and B matrices.

$$A_{42} = \frac{-(m_2 + m_p + m_{H_1}) g\cos\theta_e (\tan\psi_1 + \mu sgn(\dot{r}))}{(m_2 + m_p + m_{H_1})(\tan\psi_1 + \mu) + \frac{4\pi}{l d_m} J_{l_1}(1 - \mu\tan\psi_1)}$$

$$A_{61} = \left\{ -\frac{g\cos\theta_e}{a} (m_{H_1} + m_2 + m_p)(\tan\psi_1 - \mu sgn(\dot{\theta}))\alpha_{2e} - \frac{2}{a} \left[m_{H_1}(L_1 + r_e - L_2) + \right.\right.$$

$$\left. m_2\left(L_1 + r_e - \frac{L_2}{2}\right) + m_p (L_1 + r_e) \right] \left\{ m_{H_2} g - \frac{g\cos\theta_e}{a} \left[\frac{m_1^R L_1}{2} - \frac{a m_1^L}{2} + m_{l_1} d_{l_1} + \right.\right.$$

$$\left.\left. m_{H_1}(L_1 + r_e - L_2) + m_2\left(L_1 + r - \frac{L_2}{2}\right) + m_p (L_1 + r_e) \right] \right\}$$

$$\left. \times\ (\tan\psi_1 + \mu)(\tan\psi_1 - \mu sgn(\dot{\theta})) \right\} / \alpha_{2e}^2$$

$$A_{52} = \left\{ \frac{g\sin\theta_e}{a} \left[\frac{m_1^R L_1}{2} + m_{l_1} d_{l_1} + m_{H_1}(L_1 + r_e - L_2) + m_2\left(L_1 + r_e - \frac{L_2}{2} \right) \right.\right.$$

$$\left.\left. - \frac{a}{2} m_1^L + m_p (L_1 + r_e) \right] (\tan\psi_1 - \mu sgn(\dot{\theta})) \right\} / \alpha_{2e}$$

$$B_{41} = \frac{2(1 - \mu\tan\psi_1 sgn(\dot{r}))}{d_m \left[(m_2 + m_p + m_{H_1})(\tan\psi_1 + \mu) + \frac{4\pi}{ld_m} J_{l_1}(1 - \mu\tan\psi_1) \right]}$$

$$B_{52} = \frac{2(1 + \mu\tan\psi_1 sgn(\dot{\theta}))}{\alpha_{2e}\, d_m}$$

Since ϕ is driven by a gear train, then one value for B_{63} is obtained,

$$B_{63} = \frac{1}{\alpha_{3e}}$$

where

$$\alpha_{3e} = \left[\frac{m_1 L_1^{t\,2}}{12} + m_1\left(\frac{L_1^t}{2} - a \right)^2 + c + m_{H_1}(L_1 + r_e - L_2)^2 + m_2\left(L_1 + r_e - \frac{L_2}{2} \right)^2 \right.$$

$$\left. + \frac{m_2 L_2^2}{12} + m_p (L_1 + r_e)^2 \right] \cos^2\theta_e + m_{H_2} a^2 + J_{l_2}$$

The gain matrix of the rigid body controller is,

$$K = \begin{bmatrix} k_{11}^S & k_{12}^S & 0 & k_{14}^S & .0 & 0 & k_{11}^I & 0 & 0 \\ k_{21}^S & k_{22}^S & 0 & 0 & k_{25}^S & 0 & 0 & k_{22}^I & 0 \\ 0 & 0 & k_{33}^S & 0 & 0 & k_{36}^S & 0 & 0 & k_{33}^I \end{bmatrix}$$

where

$$K^S_{11} = \frac{W_{d1}^2 + 2P_{d1}\xi_{d1}W_{d1}}{B_{41}}$$

$$K^S_{22} = \frac{W_{d2}^2 + 2P_{d2}\xi_{d2}W_{d2} + A_{52}}{B_{52}}$$

$$K^S_{33} = \frac{W_{d3}^2 + 2P_{d3}\xi_{d3}W_{d3}}{B_{63}}$$

$$K^S_{12} = \frac{A_{42}}{B_{41}}$$

$$K^S_{21} = \frac{A_{51}}{B_{52}}$$

$$K^S_{14} = \frac{2\xi_{d1}W_{d1} + P_{d1}}{B_{41}}$$

$$K^S_{25} = \frac{2\xi_{d2}W_{d2} + P_{d2}}{B_{52}}$$

$$K^S_{36} = \frac{2\xi_{d3}W_{d3} + P_{d3}}{B_{63}}$$

$$K^I_{11} = \frac{P_{d1}W_{d1}^2}{B_{41}}$$

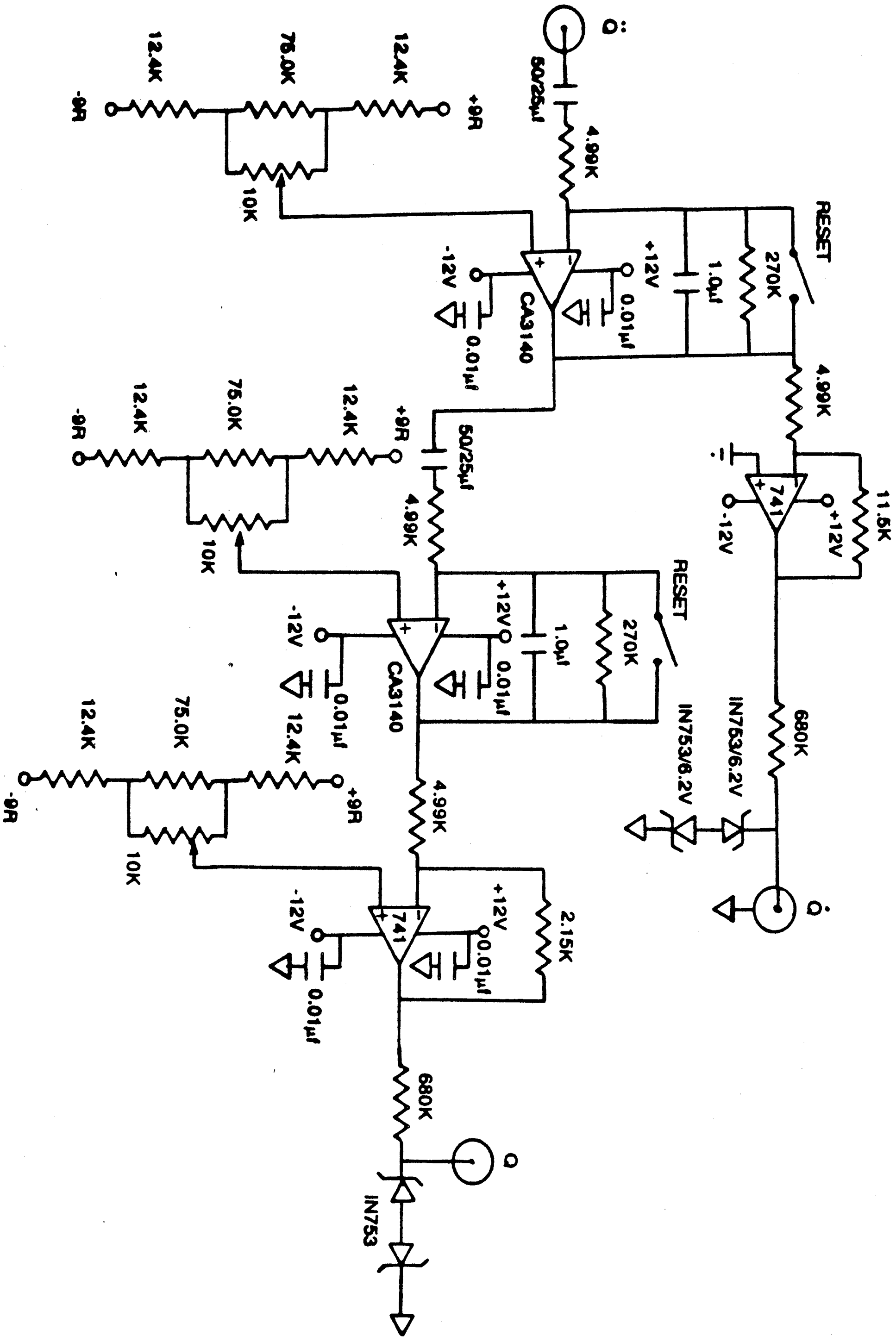
$$K^I_{22} = \frac{P_{d2}W_{d2}^2}{B_{52}}$$

$$K^I_{33} = \frac{P_{d3}W_{d3}^2}{B_{63}}$$

where ξ_{di} and W_{di} are the desired damping ratio and servo loop frequency. P_{di} is the pole assigned for the integral controller.

APPENDIX E

SCHEMATIC OF THE DOUBLE INTEGRATOR CIRCUIT


Q̈
50/25µf
4.99K
12.4K
75.0K
12.4K
-9R
+9R
10K
RESET
270K
1.0µf
+12V
-12V
0.01µf
0.01µf
CA3140
4.99K
11.5K
741
+12V
-12V
50/25µf
4.99K
12.4K
75.0K
12.4K
-9R
+9R
10K
RESET
270K
1.0µf
+12V
-12V
0.01µf
0.01µf
CA3140
680K
IN753/6.2V
IN753/6.2V
Q̇
12.4K
75.0K
12.4K
-9R
+9R
10K
4.99K
2.15K
741
+12V
-12V
0.01µf
0.01µf
680K
Q
IN753

APPENDIX F

LISTING OF THE COMPUTER CODE

FOR THE DIGITAL SIMULATION

```
C*********    MAIN PROGRAM IDENTIFICATION   ********
C
C
C       THIS PROGRAM IS WRITTEN TO SIMULATE  THE RIGID AND FLEXIBLE
C       MOTION OF A ROBOT ARM.  SEVEN DEGREES OF FREEDOM ARE USED IN
C       THIS STUDY.  THE FIRST THREE DEGREES OF FREEDOM,DENOTED Y(1)
C       TROUGH Y(3), ARE USED TO SIMULATE THE RIGID BODY MOTION,
C       WHILE THE LAST FOUR DEGREES OF FREEDOM,Y(4) THROUGH Y(7), ARE
C       USED TO SIMULATE THE FLEXIBLE MOTION WHICH IS TRUNCATED AFTER
C       ITS SECOND MODE.  THE SEVEN HIGHLY NONLINEAR SECOND ORDER
C       ORDINARY DIFFERENTIAL EQUATIONS OF MOTION ALONG WITH THE STATE
C       FEEDBACK CONTROLLER EQUATIONS ARE TRANSFORMED TO SEVENTEEN FIRST
C       ORDER ORDINARY DIFFERENTIAL EQUATIONS.  THE LATTERS ARE THEN
C       SOLVED SIMULTANEOUSLY USING GEAR'S METHOD.
C
C
C*************************************************************
C
C
C        *********     FUNCTION IDENTIFICATION        *****
C
C
C       THIS FUNCTION EVALUATES THE SGN FUNCTION
C       OF THE ABSOLUTE VALUE OF ITS ARGUMENT.
C
C
C**************************************************
C
C
      FUNCTION ASGN(X)
      REAL*8 X,ASGN,EPS1
      ASGN=0.0D0
      EPS1=0.0000000000001D0
      IF(DABS(X).GT.EPS1) GO TO 100
      ASGN=0.0D0
      GO TO 101
 100  ASGN=1.0D0
 101  RETURN
      END
C
C
C************************************************************
C
C
C      THIS FUNCTION EVALUATES THE SGN FUNCTION OF
C      ITS ARGUMENT.
C
C
C*******************************************************
C
C
      FUNCTION SGN(X)
      REAL*8 X,SGN,EPS
      SGN=0.0D0
      EPS=0.0000000000001D0
      IF(X.GT.EPS) GO TO 151
      IF(DABS(X).LE.EPS)GO TO 153
      SGN=-1.0D0
      GO TO 152
 153  SGN=0.0D0
```

```
      GO TO 152
  151 SGN=1.0D0
  152 RETURN
      END
C
C
C***************************************************************
C
C
C
C*********          MAIN PROGRAM BODY          **********
C
C
      REAL*8 TOUT,EPS,DM,H,Y(17),T0,KS11N,KS14N,IK11N
      REAL*8 WNR,WNTETA,WNFETA,POLE1,POLE2,POLE3,ZETA1
      REAL*8 ZETA2,ZETA3,MU,MH1,M1,M2,MP,L1,L2,KSY1
      REAL*8 LO,G,PIN,PIIN,B1P,B2P,B3,IK11P,IK22P,IK22N,KS12N
      REAL*8 IK33,KS11P,KS22P,KS33,KS14P,KS25P,KS36,KS22N,KS25N
      REAL*8 C1,C2,C3,JL1,L,R1,R2,R3,KS12P,KS21P,KS21N,TT3,TT4
      REAL*8 A51P,A2E,TH,M1L,M1R,LT,IL2,CONST2,CONST3
      REAL*8 RDMAX,TDMAX,RDPER,TDPER,ER1PER,ER2PER,TA1,TA2,A52P
      REAL*8 ATT1,ATT2,TT1,MUS,R11,A42N,B1N,A51N,A52N,A2,ROW,EI
      REAL*8 TW2,B2N,A42P,CONST,TW3,S
      REAL*8 A,AA1,AA2,AA3,AA4,INERTI,ML,DL,TLIMIT
      REAL*8 ATN1,ATN2,ABN1,ABN2,PHEE1,PHEE2,V,W
C
C
      COMMON /GEAR9/ NSTEP,NFE,NJE
      COMMON /GAINKP/ KS11P,KS14P,IK11P,KS22P,KS25P,
     1 IK22P,KS33,KS36,IK33,KS12P,KS21P
      COMMON /GAINKN/ KS11N,KS14N,IK11N,KS22N,KS25N,
     1 IK22N,KS12N,KS21N
      COMMON /CONTL/ C1,C2,C3
      COMMON /DATA/ M1,M2,MP,MH1,L1,L2,MU,G,KSY1,M1L,M1R,LT,IL2
     1 ,R1,R2,R3,LO,JL1,L,DM,TT3,TT4
      COMMON /DATA1/ ATT1,ATT2,MUS,CONST2,CONST3
     1 ,ML,DL,INERTI,TT1,TA1,TA2
      COMMON /SINGUL/ S,A2,ROW,EI
      COMMON /DT/ RDMAX,TDMAX,RDPER,TDPER,ER1PER,ER2PER,CONST
C
C
      REAL*4 Y1(2010),Y2(2010),Y3(2010),Y4(2010),Y5(2010),Y6(2010)
      REAL*4 Y7(2010),Y8(2010),Y9(2010),Y10(2010),Y11(2010)
      REAL*4 Y13(2010),Y14(2010),Y15(2010),Y16(2010),Y17(2010)
      REAL*4 TI(2010),Y12(2010),X(2010),AT1,YA(2010),Z(2010),ATTT
      REAL*4 D1(2010),D2(2010)
C
C
      DATA YES,NO/3HYES,2HNO/
C
      WRITE(7,10)
   10 FORMAT('1',3X,'TIME',9X,'r',14X,'THETA',11X,'PHY',11X,
     1 'Q11(T)',10X,'Q12(T)',9X,'Q21(T)')
C
C
      READ(8,71)R1,R2,R3,Y(1),Y(2)
      READ(8,71)Y(3),Y(4),Y(5),Y(6),Y(7)
   71 FORMAT(5F10.5)
   72 FORMAT(6F10.5)
      READ(8,72)WNR,WNTETA,WNFETA,POLE1,POLE2,POLE3
```

```
      READ(8,72)ZETA1,ZETA2,ZETA3,MU,MUS,G
      READ(8,603)M2,L2,A2,ROW,EI
 603  FORMAT(F15.8,F7.3,F15.10,2F10.2)
C
C                   ***  DATA  ***
C
      TT4=0.0D0
      TT3=0.0D0
      TT1=0.0D0
      TA1=0.0D0
      TA2=0.0D0
      TH=0.0D0
      N=17D0
      I=0D0
      T0=0.0D0
      H=0.00001D0
C
C        ***  INITIAL CONDITION OF THE STATE VECTOR.  ***
C
      CONST=Y(1)+((R1-Y(1))/2.0D0)
      CONST2=Y(2)+((R2-Y(2))/2.0D0)
      CONST3=Y(3)+((R3-Y(3))/2.0D0)
      TLIMIT=0.01D0
      Y(8)=0.0D0
      Y(9)=0.0D0
      Y(10)=0.0D0
      Y(11)=0.0D0
      Y(12)=0.0D0
      Y(13)=0.0D0
      Y(14)=0.0D0
      Y(15)=0.0D0
      Y(16)=0.0D0
      Y(17)=0.0D0
C
C
C        ***  DESIRED SYSTEM CHARACTERISTICS.  ***
C
      TOUT=0.0001D0
      EPS=0.00001D0
      MF=22
      INDEX=1
      M1R=0.454D0
      M1L=0.24363D0
      M1=M1L+M1R
      IL2=0.001074D0
C
C     ***  PHYSICAL SYSTEM PARAMETERS.  ***
C
      ATN1=1.8751041D0
      ATN2=4.6940911D0
      ABN1=0.7340955D0
      ABN2=1.01846644D0
      MH1=0.071D0
      MP=0.05D0
      L1=0.233D0
      KSY1=0.06997D0
      L0=0.125D0
      JL1=0.0000004464 1D0
      DM=0.00721D0
      L=0.0015875D0
```

```
      INERTI=0.000694D0
      ML=0.0687D0
      DL=0.087D0
      LT=L1+L0
C
C          *** ERROR CRITERION **
C
      R11=L1+R1
      ATT1=R1-Y(1)
      ATT2=R2-Y(2)
      ER1PER=0.0000127D0
      ER2PER=ER1PER/R11
C
C
      PHEE1=DCOSH(ATN1)-DCOS(ATN1)-ABN1*(DSINH(ATN1)-DSIN(ATN1))
      PHEE2=DCOSH(ATN2)-DCOS(ATN2)-ABN2*(DSINH(ATN2)-DSIN(ATN2))
      A=M2+MP+MH1
      PIN=INERTI+MH1*((L1+CONST-L2)**2)+(M1*(LT**2))/12.0D0+
     1 (M2*(L2**2))/12.0D0+M1*(((LT/2.0D0)-L0)**2)+M2*
     1 ((L1+CONST-(L2/2.0D0))**2)+MP*((L1+CONST)**2)
      PIIN=(M1R*L1)/2.0D0+M2*(L1+CONST-(L2/2.0D0))+
     1 MP*(L1+CONST)+(ML*DL)+MH1*(L1+CONST-L2)-((L0*M1L)/2.0D0)
      AA1=(MH1*L0)/(DSIN(KSY1)*DCOS(KSY1))
      AA2=(PIN*(DTAN(KSY1)+MU))/L0
      TW2=(12.566D0*JL1)/(L*DM)
      TW3=M2*(L1+CONST-(L2/2.0D0))+MP*(L1+CONST)
     1 +MH1*(L1+CONST-L2)
C
      A2E=AA1+TW2*L0*(1.0D0-(MU*DTAN(KSY1)))+AA2
      AA3=A*(DTAN(KSY1)+MU)+TW2*(1.0D0-MU*DTAN(KSY1))
      AA4=MH1-((PIIN*DCOS(CONST2))/L0)
C
      B3=1.0D0/(PIN*((DCOS(CONST2))**2)+(MH1*(L0**2)+IL2))
      IK33=(POLE3*(WNFETA**2))/B3
      KS33=((WNFETA**2)+2.0D0*POLE3*ZETA3*WNFETA)/B3
      KS36=(2.0D0*ZETA3*WNFETA+POLE3)/B3
C
C
      A42P=-(G*A*DCOS(CONST2)*(DTAN(KSY1)+MU))/AA3
      B1P=(2.0D0*(1.0D0-(MU*DTAN(KSY1))))/(AA3*DM)
      A51P=((-((G*DCOS(CONST2))/L0)*A*(DTAN(KSY1)+MU)*A2E)-(((
     1 2.0D0/L0)*((DTAN(KSY1)+MU)**2))*TW3*AA4*G))/(A2E**2)
      A52P=+(PIIN*G*DSIN(CONST2)*(DTAN(KSY1)+MU))/(L0*A2E)
      B2P=(2.0D0*(1.0D0-MU*DTAN(KSY1)))/(DM*A2E)
C
      IK11P=(POLE1*(WNR**2))/B1P
      IK22P=(POLE2*(WNTETA**2))/B2P
C
C
      KS11P=((WNR**2)+2.0D0*POLE1*ZETA1*WNR)/B1P
      KS22P=((WNTETA**2)+2.0D0*POLE2*ZETA2*WNTETA+A52P)/B2P
C
C
      KS12P=A42P/B1P
      KS21P=A51P/B2P
C
C
      KS14P=(2.0D0*ZETA1*WNR+POLE1)/B1P
      KS25P=(2.0D0*ZETA2*WNTETA+POLE2)/B2P
C
```

```
C
      A42N=-(G*A*DCOS(CONST2)*(DTAN(KSY1)-MU))/AA3
      B1N=(2.0D0*(1.0D0+(MU*DTAN(KSY1))))/(AA3*DM)
      A51N=((-((G*DCOS(CONST2))/LO)*A*(DTAN(KSY1)-MU)*A2E)-((
     1 2.0D0/LO)*(((DTAN(KSY1))**2)-(MU**2))*TW3*AA4*G))/(A2E**2)
      A52N=+(PIIN*G*DSIN(CONST2)*(DTAN(KSY1)-MU))/(LO*A2E)
      B2N=(2.0D0*(1.0D0+MU*DTAN(KSY1)))/(DM*A2E)
C
      IK11N=(POLE1*(WNR**2))/B1N
      IK22N=(POLE2*(WNTETA**2))/B2N
C
C
      KS11N=((WNR**2)+2.0D0*POLE1*ZETA1*WNR)/B1N
      KS22N=((WNTETA**2)+2.0D0*POLE2*ZETA2*WNTETA+A52N)/B2N
C
C
      KS12N=A42N/B1N
      KS21N=A51N/B2N
C
C
      KS14N=(2.0D0*ZETA1*WNR+POLE1)/B1N
      KS25N=(2.0D0*ZETA2*WNTETA+POLE2)/B2N
C
 15   IF(TH.LT.0.5D0)GO TO 343
      INDEX=2
C    ***  CALL GEAR SUBROUTINE TO SOLVE THE EQUATIONS OF MOTION.  ***
C
 343  CALL DGEAR (N,T0,H,Y,TOUT,EPS,MF,INDEX)
C
C
C      THE CONDITION FOR THE VARIABLE S IS INCLUDED TO POINT
C      OUT THE SINGULARITY OF THE INERTIA MATRIX WHENEVER IT
C      OCCURS AND THEN TO HALT THE EXECUTION OF THE PROGRAM.
C
C
      IF(S.EQ.0.0D0) GO TO 101
      GO TO 102
 101  WRITE(7,103) S
 103  FORMAT(10X,'S=',F6.4)
      GO TO 34
 102  WRITE(7,11) TOUT,Y(1),Y(2),Y(3),Y(4),Y(5),Y(6)
 11   FORMAT(1X,F7.4,2X,5(E14.8,2X),E14.8,6X)
C
C    ***  STORE THE NUMERICAL RESULTS FOR PLOTING.  ***
C
      I=I+1
      TH=TH+1.0D0
      Y1(I)=Y(1)
      Y2(I)=Y(2)
      Y3(I)=Y(3)
      Y4(I)=Y(4)
      Y5(I)=Y(5)
      Y6(I)=Y(6)
      Y7(I)=Y(7)
      Y8(I)=Y(8)
      Y9(I)=Y(9)
      Y10(I)=Y(10)
      Y11(I)=Y(11)
      Y12(I)=Y(12)
      Y13(I)=Y(13)
```

```
      Y14(I)=Y(14)
      Y15(I)=C1
      Y16(I)=C2
      Y17(I)=C3
      V=PHEE1*Y(4)+PHEE2*Y(5)
      W=PHEE1*Y(6)+PHEE2*Y(7)
      D1(I)=V
      D2(I)=W
      X(I)=(L1+Y(1))*DCOS(Y(2))*DCOS(Y(3))-V*DSIN(Y(2))*DCOS(Y(3))
     1 +W*DSIN(Y(3))
      YA(I)=(L1+Y(1))*DSIN(Y(2))+V*DCOS(Y(2))
      Z(I)=(L1+Y(1))*DCOS(Y(2))*DSIN(Y(3))-V*DSIN(Y(2))*DSIN(Y(3))
     1 -W*DCOS(Y(3))
C
C
C       Z(I) SHOULD BE CONSIDERED NEGATIVE. HOWEVER THE COORDINATE
C       SYSTEM ADOPTED BY THE PLOTTING ROUTINE IS THE LEFT HANDED
C       SYSTEM (i.e.X IS HORIZONTAL AND POINTING TO THE RIGHT.,Y
C       IS VERTICAL AND POINTING UPWARD.  Z IS PERPENDICULAR TO
C       THE PLANE OF THE SCREEN AND POINTING TOWARDS THE SCREEN.)
C       FOR THIS REASON Z(I) IS CONSIDERED POSITIVE IN ORDEER TO
C       MAKE IT AGREE WITH THE PHYSICAL SYSTEM.
C
      TI(I)=TOUT
      WRITE(6,208)TOUT
 208  FORMAT(2X,'TOUT=',F7.4)
      IF (INDEX.EQ.O) GO TO 13
      WRITE(7,12) INDEX
 12   FORMAT(//,17X,'ERROR RETURN WITH INDEX=',I7)
      GO TO 34
 13   TOUT=TOUT+0.0037DO
      IF(TOUT.GT.TLIMIT)GO TO 206
      GO TO 207
 206  TLIMIT=TLIMIT+0.01DO
      WRITE(6,203)
 203  FORMAT(//,2X,'WOULD YOU LIKE TO HALT THE EXECUTION ?  (Y/N)')
      READ(5,204)ATTT
 204  FORMAT(A4)
      IF(ATTT.EQ.YES)GO TO 205
 207  IF(TOUT.LE.6.ODO) GO TO 15

C
C    ***  OUTPUT THE REST OF THE NUMERICAL RESULTS.  ***
C
 205  WRITE(7,446)
 446  FORMAT(////,7X,'TIME',10X,'Q22(T)',13X,'r DOT',12X,
     1 'THETA DOT',11X,'PHY DOT',11X,'Q11(T) DOT',9X,
     1 'Q12(T) DOT')
C
C
      DO 444 ILL=1,I
      WRITE(7,445)TI(ILL),Y7(ILL),Y8(ILL),Y9(ILL),Y10(ILL),Y11(ILL)
     1 ,Y12(ILL)
 445  FORMAT(5X,F7.4,5X,6(E14.8,5X))
 444  CONTINUE
C
C
      WRITE(7,556)
 556  FORMAT(////,7X,'TIME',8X,'Q21(T) DOT',9X,'Q22(T) DOT',8X,
     1 'r CONTROLLER',5X,'THETA CONTROLLER',4X,'PHY CONTROLLER')
```

```
      DO 555 IT=1,I
      WRITE(7,592)TI(IT),Y13(IT),Y14(IT),Y15(IT),Y16(IT),Y17(IT)
 592  FORMAT(5X,F7.4,5X,5(E14.8,5X))
 555  CONTINUE
C
C
      WRITE(7,435)B1P,B2P,B1N,B2N,B3,IK11P,IK22P,IK33,
     1 KS11P,KS22P,KS11N,KS22N,KS14P,KS25P,KS14N,KS25N,KS36,
     1 KS12P,KS21P,KS12N,KS21N,KS33
 435  FORMAT(///,2X,'B1P=',E14.8,3X,'B2P=',E14.8,3X,'B1N=',E14.8,
     1 /,2X,'B2N=',E14.8,3X,'B3=',E14.8,3X,'IK11P=',E14.8,
     1 /,2X,'IK22P=',E14.8,3X,'IK33=',E14.8,3X,'KS11P=',E14.8,
     1 /,2X,'KS22P=',E14.8,3X,'KS11N=',E14.8,3X,'KS22N=',E14.8,
     1 /,2X,'KS14P=',E14.8,3X,'KS25P=',E14.8,3X,'KS14N=',E14.8,
     1 /,2X,'KS25N=',E14.8,3X,'KS36=',E14.8,3X,'KS12P=',E14.8,
     1 /,2X,'KS21P=',E14.8,3X,'KS12N=',E14.8,3X,'KS21N=',E14.8,
     1 /,2X,'KS33=',E14.8)
C
      WRITE(7,93)RDMAX,TDMAX,RDPER,TDPER
 93   FORMAT(2X,'RDMAX=',F20.10,4X,'TDMAX=',F20.10,/,
     1 2X,'RDPER=',F20.10,4X,'TDPER=',F20.10)
C
      WRITE(7,587)
 587  FORMAT(///,15X,'" POSITION OF THE END EFFECTOR."',///)
C
C
      WRITE(7,588)
 588  FORMAT(///,7X,'TIME',10X,'X COORD.',10X,'Y COORD.',10X,
     1 'Z COORD.',///)
C
C
      DO 589 IK=1,I
      WRITE(7,590)TI(IK),X(IK),YA(IK),Z(IK)
 590  FORMAT(5X,F7.4,6X,3(E14.8,4X))
 589  CONTINUE
C
      WRITE(7,600)
 600  FORMAT(//,3X,'TIME',6X,'V DEFLECTION',7X,'W DEFLECTION')
      DO 601 IKL=1,I
      WRITE(7,602)TI(IKL),D1(IKL),D2(IKL)
 602  FORMAT(2X,F7.4,3X,2(E14.8,5X))
 601  CONTINUE
C
C             ***  MAKE THE 2-D PLOTS.  ***
C
      CALL PLTF(I,Y1,Y2,Y3,Y4,Y5,Y6,Y7,Y8,Y9,Y10,Y11,
     1 Y12,Y13,Y14,Y15,Y16,Y17,D1,D2,TI)
C
C             ***  MAKE THE 3-D PLOT.  ***
C
      WRITE(6,111)
 111  FORMAT(2X,'WOULD YOU LIKE TO MAKE A 3-D PLOT ?  (YES/NO)')
      READ(5,112)AT1
 112  FORMAT(A4)
      IF(AT1.EQ.NO)GO TO 34
      CALL DIM3(I,X,YA,Z)
CT
 34   WRITE(7,16) NSTEP,NFE,NJE
 16   FORMAT(//,'PROBLEM COMPLETED IN',I10,2X,
     1 'STEPS',//,21X,I10,2X,'F EVALUATIONS',//
```

```
    1 21X,I10,2X,'J EVALUATIONS',//)
      STOP
      END
C
C
      SUBROUTINE DIFFUN (N,TIME,Y,YDOT)
C
C
C*********       PROGRAM IDENTIFICATION       **********
C
C
C       THIS SUBROUTINE DEFINES THE DIFFERENTIAL EQUATIONS OF MOTION
C
C
C*****************************************************
C
C
C*********       VARIABLE IDENTIFICATION       *********
C
C
C   M1=MASS OF THE FIRST BEAM
C   M2=MASS OF THE SECOND BEAM
C   MP=MASS OF THE PAYLOAD
C   MH1=MASS OF THE HOUSING OF THE LEADSCREW USED
C       TO MOVE THE SECOND BEAM IN THE r DIRECTION.
C   MH2=MASS OF THE HOUSING OF THE LEADSCREW USED
C       TO DRIVE THE ROBOT ARM IN THE TETA DIRECTION.
C   C1=TRANSLATIONAL CONTROLLER FORCE
C   C2=ROTATIONAL CONTROLLER TORQUE AROUND K VECTOR
C   C3=ROTATIONAL CONTROLLER TORQUE AROUND THE VERTICAL AXIS
C   L1=LENGTH OF THE FIRST BEAM
C   L2=LENGTH OF THE SECOND BEAM
C   ZETA1=DAMPING RATIO IN THE r DIRECTION
C   ZETA2=DAMPING RATIO IN THE TETA DIRECTION
C   ZETA3=DAMPING RATIO IN THE FETA DIRECTION
C   WNR=NATURAL FREQUENCY  FOR r
C   WNTETA=NATURAL FREQUENCY FOR TETA
C   WNFETA=NATURAL FREQUENCY FOR FETA
C   R1=REFERENCE INPUT IN THE r DIRECTION
C   R2=REFERENCE INPUT IN THE TETA DIRECTION
C   R3=REFERENCE INPUT IN THE FETA DIRECTION
C   IKii=GAIN FOR THE INTEGRAL CONTROLLER
C   KSii=GAIN FOR THE STATE FEEDBACK CONTROLLER.
C   Y(1)=DISPLACEMENT IN THE r DIRECTION
C   Y(2)=DISPLACEMENT IN THE TETA DIRECTION
C   Y(3)=DISPLACEMENT IN THE FETA DIRECTION
C   Y(4)=Q11(T) DISPLACEMENT.
C   Y(5)=Q12(T) DISPLACEMENT.
C   Y(6)=Q21(T) DISPLACEMENT.
C   Y(7)=Q22(T) DISPLACEMENT.
C   Y(15) TO Y(17) ARE THE INTEGRAL OF THE ERROR BETWEEN
C   THE DESIRED AND THE COMPUTED POSITIONS AND VELOCITIES.
C   A2=CROSS SECTIONAL AREA OF THE SECOND BEAM
C   ROW=DENSITY OF THE UNIFORM SECOND BEAM
C
C
C*****************************************************
C
C
C*********          STORAGE BLOCK          **********
```

```
C
C
      REAL*8 M1,M2,MP,L1,L2,R1,R2,R3,Y(N),YDOT(N),A4,B4
      REAL*8 KSY1,MH1,C1,C2,C3,ACC3,BCC,MUS,KS11N,KS14N
      REAL*8 LO,G,MU,KS11P,KS14P,KS22P,KS25P,IK11N,KS22N,N1,N2
      REAL*8 ACC7,JL1,L,KS25N,DM,KS33,KS36,IK22N,KS12N,KS21N
      REAL*8 IK11P,IK22P,IK33,ACC9,M1L,M1R,LT,IL2,ATT1,ATT2
      REAL*8 ER1PER,ER2PER,RDMAX,TDMAX,RDPER,TDPER,SGN,ASGN
      REAL*8 TT1,TT3,TT4,ABC1,ACC12,ACC13,ACC14,CONST2,CONST3
      REAL*8 CONST,TA1,TA2,ERR1,ERR2,KS12P,KS21P,ML,DL,INERTI
C
C
      REAL*8 TIME,CENTAC,DIST,PALL(7),S,PART30,PART31,PART32
      INTEGER IV(7),IJ,IN,IM,IY,IUH,JUH,K,BC,DC,PART33,PART34
      REAL*8 PBY(7),A2,ROW,A13,A14,A15,ROW7,ROW8,PART35,PART36
      REAL*8 ROW1,ROW2,ROW3,ROW4,ROW5,ROW6,PART37,PART38,PART39
      REAL*8 M2P,AL1,AL2,AL10,AL11,EI,ROW10,ROW9,PART40,PART41
      REAL*8 PART50,PART51,B(7,7),T(7,7),F(7),D(7),YVECT(7)
      REAL*8 PTB(7,7),PTF(7),PTD(7),PTBY(7),A(7,7),PART60,PART61
      REAL*8 PART1,PART2,PART3,PART4,PART5,PART62,PART63,PART64
      REAL*8 PART6,PART7,PART8,PART9,PART65,TMIN1,TMIN2
C
C
      COMMON /SINGUL/ S,A2,ROW,EI
      COMMON /GAINKP/ KS11P,KS14P,IK11P,KS22P,KS25P,
     1 IK22P,KS33,KS36,IK33,KS12P,KS21P
      COMMON /GAINKN/ KS11N,KS14N,IK11N,KS22N,KS25N,
     1 IK22N,KS12N,KS21N
      COMMON /CONTL/ C1,C2,C3
      COMMON /DATA/ M1,M2,MP,MH1,L1,L2,MU,G,KSY1,M1L,M1R,LT,IL2
     1 ,R1,R2,R3,LO,JL1,L,DM,TT3,TT4
      COMMON /DATA1/ ATT1,ATT2,MUS,CONST2,CONST3
     1 ,ML,DL,INERTI,TT1,TA1,TA2
      COMMON /DT/ RDMAX,TDMAX,RDPER,TDPER,ER1PER,ER2PER,CONST
C
C      TT1 IS INTRODUCED TO DETECT THE FIRST TIME THE INTEGRATION
C      IS CARRIED OUT.
C
      IF(TT1.GT.0.5DO)GO TO 200
      YDOT(8)=0.0DO
      YDOT(9)=0.0DO
      YDOT(10)=0.0DO
      YDOT(11)=0.0DO
      YDOT(12)=0.0DO
      YDOT(13)=0.0DO
      YDOT(14)=0.0DO
      C1=0.0DO
      C2=0.0DO
      C3=0.0DO
C
C
 200  M=7
      A13=0.76DO
      A14=0.53DO
      A15=4.34DO
      AL1=0.78DO
      AL2=0.43DO
      AL10=12.362DO
      AL11=485.519DO
C
```

```
C       INITIALIZATION OF THE T MATRIX.
C
      DO 222 I=1,7
         DO 222 J=1,7
            T(I,J)=0.0D0
 222  CONTINUE
C
C
C
C
C**************************************************
C
C
C*********            SORT BLOCK            **********
C
C
C
      CENTAC=(Y(10)**2)*((DCOS(Y(2)))**2)+(Y(9)**2)
      DIST=(L1*Y(1))+((Y(1)**2)/2.0D0)
      BC=L1+Y(1)-(L2/2.0D0)
      ROW1=ROW*A2
      ROW2=L1-(L2/2.0D0)+(Y(1)/2.0D0)
      ROW3=L2-Y(1)
      ROW4=L1+Y(1)
      ROW5=(L1**2)*Y(1)+L1*(Y(1)**2)+(Y(1)**3)/3.0D0
      ROW6=ROW1/2.0D0
      ROW7=MP*ROW4
      ROW8=((ROW1*Y(1))+MP)*G*DSIN(Y(2))
      ROW9=ROW1*G*DSIN(Y(2))
      ROW10=ROW1*(Y(10)**2)*DSIN(Y(2))*DCOS(Y(2))
      M2P=ROW1*Y(1)
C
      A4=M2+MP+MH1
      ACC9=MH1/((DSIN(KSY1))*(DCOS(KSY1)))
      ACC7=(12.566D0*JL1)/(L*DM)
      B4=M2*BC+MP*ROW4
      BCC=B4+MH1*(L1+Y(1)-L2)
      ACC3=((M1R*L1)/2.0D0)+MP*(L1+Y(1))+M2*(L1+Y(1)-(L2/2.0D0))
     1 +(ML*DL)+MH1*(L1+Y(1)-L2)-((M1L*L0)/2.0D0)
      DC=((M1*(LT**2))/12.0D0)+(M2*(L2**2))/12.0D0+M2*((L1+Y(1)
     1 -(L2/2.0D0))**2)+MP*((L1+Y(1))**2)+INERTI+MH1*((L1+Y(1)
     1 -L2)**2)+M1*(((LT/2.0D0)-L0)**2)
C
C       CHECK WHETHER THE r MOTION IS ON THE UPPER OR LOWER THREAD.
C
      PART30=((2.0D0*C1)/DM-ACC7*YDOT(8))*DSIN(KSY1)+(A4*(G*
     1 DSIN(Y(2))+YDOT(8))-BCC*(((Y(10)**2)*((DCOS(Y(2)))**2))+
     1 (Y(9)**2))+2.0D0*
     1 Y(10)*DCOS(Y(2))*((0.78D0*Y(13)+0.43D0*Y(14))*M2P+2.0D0*MP*(
     1 Y(13)-Y(14)))+2.0D0*ROW1*Y(8)*Y(10)*DCOS(Y(2))*(0.78D0*Y(6)+
     1 0.43D0*Y(7))-2.0D0*Y(9)*(M2P*(0.78D0*Y(11)+0.43D0*Y(12))+
     1 2.0D0*MP*(Y(11)-Y(12))))*DCOS(KSY1)
      PART31=(-2.0D0*Y(8)*Y(9)*ROW1*(0.78D0*Y(4)+0.43D0*Y(5))+
     1 YDOT(10)*DCOS(Y(2))*(M2P*(0.78D0*Y(6)+0.43D0*Y(7))+2.0D0*MP
     1 *(Y(6)-Y(7)))-(YDOT(9)-(Y(10)**2)*DSIN(Y(2))*DCOS(Y(2)))*
     1 (M2P*(0.78D0*Y(4)+0.43D0*Y(5))+2.0D0*MP*(Y(4)-Y(5))))
     1 *DCOS(KSY1)
      N1=PART30+PART31
      IF(N1.GE.0.0D0)GO TO 535
      GO TO 600
```

```
C               ***  LOWER THREAD LOGIC FOR r MOTION.  ***
C
 535  IF(TT1.GT.0.5D0)GO TO 542
      IF(ATT1)536,537,538
 536  C1=-KS11N*(Y(1)-CONST)-KS14N*Y(8)+IK11N*Y(15)-KS12N*Y(2)
      GO TO 542
 537  C1=0.0D0
      GO TO 547
 538  C1=-KS11P*(Y(1)-CONST)-KS14P*Y(8)+IK11P*Y(15)-KS12P*Y(2)
 542  IF(TT3.GT.0.5D0)GO TO 571
      IF(SGN(Y(8)))539,540,541
 539  C1=-KS11N*(Y(1)-CONST)-KS14N*Y(8)+IK11N*Y(15)-KS12N*Y(2)
      GO TO 546
 541  C1=-KS11P*(Y(1)-CONST)-KS14P*Y(8)+IK11P*Y(15)-KS12P*Y(2)
      GO TO 546
 540  IF(TT1.LT.0.5D0)GO TO 552
      ATT1=R1-Y(1)
      IF(ATT1)549,550,551
 549  C1=-KS11N*(Y(1)-CONST)-KS14N*Y(8)+IK11N*Y(15)-KS12N*Y(2)
      GO TO 552
 550  C1=0.0D0
      GO TO 547
 551  C1=-KS11P*(Y(1)-CONST)-KS14P*Y(8)+IK11P*Y(15)-KS12P*Y(2)
 552  IF(C1)543,547,545
C
C             ***  COMPUTATION OF THE MINIMUM  ***
C             ***  TORQUES FOR r .             ***
C
 543  ABC1=1.0D0+MUS*(DTAN(KSY1))
      ACC14=DTAN(KSY1)-MUS
      TMIN1=-(-A4*G*DSIN(Y(2))+BCC*((Y(10)**2)*((DCOS(Y(2)))**2)+
     1 (Y(9)**2))-2.0D0*Y(10)*DCOS(Y(2))*((0.78D0*Y(13)+0.43D0*
     1 Y(14))*M2P+2.0D0*MP*(Y(13)-Y(14)))+2.0D0*Y(9)*(M2P*(0.78D0*
     1 Y(11)+0.43D0*Y(12))+2.0D0*MP*(Y(11)-Y(12)))-((Y(10)**2)*
     1 DSIN(Y(2))*DCOS(Y(2))-YDOT(9))*(M2P*(0.78D0*Y(4)+0.43D0*Y(5))
     1 +2.0D0*MP*(Y(4)-Y(5)))-YDOT(10)*DCOS(Y(2))*(M2P*(0.78D0*Y(6)+
     1 0.43D0*Y(7))+2.0D0*MP*(Y(6)-Y(7))))*(DM/2.0D0)*(ACC14/ABC1) .
      IF(C1.LT.TMIN1)GO TO 546
      GO TO 547
 545  ABC1=1.0D0-MUS*(DTAN(KSY1))
      ACC14=DTAN(KSY1)+MUS
      TMIN1=-(-A4*G*DSIN(Y(2))+BCC*((Y(10)**2)*((DCOS(Y(2)))**2)+
     1 (Y(9)**2))-2.0D0*Y(10)*DCOS(Y(2))*((0.78D0*Y(13)+0.43D0*
     1 Y(14))*M2P+2.0D0*MP*(Y(13)-Y(14)))+2.0D0*Y(9)*(M2P*(0.78D0*
     1 Y(11)+0.43D0*Y(12))+2.0D0*MP*(Y(11)-Y(12)))-((Y(10)**2)*
     1 DSIN(Y(2))*DCOS(Y(2))-YDOT(9))*(M2P*(0.78D0*Y(4)+0.43D0*Y(5))
     1 +2.0D0*MP*(Y(4)-Y(5)))-YDOT(10)*DCOS(Y(2))*(M2P*(0.78D0*Y(6)+
     1 0.43D0*Y(7))+2.0D0*MP*(Y(6)-Y(7))))*(DM/2.0D0)*(ACC14/ABC1)
      IF(C1.GT.TMIN1)GO TO 546
      GO TO 547
 546  IF(TA1.LT.0.5D0)GO TO 31
      IF(RDMAX.GE.DABS(Y(8)))GO TO 30
 31   RDMAX=DABS(Y(8))
 30   TA1=TA1+1.0D0
      RDPER=RDMAX*0.00001D0
      ERR1=DABS(R1-Y(1))
      IF((DABS(Y(8)).GT.RDPER).OR.(ERR1.GT.ER1PER))GO TO 34
 571  Y(1)=R1
      C1=0.0D0
      Y(15)=0.0D0
```

```
      TT3=1.0D0
      GO TO 547
C
 34   PART38=DTAN(KSY1)+MU*SGN(Y(8))
      PART39=1.0D0-(MU*DTAN(KSY1)*SGN(Y(8)))
      A(1,1)=A4*(MU+DTAN(KSY1))+ACC7*(1.0D0-MU*DTAN(KSY1))
      A(1,2)=-(M2P*(0.78D0*Y(4)+0.43D0*Y(5))+2.0D0*MP*(Y(4)-Y(5)))*
     1 PART38
      A(1,3)=DCOS(Y(2))*(M2P*(0.78D0*Y(6)+0.43D0*Y(7))+2.0D0*MP*
     1 (Y(6)-Y(7)))*PART38
      A(1,4)=0.0D0
      A(1,5)=0.0D0
      A(1,6)=0.0D0
      A(1,7)=0.0D0
      B(1,1)=-(-2.0D0*ROW1*Y(10)*DCOS(Y(2))*(0.78D0*Y(6)+0.43D0*
     1 Y(7))+2.0D0*Y(9)*ROW1*(0.78D0*Y(4)+0.43D0*Y(5)))*(MU+
     1 DTAN(KSY1))
      B(1,2)=-(BCC*Y(9)+2.0D0*(M2P*(0.78D0*Y(11)+0.43D0*Y(12))+
     1 2.0D0*MP*(Y(11)-Y(12))))*PART38
      B(1,3)=-(BCC*Y(10)*((DCOS(Y(2)))**2)-2.0D0*DCOS(Y(2))*((
     1 0.78D0*Y(13)+0.43D0*Y(14))*M2P+2.0D0*MP*(Y(13)-Y(14)))-Y(10)
     1 *DSIN(Y(2))*DCOS(Y(2))*(M2P*(0.78D0*Y(4)+0.43D0*Y(5))+2.0D0*
     1 MP*(Y(4)-Y(5))))*PART38
      B(1,4)=0.0D0
      B(1,5)=0.0D0
      B(1,6)=0.0D0
      B(1,7)=0.0D0
      F(1)=((2.0D0*C1*PART39)/DM)
      D(1)=-A4*G*DSIN(Y(2))*PART38
      GO TO 548
 547  Y(8)=0.0D0
      A(1,1)=1.0D0
      A(1,2)=0.0D0
      A(1,3)=0.0D0
      A(1,4)=0.0D0
      A(1,5)=0.0D0
      A(1,6)=0.0D0
      A(1,7)=0.0D0
      B(1,1)=0.0D0
      B(1,2)=0.0D0
      B(1,3)=0.0D0
      B(1,4)=0.0D0
      B(1,5)=0.0D0
      B(1,6)=0.0D0
      B(1,7)=0.0D0
      F(1)=0.0D0
      D(1)=0.0D0
      GO TO 548
C
C      LOGIC FOR THE r MOTION ON THE UPPER THREAD.
C
 600  IF(TT1.GT.0.5D0)GO TO 601
      IF(ATT1)602,603,604
 602  C1=-KS11P*(Y(1)-CONST)-KS14P*Y(8)+IK11P*Y(15)-KS12P*Y(2)
      GO TO 601
 603  C1=0.0D0
      GO TO 605
 604  C1=-KS11N*(Y(1)-CONST)-KS14N*Y(8)+IK11N*Y(15)-KS12N*Y(2)
 601  IF(TT3.GT.0.5D0)GO TO 606
      IF(SGN(Y(8)))607,608,609
```

```
607   C1=-KS11P*(Y(1)-CONST)-KS14P*Y(8)+IK11P*Y(15)-KS12P*Y(2)
      GO TO 610
609   C1=-KS11N*(Y(1)-CONST)-KS14N*Y(8)+IK11N*Y(15)-KS12N*Y(2)
      GO TO 610
608   IF(TT1.LT.0.5DO)GO TO 611
      ATT1=R1-Y(1)
      IF(ATT1)612,613,614
612   C1=-KS11P*(Y(1)-CONST)-KS14P*Y(8)+IK11P*Y(15)-KS12P*Y(2)
      GO TO 611
613   C1=0.0DO
      GO TO 605
614   C1=-KS11N*(Y(1)-CONST)-KS14N*Y(8)+IK11N*Y(15)-KS12N*Y(2)
611   IF(C1)615,605,617
615   ABC1=1.0DO-MUS*(DTAN(KSY1))
      ACC14=DTAN(KSY1)+MUS
      TMIN1=-(-A4*G*DSIN(Y(2))+BCC*((Y(10)**2)*((DCOS(Y(2)))**2)+
    1 (Y(9)**2))-2.0DO*Y(10)*DCOS(Y(2))*((0.78DO*Y(13)+0.43DO*
    1 Y(14))*M2P+2.0DO*MP*(Y(13)-Y(14)))+2.0DO*Y(9)*(M2P*(0.78DO*
    1 Y(11)+0.43DO*Y(12))+2.0DO*MP*(Y(11)-Y(12)))-((Y(10)**2)*
    1 DSIN(Y(2))*DCOS(Y(2))-YDOT(9))*(M2P*(0.78DO*Y(4)+0.43DO*Y(5))
    1 +2.0DO*MP*(Y(4)-Y(5)))-YDOT(10)*DCOS(Y(2))*(M2P*(0.78DO*Y(6)+
    1 0.43DO*Y(7))+2.0DO*MP*(Y(6)-Y(7))))*(DM/2.0DO)*(ACC14/ABC1)
      IF(C1.LT.TMIN1)GO TO 610
      GO TO 605
617   ABC1=1.0DO+MUS*(DTAN(KSY1))
      ACC14=DTAN(KSY1)-MUS
      TMIN1=-(-A4*G*DSIN(Y(2))+BCC*((Y(10)**2)*((DCOS(Y(2)))**2)+
    1 (Y(9)**2))-2.0DO*Y(10)*DCOS(Y(2))*((0.78DO*Y(13)+0.43DO*
    1 Y(14))*M2P+2.0DO*MP*(Y(13)-Y(14)))+2.0DO*Y(9)*(M2P*(0.78DO*
    1 Y(11)+0.43DO*Y(12))+2.0DO*MP*(Y(11)-Y(12)))-((Y(10)**2)*
    1 DSIN(Y(2))*DCOS(Y(2))-YDOT(9))*(M2P*(0.78DO*Y(4)+0.43DO*Y(5))
    1 +2.0DO*MP*(Y(4)-Y(5)))-YDOT(10)*DCOS(Y(2))*(M2P*(0.78DO*Y(6)+
    1 0.43DO*Y(7))+2.0DO*MP*(Y(6)-Y(7))))*(DM/2.0DO)*(ACC14/ABC1)
      IF(C1.GT.TMIN1)GO TO 610
      GO TO 605
610   IF(TA1.LT.0.5DO)GO TO 619
      IF(RDMAX.GE.DABS(Y(8)))GO TO 620
619   RDMAX=DABS(Y(8))
620   TA1=TA1+1.0DO
      RDPER=RDMAX*0.00001DO
      ERR1=DABS(R1-Y(1))
      IF((DABS(Y(8)).GT.RDPER).OR.(ERR1.GT.ER1PER))GO TO 621
606   Y(1)=R1
      C1=0.0DO
      Y(15)=0.0DO
      TT3=1.0DO
      GO TO 605
C
621   PART38=DTAN(KSY1)-MU*SGN(Y(8))
      PART39=1.0DO+(MU*DTAN(KSY1)*SGN(Y(8)))
      A(1,1)=A4*(MU+DTAN(KSY1))+ACC7*(1.0DO-MU*DTAN(KSY1))
      A(1,2)=-(M2P*(0.78DO*Y(4)+0.43DO*Y(5))+2.0DO*MP*(Y(4)-Y(5)))*
    1 PART38
      A(1,3)=DCOS(Y(2))*(M2P*(0.78DO*Y(6)+0.43DO*Y(7))+2.0DO*MP*
    1 (Y(6)-Y(7)))*PART38
      A(1,4)=0.0DO
      A(1,5)=0.0DO
      A(1,6)=0.0DO
      A(1,7)=0.0DO
      B(1,1)=-(-2.0DO*ROW1*Y(10)*DCOS(Y(2))*(0.78DO*Y(6)+0.43DO*
```

```
     1 Y(7))+2.0D0*Y(9)*ROW1*(0.78D0*Y(4)+0.43D0*Y(5)))*(MU+
     1 DTAN(KSY1))
       B(1,2)=-(BCC*Y(9)+2.0D0*(M2P*(0.78D0*Y(11)+0.43D0*Y(12))+
     1 2.0D0*MP*(Y(11)-Y(12))))*PART38
       B(1,3)=-(BCC*Y(10)*((DCOS(Y(2)))**2)-2.0D0*DCOS(Y(2))*((
     1 0.78D0*Y(13)+0.43D0*Y(14))*M2P+2.0D0*MP*(Y(13)-Y(14)))-Y(10)
     1 *DSIN(Y(2))*DCOS(Y(2))*(M2P*(0.78D0*Y(4)+0.43D0*Y(5))+2.0D0*
     1 MP*(Y(4)-Y(5))))*PART38
       B(1,4)=0.0D0
       B(1,5)=0.0D0
       B(1,6)=0.0D0
       B(1,7)=0.0D0
       F(1)=((2.0D0*C1*PART39)/DM)
       D(1)=-A4*G*DSIN(Y(2))*PART38
       GO TO 548
 605   Y(8)=0.0D0
       A(1,1)=1.0D0
       A(1,2)=0.0D0
       A(1,3)=0.0D0
       A(1,4)=0.0D0
       A(1,5)=0.0D0
       A(1,6)=0.0D0
       A(1,7)=0.0D0
       B(1,1)=0.0D0
       B(1,2)=0.0D0
       B(1,3)=0.0D0
       B(1,4)=0.0D0
       B(1,5)=0.0D0
       B(1,6)=0.0D0
       B(1,7)=0.0D0
       F(1)=0.0D0
       D(1)=0.0D0
C
C       CHECK WHETHER THE THETA MOTION IS OCCURING
C       ON THE UPPER OR LOWER THREAD.
C
 548   PART32=-((2.0D0*C2)/DM-ACC7*L0*YDOT(9))*DSIN(KSY1)+(MH1*G
     1 -((ACC3*G*DCOS(Y(2)))/L0)-((YDOT(9)+(Y(10)**2)*DSIN(Y(2))*
     1 DCOS(Y(2)))/L0)*DC-((2.0D0*Y(8)*Y(9))/L0)*BCC-(ROW1/L0)*(
     1 0.78D0*YDOT(8)*L1*Y(4)+1.56D0*L1*Y(8)*Y(11)+0.43D0*YDOT(8)*
     1 L1*Y(5)+0.86D0*Y(8)*Y(12)*L1+0.78D0*Y(1)*YDOT(11)*L1+
     1 0.43D0*Y(1)*L1*YDOT(12)-0.43D0*((Y(8)**2)*Y(4)+Y(1)*YDOT(8)*
     1 Y(4)+Y(1)*Y(8)*Y(11))-0.68D0*((Y(8)**2)*Y(5)+Y(1)*YDOT(8)*
     1 Y(5)+Y(1)*Y(8)*Y(12))+0.57D0*(2.0D0*Y(1)*Y(8)*Y(11)+(Y(1)**2)
     1 *YDOT(11))+0.09D0*(2.0D0*Y(1)*Y(8)*Y(12)+(Y(1)**2)*YDOT(12))
     1 ))*DCOS(KSY1)
       PART33=(-((2.0D0*MP)/L0)*(ROW4*(YDOT(11)-YDOT(12))-YDOT(8)*
     1 (Y(4)-Y(5)))+((2.0D0*Y(10)*DSIN(Y(2))*ROW1)/L0)*((0.78D0*
     1 Y(1)*L1+0.57D0*(Y(1)**2))*Y(13)+(0.43D0*Y(1)*L1+0.09D0*(
     1 Y(1)**2))*Y(14)+(0.78D0*Y(8)*L1+0.35D0*Y(1)*Y(8))*Y(6)+
     1 (0.43D0*L1*Y(8)-0.25D0*Y(1)*Y(8))*Y(7)))*DCOS(KSY1)
       PART34=(((4.0D0*Y(10)*DSIN(Y(2))*MP*ROW4)/L0)*(Y(13)-Y(14))
     1 +((ROW1*YDOT(10)*DSIN(Y(2)))/L0)*((0.78D0*Y(1)*L1+0.57D0*
     1 (Y(1)**2))*Y(6)+(0.43D0*Y(1)*L1+0.09D0*(Y(1)**2))*Y(7))
     1 +((2.0D0*MP*YDOT(10)*DSIN(Y(2)))/L0)*ROW4*(Y(6)-Y(7))+(
     1 ROW1/L0)*(((Y(10)**2)*((DSIN(Y(2)))**2))+(Y(9)**2))*
     1 ((0.78D0*Y(1)*L1+0.57D0*(Y(1)**2))*Y(4)+(0.43D0*Y(1)*L1
     1 +0.09D0*(Y(1)**2))*Y(5)))*DCOS(KSY1)
       PART35=(((2.0D0*MP)/L0)*(((Y(10)**2)*((DSIN(Y(2)))**2))+
     1 (Y(9)**2))*(Y(4)-Y(5))*ROW4-(ROW1/L0)*(((Y(10)**2)*((
```

```
     1 DCOS(Y(2)))**2))+(Y(9)**2))*((0.78D0*Y(1)*L1+0.57D0*
     1 (Y(1)**2))*Y(4)+(0.43D0*Y(1)*L1+0.09D0*(Y(1)**2))*Y(5))
     1 -((2.0D0*MP*ROW4)/L0)*(((Y(10)**2)*((DCOS(Y(2)))**2))+
     1 (Y(9)**2))*(Y(4)-Y(5))+((2.0D0*Y(10)*ROW1*DCOS(Y(2)))/L0)*(
     1 Y(1)*(Y(4)*Y(13)+Y(5)*Y(14))+Y(8)*(0.5D0*Y(4)*Y(6)+0.5D0*Y(5)
     1 *Y(7)+0.65D0*Y(5)*Y(6)-0.65D0*Y(4)*Y(7))))*DCOS(KSY1)
       PART36=(((8.0D0*MP*Y(10)*DCOS(Y(2)))/L0)*(Y(4)*Y(13)-Y(4)*
     1 Y(14)-Y(5)*Y(13)+Y(5)*Y(14))-((8.0D0*MP*Y(9))/L0)*(Y(4)*
     1 Y(11)-Y(4)*Y(12)+Y(5)*Y(12)-Y(5)*Y(11))-((2.0D0*
     1 Y(9)*ROW1)/L0)*(0.5D0*Y(8)*((Y(4)**2)+(Y(5)**2))+Y(1)*
     1 (Y(4)*Y(11)+Y(5)*Y(12)))+((YDOT(10)*DCOS(Y(2))*ROW1)/L0)
     1 *Y(1)*(Y(4)*Y(6)+Y(5)*Y(7)))*DCOS(KSY1)
       PART37=(((4.0D0*YDOT(10)*DCOS(Y(2))*MP)/L0)*(Y(4)*Y(6)-Y(5)*
     1 Y(6)-Y(4)*Y(7)+Y(5)*Y(7))-(M2P/L0)*(YDOT(9)-(Y(10)**2)*
     1 DSIN(Y(2))*DCOS(Y(2)))*((Y(4)**2)+(Y(5)**2))-((4.0D0*MP)/L0)
     1 *(YDOT(9)-(Y(10)**2)*DSIN(Y(2))*DCOS(Y(2)))*((Y(4)**2)+
     1 (Y(5)**2)-2.0D0*Y(4)*Y(5)))*DCOS(KSY1)
       N2=PART32+PART33+PART34+PART35+PART36+PART37
       IF(N2.GE.0.0D0)GO TO 700
       GO TO 800
700    IF(TT1.GT.0.5D0)GO TO 701
       IF(ATT2)702,703,704
702    C2=-KS21P*Y(1)-KS22P*(Y(2)-CONST2)-KS25P*Y(9)+IK22P*Y(16)
       GO TO 701
703    C2=0.0D0
       GO TO 705
704    C2=-KS21N*Y(1)-KS22N*(Y(2)-CONST2)-KS25N*Y(9)+IK22N*Y(16)
701    IF(TT4.GT.0.5D0)GO TO 706
       IF(SGN(Y(9)))707,708,709
707    C2=-KS21P*Y(1)-KS22P*(Y(2)-CONST2)-KS25P*Y(9)+IK22P*Y(16)
       GO TO 710
709    C2=-KS21N*Y(1)-KS22N*(Y(2)-CONST2)-KS25N*Y(9)+IK22N*Y(16)
       GO TO 710
708    IF(TT1.LT.0.5D0)GO TO 711
       ATT2=R2-Y(2)
       IF(ATT2)712,713,714
712    C2=-KS21P*Y(1)-KS22P*(Y(2)-CONST2)-KS25P*Y(9)+IK22P*Y(16)
       GO TO 711
713    C2=0.0D0
       GO TO 705
714    C2=-KS21N*Y(1)-KS22N*(Y(2)-CONST2)-KS25N*Y(9)+IK22N*Y(16)
711    IF(C2)715,705,717
715    ACC12=1.0D0-MUS*(DTAN(KSY1))
       ACC13=DTAN(KSY1)+MUS
       PART60=-(MH1*G-((ACC3*G*DCOS(Y(2)))/L0)-(((Y(10)**2)*
     1 DSIN(Y(2))*DCOS(Y(2)))/L0)*DC-(ROW1/L0)*(0.78D0*YDOT(8)*
     1 L1*Y(4)+1.56D0*L1*Y(8)*Y(11)+0.43D0*YDOT(8)
     1 *L1*Y(5)+0.86D0*Y(8)*Y(12)*L1+0.78D0*Y(1)*YDOT(11)*L1+
     1 0.43D0*Y(1)*L1*YDOT(12)-0.43D0*((Y(8)**2)*Y(4)+Y(1)*YDOT(8)*
     1 Y(4)+Y(1)*Y(8)*Y(11))-0.68D0*((Y(8)**2)*Y(5)+Y(1)*YDOT(8)*
     1 Y(5)+Y(1)*Y(8)*Y(12))+0.57D0*(2.0D0*Y(1)*Y(8)*Y(11)+(Y(1)**2)
     1 *YDOT(11))+0.09D0*(2.0D0*Y(1)*Y(8)*Y(12)+(Y(1)**2)*YDOT(12))))
       PART61=-(-((2.0D0*MP)/L0)*(ROW4*(YDOT(11)-YDOT(12))-YDOT(8)*
     1 (Y(4)-Y(5)))+((2.0D0*Y(10)*DSIN(Y(2))*ROW1)/L0)*((0.78D0*
     1 Y(1)*L1+0.57D0*(Y(1)**2))*Y(13)+(0.43D0*Y(1)*L1+0.09D0*(
     1 Y(1)**2))*Y(14)+(0.78D0*Y(8)*L1+0.35D0*Y(1)*Y(8))*Y(6)+
     1 (0.43D0*L1*Y(8)-0.25D0*Y(1)*Y(8))*Y(7)))
       PART62=-(((4.0D0*Y(10)*DSIN(Y(2))*MP*ROW4)/L0)*(Y(13)-Y(14))
     1 +((ROW1*YDOT(10)*DSIN(Y(2)))/L0)*((0.78D0*Y(1)*L1+0.57D0*
     1 (Y(1)**2))*Y(6)+(0.43D0*Y(1)*L1+0.09D0*(Y(1)**2))*Y(7))
```

```
     1 +((2.0D0*MP*YDOT(10)*DSIN(Y(2)))/L0)*ROW4*(Y(6)-Y(7))+(
     1 ROW1/L0)*((Y(10)**2)*((DSIN(Y(2)))**2))*
     1 ((0.78D0*Y(1)*L1+0.57D0*(Y(1)**2))*Y(4)+(0.43D0*Y(1)*L1
     1 +0.09D0*(Y(1)**2))*Y(5)))
       PART63=-(((2.0D0*MP)/L0)*((Y(10)**2)*((DSIN(Y(2)))**2))
     1 *(Y(4)-Y(5))*ROW4-(ROW1/L0)*((Y(10)**2)*((
     1 DCOS(Y(2)))**2))*((0.78D0*Y(1)*L1+0.57D0*
     1 (Y(1)**2))*Y(4)+(0.43D0*Y(1)*L1+0.09D0*(Y(1)**2))*Y(5))
     1 -((2.0D0*MP*ROW4)/L0)*((Y(10)**2)*((DCOS(Y(2)))**2))
     1 *(Y(4)-Y(5))+((2.0D0*Y(10)*ROW1*DCOS(Y(2)))/L0)*(
     1 Y(1)*(Y(4)*Y(13)+Y(5)*Y(14))+Y(8)*(0.5D0*Y(4)*Y(6)+0.5D0*Y(5)
     1 *Y(7)+0.65D0*Y(5)*Y(6)-0.65D0*Y(4)*Y(7))))
       PART64=-(((8.0D0*MP*Y(10)*DCOS(Y(2)))/L0)*(Y(4)*Y(13)-Y(4)*
     1 Y(14)-Y(5)*Y(13)+Y(5)*Y(14))
     1 +((YDOT(10)*DCOS(Y(2))*ROW1)/L0)*Y(1)*
     1 (Y(4)*Y(6)+Y(5)*Y(7)))
       PART65=-(((4.0D0*YDOT(10)*DCOS(Y(2))*MP)/L0)*(Y(4)*Y(6)-Y(5)*
     1 Y(6)-Y(4)*Y(7)+Y(5)*Y(7))+(M2P/L0)*(Y(10)**2)*
     1 DSIN(Y(2))*DCOS(Y(2))*((Y(4)**2)+(Y(5)**2))+((4.0D0*MP)/L0)
     1 *(Y(10)**2)*DSIN(Y(2))*DCOS(Y(2))*((Y(4)**2)+
     1 (Y(5)**2)-2.0D0*Y(4)*Y(5)))
       TMIN2=(PART60+PART61+PART62+PART63+PART64+PART65)*(DM/2.0D0)*
     1 (ACC13/ACC12)
       IF(C2.LT.TMIN2)GO TO 710
       GO TO 705
 717   ACC12=1.0D0+MUS*(DTAN(KSY1))
       ACC13=DTAN(KSY1)-MUS
       PART60=-(MH1*G-((ACC3*G*DCOS(Y(2)))/L0)-(((Y(10)**2)*
     1 DSIN(Y(2))*DCOS(Y(2)))/L0)*DC-(ROW1/L0)*(0.78D0*YDOT(8)*
     1 L1*Y(4)+1.56D0*L1*Y(8)*Y(11)+0.43D0*YDOT(8)
     1 *L1*Y(5)+0.86D0*Y(8)*Y(12)*L1+0.78D0*Y(1)*YDOT(11)*L1+
     1 0.43D0*Y(1)*L1*YDOT(12)-0.43D0*((Y(8)**2)*Y(4)+Y(1)*YDOT(8)*
     1 Y(4)+Y(1)*Y(8)*Y(11))-0.68D0*((Y(8)**2)*Y(5)+Y(1)*YDOT(8)*
     1 Y(5)+Y(1)*Y(8)*Y(12))+0.57D0*(2.0D0*Y(1)*Y(8)*Y(11)+(Y(1)**2)
     1 *YDOT(11))+0.09D0*(2.0D0*Y(1)*Y(8)*Y(12)+(Y(1)**2)*YDOT(12))))
       PART61=-(-((2.0D0*MP)/L0)*(ROW4*(YDOT(11)-YDOT(12))-YDOT(8)*
     1 (Y(4)-Y(5)))+((2.0D0*Y(10)*DSIN(Y(2))*ROW1)/L0)*((0.78D0*
     1 Y(1)*L1+0.57D0*(Y(1)**2))*Y(13)+(0.43D0*Y(1)*L1+0.09D0*(
     1 Y(1)**2))*Y(14)+(0.78D0*Y(8)*L1+0.35D0*Y(1)*Y(8))*Y(6)+
     1 (0.43D0*L1*Y(8)-0.25D0*Y(1)*Y(8))*Y(7)))
       PART62=-(((4.0D0*Y(10)*DSIN(Y(2))*MP*ROW4)/L0)*(Y(13)-Y(14))
     1 +((ROW1*YDOT(10)*DSIN(Y(2)))/L0)*((0.78D0*Y(1)*L1+0.57D0*
     1 (Y(1)**2))*Y(6)+(0.43D0*Y(1)*L1+0.09D0*(Y(1)**2))*Y(7))
     1 +((2.0D0*MP*YDOT(10)*DSIN(Y(2)))/L0)*ROW4*(Y(6)-Y(7))+(
     1 ROW1/L0)*((Y(10)**2)*((DSIN(Y(2)))**2))*
     1 ((0.78D0*Y(1)*L1+0.57D0*(Y(1)**2))*Y(4)+(0.43D0*Y(1)*L1
     1 +0.09D0*(Y(1)**2))*Y(5)))
       PART63=-(((2.0D0*MP)/L0)*((Y(10)**2)*((DSIN(Y(2)))**2))
     1 *(Y(4)-Y(5))*ROW4-(ROW1/L0)*((Y(10)**2)*((
     1 DCOS(Y(2)))**2))*((0.78D0*Y(1)*L1+0.57D0*
     1 (Y(1)**2))*Y(4)+(0.43D0*Y(1)*L1+0.09D0*(Y(1)**2))*Y(5))
     1 -((2.0D0*MP*ROW4)/L0)*((Y(10)**2)*((DCOS(Y(2)))**2))
     1 *(Y(4)-Y(5))+((2.0D0*Y(10)*ROW1*DCOS(Y(2)))/L0)*(
     1 Y(1)*(Y(4)*Y(13)+Y(5)*Y(14))+Y(8)*(0.5D0*Y(4)*Y(6)+0.5D0*Y(5)
     1 *Y(7)+0.65D0*Y(5)*Y(6)-0.65D0*Y(4)*Y(7))))
       PART64=-(((8.0D0*MP*Y(10)*DCOS(Y(2)))/L0)*(Y(4)*Y(13)-Y(4)*
     1 Y(14)-Y(5)*Y(13)+Y(5)*Y(14))
     1 +((YDOT(10)*DCOS(Y(2))*ROW1)/L0)*Y(1)*
     1 (Y(4)*Y(6)+Y(5)*Y(7)))
       PART65=-(((4.0D0*YDOT(10)*DCOS(Y(2))*MP)/L0)*(Y(4)*Y(6)-Y(5)*
```

```
     1 Y(6)-Y(4)*Y(7)+Y(5)*Y(7))+(M2P/LO)*(Y(10)**2)*
     1 DSIN(Y(2))*DCOS(Y(2))*((Y(4)**2)+(Y(5)**2))+((4.ODO*MP)/LO)
     1 *(Y(10)**2)*DSIN(Y(2))*DCOS(Y(2))*((Y(4)**2)+
     1 (Y(5)**2)-2.ODO*Y(4)*Y(5)))
       TMIN2=(PART60+PART61+PART62+PART63+PART64+PART65)*(DM/2.ODO)*
     1 (ACC13/ACC12)
       IF(C2.GT.TMIN2)GO TO 710
       GO TO 705
710    IF(TA2.LT.O.5DO)GO TO 718
       IF(TDMAX.GE.DABS(Y(9)))GO TO 719
718    TDMAX=DABS(Y(9))
719    TA2=TA2+1.ODO
       TDPER=TDMAX*O.00001DO
       ERR2=DABS(R2-Y(2))
       IF((DABS(Y(9)).GT.TDPER).OR.(ERR2.GT.ER2PER))GO TO 720
706    Y(2)=R2
       C2=O.ODO
       Y(16)=O.ODO
       TT4=1.ODO
       GO TO 705
720    PART4O=DTAN(KSY1)-MU*SGN(Y(9))
       PART41=1.ODO+MU*DTAN(KSY1)*SGN(Y(9))
       A(2,1)=((ROW1/LO)*((O.78DO*L1-O.43DO*Y(1))*Y(4)+(O.43DO*L1
     1 -O.68DO*Y(1))*Y(5))-((2.ODO*MP)/LO)*(Y(4)-Y(5)))*PART4O
       A(2,2)=ACC9*LO+ACC7*LO*(1.ODO-MU*DTAN(KSY1))+(DTAN(KSY1)+MU)
     1 *((DC/LO)+(M2P/LO)*((Y(4)**2)+(Y(5)**2))+((4.ODO*MP)/LO)
     1 *((Y(4)-Y(5))**2))
       A(2,3)=(-((ROW1*DSIN(Y(2)))/LO)*((O.78DO*Y(1)*L1+O.57DO*
     1 (Y(1)**2))*Y(6)+(O.43DO*Y(1)*L1+O.09DO*(Y(1)**2))*Y(7))-((
     1 2.ODO*MP*DSIN(Y(2)))/LO)*ROW4*(Y(6)-Y(7))-((ROW1*DCOS(Y(2)))
     1 /LO)*Y(1)*(Y(4)*Y(6)+Y(5)*Y(7))-((4.ODO*MP*DCOS(Y(2)))/LO)*(
     1 Y(4)*Y(6)-Y(5)*Y(6)-Y(4)*Y(7)+Y(5)*Y(7)))*PART4O
       A(2,4)=((ROW1/LO)*(O.78DO*Y(1)*L1+O.57DO*(Y(1)**2))+((2.ODO*
     1 MP*ROW4)/LO))*PART4O
       A(2,5)=((ROW1/LO)*(O.43DO*Y(1)*L1+O.09DO*(Y(1)**2))-((2.ODO
     1 *MP*ROW4)/LO))*PART4O
       A(2,6)=O.ODO
       A(2,7)=O.ODO
       B(2,1)=-(-(ROW1/LO)*((1.56DO*L1+O.71DO
     1 *Y(1))*Y(11)+(O.86DO*L1-O.5DO*Y(1))*Y(12)-(O.43DO*Y(4)+
     1 O.68DO*Y(5))*Y(8)))*PART4O
       B(2,2)=-(-((8.ODO*MP)/LO)*(Y(4)*Y(11)-Y(4)*Y(12)+Y(5)*Y(12)-
     1 Y(5)*Y(11))-((2.ODO*ROW1)/LO)*(O.5DO*Y(8)*((Y(4)**2)+(
     1 Y(5)**2))+Y(1)*(Y(4)*Y(11)+Y(5)*Y(12)))-((2.ODO*Y(8)*BCC)/LO)
     1 )*(MU+DTAN(KSY1))
       PART5O=(-((Y(10)*DSIN(Y(2))*DCOS(Y(2)))/LO)*DC+((2.ODO*ROW1*
     1 DSIN(Y(2)))/LO)*((O.78DO*Y(1)*L1+O.57DO*(Y(1)**2))*Y(13)+(
     1 O.43DO*Y(1)*L1+O.09DO*(Y(1)**2))*Y(14)+(O.78DO*Y(8)*L1+O.35DO
     1 *Y(1)*Y(8))*Y(6)+(O.43DO*Y(8)*L1-O.25DO*Y(1)*Y(8))*Y(7))+((
     1 4.ODO*MP*DSIN(Y(2))*ROW4)/LO)*(Y(13)-Y(14))+((ROW1*Y(10)*
     1 ((DSIN(Y(2)))**2))/LO)*((O.78DO*Y(1)*L1+O.57DO*(Y(1)**2))*Y(4)
     1 +(O.43DO*Y(1)*L1+O.09DO*(Y(1)**2))*Y(5))+((2.ODO*MP*Y(10)*((
     1 DSIN(Y(2)))**2))/LO)*ROW4*(Y(4)-Y(5)))*PART4O
       PART51=(-((ROW1*Y(10)*((DCOS(Y(2)))**2))/LO)*((O.78DO*L1*Y(1)
     1 +O.57DO*(Y(1)**2))*Y(4)+(O.43DO*Y(1)*L1+O.09DO*(Y(1)**2))*
     1 Y(5))-((2.ODO*MP*ROW4*Y(10)*((DCOS(Y(2)))**2))/LO)*(Y(4)-Y(5))
     1 +((2.ODO*ROW1*DCOS(Y(2)))/LO)*(Y(1)*(Y(4)*Y(13)+Y(5)*Y(14))+
     1 Y(8)*(O.5DO*Y(4)*Y(6)+O.5DO*Y(5)*Y(7)+O.65DO*Y(5)*Y(6)-O.65DO
     1 *Y(4)*Y(7)))+((8.ODO*MP*DCOS(Y(2)))/LO)*(Y(4)*Y(13)-Y(4)*Y(14)
     1 -Y(5)*Y(13)+Y(5)*Y(14))+((M2P*Y(10)*DSIN(Y(2))*DCOS(Y(2)))/LO)
```

```
     1 *((Y(4)**2)+(Y(5)**2))+((4.0DO*MP*Y(10)*DSIN(Y(2))*DCOS(Y(2)))
     1 /LO)*((Y(4)-Y(5))**2))*PART40
       B(2,3)=-PART50-PART51
       B(2,4)=0.0DO
       B(2,5)=0.0DO
       B(2,6)=0.0DO
       B(2,7)=0.0DO
       F(2)=(2.0DO*PART41*C2)/DM
       D(2)=(-((ACC3*G*DCOS(Y(2)))/LO)+MH1*G)*PART40
       GO TO 721
 705   Y(9)=0.0DO
       A(2,1)=0.0DO
       A(2,2)=1.0DO
       A(2,3)=0.0DO
       A(2,4)=0.0DO
       A(2,5)=0.0DO
       A(2,6)=0.0DO
       A(2,7)=0.0DO
       B(2,1)=0.0DO
       B(2,2)=0.0DO
       B(2,3)=0.0DO
       B(2,4)=0.0DO
       B(2,5)=0.0DO
       B(2,6)=0.0DO
       B(2,7)=0.0DO
       F(2)=0.0DO
       D(2)=0.0DO
       GO TO 721
C
C
C         LOGIC FOR THE THETA MOTION CARRIED ON THE UPPER THREAD.
C
 800   IF(TT1.GT.0.5DO)GO TO 801
       IF(ATT2)802,803,804
 802   C2=-KS21N*Y(1)-KS22N*(Y(2)-CONST2)-KS25N*Y(9)+IK22N*Y(16)
       GO TO 801
 803   C2=0.0DO
       GO TO 805
 804   C2=-KS21P*Y(1)-KS22P*(Y(2)-CONST2)-KS25P*Y(9)+IK22P*Y(16)
 801   IF(TT4.GT.0.5DO)GO TO 806
       IF(SGN(Y(9)))807,808,809
 807   C2=-KS21N*Y(1)-KS22N*(Y(2)-CONST2)-KS25N*Y(9)+IK22N*Y(16)
       GO TO 810
 809   C2=-KS21P*Y(1)-KS22P*(Y(2)-CONST2)-KS25P*Y(9)+IK22P*Y(16)
       GO TO 810
 808   IF(TT1.LT.0.5DO)GO TO 811
       ATT2=R2-Y(2)
       IF(ATT2)812,813,814
 812   C2=-KS21N*Y(1)-KS22N*(Y(2)-CONST2)-KS25N*Y(9)+IK22N*Y(16)
       GO TO 811
 813   C2=0.0DO
       GO TO 805
 814   C2=-KS21P*Y(1)-KS22P*(Y(2)-CONST2)-KS25P*Y(9)+IK22P*Y(16)
 811   IF(C2)815,805,817
 815   ACC12=1.0DO+MUS*(DTAN(KSY1))
       ACC13=DTAN(KSY1)-MUS
       PART60=-(MH1*G-((ACC3*G*DCOS(Y(2)))/LO)-(((Y(10)**2)*
     1 DSIN(Y(2))*DCOS(Y(2)))/LO)*DC-(ROW1/LO)*(0.78DO*YDOT(8)*
     1 L1*Y(4)+1.56DO*L1*Y(8)*Y(11)+0.43DO*YDOT(8)
     1 *L1*Y(5)+0.86DO*Y(8)*Y(12)*L1+0.78DO*Y(1)*YDOT(11)*L1+
```

```
     1 0.43DO*Y(1)*L1*YDOT(12)-0.43DO*((Y(8)**2)*Y(4)+Y(1)*YDOT(8)*
     1 Y(4)+Y(1)*Y(8)*Y(11))-0.68DO*((Y(8)**2)*Y(5)+Y(1)*YDOT(8)*
     1 Y(5)+Y(1)*Y(8)*Y(12))+0.57DO*(2.ODO*Y(1)*Y(8)*Y(11)+(Y(1)**2)
     1 *YDOT(11))+0.09DO*(2.ODO*Y(1)*Y(8)*Y(12)+(Y(1)**2)*YDOT(12))))
       PART61=-(-((2.ODO*MP)/LO)*(ROW4*(YDOT(11)-YDOT(12))-YDOT(8)*
     1 (Y(4)-Y(5)))+((2.ODO*Y(10)*DSIN(Y(2))*ROW1)/LO)*((0.78DO*
     1 Y(1)*L1+0.57DO*(Y(1)**2))*Y(13)+(0.43DO*Y(1)*L1+0.09DO*(
     1 Y(1)**2))*Y(14)+(0.78DO*Y(8)*L1+0.35DO*Y(1)*Y(8))*Y(6)+
     1 (0.43DO*L1*Y(8)-0.25DO*Y(1)*Y(8))*Y(7)))
       PART62=-(((4.ODO*Y(10)*DSIN(Y(2))*MP*ROW4)/LO)*(Y(13)-Y(14))
     1 +((ROW1*YDOT(10)*DSIN(Y(2)))/LO)*((0.78DO*Y(1)*L1+0.57DO*
     1 (Y(1)**2))*Y(6)+(0.43DO*Y(1)*L1+0.09DO*(Y(1)**2))*Y(7))
     1 +((2.ODO*MP*YDOT(10)*DSIN(Y(2)))/LO)*ROW4*(Y(6)-Y(7))+(
     1 ROW1/LO)*((Y(10)**2)*((DSIN(Y(2)))**2))*
     1 ((0.78DO*Y(1)*L1+0.57DO*(Y(1)**2))*Y(4)+(0.43DO*Y(1)*L1
     1 +0.09DO*(Y(1)**2))*Y(5)))
       PART63=-(((2.ODO*MP)/LO)*((Y(10)**2)*((DSIN(Y(2)))**2))
     1 *(Y(4)-Y(5))*ROW4-(ROW1/LO)*((Y(10)**2)*((
     1 DCOS(Y(2)))**2))*((0.78DO*Y(1)*L1+0.57DO*
     1 (Y(1)**2))*Y(4)+(0.43DO*Y(1)*L1+0.09DO*(Y(1)**2))*Y(5))
     1 -((2.ODO*MP*ROW4)/LO)*((Y(10)**2)*((DCOS(Y(2)))**2))
     1 *(Y(4)-Y(5))+((2.ODO*Y(10)*ROW1*DCOS(Y(2)))/LO)*(
     1 Y(1)*(Y(4)*Y(13)+Y(5)*Y(14))+Y(8)*(0.5DO*Y(4)*Y(6)+0.5DO*Y(5)
     1 *Y(7)+0.65DO*Y(5)*Y(6)-0.65DO*Y(4)*Y(7))))
       PART64=-(((8.ODO*MP*Y(10)*DCOS(Y(2)))/LO)*(Y(4)*Y(13)-Y(4)*
     1 Y(14)-Y(5)*Y(13)+Y(5)*Y(14))
     1 +((YDOT(10)*DCOS(Y(2))*ROW1)/LO)*Y(1)*
     1 (Y(4)*Y(6)+Y(5)*Y(7)))
       PART65=-(((4.ODO*YDOT(10)*DCOS(Y(2))*MP)/LO)*(Y(4)*Y(6)-Y(5)*
     1 Y(6)-Y(4)*Y(7)+Y(5)*Y(7))+(M2P/LO)*(Y(10)**2)*
     1 DSIN(Y(2))*DCOS(Y(2))*((Y(4)**2)+(Y(5)**2))+((4.ODO*MP)/LO)
     1 *(Y(10)**2)*DSIN(Y(2))*DCOS(Y(2))*((Y(4)**2)+
     1 (Y(5)**2)-2.ODO*Y(4)*Y(5)))
       TMIN2=(PART60+PART61+PART62+PART63+PART64+PART65)*(DM/2.ODO)*
     1 (ACC13/ACC12)
       IF(C2.LT.TMIN2)GO TO 810
       GO TO 805
817    ACC12=1.ODO-MUS*(DTAN(KSY1))
       ACC13=DTAN(KSY1)+MUS
       PART60=-(MH1*G-((ACC3*G*DCOS(Y(2)))/LO)-(((Y(10)**2)*
     1 DSIN(Y(2))*DCOS(Y(2)))/LO)*DC-(ROW1/LO)*(0.78DO*YDOT(8)*
     1 L1*Y(4)+1.56DO*L1*Y(8)*Y(11)+0.43DO*YDOT(8)
     1 *L1*Y(5)+0.86DO*Y(8)*Y(12)*L1+0.78DO*Y(1)*YDOT(11)*L1+
     1 0.43DO*Y(1)*L1*YDOT(12)-0.43DO*((Y(8)**2)*Y(4)+Y(1)*YDOT(8)*
     1 Y(4)+Y(1)*Y(8)*Y(11))-0.68DO*((Y(8)**2)*Y(5)+Y(1)*YDOT(8)*
     1 Y(5)+Y(1)*Y(8)*Y(12))+0.57DO*(2.ODO*Y(1)*Y(8)*Y(11)+(Y(1)**2)
     1 *YDOT(11))+0.09DO*(2.ODO*Y(1)*Y(8)*Y(12)+(Y(1)**2)*YDOT(12))))
       PART61=-(-((2.ODO*MP)/LO)*(ROW4*(YDOT(11)-YDOT(12))-YDOT(8)*
     1 (Y(4)-Y(5)))+((2.ODO*Y(10)*DSIN(Y(2))*ROW1)/LO)*((0.78DO*
     1 Y(1)*L1+0.57DO*(Y(1)**2))*Y(13)+(0.43DO*Y(1)*L1+0.09DO*(
     1 Y(1)**2))*Y(14)+(0.78DO*Y(8)*L1+0.35DO*Y(1)*Y(8))*Y(6)+
     1 (0.43DO*L1*Y(8)-0.25DO*Y(1)*Y(8))*Y(7)))
       PART62=-(((4.ODO*Y(10)*DSIN(Y(2))*MP*ROW4)/LO)*(Y(13)-Y(14))
     1 +((ROW1*YDOT(10)*DSIN(Y(2)))/LO)*((0.78DO*Y(1)*L1+0.57DO*
     1 (Y(1)**2))*Y(6)+(0.43DO*Y(1)*L1+0.09DO*(Y(1)**2))*Y(7))
     1 +((2.ODO*MP*YDOT(10)*DSIN(Y(2)))/LO)*ROW4*(Y(6)-Y(7))+(
     1 ROW1/LO)*((Y(10)**2)*((DSIN(Y(2)))**2))*
     1 ((0.78DO*Y(1)*L1+0.57DO*(Y(1)**2))*Y(4)+(0.43DO*Y(1)*L1
     1 +0.09DO*(Y(1)**2))*Y(5)))
       PART63=-(((2.ODO*MP)/LO)*((Y(10)**2)*((DSIN(Y(2)))**2))
```

```
      1 *(Y(4)-Y(5))*ROW4-(ROW1/LO)*((Y(10)**2)*((
      1 DCOS(Y(2)))**2))*((0.78DO*Y(1)*L1+0.57DO*
      1 (Y(1)**2))*Y(4)+(0.43DO*Y(1)*L1+0.09DO*(Y(1)**2))*Y(5))
      1 -((2.ODO*MP*ROW4)/LO)*((Y(10)**2)*((DCOS(Y(2)))**2))
      1 *(Y(4)-Y(5))+((2.ODO*Y(10)*ROW1*DCOS(Y(2)))/LO)*(
      1 Y(1)*(Y(4)*Y(13)+Y(5)*Y(14))+Y(8)*(0.5DO*Y(4)*Y(6)+0.5DO*Y(5)
      1 *Y(7)+0.65DO*Y(5)*Y(6)-0.65DO*Y(4)*Y(7))))
        PART64=-(((8.ODO*MP*Y(10)*DCOS(Y(2)))/LO)*(Y(4)*Y(13)-Y(4)*
      1 Y(14)-Y(5)*Y(13)+Y(5)*Y(14))
      1 +((YDOT(10)*DCOS(Y(2))*ROW1)/LO)*Y(1)*
      1 (Y(4)*Y(6)+Y(5)*Y(7)))
        PART65=-(((4.ODO*YDOT(10)*DCOS(Y(2))*MP)/LO)*(Y(4)*Y(6)-Y(5)*
      1 Y(6)-Y(4)*Y(7)+Y(5)*Y(7))+(M2P/LO)*(Y(10)**2)*
      1 DSIN(Y(2))*DCOS(Y(2))*((Y(4)**2)+(Y(5)**2))+((4.ODO*MP)/LO)
      1 *(Y(10)**2)*DSIN(Y(2))*DCOS(Y(2))*((Y(4)**2)+
      1 (Y(5)**2)-2.ODO*Y(4)*Y(5)))
        TMIN2=(PART60+PART61+PART62+PART63+PART64+PART65)*(DM/2.ODO)*
      1 (ACC13/ACC12)
        IF(C2.GT.TMIN2)GO TO 810
        GO TO 805
  810   IF(TA2.LT.0.5DO)GO TO 818
        IF(TDMAX.GE.DABS(Y(9)))GO TO 819
  818   TDMAX=DABS(Y(9))
  819   TA2=TA2+1.ODO
        TDPER=TDMAX*0.00001DO
        ERR2=DABS(R2-Y(2))
        IF((DABS(Y(9)).GT.TDPER).OR.(ERR2.GT.ER2PER))GO TO 820
  806   Y(2)=R2
        C2=0.ODO
        Y(16)=0.ODO
        TT4=1.ODO
        GO TO 805
  820   PART40=DTAN(KSY1)+MU*SGN(Y(9))
        PART41=1.ODO-MU*DTAN(KSY1)*SGN(Y(9))
        A(2,1)=((ROW1/LO)*((0.78DO*L1-0.43DO*Y(1))*Y(4)+(0.43DO*L1
      1 -0.68DO*Y(1))*Y(5))-((2.ODO*MP)/LO)*(Y(4)-Y(5)))*PART40
        A(2,2)=ACC9*LO+ACC7*LO*(1.ODO-MU*DTAN(KSY1))+(DTAN(KSY1)+MU)
      1 *((DC/LO)+(M2P/LO)*((Y(4)**2)+(Y(5)**2))+((4.ODO*MP)/LO)
      1 *((Y(4)-Y(5))**2))
        A(2,3)=(-((ROW1*DSIN(Y(2)))/LO)*((0.78DO*Y(1)*L1+0.57DO*
      1 (Y(1)**2))*Y(6)+(0.43DO*Y(1)*L1+0.09DO*(Y(1)**2))*Y(7))-((
      1 2.ODO*MP*DSIN(Y(2)))/LO)*ROW4*(Y(6)-Y(7))-((ROW1*DCOS(Y(2)))
      1 /LO)*Y(1)*(Y(4)*Y(6)+Y(5)*Y(7))-((4.ODO*MP*DCOS(Y(2)))/LO)*(
      1 Y(4)*Y(6)-Y(5)*Y(6)-Y(4)*Y(7)+Y(5)*Y(7)))*PART40
        A(2,4)=((ROW1/LO)*(0.78DO*Y(1)*L1+0.57DO*(Y(1)**2))+((2.ODO*
      1 MP*ROW4)/LO))*PART40
        A(2,5)=((ROW1/LO)*(0.43DO*Y(1)*L1+0.09DO*(Y(1)**2))-((2.ODO
      1 *MP*ROW4)/LO))*PART40
        A(2,6)=0.ODO
        A(2,7)=0.ODO
        B(2,1)=-(-(ROW1/LO)*((1.56DO*L1+0.71DO
      1 *Y(1))*Y(11)+(0.86DO*L1-0.5DO*Y(1))*Y(12)-(0.43DO*Y(4)+
      1 0.68DO*Y(5))*Y(8)))*PART40
        B(2,2)=-(-((8.ODO*MP)/LO)*(Y(4)*Y(11)-Y(4)*Y(12)+Y(5)*Y(12)-
      1 Y(5)*Y(11))-((2.ODO*ROW1)/LO)*(0.5DO*Y(8)*((Y(4)**2)+(
      1 Y(5)**2))+Y(1)*(Y(4)*Y(11)+Y(5)*Y(12)))-((2.ODO*Y(8)*BCC)/LO)
      1 )*(MU+DTAN(KSY1))
        PART50=(-((Y(10)*DSIN(Y(2))*DCOS(Y(2)))/LO)*DC+((2.ODO*ROW1*
      1 DSIN(Y(2)))/LO)*((0.78DO*Y(1)*L1+0.57DO*(Y(1)**2))*Y(13)+(
      1 0.43DO*Y(1)*L1+0.09DO*(Y(1)**2))*Y(14)+(0.78DO*Y(8)*L1+0.35DO
```

```
     1 *Y(1)*Y(8))*Y(6)+(0.43DO*Y(8)*L1-0.25DO*Y(1)*Y(8))*Y(7))+((
     1 4.0DO*MP*DSIN(Y(2))*ROW4)/LO)*(Y(13)-Y(14))+((ROW1*Y(10)*
     1 ((DSIN(Y(2)))**2))/LO)*((0.78DO*Y(1)*L1+0.57DO*(Y(1)**2))*Y(4)
     1 +(0.43DO*Y(1)*L1+0.09DO*(Y(1)**2))*Y(5))+((2.0DO*MP*Y(10)*((
     1 DSIN(Y(2)))**2))/LO)*ROW4*(Y(4)-Y(5)))*PART40
       PART51=(-((ROW1*Y(10)*((DCOS(Y(2)))**2))/LO)*((0.78DO*L1*Y(1)
     1 +0.57DO*(Y(1)**2))*Y(4)+(0.43DO*Y(1)*L1+0.09DO*(Y(1)**2))*
     1 Y(5))-((2.0DO*MP*ROW4*Y(10)*((DCOS(Y(2)))**2))/LO)*(Y(4)-Y(5))
     1 +((2.0DO*ROW1*DCOS(Y(2)))/LO)*(Y(1)*(Y(4)*Y(13)+Y(5)*Y(14))+
     1 Y(8)*(0.5DO*Y(4)*Y(6)+0.5DO*Y(5)*Y(7)+0.65DO*Y(5)*Y(6)-0.65DO
     1 *Y(4)*Y(7)))+((8.0DO*MP*DCOS(Y(2)))/LO)*(Y(4)*Y(13)-Y(4)*Y(14)
     1 -Y(5)*Y(13)+Y(5)*Y(14))+((M2P*Y(10)*DSIN(Y(2))*DCOS(Y(2)))/LO)
     1 *((Y(4)**2)+(Y(5)**2))+((4.0DO*MP*Y(10)*DSIN(Y(2))*DCOS(Y(2)))
     1 /LO)*((Y(4)-Y(5))**2))*PART40
       B(2,3)=-PART50-PART51
       B(2,4)=0.0DO
       B(2,5)=0.0DO
       B(2,6)=0.0DO
       B(2,7)=0.0DO
       F(2)=(2.0DO*PART41*C2)/DM
       D(2)=(-((ACC3*G*DCOS(Y(2)))/LO)+MH1*G)*PART40
       GO TO 721
805    Y(9)=0.0DO
       A(2,1)=0.0DO
       A(2,2)=1.0DO
       A(2,3)=0.0DO
       A(2,4)=0.0DO
       A(2,5)=0.0DO
       A(2,6)=0.0DO
       A(2,7)=0.0DO
       B(2,1)=0.0DO
       B(2,2)=0.0DO
       B(2,3)=0.0DO
       B(2,4)=0.0DO
       B(2,5)=0.0DO
       B(2,6)=0.0DO
       B(2,7)=0.0DO
       F(2)=0.0DO
       D(2)=0.0DO
721    C3=-KS33*(Y(3)-CONST3)-KS36*Y(10)+IK33*Y(17)
       A(3,1)=ROW1*(Y(1)*DCOS(Y(2))*(0.78DO*Y(6)+
     1 0.43DO*Y(7))-1.3DO*Y(1)*DSIN(Y(2))*(Y(4)*Y(7)-Y(5)*Y(6))-
     1 L1*DCOS(Y(2))*(0.78DO*Y(6)+0.43DO*Y(7))-
     1 Y(1)*DCOS(Y(2))*(0.35DO*Y(6)-0.25DO*Y(7)))+MP*2.0DO*
     1 DCOS(Y(2))*(Y(6)-Y(7))
       A(3,2)=ROW1*(-Y(1)*DCOS(Y(2))*(Y(4)*Y(6)+Y(5)*Y(7))
     1 -L1*Y(1)*DSIN(Y(2))*(0.78DO*Y(6)+0.43DO*Y(7))-
     1 (Y(1)**2)*DSIN(Y(2))*(0.57DO*Y(6)+0.09DO*Y(7)))+MP*
     1 (-4.0DO*DCOS(Y(2))*(Y(4)*Y(6)-Y(4)*Y(7)-Y(5)*Y(6)+
     1 Y(5)*Y(7))-2.0DO*(L1+Y(1))*DSIN(Y(2))*(Y(6)-Y(7)))
       A(3,3)=IL2+MH1*(LO**2)+(INERTI+MH1*((L1+Y(1)-L2)**2)+
     1 ((M1*LT**2)/12.0DO)+((((LT/2.0DO)-LO)**2)*M1))*
     1 ((DCOS(Y(2)))**2)+ROW1*ROW3*(ROW2**2)*((DCOS(Y(2)))**2)+
     1 (ROW1*(ROW3**3)*((DCOS(Y(2)))**2))/12.0DO+ROW1*(Y(1)*
     1 ((DCOS(Y(2)))**2)*(Y(6)**2+Y(7)**2)+Y(1)*((DSIN(Y(2)))**2)
     1 *(Y(6)**2+Y(7)**2)+Y(1)*((DSIN(Y(2)))**2)*(Y(4)**2+Y(5)**2)
     1 -2.0DO*L1*DSIN(Y(2))*DCOS(Y(2))*Y(1)*(0.78DO*Y(4)+0.43DO*
     1 Y(5))-2.0DO*DSIN(Y(2))*DCOS(Y(2))*(Y(1)**2)*(0.57DO*
     1 Y(4)+0.09DO*Y(5))+ROW5*((DCOS(Y(2)))**2)+MP*(4.0DO*
     1 ((DCOS(Y(2)))**2)*(Y(6)**2+Y(7)**2-2.0DO*Y(6)*Y(7))+
```

```
     1 4.0D0*((DSIN(Y(2)))**2)*(Y(6)**2+Y(7)**2-2.0D0*Y(6)*
     1 Y(7)+Y(4)**2+Y(5)**2-2.0D0*Y(4)*Y(5))-4.0D0*DSIN(Y(2))*
     1 DCOS(Y(2))*ROW4*(Y(4)-Y(5))+(ROW4**2)*((DCOS(Y(2)))**2))
       A(3,4)=-ROW1*Y(1)*DSIN(Y(2))*Y(6)-MP*4.0D0*DSIN(Y(2))*
     1 (Y(6)-Y(7))
       A(3,5)=-ROW1*Y(1)*DSIN(Y(2))*Y(7)-MP*4.0D0*DSIN(Y(2))*(Y(7)
     1 -Y(6))
       A(3,6)=ROW1*(Y(1)*DSIN(Y(2))*Y(4)-L1*DCOS(Y(2))*0.78D0*
     1 Y(1)-Y(1)**2*DCOS(Y(2))*0.57D0)+MP*(4.0D0*DSIN(Y(2))*(Y(4)
     1 -Y(5))-2.0D0*ROW4*DCOS(Y(2)))
       A(3,7)=ROW1*(Y(1)*DSIN(Y(2))*Y(5)-L1*
     1 DCOS(Y(2))*0.43D0*Y(1)-(Y(1)**2)*DCOS(Y(2))
     1 *0.09D0)+MP*(4.0D0*DSIN(Y(2))*(Y(5)-Y(4))+2.0D0
     1 *ROW4*DCOS(Y(2)))
       A(4,1)=ROW1*(Y(1)/2.0D0)*(Y(4)-1.3D0*Y(5))
       A(4,2)=ROW6*(1.56D0*L1*Y(1)+1.14D0*(Y(1)**2))
     1 +MP*(2.0D0*ROW4)
       A(4,3)=-ROW1*DSIN(Y(2))*Y(1)*Y(6)-MP*4.0D0
     1 *DSIN(Y(2))*(Y(6)-Y(7))
       A(4,4)=ROW1*Y(1)+4.0D0*MP
       A(4,5)=-4.0D0*MP
       A(4,6)=0.0D0
       A(4,7)=0.0D0
       A(5,1)=ROW1*Y(1)*(0.65D0*Y(4)+0.5D0*Y(5))
       A(5,2)=ROW1*(0.43D0*L1*Y(1)+0.09D0*(Y(1)**2)).
     1 -MP*2.0D0*ROW4
       A(5,3)=-ROW1*DSIN(Y(2))*Y(1)*Y(7)-4.0D0*MP*
     1 DSIN(Y(2))*(Y(7)-Y(6))
       A(5,4)=-4.0D0*MP
       A(5,5)=ROW1*Y(1)+4.0D0*MP
       A(5,6)=0.0D0
       A(5,7)=0.0D0
       A(6,1)=ROW1*Y(1)*(0.5D0*Y(6)-0.65D0*Y(7))
       A(6,2)=0.0D0
       A(6,3)=ROW1*(Y(1)*DSIN(Y(2))*Y(4)-0.78D0*Y(1)*
     1 DCOS(Y(2))*L1-0.57D0*(Y(1)**2)*DCOS(Y(2)))+MP*(4.0D0*
     1 DSIN(Y(2))*(Y(4)-Y(5))-2.0D0*ROW4*DCOS(Y(2)))
       A(6,4)=0.0D0
       A(6,5)=0.0D0
       A(6,6)=ROW1*Y(1)+4.0D0*MP
       A(6,7)=-MP*4.0D0
       A(7,1)=ROW1*Y(1)*(0.65D0*Y(6)+0.5D0*Y(7))
       A(7,2)=0.0D0
       A(7,3)=ROW1*(Y(1)*DSIN(Y(2))*Y(5)-0.43D0*L1*Y(1)*
     1 DCOS(Y(2))-0.09D0*(Y(1)**2)*DCOS(Y(2)))+MP*(4.0D0*DSIN(Y(2))
     1 *(Y(5)-Y(4))+2.0D0*ROW4*DCOS(Y(2)))
       A(7,4)=0.0D0
       A(7,5)=0.0D0
       A(7,6)=-4.0D0*MP
       A(7,7)=ROW1*Y(1)+4.0D0*MP
C
C
C            ***  COMPUTATION OF THE CORIOLIS, CENTRIPETAL,   ***
C            ***  GRAVITY AND CONTROL TERMS.                  ***
C
C
       PART1=-ROW1*Y(10)*(ROW2**2)*((DCOS(Y(2)))**2)
     1 +ROW1*ROW3*Y(10)*ROW2*((DCOS(Y(2)))**2)-((ROW1*
     1 (ROW3**2))/4.0D0)*Y(10)*((DCOS(Y(2)))**2)+ROW1*
     1 (Y(10)*((DCOS(Y(2)))**2)*(Y(6)**2
```

```
     1 +Y(7)**2)+(Y(8)*DCOS(Y(2))-Y(1)*Y(9)*DSIN(Y(2)))
     1 *(0.78DO*Y(6)+0.43DO*Y(7))+Y(1)*DCOS(Y(2))*
     1 (0.78DO*Y(13)+0.43DO*Y(14))-Y(9)*DCOS(Y(2))*(Y(4)*Y(6)+
     1 Y(5)*Y(7))-DSIN(Y(2))*Y(1)*(0.50DO*(Y(11)*Y(6)+
     1 Y(4)*Y(13))+0.65DO*(Y(11)*Y(7)+Y(4)*Y(14))
     1 -0.65DO*(Y(12)*Y(6)+Y(5)*Y(13))+0.5DO*(Y(12)*Y(7)+
     1 Y(5)*Y(14))))
       PART2=ROW1*(-DSIN(Y(2))*(Y(11)*Y(6)+Y(12)*Y(7))
     1 -L1*Y(9)*DSIN(Y(2))*(0.78DO*Y(6)+0.43DO*Y(7))
     1 -2.ODO*Y(1)*Y(9)*DSIN(Y(2))*(0.57DO*Y(6)+0.09DO*Y(7))
     1 +Y(10)*((DSIN(Y(2)))**2)*(Y(6)**2+Y(7)**2)+
     1 (Y(8)*DSIN(Y(2))+Y(1)*Y(9)*DCOS(Y(2)))*
     1 (1.3DO*Y(6)*Y(5)-1.3DO*Y(7)*Y(4)))
       PART3=ROW1*(DSIN(Y(2))*Y(1)*(0.5DO*(Y(13)*Y(4)+
     1 Y(6)*Y(11))+0.65DO*(Y(13)*Y(5)+Y(6)*Y(12))-
     1 0.65DO*(Y(14)*Y(4)+Y(7)*Y(11))+0.5DO*(Y(14)*Y(5)
     1 +Y(7)*Y(12)))+DSIN(Y(2))*(Y(13)*Y(4)+
     1 Y(14)*Y(5))+L1*Y(9)*DSIN(Y(2))*(0.78DO*Y(6)+
     1 0.43DO*Y(7))-L1*DCOS(Y(2))*(1.56DO*Y(13)+
     1 0.86DO*Y(14)))
       PART4=ROW1*(-(Y(8)*DCOS(Y(2))-Y(1)*Y(9)*DSIN(Y(2)))*(0.35DO*
     1 Y(6)-0.25DO*Y(7))-DCOS(Y(2))*Y(1)*(0.35DO*
     1 Y(13)-0.25DO*Y(14))-2.ODO*Y(1)*DCOS(Y(2))*
     1 (0.57DO*Y(13)+0.09DO*Y(14))+Y(10)*((DSIN(Y(2)))
     1 **2)*(Y(4)**2+Y(5)**2)-2.ODO*L1*Y(10)*DSIN(Y(2))
     1 *DCOS(Y(2))*(0.78DO*Y(4)+0.43DO*Y(5))-4.ODO*Y(1)*Y(10)
     1 *DSIN(Y(2))*DCOS(Y(2))*(0.57DO*Y(4)+0.09DO*Y(5)))
       PART5=ROW1*(Y(10)*((DCOS(Y(2)))**2)*((L1**2)+
     1 2.ODO*L1*Y(1)+(Y(1)**2)))+MP*(-4.ODO*Y(9)*DSIN(Y(2))*
     1 (Y(6)-Y(7))-4.ODO*Y(10)*DCOS(Y(2))*DSIN(Y(2))*
     1 (Y(4)-Y(5))+2.ODO*ROW4*Y(10)*((DCOS(Y(2)))**2))
       B(3,1)=PART1+PART2+PART3+PART4+PART5+2.ODO*Y(10)
     1 *((DCOS(Y(2)))**2)*MH1*(L1+Y(1)-L2)
       PART6=-2.ODO*(((M1*(LT**2))/12.ODO)+M1*(((LT/2.ODO)
     1 -LO)**2))*Y(10)*DCOS(Y(2))*
     1 DSIN(Y(2))-2.ODO*ROW1*ROW3*Y(10)*(ROW2**2)*
     1 DSIN(Y(2))*DCOS(Y(2))-(ROW1*(ROW3**3)*Y(10)*DSIN(Y(2))
     1 *DCOS(Y(2)))/6.ODO+ROW1*(-2.ODO*Y(10)*Y(1)*DCOS(Y(2))*
     1 DSIN(Y(2))*(Y(6)**2+Y(7)**2)+Y(9)*Y(1)*DSIN(Y(2))*
     1 (Y(4)*Y(6)+Y(5)*Y(7))-Y(1)*DCOS(Y(2))*(Y(11)*Y(6)+
     1 Y(4)*Y(13)+Y(12)*Y(7)+Y(5)*Y(14))-Y(1)*DCOS(Y(2))*
     1 (Y(11)*Y(6)+Y(12)*Y(7)))
       PART7=ROW1*(-L1*Y(1)*Y(9)*DCOS(Y(2))*
     1 (0.78DO*Y(6)+0.43DO*Y(7))-L1*Y(1)*DSIN(Y(2))*
     1 (0.78DO*Y(13)+0.43DO*Y(14))-(Y(1)**2)*Y(9)*DCOS(Y(2))*
     1 (0.57DO*Y(6)+0.09DO*Y(7))-DSIN(Y(2))*(Y(1)**2)*
     1 (0.57DO*Y(13)+0.09DO*Y(14))+2.ODO*Y(10)*Y(1)*
     1 DSIN(Y(2))*DCOS(Y(2))*(Y(6)**2+Y(7)**2)
     1 +Y(1)*DCOS(Y(2))*(Y(13)*Y(4)+Y(14)*Y(5)))
       PART8=ROW1*(L1*DSIN(Y(2))*(0.78DO*Y(1)*Y(13)+
     1 0.43DO*Y(1)*Y(14))+(Y(1)**2)*DSIN(Y(2))*(0.57DO*Y(13)+
     1 0.09DO*Y(14))+2.ODO*DSIN(Y(2))*DCOS(Y(2))*Y(10)*
     1 Y(1)*(Y(4)**2+Y(5)**2)
     1 -2.ODO*L1*(Y(10)*((DCOS(Y(2)))**2)*Y(1)
     1 -Y(10)*((DSIN(Y(2)))**2)*Y(1))*(0.78DO*
     1 Y(4)+0.43DO*Y(5))-2.ODO*Y(10)*(Y(1)**2)*
     1 (((DCOS(Y(2)))**2)-((DSIN(Y(2)))**2))*(0.57DO*Y(4)
     1 +0.09DO*Y(5)))
       PART9=ROW1*(-2.ODO*Y(10)*DSIN(Y(2))*DCOS(Y(2))
     1 *ROW5)+MP*(Y(10)*DCOS(Y(2))*DSIN(Y(2))*(8.ODO*
```

```
1 (Y(4)**2+Y(5)**2-2.ODO*Y(4)*Y(5))-2.ODO*(ROW4**2))
1 +4.ODO*Y(9)*DSIN(Y(2))*(Y(4)*Y(6)-Y(4)*Y(7)-
1 Y(5)*Y(6)+Y(5)*Y(7))-4.ODO*DCOS(Y(2))*(2.ODO*Y(11)*
1 Y(6)+Y(4)*Y(13)-2.ODO*Y(11)*Y(7)-Y(4)*Y(14)
1 -2.ODO*Y(12)*Y(6)-Y(5)*Y(13)+2.ODO*Y(12)*Y(7)+
1 Y(5)*Y(14))-2.ODO*Y(9)*ROW4*DCOS(Y(2))*(Y(6)-Y(7))+
1 4.ODO*DCOS(Y(2))*(Y(13)*Y(4)-Y(13)*Y(5)-Y(14)*Y(4)+
1 Y(14)*Y(5))-(Y(10)*((DCOS(Y(2)))**2)*ROW4-Y(10)*
1 ((DSIN(Y(2)))**2)*ROW4)*(Y(4)-Y(5))*4.ODO)
  B(3,2)=PART6+PART7+PART8+PART9-2.ODO*Y(10)*
1 DCOS(Y(2))*DSIN(Y(2))*(INERTI+MH1*((L1+Y(1)-L2)**2))
  B(3,3)=ROW1*(Y(1)*2.ODO*(Y(6)*Y(13)+Y(7)*Y(14))+
1 Y(1)*((DSIN(Y(2)))**2)*2.ODO*(Y(4)*Y(11)+Y(5)*
1 Y(12))-2.ODO*Y(1)*DSIN(Y(2))*DCOS(Y(2))*(0.78DO*
1 L1*Y(11)+0.43DO*L1*Y(12)+0.57DO*Y(1)*Y(11)+0.09DO*
1 Y(1)*Y(12)))+MP*(8.ODO*(Y(6)*Y(13)+Y(7)*Y(14)-Y(13)
1 *Y(7)-Y(6)*Y(14))+8.ODO*((DSIN(Y(2)))**2)*(Y(4)*Y(11)+
1 Y(5)*Y(12)-Y(11)*Y(5)-Y(4)*Y(12))-4.ODO*DSIN(Y(2))*
1 DCOS(Y(2))*ROW4*(Y(11)-Y(12)))
  B(3,4)=0.ODO
  B(3,5)=0.ODO
  B(3,6)=0.ODO
  B(3,7)=0.ODO
  B(4,1)=ROW6*(Y(8)*(Y(4)-1.3DO*Y(5))+Y(9)*Y(1)*
1 3.14DO-2.ODO*Y(10)*DSIN(Y(2))*Y(6)-2.ODO*Y(8)*
1 (A13*Y(4)+A14*Y(5))+2.6DO*Y(10)*DSIN(Y(2))*
1 Y(1)*Y(7))+MP*4.ODO*Y(9)
  B(4,2)=-ROW1*Y(9)*Y(1)*Y(4)-4.ODO*MP*Y(9)*(Y(4)
1 -Y(5))
  B(4,3)=ROW6*(-2.ODO*Y(10)*((DSIN(Y(2)))**2)*Y(1)
1 *Y(4)+Y(10)*DSIN(Y(2))*DCOS(Y(2))*Y(1)*(1.56DO*
1 L1+1.14DO*Y(1)))+MP*(-4.ODO*Y(10)*((DSIN(Y(2)))**2)
1 *(Y(4)-Y(5))+2.ODO*Y(10)*DSIN(Y(2))*DCOS(Y(2))
1 *ROW4)
  B(4,4)=ROW6*2.ODO*Y(8)
  B(4,5)=-ROW6*2.6DO*Y(1)*Y(8)
  B(4,6)=-ROW6*Y(10)*4.ODO*DSIN(Y(2))*Y(1)-8.ODO
1 *MP*Y(10)*DSIN(Y(2))
  B(4,7)=8.ODO*MP*Y(10)*DSIN(Y(2))
  B(5,1)=ROW1*(Y(8)*(0.65DO*Y(4)+0.5DO*Y(5))
1 +0.86DO*Y(9)*Y(1)+Y(10)*DSIN(Y(2))*Y(7)-
1 Y(8)*(A14*Y(4)+A15*Y(5))-1.30DO*Y(10)*
1 DSIN(Y(2))*Y(1)*Y(6))-MP*4.ODO*Y(9)
  B(5,2)=ROW1*(-Y(9)*Y(1)*Y(5))-4.ODO*MP
1 *Y(9)*(Y(5)-Y(4))
  B(5,3)=ROW1*(-Y(1)*Y(10)*((DSIN(Y(2)))**2)
1 *Y(5)+Y(10)*DSIN(Y(2))*DCOS(Y(2))*Y(1)*
1 (0.43DO*L1+0.09DO*Y(1)))+MP*(-4.ODO*
1 Y(10)*((DSIN(Y(2)))**2)*(Y(5)-Y(4))-2.ODO*
1 Y(10)*DSIN(Y(2))*DCOS(Y(2))*ROW4)
  B(5,4)=ROW1*1.3DO*Y(1)*Y(8)
  B(5,5)=ROW1*Y(8)
  B(5,6)=MP*8.ODO*Y(10)*DSIN(Y(2))
  B(5,7)=-ROW1*2.ODO*Y(10)*DSIN(Y(2))*Y(1)
1 -8.ODO*MP*Y(10)*DSIN(Y(2))
  B(6,1)=ROW1*(Y(8)*(0.5DO*Y(6)-0.65DO*Y(7))
1 +Y(10)*DSIN(Y(2))*Y(4)+Y(10)*DCOS(Y(2))
1 *(-1.57DO*Y(1))+Y(10)*DSIN(Y(2))*Y(1)*
1 (-1.3DO*Y(5))-Y(8)*(A13*Y(6)+A14*Y(7)))
1 +MP*(-4.ODO*Y(10)*DCOS(Y(2)))
```

```
      B(6,2)=ROW1*(2.0D0*Y(1)*Y(10)*DCOS(Y(2))*Y(4)+
     1 1.56D0*Y(1)*L1*DSIN(Y(2))*Y(10)+1.14D0*(Y(1)**2)
     1 *Y(10)*DSIN(Y(2)))+MP*(8.0D0*Y(10)*DCOS(Y(2))
     1 *(Y(4)-Y(5))+ROW4*4.0D0*Y(10)*DSIN(Y(2)))
      B(6,3)=-ROW1*Y(1)*Y(10)*Y(6)-4.0D0*MP*Y(10)
     1 *(Y(6)-Y(7))
      B(6,4)=2.0D0*ROW1*Y(10)*DSIN(Y(2))*Y(1)+8.0D0
     1 *MP*Y(10)*DSIN(Y(2))
      B(6,5)=-8.0D0*MP*Y(10)*DSIN(Y(2))
      B(6,6)=ROW1*Y(8)
      B(6,7)=-ROW1*1.3D0*Y(1)*Y(8)
      B(7,1)=ROW1*(Y(8)*(0.65D0*Y(6)+0.5D0*Y(7))+Y(10)
     1 *DSIN(Y(2))*Y(5)-0.86D0*Y(1)*Y(10)*DCOS(Y(2))
     1 -Y(8)*(A14*Y(6)+A15*Y(7))+1.3D0*Y(10)*DSIN(Y(2))
     1 *Y(1)*Y(4))
     1 +MP*4.0D0*Y(10)*DCOS(Y(2))
      B(7,2)=ROW1*(2.0D0*Y(1)*Y(10)*DCOS(Y(2))*Y(5)+
     1 0.18D0*(Y(1)**2)*Y(10)*DSIN(Y(2))+0.86D0*L1*Y(1)
     1 *Y(10)*DSIN(Y(2)))+MP*(8.0D0*Y(10)*DCOS(Y(2))*
     1 (Y(5)-Y(4))-4.0D0*ROW4*Y(10)*DSIN(Y(2)))
      B(7,3)=-ROW1*Y(1)*Y(10)*Y(7)-4.0D0*MP*Y(10)*
     1 (Y(7)-Y(6))
      B(7,4)=-MP*8.0D0*Y(10)*DSIN(Y(2))
      B(7,5)=ROW1*2.0D0*Y(10)*DSIN(Y(2))*Y(1)+8.0D0
     1 *MP*Y(10)*DSIN(Y(2))
      B(7,6)=ROW1*1.3D0*Y(1)*Y(8)
      B(7,7)=ROW1*Y(8)
      F(3)=C3
      F(4)=0.0D0
      F(5)=0.0D0
      F(6)=0.0D0
      F(7)=0.0D0
      D(3)=0.0D0
      D(4)=-ROW1*Y(1)*G*DCOS(Y(2))*AL1-2.0D0*MP*G
     1 *DCOS(Y(2))-(EI*Y(4)*AL10)/(Y(1)**3)
     1 -((ROW1*CENTAC*DIST*(4.65D0*Y(4)-7.38D0*Y(5)))/Y(1))+
     1 ROW1*CENTAC*((3.08D0*L1+1.13D0*Y(1))*Y(4)-(6.96D0*L1
     1 +3.0D0*Y(1))*Y(5))-(((ROW7*CENTAC+ROW8)*(4.65D0*Y(4)-
     1 7.38D0*Y(5)))/Y(1))+ROW9*(3.08D0*Y(4)-6.96D0*Y(5))
      D(5)=-ROW1*Y(1)*G*DCOS(Y(2))*AL2+2.0D0*MP
     1 *G*DCOS(Y(2))-(EI*Y(5)*AL11)/(Y(1)**3)
     1 -((ROW1*CENTAC*DIST*(-7.38D0*Y(4)+32.42D0*Y(5)))/Y(1))+
     1 ROW1*CENTAC*(-(6.96D0*L1+3.0D0*Y(1))*Y(4)+(23.77D0*L1
     1 +9.73D0*Y(1))*Y(5))-(((ROW7*CENTAC+ROW8)*(-7.38D0*Y(4)+
     1 32.42D0*Y(5)))/Y(1))+ROW9*(-6.96D0*Y(4)+23.77D0*Y(5))
      D(6)=-(AL10*EI*Y(6))/(Y(1)**3)-((ROW1*CENTAC*DIST*(4.65D0*Y(6)
     1 -7.38D0*Y(7)))/Y(1))+ROW1*CENTAC*((3.08D0*L1+1.13D0*Y(1))
     1 *Y(6)-(6.96D0*L1+3.0D0*Y(1))*Y(7))-(((ROW7*CENTAC
     1 +ROW8)*(4.65D0*Y(6)-7.38D0*Y(7)))/Y(1))+ROW9*(
     1 3.08D0*Y(6)-6.96D0*Y(7))
      D(7)=-(AL11*EI*Y(7))/(Y(1)**3)-((ROW1*CENTAC*DIST*(-7.38D0*
     1 Y(6)+32.42D0*Y(7)))/Y(1))+ROW1*CENTAC*(-(6.96D0*L1
     1 +3.0D0*Y(1))*Y(6)+(23.77D0*L1+9.73D0*Y(1))*Y(7))
     1 -(((ROW7*CENTAC+ROW8)*(-7.38D0*Y(6)+32.42D0*Y(7)))
     1 /Y(1))+ROW9*(-6.96D0*Y(6)+23.77D0*Y(7))
      YVECT(1)=Y(8)
      YVECT(2)=Y(9)
      YVECT(3)=Y(10)
      YVECT(4)=Y(11)
      YVECT(5)=Y(12)
```

```
      YVECT(6)=Y(13)
      YVECT(7)=Y(14)
C
C
C             ***  COMBINING THE TERMS ON THE LEFT HAND SIDE TO  ***
C             ***  ONE SINGLE VECTOR.                            ***
C
C
      DO 60 IN=1,M
      PBY(IN)=0.0D0
         DO 60 IM=1,M
            PBY(IN)=PBY(IN)+(-B(IN,IM))*YVECT(IM)
 60   CONTINUE
      DO 61 IY=1,M
         PALL(IY)=0.0D0
         PALL(IY)=PBY(IY)+F(IY)+D(IY)
 61   CONTINUE
C
C
C          ***  LU DECOMPOSITION OF THE (7,7) INERTIA MATRIX.  ***
C
C
      CALL DLUD(M,7,A,7,T,IV)
      IF(IV(M).EQ.0) GO TO 90
      GO TO 91
 90   S=0.0D0
      GO TO 14
 91   S=1.0D0
C
C        ***  SOLVING FOR THE ACCELERATION TERMS.  ***
C
      CALL DBS(M,7,T,IV,PALL)
C
C      ***  THE EQUATIONS OF MOTION IN FIRST ORDER FORM.  ***
C
      YDOT(1)=Y(8)
      YDOT(2)=Y(9)
      YDOT(3)=Y(10)
      YDOT(4)=Y(11)
      YDOT(5)=Y(12)
      YDOT(6)=Y(13)
      YDOT(7)=Y(14)
      YDOT(8)=PALL(1)
      YDOT(9)=PALL(2)
      YDOT(10)=PALL(3)
      YDOT(11)=PALL(4)
      YDOT(12)=PALL(5)
      YDOT(13)=PALL(6)
      YDOT(14)=PALL(7)
      YDOT(15)=R1-Y(1)
      YDOT(16)=R2-Y(2)
      YDOT(17)=R3-Y(3)
      TT1=TT1+1.0D0
 14   RETURN
      END
C
C
C
C*****************************************************
C
```

```
C
C
      SUBROUTINE PEDERV (N,T,Y,PD,NO)
      RETURN
      END
      SUBROUTINE PLTF(I,Y1,Y2,Y3,Y4,Y5,Y6,Y7,Y8,Y9,Y10,Y11,
     1 Y12,Y13,Y14,Y15,Y16,Y17,D1,D2,TI)
C
C
C               ***  SUBROUTINE IDENTIFICATION ***
C
C
C       THIS SUBROUTINE IS WRITTEN TO MAKE TWO DIMENSIONAL CURVES.
C
C
      REAL*4 XBOX(50),YBOX(50),H1,H2,Y1(2010),Y2(2010)
      REAL*4 Y3(2010),Y4(2010),Y5(2010),Y6(2010),Y7(2010),Y8(2010)
      REAL*4 Y9(2010),Y10(2010),Y11(2010),Y12(2010),Y13(2010)
      REAL*4 Y15(2010),Y16(2010),Y17(2010),Y14(2010)
      REAL*4 TI(2010),X1(100),X2(100),X3(100),X4(100),X5(100)
      REAL*4 D1(2010),D2(2010),X6(100),X7(100)
      REAL*4 X8(100),X9(100),X10(100),X11(100),X12(100),X13(100)
      REAL*4 X14(100),X15(100),Z1,Z2,Z3,Z4,Z5,Z6,Z7,Z8
      ICOUNT=1
C
C       DATA FOR DRAWING THE BOX THAT WILL CONTAIN FOUR PLOTS.
C
      DATA XBOX/0.0,8.5,8.5,0.0,0.0/,YBOX/0.0,0.0,11.0,11.0,0.0/
      DATA H1,H2/0.12,0.1/
C
C       READ THE COORDINATES OF THE ORIGIN FOR THE FOUR PLOTS.
C
      READ(8,45)Z1,Z2,Z3,Z4,Z5,Z6,Z7,Z8
 45   FORMAT(8F5.3)
C
C       HEAD TITLES OF THE PLOTS.
C
      READ(8,44)(X1(IL),IL=1,20)
      READ(8,44)(X2(IL),IL=1,20)
      READ(8,44)(X3(IL),IL=1,20)
C
C       SET THE LETTERING SIZE FOR THE TITLE ON A LINEAR AXIS.
C
      CALL PLOTS
      CALL PAXTTL(H1)
C
C
C       SET THE LETTERING SIZE FOR THE NUMBERING ON A LINEAR SCALE.
C
C
      CALL PAXVAL(H2)
C
C
C       DRAW THE CONTOUR WHICH SURROUNDS THE FOUR FIGURES.
C
 489  CALL PLINE(XBOX,YBOX,5,1,0,0,0)
C
C       READ THE TITLES OF THE PLOTS.  NOTE THAT THE LATTERS ARE
C       UPDATED EVERY TIME FOUR PLOTS ARE DRAWN.
C
```

```
C
      READ(8,44)(X4(IL),IL=1,20)
      READ(8,44)(X5(IL),IL=1,20)
      READ(8,44)(X6(IL),IL=1,20)
      READ(8,44)(X7(IL),IL=1,20)
      READ(8,44)(X8(IL),IL=1,20)
      READ(8,44)(X9(IL),IL=1,20)
      READ(8,44)(X10(IL),IL=1,20)
      READ(8,44)(X11(IL),IL=1,20)
      READ(8,44)(X12(IL),IL=1,20)
      IF(ICOUNT.GT.4)GO TO 371
      READ(8,44)(X13(IL),IL=1,20)
      READ(8,44)(X14(IL),IL=1,20)
      READ(8,44)(X15(IL),IL=1,20)
 371  CONTINUE
 44   FORMAT(20A4)
C
C      WRITE UP OF THE OVERALL TITLE OF THE RUN.
C
      CALL PSYM(1.05,10.5,0.25,X1,0.0,40,0)
      CALL PSYM(1.75,10.15,0.25,X2,0.0,40,0)
      CALL PSYM(1.05,9.8,0.25,X3,0.0,40,0)
C
C
C      PLOTTING THE CURVES.
C
C
      IF(ICOUNT.GT.1)GO TO 500
      CALL PLTCUR(I,TI,Y1,X4,Z1,Z2)
      CALL PLTCUR(I,TI,Y2,X5,Z3,Z4)
      CALL PLTCUR(I,TI,Y3,X6,Z5,Z6)
      CALL PLTCUR(I,TI,Y4,X7,Z7,Z8)
      GO TO 503
C
C
 500  IF(ICOUNT.GT.2)GO TO 501
      CALL PLTCUR(I,TI,Y5,X4,Z1,Z2)
      CALL PLTCUR(I,TI,Y6,X5,Z3,Z4)
      CALL PLTCUR(I,TI,Y7,X6,Z5,Z6)
      CALL PLTCUR(I,TI,Y8,X7,Z7,Z8)
      GO TO 503
C
C
 501  IF(ICOUNT.GT.3)GO TO 502
      CALL PLTCUR(I,TI,Y9,X4,Z1,Z2)
      CALL PLTCUR(I,TI,Y10,X5,Z3,Z4)
      CALL PLTCUR(I,TI,Y11,X6,Z5,Z6)
      CALL PLTCUR(I,TI,Y12,X7,Z7,Z8)
      GO TO 503
C
C
 502  IF(ICOUNT.GT.4)GO TO 504
      CALL PLTCUR(I,TI,Y13,X4,Z1,Z2)
      CALL PLTCUR(I,TI,Y14,X5,Z3,Z4)
      CALL PLTCUR(I,TI,Y15,X6,Z5,Z6)
      CALL PLTCUR(I,TI,Y16,X7,Z7,Z8)
      GO TO 503
C
C
 504  CALL PLTCUR(I,TI,Y17,X4,Z1,Z2)
```

```
      CALL PLTCUR(I,TI,D1,X5,Z3,Z4)
      CALL PLTCUR(I,TI,D2,X6,Z5,Z6)
      CALL PSYM(.75,5.275,0.15,X7,0.0,40,0)
      CALL PSYM(.75,5.025,0.15,X8,0.0,40,0)
      CALL PSYM(4.75,5.275,0.15,X9,0.0,40,0)
      CALL PSYM(4.75,5.025,0.15,X10,0.0,40,0)
      CALL PSYM(.75,0.9,0.15,X11,0.0,40,0)
      CALL PSYM(.75,0.65,0.15,X12,0.0,40,0)
      CALL PLTEND
      IF(ICOUNT.EQ.5)GO TO 511
C
C
 503  CALL PSYM(.75,5.275,0.15,X8,0.0,40,0)
      CALL PSYM(.75,5.025,0.15,X9,0.0,40,0)
      CALL PSYM(4.75,5.275,0.15,X10,0.0,40,0)
      CALL PSYM(4.75,5.025,0.15,X11,0.0,40,0)
      CALL PSYM(.75,0.9,0.15,X12,0.0,40,0)
      CALL PSYM(.75,0.65,0.15,X13,0.0,40,0)
      CALL PSYM(4.75,0.9,0.15,X14,0.0,40,0)
      CALL PSYM(4.75,0.65,0.15,X15,0.0,40,0)
      CALL PLTEND
 511  ICOUNT=ICOUNT+1
C
C
      IF(ICOUNT.LE.5)GO TO 489
      RETURN
      END
C
C
C       THIS SUBROUTINE MAKES INDIVIDUAL PLOTTING.
C
C
      SUBROUTINE PLTCUR(I,TI,Y1,X1,Z1,Z2)
C
      REAL*4 Z1,Z2,TI(220),Y1(220),X1(100)
C
C
      CALL PSCALE(3.0,0.5,TIMIN,TIF,TI,I,1)
      CALL PSCALE(3.0,0.5,Y1MIN,Y1F,Y1,I,1)
      CALL PLTOFS(TIMIN,TIF,Y1MIN,Y1F,Z1,Z2)
      CALL PLINE(TI,Y1,I,1,0,0,1)
      CALL PAXIS(Z1,Z2,'TIME, t  [SECOND]',-17,3.0,
     1 0.0,TIMIN,TIF,0.5)
      CALL PAXIS(Z1,Z2,X1,30,3.0,90.0,Y1MIN,Y1F,0.5)
C
      RETURN
      END
C
C
C       THIS SUBROUTINE MAKES 2 PLOTS ON THE SAME PLOTTING SPACE.
C
C
      SUBROUTINE PLTDCU(I,TI,Y1,Y2,X1,Z1,Z2)
C
C
      REAL*4 Z1,Z2,TI(220),Y1(220),Y2(220),X1(100)
C
C
      CALL PSCALE(3.0,0.5,TIMIN,TIF,TI,I,1)
      CALL PSCALE(3.0,0.5,Y1MIN,Y1F,Y1,I,1,Y2,I,1)
```

```
      CALL PLTOFS(TIMIN,TIF,Y1MIN,Y1F,Z1,Z2)
      CALL PLINE(TI,Y1,I,1,0,0,1)
      CALL PDSHLN(TI,Y2,I,1,0.05,1)
      CALL PAXIS(Z1,Z2,'TIME, t  [SECOND]',-17,3.0,
     1 0.0,TIMIN,TIF,0.5)
      CALL PAXIS(Z1,Z2,X1,25,3.0,90.0,Y1MIN,Y1F,0.5)
C
C
      RETURN
      END
      SUBROUTINE DIM3(I,X,YA,Z)
C
C
      REAL*4 XMAX,YAMAX,ZMAX,XMIN,YAMIN,ZMIN
      REAL*4 X(2010),AXLGTH,FACTOR,YA(2010),Z(2010)
      REAL*4 YES,ATT,OVMIN,OVMAX,AT
      INTEGER*4 N,I
C
      DATA PIC1,PIC2,YES/4HPIC1,4HPIC2,3HYES/
      N=I
      XMAX=0.0
      YAMAX=0.0
      ZMAX=0.0
      XMIN=0.0
      YAMIN=0.0
      ZMIN=0.0
C
C       SEARCH FOR THE MAXIMUM VALUE IN THE X ARRAY.
C
      DO 30 IJ=1,N
      IF(IJ.EQ.1)GO TO 31
      IF(XMAX.GE.X(IJ))GO TO 30
 31   XMAX=X(IJ)
 30   CONTINUE
C
C        SEARCH FOR THE MAXIMUM VALUE IN THE YA ARRAY.
C
      DO 32 IJ=1,N
      IF(IJ.EQ.1)GO TO 33
      IF(YAMAX.GE.YA(IJ)) GO TO 32
 33   YAMAX=YA(IJ)
 32   CONTINUE
C
C       SEARCH FOR THE MAXIMUM VALUE IN THE Z ARRAY.
C
      DO 34 IJ=1,N
      IF(IJ.EQ.1)GO TO 35
      IF(ZMAX.GE.Z(IJ))GO TO 34
 35   ZMAX=Z(IJ)
 34   CONTINUE
C
C       SEARCH FOR THE MINIMUM VALUES IN THE X,YA,AND Z ARRAYS.
C
      DO 36 IJ=1,N
      IF(IJ.EQ.1)GO TO 37
      IF(XMIN.LE.X(IJ))GO TO 36
 37   XMIN=X(IJ)
 36   CONTINUE
C
      DO 38 IJ=1,N
```

```
      IF(IJ.EQ.1)GO TO 39
      IF(YAMIN.LE.YA(IJ))GO TO 38
 39   YAMIN=YA(IJ)
 38   CONTINUE
C
      DO 40 IJ=1,N
      IF(IJ.EQ.1)GO TO 41
      IF(ZMIN.LE.Z(IJ))GO TO 40
 41   ZMIN=Z(IJ)
 40   CONTINUE
C
C      DETERMINE THE LENGTH OF THE AXES TO BE PLOTTED IN THE THREE
C      DIMENSIONAL PLOTS.
C
      IF(XMAX.GE.YAMAX)GO TO 50
      IF(YAMAX.GE.ZMAX)GO TO 52
      GO TO 53
 50   IF(XMAX.GE.ZMAX)GO TO 51
 53   OVMAX=ZMAX
      GO TO 54
 51   OVMAX=XMAX
      GO TO 54
 52   OVMAX=YAMAX
C
 54   IF(XMIN.LE.YAMIN)GO TO 60
      IF(YAMIN.LE.ZMIN)GO TO 62
      GO TO 63
 60   IF(XMIN.LE.ZMIN) GO TO 61
 63   OVMIN=ZMIN
      GO TO 64
 61   OVMIN=XMIN
      GO TO 64
 62   OVMIN=YAMIN
 64   IF(ABS(OVMAX).GE.ABS(OVMIN)) GO TO 81
      AXLGTH=ABS(OVMIN)
      GO TO 82
 81   AXLGTH=ABS(OVMAX)
 82   FACTOR=0.8/AXLGTH
      DO 666 ILF=1,N
      X(ILF)=X(ILF)*FACTOR
      YA(ILF)=YA(ILF)*FACTOR
      Z(ILF)=Z(ILF)*FACTOR
 666  CONTINUE
C
C
      WRITE(7,432)XMAX,YAMAX,ZMAX,XMIN,YAMIN,ZMIN
 432  FORMAT(2X,'XMAX=',F15.8,10X,'YAMAX=',F15.8,10X,'ZMAX=',F15.8,
     1 /,2X,'XMIN=',F15.8,10X,'YAMIN=',F15.8,10X,'ZMIN=',F15.8)
      WRITE(7,433)OVMAX,OVMIN
 433  FORMAT(2X,'OVMAX=',F15.8,10X,'OVMIN=',F15.8)
C
      WRITE(7,77)FACTOR,AXLGTH
 77   FORMAT(///,5X,'FACTOR=',F15.7,7X,'AXLGTH=',F15.7)
C
C      THIS IS AN INTERACTIVE PORTION OF THE PROGRAM TO ALLOW THE
C      USER TO VIEW THE THREE DIMENSIONAL PLOT FROM THE DESIRED
C      ANGLE.
C
 44   WRITE(6,627)
 627  FORMAT(2X,'WHAT ARE THE DESIRED ANGLES TO VIEW THE PLOT ?')
```

```
      WRITE(6,676)
676   FORMAT(2X,'THE ANGLE FOR THE X-AXIS IS:   (DEGREES)')
      READ(5,845)AX
845   FORMAT(F15.7)
      WRITE(6,901)
901   FORMAT(2X,'THE ANGLE FOR THE Y-AXIS IS:   (DEGREES)')
      READ(5,902)AY
902   FORMAT(F15.7)
      WRITE(6,601)
601   FORMAT(2X,'THE ANGLE FOR THE Z-AXIS IS:   (DEGREES)')
      READ(5,617)AZ
617   FORMAT(F15.7)
      AAX=(AX*3.141592654)/180.0
      AAY=(AY*3.141592654)/180.0
      AAZ=(AZ*3.141592654)/180.0
C
C       START PLOTTING THE THREE DIMENSIONAL PLOT.
C
      CALL IGINIT
      CALL IGBGNS(PIC1)
      CALL IGBGNS(PIC2)
      CALL IGMA(-0.8,0.,0.)
      CALL IGDA(0.8,0.,0.)
      CALL IGMA(0.,-0.8,0.)
      CALL IGDA(0.,0.8,0.)
      CALL IGMA(0.,0.,-0.8)
      CALL IGDA(0.,0.,0.8)
      CALL IGMA(X(1),YA(1),Z(1))
      DO 45 IT=2,N
      CALL IGDA(X(IT),YA(IT),Z(IT))
 45   CONTINUE
      CALL IGTXT('<RSCL>',2.0)
      CALL IGMA(.4,-0.05,0.)
      CALL IGTXT('X<E>')
      CALL IGMA(-0.05,-0.05,0.)
      CALL IGTXT('O<E>')
      CALL IGMA(-0.05,.4,0.)
      CALL IGTXT('Y<E>')
      CALL IGMA(-0.05,0.,.4)
      CALL IGTXT('Z<E>')
      CALL IGTRAN(PIC2,'ROTY',AAY,'ROTX',AAX,'ROTZ',AAZ)
      CALL IGENDS(PIC2)
      CALL IGTXT('<RSCL>',2.0)
      CALL IGMA(-0.4,-0.85,0.)
      CALL IGTXT('FIGURE 10: END EFFECTOR POSITION
     1 <CRLF>FOR THE CASE OF R=2.0 METER,
     1 <CRLF>THETA=0.5 RAD. AND PHI=0.5 RAD.<E>')
      CALL IGENDS(PIC1)
      CALL IGDRON('TERMINAL')
C
C      THIS INTERACTIVE PORTION OF THE PROGRAM ALLOWS THE USER
C      TO MAKE HARD COPY OF THE THREE DIMENSIONAL PLOT IF DESIRED
C      AND TO SPECIFY ANOTHER SET OF ANGLES IF THE PLOT IS TO BE
C      VIEWED FROM DIFFERENT ANGLE.
C
      WRITE(6,48)
 48   FORMAT(2X,'DOES THE PLOT LOOKS GOOD ?',///,2X,
     1 'WOULD YOU LIKE TO MAKE A HARD COPY ?   (YES/NO)')
      READ(5,49)ATT
 49   FORMAT(A4)
```

```
      IF(ATT.EQ.YES)GO TO 888
      GO TO 889
888   CALL IGDRON('CALCOMP')
889   WRITE(6,324)
324   FORMAT(2X,'WOULD YOU LIKE TO TRY DIFFERENT ANGLES ?   (Y/N)')
      READ(5,325)AT
325   FORMAT(A4)
      IF(AT.EQ.YES) GO TO 44
      RETURN
      END
 0.5       0.5       0.5       0.4       0.0
 0.0       0.0       0.0       0.0       0.0
 8.0       8.0      16.0      50.0      50.0      50.0
 1.0       1.0       1.0       0.085     0.1       9.81
   0.04286     0.5       0.00003167    2707.0      5.67

.75 6.05 4.75 6.05 0.75 1.5254.75 1.525
RIGID AND FLEXIBLE MOTION CASE,
DESIRED POSITION : R=0.5 M
THETA=0.5 RAD. AND PHI=0.5 RAD.
  r DISPLACEMENT  [METER]
   THETA ROTATION [RAD]
     PHI ROTATION [RAD]
DISPLACEMENT, q (t) [M]
FIGURE 1.  GENERAL MOTION,
r DISPLACEMENT.
FIGURE 2.  GENERAL MOTION,
THETA DISPLACEMENT.
FIGURE 3.  GENERAL MOTION,
PHI ROTATION.
FIGURE 4.  GENERAL MOTION,
q (t) DISPLACEMENT.
DISPLACEMENT, q (t) [M]
DISPLACEMENT, q (t) [M]
DISPLACEMENT, q (t) [M]
   VELOCITY, r [M/SEC]
FIGURE 5.  GENERAL MOTION,
q (t) DISPLACEMENT.
FIGURE 6.  GENERAL MOTION,
q (t) DISPLACEMENT.
FIGURE 7.  GENERAL MOTION,
q (t) DISPLACEMENT.
FIGURE 8.  GENERAL MOTION,
r VELOCITY.
ANGULAR VELOCITY,    [RAD/SEC]
ANGULAR VELOCITY,    [RAD/SEC]
VELOC, q (t) [M/SEC]
VELOCITY, q (t) [M/SEC]
FIGURE 9.  GENERAL MOTION,
ANGULAR VELOCITY  .
FIGURE 10.  GENERAL MOTION,
ANGULAR VELOCITY  .
FIGURE 11.  GENERAL MOTION,
q (t) VELOCITY.
FIGURE 12.  GENERAL MOTION,
q (t) VELOCITY.
VELOCITY, q (t) [M/SEC]
VELOCITY, q (t) [M/SEC]
CONTROL TORQUE, T1 [N-M]
CONTROL TORQUE, T2 [N-M]
```

FIGURE 13. GENERAL MOTION,
q (t) VELOCITY.
FIGURE 14. GENERAL MOTION,
q (t) VELOCITY.
FIGURE 15. GENERAL MOTION,
CONTROL TORQUE FOR r.
FIGURE 16. GENERAL MOTION,
CONTROL TORQUE FOR .
CONTROL TORQUE, T3 [N-M]
V DEFLECTION [M].
W DEFLECTION [M].
FIGURE 17. GENERAL MOTION,
CONTROL TORQUE FOR .
FIGURE 18. GENERAL MOTION,
VERTICAL DEFLECTION V.
FIGURE 19. GENERAL MOTION,
HORIZONTAL DEFLECTION W.

APPENDIX G

LISTING OF THE COMPUTER CODE

FOR THE EXPERIMENTAL WORK

```
C
C
      INTEGER*2 ENC2V,CJ1,CJ2,DACVAL,DACHN,N,TERMCT,SOURCF
     1 ,ADCAVL,ADCHN
C
C
      COMMON /FRQ1/FREQ,N,DT
      COMMON /ENCBL/CJ1,CJ2,ENC2V
      COMMON /DACBL/DACHN,DACVAL
C     COMMON /ADCBL/ADCHN,ADCVAL
      COMMON /COUNT/TERMCT,SOURCF
      COMMON /STEP/K
C
      OPEN(17,FILE='DATA.TXT',STATUS='NEW')
C
      CALL INIT
      CALL CLKFRQ(FREQCK)
      DT=1.0/FREQCK
      WRITE(*,50)TERMCT,SOURCF,FREQCK,DT
   50 FORMAT(2X,'THE TERMINAL COUNT IS =',I6,/,2X,
     1 'THE SOURCE FREQUENCY IS =',I2,/,2X,
     1 'FREQUENCY DESIRED IS =',F15.8,/,2X,
     1 'SAMPLING PERIOD IS =',F15.8,///)
C
      CALL CLOCK
C
   14 WRITE(*,10)
   10 FORMAT(2X,'ENTER JOINT 2 CHANNEL NUMBER.(REAL NUMBER)')
      READ(*,11)DACH
   11 FORMAT(F15.8)
      DACHN=IFIX(DACH)
C
      WRITE(*,12)DACHN
   12 FORMAT(2X,'CHANNEL NUMBER CHOSEN FOR JOINT 2 IS ',I3,//,2X,
     1 'IS THIS SELECTION CORRECT ? (Y/N)')
      READ(*,13)AF
   13 FORMAT(A1)
      IF(AF.EQ.'N')GO TO 14
C
      DACVAL=0
      CALL DAC
C
      CJ1=2
      CJ2=3
C
      FREQ=FREQCK
      CALL CLSLP
C
      STOP
      END
```

Microsoft FORTRAN77 V3.31

```
>Name      Type          Offset       P Class
>
>ADCAVL    INTEGER*2     *****
>ADCHN     INTEGER*2     *****
>AF        REAL          438
>CJ1       INTEGER*2     0            /ENCBL /
>CJ2       INTEGER*2     2            /ENCBL /
>DACH      REAL          330
>DACHN     INTEGER*2     0            /DACBL /
>DACVAL    INTEGER*2     2            /DACBL /
>DT        REAL          6            /FRQ1  /
>ENC2V     INTEGER*2     4            /ENCBL /
>FREQ      REAL          0            /FRQ1  /
>FREQCK    REAL          16
>IFIX                                 INTRINSIC
>K         INTEGER*4     0            /STEP  /
>N         INTEGER*2     4            /FRQ1  /
>SOURCF    INTEGER*2     2            /COUNT /
>TERMCT    INTEGER*2     0            /COUNT /
>
>
>  51
>  52
>
>Name      Type          Size         Class
>
>CLKFRQ                               SUBROUTINE
>CLOCK                                SUBROUTINE
>CLSLP                                SUBROUTINE
>COUNT                   4            COMMON
>DAC                                  SUBROUTINE
>DACBL                   4            COMMON
>ENCBL                   6            COMMON
>FRQ1                    10           COMMON
>INIT                                 SUBROUTINE
>MAIN                                 PROGRAM
>STEP                    4            COMMON
>
>
>Pass One   No Errors Detected
>         52 Source Lines
```

Microsoft FORTRAN77 V3.31

```
      SUBROUTINE CLKFRQ(FREQCK)

C     THIS ROUTINE IS TO INPUT CLOCK FREQUENCY, AND THEN
C     DETERMINE TERMCT AND SOURCF FOR THE LABTENDER CLOCK.

      INTEGER*2 TERMCT,SOURCF
      COMMON/COUNT/TERMCT,SOURCF

      SOURCF = 5

    4 WRITE(*,7)
    7 FORMAT('0Enter Frequency of Clock Interrupts (Hz)'/
     *    ' Typical value: 1000.   Maximum: 5000.'/
     *    ' Include a decimal point.')
      READ (*,8) FREQCK
    8 FORMAT(F15.8)
      IF(FREQCK .LE. 5000.) GOTO 9
      WRITE(*,*) 'Clock frequency must be less than 5000. Hz.'
      GOTO 4
    9 CONTINUE

C     TESTCT = REAL VALUE FOR TERMCT, NECESSARY IF TERMCT > 32767.
   10 TESTCT = AINT((1*10**SOURCF)/FREQCK)

C     WRITE STATEMENTS FOR VARIOUS INTERMEDIATE VALUES ARE COMMENTED
C     OUT IN THIS PROGRAM TO MAKE THEM INVISIBLE TO THE USER.  YOU MAY WISH
C     TO USE THE WRITES TO SEE HOW THE PROGRAM WORKS, BUT DO NOT LEAVE
C     THEM IN YOUR FINAL PROGRAM.

      WRITE(*,15) TESTCT,SOURCF
   15 FORMAT(' TESTCT = ',F12.3,'   SOURCF = ',I2)

C     CHEK TO BE SURE TESTCT AND SOURCF ARE WITHIN THE DESIRED RANGE.
      IF (TESTCT .GT. 32767.) THEN
      SOURCF = SOURCF - 1
      GOTO 10
      ELSEIF (TESTCT .LT. 2.) THEN
      WRITE(*,17) FREQCK
   17 FORMAT(' Clock Frequency of ',F12.3,' is too high.')
      GOTO 4
      ELSEIF (SOURCF .LT. 2) THEN
      WRITE(*,18) FREQCK
   18 FORMAT(' Clock Frequency of ',F12.3,' Hz is too low.')
      GOTO 4
      ELSE
      CONTINUE
      TERMCT = INT(TESTCT)
      ENDIF

      WRITE(*,35) TERMCT,SOURCF
   35 FORMAT(' TERMCT =  ',I6,'     SOURCF = ',I2)
```

```
>   53 C      FROUT = (1*10**SOURCF)/REAL(TERMCT)
>   54 C      FREQK = FROUT/1000.
>   55 C      ERROR = ((FROUT - FREQCK)/FREQCK)*100.
>   56 C      WRITE(*,40) FREQCK,FROUT,ERROR,FREQK
>   57 C  40 FORMAT(' CLOCK FREQUENCY INPUT = ',F12.3,/
>   58 C   *     ' COUNTER FREQUENCY OUT = ',F12.3,/
>   59 C   *     ' ERROR                 = ',F12.3,' %',/
>   60 C   *     ' FREQ. IN KHz = ',F12.3)
>   61
>   62        RETURN
>   63        END
>
>Name      Type        Offset      P Class
>
>AINT                              INTRINSIC
>FREQCK    REAL        0           *
>INT                               INTRINSIC
>SOURCF    INTEGER*2   2           /COUNT /
>TERMCT    INTEGER*2   0           /COUNT /
>TESTCT    REAL        192
>
>
>
>Name      Type        Size        Class
>
>CLKFRQ                            SUBROUTINE
>COUNT                 4           COMMON
>
>
>Pass One   No Errors Detected
>       63 Source Lines
```

```
C
      SUBROUTINE CLSLP
C
C     *** THIS SUBROUTINE PERFORMS THE CLOSED LOOP RUN  ***
C     *** WHICH CONSISTS OF THE RIGID BODY CONTROLLER. ***
C     *** THE LATTER IS A POINT TO POINT CONTROLLER. ***
C
      INTEGER*2 N,K,CJ1,CJ2,ENC2V,DACVAL,DACHN,ICL,M(3000),J,
     1IFREQ,K1,K2,K3,ADCVAL,ADCHN,NF
C
      DIMENSION P(420),V(420),E(420),EI(420),VI(840),RM1(420),T(1000)
      DIMENSION PV(840)
C
      COMMON /FRQ1/FREQ,N,DT
      COMMON /ENCBL/CJ1,CJ2,ENC2V
      COMMON /DACBL/DACHN,DACVAL
      COMMON /ADCBL/ADCHN,ADCVAL
      COMMON /STEP/K
C
      IFREQ=IFIX(FREQ)
   16 WRITE(*,10)
   10 FORMAT(20X,'***  CLOSED LOOP RUN.  ***',/,20X,
     1 '***  RIGID BODY CONTROLLER ONLY.  ***',///,2X,
     1 '***  CHANNEL NUMBER CHOSEN FOR q dot IS 0  ***',/,
     1 2X,'*** .CHANNEL NUMBER CHOSEN FOR q IS 1  ***',////,2X,
     1 'ENTER CLOSED LOOP REFERENCE VALUE IN DEGREES.',/,2X,
     1 'TYPICAL VALUE IS BETWEEN 10 AND -20.0 DEGREES.')
      READ(*,11)CL
   11 FORMAT(F15.8)
C
C
      D=1.0
      ICL=IFIX(CL*31.25)
      AK1=-57.37/(D*31.25)
      AK2=-8.75/(D*31.25)
      AK3=-51.52/(D*31.25)
C
      WRITE(*,9)
    9 FORMAT(2X,'ENTER DURATION OF THE RUN. (SECONDS)',/,
     12X,'****  MAXIMUM 10 SECONDS.  ****')
      READ(*,8)DN
    8 FORMAT(F15.8)
      N=IFIX(FREQ*DN)
      NF=N+IFREQ*5
C
      WRITE(*,14)CL,ICL,D,N,DT,AK1,AK2,AK3
   14 FORMAT(2X,'CLOSED LOOP REFERENCE VALUE=',F15.8,'DEGREES.',/,
     12X,'CLOSED LOOP REFERENCE VALUE=',I7,'COUNTS',/,2X,
     1'SCALING FACTOR=',F10.5,3X,'N=',I7,2X,'DT=',F15.8,//,30X,
     1'CONTROLLER GAINS ARE:',//,2X,'K1=',F15.5,3X,'K2=',F15.5,
     1'K3=',F15.5,//,2X,'ARE THEY CORRECT ? (Y/N)')
      READ(*,15)AT1
```

```
   15 FORMAT(A1)
      IF(AT1.NE.'Y')GO TO 16
C
C     INITIALIZATION OF THE POSITION VECTOR, THE VELOCITY VECTOR AND
C     THE ERROR VECTOR.
C
      K=1
      CALL ENC
C
      ADCHN=0
      CALL ADC
      VI(K)=ADCVAL
C
      ADCHN=1
      CALL ADC
      PV(K)=ADCVAL
C
      P(K)=ENC2V
      POS=P(K)
      POST=P(K)
      ERROR=0.0
      EI(K)=0.0
      V(K)=0.0
      E(K)=ENC2V-ICL
      M(K)=0
C
      K=2
      CALL ENABLE
C
   18 J=K
      CALL ENC
C
      ADCHN=0
      CALL ADC
      VI(K)=ADCVAL
C
      ADCHN=1
      CALL ADC
      PV(K)=ADCVAL
C
      P(K)=ENC2V
      V(K)=(P(K)-POS)*FREQ
      E(K)=P(K)-ICL
      EI(K)=E(K)*DT+ERROR
      POS=P(K)
      ERROR=EI(K)
C
C     COMPUTATION OF THE CONTROL SIGNAL
C
```

```
>
>          RM=-AK1*(P(K)-POST)-AK2*V(K)-AK3*EI(K)
>          M(K)=IFIX(RM)
>          IF(M(K).GE.127) M(K)=127
>          IF(M(K).LE.-128) M(K)=-128
>   C
>   C      SEND OUT THE CONTROL SIGNAL.
>   C
>          DACVAL=M(K)
>          CALL DAC
>   C
>   C      WAIT FOR THE INTERRUPT SERVICE ROUTINE TO INCREMENT K.
>   C
>      17  IF(J.EQ.K)GO TO 17
>   C
>   C      CHECK WETHER THE TIME ASSIGNED FOR THIS MOVE HAS ELAPSED.
>   C
>          IF(K.LE.N)GO TO 18
>   C
>      68  J=K
>   C
>          ADCHN=0
>          CALL ADC
>          VI(K)=ADCVAL
>   C
>          ADCHN=1
>          CALL ADC
>          PV(K)=ADCVAL
>   C
>   C      SHUT OFF THE MOTOR.
>   C
>          DACVAL=0
>          CALL DAC
>   C
>      67  IF(J.EQ.K)GO TO 67
>   C
>          IF(K.LE.NF)GO TO 68
>   C      DISABLE THE INTERRUPT.
>   C
>          CALL DISABL
>   C
>   C      OUTPUT THE RESULTS.
>   C
>          DO 70 I=1,N
>          P(I)=P(I)/31.25
>          V(I)=V(I)/31.25
>          E(I)=E(I)/31.25
>          RM1(I)=M(I)*0.039055118
>          T(I)=(I-1)*DT
>      70  CONTINUE
>   C
>   C      WRITE(*,19)
```

```
>
> C 19 FORMAT(//,4X,'TIME',5X,'POSITION',4X,'VELOCITY',6X,'ERROR',
> C    15X,'CONTROL'/,9X,'(DEGREES)',3X,'(DEG/SEC)',3X,'(DEGREES)',
> C    13X,'(VOLTS)',//)
> C       DO 20 I=1,N
> C       WRITE(*,21)T(I),P(I),V(I),E(I),RM1(I)
> C 21 FORMAT(2X,F8.5,2X,F10.5,2X,F10.5,2X,F10.5,2X,F10.5)
> C 20 CONTINUE
> C
> C       WRITE(*,22)
> C 22 FORMAT(///,4X,'TIME',4X,'VIBRATION MEASUREMENT',/,17X
> C    1,'(MM/SECOND)',//)
>         DO 94 I=1,NF
>         T(I)=(I-1)*DT
>    94   CONTINUE
> C
> C       DO 89 I=1,NF
> C       WRITE(*,23)T(I),VI(I)
> C 23 FORMAT(2X,F8.5,2X,F10.5)
> C 89 CONTINUE
> C
>         DO 111 I=1,N
>         WRITE(17,112)T(I),P(I),V(I),E(I),RM1(I)
>    112 FORMAT(2X,F8.5,2X,F10.5,2X,F10.5,2X,F10.5,2X,F10.5)
>    111 CONTINUE
> C
>         DO 113 I=1,NF
>         WRITE(17,114)T(I),VI(I),PV(I)
>    114 FORMAT(2X,F8.5,2X,F10.2,2X,F10.2)
>    113 CONTINUE
> C
>         WRITE(*,24)
>    24   FORMAT(2X,'WOULD YOU LIKE TO MAKE ANOTHER RUN ? (Y/N)')
>         READ(*,25)AT2
>    25   FORMAT(A1)
>         IF(AT2.EQ.'Y')GO TO 16
> C
>         RETURN
>         END
>
```

```
                                              Microsoft FORTRAN77 V3.31
>
>Name        Type          Offset          P Class
>
>ADCHN       INTEGER*2     0               /ADCBL /
>ADCVAL      INTEGER*2     2               /ADCBL /
>AK1         REAL          25584
>AK2         REAL          25588
>AK3         REAL          25592
>AT1         REAL          26012
>AT2         REAL          26182
>CJ1         INTEGER*2     0               /ENCBL /
>CJ2         INTEGER*2     2               /ENCBL /
>CL          REAL          25514
>D           REAL          25578
>DACHN       INTEGER*2     0               /DACBL /
>DACVAL      INTEGER*2     2               /DACBL /
>DN          REAL          25694
>DT          REAL          6               /FRQ1  /
>E           REAL          21776
>EI          REAL          23456
>ENC2V       INTEGER*2     4               /ENCBL /
>ERROR       REAL          26028
>FREQ        REAL          0               /FRQ1  /
>I           INTEGER*4     26038
>ICL         INTEGER*2     25582
>IFIX                                      INTRINSIC
>IFREQ       INTEGER*2     25136
>J           INTEGER*2     26032
>K           INTEGER*2     0               /STEP  /
>K1          INTEGER*2     *****
>K2          INTEGER*2     *****
>K3          INTEGER*2     *****
>M           INTEGER*2     15776
>N           INTEGER*2     4               /FRQ1  /
>NF          INTEGER*2     25706
>P           REAL          16
>POS         REAL          26020
>POST        REAL          26024
>PV          REAL          12416
>RM          REAL          26034
>RM1         REAL          6736
>T           REAL          8416
>V           REAL          1696
>VI          REAL          3376
>
>
>  191
>
>Name        Type          Size            Class
>
>ADC                                       SUBROUTINE
```

```
                                         Microsoft FORTRAN77 V3.31
>
>ADCBL                    4              COMMON
>CLSLP                                   SUBROUTINE
>DAC                                     SUBROUTINE
>DACBL                    4              COMMON
>DISABL                                  SUBROUTINE
>ENABLE                                  SUBROUTINE
>ENC                                     SUBROUTINE
>ENCBL                    6              COMMON
>FRQ1                     10             COMMON
>STEP                     2              COMMON
>
>
>Pass One    No Errors Detected
>       191 Source Lines
#
```

```
The IBM Personal Computer Assembler 01-01-80        PAGE   1-1
>Experimental Work CODE
>
>
>
>                  TITLE  EXPERIMENTAL WORK ASSEMBLY CODE
>
>                  PAGE  ,132     ;Set page width to 132 characters.
>
>                      ;    This routine initializes the digital I/O port and
>                      ;    sets up the IRQ4 interrupt vector.
>                      ;    FORTRAN "CALL INIT" calls this procedure.
>
>0000              DATA SEGMENT PUBLIC 'DATA'
>                      ;    For local assembler program data storage.
>                      ;    Not used, but required for linking
>0000              DATA ENDS
>
>0000              STEP$A SEGMENT COMMON '$STEP'
>
>                      ;    This segment is equivalent to FORTRAN's
>                      ;    COMMON/STEP/K.  The $ and A are
>                      ;    required to match the segment name and
>                      ;    class that the linker uses for the FORTRAN
>                      ;    common.  Eleven variables must be declared
>                      ;    INTEGER*2 in the FORTRAN calling routine
>                      ;    before the common block.
>
>0000  ????        K     DW   ?
>
>0002              STEP$A ENDS
>
>                  DGROUP GROUP DATA,STEP$A
>                      ;    The DATA,STEP$A segment will be linked into
>                      ;    the group called DGROUP to match Microsoft
>                      ;    FORTRAN convention.
>
>0000              CODE SEGMENT PUBLIC 'CODE'
>
>                  ASSUME CS:CODE,DS:DGROUP,SS:DGROUP
>                  PUBLIC  INIT                ;    Make the INIT label available
>                                              ;    to other segments.
>
>                      ;    Labtender Address, in IBM PC, for SETDDA routine.
>= 033F            IOCOM  EQU  831             ;    Parallel Port Control Register.
>
>0000              INIT   PROC FAR
>
>0000  55          PUSH   BP                   ;    Save calling framepointer.
>0001  8B EC       MOV    BP,SP
>
>0003  B2 49       MOV DL,73                   ;    Print an 'I' to the screen
>0005  B4 02       MOV AH,2                    ;    for debugging
>0007  CD 21       INT 21H
```

```
The IBM Personal Computer Assembler 01-01-80      PAGE   1-2
>Experimental Work Code
>                    INTERRUPT SERVICE ROUTINE
>
>
>                       ;   Load the IRQ4 interrupt vector (address to
>                       ;   go to upon interrupt 4).
>
>0009  FA            CLI                    ;   Disable interrupts while we are
>                                           ;   modifying them to prevent any
>                                           ;   catastrophes.
>000A  06            PUSH ES                ;   Save Extra Segment
>000B  B8 0000       MOV AX,0000H
>
>000E  8E C0         MOV ES,AX              ;   Point Extra Segment at the ISR address
>0010  BF 0030       MOV DI,30H             ;   Table.  Offset of entry for IRQ 4 in the
>0013  B8 0020 R     MOV AX,OFFSET ISR      ;   address table.  Get offset of DDASER,
>                                           ;   our interrupt service routine.
>
>0016  AB            STOSW                  ;   Load offset in AX into address in DI,
>                                           ;   and add 2 to DI to prepare to store CS.
>0017  8C C8         MOV AX,CS              ;   Get Code Segment for assembly code.
>0019  AB            STOSW                  ;   Load code segment in AX into address
>                                           ;    in DI.  Now the 4-byte address of our
>                                           ;   DDASER begins at memorY location 30H.
>                                           ;   An IRQ4 Interrupt (if enabled) will
>                                           ;   cause the microprocessor to imme-
>                                           ;   diately go to that address, and start
>                                           ;   performing our DDASER.
>001A  07            POP ES                 ;   Restore Extra Segment
>001B  FB            STI                    ;   Enable interrupts
>
>001C  8B E5         MOV    SP,BP           ;   Restore framepointer
>001E  5D            POP    BP
>001F  CB            RET                    ;   Return
>
>0020               INIT ENDP
>
>                   ;---------------------------------------------------------
>                       SUBTTL INTERRUPT SERVICE ROUTINE
>                   ;---------------------------------------------------------
>
>                   ;   Labtender Addresses, in IBM PC, for DDASER routine.
>= 0332          INTCLR EQU  818        ;   Timer Interrupt Clear Address.
>
>                   ;   8259 Programmable Interrupt Controller Addresses
>                   ;   in IBM PC.
>= 0020          ICR  EQU  20H          ;   Interrupt Command Register.
>= 0021          IMR  EQU  21H          ;   Interrupt Mask Register.
>
>                   ;   Constants for DDASER routine.
>= 0020             EOI  EQU  20H       ;   End of Interrupt signal for
>                                       ;   8259 Interrupt Controller.
```

```
The IBM Personal Computer Assembler 01-01-80      PAGE   1-3
>Experimental Work Code
>                    INTERRUPT SERVICE ROUTINE
>0020                ISR PROC FAR
>0020  55            PUSH   BP          ;   Save calling framepointer.
>0021  8B EC         MOV    BP,SP
>
>0023  50            PUSH AX            ;   Save all the registers
>0024  53            PUSH BX            ;   we will be modifying.
>0025  51            PUSH CX            ;   This is done automatically
>0026  52            PUSH DX            ;   only during FORTRAN CALLs,
>                                       ;   not interrupts.
>
>0027  FF 06 0000 R      INC    K        ;   K = K + 1, count another
>                                       ;   time step.
>002B                CLKEOI:
>002B  BA 0332           MOV DX,INTCLR  ;   Timer Interrupt Clear
>002E  B0 00             MOV AL,0       ;   Data could be anything
>0030  EE                OUT DX,AL      ;   Write to clear the interrupt.
>
>0031  B0 20             MOV    AL,EOI  ;   Get end-of interrupt signal
>0033  BA 0020           MOV    DX,ICR  ;   Send it to the 8259's

>
>0036  EE                OUT    DX,AL   ;   Interrupt Command Register
>
>0037  5A                POP  DX        ;   Restore registers
>0038  59                POP  CX
>0039  5B                POP  BX
>003A  58                POP  AX
>
>003B  8B E5             MOV    SP,BP   ;   Restore framepointer
>003D  5D                POP    BP
>
>003E  CF                IRET           ;   Interrupt return.
>003F                    ISR ENDP
>
>003F                    CODE ENDS
>
>                        END
```

```
The IBM Personal Computer Assembler 01-01-80       PAGE    Symbols-1
>Experimental Work Code
>
>
>Segments and groups:
>
>           N a m e                 Size     align    combine    class
>
>CODE . . . . . . . . . . . . . .   003F     PARA     PUBLIC     'CODE'
>DGROUP . . . . . . . . . . . . .   GROUP
> DATA . . . . . . . . . . . . . .  0000     PARA     PUBLIC     'DATA'
> STEP$A . . . . . . . . . . . . .  0002     PARA     WORD   '   $STEP'
>
>Symbols:
>
>           N a m e                 Type     Value    Attr
>
>CLKEOI . . . . . . . . . . . . .   L NEAR   002B     CODE
>EOI. . . . . . . . . . . . . . .   Number   0020
>ICR. . . . . . . . . . . . . . .   Number   0020
>IMR. . . . . . . . . . . . . . .   Number   0021
>INIT . . . . . . . . . . . . . .   F PROC   0000     CODE       Global  Length=0020
>INTCLR . . . . . . . . . . . . .   Number   0332
>IOCOM. . . . . . . . . . . . . .   Number   033F
>ISR. . . . . . . . . . . . . . .   F PROC   0020     CODE       Length =001F
>K. . . . . . . . . . . . . . . .   L WORD   0000     STEP$A
>
>Warning Severe
>Errors  Errors

>     0
#
```

```
The IBM Personal Computer Assembler 01-01-80      PAGE 1-1
>CLOCK - ROUTINE TO INITIALIZE COUNTERS 1 AND 2
>
>
>
>             TITLE  CLOCK - ROUTINE TO INITIALIZE COUNTERS 1 AND 2
>
>             PAGE  ,132     ;Set page width to 132 characters
>
>                   ;   CLOCK is a routine which initializes Counter 1 and Counter 2
>                   ;   in the Labtender. Counters can be used to count and/or
>                   ;   generate various signals. The frequency sources, signal
>                   ;   outputs, and maximum counts are all programmable.
>                   ;   Counter 1 will be used as a clock, generating pulses at a
>                   ;   desired frequency. Counter 2 will count the number of pulses
>                   ;   that Counter 1 generates. FORTRAN "CALL CLOCK" calls this
>                   ;   procedure.
>
>                   ;   COUNTER 1:
>                   ;   TERMCT and SOURCF control the frequency generated by
>                   ;   counter 1. TERMCT, the terminal count for Counter 1,
>                   ;   is the number of pulses Counter 1 will count from the source
>                   ;   frequency before generating a pulse. Counter 1 counts down
>                   ;   from TERMCT to zero repetitively.
>                   ;   SOURCF selects the source frequency for Counter 1 from the
>                   ;   Labtender. NOTE THAT SOURCF ISN'T THE ACTUAL SOURCE
>                   ;   frequency; It is used to SELECT A FREQUENCY SOURCE
>                   ;   ON THE LABTENDER.
>                   ;   The Labtender has a 1 MHz crystal which generates 5 possible
>                   ;   source frequencies, from 1 MHz (10^6) to 100 Hz (10^2).
>                   ;   To choose a source frequency, we define SOURCF as the power
>                   ;   of 10 of the desired source frequency from the Labtender for
>                   ;   Counter 1. We generally need frequencies below 5 kHz, and
>                   ;   higher frequencies have occasionally caused problems, so we
>                   ;   will limit the source frequency to 100 kHz and the clock
>                   ;   frequency to 5 kz. SOURCF must then be between 2 and 5.
>                   ;   TERMCT must be between 2 and 32767. A program called
>                   ;   CLKFRQ.FOR is available to select TERMCT and SOURCF, and
>                   ;   can be copied into your program and modified as desired.
>                   ;   Counter 1 will generate a pulse on pin OUT 1 every time it
>                   ;   counts TERMCT pulses. The output frequency of Counter 1
>                   ;   (the 'clock' frequency) is (1*10**SOURCF)/TERMCT. To
>                   ;   generate a 500 Hz clock, choose 500 =10000/20 so SOURCF
>                   ;   = 4 and TERMCT = 20, for example.
>                   ;   The OUT1 pin can be used to generate timer interrupts.
>                   ;   Enabling the IRQ4 interrupt will cause the timer interrupts
>                   ;   to be sent to the processor. If we don't enable IRQ4 then
>                   ;   the interrupts will be ignored.
>
>                   ;   COUNTER 2:
>                   ;   Counter 2 counts the pulses generated from Counter 1. The
```

```
The IBM Personal Computer Assembler 01-01-80       PAGE   1-2
>CLOCK - ROUTINE TO INITIALIZE COUNTERS 1 AND 2
>
>                    ;   assembler routine CLKOUT(CLKTIC) can then be used to read
>                    ;   the value in Counter 2 and return it to FORTRAN as CLKTIC,
>                    ;   the number of clock counts.
>0000            DATA SEGMENT PUBLIC 'DATA'
>                    ;   For local assembler program data storage.
>                    ;   Not used, but required for linking
>0000            DATA ENDS
>
>
>0000            COUNT$A SEGMENT COMMON '$COUNT'
>
>                    ;   This segment is equivalent to COMMON/COUNT/
>                    ;   TERMCT,SOURCF in FORTRAN.
>                    ;   The $ and A are required to match the segment name
>                    ;   and class that the linker uses for the FORTRAN common.
>                    ;   TERMCT and SOURCF must be declared INTEGER*2 in the
>                    ;   FORTRAN calling routine before the common block.
>
>0000 ????       TERMCT DW  ?
>0002 ????       SOURCF DW  ?
>
>0004            COUNT$A ENDS
>
>                DGROUP GROUP DATA,COUNT$A
>                    ;   The DATA and COUNT$A segments will be linked into
>                    ;   the group called DGROUP, to match Microsoft
>                    ;   FORTRAN convention.
>
>0000            CODE SEGMENT PUBLIC 'CODE'
>
>                ASSUME CS:CODE,DS:DGROUP,SS:DGROUP
>                PUBLIC CLOCK            ;   Make the CLOCK label available
>                                        ;   to other segments.
>
>                ;   Labtender Addresses, in IBM PC, for the programmable timer.
>
>= 0338          CLKDATA EQU  824        ;   Data Port for 9513 Timer.
>= 0339          CLKCOM  EQU  825        ;   Command Port for 9513 timer.
>
>0000            CLOCK PROC FAR
>
>0000  55            PUSH   BP           ;   Save calling framepointer.
>0001  8B EC         MOV    BP,SP
>
>0003  B2 43         MOV DL,67           ;   Print a 'C' to the screen
>0005  B4 02         MOV AH,2            ;   for debugging
>0007  CD 21         INT 21H
>                    ;   First reset all counters to zero.
>0009  BA 0339       MOV DX,CLKCOM       ;   Master Reset command.
```

```
The IBM Personal Computer Assembler 01-01-80       PAGE   1-3
>CLOCK - ROUTINE TO INITIALIZE COUNTERS 1 AND 2
>
>
>000C  B0 FF           MOV AL,255
>000E  EE              OUT DX,AL
>
>                      ;    Set up the Master Mode Register.
>
>000F  BA 0339         MOV DX,CLKCOM     ;    Set Data Pointer to Master
>0012  B0 17           MOV AL,23         ;    mode register
>0014  EE              OUT DX,AL
>
>0015  BA 0338         MOV DX,CLKDATA    ;    Write data to Master Mode
>0018  B0 00           MOV AL,0          ;    Register, lo-byte:
>001A  EE              OUT DX,AL         ;    Use default values
>
>001B  B0 80           MOV AL,128        ;    Write data to Master Mode
>                                        ;    Register, hi byte:
>001D  EE              OUT DX,AL         ;    Use BCD division when dividing
>                                        ;    frequencies.  Ex: 1 MHz
>                                        ;    crystal/1000 (base 10)=1000Hz
>
>                      ;    Set up Counter 1.
>
>001E  BA 0339         MOV DX,CLKCOM     ;    Disarm Counter 1, so we can
>0021  B0 C1           MOV AL,193        ;    write to its control port
>0023  EE              OUT DX,AL
>
>0024  B0 01           MOV AL,1          ;    Set Data Pointer to Counter 1's
>0026  EE              OUT DX,AL         ;    Mode Register.
>
>0027  BA 0338         MOV DX,CLKDATA    ;    Write Data to Counter 1 Mode Register
>002A  B0 21           MOV AL,33         ;    lo-byte, Count repetitively, reload
                                         ;    from load register,
>002C  EE              OUT DX,AL         ;    Count down, Terminal Count pulse high.
>
>                      ;    Manipulate SOURCF to create command to select source
>                      ;    frequency.  By coincidence, (17 - SOURCF) will be correct
>                      ;    command.
>002D  8B 0E 0002 R       MOV CX,SOURCF  ;    Put SOURCF in CX.
>0031  B8 0011            MOV AX,17      ;    AX = 17
>0034  2B C1              SUB AX,CX      ;    AX <-- AX - CX.  Now AL has command.
>0036  EE                 OUT DX,AL      ;    Write Data to Counter 1 Mode Register
>                                        ;    hi-byte:  Count on rising
>                                        ;    edge, source frequency = 10**SOURCF
>
>                      ;   Put TERMCT in Counter 1's Load Register.  TERMCT is the value
>                      ;   from which Counter 1 will start counting down.
>
>0037  BA 0339            MOV DX,CLKCOM  ;    Set Data Pointer to Counter 1's Load
>003A  B0 09              MOV AL,9       ;    Register.
>003C  EE                 OUT DX,AL
```

```
The IBM Personal Computer Assembler 01-01-80        PAGE  1-4
>CLOCK - ROUTINE TO INITIALIZE COUNTERS 1 AND 2
>
>003D  BA 0338     MOV DX,CLKDATA    ;   Write data to Counter 1's Load Register.
>0040  A1 0000 R   MOV AX,TERMCT     ;   Put TERMCT in AX.
>0043  EE          OUT DX,AL         ;   Send lo-byte to Load Register.
>0044  8A C4       MOV AL,AH         ;   Put hi-byte of TERMCT into AL.
>0046  EE          OUT DX,AL         ;   Send hi-byte to Load Register.
>
>                  ;   Set up Counter 2.
>
>0047  BA 0339     MOV DX,CLKCOM     ;   Disarm Counter 2, so we can
>004A  B0 C2       MOV AL,194        ;   write to its control port.
>004C  EE          OUT DX,AL
>
>004D  B0 02       MOV AL,2          ;   Set Data Pointer to Counter 2's
>004F  EE          OUT DX,AL         ;   Mode Register.
>
>0050  BA 0338     MOV DX,CLKDATA    ;   Write Data to Counter 2 Mode
>0053  B0 2A       MOV AL,42         ;   Register, lo-byte: Count repetitively,
>                                    ;   reload from load register,
>0055  EE          OUT DX,AL         ;   Count up, square wave output.
>
>0056  B0 00       MOV AL,0          ;   Write Data to Counter 2 Mode
                                     ;   Register,  hi-byte:
>0058  EE          OUT DX,AL         ;   Count on rising edge, source
>                                    ;   frequency = Counter 1.  Each time
>                                    ;   Counter 1 =0 we get a pulse.
>                  ;   Put zero in Counter 2's Load Register.
>
>0059  BA 0339     MOV DX,CLKCOM     ;   Set Data Pointer to Counter
>005C  B0 0A       MOV AL,10         ;   1's Load Register.
>005E  EE          OUT DX,AL
>
>005F  B8 0000     MOV AX,0          ;   Put zero in AX.
>0062  EE          OUT DX,AL         ;   Send lo-byte to Load Register.
>0063  8A C4       MOV AL,AH         ;   Put hi-byte of AX into AL.
>0065  EE          OUT DX,AL         ;   Send hi-byte to Load Register.
>
>                  ;   Turn on Counters 1 & 2.
>
>0066  B0 63       MOV AL,99         ;   Command to Load and Arm
>0068  EE          OUT DX,AL         ;   Counters 1 & 2
>
>0069  8B E5       MOV   SP,BP       ;   Restore framepointer
>006B  5D          POP   BP
>006C  CB          RET               ;   Return
>
>006D              CLOCK ENDP
>
>006D              CODE ENDS
>
>                  END
```

```
The IBM Personal Computer Assembler 01-01-80      PAGE    Symbols-1
>CLOCK - ROUTINE TO INITIALIZE COUNTERS 1 AND 2
>
>
>Segments and groups:
>
>          N a m e               Size     align    combine   class
>
>CODE . . . . . . . . . . . . . .   006D     PARA     PUBLIC    'CODE'
>DGROUP . . . . . . . . . . . . .   GROUP
> DATA . . . . . . . . . . . . . .  0000     PARA     PUBLIC    'DATA'
> COUNT$A. . . . . . . . . . . . .  0004     PARA     WORD      '$COUNT'
>
>Symbols:
>
>          N a m e               Type     Value    Attr
>
>CLKCOM . . . . . . . . . . . . .   Number   0339
>CLKDATA. . . . . . . . . . . . .   Number   0338
>CLOCK. . . . . . . . . . . . . .   F PROC   0000     CODE      Global  Length=006D
>SOURCF . . . . . . . . . . . . .   L WORD   0002     COUNT$A
>TERMCT . . . . . . . . . . . . .   L WORD   0000     COUNT$A
>
>Warning Severe
>Errors  Errors

>    0
```

```
The IBM Personal Computer Assembler 01-01-80        PAGE   1-1
>DISABL - ROUTINE TO DISABLE INTERRUPTS
>
>
>
>                  TITLE  DISABL - ROUTINE TO DISABLE INTERRUPTS
>
>                  PAGE  ,132     ;Set page width to 132 characters.
>
>                      ;    FORTRAN "CALL DISABL" calls this procedure.
>
>0000              DATA SEGMENT PUBLIC 'DATA'
>                      ;    For local assembler program data storage.
>                      ;    Not used, but required for linking
>0000              DATA ENDS
>
>                  DGROUP GROUP DATA       ;    The DATA segment will be
>                                          ;    linked into the group called
>                                          ;    DGROUP to match Microsoft
>                                          ;    FORTRAN convention.
>0000              CODE SEGMENT PUBLIC 'CODE'
>
>                  ASSUME CS:CODE,DS:DGROUP,SS:DGROUP
>                  PUBLIC DISABL           ;    Make the DISABL label avai-
>                                          ;    lable to other segments.
>
>                  ;    Labtender Addresses, in IBM PC, for initialization routine.
>= 0332            INTCLR EQU  818         ;    Timer Interrupt Clear Address.
>= 033D            IOPORTB EQU  829        ;    Parallel Port B Data.
>
>                  ;    8259 Programmable Interrupt Controller Addresses in IBM PC.
>= 0021            IMR   EQU 021H          ;    Interrupt Mask Register.
>
>                      ;   Constants:
>= 0000            TRAIL  EQU  0H
>
>0000              DISABL PROC FAR
>
>0000  55          PUSH   BP               ;    Save calling framepointer
>0001  8B EC       MOV    BP,SP
>
>0003  BA 0021     MOV DX,IMR
>0006  EC          IN AL,DX                 ;    Get contents of IMR.
>0007  0C 10       OR AL,00010000B          ;    Mask to block bit 4. 1=disable.
>0009  EE          OUT DX,AL                ;    Reset the IMR.
>
>000A  BA 0332        MOV DX,INTCLR         ;    Clear timer interrupt,
>000D  B0 00          MOV AL,0              ;    just in case it's not already.
>000F  FA             CLI                   ;    Clear interrupt flag
>0010  EE             OUT DX,AL             ;    Write to clear the interrupt
>0011  FB             STI                   ;    Reset the interrupt flag
>
>0012  BA 033D        MOV    DX,IOPORTB     ;    To allow the stepper motor to be
```

```
The IBM Personal Computer Assembler 01-01-80       PAGE   1-2
>DISABL - ROUTINE TO DISABLE INTERRUPTS
>
>
>                                        ;   manually reset, we must leave the
>0015  B0 00         MOV    AL,TRAIL      ;   output to it high, so we don't
>0017  EE            OUT    DX,AL         ;   complement (NOT) the low trail here.
>
>0018  B2 44         MOV DL,68            ;   Print a 'D' to the screen
>001A  B4 02         MOV AH,2             ;   for debugging
>001C  CD 21         INT 21H
>
>001E  8B E5         MOV    SP,BP         ;   Restore framepointer
>0020  5D            POP    BP
>0021  CB            RET                  ;   Return
>
>0022                DISABL ENDP
>
>0022                CODE ENDS
>
>                    END
>
>Segments and groups:
>
>         N a m e              Size     align    combine   class
>
>CODE . . . . . . . . . . . . . .  0022     PARA     PUBLIC    'CODE'
>DGROUP . . . . . . . . . . . . .  GROUP
> DATA . . . . . . . . . . . . . . 0000     PARA     PUBLIC    'DATA'
>
>Symbols:
>
>         N a m e              Type     Value    Attr
>
>DISABL . . . . . . . . . . . . .  F PROC   0000     CODE      Global  Length=0022
>IMR. . . . . . . . . . . . . . .  Number   0021
>INTCLR . . . . . . . . . . . . .  Number   0332
>IOPORTB. . . . . . . . . . . . .  Number   033D
>TRAIL. . . . . . . . . . . . . .  Number   0000
>
>Warning Severe
>Errors  Errors

>    0
#
```

```
 The IBM Personal Computer Assembler 01-01-80      PAGE   1-1
>ENABLE - ROUTINE TO ENABLE INTERRUPTS
>
>
>
>                  TITLE  ENABLE - ROUTINE TO ENABLE INTERRUPTS
>
>
>                  PAGE  ,132      ;Set page width to 132 characters.
>
>                       ;    FORTRAN "CALL ENABLE" calls this procedure.
>
>0000                        DATA SEGMENT PUBLIC 'DATA'
>                       ;    For local assembler program data storage.
>                       ;    Not used, but required for linking
>0000                        DATA ENDS
>
>                       DGROUP GROUP DATA    ;    The DATA segment will be
>                                            ;    linked into the group called
>                                            ;    DGROUP to match Microsoft
>                                            ;    FORTRAN convention.
>0000                        CODE SEGMENT PUBLIC 'CODE'
>
>                            ASSUME CS:CODE,DS:DGROUP,SS:DGROUP
>                            PUBLIC ENABLE    ;    Make the ENABLE label avai-
>                                            ;    lable to other segments.
>
>                       ;    Labtender Address, in IBM PC.
>= 0332                      INTCLR EQU  818  ;    Timer Interrupt Clear address.
>
>                  ;    8259 Programmable Interrupt Controller Addresses in IBM PC.
>= 0021                      IMR    EQU  021H ;    Interrupt Mask Register.
>
>0000                        ENABLE PROC FAR
>
>0000  55                    PUSH   BP        ;    Save calling framepointer.
>0001  8B EC                 MOV    BP,SP
>
>0003  BA 0332               MOV DX,INTCLR    ;    Clear timer interrupt,
>0006  B0 00                 MOV AL,0         ;    just in case it's not already.
>0008  EE                    OUT DX,AL        ;    Write to clear the interrupt
>
>0009  B2 45                 MOV DL,69        ;    Print an 'E' to the screen.
>000B  B4 02                 MOV AH,2         ;    for debugging
>000D  CD 21                 INT 21H
>
>000F  BA 0021               MOV DX,IMR
>0012  EC                    IN AL,DX         ;    Get contents of IMR.
>0013  24 EF                 AND AL,11101111B ;    Mask to include bit 4.
>0015  FA                    CLI              ;    Clear interrupt flag
>0016  EE                    OUT DX,AL        ;    Reset the IMR.
>0017  FB                    STI              ;    Reset the interrupt flag
```

```
The IBM Personal Computer Assembler 01-01-80        PAGE   1-2
>ENABLE - ROUTINE TO ENABLE INTERRUPTS
>
>
>0018  8B E5            MOV    SP,BP        ;   Restore framepointer
>001A  5D               POP    BP
>001B  CB               RET                 ;   Return
>
>001C                   ENABLE ENDP
>001C                   CODE ENDS
>
>                       END
>
>Segments and groups:
>
>         Name               Size     align    combine    class
>
>CODE . . . . . . . . . . . . . .   001C     PARA     PUBLIC     'CODE'
>DGROUP . . . . . . . . . . . . .   GROUP
> DATA . . . . . . . . . . . . . .  0000     PARA     PUBLIC     'DATA'
>
>Symbols:
>
>         Name               Type     Value    Attr
>
>ENABLE . . . . . . . . . . . . .   F PROC   0000     CODE       Global Length=001C
>IMR. . . . . . . . . . . . . . .   Number   0021
>INTCLR . . . . . . . . . . . . .   Number   0332
>
>Warning Severe
>Errors  Errors

>     0
```

```
The IBM Personal Computer Assembler 01-01-80       PAGE   1-1
>ENC - ENCODER READING ROUTINE
>
>
>                TITLE ENC - ENCODER READING ROUTINE
>                    ;   This routine reads the contents of the counters
>                    ;   of the second joints and sends their value in the
>                    ;   variable 'enc2V' to the fortran code.
>
>                PAGE  ,132      ;Set page width to 132 characters.
>
>0000            DATA SEGMENT PUBLIC 'DATA'
>                    ;   For local assembler program data storage
>                    ;   not used here, but required for linking purposes.
>0000            DATA ENDS
>
>0000            ENCBL$A SEGMENT COMMON '$ENCBL'
>                    ;   This segment is equivalent to COMMON/
>                    ;   ENCBLK/CJ1,CJ2,ENC2V
>                    ;   in FORTRAN.  The $ sign and A are required to
>                    ;   match the segment name and class that the linker
>                    ;   uses for the FORTRAN common statement.  ENC2V,
>                    ;   CJ1 and CJ2 must be declared INTEGER*2 in the
>                    ;   FORTRAN calling routine before the common block.
>0000  ????      CJ1   DW   ?
>0002  ????      CJ2   DW   ?
>0004  ????      ENC2V DW   ?
>0006            ENCBL$A ENDS
>                    ;
>                DGROUP GROUP DATA,ENCBL$A
>                    ;   The data and ENCBLK$A segments will be
>                    ;   linked in the group called DGROUP, to match
>                    ;   MICROSOFT FORTRAN convention.
>0000            CODE SEGMENT PUBLIC 'CODE'
>                    ;
>                ASSUME CS:CODE,DS:DGROUP,SS:DGROUP
>                PUBLIC LP
>                PUBLIC ENC
>                    ;   make ENC label available to other segments
>                ;--------------------------------------
>                SUBTTL READING OF THE ENCODER
>                ;--------------------------------------
>= 0330          SLOTNO  EQU 816
>= 033F          ENCM    EQU SLOTNO+15
>= 033D          OUTPORT EQU SLOTNO+13
>
>0000            ENC PROC FAR
>
>0000  55        PUSH   BP             ;    save calling framepointer
>0001  8B EC     MOV    BP,SP          ;    on the stack.
>
>0003  50        PUSH   AX             ;    save AX register.
>0004  53        PUSH   BX             ;    save BX register.
```

```
The IBM Personal Computer Assembler 01-01-80       PAGE   1-2
>ENC - ENCODER READING ROUTINE.
>                READING OF THE ENCODER
>
>0005  52         PUSH   DX                  ;  save DX register.
>0006  BA 033F    LP: MOV DX,ENCM             ;  setting the i/o ports by writing
>0009  B0 90      MOV AL,090H                 ;  a value to AL according to the
>000B  EE         OUT DX,AL                   ;  labtender specifications and
>                                             ;  sending the latter to the control
>                                             ;  port through the DX register.
>                                             ;  This will determine which port
>                                             ;  is to be used as input port and
>                                             ;  which to be used as output port.
>
>000C  B0 08          MOV AL,08H              ;  strobing the counters by sending
>000E  BA 033D        MOV DX,OUTPORT          ;  a high signal to bit 4 through the
>0011  EE             OUT DX,AL               ;  output port. Physically, this will
>                                             ;  lock all the counters, mount their
>                                             ;  contents on the latches and wait.
>
>0012  8B 1E 0002 R       MOV BX,CJ2          ;  get the address of the counter to
>0016  8A C3              MOV AL,BL           ;  be read. Mask off high byte since
>0018  EE                 OUT DX,AL           ;  only low byte is used. Send it
>0019  4A                 DEC DX              ;  through the DX register. Move to
>001A  EC                 IN  AL,DX           ;  the input port. Read the counter.
>
>001B  24 0F              AND AL,0FH          ;  Since only 12 bits are assigned to
>                                             ;  joints 2. Therefore, the 4 most
>                                             ;  significant bits of the upper byte
>                                             ;  of joint 2 word must be masked.
>
>001D  8A E0              MOV AH,AL           ;  Save the counter reading in AH.
>001F  42                 INC DX              ;  Move back to the output port.
>0020  8B 1E 0000 R       MOV BX,CJ1          ;  Read the address of the low byte
>                                             ;  in joint 2 counter word.
>0024  8A C3              MOV AL,BL           ;  Mask off high byte since only
>                                             ;  low byte is used.
>0026  EE                 OUT DX,AL           ;  Send it.
>0027  4A                 DEC DX              ;  Move to the input port.
>0028  EC                 IN  AL,DX           ;  Read the content of the counter in AL.
>0029  2D 0412            SUB AX,0412H        ;  Set up the origin of the theta motion.
>002C  A3 0004 R          MOV ENC2V,AX        ;  Transfer the data to FORTRAN.
>
>002F  58                 POP AX              ;  restore registers
>0030  5B                 POP BX              ;
>0031  5A                 POP DX              ;
>
>0032  8B E5              MOV SP,BP           ;  restore framepointer
>0034  5D                 POP BP              ;
>
>0035  CB                 RET                 ;  return
>0036                     ENCENDP
>0036                     CODEENDS
>                         END
```

```
The IBM Personal Computer Assembler 01-01-80       PAGE    Symbols-1
>ENC - ENCODER READING ROUTINE
>
>
>Segments and groups:
>
>           N a m e                 Size     align     combine        class
>
>CODE . . . . . . . . . . . . . .   0036     PARA      PUBLIC         'CODE'
>DGROUP . . . . . . . . . . . . .   GROUP
> DATA . . . . . . . . . . . . . .  0000     PARA      PUBLIC         'DATA'
> ENCBL$A. . . . . . . . . . . . .  0006     PARA      WORD           '$ENCBL'
>
>Symbols:
>
>           N a m e                 Type     Value     Attr
>
>CJ1. . . . . . . . . . . . . . .   L WORD   0000      ENCBL$A
>CJ2. . . . . . . . . . . . . . .   L WORD   0002      ENCBL$A
>ENC. . . . . . . . . . . . . . .   F PROC   0000      CODE           Global  Length
>ENC2V. . . . . . . . . . . . . .   L WORD   0004      ENCBL$A
>ENCM . . . . . . . . . . . . . .   Number   033F
>LP . . . . . . . . . . . . . . .   L NEAR   0006      CODE           Global
>OUTPORT. . . . . . . . . . . . .   Number   033D
>SLOTNO . . . . . . . . . . . . .   Number   0330
>
>Warning Severe
>Errors  Errors

>     0
#
```

```
The IBM Personal Computer Assembler 01-01-80        PAGE   1-1
>
>
>
>
>                        ;     ANALOG TO DIGITAL CONVERSION ROUTINE
>
>0000            DATA SEGMENT PUBLIC 'DATA'
>                        ;     For local assembler program data storage
>                        ;     Not used here, but required for linking
>0000            DATA ENDS
>
>0000            ADCBL$A SEGMENT COMMON '$ADCBL'
>
>                        ;     This segment is equivalent to COMMON/ADCBL
>                        ;     /ADCHN,ADCVAL
>                        ;     in FORTRAN. The $ and A are required to match the segment
>                        ;     name and class that the linker uses for the FORTRAN common.
>                        ;     ADCHN and ADCVAL must be declared INTEGER*2 in the
>                        ;     FORTRAN calling routine before the common block.
>
>0000  ????      ADCHN  DW  ?
>0002  ????      ADCVAL DW  ?
>
>0004            ADCBL$A ENDS
>
>                DGROUP GROUP DATA,ADCBL$A
>                        ;     The DATA and ADCBL$A segments will be
>                        ;     linked into the group called DGROUP,
>                        ;     to match Microsoft FORTRAN convention.
>
>0000            CODE SEGMENT PUBLIC 'CODE'
>
>                   ASSUME  CS:CODE,DS:DGROUP,SS:DGROUP
>                PUBLIC ADC                ;     Make ADC label available
>                                          ;     to other segments.
>
>                        ;---------------------------------------------
>                         SUBTTL  A/D CONVERSION
>                        ;---------------------------------------------
>                        ;     ADCHN is passed to ADC from the calling
>                        ;     program, and ADCVAL
>                        ;     is returned through the COMMON/ADCBL/ADCHN,
>                        ;     ADCVAL block.
>                        ;     FORTRAN "CALL ADC" calls this A/D procedure
>
>
>= 0330          LABTDR  EQU 816           ;     Address of Labtender in
>                                          ;     IBM-PC memory
>= 0331          ADDATA  EQU 817           ;     A/D Converter data register
```

```
The IBM Personal Computer Assembler 01-01-80      PAGE   1-2
>
>                    A/D CONVERSION
>
>
>
>0000              ADC PROC FAR
>
>0000  55          PUSH   BP               ;   Save calling framepointer
>
>0001  8B EC       MOV   BP,SP             ;   on the stack.
>
>0003  BA 0330        MOV    DX,LABTDR      ;   Indirect addressing using
>                                           ;   DX will have to
>                                           ;   be used when using IN
>                                           ;   and OUT commands.
>0006  A1 0000 R      MOV    AX,ADCHN       ;   Get channel number
>0009  B4 00          MOV     AH,0          ;   Mask off high byte, we
>                                           ;   only use AL
>
>                     ;   Now create the proper command byte for the Labtender's A/D
>                     ;   routine,based on the channel number.  Use the AL register.
>
>000B  D0 E0          SHL    AL,1           ;   Shift ADCHN left 1 bit
>                                           ;   (from bits 0-2 to 1-3)
>000D  24 3F          AND    AL,3FH         ;   Mask off bits 6 and 7
>000F  0C 01          OR     AL,1           ;   Put a 1 in bit 0 to start conversion
>0011  EE             OUT    DX,AL          ;   Send command to ADC command port

>
>                     ;   Now create a byte which will match the value in the
>                     ;   Labtender's status port when the A/D conversion is complete.
>
>0012  24 FE          AND    AL,0FEH        ;   Mask off error bit (bit 0),
>                                           ;   leave bits 1-6
>0014  0C 80          OR     AL,80H         ;   the same so we're still
>                                           ;   looking at the same
>                                           ;   channel, and put a 1 in bit 7
>                                           ;   (status bit -a 1 indicates that con-
>                                           ;   version is complete).
>0016  8A E0          MOV    AH,AL          ;   Save the reference for
>                                           ;   'conversion complete' in AH.
>
>0018                 CYCLE:
>0018  EC             IN    AL,DX           ;   Read status port.
>0019  3A C4          CMP    AL,AH          ;   Conversion complete?
>001B  75 FB          JNE    CYCLE          ;   Wait until it is.
>
>001D  BA 0331        MOV    DX,ADDATA      ;   Point to data port.
>0020  EC             IN    AL,DX           ;   Read data.
>0021  B4 00          MOV     AH,0          ;   Clear hi-byte of AX
>0023  2D 0080        SUB    AX,128         ;   Convert from unsigned
>                                           ;   TO SIGNED
```

```
The IBM Personal Computer Assembler 01-01-80      PAGE   1-3
>
>                      A/D CONVERSION
>
>                                                  ;   0 to 255 becomes -128 to 127
>0026  A3 0002 R    MOV   ADCVAL,AX                ;   Return A/D value to
>                                                  ;   calling program.
>
>0029  8B E5        MOV   SP,BP                    ;   Restore framepointer
>002B  5D           POP   BP
>002C  CB           RET                            ;   Return
>
>
>002D               ADCENDP
>
>002D               CODEENDS
>
>                   END
>
```

```
The IBM Personal Computer Assembler 01-01-80      PAGE   Symbols-1
>
>
>
>Segments and groups:
>
>          N a m e              Size     align    combine   class
>
>CODE . . . . . . . . . . . . . .   002D     PARA     PUBLIC    CODE
>DGROUP . . . . . . . . . . . . .   GROUP
> DATA . . . . . . . . . . . . . .  0000     PARA     PUBLIC    'DATA'
> ADCBL$A. . . . . . . . . . . . .  0004     PARA     WORD      '$ADCBL'
>
>Symbols:
>
>          N a m e              Type     Value    Attr
>
>ADC. . . . . . . . . . . . . . .   FPROC    0000     CODE      Global Length=002D
>ADCHN. . . . . . . . . . . . . .   LWORD    0000     ADCBL$A
>ADCVAL . . . . . . . . . . . . .   LWORD    0002     ADCBL$A
>ADDATA . . . . . . . . . . . . .   Number   0331
>CYCLE. . . . . . . . . . . . . .   LNEAR    0018     CODE
>LABTDR . . . . . . . . . . . . .   Number   0330
>
>Warning Severe
>Errors  Errors

>     0
#
```

```
 The IBM Personal Computer Assembler 01-01-80       PAGE  1-1
>DIGITAL TO ANALOG CONVERSION ROUTINE
>
>
>
>                 TITLE  DIGITAL TO ANALOG CONVERSION ROUTINE
>
>                 PAGE  ,132   ;Set page width to 132 characters.
>
>                     ;   This routine sends the digital to analog conversion value
>                     ;   (DACVAL) for a specified channel (DACHN).
>                     ;   FORTRAN "CALL DAC" calls this D/A procedure.
>
>0000             DATA SEGMENT PUBLIC 'DATA'
>                     ;   For local assembler program data storage
>                     ;   Not used here, but required for linking
>0000             DATA ENDS
>
>0000             DACBL$A SEGMENT COMMON '$DACBL'
>                     ;   This segment is equivalent to COMMON/DACBL/DACHN,
>                     ;   DACVAL in FORTRAN. The $ and A are required to match
>                     ;   the segment name and class that the linker uses for the
>                     ;   FORTRAN common.DACHN and DACVAL must be declared
>                     ;   INTEGER*2 in the FORTRAN calling routine before the
>                     ;   common block.
>
>0000  ????       DACHN  DW  ?
>0002  ????       DACVAL  DW  ?
>
>0004             DACBL$A ENDS
>
>                 DGROUP GROUP DATA,DACBL$A
>                     ;The DATA and DACBL$A segments will be
>                     ;linked into the group called DGROUP,
>                     ;to match Microsoft FORTRAN convention.
>
>0000             CODE SEGMENT PUBLIC 'CODE'
>
>                 ASSUME CS:CODE,DS:DGROUP,SS:DGROUP
>                 PUBLIC  DAC       ;Make DAC label available to other segments.
>
>= 0334           DACNTL  EQU  820        ;D/A control port location in memory
>= 0335           DADATA  EQU  821        ;D/A converter data register
>
>0000             DAC PROC FAR
>
>0000  55         PUSH   BP                ;Save calling framepointer
>0001  8B EC      MOV    BP,SP             ;on the stack.
>
>0003  BA 0335        MOV    DX,DADATA     ;Indirect addressing using DX will have to
>                                          ;be used when using IN and OUT commands.
>
>0006  8B 1E 0002 R      MOV    BX,DACVAL     ;Get D/A value
>000A  B7 00             MOV    BH,0          ;Mask off hi byte
```

```
The IBM Personal Computer Assembler 01-01-80       PAGE   1-2
>DIGITAL TO ANALOG CONVERSION ROUTINE
>
>
>
>000C  81 C3 0080       ADD    BX,128        ;Convert from signed to unsigned
>                                            ;-128 to 127 becomes 0 to 255
>
>0010  A1 0000 R        MOV    AX,DACHN      ;Get channel number
>0013  B4 00            MOV    AH,0          ;Mask off high byte, we only use AL
>                  ;  Now create the proper command byte for the Labtender's D/A
>                  ;  routine based on the channel number.  Use the AL register
>0015  24 0F            AND    AL,0FH        ;Mask of bits 4-7
>0017  0C 08            OR     AL,8          ;Set bit 3 hi
>0019  8A E0            MOV    AH,AL         ;Save D/A command byte in AH
>
>001B  8A C3            MOV    AL,BL         ;Get DACVAL (only lower byte)
>001D  EE               OUT    DX,AL         ;Load value into D/A data register
>001E  BA 0334          MOV    DX,DACNTL     ;Point to D/A control word
>0021  8A C4            MOV    AL,AH         ;Get D/A command byte
>0023  EE               OUT    DX,AL         ;Start conversion
>
>0024  8B E5            MOV    SP,BP         ;Restore framepointer
>0026  5D               POP    BP
>0027  CB               RET                  ;Return
>
>0028                   DAC ENDP
>
>0028                   CODE ENDS
>
>                       END
```

```
The IBM Personal Computer Assembler 01-01-80        PAGE   Symbols-1
>DIGITAL TO ANALOG CONVERSION ROUTINE
>
>
>Segments and groups:
>
>           N a m e          Size     align    combine      class
>
>CODE . . . . . . . . . . . . . .   0028     PARA     PUBLIC       'CODE'
>DGROUP . . . . . . . . . . . . .   GROUP
> DATA . . . . . . . . . . . . . .   0000     PARA     PUBLIC       'DATA'
> DACBL$A. . . . . . . . . . . . .   0004     PARA     WORD  '      $DACBL'
>
>Symbols:
>
>           N a m e          Type     Value    Attr
>
>DAC. . . . . . . . . . . . . . .   FPROC    0000     CODE   Global Length= 0028
>DACHN. . . . . . . . . . . . . .   LWORD    0000     DACBL$A
>DACNTL . . . . . . . . . . . . .   Number   0334
>DACVAL . . . . . . . . . . . . .   LWORD    0002     DACBL$A
>DADATA . . . . . . . . . . . . .   Number   0335
>
>Warning Severe
>Errors  Errors

>      0
#
```

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Thompson, B.S., and Sung, C.K. "A Variational Formulation for the Dynamic Viscoelastic Finite Element Analysis of Robotic Manipulators Constructed from Composite Materials," Presented at the *ASME Design and Production Engineering Conference*, Dearborn, MI, September 1983.

[2] Asada, H., Kanade, T., and Takeyama, I., "Control of a Direct Drive Arm," *in Robotics Research and Advanced Applications*, Book, W.J. (ed.), ASME Booklet, November 1982, pp. 63-72

[3] Kuntze, H.-B., and Jacubusch, A., "On the Closed-Loop Control of An Elastic Industrial Robot," *American Control Conference Proceedings*, June 1984, pp. 1217-1223.

[4] Zalucky, A., and Hardt, D.E., "Active Control of Robot Structure Deflections," *in Robotics Research and Advanced Applications*, Book, W.J. (ed.), ASME Booklet, November 1982, pp. 83-100.

[5] Paul, R.P., *Robot Manipulators-Mathematics, Programming and Control*, MIT Press, Cambridge, 1981.

[6] Vukobratovic, M., and Potkonjak, V., *Dynamics of Manipulation Robots*, Springer-Verlag, New York, 1982.

[7] Brady, M., *et al*, *Robot Motion: Planning and Control*, MIT Press, Cambridge, 1982.

[8] Turney, J.L., Mudge, T.N., and Lee, C.S.G., "Equivalence of Two Formulations for Robot Arm Dynamics," Report No. RSD-TR-3-82, Center for Robotics and Integrated Manufacturing, College of Engineering, University of Michigan, Ann Arbor, MI, December 1980.

[9] Turney, J.L., Mudge, T.N., and Lee, C.S.G., "Connection Between Formulations of Robot Arm Dynamics With Applications to Simulation and Control," Report No. RSD-TR-4-82, Center for Robotics and Integrated Manufacturing, College of Engineering, University of Michigan, Ann Arbor, MI, November 1981.

[10] Winfrey, R.C., "Elastic Link Mechanism Dynamics," *Journal of Engineering for Industry,* Transactions of the ASME, B 93 (1971), pp. 268-272.

[11] Sunada, W.H., and Dubowsky, S., "On the Dynamic Analysis and Behavior of Industrial Robotic Manipulators with Elastic Members," *ASME Paper No. 82-Det-45.*

[12] Shabana, A., and Wehage, R.A. "Variable Degree of Freedom Component Mode Analysis of Inertia Variant Flexible Mechanical Systems," *ASME Paper No 82-Det-93.*

[13] Turcic, D.A., and Midha, A., "Generalized Equations of Motion For The Dynamic Analysis of Elastic Mechanism Systems," *Journal of Dynamic Systems, Measurement and Control,* Transactions of the ASME, Vol. 106, December 1984, pp. 255-260.

[14] Turcic, D.A., and Midha, A., "Dynamic Analysis of Elastic Mechanism Systems. Part I: Applications," *Journal of Dynamic Systems, Measurement, and Control,* Transactions of the ASME, Vol. 106, December 1984, pp. 249-254.

[15] Turcic, D.A., Midha, A., and Bosnik, J.R., "Dynamic Analysis of Elastic Mechanism Systems. Part II: Experimental Results," *Journal of Dynamic Systems, Measurement and Control,* Transactions of the ASME, Vol. 106, December 1984, pp. 255-260.

[16] Wielenga, T.J., "Simplifications in the Simulation of Mechanisms Containing Flexible Members," Ph.D. Thesis, University of Michigan, March 1984.

[17] Book, W.J., Maizza-Neto, O., and Whitney, D.E., "Feedback Control of Two Beams, Two Joint Systems with Distributed Flexibility," *Journal of Dynamic Systems, Measurement and Control,* Transactions of the ASME, Vol. 97, No. 4, December 1975, pp. 424-431.

[18] Chalhoub, N.G., and Ulsoy, A.G. "Dynamic Simulation of a Flexible Robot Arm and Controller," *Report No. RSD-TR-13-83,* Center for Robotics and Integrated Manufacturing, College of Engineering, University of Michigan, Ann Arbor, MI, September 1983.

[19] Book, W.J., "Recursive Lagrangian Dynamics of Flexible Manipulator Arms Via Transformation Matrices," Report No. CMU-RI-TR-83-23, The Robotics Institute, Carnegie-Mellon University, Pittsburgh, PA, 1983.

[20] Dubowsky, S., and Desforges, D.T., "The Application of Model-Reference Adaptive Control to Robotic Manipulators," *Journal of Dynamic Systems, Measurement and Control,* Transactions of the ASME, Vol. 101, No. 3, September 1979, pp. 193-200.

[21] Horowitz, R., and Tomizuka, M., "An Adaptive Control Scheme for Mechanical Manipulators-Compensation of Nonlinearity and Decoupling Control," *ASME Paper No. 80-WA/DSC-6.*

[22] Landau, Y.D. *Adaptive Control Systems,* Marcel Dekker, 1979.

[23] Balestrino, A., De Maria, G., and Sciavicco, L., "An Adaptive Model Following Control for Robotic Manipulators," *Journal of Dynamic Systems, Measurement, and Control,* Transactions of the ASME, Vol. 105, September 1983, pp. 143-151.

[24] Lee, C.S.G., and Lee, B.H., "Resolved Motion Adaptive Control for Mechanical Manipulators," *Journal of Dynamic Systems, Measurement, and Control,* Transactions of the ASME, Vol. 106, June 1984, pp. 134-142.

[25] Donalson, D.D., and Leondes, C.T., "A model Referenced Parameter Tracking Technique for Adaptive Control Systems: Part I - The Principles of Adaptation and Part II - Stability Analysis by the Second Method of Lyapunov," *Transactions of the IEEE on Applications and Industry,* Vol. 82, No. 68, September 1963, pp. 241-262.

[26] Gilbert, E., and Ha, I.J. "An Approach to Nonlinear Feedback Control With Applications to Robotics," *IEEE Transactions on Systems, Man, and Cybernetics,* Vol. SMC-14, No. 6, November/December 1984, pp. 879-884.

[27] Sangveraphunsiri, V., and Book, W.J., "An Approach to the Minimum Time Control of a Simple Flexible Arm," *in Robotics Research and Advanced Applications,* Book, W.J. (ed.), ASME Booklet, November 1982, pp. 219-232.

[28] Kahn, M.E., and Roth, B., "The Near-Minimum Time Control of Open-Loop Articulated Kinematic Chains," *Journal of Dynamic Systems, Measurement and Control,* Transactions of the ASME, September 1971.

[29] Wang, S.S., and Sharma, A., "Dynamic Control Models of a Robot for High Speed and High Precision Assembly with Tolerance Constraints," *Proceedings of the 1984 American Control Conference,* Vol. 2, San Diego, California, 1984, pp. 872-877.

[30] Balas, M., "Feedback Control of Flexible Systems," *IEEE Transactions on Automatic Control,* Vol AC-23, No. 4, August 1978.

[31] Takahashi, Y., Rabins, M., and Auslander, D., *Control and Dynamic Systems,* Addison-Wesley, November 1972, pp. 431-452.

[32] Meckl, P.H., and Seering, W.P., "Active Damping in a Three-Axis Robotic Manipulator," Presented at the *Ninth ASME Vibration Conference,* Dearborn, MI, September 1983.

[33] Meckl, P.H., and Seering, W.P., "Minimizing Residual Vibration for Point-to-Point Motion," *ASME Paper No. 85-DET-17.*

[34] Usoro, P.B., Nadira, R., and Mahil, S.S., "Control of Lightweight Flexible Manipulators: A Feasibility Study," *Proceedings of the 1984 American Control Conference,* Vol. 2, San Diego, California, 1984, pp. 1209-1216.

[35] Fukuda, T., and Kuribayashi, Y., "Flexibility Control of Elastic Robotic Arms and its Application to Contouring Control," *IEEE Computer Society, International Conference on Robotics,* Atlanta, March 1984.

[36] Cannon, R.H., and Schmitz, E., "Initial Experiments on the End-Point Control of a Flexible One Link Robot," *The International Journal of Robotics Research,* November 1983.

[37] Book, W.J., and Majett, M., "Controller Design for Flexible, Distributed Parameter Mechanical Arms Via Combined State Space and Frequency Domain Techniques," *in Robotics Research and Advanced Applications,* Book, W.J. (ed.), ASME Booklet, November 1982, pp. 101-120.

[38] Cannon, R.H., *et al,* First Annual Report of the Center for Automation and Manufacturing Science, Stanford University, November 1983.

[39] Singh, S.N., and Schy, A.A., "Decomposition and State Variable Feedback Control of Elastic Robotic Systems," *Proceedings of the 1985 American Control Conference,* Vol. 2, Boston, MA, 1985, pp. 375-380.

[40] Song, J.O., and Haug, E.J., "Dynamic Analysis of Planar Flexible Mechanisms," *Computer Methods in Applied Mechanics and Engineering,* 24 (1980), pp. 359-381.

[41] Meirovitch, L., *Elements of Vibration Analysis,* McGraw Hill, New York 1975.

[42] Greenwood, D.T., *Principles of Dynamics,* Prentice Hall, New Jersey, 1965.

[43] Young, D., "Vibration of Rectangular Plates by the Ritz Method," *Journal of Applied Mechanics,* Vol. 72, December 1950, pp. 448-453.

[44] Young, D., and Felgar, R.P., "Tables of Characteristic Functions Representing Normal Modes of Vibration of a Beam," *The University of Texas Publication 4913,* July 11, 1949.

[45] Meirovitch, L., *Analytical Methods in Vibrations,* McGraw Hill, New York, 1967, pp. 440-445.

[46] Desai, C., and Abel, J., *Introduction to the Finite Element Method, A Numerical Method for Engineering Analysis,* Van Nostrand Reinhold, 1972.

[47] Shigley, J.E., *Mechanical Engineering Design,* McGraw Hill, New York, 1977.

[48] Marui, E., and Kato, S., "Forced Vibration of a Base-Excited Single-Degree-of-Freedom System with Coulomb Friction," *Journal of Dynamic Systems, Measurement, and Control,* Transactions of the ASME, Vol. 106, December 1984.

[49] D'Azzo, J., and Houpis, C.H., *Linear Control System Analysis and Design,* McGraw Hill, New York, 1981.

[50] Gear, C.W., "Ordinary Differential Equation System Solver," *University of Michigan Computing Center Report,* December 1974.

[51] Owens, D.H., *Multivariable and Optimal Systems,* Academic Press, 1981.

[52] Chen, C.T., *Introduction to Linear Systems Theory,* Holt, Rinehart and Winston, 1970.

[53] Kailath, T., *Linear Systems,* Prentice-Hall, 1980.

[54] Elbert, T.F., *Estimation and Control of Systems,* Van Nostrand Reinhold, 1984.

[55] Wonham, W.M., "On Pole Assignment in Multi-Input Controllable Linear Systems," *IEEE Transactions on Automatic Control,* Vol. AC-12, No. 6, December 1967, pp. 660-665.

[56] Moore, B.C., "On the Flexibility Offered by State Feedback in Multivariable Systems Beyond Closed Loop Eigenvalue Assignment," *IEEE Transactions on Automatic Control,* Vol. AC-21, October 1976, pp. 689-692.

[57] Chang, M.J., "Eigenstructure Assignment by State Feedback," *Proceedings of the 1984 American Control Conference,* Vol.1, 1984, pp. 372-377.

[58] Rangan, K.V., "Position and Velocity Measurement by Optical Shaft Encoders," Robotics Institute, Carnegie-Mellon University, Pittsburgh, Pennsylvania, June 1982.

[59] Desilva, C.W., "Motion Sensors in Industrial Robots," Mechanical Engineering, June 1985, pp. 40-51.

[60] Gervater, W.B., "Basic Relations for Control of Flexible Vehicles," A.I.A.A. Journal, Vol. 8, No. 4, April 1970, pp. 666-672.

[61] Holman, J.P., *Experimental Methods for Engineers,* McGraw-Hill, Third Edition, 1978.

[62] Hayt, W.H., and Kemmerly, J.E., *Engineering Circuit Analysis,* McGraw-Hill, Third Edition, 1978.

[63] Franklin, G.F., and Powell, J.D., *Digital Control of Dynamic Systems,* Addison-Wesley, June 1981.

[64] Astrom, K.J., and Wittenmark, B., *Computer Controlled Systems: Theory and Design,* Prentice Hall, 1984.