# SYNTHESIS TECHNIQUE FOR THE COMPENSATION OF ELECTROHYDRAULIC SERVOSYSTEMS WITH DYNAMIC PRESSURE FEEDBACK

By

KENNETH JOSEPH STOSS

Bachelor of Science

Kansas State University

Manhattan, Kansas

1977

Submitted to the Faculty of the Graduate College
of the Oklahoma State University
in partial fulfillment of the requirements
for the Degree of
MASTER OF SCIENCE
December, 1983

SYNTHESIS TECHNIQUE FOR THE COMPENSATION OF

ELECTROHYDRAULIC SERVOSYSTEMS WITH

DYNAMIC PRESSURE FEEDBACK

Thesis Approved:

Karl N. Reid
Thesis Adviser

Gary E. Young

James R. Rowland

Norman N. Durham
Dean of the Graduate College

## ACKNOWLEDGMENTS

I wish to express my appreciation to my major advisor, Dr. Karl Reid, for his guidance and assistance throughout my masters program. A note of appreciation goes out to my committee members, Dr. James Rowland and Dr. Gary Young, for their suggestions in the preparation of the final version of this thesis.

Dr. John Chandler, Dept. of Computer Science, was quite helpful with his advice on some of the computer aspects of this masters project. Mr. Sankaran Mohan also took the time to explain and work with me in learning to use the Oklahoma State University computer system. My thanks go to both of these gentlemen.

I also want to thank the Abex Corporation for supplying information on hydraulic components which they manufactured and were used in the course of this project.

Lastly and definitely not least, I wish to say thank you to my wife, Maureen. Her patience, understanding, and encouragement helped make my graduate work much easier. I am truly indebted to her and am appreciative of all that she has given to me.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

LIST OF FIGURES

Figure | Page

## NOMENCLATURE

| | |
|---|---|
| ABS | = Absolute Value |
| AE | = Cross-sectional area of spool exposed to $P_3$ and $P_4$ ($in^2$) |
| AF | = Cross-sectional area of spool exposed to $P_1$ and $P_2$ ($in^2$) |
| AG | = Area of mechanical dynamic pressure feedback piston exposed to $P_1$ and $P_2$ ($in^2$) |
| AP | = Area of mechanical dynamic pressure feedback piston exposed to $P_5$ and $P_6$ ($in^2$) |
| $\beta$ | = Bulk modulus of fluid ($lb_f/in^2$) |
| $C_D$ | = Orifice discharge coefficient |
| $C_F$ | = Friction coefficient of the actuator |
| $C_R$ | = Radial clearance between valve spool and sleeve (in) |
| $C_S$ | = Slip flow coefficient of the actuator |
| $C_{DM}$ | = Viscous drag coefficient of the actuator |
| $C_V$ | = Velocity coefficient of the steady state flow force |
| $D_1$ | = Diameter of flapper nozzle control orifice (in) |
| $D_2$ | = Diameter of valve spool (in) |
| $D_3$ | = Diameter of fixed orifice upstream from flapper nozzle (in) |
| $D_4$ | = Diameter of spool area exposed to $P_1$ and $P_2$ (in) |
| $D_5$ | = Diameter of mechanical dynamic pressure feedback piston exposed to $P_5$ and $P_6$ (in) |
| $D_6$ | = Diameter of mechanical dynamic pressure feedback piston exposed to $P_1$ and $P_2$ (in) |
| $D_7$ | = Diameter of flapper nozzle orifice for dynamic pressure feedback control (in) |
| $D_M$ | = Volumetric displacement of the actuator ($in^3$/rad) |

| | | |
|---|---|---|
| DSET | = | Minimum desired response settling time (secs) |
| $F_1$ | = | Pressure force acting on flapper nozzle valve ($lb_f$) |
| $F_2$ | = | Pressure force acting on flapper nozzle valve from mechanical dynamic pressure feedback unit ($lb_f$) |
| $F_S$ | = | Spring force ($lb_f$) |
| $F_T$ | = | Torquemotor force ($lb_f$) |
| $F_V$ | = | Viscous damping force on valve spool ($lb_f$) |
| $F_{SS1}$, $F_{SS2}$ | = | Steady state flow forces ($lb_f$) |
| FOBJ | = | Objective function for minimization |
| i | = | Subscript indicating initial conditions |
| I | = | Step input to the system (volts) |
| II | = | Error signal to servoamplifier (volts) |
| J | = | Polar moment of inertia of the actuator (in $lb_f$ $sec^2$/rad) |
| $K_1$ | = | Torquemotor constant ($lb_f$/ma) |
| $K_2$ | = | Torquemotor constant ($lb_f$/in) |
| $K_3$ | = | Spring constant of cantilever spring ($lb_f$/in) |
| $K_4$ | = | Spring constant of mechanical dynamic pressure feedback unit ($lb_f$/in) |
| $K_A$ | = | Servoamplifier gain (ma/volt) |
| $K_D$ | = | Dynamic pressure feedback gain (volts $in^2/lb_f$) |
| $K_F$ | = | Position feedback gain (volts/rad) |
| $K_P$ | = | Pressure feedback gain (volts $in^2/lb_f$) |
| $K_V$ | = | Servovalve gain (in/ma) |
| L | = | Distance between ports for unsteady flow forces (in) |
| $L_D$ | = | Spool length for viscous damping (in) |
| M | = | Mass of valve spool ($lb_f$ $sec^2$/in) |
| $P_1$, $P_2$ | = | Load line pressures ($lb_f/in^2$) |
| $P_3$, $P_4$ | = | Spool control pressures ($lb_f/in^2$) |

| | | |
|---|---|---|
| $P_5$, $P_6$ | = | Pressures of mechanical dynamic pressure feedback unit ($lb_f/in^2$) |
| $P_D$ | = | Dynamic pressure feedback ($lb_f/in^2$) |
| $P_E$ | = | Exhaust pressure ($lb_f/in^2$) |
| $P_M$ | = | Differential pressure of load lines ($lb_f/in^2$) |
| $P_S$ | = | Supply pressure ($lb_f/in^2$) |
| $\rho$ | = | Fluid density ($lb_f$ $sec^2/in^4$) |
| $Q_1$, $Q_2$, $Q_3$, $Q_4$ | = | Volumetric flow through the servovalve ($in^3/sec$) |
| $Q_A$, $Q_F$ | = | Volumetric flow through pilot stage fixed orifices ($in^3/sec$) |
| $Q_C$, $Q_D$ | = | Volumetric flow through flapper nozzles ($in^3/sec$) |
| $Q_H$, $Q_J$ | = | Volumetric flow from mechanical dynamic pressure feedback unit ($in^3/sec$) |
| $Q_M$ | = | Volumetric flow through servovalve, linearized model ($in^3/sec$) |
| s | = | Laplace variable, transformation with respect to time |
| $S_1$, $S_2$, $S_3$, $S_4$, $S_5$ | = | Switches used for simulating the uncompensated system, mechanical pressure feedback system, electrical pressure feedback system, mechanical dynamic pressure feedback system, and electrical dynamic pressure feedback system, respectively |
| $S_S$ | = | Static stiffness (in $lb_f$/rad) |
| $T_L$ | = | Load torque applied to actuator (in $lb_f$) |
| TSET | = | System response settling time (secs) |
| $\tau$ | = | Time constant of dynamic pressure feedback element (secs) |
| $\theta$ | = | Angular displacement of the actuator (rad) |
| $\mu$ | = | Fluid absolute viscosity ($lb_f$ $sec/in^2$) |
| $V_1$, $V_2$ | = | Volume of fluid under compression in right and left load lines ($in^3$) |
| Visd | = | Viscous drag external to the actuator (in $lb_f$ sec) |

$X$ = Displacement of the torquemotor (in)

$X_0$ = Displacement of flapper nozzle valve at null (in)

$Y$ = Displacement of the valve spool (in)

$Y_0$ = Effective length of the volume of oil at the right and left ends of spool (in)

$Z$ = Displacement of mechanical dynamic pressure feedback unit (in)

# CHAPTER I

# INTRODUCTION

## Background

Electrohydraulic servosystems are used in many controls applications requiring position control with fast dynamic response and high stiffness. Such servosystems basically consist of a servoamplifier, servovalve, actuator, transmission lines to connect the servovalve and actuator, and a feedback mechanism to produce a position type system as Figure 1 illustrates.

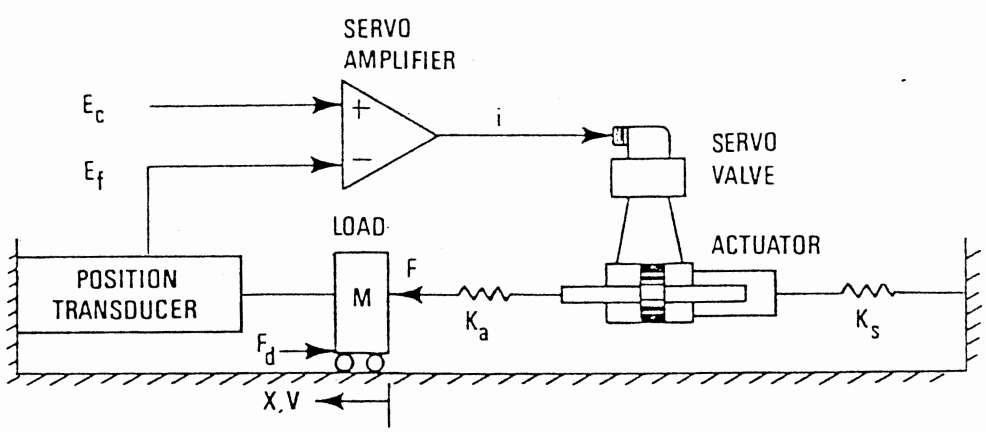


Figure 1. Schematic Diagram of Electrohydraulic Position Control Servosystem

Three important criteria used to define servosystem performance are speed of response, degree of stability, and static stiffness. Speed of response and degree of stability are dynamic performance measures which can be characterized by the system rise time and settling time respectively, assuming a time-domain step input to the system. Static stiffness is a direct function of the loop gain.

Servosystem designers often compromise on one or more of the performance criteria in order to obtain acceptable system performance. It is not unusual for electrohydraulic servosystems to be lightly damped. As the servosystem loop gain is increased to meet the static stiffness performance criterion, the speed of response improves as well but this occurs at the expense of degree of stability. Often, instability results for the system loop gains which are high enough to provide adequate static stiffness.

Electrohydraulic servovalves used in a position control system typically have minimum radial clearance and underlap or overlap. These servovalves operate near the origin of the steady-state valve characteristics (pressure-flow-displacement curves) when there is no external load force applied to the system actuator. Near the origin of the steady-state valve characteristics, the slope of the curves is essentially zero. This slope is an important factor which influences the damping or degree of stability of the servosystem; for low values of slope, low damping results unless damping is provided by other means. Typical valve characteristics are shown in Figure 2.

Various means of damping enhancement such as valve spool underlap, actuator by-pass leakage, and pressure feedback have been devised to obtain better dynamic performance. The greater the valve underlap, by-pass leakage, or pressure feedback, the steeper the effective slopes of the valve characteristic curves in the vicinity of the origin: a greater damping results. An example of the influence of valve underlap on the characteristic curves is shown in Figure 3. For all realizable values of pressure, flow, and displacement, the curves have a non-zero slope. For operation near the origin of the valve characteristic curves, the system damping is greater than in the case of a valve with minimum underlap.

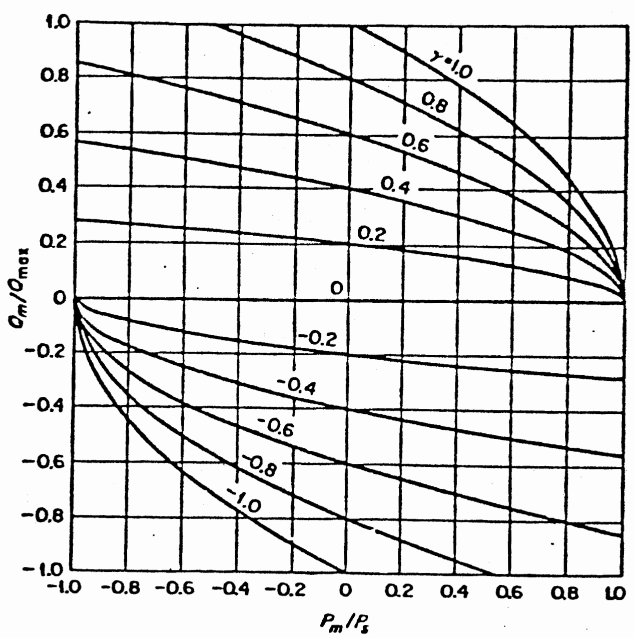


Figure 2. Steady-State Valve Characteristics, Zero-Lap Valve

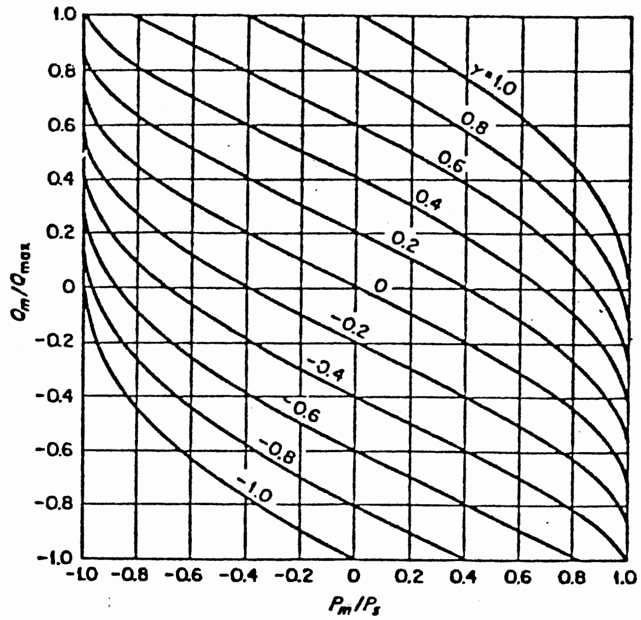


Figure 3. Steady-State Valve Characteristics, Maximum Valve Underlap

The introduction of damping enhancement in the system improves the degree of stability but often compromises other performance measures. Spool underlap is a simple means of enhancing damping, but it results in quiescent power loss. Flow passes through the system regardless of demand. Actuator by-pass leakage enhances damping also, but at the expense of reduction in power delivered to the load. Addition of external load damping results in the same effect. Pressure feedback produces enhanced damping without quiescent power loss. A loss of static stiffness results with the implementation of any of the above damping enhancement methods.

Advancements have been made in electrohydraulic servosystem performance through the utilization of dynamic pressure feedback.

Realizable increases in performance can be produced with dynamic pressure feedback if the resulting complexity can be accepted. The appealing feature of dynamic pressure feedback is that it is active only during the transient period when damping is required. Feedback is attenuated or non-existant during steady-state operation. Thus, static stiffness is not affected but stability is enhanced. Dynamic pressure feedback is functionally a high pass filter.

Moog, Inc., has done much work with electrohydraulic servosystem design as documented by Geyer (5). Measured frequency responses for electrohydraulic position control systems with and without damping enhancement are shown in Figures 4 through 7. In all four systems, the amplifier gain was set to achieve a peak amplitude ratio of 1.25 (±2 db). In the last three cases additional damping was introduced to produce an equivalent load damping ratio of 0.6. Static stiffness for those four systems was measured by applying an external force and measuring the load deflection. A summary of results from the tests can be found in Table I. The servosystem which utilized dynamic pressure feedback produced the best dynamic and static performance.

Limited information is available in the open literature concerning the optimum design of systems with dynamic pressure feedback. Geyer (5) discussed static stiffness determination through loop gain adjustment and suggested setting the dynamic pressure feedback element time constant such that the corner frequency is about one-third the actuator-load natural frequency. Morse (9) recommended setting the corner frequency of the feedback element about a decade below the actuator-load natural frequency. An objective of this thesis was to develop a logical procedure for selecting the time constant.

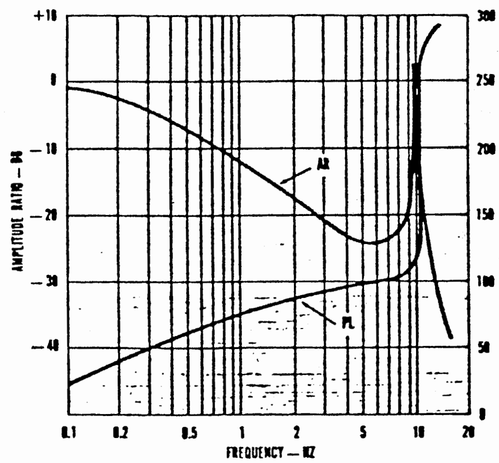


Figure 4. Measured System Response With Flow Control Servovalve

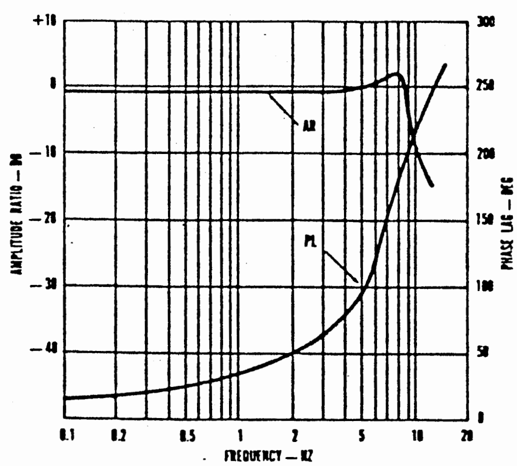


Figure 5. Measured System Response With Flow Control Servovalve and Bypass Orifice

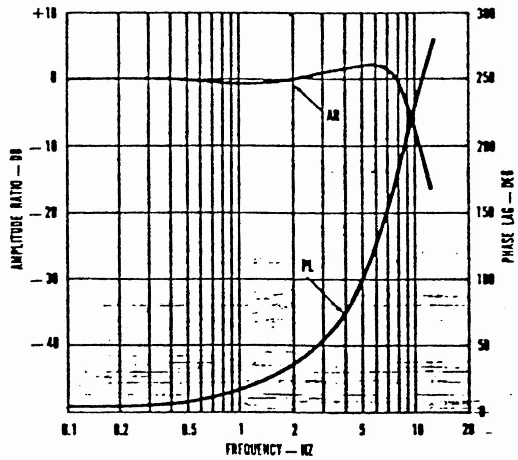


Figure 6. Measured System Response With PQ Servovalve

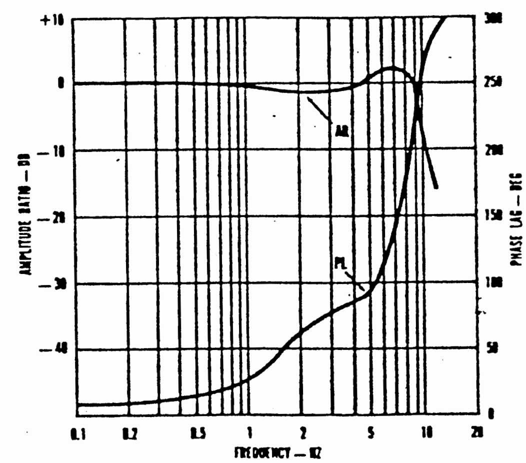


Figure 7. Measured System Response With DPF Servovalve

TABLE I

VARIOUS POSITION SERVOS PERFORMANCE COMPARISON

| Servo Configuration | Bandwidth (±2db) hz | 90° Phase Lag hz | Static Load Stiffness $lb_f$/in |
|---|---|---|---|
| Flow Control Servovalve | 0.15 | 0.37 | 9,000 |
| Flow Control Servovalve With Bypass Orifice | 8.8 | 5.0 | 5,100 |
| PQ Servovalve | 8.8 | 5.0 | 2,500 |
| DPF Servovalve | 9.2 | 5.0 | 60,000 |

## Problem Statement and Scope of Study

Through the course of this study it was assumed that the basic electrohydraulic position control servosystem had already been designed and the only parameter of that servosystem which remained undetermined was the loop gain. Further, it was assumed that the system was lightly damped such that when the loop gain was increased to provide adequate static stiffness, the system transient response was too oscillatory to be useful.

The problem was to develop a logical procedure for synthesizing a dynamic pressure feedback network for compensating a lightly damped, electrohydraulic position control servosystem. The synthesis required the determination of the servosystem loop gain as well as the feedback network time constant and gain such that the three important performance criteria, static stiffness, speed of response, and degree of stability, were satisfied.

The scope of this study included:

1. Derivation of a mathematical model of the existing servosystem. Due to the nature of the physical processes involved, the model includes non-linear and linear algebraic and differential equations.
2. Computer simulation of the basic servosystem utilizing the non-linear mathematical model. The model was structured such that the user could add mechanical and electrical pressure feedback as well as mechanical and electrical dynamic pressure feedback.
3. Computer simulation of the basic system utilizing a linearized mathematical model. This model was structured such that the user could add mechanical and electrical pressure feedback as well as mechanical and electrical dynamic pressure feedback.
4. Validation of the models via laboratory measurements. An actual system was available in the School of Mechanical and Aerospace Engineering Systems Laboratory.
5. Development of a procedure to synthesize a dynamic pressure feedback network to enhance the damping of a lightly damped, electrohydraulic, position control servosystem.

## Summary

A synthesis technique was developed to add dynamic pressure feedback to a lightly damped, electrohydraulic position control servosystem. In this technique the static stiffness is first satisfied and then the pressure feedback gain and time constant of the dynamic pressure feedback element are sized to satisfy the dynamic performance criteria. It is assumed that the parameters required can be achieved in actual hardware; otherwise design compromises have to be made.

The synthesis procedure is sequential. At each step of the synthesis procedure, results are compared with design specifications. If the specifications cannot be satisfied, design compromises must be made. The steps in the procedure are:

1. The loop gain is determined such that the static stiffness performance criterion is satisfied.
2. The pressure feedback gain is determined to provide the maximum degree of stability for the system (this performance measure is characterized by the transient response settling time). For reasons explained in Chapter III, the feedback element time constant is set to infinity for this determination.
3. The time constant is reduced from infinity until the response settling time is equal to the maximum allowable value. Evaluate the transient speed of response (performance measure characterized by response rise time).

The feedback element corner frequency (reciprocal of time constant) range of one-third to one-tenth the actuator-load open-loop natural frequency suggested by Geyer (5) and Morse (9) did not have any particular significance in the above procedure. It appears that the time constant range could be attributed to practical limitations of hardware implementation using a mechanical, dynamic pressure feedback network.

# CHAPTER II

# MODEL FORMULATION

## Modelling Assumptions

A system model can become unduly complex unless simplifying assumptions are made which eliminate higher order effects. Care must be exercised in making assumptions so that a loss of pertinent information does not occur. The assumptions employed in developing the models are:

1. The valve is symmetrical with no underlap or overlap.
2. The steady-state orifice equation holds for each orifice and all discharge coefficients are constant and equal.
3. All connecting passages are short in length and large in diameter, i.e., resistance and "transmission line" effects are negligble.
4. The supply and exhaust pressures are constant.
5. The temperature, viscosity, and bulk modulus of the fluid are constant.
6. The change of fluid density is small compared to the density of the fluid itself. The time rate of change in density is not negligble, i.e., compressibility effects are important.
7. Static equations describe the torquemotor and mechanical high pass filter because their dynamics are of high enough order to be considered insignificant.

8. The steady-state flow force jet angle is assumed to be constant.

## System Equations

Five electrohydraulic servosystem models are developed in this section. All five models describe the same basic servosystem. One model incorporates no means of damping enhancement. The other four models incorporate damping enhancement via pressure or dynamic pressure feedback. The basic servosystem is described first since it is the basis for the remaining models. The additions and changes to the basic system equations required to describe the other four models are documented separately. Terms used in describing equations can be referenced in the section titled "Nomenclature".

### Basic Servosystem

A schematic diagram of the basic servosystem is shown in Figure 8. The describing equations are as follows:

Servoamplifier error signal:

$$II = I - K_F \cdot \theta \tag{2.1}$$

Pilot stage flapper-nozzle valve force:

$$F_1 = 0.25 \cdot \pi \cdot D_1^2 \cdot (P_3 - P_4) \tag{2.2}$$

Cantilever spring force:

$$F_S = K_3 \cdot (X - Y) \tag{2.3}$$

Torquemotor force:

$$F_T = K_1 \cdot II - K_2 \cdot X \tag{2.4}$$

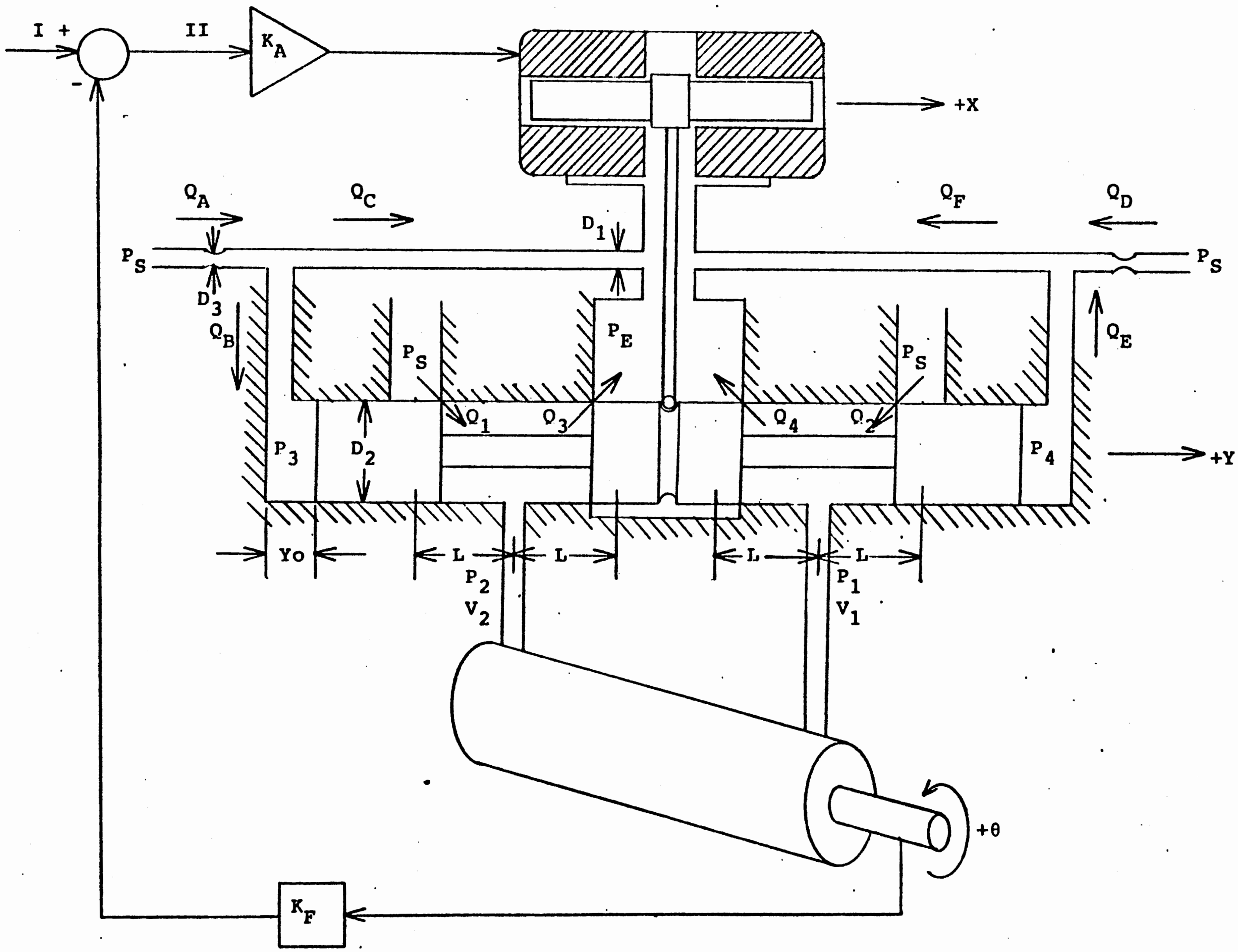


Figure 8. Schematic Diagram of Basic Servosystem

Flapper-nozzle valve orifice flow rates:

$$Q_A = 0.25 \cdot \pi \cdot D_3^2 \cdot C_D \cdot (2/\rho)^{1/2} \cdot (P_S - P_3)^{1/2} \tag{2.5}$$

$$Q_C = \pi \cdot D_1 \cdot C_D \cdot (2/\rho)^{1/2} \cdot (X_0 + X) \cdot (P_3 - P_E)^{1/2} \tag{2.6}$$

$$Q_F = \pi \cdot D_1 \cdot C_D \cdot (2/\rho)^{1/2} \cdot (X_0 - X) \cdot (P_4 - P_E)^{1/2} \tag{2.7}$$

$$Q_D = 0.25 \cdot \pi \cdot D_3^2 \cdot C_D \cdot (2/\rho)^{1/2} \cdot (P_S - P_4)^{1/2} \tag{2.8}$$

Spool end chamber continuity:

$$\frac{dP_3}{dt} = \frac{\beta}{(Y_0 + Y)}\left(\frac{Q_A - Q_C}{AE} - \frac{dY}{dt}\right) \tag{2.9}$$

$$\frac{dP_4}{dt} = \frac{\beta}{(Y_0 - Y)}\left(\frac{Q_D - Q_F}{AE} + \frac{dY}{dt}\right) \tag{2.10}$$

Spool valve flow rates for $X > 0$:

$$Q_2 = \pi \cdot D_2 \cdot C_D \cdot Y \cdot [2 \cdot (P_S - P_1)/\rho]^{1/2} \tag{2.11}$$

$$Q_3 = \pi \cdot D_2 \cdot C_D \cdot Y \cdot [2 \cdot (P_2 - P_E)/\rho]^{1/2} \tag{2.12}$$

Spool valve flow rates for $X < 0$:

$$Q_1 = -\pi \cdot D_2 \cdot C_D \cdot Y \cdot [2 \cdot (P_S - P_2)/\rho]^{1/2} \tag{2.13}$$

$$Q_4 = -\pi \cdot D_2 \cdot C_D \cdot Y \cdot [2 \cdot (P_1 - P_E)/\rho]^{1/2} \tag{2.14}$$

Actuator chamber continuity:

$$\frac{dP_1}{dt} = \frac{\beta}{V_1}\left(D_M \cdot \frac{d\theta}{dt} + \frac{C_S \cdot D_M \cdot P_M}{\mu} - Q_{2,4}\right) \tag{2.15}$$

$$\frac{dP_2}{dt} = \frac{\beta}{V_2}\left(Q_{3,1} - D_M \cdot \frac{d\theta}{dt} - \frac{C_S \cdot D_M \cdot P_M}{\mu}\right) \tag{2.16}$$

Viscous damping force:

$$F_V = \frac{\pi \cdot \mu \cdot D_2 \cdot L_D}{C_R} \cdot \frac{dY}{dt} \quad (2.17)$$

Steady-state flow forces on the valve spool:

$$F_{SS1} = -C_V \cdot C_D \cdot \pi \cdot D_2 \cdot Y \cdot (P_1 - P_E + P_M) \cdot \cos 69° \quad (2.18)$$

$$F_{SS2} = -C_V \cdot C_D \cdot \pi \cdot D_2 \cdot Y \cdot (P_1 - P_E - P_M) \cdot \cos 69° \quad (2.19)$$

Unsteady flow forces on the valve spool for $X > 0$:

$$F_{US1} = \rho^{\frac{1}{2}} L C_D \pi D_2 \{[2(P_S - P_1)]^{\frac{1}{2}} \cdot \frac{dY}{dt} - \frac{1}{2} \cdot Y[2(P_S - P_1)]^{-\frac{1}{2}} \cdot \frac{dP_1}{dt}\} \quad (2.20)$$

$$F_{US2} = -\rho^{\frac{1}{2}} L C_D \pi D_2 \{[2(P_2 - P_E)]^{\frac{1}{2}} \cdot \frac{dY}{dt} + \frac{1}{2} Y[2(P_2 - P_E)]^{-\frac{1}{2}} \cdot \frac{dP_2}{dt}\} \quad (2.21)$$

Unsteady flow forces on the valve spool for $X < 0$:

$$F_{US1} = -\rho^{\frac{1}{2}} L C_D \pi D_2 \{[2(P_1 - P_E)]^{\frac{1}{2}} \cdot \frac{dY}{dt} + \frac{1}{2} Y[2(P_1 - P_E)]^{-\frac{1}{2}} \cdot \frac{dP_1}{dt}\} \quad (2.22)$$

$$F_{US2} = \rho^{\frac{1}{2}} L C_D \pi D_2 \{[2(P_S - P_2)]^{\frac{1}{2}} \cdot \frac{dY}{dt} - \frac{1}{2} Y[2(P_S - P_2)^{-\frac{1}{2}} \cdot \frac{dP_2}{dt}\} \quad (2.23)$$

Valve spool force balance:

$$M \cdot \frac{d^2Y}{dt^2} = F_S + (P_3 - P_4) \cdot AE + F_{US1} + F_{US2} + F_{SS1} + F_{SS2} - F_V \quad (2.24)$$

Servoactuator torque balance:

$$J \cdot \frac{d^2\theta}{dt^2} = (1 - C_F) \cdot P_M \cdot D_M - (C_{DM} \cdot D_M \cdot \mu + Visd) \cdot \frac{d\theta}{dt} \quad (2.25)$$

Basic Servosystem With Mechanical Pressure Feedback

A schematic diagram of the basic servosystem with mechanical pressure feedback is shown in Figure 9. The system model is

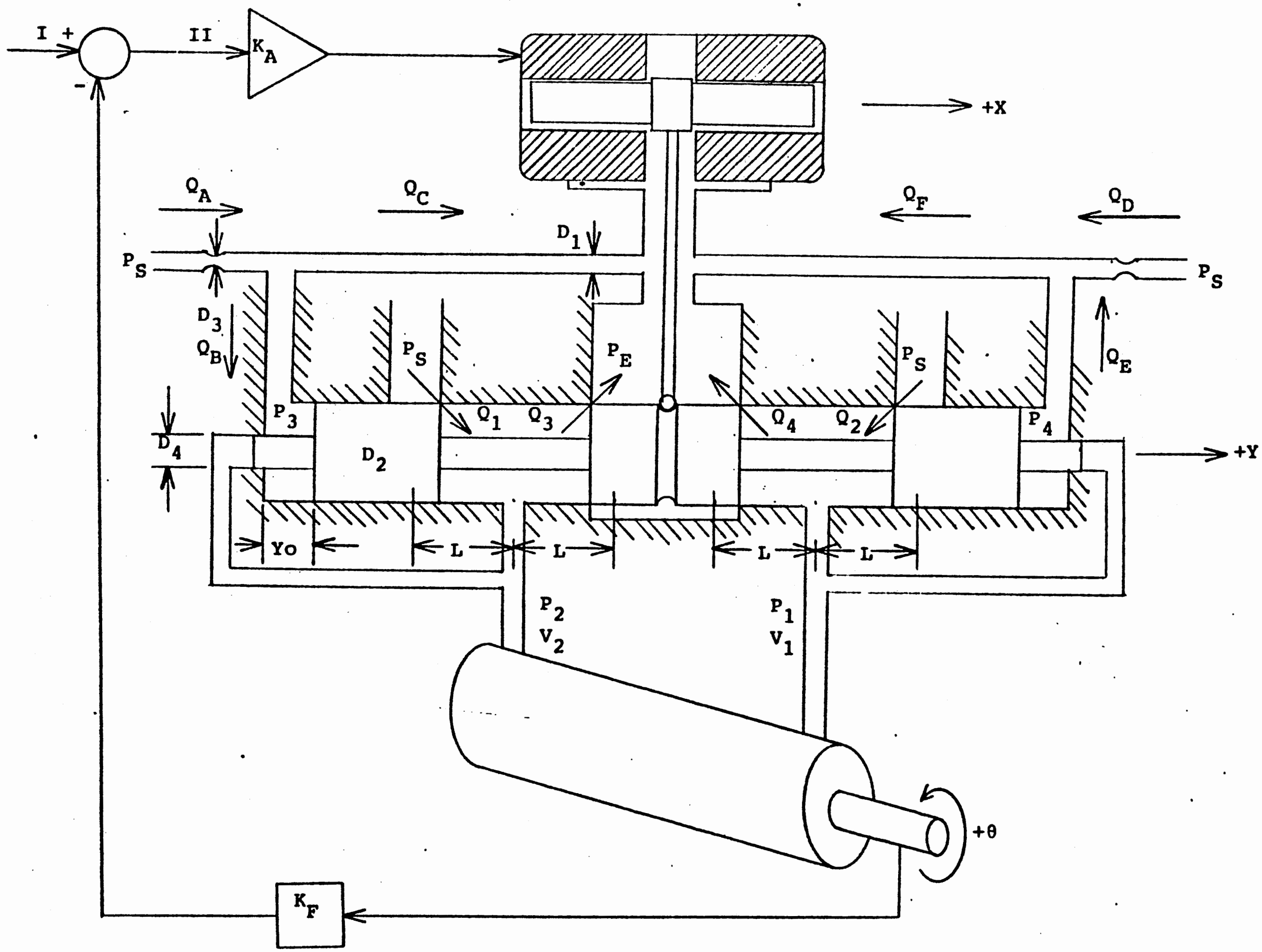


Figure 9. Schematic Diagram of Basic Servosystem With Mechanical Pressure Feedback

identical to that of the basic servosystem except for the pressure feedback effect. Equation (2.24) must be modified to include the load differential pressure, $P_M$, acting on the valve spool force balance as follows:

$$M \cdot \frac{d^2Y}{dt^2} = F_S + (P_3 - P_4)AE + P_M AF + F_{US1} + F_{US2} + F_{SS1} + F_{SS2} - F_V \quad (2.24a)$$

Equations (2.1) through (2.25), excepting (2.24), remained unchanged to describe the servosystem with mechanical pressure feedback.

## Basic Servosystem With Electrical Pressure Feedback

Equations (2.2) through (2.25) are utilized to describe the servosystem with electrical pressure feedback. The error signal to the amplifier, equation (2.1), must be modified to include an error term proportional to the load differential pressure. The modified equation is as follows:

$$II = I - K_F \cdot \theta - K_P \cdot P_M \quad (2.1a)$$

A schematic diagram showing the servosystem with electrical pressure feedback is shown in Figure 10.

## Basic Servosystem With Mechanical Dynamic Pressure Feedback

In this case, a "mechanical" high pass filter is added to the basic servosystem as shown schematically in Figure 11. Equations (2.1) through (2.25) are used for this system alteration.

Mechanical dynamic pressure feedback results in the addition of another force on the flapper as follows:

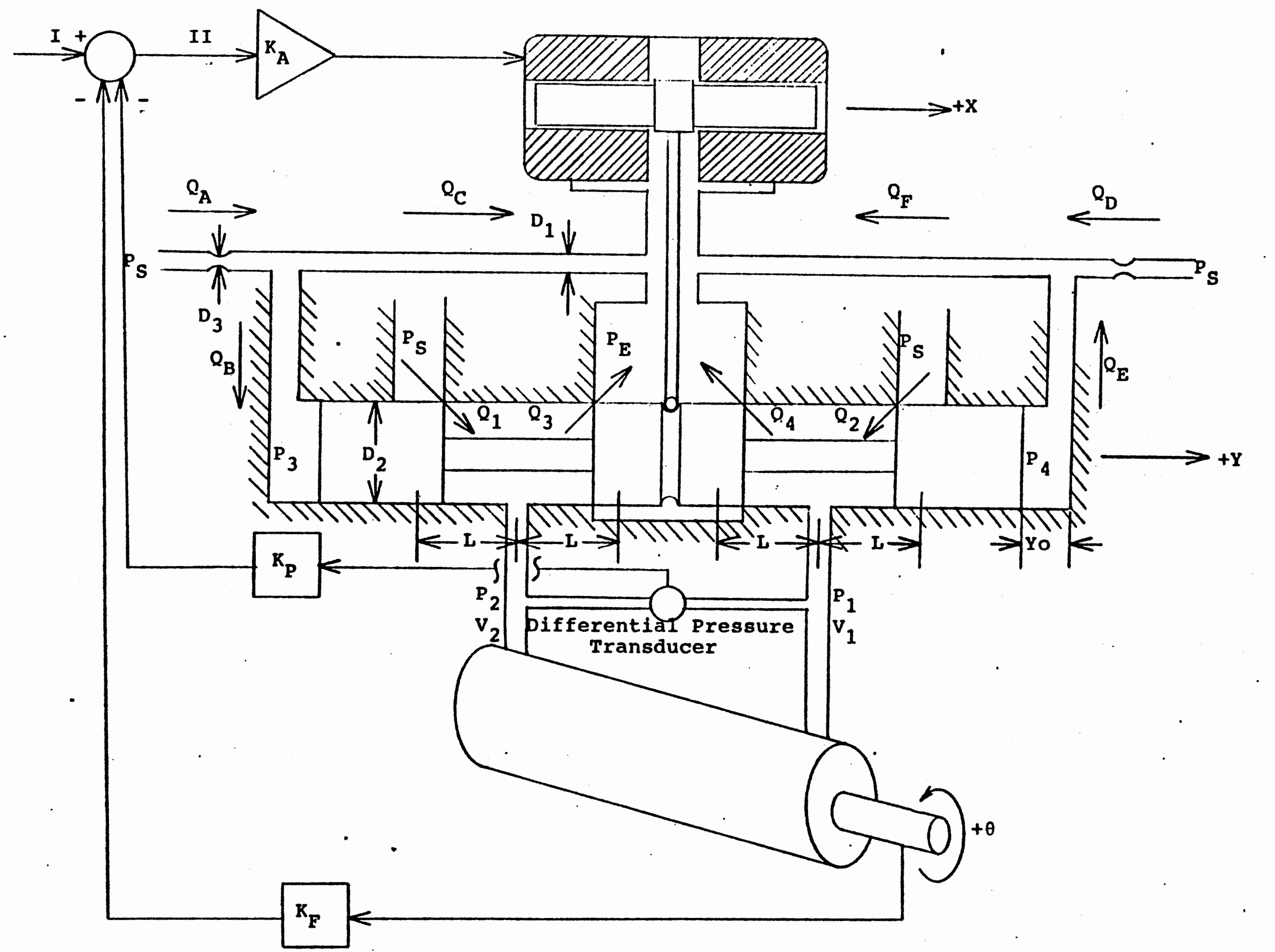


Figure 10. Schematic Diagram of Basic Servosystem With Electrical Pressure Feedback

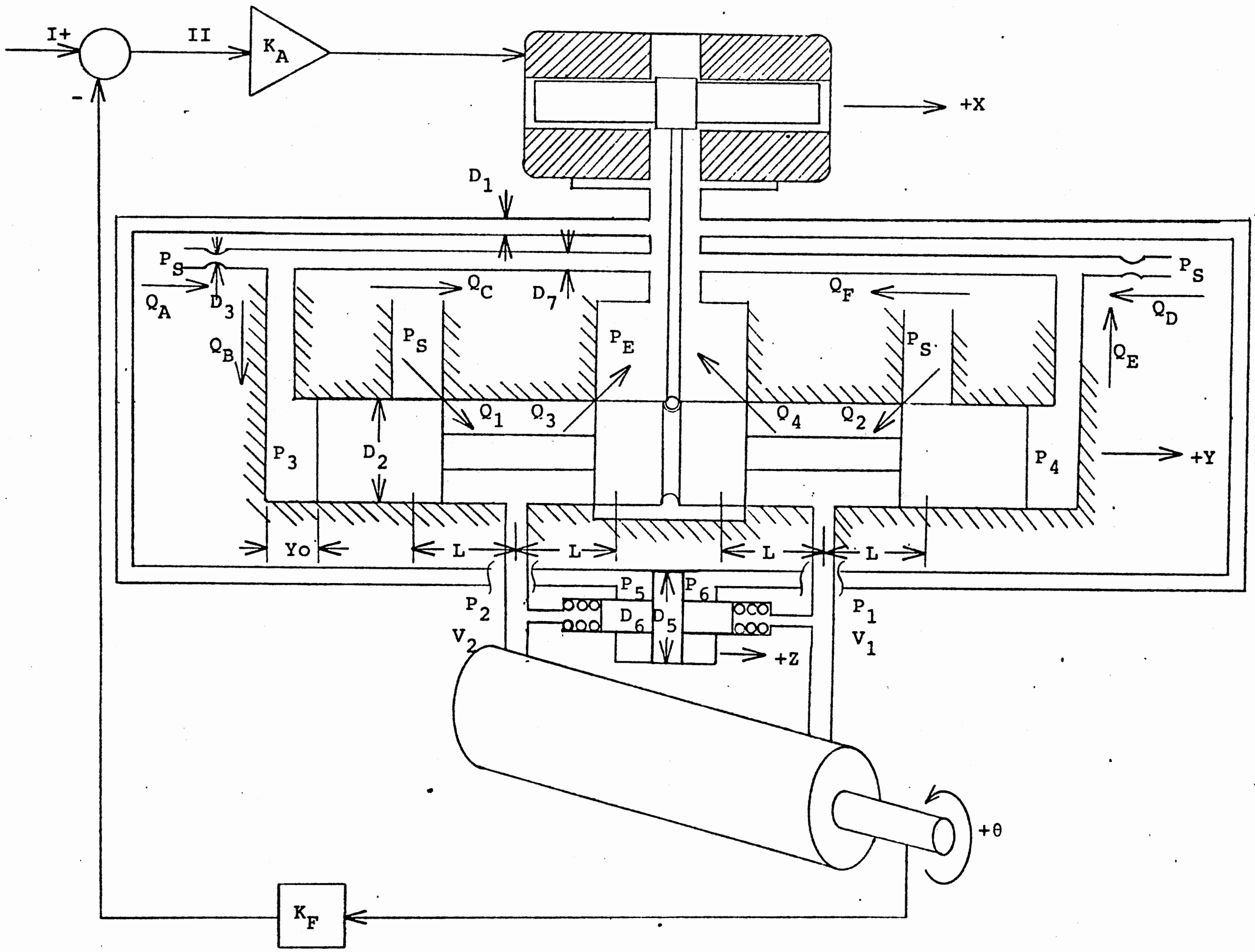


Figure 11. Schematic Diagram of Servosystem With Mechanical Dynamic Pressure Feedback

$$F_2 = 0.25 \cdot \pi \cdot D_7^2 \cdot (P_5 - P_6) \qquad (2.26)$$

Five additional equations must be added to the basic servosystem model in order to describe the filter. The flow rates into and out of the mechanical high pass filter are described by

$$Q_H = \pi \cdot D_7 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (X_0 + X) \cdot (P_5 - P_E)^{\frac{1}{2}} \text{ , and} \qquad (2.27)$$

$$Q_J = \pi \cdot D_7 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (X_0 - X) \cdot (P_6 - P_E)^{\frac{1}{2}} \qquad (2.28)$$

The velocity of the filter piston is

$$\frac{dZ}{dt} = \frac{Q_H}{AP} = \frac{-Q_J}{AP} \qquad (2.29)$$

The force balance on the filter piston for $\frac{dZ}{dt} > 0$ is

$$P_6 \cdot AP = P_M \cdot AG - 2 \cdot K_4 \cdot Z \qquad (2.30)$$

The force balance on the filter piston for $\frac{dZ}{dt} < 0$ is

$$P_5 \cdot AP = 2 \cdot K_4 \cdot Z - P_M \cdot AG \qquad (2.31)$$

<u>Basic Servosystem With Electrical Dynamic Pressure Feedback</u>

A schematic diagram of the servosystem compensated with electrical dynamic pressure feedback is shown in Figure 12. Equations (2.2) through (2.25) are used to describe this system. The error signal to the servoamplifier must be altered to include an additional term. This term is proportional to the feedback passed through the filter. The modified error signal is

$$II = I - K_F \cdot \theta - K_D \cdot P_D \qquad (2.1b)$$

An additional equation required to describe the electrical high

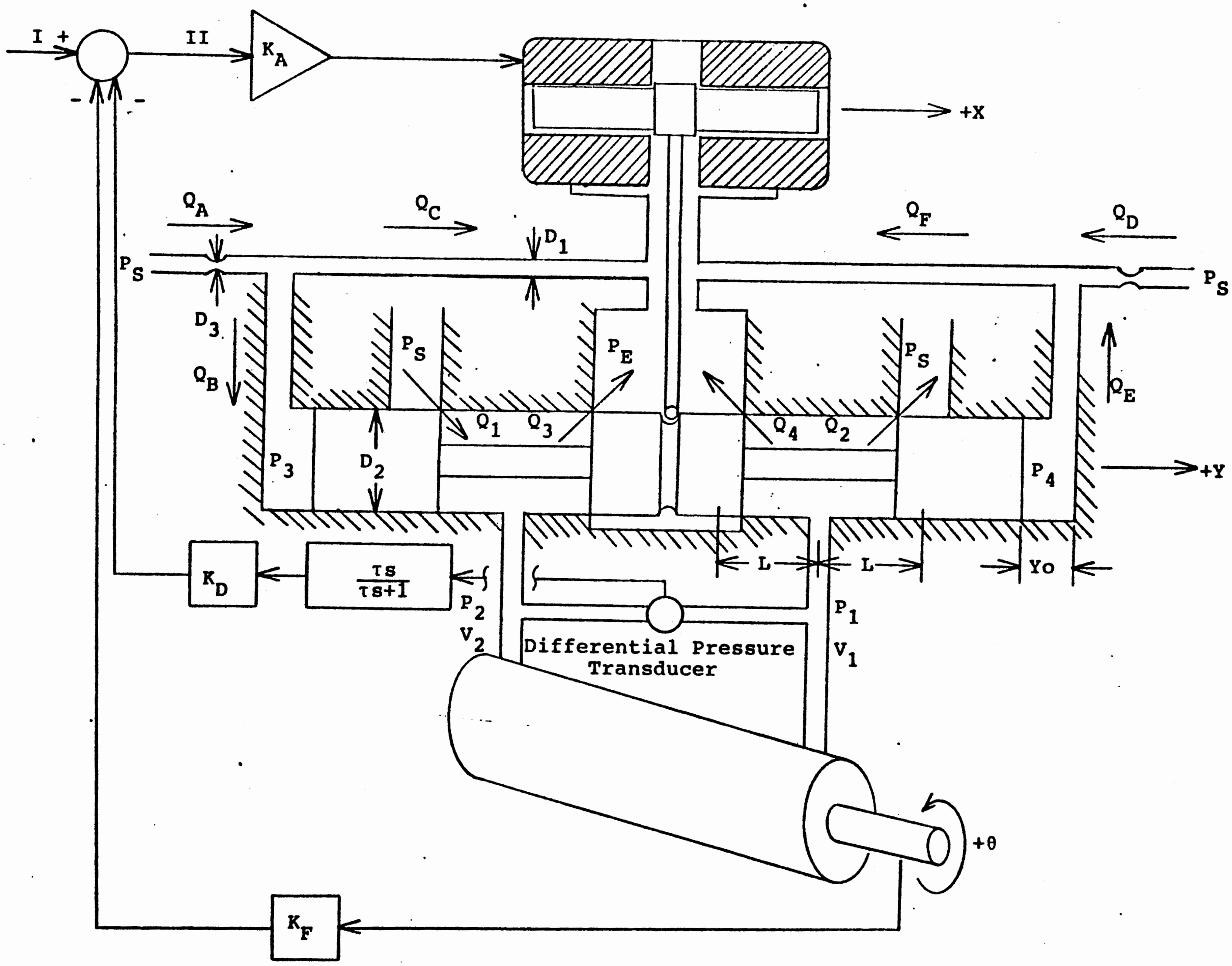


Figure 12. Schematic Diagram of Servosystem With Electrical Dynamic Pressure Feedback

pass filter is

$$P_D = e^{-t/\tau} \cdot \frac{dP_M}{dt} \quad (2.32)$$

## Linearized Equations

Non-linear system synthesis is not a well established discipline. Most control system synthesis techniques are based on linear system concepts. The synthesis procedure developed in Chapter III is based on a linear system model.

The linearized equations which describe each of the five models are presented separately below. The method used to obtain the linearized equations about the operating point as well as the definition of each linearization constant can be found in Appendix A.

### Basic Servosystem

Servoamplifier error signal:

$$II = I + C_1 \cdot \theta \quad (2.33)$$

Pilot stage flapper-nozzle valve force:

$$F_1 = C_2 \cdot (P_3 - P_4) \quad (2.34)$$

Cantilever spring force:

$$F_S = C_{23} \cdot (X - Y) \quad (2.35)$$

Torquemotor force:

$$F_T = K_1 \cdot II - K_2 \cdot X \quad (2.36)$$

Flapper-nozzle valve orifice flow rates:

$$Q_A = C_6 \cdot P_3 + C_7 \quad (2.37)$$

$$Q_C = C_8 \cdot X + C_9 \cdot P_3 + C_{10} \quad (2.38)$$

$$Q_F = C_{11} \cdot X + C_{12} \cdot P_4 + C_{13} \quad (2.39)$$

$$Q_D = C_{14} \cdot P_4 + C_{15} \quad (2.40)$$

Spool end chamber continuity:

$$sP_3 = C_{34} \cdot Y + C_{35} \cdot (Q_A - Q_C) + C_{36} \cdot sY + C_{37} \quad (2.41)$$

$$sP_4 = C_{38} \cdot Y + C_{39} \cdot (Q_D - Q_F) + C_{40} \cdot sY + C_{41} \quad (2.42)$$

Spool valve flow rate:

$$Q_M = C_{69} \cdot Y + C_{70} \cdot P_M \quad (2.43)$$

Actuator chamber continuity:

$$sP_1 = C_{42} \cdot s\theta + C_{43} \cdot P_M + C_{44} \cdot Q_M \quad (2.44)$$

$$sP_2 = C_{45} \cdot s\theta + C_{46} \cdot P_M + C_{47} \cdot Q_M \quad (2.45)$$

Viscous damping force:

$$F_V = C_{65} \cdot sY \quad (2.46)$$

Steady-state flow forces on the valve spool:

$$F_{SS1} = C_{16} \cdot Y + C_{17} \cdot P_M + C_{18} \quad (2.47)$$

$$F_{SS2} = C_{19} \cdot Y + C_{20} \cdot P_M + C_{21} \quad (2.48)$$

In a well designed valve the unsteady flow forces effectively cancel one another. The valve utilized in this study was so designed. To avoid additional complication in the linearized model the unsteady flow forces were omitted.

Valve spool force balance:

$$M \cdot s^2 Y = F_S + AE \cdot (P_3 - P_4) + F_{SS1} + F_{SS2} - F_V \quad (2.49)$$

Servoactuator torque balance:

$$J \cdot s^2 \theta = (1 - C_F) \cdot D_M \cdot P_M - (C_{DM} \cdot D_M \cdot \mu + Visd) \cdot s\theta \quad (2.50)$$

Basic Servosystem With Mechanical Pressure Feedback

The linear system utilizes equations (2.33) through (2.50) except that equation (2.49) must be modified to include the load differential pressure in the valve spool force balance. The modified force balance is

$$M \cdot s^2 Y = F_S + AE \cdot (P_3 - P_4) + AF \cdot P_M + F_{SS1} + F_{SS2} - F_V \qquad (2.49a)$$

Basic Servosystem With Electrical Pressure Feedback

Equations (2.34) through (2.35) are employed in the linear electrical pressure feedback system model. The error signal to the amplifier, equation (2.33), must be modified to include the pressure feedback effect. The modified error equation is

$$II = I + C_1 \cdot \theta + C_{31} \cdot P_M \qquad (2.33a)$$

Basic Servosystem With Mechanical Dynamic Pressure Feedback

When the "mechanical" high pass filter is added to the servosystem equations (2.33) through (2.50) which describe the basic servosystem remain unchanged. The following equations must be included for the mechanical dynamic pressure feedback system model:

The additional force on the flapper is

$$F_2 = C_{33} \cdot (P_5 - P_6) \qquad (2.51)$$

The flow rates into and out of the mechanical dynamic high pass filter are

$$Q_H = C_{59} \cdot X + C_{60} \cdot P_5 + C_{61} \tag{2.52}$$

$$Q_J = C_{62} \cdot X + C_{63} \cdot P_6 + C_{64} \tag{2.53}$$

The velocity of the filter piston is

$$sZ = \frac{Q_H}{AP} = \frac{-Q_J}{AP} \tag{2.54}$$

The force balance on the filter piston for $\frac{dZ}{dt} > 0$ is

$$P_5 \cdot AP = 2 \cdot K_4 \cdot Z - P_M \cdot AG \tag{2.55}$$

The force balance on the filter piston for $\frac{dZ}{dt} < 0$ is

$$P_6 \cdot AP = P_M \cdot AG - 2 \cdot K_4 \cdot Z \tag{2.56}$$

Basic Servosystem With Electrical Dynamic Pressure Feedback

Equations (2.34) through (2.50) are used to describe the linear system model which incorporated electrical dynamic pressure feedback. Equation (2.33) must be modified to include a term proportional to the signal passed through the high pass filter. The modified error equation is

$$II = I + C_1 \cdot \theta + C_{32} \cdot P_D \tag{2.33b}$$

The signal passed through the high pass filter is

$$P_D = \frac{C_{48}}{\tau s + 1} \cdot sP_M \tag{2.57}$$

Simulation Models

The equations which describe the five servosystem models simplify when advantage is taken of the system symmetry and the equations are linearized about the origin of the valve characteristic curves. When the simplified equations which describe the basic servosystem with or without pressure feedback were combined, an eighth-order, closed-loop transfer function was formed. Equations which describe the dynamic pressure feedback system were combined to produce a ninth-order transfer function. These models were used to observe the system transient response. The linear simulation program presented in Appendix C incorporates these transfer functions.

## Model Used in Feedback Network Synthesis

One additional assumption was made to produce the procedure outlined in Chapter III to synthesize the dynamic pressure feedback network to enhance system damping. That assumption was that the dynamics of the servovalve are insignificant compared to other system dynamics. The servovalve could thus be treated as a static gain. The valve gain was obtained by combining equations (2.34) through (2.42) and equations (2.47) through (2.49) with the dynamic terms set equal to zero.

Important actuator and load dynamics include fluid compressibility and load inertia. These dynamics produce a third-order model to describe the basic closed-loop servosystem. The addition of pressure feedback does not change the order of the system model. The transfer function for either case is of the following form:

$$\frac{\theta}{I} = \frac{K}{a_3 s^3 + a_2 s^2 + a_1 s + a_0} \qquad (2.58)$$

Dynamic pressure feedback provided by means of a high-pass filter increases the order of the closed-loop system model to four. The transfer function is of the following form:

$$\frac{\theta}{I} = \frac{K \cdot (\tau s + 1)}{a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 a + a_0} \tag{2.59}$$

Definitions of the constants in equations (2.58) and (2.59) for the three system types are given in Appendix D. Block diagrams for the three system types are shown in Figures 13 through 15. For equations (2.58) and (2.59) the input torque, $T_L$, is set equal to zero.

Static Stiffness Determination

System static stiffness was one of the important performance criteria addressed by this study. Referring to Figures 13 through 15, the static stiffness is determined by calculating the steady-state actuator deflection, $\theta_{SS}$, for a given input torque, $T_L$, and for I=0.

The static stiffness for the systems studied can be calculated using the following equation:

$$S_S = \frac{T_L}{\theta} = \frac{\frac{\partial Q}{\partial Y} \cdot K_A \cdot K_V \cdot (1 - C_F) \cdot D_M}{\frac{C_S \cdot D_M}{\mu} + \frac{\partial Q}{\partial Y} \cdot K_A \cdot K_V \cdot K_P} \tag{2.60}$$

For systems which employ only the basic servosystem or the basic servosystem with dynamic pressure feedback, $K_P$ is equal to zero.

## Experimental Validation of Mathematical Models

The mathematical models presented in this chapter were derived for an actual servosystem. Both the non-linear and linear models were simulated on an IBM 3081D computer using the simulation package CSMP-360 (Continuous System Modelling Program). Program listings of the non-linear and linear model simulations can be found in Appendices

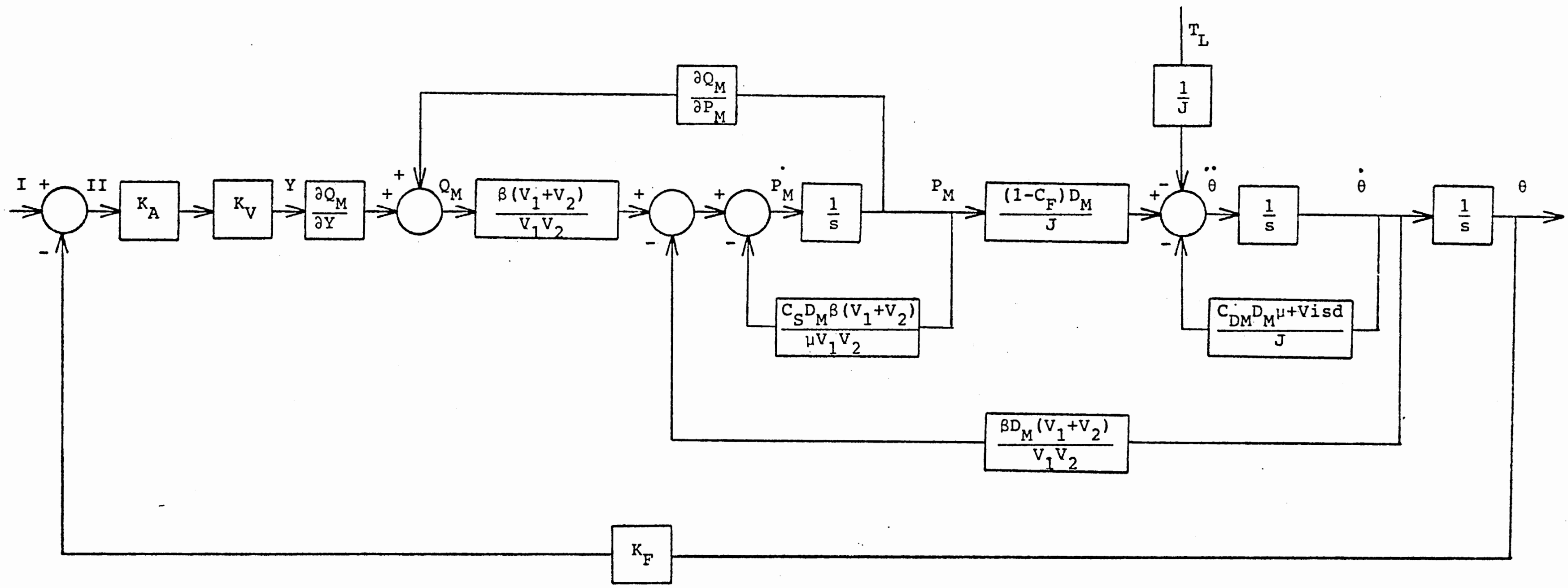


Figure 13. Block Diagram of the Basic Servosystem

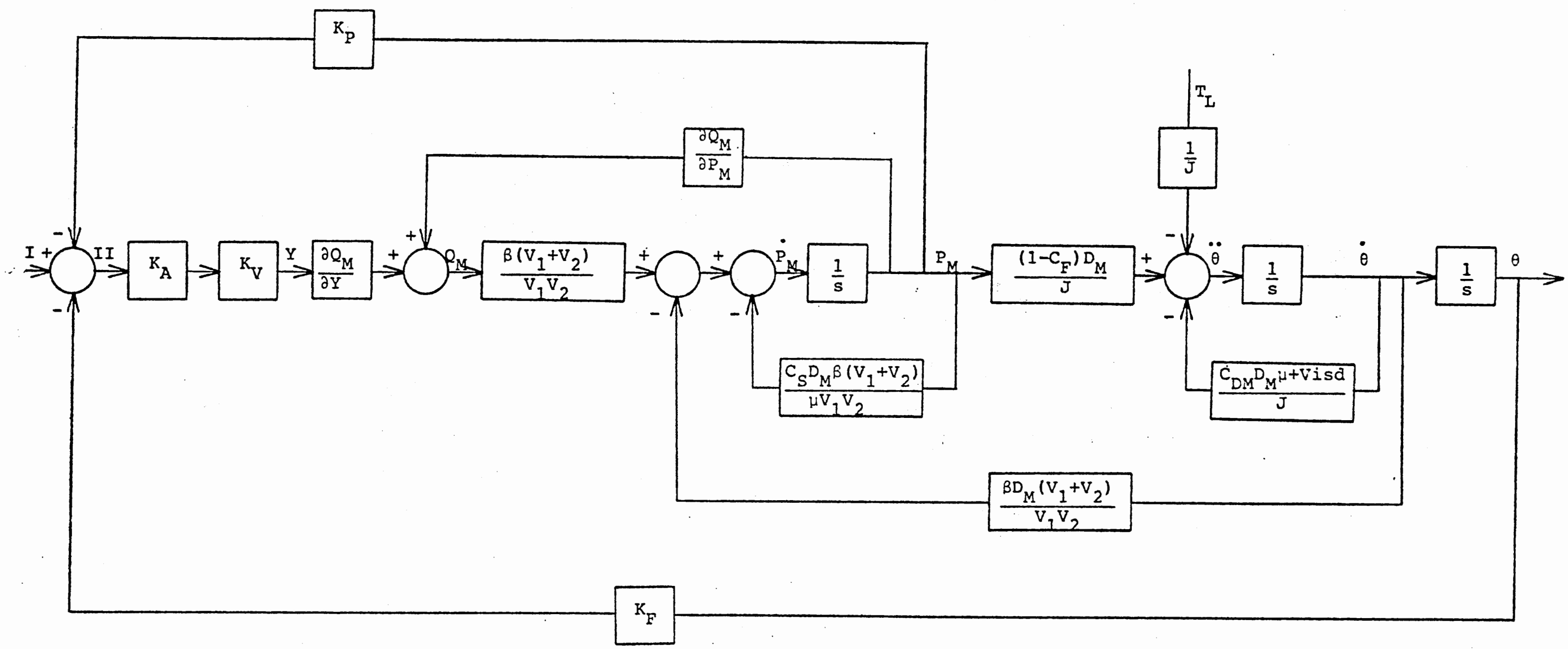


Figure 14. Block Diagram of Basic Servosystem With Pressure Feedback

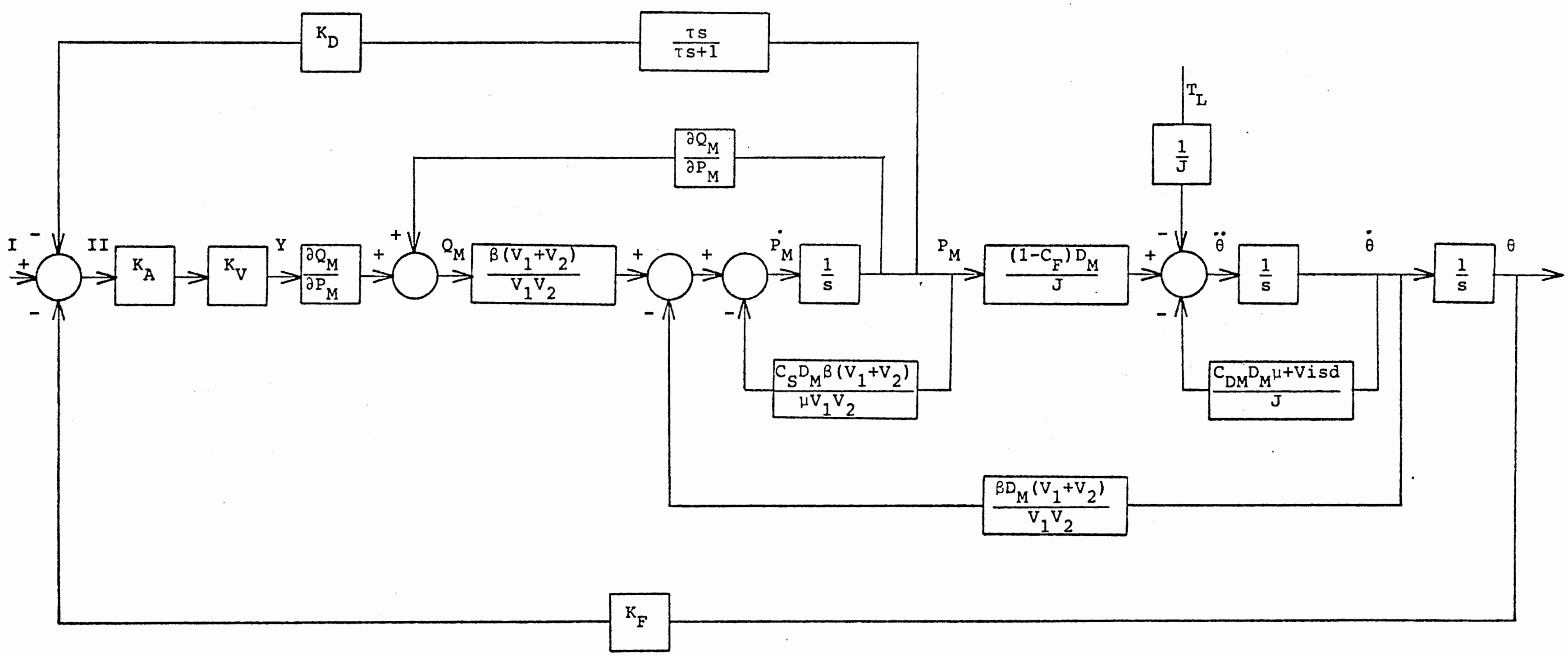


Figure 15. Block Diagram of Basic Servosystem With Dynamic Pressure Feedback

B and C, respectively.

Without an experimental validation, it remains questionable whether or not the mathematical models adequately describe the positional response of the system to a step input. Further, the initial premise that a linear system approximation can adequately describe the non-linear system remains in question. Both of these questions were answered by comparing measured systems results with the simulation predictions.

A schematic diagram of the test set-up for the experimental study is shown in Figure 16. An electronic function generator was used to provide a time-domain step input to the system. The positional response was measured with a strip-chart recorder. No external load torque was introduced into the system.

Data from the experiment was transferred to a computer data set so that measured results could be plotted directly against simulation results. Figure 17 shows a comparison of the measured results and the computer simulations for the basic servosystem with no additional means of damping enhancement. These results indicate that the linear model is adequate within the range of variables considered.

## Comparison of Damping Enhancement Via Pressure and Dynamic Pressure Feedback

The basic premise here and throughout this thesis is as follows: (1) an electrohydraulic position control system has been selected for a given application, (2) all static parameters have been selected except the loop gain, and (3) the open-loop system is so lightly damped that some means of damping enhancement is required. The problem is to

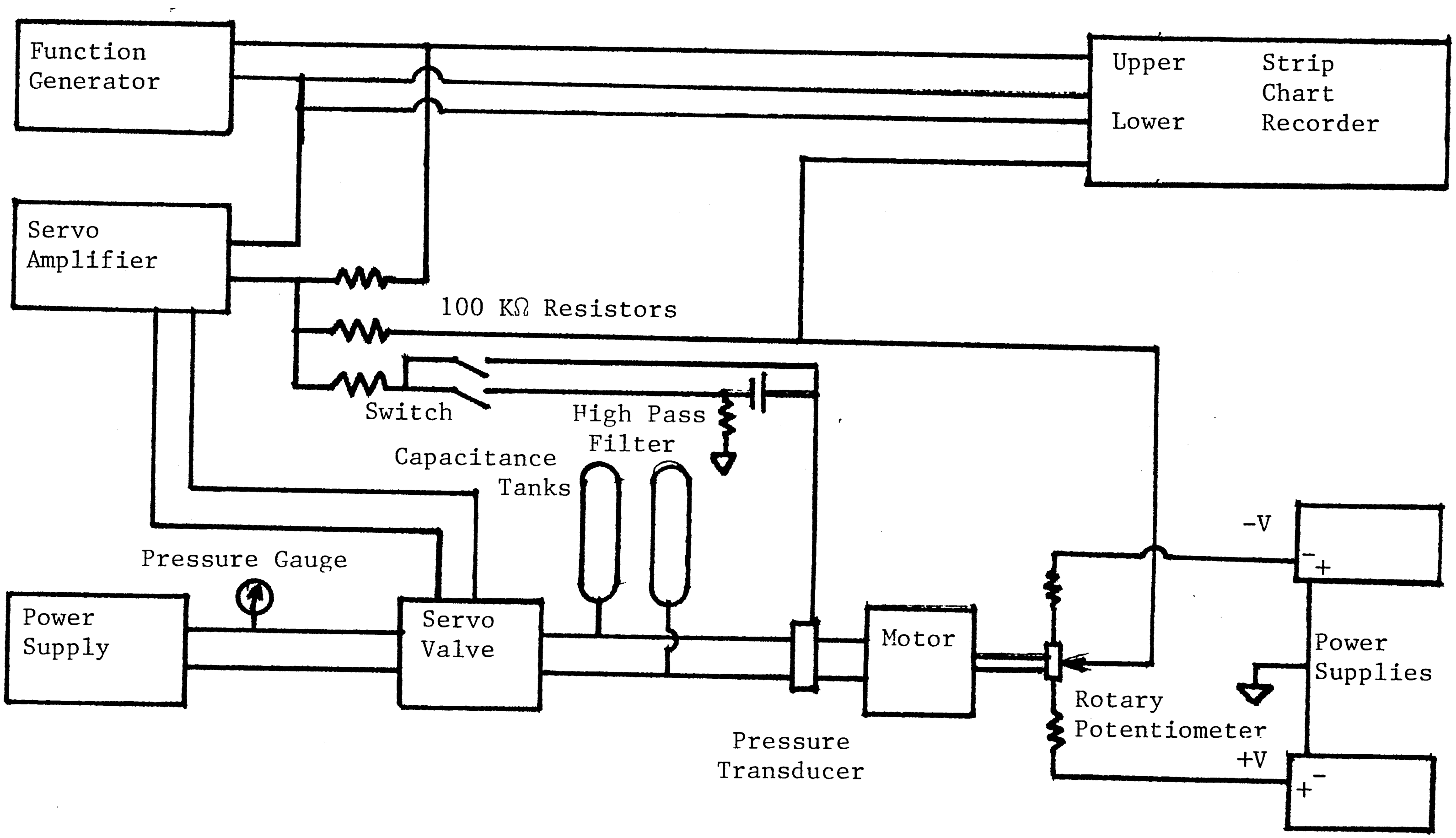


Figure 16. Test Set Up

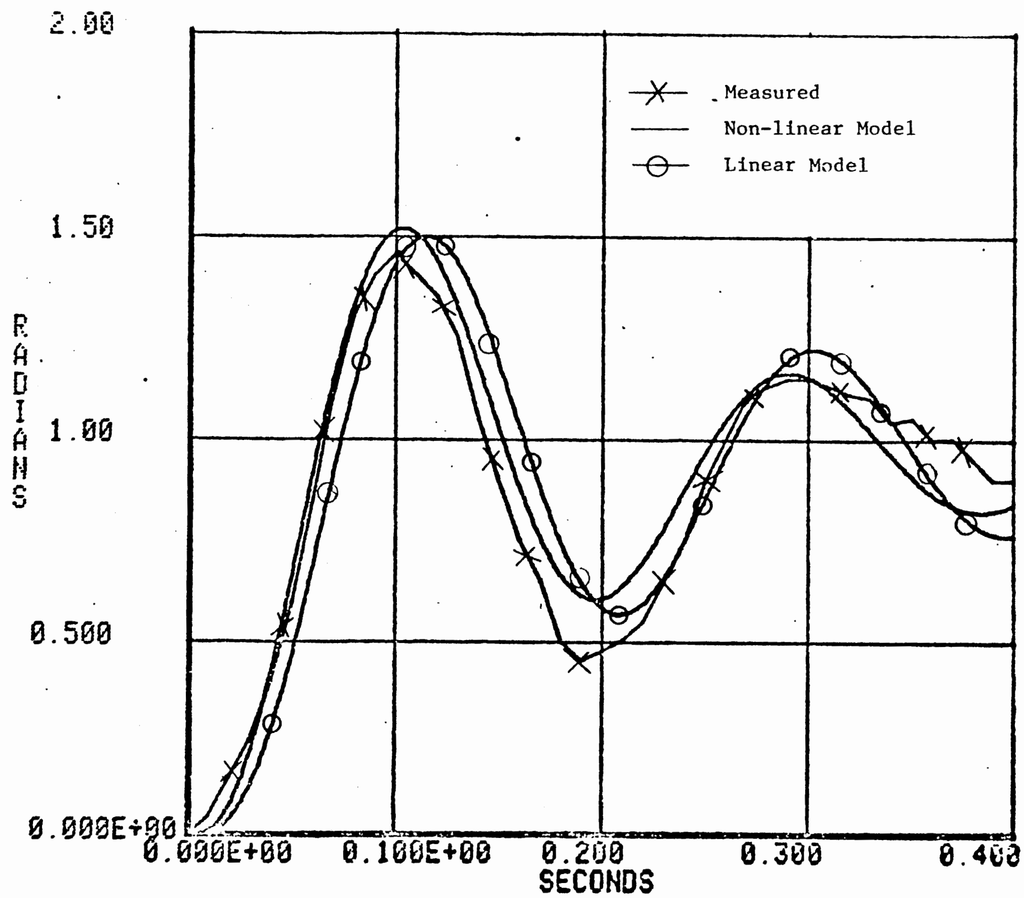


Figure 17. Comparison of Measured Response and Computer Simulations for the Basic Servosystem

finalize the design of the system so that it will meet defined static stiffness and dynamic response requirements.

To demonstrate the benefits of dynamic pressure feedback two comparisons are presented. Transient system responses for each comparison are shown in Figures 18 and 19. Tables II and III contain system performance measurements for the comparisons shown in Figures 18 and 19, respectively. System parameters utilized in making these comparisons can be found in Appendix F.

The first comparison (see Figure 18) shows the effect on system transient response as one parameter at a time is varied to add a dynamic pressure feedback network to a lightly damped servosystem. For illustration purposes the servoamplifier gain was fixed to provide a specified static stiffness for the basic servosystem as shown in Table II. A low degree of stability often results in the achievement of that static stiffness. Pressure feedback was added to the system without changing the servoamplifier gain; the degree of stability improved but the calculated static stiffness decreased considerably. A high pass filter was then placed in the system pressure feedback loop making the pressure feedback dynamic pressure feedback. Static stiffness returned to the level of the basic servosystem without damping enhancement with an improved degree of stability.

The second comparison was made among the same three system types discussed above. The comparison was based on each system producing a fifteen percent peak overshoot. Transient response for these systems can be seen in Figure 19 and Table III shows the system performance measurements.

In this case the dynamic pressure feedback system was able to

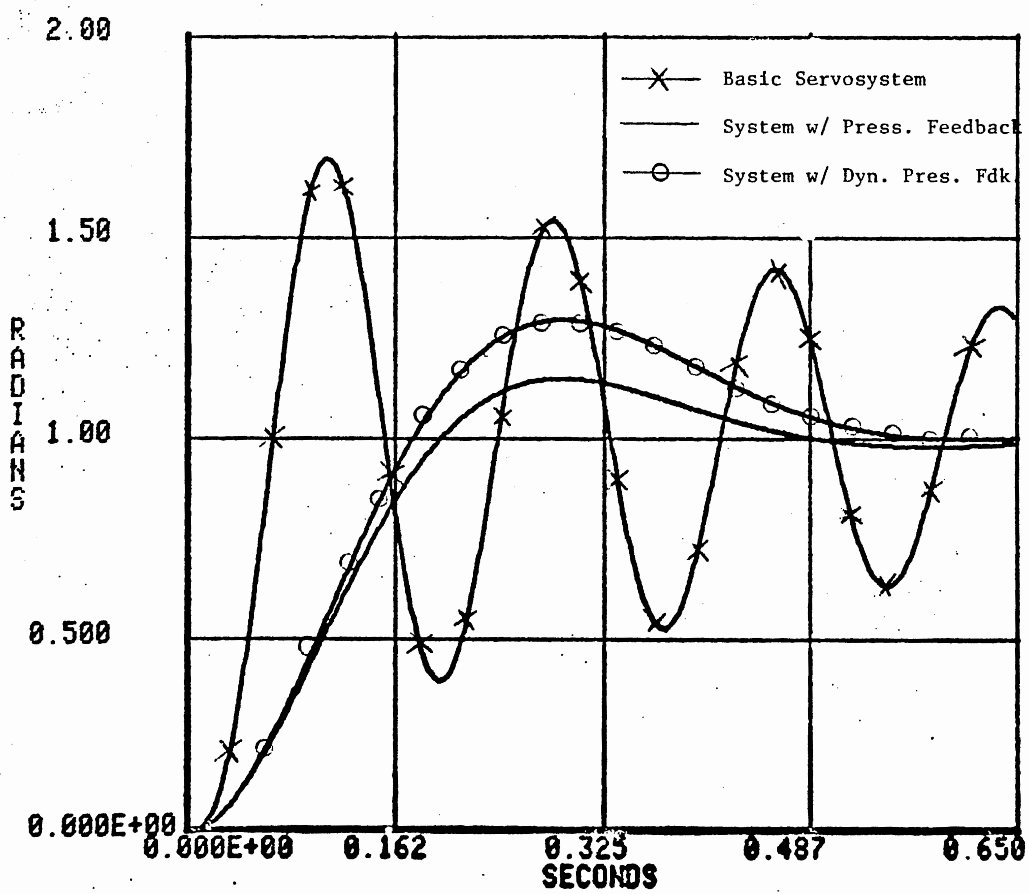


Figure 18. Comparison of Basic Servosystem, Pressure Feedback System, and Dynamic Pressure Feedback Systems Varying One Parameter at a Time

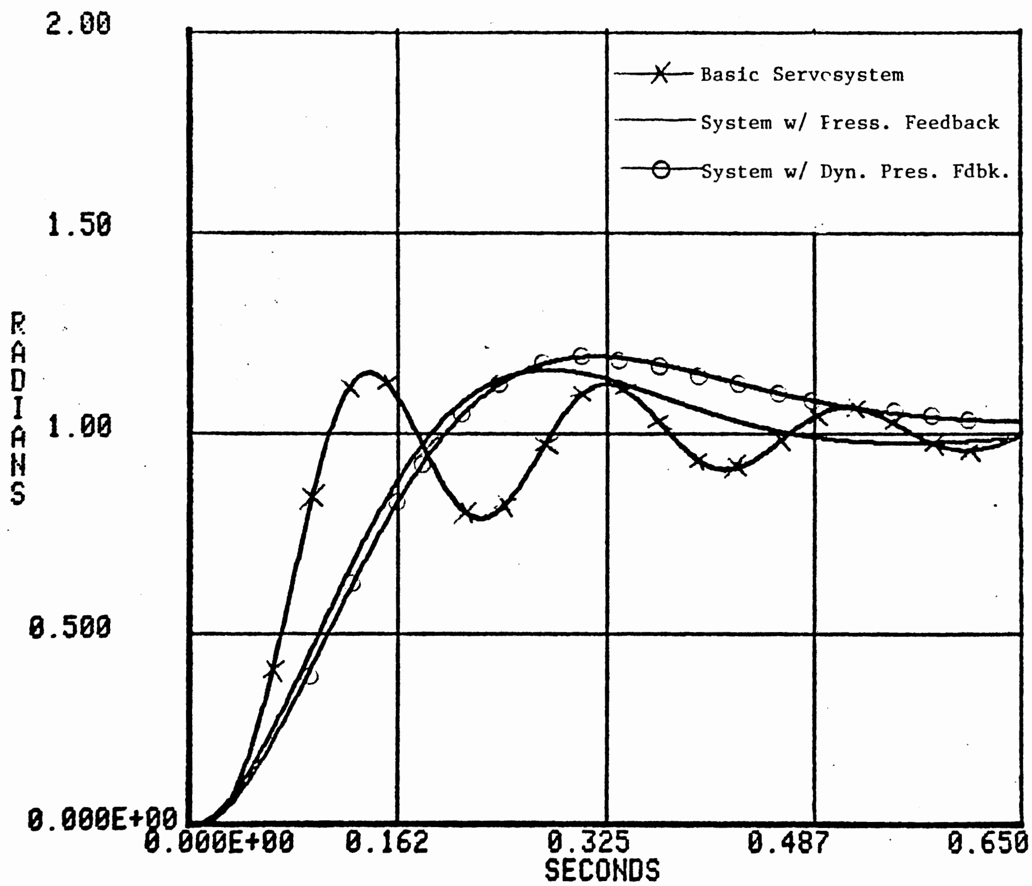


Figure 19. Comparison of Basic Servosystem, Pressure Feedback System, and Dynamic Pressure Feedback Systems With Constant Percent Overshoot

TABLE II

COMPARISON OF DYNAMIC PRESSURE FEEDBACK, PRESSURE FEEDBACK, AND BASIC SERVO-SYSTEMS AS ONE PARAMETER IS VARIED AT A TIME

| System Type | Servoamplifier Gain ma/volt | Pressure Feedback volts/psi | Time Constant seconds | Static Stiffness in $lb_f$/rad | Settling Time seconds | Rise Time seconds |
|---|---|---|---|---|---|---|
| Basic | 0.450 | ------ | ---- | 337 | >>0.65 | 0.037 |
| Pressure Feedback | 0.450 | $3.5 \times 10^{-2}$ | $\infty$ | 0.34 | 0.40 | 0.132 |
| Dynamic Pressure Feedback | 0.450 | $3.5 \times 10^{-2}$ | 0.5 | 337 | 0.49 | 0.118 |

TABLE III

COMPARISON OF DYNAMIC PRESSURE FEEDBACK, PRESSURE FEEDBACK, AND BASIC SERVO-SYSTEMS WITH 15% PEAK OVERSHOOT

| System Type | Servoamplifier Gain ma/volt | Pressure Feedback volts/psi | Time Constant seconds | Static Stiffness in $lb_f$/rad | Settling Time seconds | Rise Time seconds |
|---|---|---|---|---|---|---|
| Basic | 0.130 | ------ | ---- | 97 | 0.60 | 0.063 |
| Pressure Feedback | 0.450 | $3.5 \times 10^{-2}$ | $\infty$ | 0.34 | 0.40 | 0.132 |
| Dynamic Pressure Feedback | 0.208 | $3.5 \times 10^{-2}$ | 0.5 | 156 | 0.55 | 0.132 |

have a higher static stiffness and degree of stability than the basic servosystem with no damping enhancement. Pressure feedback produced an improved degree of stability compared to either dynamic pressure feedback or no feedback, but with inferior static stiffness.

In both comparisons it has been shown that a system which utilizes dynamic pressure feedback can provide a static stiffness at least as good as the basic servosystem with a higher degree of stability. A system with dynamic pressure feedback produces a higher static stiffness than a system with pressure feedback. Degree of stability decreases slightly with the addition of the high pass filter to the pressure feedback loop.

# CHAPTER III

# PROCEDURE FOR DYNAMIC PRESSURE FEEDBACK NETWORK PARAMETER DETERMINATION

## Definition of Problem Class

It is assumed that the basic electrohydraulic position control system has been designed to meet all static performance requirements except stiffness. Further, it is assumed that the open-loop servo-system is lightly damped and that an increase in the loop gain to satisfy the system static stiffness requirement results in an unsatisfactory closed-loop dynamic performance. Finally, it is assumed that damping enhancement to improve the closed-loop dynamic performance is to be achieved using dynamic pressure feedback.

## Assumptions to Outline Synthesis Procedure

To outline the synthesis procedure, the following assumptions were made:

1. The static stiffness requirement can be decoupled from the dynamic performance requirements.
2. The system maximum degree of stability occurs with an infinite feedback network time constant.
3. For a given loop gain, there is an optimum value of pressure feedback gain which provides the maximum degree of stability

(minimum settling time).

4. The pressure feedback gain which provides the maximum degree of stability for an infinite time constant also provides a near optimum degree of stability for a different time constant.

5. The loop gain required to provide the desired level of static stiffness is attainable.

6. The required pressure feedback gain is attainable.

7. The required dynamic pressure feedback network time constant is attainable.

Maximum Degree of Stability Criterion - A Proof

The root locus and root contour concepts underlie the assumptions concerning the maximum degree of stability. A root locus can be plotted for the model of the system as loop gain is increased from zero to infinity. The closed-loop poles for a given loop gain are observed as a particular set of points on the locus. All system parameters not contributing to the loop gain are fixed at some nominal value.

When a parameter other than loop gain is of interest, a separate root locus can be drawn for each constant value of that parameter. Another way to observe the effect of parameters other than loop gain on the closed-loop poles is through a root contour. The effect of a parameter, for example pressure feedback gain, on the closed-loop poles can be observed if another closed-loop transfer function is defined which has the same characteristic equation as the original system. Refer to equation (2.59). The parameter of interest, $K_D$ in this case, must replace loop gain in the transfer function open-loop gain position. The new system has the same stability properties as the original system since the same characteristic equation is used; thus, the eigenvalues are the same as well. Rules for constructing the root contour are the same as those employed in constructing the root locus (12).

The new closed-loop transfer function for the dynamic pressure feedback system with pressure feedback gain in the open-loop gain position is

$$1 + GH_{NEW} = 1 + \frac{K_D \cdot (b_3 s^3 + b_2 s^2)}{a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0} \tag{3.1}$$

The new open-loop transfer function is thus

$$GH_{NEW} = \frac{K_D \cdot (b_3 s^3 + b_2 s^2)}{a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0} \tag{3.2}$$

Definitions of the constants in Equations (3.1) and (3.2) are given in Appendix D.

Figures 20 through 22 each show a root contour drawn with pressure feedback gain as the adjustable parameter. Three pressure feedback network time constants are considered: infinity, 1.0 second, and 0.2 seconds. These particular time constants were chosen because they are in the vicinity of practical interest for the system under study. These figures were drawn from information obtained in the solution of the example problem in Chapter IV. Initial pole locations for the root contours (denoted by "X" in Figures 20 through 22) were established by setting the loop gain for the basic servosystem with

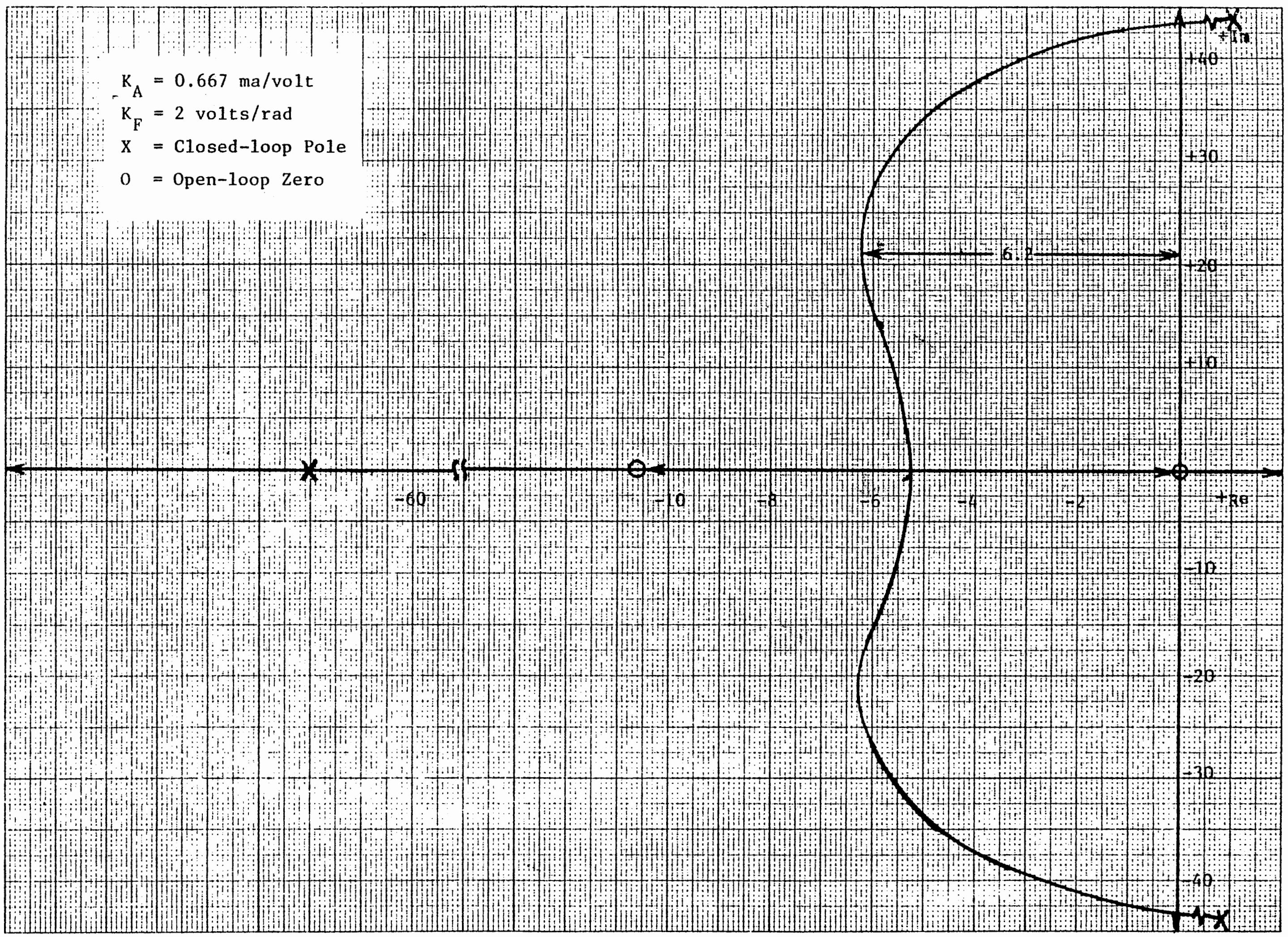


Figure 20. Root Contour of Dynamic Pressure Feedback Servosystem as Function of Pressure Feedback Gain, $\tau = \infty$

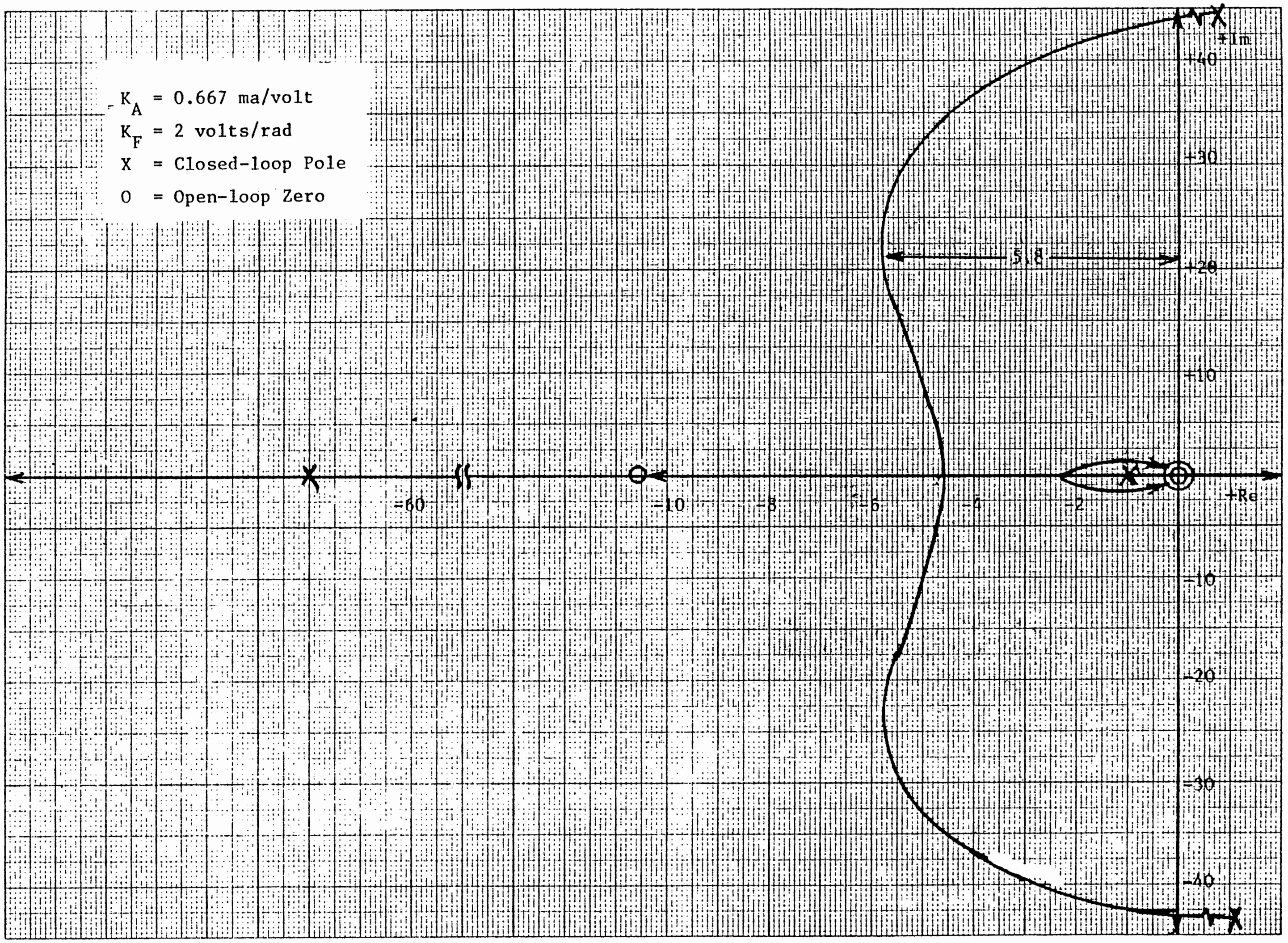


Figure 21. Root Contour of Dynamic Pressure Feedback Servosystem as Function of Pressure Feedback Gain, $\tau = 1.0$ seconds

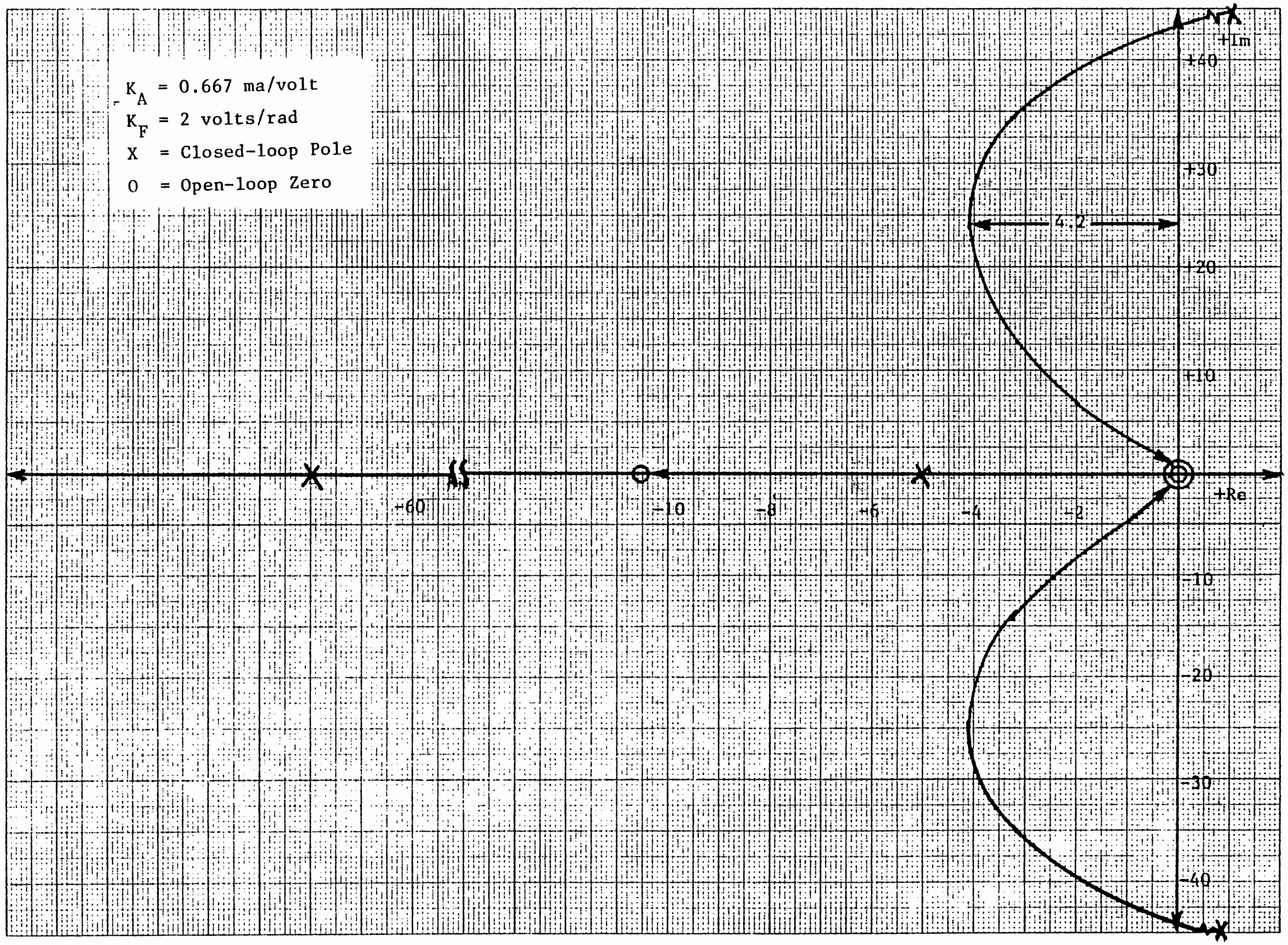


Figure 22. Root Contour of Dynamic Pressure Feedback Servosystem as Function of Pressure Feedback Gain, $\tau = 0.2$ seconds

with no damping enhancement to provide adequate static stiffness. The uncompensated system is clearly unstable.

The degree of stability, which is characterized by the transient settling time, is a direct function of the real component of each closed loop pole, particularly the dominant complex conjugate pole pair. There is a maximum degree of stability when the dominant complex pair of poles move to their furthermost point in the left half s-plane. A comparison of the three root contours shows that the complex conjugate pair move furthermost to the left with an infinite time constant. As the time constant decreases from infinity, the minimum settling time possible increases.

There is a unique value of pressure feedback gain which produces the minimum settling time in a system with an infinite feedback network time constant. The pressure feedback gain which provides the minimum settling time for a system with a time constant other than infinity is not the same value. However, calculated results in Table IV show that the effect on the degree of stability is small if the pressure feedback gain which provides minimum settling time for an infinite time constant is used when a different time constant is employed.

## Dynamic Pressure Feedback Network Synthesis Procedure

The synthesis procedure is presented in the flow chart of Figure 23. The designer must first determine if the system performance criteria can be satisfied with no damping enhancement or with pressure feedback alone. These steps have been included in the synthesis procedure.

## TABLE IV

### COMPARISON OF SETTLING TIMES FOR DIFFERENT TIME CONSTANTS

| Time Constant (seconds) | Optimum Pressure Feedback Gain (volts/psi) | Minimum Settling Time (seconds) | Settling Time $K_D$=0.01786 volt/psi (seconds) | Settling Time Error (percent) |
|---|---|---|---|---|
| ∞ | 0.01786 | 0.4879 | 0.4879 | 0.0 |
| 1.0 | ------- | ------ | ------ | --- |
| 0.6 | 0.01409 | 0.5811 | 0.5876 | 1.1 |
| 0.5 | 0.01404 | 0.5946 | 0.6014 | 1.1 |
| 0.4 | 0.01402 | 0.6148 | 0.6219 | 1.1 |
| 0.3 | 0.01398 | 0.6519 | 0.6597 | 1.2 |
| 0.2 | 0.01345 | 0.7517 | 0.7657 | 1.9 |

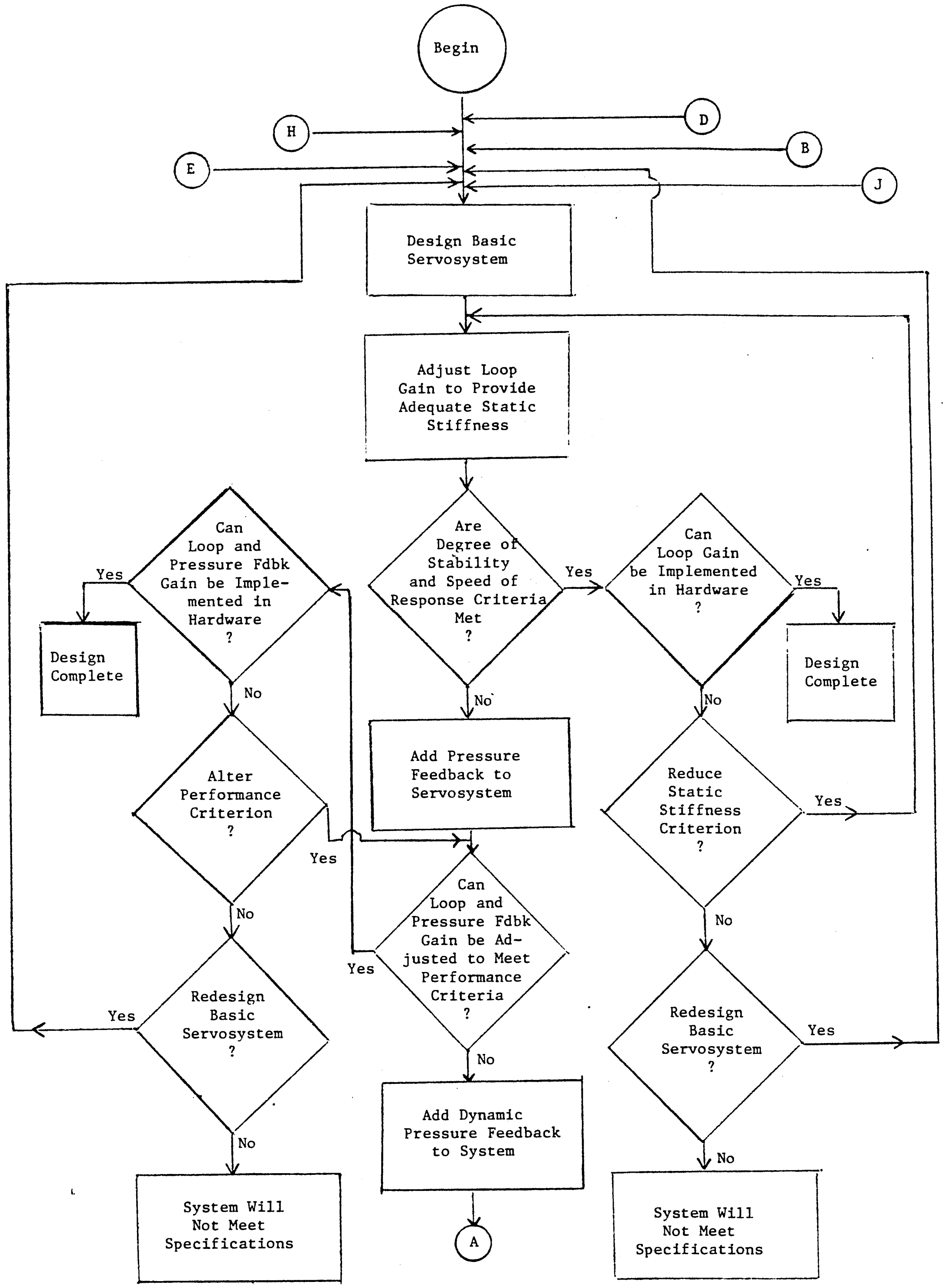


Figure 23. Flow Chart of the Synthesis Process

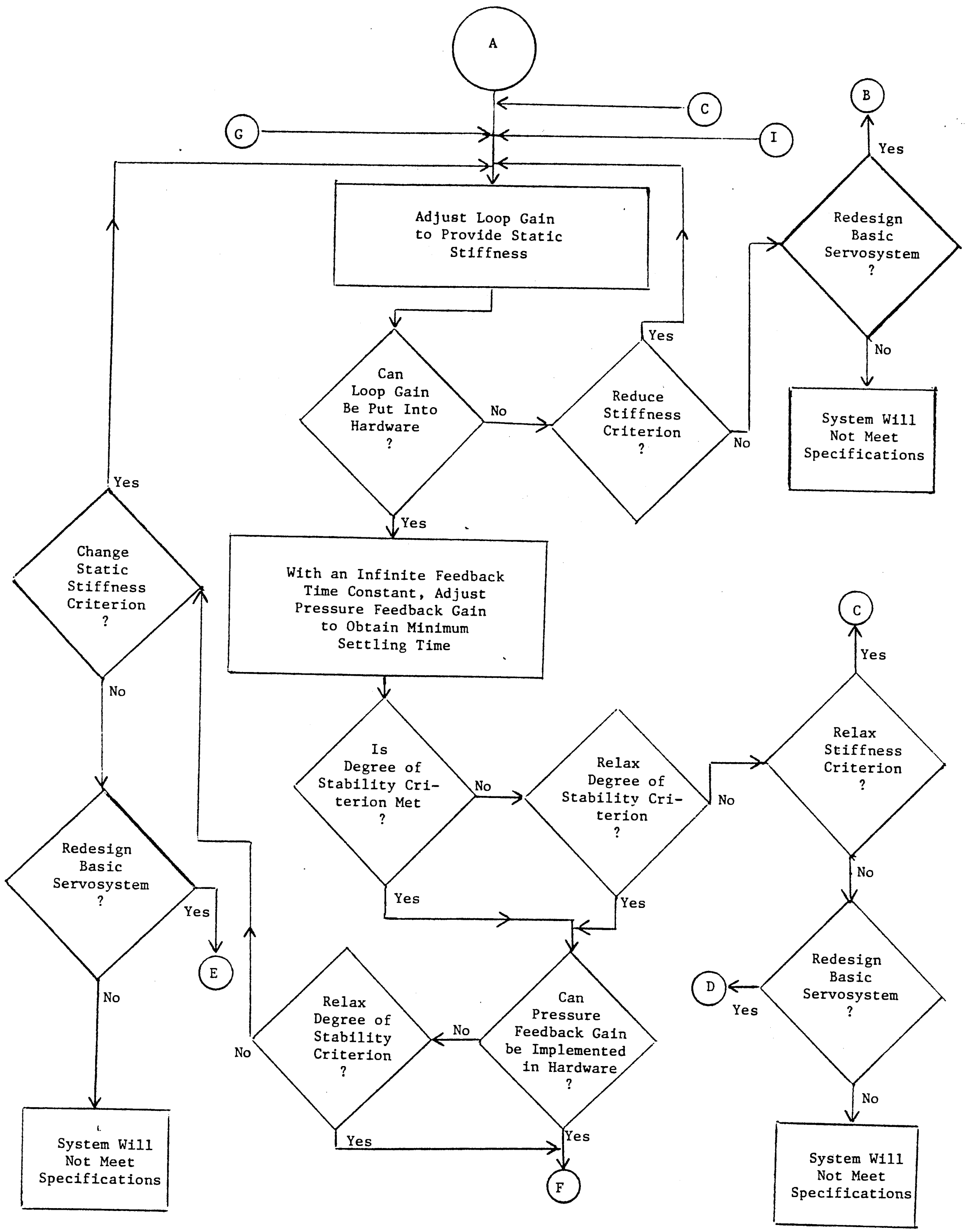


Figure 23. (Continued)

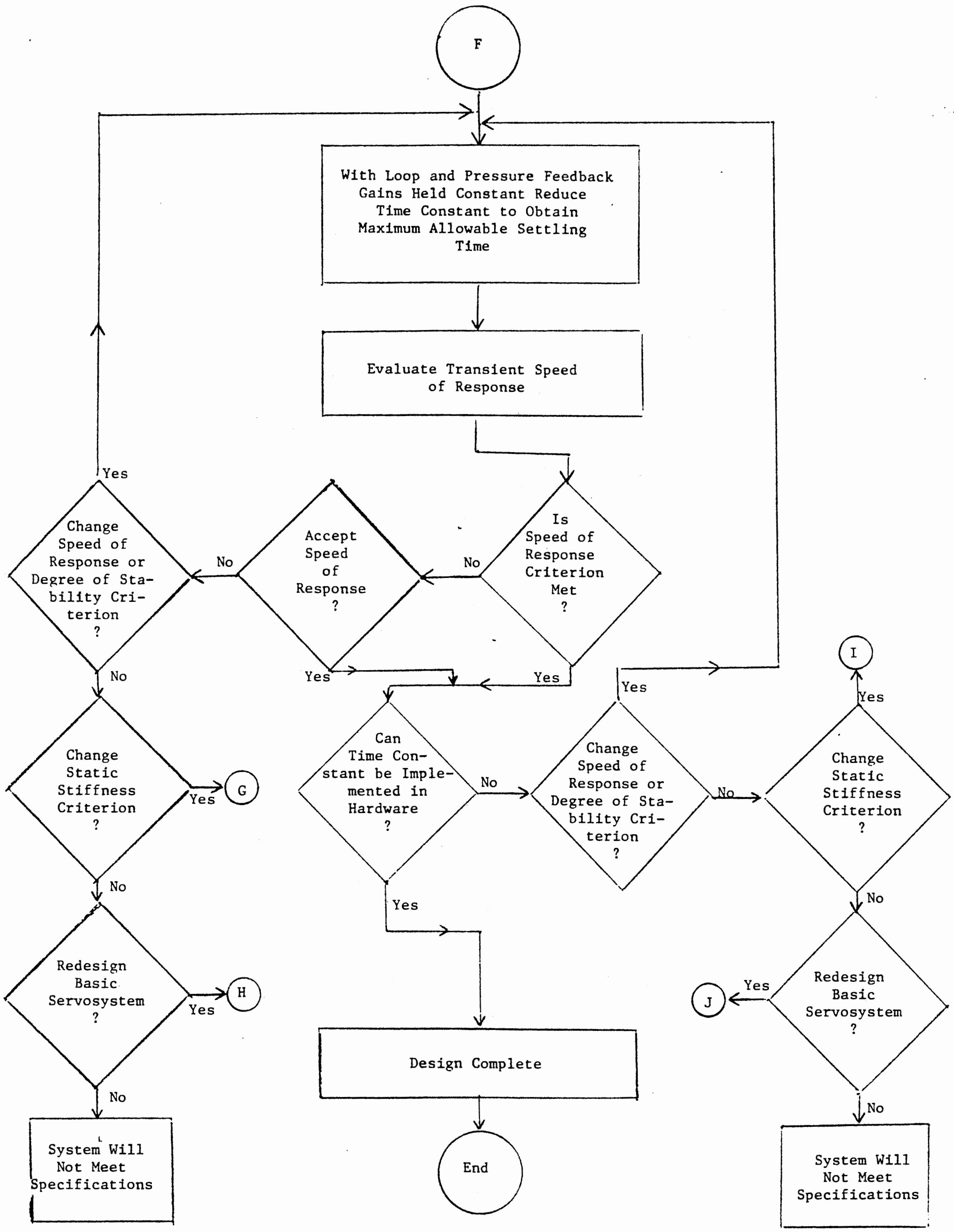


Figure 23. (Continued)

Once the need for dynamic pressure feedback has been established, the designer can use the computer program in Appendix E to make the necessary calculations for the dynamic pressure feedback network parameters. A discussion of how the dynamic pressure feedback synthesis was incorporated into the computer program can be found in the following section.

## Dynamic Pressure Feedback Synthesis Implementation

For the system with dynamic pressure feedback, the loop gain required to satisfy the system static stiffness performance criterion can be determined by direct computation; but such is not the case for the parameters affecting the system dynamic performance. Near optimum values of pressure feedback gain and network time constant can be calculated using an optimization program. The optimization routine STEPIT from the Oklahoma State University WATFIV FORTRAN computer library was used in this study.

A computer program was developed to supply STEPIT with the pertinent information on system description and specifications. Further, the computer program evaluates the minimization function required by STEPIT to determine how a parameter should be adjusted. System parameters are entered into the computer program via DATA statements. More information on the main program and the associated subroutines which comprise the computer program and how they work with STEPIT can be found in Appendix E.

A minimum of two executions of the computer program are required in the synthesis process. The first execution calculates the minimum servoamplifier gain required to produce the desired static stiffness. The static stiffness is a function of the system loop gain. For this system the servoamplifier gain, position feedback gain, servoactuator volumetric displacement, and valve flow gain comprise the system parameters in the loop gain. It is assumed that the servoactuator volumetric displacement, the valve flow gain, and the position feedback gain were all predetermined when the basic servosystem was designed.

Since the loop gain is a simple product of the parameters which comprise the loop, a change can be made in the servoamplifier and position gains as long as their product remains constant. The first execution of the computer program should be redone if the position feedback gain is altered from the value initially supplied.

The equation required to calculate the servoamplifier gain is

$$K_A = \frac{\frac{C_S \cdot D_M}{\mu} \cdot S_S}{\frac{\partial Q}{\partial Y} \cdot K_V \cdot K_F \cdot (1 - C_F) \cdot D_M} \qquad (3.3)$$

The parameters in equation (3.3) are defined in the Nomenclature. Equation (3.3) is derived from equation (2.60).

During the first execution of STEPIT, the third-order transfer function system model for a pressure feedback system, equation (2.58), i.e., the dynamic pressure feedback model with an infinite time constant, is utilized. The servoamplifier gain, $K_A$, and pressure feedback gain, $K_P$, are allowed to vary in order to determine the pressure feedback gain which produces the maximum degree of stability. The servoamplifier gain is not allowed to fall below the value calculated with equation (3.3) in order for the static stiffness requirement to be met. The initial value of the servoamplifier gain utilized is that value calculated with equation (3.3). The initial value of pressure feedback gain is user supplied.

During the first execution of the program, STEPIT determines the pressure feedback gain necessary to minimize the settling time. The function which STEPIT attempts to minimize is the settling time.

This first program execution requires some information to be user supplied. The information is entered into the computer via four statements.

(1) The required static stiffness is entered with the statement:

STIFF = AAA.A , (3.4)

where AAA.A is the desired static stiffness.

(2) The third-order pressure feedback model is selected for the first program execution by setting the following internal flag in the program:

SET = 3.0 (3.5)

(3) The initial value of the pressure feedback gain is supplied with the statement:

X(2) = BBB.B , (3.6)

where BBB.B is the initial pressure feedback gain supplied by the user.

(4) The minimization function for STEPIT to determine the pressure feedback gain necessary to provide the minimum settling time is the statement:

FOBJ = TSET (3.7)

The program is submitted and the information returned includes static stiffness obtained, servoamplifier gain, position feedback gain, rise time, and settling time. If the servoamplifier gain is not possible to implement in hardware, the other information returned would not be applicable.

The second execution of the program utilizes the dynamic pressure feedback system model. The servoamplifier gain, position feedback gain, and pressure feedback gain are fixed at the values returned by the first execution. Only the time constant is allowed to decrease from infinity such that the transient settling time can increase to its maximum allowable level. This is so the system will have the stiffness not achieveable with pressure feedback.

The function for STEPIT to minimize is changed for the second execution. The function must have its minimum value when the difference between the transient settling time and the maximum allowable settling time is zero.

The computer statements utilized in the second execution of the program follow. The internal program flag which sets up the fourth-order transfer function model, equation (2.59), for the dynamic pressure feedback system is

SET = 5.0 (3.8)

The servoamplifier and pressure feedback gains returned by the first execution are supplied to the second run with the following statements:

X(1) = CCC.C , (3.9)

X(2) = DDD.D , (3.10)

where CCC.C and DDD.D are servoamplifier and pressure feedback gains, respectively.

To prevent $K_A$ and $K_D$ from changing values, the following computer statements are inserted as an indicator to STEPIT that they are not to be varied:

MASK(1) = 1 (3.11)

MASK(2) = 1 (3.12)

The initial value of time constant is user entered with the statement

X(3) = EEE.E , (3.13)

where EEE.E is the time constant initial value.

The maximum allowable settling time for the transient response is entered with the following statement:

DSET = FFF.F , (3.14)

where FFF.F is the maximum allowable settling time.

The minimization function for the second program execution is a minimum when its value equals zero. The statement is

FOBJ = DABS(TSET - DSET) (3.15)

The program is again submitted and the pertinent information returned includes feedback network time constant, rise time, and settling time.

It appears that the time constant range suggested by Geyer (5) and Morse (9) may be related to the practical implementation of the time constant in hardware. Such a consideration was not incorporated within the computer algorithm used for the synthesis in this thesis.

CHAPTER IV

# APPLICATION OF THE SYNTHESIS PROCEDURE - AN EXAMPLE

## Problem Statement

The system considered is a lightly damped, electrohydraulic, position control servosystem. The requirements and specifications for the example are as follows:

| | |
|---|---|
| Minimum static stiffness: | 1,000 in $lb_f$/rad |
| Maximum settling time: | 0.625 seconds |
| Maximum rise time: | 0.055 seconds |
| Position feedback gain: | 2 volts/rad |
| Step input to the system: | 1.5 volts |
| Actuator-load open-loop natural frequency: | 23.0 rad/seconds |
| Actuator-load open-loop damping ratio: | 0.3 |

The remaining parameters except servoamplifier gain are the same as used for the system simulation (see Chapter II) and are given in Appendix F. The servoamplifier gain has not been determined.

The static stiffness is a function of the system loop gain. For this system the servoamplifier gain, position feedback gain, servo-actuator volumetric displacement, and valve flow gain comprise the system elements in the loop gain. System specifications fixed the position feedback gain. For the basic servosystem with no damping enhancement, the static stiffness must be obtained through the adjustment of the servoamplifier gain.

The basic servosystem servoamplifier gain required to maintain static stiffness is calculated with equation (3.3). For the basic servosystem, an amplifier gain of 0.667 ma/volt is required to maintain 1,000 in $lb_f$/rad static stiffness. This gain produces a system which is unstable. The transient response for this system is shown in Figure 24.

Pressure feedback can be added to the system to improve the dynamic performance. To calculate the static stiffness for the system which employs pressure feedback, the equation used is

$$S_S = \frac{\frac{\partial Q}{\partial Y} \cdot K_A \cdot K_V \cdot K_F \cdot (1 - C_F) \cdot D_M}{\frac{C_S \cdot D_M}{\mu} + \frac{\partial Q}{\partial Y} \cdot K_A \cdot K_V \cdot K_P} \tag{4.1}$$

Figure 25 shows a plot of static stiffness versus pressure feedback gain for different values of servoamplifier gain. When the amplifier gain is increased from 0.667 ma/volt, the pressure feedback gain required to maintain the required static stiffness can be determined with equation (4.1).

If 1,000 in $lb_f$/rad stiffness is maintained as shown in Figure 25 and the servoamplifier gain is increased, the servoamplifier gain increases at a rate greater than the pressure feedback gain. The result is that the system is initially unstable and the degree of stability monotonically decreases. The poles of the transfer function move toward positive infinity. Pressure feedback will not suffice to satisfy both the static stiffness and degree of stability criteria.

Some means of damping enhancement is required to meet the static

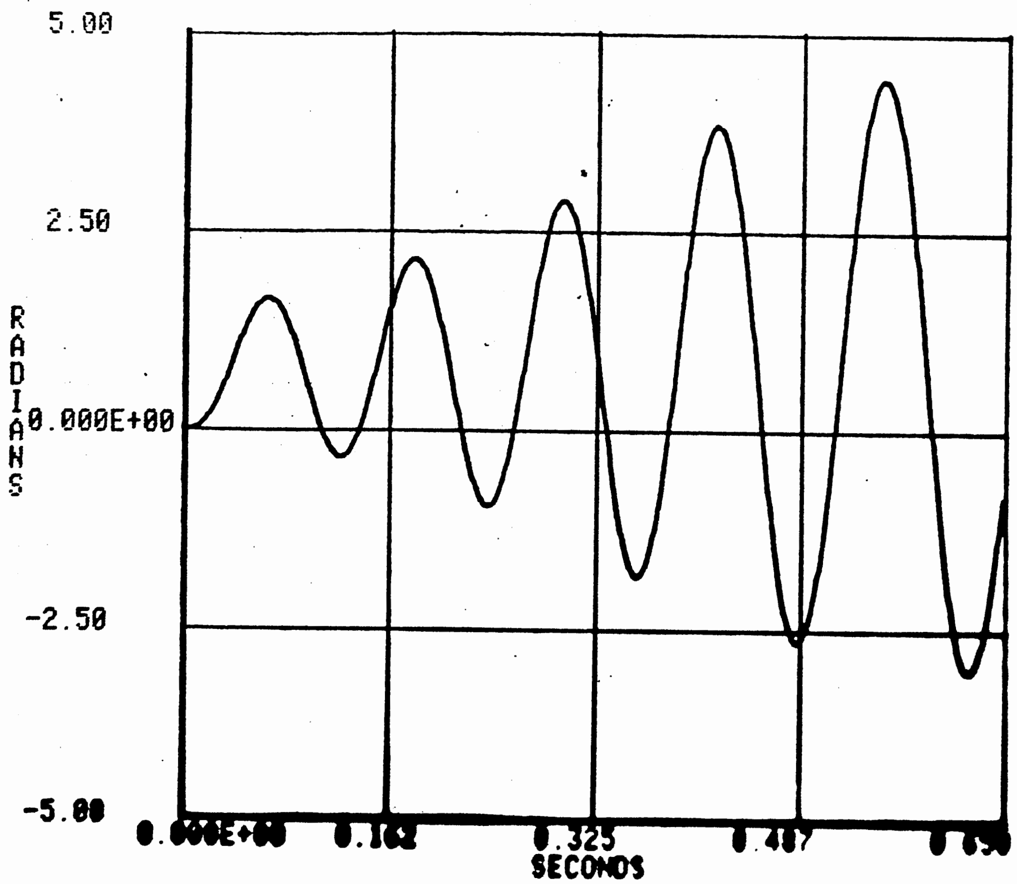


Figure 24. Transient Response of System With No Damping Enhancement, $K_A$ = 0.667 ma/volt, $K_F$ = 2 volts/rad

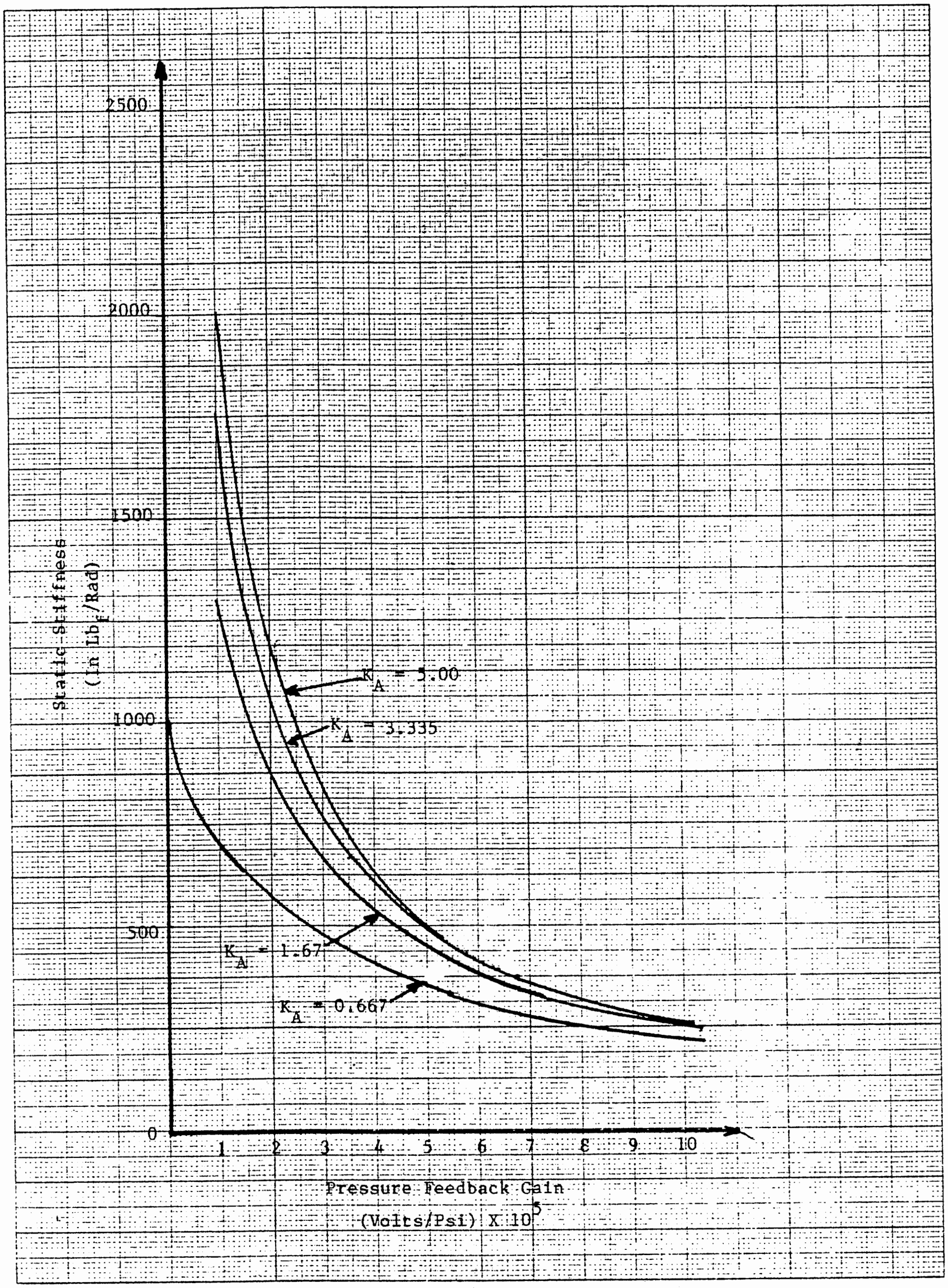


Figure 25. Plot of Static Stiffness Versus Pressure Feedback Gain for Pressure Feedback System in Example Problem

and dynamic specifications. Pressure feedback is not capable of producing the desired results. Dynamic pressure feedback was selected to enhance system damping and to avoid the undesirable effects introduced into the system via many other damping enhancement methods.

## Problem Solution

The synthesis procedure presented in Chapter III was utilized in the design of a dynamic pressure feedback network for the example system. The static stiffness was entered into the program with the statement

STIFF = 1000.0 (4.2)

The initial value of pressure feedback gain which is an estimate was entered as

X(2) = 2.5D-02 (4.3)

The output from the first execution of STEPIT can be seen in Figure 26. The key information returned is as follows:

Servoamplifier gain required for static stiffness:

0.667 ma/volt

Pressure feedback gain required for minimum settling time (infinite time constant):

0.0179 volt/psi

Transient response rise time:

0.052 seconds

Transient response settling time:

0.49 seconds

The servoamplifier gain and pressure feedback gain returned by the first execution appear to be achievable in actual hardware. The

```
THE MINIMUM REQUIRED STATIC STIFFNESS (IN*LBF/RAD) IS        0.10000D 04
THE REQUIRED AMPLIFIER GAIN (MA/VOLT) IS                     0.66724D 00
THE STATIC STIFFNESS ACTUALLY OBTAINED (IN*LBF/RAD) IS       0.10000E 04
FINAL VALUE FOR AMPLIFIER GAIN (MA/VOLT) IS                  0.66724D 00
FINAL VALUE FOR FEEDBACK GAIN (VOLT/PSI) IS                  0.17862D-01
THE CLOSED LOOP POLES ARE
      X(1)=    -0.62462D 01+(    0.23600D 02) I
      X(2)=    -0.62462D 01+(   -0.23600D 02) I
      X(3)=    -0.29565D 03+(    0.28106D-17) I
STEADY STATE DISPLACEMENT (RAD) IS                           0.75000D 00
COEFFICIENT OF FIRST REAL POLE TERM IS                      -0.53016D-02
EXPONENT OF THE FIRST REAL POLE IS                           0.29565D 03
COEFFICIENT OF SINUSOIDAL TERM IS                           -0.78995D 00
EXPONENT OF SINUSOIDAL TERM IS                               0.62462E 01
RESPONSE DAMPED NATURAL FREQUENCY (RAD/SEC) IS               0.23600D 02
PHASE SHIFT OF SYSTEM RESPONSE (RAD) IS                      0.12307D 01
RESPONSE PEAK DISPLACEMENT (RAD) IS                          0.10643D 01
TIME (SECS) PEAK DISPLACEMENT OCCURS IS                      0.14753D 00
RESPONSE RISE TIME (SECS) IS                                 0.52187D-01
RESPONSE SETTLING TIME (SECS) IS                             0.48791D 00
RESPONSE ENVELOPE VALUE (RAD) AT WHICH SETTLING TIME OCCURS  0.71250D 00
THE OPEN LOOP POLES ARE
      X(1)=     0.00000D 00+(    0.00000D 00) I
      X(2)=    -0.29182D 02+(    0.28227D 02) I
      X(3)=    -0.29182D 02+(   -0.28227D 02) I
```

Figure 26. Output Returned by the First Execution of STEPIT in Example Problem

settling time obtained for the transient response more than meets the specifications. The rise time is less than the specified maximum.

For the second execution of the synthesis program, the maximum allowable settling time was entered with the statement

DSET = 0.625 (4.4)

The servoamplifier gain and the pressure feedback gain were entered into the program prior to the second execution with the statements of

X(1) = 0.667 , and (4.5)

X(2) = 0.0179 (4.6)

The initial value of the feedback network time constant supplied to the program was

X(3) = 0.5 (4.7)

Output from this execution is shown in Figure 27. The pertinent information from the output is as follows:

Feedback network time constant: 0.39 seconds

Transient response rise time: 0.050 seconds

The transient response rise time does not exceed the system specifications. The time constant is such that the corner frequency of the high pass filter is in the range of one-third to one-tenth the open-loop actuator-load natural frequency as suggested by Geyer (5) and Morse (9). It appears that the time constant could be implemented in hardware. The design of the dynamic pressure feedback network has been completed and the performance criteria satisfied. Figure 28 shows the transient response for the resulting system with dynamic pressure feedback.

```
THE MINIMUM REQUIRED STATIC STIFFNESS (IN*LBF/RAD) IS          0.10000D 04
THE REQUIRED AMPLIFIER GAIN (MA/VOLT) IS                        0.66724D 00
THE STATIC STIFFNESS ACTUALLY OBTAINED (IN*LBF/RAD) IS          0.10000E 04
FINAL VALUE FOR AMPLIFIER GAIN (MA/VOLT) IS                     0.66724D 00
FINAL VALUE FOR FEEDBACK GAIN (VOLT/PSI) IS                     0.17860D-01
FINAL VALUE OF FEEDBACK TIME CONSTANT (1/SEC) IS                0.38846D 00
THE CLOSED LOOP POLES ARE
      X(1)=    -0.26552D 01+(    0.00000D 00) I
      X(2)=    -0.51172D 01+(    0.23397D 02) I
      X(3)=    -0.51172D 01+(   -0.23397D 02) I
      X(4)=    -0.29781D 03+(   -0.37321D-16) I
STEADY STATE DISPLACEMENT (RAD) IS                              0.75000D 00
COEFFICIENT OF FIRST REAL POLE TERM IS                          0.24665D-01
EXPONENT OF THE FIRST REAL POLE IS                              0.26552D 01
COEFFICIENT OF SECOND REAL POLE TERM IS                        -0.51484D-02
EXPONENT OF SECOND REAL POLE IS                                 0.29781D 03
COEFFICIENT OF SINUSOIDAL TERM IS                              -0.80345D 00
EXPONENT OF SINUSOIDAL TERM IS                                  0.51172E 01
RESPONSE DAMPED NATURAL FREQUENCY (RAD/SEC) IS                  0.23397D 02
PHASE SHIFT OF SYSTEM RESPONSE (RAD) IS                         0.12791D 01
RESPONSE PEAK DISPLACEMENT (RAD) IS                             0.11459D 01
TIME (SECS) PEAK DISPLACEMENT OCCURS IS                         0.14674D 00
RESPONSE RISE TIME (SECS) IS                                    0.49969D-01
RESPONSE SETTLING TIME (SECS) IS                                0.62500D 00
RESPONSE ENVELOPE VALUE (RAD) AT WHICH SETTLING TIME OCCURS     0.78750D 00
THE OPEN LOOP POLES ARE
      X(1)=     0.00000D 00+(    0.00000D 00) I
      X(2)=    -0.25743D 01+(    0.00000D 00) I
      X(3)=    -0.29182D 02+(    0.28227D 02) I
      X(4)=    -0.29182D 02+(   -0.28227D 02) I
```

Figure 27. Output Returned by the Second Execution of STEPIT in Example Problem

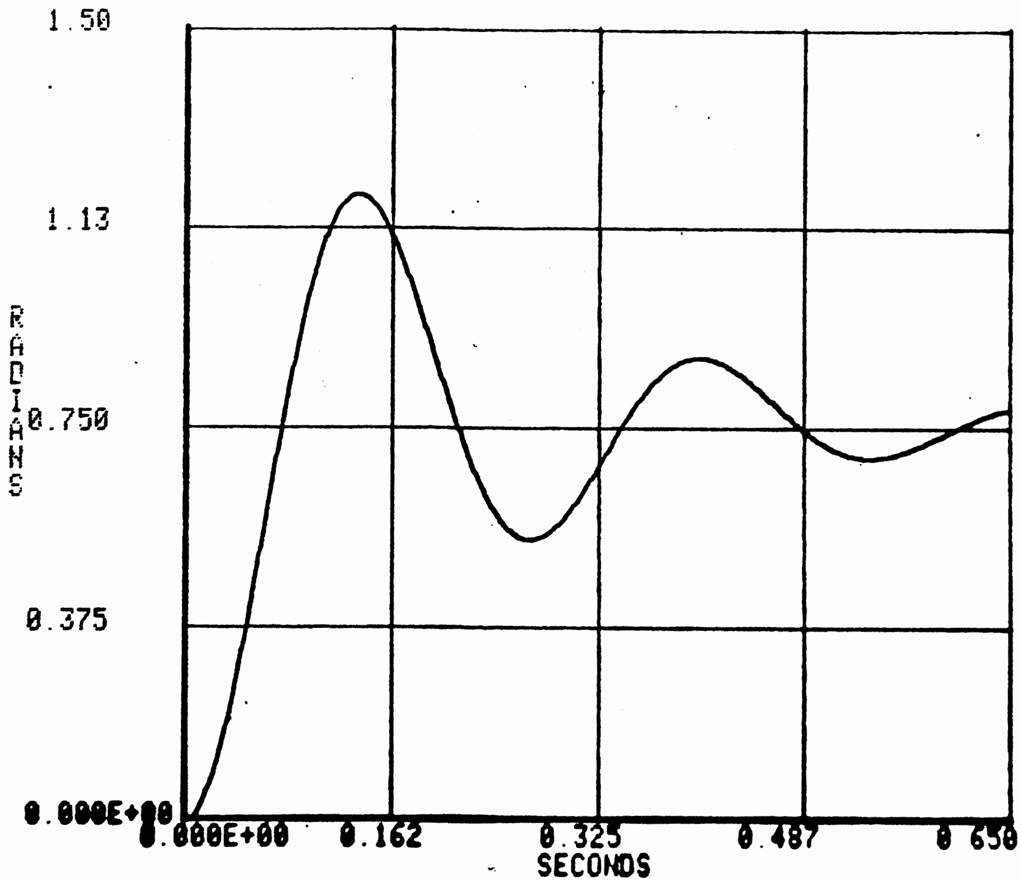


Figure 28. Transient Response for the System With Dynamic Pressure Feedback in the Example Problem

# CHAPTER V

# CONCLUSIONS AND RECOMMENDATIONS

## Conclusions

A method was outlined to independently select the parameters of a dynamic pressure feedback network to compensate a lightly damped, electrohydraulic position control servosystem.

The synthesis procedure developed places no restriction on the time constant of the feedback network. It appears that the feedback network corner frequency (reciprocal of time constant) range of one-third to one-tenth the actuator-load natural frequency suggested by Geyer (5) and Morse (9) may be related to limitations of hardware implementation. The work of this study showed no special significance to this corner frequency range on system performance.

With all system parameters held constant except time constant the degree of stability (minimum settling time) increased with time constant for the class of systems considered in this study. When all system parameters were held constant except pressure feedback gain it was found that there is a unique pressure feedback gain which provides the minimum settling time as demonstrated by the root contours of Chapter III. The pressure feedback gain which produces the maximum degree of stability in systems with feedback network time constants other than infinity is close enough to that for an infinite time constant to have minimal effect on the settling time for those systems.

## Recommendations

The study undertaken assumed that the significant servosystem dynamics were load inertia and fluid compressibility (in the actuator chambers). The synthesis technique should be extended to account for valve and/or transmission line dynamics.

The time constant of a mechanical dynamic pressure feedback network is a variable and is sensitive to the amplitude of the load pressure. In order to minimize the pressure feedback network sensitivity to pressure feedback amplitudes and possibly satisfy the performance criteria with less compromise the study of dynamic pressure feedback should be extended to include non-linear and optimal control theory.

BIBLIOGRAPHY


1. Blackburn, J.F., Reethof, G., and Shearer, J.L., Fluid Power Control, M.I.T. Press, Cambridge, Mass., 1960.

2. Chandler, John P., "Instructions on Using STEPIT to Minimize or Maximize a Function", Unpublished, Stillwater, Oklahoma, Oklahoma State University, 1973.

3. Dorf, Richard C., Modern Control Systems, Addison-Wesley, Menlo Park, California, 1974.

4. "Dynamic Pressure Feedback, Controlled Damping by Use of the Dowty Moog Valve", Aircraft Engineering, June, 1960, pp. 171-176.

5. Geyer, L.H., "Controlled Damping Through Dynamic Pressure Feedback", Moog Technical Bulletin 101, Moog, Inc., Controls Division, East Aurora, New York, 1972.

6. McCloy, D., Martin, H.R., The Control of Fluid Power, John Wiley, New York, 1973.

7. Merritt, H.D., Hydraulic Control Systems, John Wiley, New York, 1967.

8. Meyfarth, P.F., Dynamic Response Plots and Design Charts for Third-Order Linear Systems, M.I.T. Dynamic Analysis and Control Laboratory, Research Memorandum. 7401-3, September, 1958.

9. Morse, A.C., Electrohydraulic Servomechanisms, McGraw-Hill, New York, 1963.

10. Reid, Karl N., Classnotes for MAE 5453, Fluid Power Control I, Unpublished, Stillwater, Oklahoma, Oklahoma State University, Fall 1981.

11. Richardson, H.H., "The Analytical Design of Valve-Controlled Hydraulic Power and Control Systems. A Case Study", Unpublished Paper, M.I.T., Cambridge, Mass., 1962.

12. Rowland, James R., Linear Control Systems, Unpublished, Stillwater, Oklahoma, Oklahoma State University, 1983.

13. Shampine, L.F., Allen, R.C., Numerical Computing: An Introduction, Saunders Publishing Co., Philadelphia, 1973.

14. Speckhart, F.H., Green, W.L., A Guide to Using CSMP - The Continuous System Modelling Program, Prentice-Hall, Englewood Cliffs, New Jersey, 1976.

15. Thayer, W.J., "Transfer Functions for Moog Servovalves", Moog Technical Bulletin 103, Moog, Inc., Controls Division, East Aurora, New York, 1965.

# APPENDIXES

APPENDIX A

# EQUATION LINEARIZATION

## Linearization Technique

Many differential equations which describe a system are non-linear. Linear system theory is not applicable to those equations without first transforming them into a linearized form. Once line-arized the principle of superposition exists for the system equations.

The non-linear equations must be linearized about an operating point. The same non-linear equations linearized about two different operating points produces two different sets of equations.

Linearization was done about the origin of the pressure-flow curves for the position control system. A hydraulic servosystem op-erates about the origin of the valve pressure-flow-curves which is the point the system operates statically. The linear theory concept of the transfer function assumes that all initial conditions are zero referring to the origin of the pressure-flow curves.

Usage of derivatives and partial derivatives of the equations de-scribing the system is part of the linearization procedure. Equation derivatives give the rate of change of a variable described by the differential equation. Small changes in the variable can be approx-imated by the product of the derivative and the incremental change in independent variable with which the derivative was taken.

Two examples are presented to exemplify the linearization procedure. The first example involves a function of a single variable and uses the derivative to obtain the linearization constant. The second example uses partial derivatives for a function of two or more variables.

Example 1. Define a function of a single variable.

$$Y = f(X) \tag{A.1}$$

Take the derivative of the function.

$$\frac{dY}{dX} = f'(X) \tag{A.2}$$

Multiply both sides of the equation by dX.

$$dY = f'(X) \cdot dX \tag{A.3}$$

Approximate the infinitesimal changes of the differential for small changes with Δ.

$$\Delta Y = f'(X) \cdot \Delta X \tag{A.4}$$

ΔX and ΔY can be defined in terms of differences.

$$\Delta X = X - X_i \tag{A.5}$$

$$\Delta Y = Y - Y_i \tag{A.6}$$

f'(X) is evaluated at the operating point and equations (A.5) and (A.6) can be substituted into equation (A.4).

$$Y - Y_i = f'(X_i) \cdot (X - X_i) \tag{A.7}$$

Regroup the terms.

$$Y = f'(X_i) + Y_i - f'(X_i) \cdot X_i \tag{A.8}$$

Equation (A.1) must hold true at the operating point

$$Y_i = f(X_i) \tag{A.9}$$

Substitute equation (A.9) into equation (A.8).

$$Y = f'(X_i)\cdot X + f(X_i) - f'(X_i)\cdot X_i \tag{A.10}$$

The last two terms of equation (A.10) are the offset of the linearized equation from the origin. These terms will cancel if the value of the dependent variable is zero at the operating point.

Example 2. Define a function of two or more variables.

$$Y = f(X,Z) \tag{A.11}$$

Take the differential of the equation

$$dY = \frac{\partial f(X,Z)}{\partial X}\cdot dX + \frac{\partial f(X,Z)}{\partial Z}\cdot dZ \tag{A.12}$$

Approximate infinitesimal changes of the differential for small changes with $\Delta$.

$$\Delta Y = \frac{\partial f(X,Z)}{\partial X}\cdot \Delta X + \frac{\partial f(X,Z)}{\partial Z}\cdot \Delta Z \tag{A.13}$$

$\Delta X$, $\Delta Y$, and $\Delta Z$ can be defined in terms of differences.

$$\Delta X = X - X_i \tag{A.14}$$

$$\Delta Y = Y - Y_i \tag{A.15}$$

$$\Delta Z = Z - Z_i \tag{A.16}$$

The partial derivatives are evaluated at the operating point and equations (A.14), (A.15), and (A.16) can be substituted into equation (A.13).

$$Y - Y_i = \frac{\partial f(X_i,Z_i)}{\partial X}\cdot (X - X_i) + \frac{\partial f(X_i,Z_i)}{\partial Z}\cdot (Z - Z_i) \tag{A.17}$$

Regroup the terms of equation (A.17).

$$Y = \frac{\partial f(X_i,Z_i)}{\partial X}\cdot X + \frac{\partial f(X_i,Z_i)}{\partial Z}\cdot Z + Y_i - \frac{\partial f(X_i,Z_i)}{\partial X}\cdot X_i - \frac{\partial f(X_i,Z_i)}{\partial Z}\cdot Z_i \tag{A.18}$$

Equation (A.11) must hold true about the operating point

$$Y_i = f(X_i, Z_i) \tag{A.19}$$

Substitute equation (A.19) into (A.18).

$$Y = \frac{\partial f(X_i, Z_i)}{\partial X} \cdot X + \frac{\partial f(X_i, Z_i)}{\partial Z} \cdot Z + f(X_i, Z_i) - \frac{\partial f(X_i, Z_i)}{\partial X} \cdot X_i - \frac{\partial f(X_i, Z_i)}{\partial Z} \cdot Z_i \tag{A.20}$$

The last three terms of equation (A.20) may or may not cancel one another. Those terms represent the offset of the dependent variables at the operating point.

## Linearized System Equations

System equations for this study were linearized using the technique presented. Algebra of combining the linearized equations was simplified by incorporating the initial conditions prior to combining them. Since the operating point was the origin of the pressure-flow curves a number of the linearization constants were equal to zero.

The initial conditions which provided the simplifications were the following:

$$X_i = 0 \tag{A.21}$$

$$Y_i = 0 \tag{A.22}$$

$$sY_i = 0 \tag{A.23}$$

$$P_{3i} = P_{4i} \tag{A.24}$$

$$P_{1i} = P_{2i} \tag{A.25}$$

$$P_{Mi} = 0 \tag{A.26}$$

$$P_{5i} = P_E \tag{A.27}$$

$$P_{6i} = P_E \tag{A.28}$$

$$Q_{Ai} = Q_{Ci} \tag{A.29}$$

$$Q_{Di} = Q_{Fi} \tag{A.30}$$

$$sP_{1i} = 0 \tag{A.31}$$

$$sP_{2i} = 0 \tag{A.32}$$

$$sP_{3i} = 0 \tag{A.33}$$

$$sP_{4i} = 0 \tag{A.34}$$

$$Q_H = -Q_J \tag{A.35}$$

$$P_5 = -P_6 \tag{A.36}$$

The definition of each linearization constant from the linearized equations presented in Chapter II is now presented. These constants are derivatives and partial derivatives of the equation evaluated at the operating point. The original linearized equation is also presented for the sake of continuity.

Error signal to the amplifier,

$$II = I + C_1 \cdot \theta + C_{31} \cdot P_M + C_{32} \cdot P_D \tag{A.37}$$

$$C_1 = -K_F \tag{A.38}$$

$$C_{31} = -K_P \cdot S_3 \tag{A.39}$$

$$C_{32} = -K_D \cdot S_5 \tag{A.40}$$

Pressure forces acting on the flapper nozzle valve,

$$F_1 = C_2 \cdot (P_3 - P_4) \tag{A.41}$$

$$F_2 = C_{33} \cdot (P_5 - P_6) \tag{A.42}$$

$$C_2 = 0.25 \cdot \pi \cdot D_1^2 \tag{A.43}$$

$$C_{33} = 0.25 \cdot \pi \cdot D_7^2 \tag{A.44}$$

Mechanical spring force between first and second stages,

$$F_S = C_{23} \cdot (X - Y) \tag{A.45}$$

$$C_{23} = K_3 \tag{A.46}$$

Displacement of the torquemotor,

$$X = C_3 \cdot II + C_4 \cdot Y + C_5 \cdot F_1 + C_5 \cdot F_2 \tag{A.47}$$

$$C_3 = K_1 \cdot K_A \,/\, (K_2 + K_3) \tag{A.48}$$

$$C_4 = K_3 \,/\, (K_2 + K_3) \tag{A.49}$$

$$C_5 = 1 \,/\, (K_2 + K_3) \tag{a.50}$$

Flow through fixed orifice upstream of flapper valve left side,

$$Q_A = C_6 \cdot P_3 + C_7 \tag{A.51}$$

$$C_6 = -0.125 \cdot \pi \cdot D_3^2 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (P_S - P_{3i})^{-\frac{1}{2}} \tag{A.52}$$

$$C_7 = 0.125 \cdot \pi \cdot D_3^2 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (P_S - P_{3i})^{-\frac{1}{2}} \cdot P_{3i}$$

$$+ 0.25 \cdot \pi \cdot D_3^2 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (P_S - P_{3i})^{\frac{1}{2}} \tag{A.53}$$

Flow through left side of flapper nozzle valve,

$$Q_C = C_8 \cdot X + C_9 \cdot P_3 + C_{10} \tag{A.54}$$

$$C_8 = \pi \cdot D_1 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (P_{3i} - P_E)^{\frac{1}{2}} \tag{A.55}$$

$$C_9 = 0.5 \cdot \pi \cdot D_1 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (X_0 + X_i) \cdot (P_{3i} - P_E)^{-\frac{1}{2}} \tag{A.56}$$

$$C_{10} = \pi \cdot D_1 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (X_0 + X_i) \cdot (P_{3i} - P_E)^{\frac{1}{2}}$$

$$- \pi \cdot D_1 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (P_{3i} - P_E)^{\frac{1}{2}} \cdot X_i$$

$$-0.5 \cdot \pi \cdot D_1 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (X_0 + X_i) \cdot (P_{3i} - P_E)^{-\frac{1}{2}} \cdot P_{3i} \tag{A.57}$$

Flow through right side of flapper nozzle valve,

$$Q_F = C_{11} \cdot X + C_{12} \cdot P_4 + C_{13} \tag{A.58}$$

$$C_{11} = -\pi \cdot D_1 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (P_{4i} - P_E)^{\frac{1}{2}} \tag{A.59}$$

$$C_{11} = -C_8 \tag{A.60}$$

$$C_{12} = 0.5 \cdot \pi \cdot D_1 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (X_0 - X_i) \cdot (P_{5i} - P_E)^{-\frac{1}{2}} \tag{A.61}$$

$$C_{12} = C_9 \tag{A.62}$$

$$C_{13} = \pi \cdot D_1 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (P_{4i} - P_E)^{\frac{1}{2}} \cdot X_i$$
$$- 0.5 \cdot \pi \cdot D_1 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (X_0 - X_i) \cdot (P_{4i} - P_E)^{-\frac{1}{2}} \cdot P_{4i}$$
$$+ \pi \cdot D_1 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (X_0 - X_i) \cdot (P_{4i} - P_E)^{\frac{1}{2}} \tag{A.63}$$

$$C_{13} = C_{10} \tag{A.64}$$

Flow through the fixed orifice upstream of right side of flapper valve,

$$Q_D = C_{14} \cdot P_4 + C_{15} \tag{A.65}$$

$$C_{14} = -0.125 \cdot \pi \cdot D_3^2 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (P_S - P_{4i})^{-\frac{1}{2}} \tag{A.66}$$

$$C_{14} = C_6 \tag{A.67}$$

$$C_{15} = 0.125 \cdot \pi \cdot D_3^2 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (P_S - P_{4i})^{-\frac{1}{2}} \cdot P_{4i}$$
$$+ 0.25 \cdot \pi \cdot D_3^2 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (P_S - P_{4i})^{\frac{1}{2}} \tag{A.68}$$

$$C_{15} = C_7 \tag{A.69}$$

Time rate of pressure change on left side of valve spool,

$$sP_3 = C_{34} \cdot Y + C_{35} \cdot (Q_A - Q_C) + C_{36} \cdot sY + C_{37} \tag{A.70}$$

$$C_{34} = \frac{-\beta}{(Y_0 + Y_i)^2}\left(\frac{Q_{Ai} - Q_{Ci}}{AE} - sY_i\right) = 0 \qquad (A.71)$$

$$C_{35} = \frac{\beta}{AE\ (Y_0 + Y_i)} \qquad (A.72)$$

$$C_{36} = \frac{-\beta}{(Y_0 + Y_i)} \qquad (A.73)$$

$$C_{37} = -C_{34}\bullet Y_i - C_{35}\bullet(Q_{Ai} - Q_{Ci}) - C_{36}\bullet sY_i + sP_{3i} = 0 \qquad (A.74)$$

Time rate of pressure change on right side of valve spool,

$$sP_4 = C_{38}\bullet Y + C_{39}\bullet(Q_D - Q_F) + C_{40}\bullet sY + C_{41} \qquad (A.75)$$

$$C_{38} = \frac{\beta}{(Y_0 - Y_i)^2}\left(\frac{Q_{Di} - Q_{Fi}}{AE} + sY_i\right) = 0 \qquad (A.76)$$

$$C_{39} = \frac{\beta}{AE\ (Y_0 - Y_i)} \qquad (A.77)$$

$$C_{39} = C_{35} \qquad (A.78)$$

$$C_{40} = \frac{\beta}{(Y_0 - Y_i)} \qquad (A.79)$$

$$C_{41} = -C_{38}\bullet Y_i - C_{39}\bullet(Q_{Di} - Q_{Fi}) - C_{40}\bullet sY_i + sP_{4i} = 0 \qquad (A.80)$$

Flow through the valve and motor,

$$Q_M = C_{69}\bullet Y + C_{70}\bullet P_M \qquad (A.81)$$

$$C_{69} = -\ (QMAX \div YMAX)\bullet[1 - (P_M \div P_S)]^{\frac{1}{2}} \qquad (A.82)$$

$$C_{70} = -\ (QMAX \div P_S)\bullet(I \div IMAX)\bullet[1 - (P_M \div P_S)]^{-\frac{1}{2}} \qquad (A.83)$$

Time rate of pressure change in right load line,

$$sP_1 = C_{42}\bullet s\theta + C_{43}\bullet P_M + C_{44}\bullet Q_M \qquad (A.84)$$

$$C_{22} = \frac{C_S \cdot D_M}{\mu} \tag{A.85}$$

$$C_{42} = \frac{\beta \cdot D_M}{V_1} \tag{A.86}$$

$$C_{43} = \frac{C_{22} \cdot \beta}{V_1} \tag{A.87}$$

$$C_{44} = \frac{-\beta}{V_1} \tag{A.88}$$

Time rate of pressure change in left load line,

$$sP_2 = C_{45} \cdot s\theta + C_{46} \cdot P_M + C_{47} \cdot Q_M \tag{A.89}$$

$$C_{45} = \frac{-\beta \cdot D_M}{V_2} \tag{A.90}$$

$$C_{46} = \frac{-C_{22} \cdot \beta}{V_2} \tag{A.91}$$

$$C_{47} = \frac{\beta}{V_2} \tag{A.92}$$

The electrical dynamic pressure feedback element,

$$P_D = \frac{C_{48} \cdot sP_M}{\tau s + 1} \tag{A.93}$$

$$C_{48} = \tau \cdot S_5 \tag{A.94}$$

Flow from mechanical dynamic pressure feedback unit on left side,

$$Q_H = C_{59} \cdot X + C_{60} \cdot P_5 + C_{61} \tag{A.95}$$

$$C_{59} = \pi \cdot D_7 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (P_{5i} - P_E)^{\frac{1}{2}} = 0 \tag{A.96}$$

$$C_{60} = 0.5 \cdot \pi \cdot D_7 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (X_0 + X_i) \cdot (P_{5i} - P_E)^{-\frac{1}{2}} \tag{A.97}$$

$$C_{60} = \infty \tag{A.98}$$

$$C_{61} = Q_{Hi} - \pi \cdot C_D \cdot D_7 \cdot (2/\rho)^{\frac{1}{2}} \cdot (P_{5i} - P_E)^{\frac{1}{2}} \cdot X_i$$

$$- 0.5 \cdot \pi \cdot C_D \cdot D_7 \cdot (2/\rho)^{\frac{1}{2}} \cdot (X_0 + X_i) \cdot (P_{5i} - P_E)^{-\frac{1}{2}} \cdot P_{5i} \tag{A.99}$$

$$C_{61} = 0 \tag{A.100}$$

Flow from right side of mechanical dynamic pressure feedback unit,

$$Q_J = C_{62} \cdot X + C_{63} \cdot P_6 + C_{64} \tag{A.101}$$

$$C_{62} = -\pi \cdot D_7 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (P_{6i} - P_E)^{\frac{1}{2}} = 0 \tag{A.102}$$

$$C_{63} = 0.5 \cdot \pi \cdot D_9 \cdot C_D \cdot (2/\rho)^{\frac{1}{2}} \cdot (X_0 - X_i) \cdot (P_{6i} - P_E)^{-\frac{1}{2}} \tag{A.103}$$

$$C_{63} = \infty \tag{A.104}$$

$$C_{64} = Q_{Ji} + \pi \cdot C_D \cdot D_7 \cdot (2/\rho)^{\frac{1}{2}} \cdot (P_{6i} - P_E)^{\frac{1}{2}} \cdot X_i$$

$$- 0.5 \cdot \pi \cdot C_D \cdot D_7 \cdot (2/\rho)^{\frac{1}{2}} \cdot (X_0 - X_i) \cdot (P_{6i} - P_E)^{-\frac{1}{2}} P_{6i} \tag{A.105}$$

$$C_{64} = 0 \tag{A.106}$$

$$Q = Q_J = -Q_H = -C_{60} \cdot P_6 \tag{A.107}$$

Viscous damping force on the valve spool,

$$F_V = C_{65} \cdot sY \tag{A.108}$$

$$C_{65} = \mu \cdot \pi \cdot D_2 \cdot L_D \; / \; C_R \tag{A.109}$$

Steady state flow forces on valve spool,

$$F_{SS1} = C_{16} \cdot Y + C_{17} \cdot P_M + C_{18} \tag{A.110}$$

$$C_{16} = -C_V \cdot C_D \cdot \pi \cdot D_2 \cdot (P_S - P_E + P_{Mi}) \cdot \cos 69° \tag{A.111}$$

$$C_{17} = -C_V \cdot C_D \cdot \pi \cdot D_2 \cdot Y_i \cdot \cos 69° = 0 \tag{A.112}$$

$$C_{18} = C_V \cdot C_D \cdot \pi \cdot D_2 \cdot (P_S - P_E + P_{Mi}) \cdot Y_i$$

$$- C_V \cdot C_D \cdot \pi \cdot D_2 \cdot Y_i \cdot P_{Mi}$$

$$+ C_V \cdot C_D \cdot \pi \cdot D_2 \cdot Y_i \cdot (P_S - P_E + P_{Mi}) = 0 \quad \text{(A.113)}$$

$$F_{SS2} = C_{19} \cdot Y + C_{20} \cdot P_M + C_{21} \quad \text{(A.114)}$$

$$C_{19} = -C_V \cdot C_D \cdot \pi \cdot D_2 \cdot (P_S - P_E - P_{Mi}) \cdot \cos 69° \quad \text{(A.115)}$$

$$C_{20} = -C_V \cdot C_D \cdot \pi \cdot D_2 \cdot Y_i \cdot \cos 69° = 0 \quad \text{(A.116)}$$

$$C_{21} = C_V \cdot C_D \cdot \pi \cdot D_2 \cdot (P_S - P_E - P_{Mi}) \cdot Y_i$$

$$+ -C_V \cdot C_D \cdot \pi \cdot D_2 \cdot Y_i \cdot P_{Mi}$$

$$+ -C_V \cdot C_D \cdot \pi \cdot D_2 \cdot Y_i \cdot (P_S - P_E - P_{Mi}) = 0 \quad \text{(A.117)}$$

Acceleration of the valve spool,

$$s^2 Y = C_{66} \cdot F_S + C_{66} \cdot AE \cdot (P_3 - P_4) + C_{66} \cdot AF \cdot P_M$$

$$+ C_{66} \cdot F_{SS1} + C_{66} \cdot F_{SS2} - C_{66} \cdot F_V \quad \text{(A.118)}$$

$$C_{66} = 1 \;/\; M \quad \text{(A.119)}$$

Angular acceleration of the shaft,

$$s^2 \theta = C_{67} \cdot P_M + C_{68} \cdot s\theta \quad \text{(A.120)}$$

$$C_{67} = (1 - C_F) \cdot D_M \;/\; J \quad \text{(A.121)}$$

$$C_{68} = -(C_{DM} \cdot D_M \cdot \mu + Visd) \;/\; J \quad \text{(A.122)}$$

The linearized equations which describe the mechanical dynamic pressure feedback high pass filter are combined to form its transfer function. The constant $C_{60}$ is the slope of the pressure-flow curve for the feedback orifices. Initially $C_{60}$ has a value of infinity but very quickly drops to a finite value. $C_{60}$ was assigned an effective slope for simulation purposes. The form of the feedback element transfer function is

$$\frac{P_D}{P_M} = \frac{\dfrac{- AP \cdot AG}{2 \cdot K_4 \cdot C_{60}} \cdot s}{\dfrac{AP^2}{2 \cdot K_4 \cdot C_{60}} \cdot s + 1} \tag{A.123}$$

# APPENDIX B

## NON-LINEAR SIMULATION PROGRAM

The CSMP program listing for the non-linear system simulations is contained in this appendix. Equations used in the simulation are the equations presented in Chapter II. The program includes the capabilities of adding mechanical or electrical pressure feedback, mechanical or electrical dynamic pressure feedback to the basic uncompensated servosystem.

This particular simulation solves all the system equations simultaneously. Any of the system variables is available for printer plots. Time, angular position, velocity, and acceleration are written to a TSO data set for continuous plotting purposes. Using other programs available for plotting purposes the data can be viewed as a continuous graph on a Tektronix CRT.

Basically the CSMP program consists of three main segments. These are INITIAL, DYNAMIC, and TERMINAL segments. The INITIAL segment must appear first. It sets up the information required to perform simulation and is therefore executed only once. PARAMETER, CONSTANT, and INCON statements within the INITIAL segment contain values of system parameters, constants, and initial conditions to be used in the simulation.

The DYNAMIC segment contains the equations actually used to describe the system. This program segment is continually re-executed with a specified time step for the duration of the simulation. Integrations are performed with this segment during the course of the simulation.

The TERMINAL segment follows the DYNAMIC segment and like the INITIAL segment is executed only once. Output is set-up, integration technique defined, integration time step fixed, and simulation time duration specified. The particular integration routine used through the course of this study was a fixed step Runge-Kutta. The Runge-Kutta does quite well for general engineering work due to its low error. Close match between experimental and simulation results attest to this fact.

Additional capability is added to the CSMP simulation with MACRO statements which precede the INITIAL segment. MACRO statements are executable subprograms called by the INITIAL or DYNAMIC segments to make internal program changes during execution. When MACRO statements are included in the DYNAMIC segment they are executed as any other statement at each step of the integration.

```
****CONTINUOUS SYSTEM MODELING PROGRAM****

*** VERSION 1.3 ***

*    THE MACRO MODEL SETS THE CONSTANTS TO ZERO OUT THE INCORRECT
*    MODEL AND ALLOWS THE CORRECT MODEL TO BE SIMULATED
MACRO  S1,S2,S3,S4,S5=MODEL(SET)
PROCEDURAL
      S1=1.0
      S2=0.0
      S3=0.0
      S4=0.0
      S5=0.0
      IF (SET.NE.2.0)  GO TO 75
      S1=0.0
      S2=1.0
      GO TO 105
   75 IF (SET.NE.3.0)  GO TO 85
      S1=0.0
      S3=1.0
      GO TO 105
   85 IF (SET.NE.4.0)  GO TO 95
      S1=0.0
      S4=1.0
      GO TO 105
   95 IF (SET.NE.5.0)  GO TO 105
      S1=0.0
      S5=1.0
  105 CONTINUE
ENDMAC

*    THE MACRO TQMP STOPS THE TORQUEMOTOR DISPLACEMENT ONCE IT REACHES
*    MAXIMUM DISPLACEMENT IN EITHER DIRECTION
MACRO  XX=TQMP(X,XO)
PROCEDURAL
      XX=X
      IF (X.GE.(-1.0*XO))  GO TO 115
      XX=-1.0*XO
  115 IF (X.LE.XO)  GO TO 125
      XX=XO
  125 CONTINUE
ENDMAC

*    THE MACRO GO1 DETERMINES DIRECTION OF OIL FLOW DEPENDING UPON
*    THE PRESSURE DIFFERENTIAL ACROSS THE ORIFICE
MACRO  GO1=PILOT1(P1,P4P)
PROCEDURAL
      GO1=1.0
      IF (P4P.GT.P1)  GO1=-1.0
ENDMAC

*    THE MACRO GO2 DETERMINES DIRECTION OF OIL FLOW DEPENDING UPON
*    THE PRESSURE DIFFERENTIAL ACROSS THE ORIFICE
MACRO  GO2=PILOT2(P4P,PE)
PROCEDURAL
      GO2=1.0
      IF (P4P.LT.PE) GO2=-1.0
ENDMAC

*    THE MACRO GO3 DETERMINES DIRECTION OF OIL FLOW DEPENDING UPON
*    THE PRESSURE DIFFERENTIAL ACROSS THE ORIFICE
MACRO  GO3=PILOT3(P5P,PE)
PROCEDURAL
```

```
      GO3=1.0
      IF (P5P.LT.PE) GO3=-1.0
ENDMAC

*    THE MACRO GO4 DETERMINES DIRECTION OF OIL FLOW DEPENDING UPON
*    THE PRESSURE DIFFERENTIAL ACROSS THE ORIFICE
MACRO  GO4=PILOT4(P1,P5P)
PROCEDURAL
      GO4=1.0
      IF (P5P.GT.P1) GO4=-1.0
ENDMAC

*    THE MACRO SWITCH CHANGES THE VALVE PRESSURE DROP EQUATIONS TO
*    THE APPROPRIATE PRESSURES DEPENDING UPON THE SIDE OF NULL THE
*    SPOOL IS LOCATED
MACRO  A,B,SGN=SWITCH(Y)
PROCEDURAL
      IF (Y) 35,25,25
   25 A=0.0
      B=1.0
      SGN=1.0
      GO TO 45
   35 A=1.0
      B=0.0
      SGN=-1.0
   45 CONTINUE
ENDMAC

*    THE MACRO FILT6 PREVENTS THE PRESSURE P6 FROM CAVITATING
MACRO  P6P=FILT6(P6,PE)
PROCEDURAL
      P6P=P6
      IF (P6.LT.PE)  P6P=PE
ENDMAC

*    THE MACRO FILT7 PREVENTS THE PRESSURE P7 FROM CAVITATING
MACRO  P7P=FILT7(P7,PE)
PROCEDURAL
      P7P=P7
      IF (P7.LT.PE)  P7P=PE
ENDMAC

*    THE MACRO DERI3 DETERMINES FLOW DIRECTION ACROSS ONE SPOOL LAND
*    DEPENDING UPON THE PRESSURE DIFFERENTIAL
MACRO  SP=DERI3(DELP3)
PROCEDURAL
      SP=1.0
      IF (DELP3.LT.0.0) SP=-1.0
ENDMAC

*    THE MACRO DERI4 DETERMINES FLOW DIRECTION ACROSS ONE SPOOL LAND
*    DEPENDING UPON THE PRESSURE DIFFERENTIAL
MACRO  TP=DERI4(DELP4)
PROCEDURAL
      TP=1.0
      IF (DELP4.LT.0.0) TP=-1.0
ENDMAC

*    THE MACRO FLOFOR PREVENTS THE UNSTEADY FLOW FORCES FROM GOING TO
*    INFINITY AS THE PRESSURE DROP ACROSS THE LAND GOES TO ZERO
MACRO  DEL3,DEL4,C,D=FLOFOR(DELP3,DELP4)
PROCEDURAL
      C=1.0
      D=1.0
      DEL3=DELP3
      DEL4=DELP4
```

```
      IF (ABS(DEL3).GT.1.0)  GO TO 55
      C=0.0
      DEL3=1.0
   55 IF (ABS(DEL4).GT.1.0)  GO TO 65
      D=0.0
      DEL4=1.0
   65 CONTINUE
ENDMAC

*    THE MACRO COUL DETERMINES THE DIRECTION OF COULOMB FRICTION
MACRO  FDIR=COUL(THED)
PROCEDURAL
      IF (THED)  135,145,155
  135 FDIR=-1.0
      GO TO 165
  145 FDIR=0.0
      GO TO 165
  155 FDIR=1.0
  165 CONTINUE
ENDMAC

INITIAL
*    THE PARAMETER SET DETERMINES WHICH MODEL IS BEING STUDIED
*        SET=1.0  UNCOMPENSATED MODEL
*        SET=2.0  MECHANICAL PRESSURE FEEDBACK MODEL
*        SET=3.0  ELECTRICAL PRESSURE FEEDBACK MODEL
*        SET=4.0  MECHANICAL DYNAMIC PRESSURE FEEDBACK MODEL
*        SET=5.0  ELECTRICAL DYNAMIC PRESSURE FEEDBACK MODEL
    PARAMETER         SET=3.0

*    I - STEP INPUT TO THE SYSTEM (VOLTS)
    PARAMETER         I=1.5

*    BETA - OIL BULK MODULUS (PSI)
*    MU - OIL ABSOLUTE VISCOSITY (LBF*SEC/IN**2)
    PARAMETER         BETA=150000.0, MU=2.0E-06,...

*    CD - ORIFICE DISCHARGE COEFFICIENT
*    RHO - OIL DENSITY (LBF*SEC**2/IN**4)
*    P1 - SUPPLY PRESSURE (PSI)
     CD=0.625,        RHO=7.85E-05,  P1=1100.0,...

*    V1 - VOLUME UNDER COMPRESSION OF P3 (IN**3)
*    V2 - VOLUME UNDER COMPRESSION OF P2 (IN**3)
*    ANG - COS 69 DEG.  USED IN STEADY STATE FLOW FORCES
     V1=25.0,         V2=25.0,        ANG=0.3584

*    CS - MOTOR SLIP COEFFICIENT
*    CV - FLUID VELOCITY COEFFICIENT FOR STEADY STATE FLOW FORCES
    PARAMETER         CS=0.88E-08,   CV=0.98,...

*    CF - MOTOR FRICTION COEFFICIENT
*    CDM - MOTOR VISCOUS DRAG COEFFICIENT
*    DM - MOTOR DISPACEMENT (IN**3/RAD)
     CF=0.10,         CDM=160000.0,  DM=1.512E-02,...

*    J - MOTOR ROTARY INERTIA (IN*LBF*SEC**2/RAD)
*    CFRIC - COULOMB FRICTION (IN*LBF)
*    TL - EXTERNAL LOAD TORQUE (IN*LBF)
     J=2.16E-03,      CFRIC=0.0,      TL=0.0,...

*    VISD - VISCOUS DRAG EXTERNAL TO MOTOR (IN*LBF*SEC)
     VISD=0.018

*    KWFL - SERVOAMPLIFIER GAIN (MA/VOLT)
*        KWFL1 UNCOMPENSATED MODEL SERVOAMPLIFIER GAIN
```

```
*       KWFL2 MECHANICAL PRESSURE FEEDBACK MODEL SERVOAMPLIFIER GAIN
*       KWFL3 ELECTRICAL PRESSURE FEEDBACK MODEL SERVOAMPLIFIER GAIN
*       KWFL4 MECHANICAL DPF MODEL SERVOAMPLIFIER GAIN
*       KWFL5 ELECTRICAL DPF MODEL SERVOAMPLIFIER GAIN
    PARAMETER        KWFL1=0.667,   KWFL2=0.57,...
      KWFL3=0.667,   KWFL4=1.00,    KWFL5=0.747

*    K1 - TORQUEMOTOR CONSTANT (LBF/MA)
*    K2 - TORQUEMOTOR CONSTANT (LBF/IN)
    PARAMETER        K1=0.05,       K2=140.0,...

*    K3 - MECHANICAL FEEDBACK CONSTANT (LBF/IN)
*    K5 - SPRING RATE IN MECH DPF UNIT (LBF/IN)
      K3=22.5,       K5=200.0

*    D1 - FLAPPER NOZZLE DIAMETER (IN)
*    D2 - VALVE SPOOL DIAMETER (IN)
    PARAMETER        D1=0.023,      D2=0.275,...

*    D3 - ORIFICE DIAMETER UPSTREAM OF FLAPPER (IN)
*    D6 - DIAMETER OF AREA ON WHICH MECH PRESSURE FEEDBACK ACTS (IN)
*    D7 - LARGE DIAMETER OF MECH DPF PISTON (IN)
      D3=0.012,      D6=0.05,       D7=1.7039,...

*    D8 - SMALL DIAMETER OF MECH DPF PISTON (IN)
*    D9 - DIAMETER OF MECH DPF ORIFICE WHICH ACTS ON FLAPPER (IN)
      D8=1.3215,     D9=0.010

*    XO - NULL DISPLACEMENT OF FLAPPER (IN)
*    YO - NULL LENGTH OF VOLUME ON EACH END OF SPOOL (IN)
    PARAMETER        XO=0.0018,     YO=0.40,...

*    CR - VALVE SPOOL RADIAL CLEARANCE (IN)
*    LD - SPOOL LENGTH FOR VISCOUS DAMPING (IN)
*    L - DISTANCE BETWEEN PORTS FOR UNSTEADY FLOW FORCES (IN)
      CR=0.00005,    LD=0.384,      L=0.29,...

*    M - MASS OF SPOOL VALVE (LBF*SEC**2/IN**4)
      M=3.2071E-05

*    KFBK - POSITION FEEDBACK GAIN (VOLTS/RAD)
*    KAMP - ELECTRICAL PRESSURE FEEDBACK GAIN (VOLTS/PSI)
    PARAMETER        KFBK=2.00,     KAMP=6.5E-03,...

*    KAMPD - ELECTRICAL DPF FEEDBACK GAIN (VOLTS/PSI)
*    TAU3 - ELECTRICAL DPF TIME CONSTANT (1/SEC)
      KAMPD=5.0E-03,         TAU3=0.32

    CONSTANT  FLAG=9.8765E+00,  COUNT=0.0,  CYCLE=19.0
    CONSTANT         PI=3.14159

*    YDIC - INITIAL SPOOL VELOCITY (IN/SEC)
*    YIC - INITIAL SPOOL DISPLACEMENT (IN)
    INCON            YDIC=0.0,      YIC=0.0,...

*    THETIC - INITIAL MOTOR DISPLACEMENT (RAD)
*    THEDIC - INITIAL MOTOR VELOCITY (RAD/SEC)
*    CAPIC - INITAL VALUE OF REAL POLE
      THETIC=0.0,    THEDIC=0.0,    CAPIC=0.0,...

*    ZIC - INITIAL MECH DPF UNIT DISPLACEMENT (IN)
      ZIC=0.0

*    PE - EXHAUST PRESSURE (PSI)
    PARAMETER        PE=0.0
```

```
*     P2IC,P3IC -INITIAL PRESSURES IN LOAD LINES (PSI)
       P2IC=0.5*(P1+PE)
       P3IC=0.5*(P1+PE)

       S1,S2,S3,S4,S5=MODEL(SET)
       KWFL=KWFL1*S1+KWFL2*S2+KWFL3*S3+KWFL4*S4+KWFL5*S5
       CON1=CD*PI*D2*SQRT(2.0/RHO)
       CON2=0.25*PI*D3**2*CD*SQRT(2.0/RHO)
       CON3=CD*PI*D1*SQRT(2.0/RHO)
       CON4=-2.0*CV*CD*PI*D2*ANG
       CON5=SQRT(RHO)*L*CD*PI*D2
       CON6=MU*PI*D2*LD/CR
       CON10=0.25*PI*D1**2
       CON12=0.25*PI*D9**2
       CON14=CD*PI*D9*SQRT(2.0/RHO)

*     AF - AREA ON SPOOL MECH PRESSURE FEEDBACK ACTS (IN**2)
       AF=0.25*PI*D6**2*S2

*     AE - NET AREA ON SPOOL END FOR CONTROL (IN**2)
       AE=0.25*PI*D2**2-AF

*     AG - AREA OF MECH DPF UNIT ON WHICH LINE PRESSURE ACTS (IN**2)
       AG=0.25*PI*D8**2

*     AP - AREA OF MECH DPF UNIT WHICH HOLDS PRESSURE TO FLAPPER (IN**2)
       AP=D7**2*0.25*PI-AG

*     P4IC,P5IC - INITIAL PRESSURE ON ENDS OF VALVE SPOOL (PSI)
       P4IC=(P1*D3**4+16.0*PE*D1**2*X0**2)/(D3**4+16.0*D1**2*X0**2)

       P5IC=P4IC

DYNAMIC
       A,B,SGN=SWITCH (Y)

*     DELP3,DELP4 - PRESSURE DROP ACROSS THE TWO VALVE LANDS
       DELP3=(B*P1-SGN*P3P-A*PE)
       DELP4=(A*P1+SGN*P2P-B*PE)

       DEL3,DEL4,C,D=FLOFOR(DELP3,DELP4)
       SP=DERI3(DELP3)
       TP=DERI4(DELP4)

*     QQ1,QQ4 - FLOW THROUGH THE TWO LOAD LINES
       QQ1=-1.0*Y*CON1*SQRT(ABS(DELP4))*TP
       QQ4=-1.0*Y*CON1*SQRT(ABS(DELP3))*SP

*     II - ERROR SIGNAL FED TO SERVOAMPLIFIER
       II=I-KFBK*THET-KAMP*DPM*S3-KAMPD*DPF*S5

*     X - DISPLACEMENT OF TORQUEMOTOR
       X=(K1*KWFL*II+K3*Y+TMF1+TMF2)/(K3+K2)

*     TMF1 - NET FORCE ACTING ON FLAPPER BY PRESSURES ON END OF SPOOL
       TMF1=(P4P-P5P)*CON10

*     TMF2 - NET FORCE ACTING ON FLAPPER BY MECH DPF UNIT
       TMF2=(P6P-P7P)*CON12

       XX=TQMP(X,X0)
       GO1=PILOT1(P1,P4P)

*     QA - FLOW THROUGH ONE FIXED ORIFICE UPSTREAM OF FLAPPER
       QA=CON2*SQRT(ABS(P1-P4P))*GO1
```

```
      GO2=PILOT2(P4P,PE)
*     QC - FLOW THROUGH ONE SIDE OF FLAPPER NOZZLE
      QC=CON3*(XO+XX)*SQRT(ABS(P4P-PE))*GO2
      GO3=PILOT3(P5P,PE)
*     QF - FLOW THROUGH ONE SIDE OF FLAPPER NOZZLE
      QF=CON3*(XO-XX)*SQRT(ABS(P5P-PE))*GO3
      GO4=PILOT4(P1,P5P)
*     QD - FLOW THROUGH ONE FIXED ORIFICE UPSTREAM OF FLAPPER
      QD=CON2*SQRT(ABS(P1-P5P))*GO4
*     FSS1,FSS2 - STEADY STATE FLOW FORCES ACTING ON VALVE SPOOL
      FSS1=CON4*Y*ABS(P1-A*P2P-B*P3P)
      FSS2=CON4*Y*ABS(B*P2P+A*P3P-PE)
*     P4DOT - TIME RATE OF CHANGE OF PRESSURE ON ONE END OF SPOOL
      P4DOT=BETA*((QA-QC)/AE-YDOT)/(YO+Y)
      P4P = INTGRL (P4IC, P4DOT)
*     P5DOT - TIME RATE OF CHANGE OF PRESSURE ON ONE END OF SPOOL
      P5DOT=BETA*((QD-QF)/AE+YDOT)/(YO-Y)
      P5P = INTGRL (P5IC, P5DOT)
*     FUS1,FUS2 - UNSTEADY FLOW FORCES ACTING ON VALVE SPOOL
      FUSA1=SGN*CON5*SQRT(ABS(2.0*DELP3))*YDOT*SP
      FUSA2=-1.0*CON5*0.5*Y*(1.0/(SQRT(ABS(2.0*DEL3))))*P3DOT*C*SP
      FUS1=FUSA1+FUSA2
      FUSB1=-1.0*SGN*CON5*SQRT(ABS(2.0*DELP4))*YDOT*TP
      FUSB2=-1.0*CON5*0.5*Y*(1.0/(SQRT(ABS(2.0*DEL4))))*P2DOT*D*TP
      FUS2=FUSB1+FUSB2
*     P3DOT - TIME RATE OF CHANGE OF ONE LOAD LINE PRESSURE
      P3DOT=BETA*(DM*THED+CS*DM*(P2P-P3P)/MU-QQ4)/V1
      P3P=INTGRL(P3IC,P3DOT)
*     P2DOT - TIME RATE OF CHANGE OF ONE LOAD LINE PRESSURE
      P2DOT=BETA*(QQ1-DM*THED-CS*DM*(P2P-P3P)/MU)/V2
      P2P=INTGRL(P2IC,P2DOT)
*     DPMD - TIME RATE OF CHANGE OF LOAD LINE DIFFERENTIAL PRESSURE
      DPMD=P2DOT-P3DOT
      DPFD=DPMD*TAU3*S5
*     DPF - ELECTRICAL DYNAMIC PRESSURE FEEDBACK SIGNAL
      DPF=REALPL(CAPIC,TAU3,DPFD)
*     DPM - LOAD LINE DIFFERENTIAL PRESSURE
      DPM=P2P-P3P
      P6P=FILT6(P6,PE)
*     QI - FLOW THROUGH ONE SIDE OF MECH DPF UNIT
      QI=CON14*(XO+XX)*SQRT(ABS(P6P-PE))*S4
      P7P=FILT7(P7,PE)
*     QJ - FLOW THROUGH ONE SIDE OF MECH DPF UNIT
      QJ=CON14*(XO-XX)*SQRT(ABS(P7P-PE))*S4
*     ZDOT - TIME RATE OF DISPLACEMENT CHANGE OF MECH DPF UNIT
      ZDOT=(QJ/AP-QI/AP)*S4
      Z=INTGRL(ZIC,ZDOT)
```

```
*     P6,P7 - PRESSURE ON TWO SIDE OF MECH DPF UNIT
       P6=(2.0*K5*Z/AP-DPM*AG/AP)
       P7=(DPM*AG/AP-2.0*K5*Z/AP)

*     FVD - VISCOUS DAMPING FORCE ACTING ON VALVE SPOOL
       FVD=CON6*YDOT

*     FS - MECHANICAL SPRING FORCE BETWEEN FIRST AND SECOND STAGES
       FS=K3*(XX-Y)

*     YDDOT - ACCELERATION OF VALVE SPOOL
       YDDOT=(FS+(P4P-P5P)*AE+(P2P-P3P)*AF+FUS1+FUS2+FSS1+FSS2-FVD)/M
       YDOT=INTGRL(YDIC,YDDOT)
       Y=INTGRL(YIC,YDOT)

       FDIR=COUL(THED)

*     THEDD - ANGULAR ACCELERATION OF SERVOMOTOR
       THEDD=((1.0-CF)*DM*(P2P-P3P)-CDM*DM*MU*THED-CFRIC*FDIR-TL-VISD*...
          THED)/J
       THED=INTGRL(THEDIC,THEDD)
       THET=INTGRL(THETIC,THED)

       NOSORT
       CALL DEBUG (1,0.0)

*     THESE STATEMENTS WRITE TO TSO DATA SET FOR PLOTTING
       IF (KEEP.NE.1)  GO TO 500
       CYCLE=CYCLE+1.0
       IF (CYCLE.NE.20.0)  GO TO 500
       WRITE (8,600)  TIME,THET,THED,THEDD
   600 FORMAT (T5,4(E14.6,5X))
       CYCLE=0.0
       COUNT=COUNT+1.0
   500 CONTINUE

TERMINAL

       TIMER  FINTIM=0.65,DELT=5.0E-05,PRDEL=2.5E-03,OUTDEL=2.5E-03
       WRITE (8,700)  FLAG,COUNT
   700 FORMAT (T5,E14.6,5X,E14.6)
       METHOD  RKSFX
       PRTPLT  THET (II,XX,DPM)
END
STOP
```

APPENDIX C

TRANSFER FUNCTION SIMULATION

The CSMP program used to simulate the five transfer function valve models used in this study is contained in this appendix. These five models are the basic uncompensated servosystem with the means to add mechanical or electrical pressure feedback, mechanical or electrical dynamic pressure feedback to the basic system. Complete documentation is contained in the program.

Constants for the simulation are set up by the INITIAL segment. System parameters, constants, and initial conditions are entered through PARAMETER, CONSTANT, and INCON statements. Subscripted "C's" are the linearization constants defined in Appendix A. The algebra required to reduce the system equations to the transfer function is contained in the subscripted "T's" of the INITIAL segment. Constants inappropriate for a particular model are cancelled by switches set up by the MACRO MODEL.

Once the constants of the system transfer function are evaluated the dynamic response is produced by the DYNAMIC program segment. Values of time and actuator displacement are written into a TSO data set for continuous data plotting. Continuous plots are obtained through other programs.

Following the DYNAMIC segment is the TERMINAL segment. This segment sets up output, defines integration technique, sets integration time step, and sets the simulation duration time. Runge-Kutta integration proved to be quite adequate for the transfer function as it had been with the non-linear model simulation.

```
      ****CONTINUOUS SYSTEM MODELING PROGRAM****

                  *** VERSION 1.3 ***

*    THE MACRO MODEL SETS THE CONSTANTS TO ZERO OUT THE INCORRECT
*    MODEL AND ALLOWS THE CORRECT MODEL TO BE SIMULATED
MACRO  S1,S2,S3,S4,S5=MODEL(SET)
PROCEDURAL
      S1=1.0
      S2=0.0
      S3=0.0
      S4=0.0
      S5=0.0
      IF (SET.NE.2.0)  GO TO 75
      S1=0.0
      S2=1.0
      GO TO 105
   75 IF (SET.NE.3.0)  GO TO 85
      S1=0.0
      S3=1.0
      GO TO 105
   85 IF (SET.NE.4.0)  GO TO 95
      S1=0.0
      S4=1.0
      GO TO 105
   95 IF (SET.NE.5.0)  GO TO 105
      S1=0.0
      S5=1.0
  105 CONTINUE
ENDMAC

INITIAL
      NOSORT

*    THE PARAMETER SET DETERMINES WHICH MODEL IS BEING STUDIED
*        SET=1.0  UNCOMPENSATED MODEL
*        SET=2.0  MECHANICAL PRESSURE FEEDBACK MODEL
*        SET=3.0  ELECTRICAL PRESSURE FEEDBACK MODEL
*        SET=4.0  MECHANICAL DYNAMIC PRESSURE FEEDBACK MODEL
*        SET=5.0  ELECTRICAL DYNAMIC PRESSURE FEEDBACK MODEL
    PARAMETER         SET=3.0

*    I - STEP INPUT TO THE SYSTEM (VOLTS)
    PARAMETER         I=1.5

*    BETA - OIL BULK MODULUS (PSI)
*    MU - OIL ABSOLUTE VISCOSITY (LBF*SEC/IN**2)
    PARAMETER         BETA=150000.0, MU=2.0E-06,...

*    CD - ORIFICE DISCHARGE COEFFICIENT
*    RHO - OIL DENSITY (LBF*SEC**2/IN**4)
*    P1 - SUPPLY PRESSURE (PSI)
     CD=0.625,        RHO=7.85E-05,  P1=1100.0,...

*    V1 - VOLUME UNDER COMPRESSION OF P3 (IN**3)
*    V2 - VOLUME UNDER COMPRESSION OF P2 (IN**3)
*    ANG - COS 69 DEG.  USED IN STEADY STATE FLOW FORCES
     V1=25.0,         V2=25.0,         ANG=0.3584

*    CS - MOTOR SLIP COEFFICIENT
*    CV - FLUID VELOCITY COEFFICIENT FOR STEADY STATE FLOW FORCES
    PARAMETER         CS=0.88E-08,   CV=0.98,...
```

```
*    CF - MOTOR FRICTION COEFFICIENT
*    CDM - MOTOR VISCOUS DRAG COEFFICIENT
*    DM - MOTOR DISPACEMENT (IN**3/RAD)
      CF=0.10,        CDM=160000.0,  DM=1.512E-02,...

*    J - MOTOR ROTARY INERTIA (IN*LBF*SEC**2/RAD)
      J=2.16E-03

*    KWFL - SERVOAMPLIFIER GAIN (MA/VOLT)
*        KWFL1 UNCOMPENSATED MODEL SERVOAMPLIFIER GAIN
*        KWFL2 MECHANICAL PRESSURE FEEDBACK MODEL SERVOAMPLIFIER GAIN
*        KWFL3 ELECTRICAL PRESSURE FEEDBACK MODEL SERVOAMPLIFIER GAIN
*        KWFL4 MECHANICAL DPF MODEL SERVOAMPLIFIER GAIN
*        KWFL5 ELECTRICAL DPF MODEL SERVOAMPLIFIER GAIN
     PARAMETER        KWFL1=0.667,   KWFL2=0.667,...
      KWFL3=0.667,    KWFL4=0.667,   KWFL5=0.667

*    K1 - TORQUEMOTOR CONSTANT (LBF/MA)
*    K2 - TORQUEMOTOR CONSTANT (LBF/IN)
     PARAMETER        K1=0.05,       K2=140.0,...

*    K3 - MECHANICAL FEEDBACK CONSTANT (LBF/IN)
*    K5 - SPRING RATE IN MECH DPF UNIT (LBF/IN)
      K3=22.5,        K5=200.0

*    D1 - FLAPPER NOZZLE DIAMETER (IN)
*    D2 - VALVE SPOOL DIAMETER (IN)
     PARAMETER        D1=0.023,      D2=0.275,...

*    D3 - ORIFICE DIAMETER UPSTREAM OF FLAPPER (IN)
*    D6 - DIAMETER OF AREA ON WHICH MECH PRESSURE FEEDBACK ACTS (IN)
*    D7 - LARGE DIAMETER OF MECH DPF PISTON (IN)
      D3=0.012,       D6=0.05,       D7=1.7039,...

*    D8 - SMALL DIAMETER OF MECH DPF PISTON (IN)
*    D9 - DIAMETER OF MECH DPF ORIFICE WHICH ACTS ON FLAPPER (IN)
      D8=1.3215,      D9=0.010

*    XO - NULL DISPLACEMENT OF FLAPPER (IN)
*    YO - NULL LENGTH OF VOLUME ON EACH END OF SPOOL (IN)
     PARAMETER        XO=0.0018,     YO=0.40,...

*    CR - VALVE SPOOL RADIAL CLEARANCE (IN)
*    LD - SPOOL LENGTH FOR VISCOUS DAMPING (IN)
*    L - DISTANCE BETWEEN PORTS FOR UNSTEADY FLOW FORCES (IN)
      CR=0.00005,     LD=0.384,      L=0.29,...

*    M - MASS OF SPOOL VALVE (LBF*SEC**2/IN**4)
      M=3.2071E-05,   ZO=1.00

*    KFBK - POSITION FEEDBACK GAIN (VOLTS/RAD)
*    KAMP - ELECTRICAL PRESSURE FEEDBACK GAIN (VOLTS/PSI)
     PARAMETER        KFBK=2.000,    KAMP=6.5E-03,...

*    KAMPD - ELECTRICAL DPF FEEDBACK GAIN (VOLTS/PSI)
*    TAU3 - ELECTRICAL DPF TIME CONSTANT (1/SEC)
      KAMPD=0.1786E-01,   TAU3=0.3885

*    VISD - VISCOUS DRAG EXTERNAL TO MOTOR (IN*LBF*SEC)
     PARAMETER   VISD=0.018

     CONSTANT         PI=3.14159

*    PE - EXHAUST PRESSURE (PSI)
     INCON            PE=0.0
```

```
CONSTANT  FLAG=9.8765E+00,  COUNT=0.0,  CYCLE=19.0
PARAMETER    IMAX= 36.5,  PM=850.0,...
  YMAX=0.025

  S1,S2,S3,S4,S5=MODEL(SET)
  KWFL=KWFL1*S1+KWFL2*S2+KWFL3*S3+KWFL4*S4+KWFL5*S5

*  P2IC,P3IC -INITIAL PRESSURES IN LOAD LINES (PSI)
  P2IC=0.5*(P1+PE)
  P3IC=0.5*(P1+PE)

*  P4IC,P5IC - INITIAL PRESSURE ON ENDS OF VALVE SPOOL (PSI)
  P4IC=(P1*D3**4+16.0*PE*D1**2*XO**2)/(D3**4+16.0*D1**2*XO**2)
  P5IC=P4IC

  P6IC=PE
  P7IC=PE
  CON1=CD*PI*D2*SQRT(2.0/RHO)
  CON2=0.25*PI*D3**2*CD*SQRT(2.0/RHO)
  CON3=CD*PI*D1*SQRT(2.0/RHO)
  CON4=-2.0*CV*CD*PI*D2*ANG
  CON6=MU*PI*D2*LD/CR
  CON10=0.25*PI*D1**2
  CON12=0.25*PI*D9**2
  CON14=CD*PI*D9*SQRT(2.0/RHO)

*  AF - AREA ON SPOOL MECH PRESSURE FEEDBACK ACTS (IN**2)
  AF=0.25*PI*D6**2*S2

*  AE - NET AREA ON SPOOL END FOR CONTROL (IN**2)
  AE=0.25*PI*D2**2-AF

*  AG - AREA OF MECH DPF UNIT ON WHICH LINE PRESSURE ACTS (IN**2)
  AG=0.25*PI*D8**2

*  AP - AREA OF MECH DPF UNIT WHICH HOLDS PRESSURE TO FLAPPER (IN**2)
  AP=D7**2*0.25*PI-AG

  QMAX=PI*D2*YMAX*CD*SQRT((P1-PE)/RHO)
  C1=-1.0*KFBK
  C2=CON10
  C3=K1*KWFL/(K2+K3)
  C4=K3/(K2+K3)
  C5=1.0/(K2+K3)
  C6=-0.5*CON2/(SQRT(P1-P4IC))
  C7=0.5*CON2*P4IC/(SQRT(P1-P4IC))+CON2*SQRT(P1-P4IC)
  C8=CON3*SQRT(P4IC-PE)
  C9=0.5*CON3*XO/(SQRT(P4IC-PE))
  C10=CON3*XO*SQRT(P4IC-PE)-0.5*CON3*XO*P4IC/(SQRT(P4IC-PE))
  C16=CON4*(P1-P2IC)
  C22=CS*DM/MU
  C23=K3
  C24=-1.0*CON1*SQRT(P1-P2IC)
  C31=-1.0*KAMP*S3
  C32=-1.0*KAMPD*S5
  C33=CON12
  C34=0.0
  C35=BETA/(AE*YO)
  C36=-1.0*BETA/YO
  C42=BETA*DM/V1
  C43=BETA*C22/V1
  C44=-1.0*BETA/V1
  C45=-1.0*BETA*DM/V2
  C46=-1.0*BETA*C22/V2
  C47=BETA/V2
  C48=TAU3*S5
```

```
      C49=AG*S4
      C50=AP
      C51=-2.0*K5
      C53=BETA/(AP*ZO)
      C54=-1.0*BETA/ZO
*     C60 - EFFECTIVE SLOPE OF PRESSURE-FLOW CURVE OF MECH DPF UNIT
      C60=3.2824E-05
      C65=CON6
      C66=1.0/M
      C67=(1.0-CF)*DM/J
      C68=-1.0*(CDM*DM*MU+VISD)/J
*     C69 - VALVE FLOW GAIN
      C69=-1.0*(QMAX/YMAX)*SQRT(1.0-I*PM/(ABS(I)*P1))
*     C70 - VALVE PRESSURE GAIN
      C70=-1.0*(QMAX/P1)*(I/IMAX)/(SQRT(1.0-I*PM/(ABS(I)*P1)))
      T1=C43-C46+C70*(C44-C47)
      T2=1.0/(C67*C69*(C47-C44))
      T3=(T1-C68)/(C67*C69*(C47-C44))
      T4=-1.0*(T1*C68+C67*(C45-C42))/(C67*C69*(C47-C44))
      T5=C1*TAU3+C31*TAU3*(C45-C42)+C32*C48*(C45-C42)
      T6=C1*(TAU3*T1+1.0)+C31*(C45-C42)
      T7=C1*T1
      T8=C31*TAU3*(C47-C44)*C69+C32*C48*C69*(C47-C44)
      T9=C31*C69*(C47-C44)
      T10=C9*C35-C6*C35
      T11=C50*C49/(2.0*K5*C60)
      T12=C50*C50/(2.0*K5*C60)
      T13=-1.0*C69*(C47-C44)*T11
      T14=-1.0*T11*(C45-C42)
      T15=C3*T5*T12+C5*C33*T14*TAU3
      T16=C3*(T6*T12+T5*T10*T12+T5)+C5*C33*T14*(T10*TAU3+1.0)
      T17=C3*(T5*T10+T7*T12+T6*T10*T12+T6)+C5*C33*T14*T10
      T18=C3*(T6*T10+T7*T10*T12+T7)
      T19=C3*T7*T10
      T20=C4*T12*TAU3+2.0*C2*C5*C36*T12*TAU3
      T21=C3*T8*T12+C4*(T1*T12*TAU3+T12+T10*T12*TAU3+TAU3) ...
        +2.0*C2*C5*C36*(T12*T1*TAU3+T12+TAU3)+C5*C33*T13*TAU3
      T22=C3*(T9*T12+T8*T10*T12+T8)+C4*(T10*TAU3 +T1*T12+T1*T10  ...
        *T12*TAU3+T10*T12+T1*TAU3+1.0)+2.0*C2*C5*C36*(T1*T12+T1* ...
        TAU3+1.0)+C5*C33*T13*(T10*TAU3+1.0)
      T23=C3*(T8*T10+T9*T10*T12+T9)+C4*(T1*T10*T12+T1+T1*T10*TAU3 ...
        +T10)+2.0*C2*C5*C36*T1+C5*C33*T13*T10
      T24=C3*T9*T10+C4*T1*T10
      T25=T10+2.0*C2*C5*C8*C35
      T26=T1+T10+C65*C66
      T27=T1*T10+C66*C65*(T1+T10)-2.0*C36*C66*AE+C66*C23-2.0*C16*C66
      T28=C66*C65*T1*T10-2.0*C36*C66*AE*T1+C66*C23*(T1+T10)- ...
        C66*C69*AF*(C47-C44)-2.0*C16*C66*(T1+T10)
      T29=C66*C23*T1*T10-C66*C69*AF*(C47-C44)*T10-2.0*C16*C66*T1*T10
      T30=C66*C23*T10-2.0*C8*C35*C66*AE
      T31=T12*TAU3
      T32=T12*T25*TAU3+T12+TAU3+T12*TAU3*T26
      T33=T12*T25+T25*TAU3+1.0+T26*(T12*T25*TAU3+T12+TAU3)+T27*T12*TAU3
      T34=T25+T26*(T12*T25+T25*TAU3+1.0)+T27*(T12*T25*TAU3+ ...
        T12+TAU3)+T12*T28*TAU3
      T35=T25*T26+T27*(T12*T25+T25*TAU3+1.0)+T28*(T12*T25*TAU3+ ...
        T12+TAU3)+T29*T12*TAU3
      T36=T25*T27+T28*(T12*T25+T25*TAU3+1.0)+T29*(T12*T25*TAU3+T12+TAU3)
      T37=T25*T28+T29*(T12*T25+T25*TAU3+1.0)
      T38=T25*T29
      T39=-1.0*C66*AF*(C45-C42)*T12*TAU3
      T40=-1.0*C66*AF*(C45-C42)*(T10*T12*TAU3+T12*T25*TAU3+ ...
        T12+TAU3)
      T41=-1.0*C66*AF*(C45-C42)*(T10*(T12*T25*TAU3+T12+TAU3)+ ...
        T12*T25+T25*TAU3+1.0)
      T42=-1.0*C66*AF*(C45-C42)*(T25+T10*(T12*T25+T25*TAU3+1.0))
```

```
      T43=-1.0*C66*AF*(C45-C42)*T10*T25
      T44=-1.0*C66*C23*T15
      T45=-1.0*(T15*T30+T16*C66*C23)
      T46=-1.0*(T16*T30+T17*C66*C23)
      T47=-1.0*(T17*T30+T18*C66*C23)
      T48=-1.0*(T18*T30+T19*C66*C23)
      T49=-1.0*T19*T30
      T50=-1.0*C66*C23*T20
      T51=-1.0*(T20*T30+T21*C66*C23)
      T52=-1.0*(T21*T30+T22*C66*C23)
      T53=-1.0*(T22*T30+T23*C66*C23)
      T54=-1.0*(T23*T30+T24*C66*C23)
      T55=-1.0*T24*T30
      T56=T33+T50
      T57=T34+T51
      T58=T35+T52
      T59=T36+T53
      T60=T37+T54
      T61=T38+T55
      T62=T39+T44
      T63=T40+T45
      T64=T41+T46
      T65=T42+T47
      T66=T43+T48
      T67=T2*T31
      T68=T2*T32+T3*T31
      T69=T2*T56+T3*T32+T4*T31
      T70=T2*T57+T3*T56+T4*T32
      T71=T2*T58+T3*T57+T4*T56
      T72=T2*T59+T3*T58+T4*T57
      T73=T2*T60+T3*T59+T4*T58
      T74=T2*T61+T3*T60+T4*T59
      T75=T3*T61+T4*T60
      T76=T4*T61
      T77=T62+T72
      T78=T63+T73
      T79=T64+T74
      T80=T65+T75
      T81=T66+T76
      T82=C3*T12*TAU3*C66*C23
      T83=C3*(T30*T12*TAU3+C66*C23*(T1*T12*TAU3+T10*T12*TAU3+...
        T12+TAU3))
      T84=C3*(T30*(T1*T12*TAU3+T12+T10*T12*TAU3+TAU3)+C66*C23* ...
        (T10*TAU3+T1*T12+T1*T10*T12*TAU3+T10*T12+T1*TAU3+1.0))
      T85=C3*(T30*(T10*TAU3+T1*T12+T1*T10*T12*TAU3+T10*T12+T1*TAU3 ...
        +1.0)+C66*C23*(T1*T10*T12+T1+T1*T10*TAU3+T10))
      T86=C3*(T30*(T1*T10*T12+T1+T1*T10*TAU3+T10)+C66*C23*T1*T10)
      T87=C3*T30*T1*T10
      T88=T82/T87
      T89=T83/T87
      T90=T84/T87
      T91=T85/T87
      T92=T86/T87
      T93=T67/T49
      T94=T68/T49
      T95=T69/T49
      T96=T70/T49
      T97=T71/T49
      T98=T77/T49
      T99=T78/T49
      T100=T79/T49
      T101=T80/T49
      T102=T81/T49
DYNAMIC
      W1=INTGRL(0.0,W2)
      W2=INTGRL(0.0,W3)
```

```
      W3=INTGRL(0.0,W4)
      W4=INTGRL(0.0,W5)
      W5=INTGRL(0.0,W6)
      W6=INTGRL(0.0,W7)
      W7=INTGRL(0.0,W8)
      W8=INTGRL(0.0,W9)
      W9=INTGRL(0.0,W10)
      W10=INTGRL(0.0,W10DOT)
      W10DOT=(I-T94*W10-T95*W9-T96*W8-T97*W7-T98*W6-T99*W5-T100*W4 ...
        -T101*W3-T102*W2-W1)/T93
      THET=(T87/T49)*(T88*W6+T89*W5+T90*W4+T91*W3+T92*W2+W1)
      RESP=KFBK*THET/I
      NOSORT
      CALL DEBUG (1,0.0)
*    THESE STATEMENTS WRITE TO TSO DATA SET FOR PLOTTING
      IF (KEEP.NE.1)  GO TO 500
      CYCLE=CYCLE+1.0
      IF (CYCLE.NE.20.0)  GO TO 500
      WRITE (8,600)  TIME,THET
  600 FORMAT (T5,2(E14.6,5X))
      CYCLE=0.0
      COUNT=COUNT+1.0
  500 CONTINUE
TERMINAL
      WRITE (8,700)  FLAG,COUNT
      TIMER  FINTIM=0.65,DELT=5.0E-05,PRDEL=2.5E-03,OUTDEL=2.5E-03
  700 FORMAT (T5,E14.6,5X,E14.6)
      METHOD  RKSFX
      PRTPLT  THET  (RESP)
END
STOP
```

## APPENDIX D

## TRANSFER FUNCTION CONSTANTS

Solution models for systems of the type studied are third order for uncompensated systems or systems utilizing pressure feedback. Dynamic pressure feedback system solution models are fourth order. The constants of the transfer function are terms including the various system parameters. Transfer functions for the general systems were derived for this study. Specific transfer functions for specific systems can be evaluated knowing the system parameters.

Transfer functions of the third order solution models used are of the form

$$\frac{\theta}{I} = \frac{K}{a_3 s^3 + a_2 s^2 + a_1 s + a_0} \qquad \text{(D.1)}$$

Solution models of the fourth order dynamic pressure feedback model are of the form

$$\frac{\theta}{I} = \frac{K \cdot (\tau s + 1)}{a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0} \qquad \text{(D.2)}$$

The definition of each constant of (D.1) for the system with no additional compensation is described below. Subscripted "C's" represent constants from the linearized equations described in Appendix A.

$$K = (1 - C_F) \cdot D_M \cdot K_V \cdot K_A \cdot C_{69} \cdot (C_{47} - C_{44}), \qquad \text{(D.3)}$$

$K_V$ is the valve gain.

$$a_3 = J \tag{D.4}$$

$$a_2 = J \cdot [C_{43} - C_{46} + C_{70} \cdot (C_{44} - C_{47})] + (C_{DM} \cdot D_M \cdot \mu + Visd) \tag{D.5}$$

$$a_1 = [C_{43} - C_{46} + C_{70} \cdot (C_{44} - C_{47})] \cdot (C_{DM} \cdot D_M \cdot \mu + Visd) - (1 - C_F) \cdot D_M \cdot (C_{45} - C_{42}) \tag{D.6}$$

$$a_0 = K_F \cdot K_V \cdot K_A \cdot (1 - C_F) \cdot D_M \cdot C_{69} \cdot (C_{47} - C_{44}) \tag{D.7}$$

Systems utilizing pressure feedback also have a transfer function of the form (D.1). The definition of each transfer function constant for this system follows.

$$K = (1 - C_F) \cdot D_M \cdot K_V \cdot K_A \cdot C_{69} \cdot (C_{47} - C_{44}) \tag{D.8}$$

$$a_3 = J \tag{D.9}$$

$$a_2 = J \cdot [C_{43} - C_{46} + C_{70} \cdot (C_{44} - C_{47})] + (C_{DM} \cdot D_M \cdot \mu + Visd) + J \cdot K_P \cdot K_A \cdot K_V \cdot C_{69} \cdot (C_{47} - C_{44}) \tag{D.10}$$

$$a_1 = [C_{43} - C_{46} + C_{70} \cdot (C_{44} - C_{47})] \cdot (C_{DM} \cdot D_M \cdot \mu + Visd) - (1 - C_F) \cdot D_M \cdot (C_{45} - C_{42}) + K_P \cdot (C_{DM} \cdot D_M \cdot \mu + Visd) \cdot K_A \cdot K_V \cdot C_{69} \cdot (C_{47} - C_{44}) \tag{D.11}$$

$$a_0 = K_F \cdot K_V \cdot K_A \cdot (1 - C_F) \cdot D_M \cdot C_{69} \cdot (C_{47} - C_{44}) \tag{D.12}$$

Systems utilizing pressure feedback have a transfer function of the form (D.2). The definition of each constant of the fourth order transfer function utilizing dynamic pressure feedback follows.

$$K = K_A \cdot K_V \cdot C_{69} \cdot (1 - C_F) \cdot D_M \cdot (C_{47} - C_{44}) \tag{D.13}$$

$$a_4 = J \cdot \tau \tag{D.14}$$

$$a_3 = \tau \cdot (J \cdot [C_{43} - C_{46} + C_{70} \cdot (C_{44} - C_{47})] + [C_{DM} \cdot D_M \cdot \mu + Visd])$$
$$+ J + K_A \cdot K_V \cdot K_D \cdot C_{69} \cdot (C_{47} - C_{44}) \cdot \tau \cdot J \quad (D.15)$$

$$a_2 = \tau \cdot ([C_{43} - C_{46} + C_{70} \cdot (C_{44} - C_{47})] \cdot [C_{DM} \cdot D_M \cdot \mu + Visd]$$
$$- [1 - C_F] \cdot D_M \cdot [C_{45} - C_{42}]) + J \cdot [C_{43} - C_{46} + C_{70} \cdot (C_{44} - C_{47})]$$
$$+ (C_{DM} \cdot D_M \cdot \mu + Visd)$$
$$+ K_A \cdot K_V \cdot K_D \cdot C_{69} \cdot (C_{47} - C_{44}) \cdot \tau \cdot (C_{DM} \cdot D_M \cdot \mu + Visd) \quad (D.16)$$

$$a_1 = [C_{43} - C_{46} + C_{70} \cdot (C_{44} - C_{47})] \cdot [C_{DM} \cdot D_M \cdot \mu + Visd]$$
$$- (1 - C_F) \cdot D_M \cdot (C_{45} - C_{42})$$
$$+ \tau \cdot K_F \cdot K_A \cdot K_V \cdot (1 - C_F) \cdot D_M \cdot C_{69} \cdot (C_{47} - C_{44}) \quad (D.17)$$

$$a_0 = K_F \cdot K_A \cdot K_V \cdot C_{69} \cdot (1 - C_F) \cdot D_M \cdot (C_{47} - C_{44}) \quad (D.18)$$

When developing the root locus as a function of feedback gain the transfer function in (D.2) was rearranged with the feedback gain in the open loop gain position. The new open loop transfer function for the feedback gain is of the following form.

$$GH = \frac{K_D \cdot (b_3 s^3 + b_2 s^2)}{a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0} \quad (D.19)$$

The definition of each constant of the transfer function follows.

$$b_3 = K_A \cdot K_V \cdot C_{69} \cdot (C_{47} - C_{44}) \cdot \tau \cdot J \quad (D.20)$$

$$b_2 = K_A \cdot K_V \cdot C_{69} \cdot (C_{47} - C_{44}) \cdot \tau \cdot (C_{DM} \cdot D_M \cdot \mu + Visd) \quad (D.21)$$

$$a_4 = J \cdot \tau \quad (D.22)$$

$$a_3 = J + \tau \cdot (J[C_{43} - C_{46} + C_{70} \cdot (C_{44} - C_{47})]$$
$$+ [C_{DM} \cdot D_M \cdot \mu + Visd]) \quad (D.23)$$

$$a_2 = J \cdot [C_{43} - C_{46} + C_{70} \cdot (C_{44} - C_{47})] + (C_{DM} \cdot D_M \cdot \mu + Visd)$$

$$+ \tau \cdot ([C_{43} - C_{46} + C_{70} \cdot (C_{44} - C_{47})] \cdot [C_{DM} \cdot D_M \cdot \mu + Visd]$$

$$- [1 - C_F] \cdot D_M \cdot [C_{45} - C_{42}]) \qquad (D.24)$$

$$a_1 = [C_{43} - C_{46} + C_{70} \cdot (C_{44} - C_{47})] \cdot [C_{DM} \cdot D_M \cdot \mu + Visd]$$

$$- [1 - C_F] \cdot D_M \cdot [C_{45} - C_{42}]$$

$$+ \tau \cdot K_F \cdot K_A \cdot K_V \cdot C_{69} \cdot (1 - C_F) \cdot D_M \cdot (C_{47} - C_{44}) \qquad (D.25)$$

$$a_0 = K_F \cdot K_A \cdot K_V \cdot C_{69} \cdot (1 - C_F) \cdot D_M \cdot (C_{47} - C_{44}) \qquad (D.26)$$

# APPENDIX E

## PERFORMANCE OPTIMIZATION PROGRAM

A listing of the user supplied program used to determine the optimum values of the three adjustable system parameters is presented in this appendix. The program and all related subroutines are written in WATFIV FORTRAN. Computational variables are double precision to minimize round-off error in the optimization process.

A main program and five subroutines comprise the user supplied program for evaluating the objective function for minimization. SERVO, MULER, DISP, ENVEL, and INTPOL are the five subroutines. The main program sets up the input for optimization.

Desired system static stiffness is user entered. System parameters are input into the program through DATA statements. The main program calculates the necessary servoamplifier gain required to maintain static stiffness. That amplifier gain value is assigned to XMIN(1) to prevent the gain from falling below the level required to maintain static stiffness. Simultaneously that amplifier gain is assigned to the program adjustable parameter X(1).

X(2) and X(3) represent the initial values of feedback gain and feedback time constant supplied by the user to the program. These values are not necessary for simulation. Setting MASK(1) and MASK(2) non-zero prevents the values of X(1) and X(2) from varying the second step of the synthesis process.

The system model to be used in the optimization program is determined by the variable SET. SET equal to 1.0 models the uncompensated system, equal to 3.0 is the pressure feedback model, and equal to 5.0 the dynamic pressure feedback model. Depending upon the particular model different initial values of the system parameters are supplied.

The main program calls STEPIT which is in constant communication with SERVO. SERVO evaluates the objective function supplied for minimization. Settling time was the only value used in computing the objective function for this project but SERVO also computes rise time, peak overshoot, and time of peak overshoot. These values can be incorporated into the objective function as desired by the user.

STEPIT receives the fixed system parameters from the main program and increments the adjustable parameters in such a manner as to minimize the objective function. SERVO evaluates the system transfer function with parameter values supplied by STEPIT. SERVO in turn calls MULER to locate the transfer function poles.

MULER locates the transfer function poles using Muller's method. Determination of all the system poles is not guaranteed with Muller's method but as a general rule is quite reliable.

Nature of the poles was known through the root locus presented in Chapter III. The time solution was obtained according to the procedure outlined in Chapter III. SERVO calculates the rise time, peak displacement, and peak time utilizing the subroutines DISP and INTPOL. DISP evaluates the value of the time response for various time values. INTPOL in turn interpolates between values returned by DISP to determine the time at a specific displacement.

Simultaneously, SERVO evaluates the response settling time using the envelope of the time response. Subroutines ENVEL and INTPOL are utilized by SERVO in obtaining the settling time. ENVEL evaluates the value of the response envelope for various time values. INTPOL in turn interpolates between values returned by ENVEL to determine the specific time settling time occurs.

STEPIT keeps track of the direction with which the objective function is changing. It increments the adjustable parameters and the process of re-evaluating the objective function begins again. Final parameter values which STEPIT returns are the starting points for system hardware selection.

```
$JOB            ,TIME=(1,00)
      DOUBLE PRECISION  XMAX,XMIN,DELTX,DELMN,ERR,FOBJ,X
      DOUBLE PRECISION  TRISE,TSET,DRISE,DSET,DPEAK
      DOUBLE PRECISION  PFE,PFF,PFG,PFL,AA,BB,CC,FREQ,PHI,TIME,THET,SGN
      DOUBLE PRECISION  COE(20),ROOTR(20),ROOTI(20),XN(10),FN(10),DISL
      DOUBLE PRECISION  ABSERR,ANS,ERROR
      DOUBLE PRECISION  STIFF,CON1,CON2,CON3,CON4,CON10,P2IC,P4IC,AE,
     1 C1,C2,C4,C5,C6,C8,C9,C16,C22,C23,C24,C30,C35,C42,C44,C45,C47,
     2C69,C43,C46,C70,T1,T10,KWFL,KVAL,II,KFBK,XO,YO,CS,CF,CDM,DM,
     3MU,P1,PE,RHO,CV,ANG,PJ,BETA,V1,V2,CD,D1,D2,D3,K1,K2,K3,PI,VISD
      EXTERNAL  SERVO
      COMMON  TRISE,TSET,DRISE,DSET
      COMMON  PFE,PFF,PFG,PFL,AA,BB,CC,FREQ,PHI,TIME,THET,SGN
      COMMON  COE,ROOTR,ROOTI
      COMMON  XN,FN,DISL,ABSERR,ANS,ERROR
      COMMON  N1,NFAIL,IQUIT,NN,MAXDEG,JJ
      COMMON /PASS/PI,C1,C42,C43,C44,C45,C46,C47,C69,C70,T1,KVAL,CF,
     1DM,PJ,CDM,MU,II,DPEAK,VISD,SET
      COMMON /CSTEP/ X(20),XMAX(20),XMIN(20),DELTX(20),DELMN(20),
     *        ERR(20,21),FOBJ,NV,NTRAC,MATRX,MASK(20),
     *        NFMAX,NFLAT,JVARY,NXTRA,KFLAG,NOREP,KERFLW,KW
C  SYSTEM PARAMETERS ARE ENTERED INTO THE OPTIMIZATION PROGRAM
C  THROUGH THESE DATA STATEMENTS.
      DATA  II,KFBK,XO,YO/1.50D+00,2.000D+00,1.8D-03,0.4D+00/
      DATA  CS,CF,CDM,DM,MU/0.88D-08,1.0D-01,1.6D+05,1.512D-02,2.0D-06/
      DATA  P1,PE,RHO,CV,ANG/1.1D+03,0.D+00,7.85D-05,9.8D-01,3.584D-01/
      DATA  PJ,BETA,V1,V2,CD/2.16D-03,1.50D+05,2.5D+01,2.5D+01,6.25D-01/
      DATA  D1,D2,D3,K1,K2,K3/2.3D-02,2.75D-01,1.2D-02,5.0D-02,1.4D+02,
     12.25D+01/
      DATA  PI,ABSERR/3.141592654D+00,5.0D-10/
      DATA  VISD/0.018D+00/
      DATA  IQUIT,NN,MAXDEG/0,10,8/
C  DRISE IS DESIRED RISE TIME.
      DRISE=2.69217D-02
C  DSET IS DESIRED SETTLING TIME.
      DSET=0.625
      DPEAK=1.0384
C  STIFF IS THE MINIMUM STATIC STIFFNESS DESIRED.
      STIFF=1000.0
      CON1=CD*PI*D2*DSQRT(2.0/RHO)
      CON2=0.25*PI*D3**2*CD*DSQRT(2.0/RHO)
      CON3=CD*PI*D1*DSQRT(2.0/RHO)
      CON4=-2.0*CV*CD*PI*D2*ANG
      CON10=0.25*PI*D1**2
      P2IC=0.5*(P1+PE)
      P4IC=(P1*D3**4+16.0*PE*D1**2*XO**2)/(D3**4+16.0*D1**2*XO**2)
      AE=0.25*PI*D2**2
      C1=-1.0*KFBK
      C2=CON10
      C4=K3/(K2+K3)
      C5=1.0/(K2+K3)
      C6=-0.5*CON2/(DSQRT(P1-P4IC))
      C8=CON3*DSQRT(P4IC-PE)
      C9=0.5*CON3*XO/(DSQRT(P4IC-PE))
      C16=CON4*(P1-P2IC)
      C22=CS*DM/MU
      C23=K3
      C24=-1.0*CON1*DSQRT(P1-P2IC)
      C30=1.0/((1.0-CF)*DM)
      C35=BETA/(YO*AE)
```

```
      C42=BETA*DM/V1
      C43=C22*BETA/V1
      C44=-1.0*BETA/V1
      C45=-1.0*BETA*DM/V2
      C46=-1.0*C22*BETA/V2
      C47=BETA/V2
      C69=-9.636D+02
      C70=-3.9161E-03
      T1=C43-C46+C70*(C44-C47)
      T10=C9*C35-C6*C35
      KWFL=(-1.0*STIFF*(K2+K3)/(K1*C1))*((2.0*C2*C5*C8-C6+C9)/
     1(C6*C23-C9*C23+2.0*C8*AE)*((2.0*C16-C23)*C22*C30/C24)
     2-C4*C22*C30/C24)
      KVAL=(K1/(K2+K3))*(T10*C23-2.0*C8*C35*AE)/(C4*(2.0*C8*C35*AE-
     1T10*C23)+(C23-2.0*C16)*(T10+2.0*C2*C5*C8*C35))
      CALL STSET
      NTRAC=0
C  SETTING MASK NONZERO PROHIBITS ANY CORRESPONDING CHANGE IN X.
C     MASK(1)=1
C     MASK(2)=1
C  SET DETERMINES WHICH MODEL IS OF INTEREST.
C    SET=1.0  UNCOMPENSATED SERVOSYSTEM
C    SET=3.0  PRESSURE FEEDBACK SERVOSYSTEM.
C    SET=5.0  DYNAMIC PRESSURE FEEDBACK SERVOSYSTEM.
      SET=3.0
C  XMIN(1)  MINIMUM VALUE OF AMPLIFIER GAIN.
      XMIN(1)=KWFL
C  XMIN(2)  MINIMUM VALUE OF FEEDBACK GAIN
      XMIN(2)=0.00
C  XMIN(3)  MINIMUM VALUE OF FEEDBACK TIME CONSTANT
      XMIN(3)=1.0D-07
      JJ=0
      IF (SET.NE.3.0) GO TO 25
      N1=3
      NV=2
      X(1)=KWFL
      X(2)=2.5D-02
      X(3)=0.0
      GO TO 45
   25 IF (SET.NE.5.0) GO TO 35
      N1=4
      NV=3
      X(1)=KWFL
      X(2)=1.786D-02
      X(3)=0.5
      GO TO 45
   35 N1=3
      NV=1
      X(1)=0.29
      X(2)=0.0
      X(3)=0.0
   45 CONTINUE
      CALL STEPIT(SERVO)
      WRITE (6,55)  STIFF
   55 FORMAT('1',T10,'THE MINIMUM REQUIRED STATIC STIFFNESS (IN*LBF/RAD)
     1 IS',T70,E12.5)
      WRITE (6,65) KWFL
   65 FORMAT(/,T10,'THE REQUIRED AMPLIFIER GAIN (MA/VOLT) IS',T70,E12.5)
      ST=((2.0*C16-C23)*C22*C30)/C24*(2.0*C2*C5*C8-C6+C9)/
     1(C6*C23-C9*C23+2.0*C8*AE)-C4*C22*C30/C24
```

```
      STIF=-1.0*K1*C1*X(1)/(ST*(K2+K3))
      WRITE (6,75)  STIF
   75 FORMAT(/,T10,'THE STATIC STIFFNESS ACTUALLY OBTAINED (IN*LBF/RAD)
     1IS',T70,E12.5)
      WRITE (6,85) X(1)
   85 FORMAT(/,T10,'FINAL VALUE FOR AMPLIFIER GAIN (MA/VOLT) IS',T70,
     1E12.5)
      IF (SET.EQ.1.0)  GO TO 115
      WRITE (6,95)  X(2)
   95 FORMAT (/,T10,'FINAL VALUE FOR FEEDBACK GAIN (VOLT/PSI) IS',
     1T70,E12.5)
      IF (SET.EQ.3.0)  GO TO 115
      WRITE (6,105)  X(3)
  105 FORMAT(/,T10,'FINAL VALUE OF FEEDBACK TIME CONSTANT (1/SEC) IS',
     1T70,E12.5)
  115 CONTINUE
      JJ=1
      CALL SERVO
      JJ=2
      X(1)=0.0
      CALL SERVO
      WRITE (6,195)
  195 FORMAT ('1')
      STOP
      END

      SUBROUTINE  SERVO
C  THE SUBROUTINE SERVO CALCULATES THE SOLUTION OF THE SYSTEM IN THE
C  TIME DOMAIN.  FROM THIS SOLUTION THE RISE TIME AND SETTLING TIME
C  ARE CALCULATED.
      DOUBLE PRECISION  XMAX,XMIN,DELTX,DELMN,ERR,FOBJ,X
      DOUBLE PRECISION  MU,II,CF,CDM,DM,PI,PJ,ABSERR,KVAL,C1,C42,C43,
     1C44,C45,C46,C47,C69,C70,T1,C32,VISD
      DOUBLE PRECISION      R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11
      DOUBLE PRECISION  TRISE,TSET,DRISE,DSET,DPEAK,PEAK
      DOUBLE PRECISION  PFE,PFF,PFG,PFL,AA,BB,CC,FREQ,PHI,TIME,THET,SGN
      DOUBLE PRECISION  COE(20),ROOTR(20),ROOTI(20),XN(10),FN(10),DISL
      DOUBLE PRECISION  ANS,ERROR,DC,PFH,PFJ,PFK,TIML,TIMP,TRL,TRU
      COMMON  TRISE,TSET,DRISE,DSET
      COMMON  PFE,PFF,PFG,PFL,AA,BB,CC,FREQ,PHI,TIME,THET,SGN
      COMMON  COE,ROOTR,ROOTI
      COMMON  XN,FN,DISL,ABSERR,ANS,ERROR
      COMMON  N1,NFAIL,IQUIT,NN,MAXDEG,JJ
      COMMON /PASS/PI,C1,C42,C43,C44,C45,C46,C47,C69,C70,T1,KVAL,CF,
     1DM,PJ,CDM,MU,II,DPEAK,VISD,SET
      COMMON /CSTEP/ X(20),XMAX(20),XMIN(20),DELTX(20),DELMN(20),
     *      ERR(20,21),FOBJ,NV,NTRAC,MATRX,MASK(20),
     *      NFMAX,NFLAT,JVARY,NXTRA,KFLAG,NOREP,KERFLW,KW
C  STATEMENTS C1 THROUGH R8 CALCULATES CONSTANTS DESCRIBING THE SYSTEM
C  AND ARE USED TO DETERMINE THE COEFFICIENTS OF THE DIFFERENTIAL
C  EQUATION.
      C32=-1.0*X(2)
      R1=(1.-CF)*DM
      R2=PJ*T1+(CDM*DM*MU+VISD)
      R3=T1*(CDM*DM*MU+VISD)-(1.0-CF)*DM*(C45-C42)
      R4=X(1)*KVAL*R1*(C47-C44)*C69
      R5=X(3)*R2+PJ-X(1)*KVAL*C69*(C47-C44)*C32*X(3)*PJ
      R6=X(3)*R3+R2-X(1)*KVAL*C69*(C47-C44)*C32*X(3)*(CDM*DM*MU+VISD)
      R7=R3-C1*R4*X(3)
      R8=-1.0*C1*R4
```

```
      R9=R1*R2+R4*X(2)*PJ
      R10=R1*R3+R4*X(2)*(CDM*DM*MU+VISD)
      R11=-1.0*R1*R4*C1
C  COE(1) THROUGH COE(5) ARE THE COEFFICIENTS OF THE DIFFERENTIAL
C  EQUATION DESCRIBING THE SYSTEM.
      IF (SET.NE.3.0)  GO TO 50
      COE(1)=PJ*R1
      COE(2)=R9
      COE(3)=R10
      COE(4)=R11
      GO TO 70
   50 IF (SET.NE.5.0)  GO TO 60
      COE(1)=X(3)*PJ
      COE(2)=X(3)*R2+PJ-X(1)*KVAL*C69*(C47-C44)*C32*X(3)*PJ
      COE(3)=X(3)*R3+R2-X(1)*KVAL*C69*(C47-C44)*C32*X(3)*(CDM*DM*MU+
     1VISD)
      COE(4)=R3-C1*X(3)*X(1)*KVAL*R1*(C47-C44)*C69
      COE(5)=-1.0*C1*X(1)*KVAL*R1*(C47-C44)*C69
      GO TO 70
   60 COE(1)=PJ
      COE(2)=R2
      COE(3)=R3
      COE(4)=-1.0*C1*R4
   70 CONTINUE
C  THE SUBROUTINE MULER SOLVES FOR THE ROOTS OF THE EQUATION.
      CALL MULER
      IF (NFAIL.EQ.0)  GO TO 140
      WRITE (6,130)
  130 FORMAT (//,T10,' THE SUBROUTINE MULER FAILDED TO FIND ALL ROOTS.')
      FOBJ=0.0
      GO TO 580
  140 IF (JJ.EQ.0)  GO TO 165
      IF (JJ.EQ.1)  WRITE (6,150)
  150 FORMAT(/,T10,'THE CLOSED LOOP POLES ARE')
      IF (JJ.EQ.2) WRITE(6,151)
  151 FORMAT(/,T10,'THE OPEN LOOP POLES ARE')
      WRITE (6,160) (I,ROOTR(I),ROOTI(I),I=1,N1)
  160 FORMAT ( /,15X,'X(',I1,')= ',E15.5,'+(',E15.5,') I')
  165 CONTINUE
      IF (JJ.EQ.2) GO TO 580
      KK=0
      LL=0
C  THESE NEXT STATEMENTS CHECK WHETHER OR NOT THE SYSTEM IS STABLE
C  BY LOOKING AT THE SIGN ON THE REAL PART OF THE ROOTS OF THE
C  EQUATION.
      DO 170 I=1,N1
      IF (ROOTR(I).LE.0.0)  GO TO 170
      WRITE (6,300)
  300 FORMAT (//,T10,' THE SYSTEM IS UNSTABLE.')
      FOBJ=0.0
      WRITE (6,302)  X(1),X(2),X(3)
  302 FORMAT (T10,'X(1)= ',D14.5,5X,'X(2)= ',D14.5,5X,'X(3)= ',D14.5)
      GO TO 580
  170 CONTINUE
C  THESE NEXT STATEMENTS CHECK WHETHER OR NOT THE SYSTEM IS TOO STABLE
C  BY ALL THE ROOTS BEING REAL AND NEGATIVE NOT HAVING COMPLEX
C  CONJUGATES.
      DO 310 I=1,N1
      IF (DABS(ROOTI(I)).GE.1.0D-05)  GO TO 420
  310 CONTINUE
```

```
      WRITE (6,320)
  320 FORMAT (//,T10,' ALL ROOTS ARE REAL.')
      FOBJ=0.0
      GO TO 580
C  FIND OUT WHICK ROOTS ARE REAL AND WHICH ONES ARE COMPLEX CONJUGATES
C  FOR PARTIAL FRACTION EXPANSION.
C  CALCULATE THE VALUES REQUIRED FOR THE PARTIAL FRACTION EXPANSION.
  420 DO 460 I=1,N1
      IF (DABS(ROOTI(I)).GE.1.0D-05)  GO TO 450
      KK=KK+1
      GO TO (430,440),KK
  430 AA=-1.0*ROOTR(I)
      GO TO 460
  440 BB=-1.0*ROOTR(I)
      GO TO 460
  450 LL=LL+1
      IF (LL.NE.1)  GO TO 460
      CC=-2.0*ROOTR(I)
      DC=ROOTR(I)**2+ROOTI(I)**2
  460 CONTINUE
      IF (N1.EQ.3)  BB=0.0
      IF (N1.EQ.3)  GO TO 475
C  PFE IS STEADY STATE VALUE.
      PFE=R4*II/(X(3)*AA*BB*DC*PJ)
C  PFG IS THE COEFFICIENT OF ONE OF THE REAL POLES.
      PFG=((AA*CC-AA**2-DC)*(R4*II/PJ-(PFE*AA*BB*CC+PFE*AA*DC-PFE*AA
     1**2*BB))+(PFE*AA*CC-PFE*AA**2)*(CC*AA*BB-AA**2*BB-BB*DC))/((AA*
     2CC-AA**2-DC)*(AA*BB**2-AA**2*BB+AA*DC-BB*DC)-(AA*CC+BB**2-BB*CC-
     3AA**2)*(CC*AA*BB-AA**2*BB-BB*DC))
      PFH=(PFE*AA**2-PFE*AA*CC-PFG*(AA*CC+BB**2-BB*CC-AA**2))/
     1(AA*CC-AA**2-DC)
      PFJ=-1.0*PFE*AA+PFG*(BB-AA)+PFH*(CC-AA)
C  PFF IS THE COEFFICIENT OF ONE OF THE REAL POLES.
      PFF=-1.0*PFE-1.0*PFG-1.0*PFH
      GO TO 478
  475 PFG=0.0
      PFE=R4*II/(PJ*AA*DC)
      PFH=(PFE*AA**2-PFE*AA*CC)/(AA*CC-DC-AA**2)
      PFJ=PFH*(CC-AA)-PFE*AA
      PFF=-1.0*PFE-PFH
C  FREQ IS THE DAMPED NATURAL FREQUENCY OF THE COMPLEX CONJUGATES.
  478 FREQ=DSQRT(4.0*DC-CC**2)/2.0
      PFK=(PFJ-PFH*CC/2.0)/FREQ
C  PHI IS THE PHASE SHIFT IN THE SINE TERM IN THE COMPLEX CONJUGATES.
      PHI=DATAN(PFH/PFK)
C  PFL IS THE COEFFICIENT OF THE COMPLEX CONJUGATE POLES.
      PFL=PFK/(DCOS(PHI))
      IF (JJ.NE.1)  GO TO 105
      WRITE (6,15) PFE
   15 FORMAT(/,T10,'STEADY STATE DISPLACEMENT (RAD) IS',T70,E12.5)
      WRITE (6,25) PFF
   25 FORMAT(/,T10,'COEFFICIENT OF FIRST REAL POLE TERM IS',T70,E12.5)
      WRITE (6,35) AA
      IF (N1.EQ.3) GO TO 58
   35 FORMAT(/,T10,'EXPONENT OF THE FIRST REAL POLE IS',T70,E12.5)
      WRITE (6,45) PFG
   45 FORMAT(/,T10,'COEFFICIENT OF SECOND REAL POLE TERM IS',T70,E12.5)
      WRITE (6,55)  BB
   55 FORMAT(/,T10,'EXPONENT OF SECOND REAL POLE IS',T70,E12.5)
   58 CCC=0.5*CC
```

```
      WRITE(6,65) PFL
   65 FORMAT(/,T10,'COEFFICIENT OF SINUSOIDAL TERM IS',T70,E12.5)
      WRITE (6,75) CCC
   75 FORMAT(/,T10,'EXPONENT OF SINUSOIDAL TERM IS',T70,E12.5)
      WRITE (6,85) FREQ
   85 FORMAT(/,T10,'RESPONSE DAMPED NATURAL FREQUENCY (RAD/SEC) IS',
     1T70,E12.5)
      WRITE (6,95)  PHI
   95 FORMAT(/,T10,'PHASE SHIFT OF SYSTEM RESPONSE (RAD) IS',T70,E12.5)
  105 CONTINUE
      M=0
C  TIMP IS THE TIME WHERE THE FIRST MAXIMUM OF THE SINE TERM OCCURS.
      TIMP=(1.5*PI-PHI)/FREQ
      TIME=TIMP
      TIML=TIMP
      CALL DISP
      PEAK=THET
      IF (JJ.NE.1)  GO TO 145
      WRITE (6,115)  THET
  115 FORMAT (/,T10,'RESPONSE PEAK DISPLACEMENT (RAD) IS',T70,E12.5)
      WRITE (6,135) TIMP
  135 FORMAT(/,T10,'TIME (SECS) PEAK DISPLACEMENT OCCURS IS',T70,E12.5)
  145 CONTINUE
      IF (THET.LT.(0.9*PFE))  GO TO 497
      TIME=TIMP+0.02*PI/FREQ
  490 TIME=TIME-0.02*PI/FREQ
      CALL DISP
      IF (THET.GE.(0.9*PFE))  GO TO 490
      TIML=TIME
C  OBTAIN DATA POINTS TO INTERPOLATE FOR 90% VALUE IN RISE TIME.
      DO 495 K=1,10
      PK=K
      FN(K)=TIML+(PK-1.0)*(0.02*PI/FREQ)/9.0
      TIME=FN(K)
      CALL DISP
      XN(K)=THET
  495 CONTINUE
      DISL=0.9*PFE
C  INTERPOLATE TO THE FIND THE 90% POINT.
      CALL INTPOL
      TRU=ANS
      M=M+1
  497 TIME=TIML
  500 TIME=TIME-0.02*PI/FREQ
      CALL DISP
      IF (THET.GE.(0.1*PFE))  GO TO 500
      TIML=TIME
C  OBTAIN DATA POINTS TO INTERPOLATE FOR 10% VALUE IN RISE TIME.
      DO 505 K=1,10
      PK=K
      FN(K)=TIML+(PK-1.0)*(0.02*PI/FREQ)/9.0
      TIME=FN(K)
      CALL DISP
      XN(K)=THET
  505 CONTINUE
      DISL=0.1*PFE
C  INTERPOLATE TO FIND THE 10% POINT.
      CALL INTPOL
      TRL=ANS
C  IN THE EVENT THAT THE SOLUTION HAS A VERY DOMINANT REAL POLE THE
```

```
C  FIRST SINE PEAK MAY NOT HAVE REACHED THE 90% POINT.  IF THIS IS
C  TRUE FIND WHERE THE 90% POINT IS LOCATED.
      IF (M.NE.0)  GO TO 520
      TIME=TIMP
  510 TIME=TIME+2.0*PI/FREQ
      CALL DISP
      IF (TIME.LT.(100.*PI/FREQ))  GO TO 525
      WRITE (6,535)
  535 FORMAT(//,T10,' THE RISE TIME IS TOO SLOW.')
      FOBJ=0.0
      GO TO 580
  525 IF (THET.LT.(0.9*PFE)) GO TO 510
  527 TIME=TIME-0.02*PI/FREQ
      CALL DISP
      IF (THET.GE.(0.9*PFE)) GO TO 527
      TIML=TIME
C  OBTAIN DATA POINTS TO INTERPOLATE FOR 90% VLAUE IN RISE TIME.
      DO 515 K=1,10
      PK=K
      FN(K)=TIML+(PK-1.0)*(0.02*PI/FREQ)/9.0
      TIME=FN(K)
      CALL DISP
      XN(K)=THET
  515 CONTINUE
      DISL=0.9*PFE
C  INTERPOLATE TO FIND THE 90% POINT.
      CALL INTPOL
      TRU=ANS
C  CALCULATE THE RISE TIME OF THE SYSTEM.   IE, THE TIME TO GO FOR THE
C  FIRST TIME FROM THE 10% VALUE OF STEADY STATE TO THE 90% VALUE OF
C  STEADY STATE.
  520 TRISE=TRU-TRL
      IF (JJ.EQ.1)  WRITE(6,155) TRISE
  155 FORMAT(/,T10,'RESPONSE RISE TIME (SECS) IS',T70,E12.5)
C  THE NEXT THING NECESSARY TO CHECK IS THE RESPONSE SETTLING TIME.
C  THE TIME WHEN THE BOUNDS ON THE SOLUTION COMES WITHIN 5% OF THE
C  STEADY STATE VALUE IS THE POINT WANTED.
      TIME=0.02*PI/FREQ
      SGN=1.0
C  CHECK THE ENVELOPE OF THE SINE WAVE ON THE LOW SIDE OF STEADY STATE
C  VALUE.
  540 CALL ENVEL
      IF (THET.GT.(0.95*PFE)) GO TO 550
      TIME=TIME+0.02*PI/FREQ
      IF (TIME.GT.(100.*PI/FREQ)) GO TO 545
      GO TO 540
  545 WRITE (6,547)
  547 FORMAT (1X,'THE SETTLING TIME IS TOO LARGE.')
      FOBJ=0.0
      GO TO 580
  550 TIMU=TIME
      TIML=TIME-0.02*PI/FREQ
C  OBTAIN DATA POINTS FOR 95% ENVELOPE VALUE IN SETTLING TIME.
C  INTERPOLATE TO FIND 95% POINT.
      DO 560 K=1,10
      PK=K
      FN(K)=TIML+(TIMU-TIML)*(PK-1.0)/9.0
      TIME=FN(K)
      CALL ENVEL
      XN(K)=THET
```

```
  560 CONTINUE
      DISL=0.95*PFE
      CALL INTPOL
      TSET=ANS
C  CHECK THE ENVELOPE OF THE SINE WAVE ON THE HIGH SIDE OF STEADY
C  STATE VALUE.
      SGN=-1.0
      TIME=TSET
      CALL ENVEL
C  IF THE 105% POINT DOES NOT MEET THE SETTLING TIME REQUIREMENTS
C  PROCEED AS BEFORE TO FIND SETTLING TIME USING THE UPPER BOUND.
      IF (THET.LT.(1.05*PFE)) GO TO 610
  565 CALL ENVEL
      IF (THET.LT.(1.05*PFE)) GO TO 590
      TIME=TIME+0.02*PI/FREQ
      IF (TIME.GT.(100.*PI/FREQ)) GO TO 545
      GO TO 565
  590 TIMU=TIME
      TIML=TIME-0.02*PI/FREQ
      DO 600 K=1,10
      PK=K
      FN(K)=TIML+(TIMU-TIML)*(PK-1.0)/9.0
      TIME=FN(K)
      CALL ENVEL
      XN(K)=THET
  600 CONTINUE
      DISL=1.05*PFE
      CALL INTPOL
  610 TSET=ANS
      IF (JJ.NE.1)  GO TO 185
      WRITE (6,166) TSET
  166 FORMAT(/,T10,'RESPONSE SETTLING TIME (SECS) IS',T70,E12.5)
      WRITE (6,175) DISL
  175 FORMAT(/,T10,'RESPONSE ENVELOPE VALUE (RAD) AT WHICH SETTLING TIME
     1 OCCURS',T70,E12.5)
  185 CONTINUE
C  THE OBJECTIVE FUNCTION FOR MINIZATION
      FOBJ=TSET
C     FOBJ=DABS(TSET-DSET)
  580 RETURN
      END

      SUBROUTINE  MULER
C                                                                       MULER  2
C  MULER 3.0          A.N.S.I. STANDARD FORTRAN         JUNE 1974       MULER  3
C                                                                       MULER  4
C  FINDS ALL ROOTS OF A POLYNOMIAL HAVING REAL COEFFICIENTS, USING      MULER  5
C  MULLER-S METHOD.                                                     MULER  6
C                                                                       MULER  7
C  J. P. CHANDLER, COMPUTER SCIENCE DEPT., OKLAHOMA STATE UNIVERSITY    MULER  8
C  (THE ORIGINAL VERSION OF THIS ROUTINE WAS DISTRIBUTED BY -SHARE-.)   MULER  9
C                                                                       MULER 10
C  MULLER-S METHOD GIVES SLOW CONVERGENCE AND POOR ACCURACY ON          MULER 11
C  MULTIPLE OR CLUSTERED ROOTS.  SO DO MOST OTHER METHODS.              MULER 12
C  MULLER-S METHOD IS NOT INHERENTLY RESTRICTED TO REAL COEFFICIENTS.   MULER 13
C                                                                       MULER 14
C  P. HENRICI, -ELEMENTS OF NUMERICAL ANALYSIS- (WILEY)                 MULER 15
C                                                                       MULER 16
C  INPUT QUANTITIES ....                                                MULER 17
C     COE(J)     --  THE COEFFICIENT OF Z**(N1+1-J), J=1,...,N1+1       MULER 1
```

```
C     N1          --  THE ORDER OF THE POLYNOMIAL                      MULER 19
C     IQUIT       --  =1 IF MULER IS TO RETURN IMMEDIATELY UPON FINDING  MULER 20
C                        A ROOT THAT IS REAL AND POSITIVE,             MULER 21
C                     =0 IF MULER IS TO COMPUTE ALL ROOTS              MULER 22
C                                                                      MULER 23
C  OUTPUT QUANTITIES ....                                              MULER 24
C     ROOTR(K)  --  THE REAL PART OF THE K-TH ROOT, K=1,...,N1         MULER 25
C     ROOTI(K)  --  THE IMAGINARY PART OF THE K-TH ROOT                MULER 26
C     NFAIL     --  RETURNED NONZERO IF MULER FAILED TO FIND ALL ROOTS MULER 27
C                                                                      MULER 28
C  ALL FLOATING-POINT VARIABLES ARE DOUBLE PRECISION ....              MULER 29
C                                                                      MULER 30
      DOUBLE PRECISION  AXI,AXR,TEM,TE1,TE2,TE3,TE4,TE5,TE6,TE7,TE8
      DOUBLE PRECISION  TE9,TE10,TE11,TE12,TE13,TE14,TE15,TE16,BELL
      DOUBLE PRECISION  DE15,DE16,HELL,TEMI,TEMR,TEM1,ALP1I,ALP1R,ALP2I
      DOUBLE PRECISION  ALP2R,ALP3I,ALP3R,ALP4I,ALP4R,BET1I,BET1R
      DOUBLE PRECISION  BET2I,BET2R,BET3I,BET3R,ARG,QSQRT,DSQRT,RZERO
      DOUBLE PRECISION  RHALF,RUNIT,RTWO,RFOUR,RSMAL,EPS,CONSA,CONSB
      DOUBLE PRECISION  CONSC,CONSD,AA,BB,   ABT,TMAX,TEM2
      DOUBLE PRECISION  COE(20),ROOTR(20),ROOTI(20)
C                                                                      MULER 37
      DOUBLE PRECISION    TRISE,TSET,DRISE,DSET,PFE,PFF,PFG,PFL,AZ
      DOUBLE PRECISION  BZ,CZ,FREQ,PHI,TIME,THET,SGN,XN(10),FN(10)
      DOUBLE PRECISION  DISL,ABSERR,ANS,ERROR
      COMMON  TRISE,TSET,DRISE,DSET
      COMMON  PFE,PFF,PFG,PFL,AZ,BZ,CZ,FREQ,PHI,TIME,THET,SGN
      COMMON  COE,ROOTR,ROOTI
      COMMON  XN,FN,DISL,ABSERR,ANS,ERROR
      COMMON  N1,NFAIL,IQUIT,NN,MAXDEG,JJ
C                                                                      MULER 39
C     QSQRT(ARG)=SQRT(ARG)                                             MULER 40
      QSQRT(ARG)=DSQRT(ARG)                                            MULER 41
C                                                                      MULER 42
C                           SET ALL CONSTANTS.                         MULER 43
C                           KW ...  LOGICAL UNIT NUMBER OF THE PRINTER MULER 44
      KW=6                                                             MULER 45
C                           CONSA, ETC. ...  THREE STARTING POINTS     MULER 46
      CONSA=.8                                                         MULER 47
      CONSB=.85                                                        MULER 48
      CONSC=.9                                                         MULER 49
C                           RSMAL ...  RELATIVE CONVERGENCE TOLERANCE  MULER 50
C                                FOR THE MAGNITUDE OF THE POLYNOMIAL   MULER 51
C     RSMAL=1.E-20                                                     MULER 52
      RSMAL=1.E-10                                                     MULER 53
C                           EPS ...  RELATIVE CONVERGENCE TOLERANCE    MULER 54
C                                FOR THE STEP SIZE (EPS AND RSMAL MUST MULER 55
C                                BE .GT. THE RELATIVE ERROR OF THE     MULER 56
C                                FLOATING-POINT ARITHMETIC USED.)      MULER 57
      EPS=1.0E-10
C                           CONSD ...  RELATIVE TOLERANCE FOR CHECKING MULER 59
C                                WHETHER A ROOT IS REAL                MULER 60
      CONSD=1.E-5                                                      MULER 61
C                           ITMAX ...  MAX. NO. OF ITERATIONS PER ROOT MULER 62
      ITMAX=100                                                        MULER 63
      RZERO=0.                                                         MULER 64
      RHALF=.5                                                         MULER 65
      RUNIT=1.                                                         MULER 66
      RTWO=2.                                                          MULER 67
      RFOUR=4.                                                         MULER 68
C                           INITIALIZE.                                MULER 69
```

```
      NFAIL=0
      MC=10
      N2=N1+1
      N4=0
      I=N1+1
C                             REMOVE ANY ZERO ROOTS.
  100 IF(COE(I))120,110,120
  110 N4=N4+1
      ROOTR(N4)=RZERO
      ROOTI(N4)=RZERO
      I=I-1
      IF(N4-N1)100,630,100
C
C  COMPUTE THE FIRST THREE (POSITIVE REAL) ITERATES FOR THE NEXT ROOT.
C
  120 AXR=CONSA
      AXI=RZERO
      L=1
      N3=1
      ALP1R=AXR
      ALP1I=AXI
      M=1
      GO TO 680
  130 BET1R=TEMR
      BET1I=TEMI
      AXR=CONSB
      ALP2R=AXR
      ALP2I=AXI
      M=2
      GO TO 680
  140 BET2R=TEMR
      BET2I=TEMI
      AXR=CONSC
      ALP3R=AXR
      ALP3I=AXI
      M=3
      GO TO 680
C                             THE THREE ITERATES ARE COMPLETE.
  150 BET3R=TEMR
      BET3I=TEMI
C
C  BEGIN THE NEXT ITERATION OF MULLER-S METHOD.
C  COMPUTE (IN HENRICI-S NOTATION) H(N)=X(N)-X(N-1)=ALP(N)-ALP(N-1).
C
  160 TE1=ALP1R-ALP3R
      TE2=ALP1I-ALP3I
      TE5=ALP3R-ALP2R
      TE6=ALP3I-ALP2I
      TEM=TE5*TE5+TE6*TE6
      IF(TEM)170,180,170
  170 TE3=(TE1*TE5+TE2*TE6)/TEM
      TE4=(TE2*TE5-TE1*TE6)/TEM
      GO TO 190
  180 TE3=RZERO
      TE4=RZERO
  190 TE7=TE3+RUNIT
      TE9=TE3*TE3-TE4*TE4
      TE10=RTWO*TE3*TE4
      DE15=TE7*BET3R-TE4*BET3I
      DE16=TE7*BET3I+TE4*BET3R
```

```
      TE11=TE3*BET2R-TE4*BET2I+BET1R-DE15                               MULER130
      TE12=TE3*BET2I+TE4*BET2R+BET1I-DE16                               MULER131
      TE7=TE9-RUNIT                                                     MULER132
      TE1=TE9*BET2R-TE10*BET2I                                          MULER133
      TE2=TE9*BET2I+TE10*BET2R                                          MULER134
      TE13=TE1-BET1R-TE7*BET3R+TE10*BET3I                               MULER135
      TE14=TE2-BET1I-TE7*BET3I-TE10*BET3R                               MULER136
      TE15=DE15*TE3-DE16*TE4                                            MULER137
      TE16=DE15*TE4+DE16*TE3                                            MULER138
      TE1=TE13*TE13-TE14*TE14-RFOUR*(TE11*TE15-TE12*TE16)               MULER139
      TE2=RTWO*TE13*TE14-RFOUR*(TE12*TE15+TE11*TE16)                    MULER140
      TEM=QSQRT(TE1*TE1+TE2*TE2)                                        MULER141
C                                                                       MULER142
C                                   TEST THE SIGN....                   MULER143
      IF(TE1)200,200,240                                                MULER144
  200 TE4=QSQRT(RHALF*(TEM-TE1))                                        MULER145
      IF(TE4)230,210,230                                                MULER146
  210 WRITE(KW,220) TEM,TE1,TE4                                         MULER147
  220 FORMAT (//,T27,'ERROR IN SUBROUTINE MULLER',3E20.8)
      NFAIL=1                                                           MULER149
      MC=0                                                              MULER150
      TE3=RZERO                                                         MULER151
      GO TO 290                                                         MULER152
  230 TE3=RHALF*TE2/TE4                                                 MULER153
      GO TO 290                                                         MULER154
  240 TE3=QSQRT(RHALF*(TEM+TE1))                                        MULER155
C                                                                       MULER156
C                                   TEST THE SIGN....                   MULER157
      IF(TE2)250,260,260                                                MULER158
  250 TE3=-TE3                                                          MULER159
  260 IF(TE3)280,270,280                                                MULER160
  270 NFAIL=2                                                           MULER161
      GO TO 630                                                         MULER162
  280 TE4=RHALF*TE2/TE3                                                 MULER163
  290 TE7=TE13+TE3                                                      MULER164
      TE8=TE14+TE4                                                      MULER165
      TE9=TE13-TE3                                                      MULER166
      TE10=TE14-TE4                                                     MULER167
      TE1=RTWO*TE15                                                     MULER168
      TE2=RTWO*TE16                                                     MULER169
      IF(TE7*TE7+TE8*TE8-TE9*TE9-TE10*TE10)300,300,310                  MULER170
  300 TE7=TE9                                                           MULER171
      TE8=TE10                                                          MULER172
  310 TEM=TE7*TE7+TE8*TE8                                               MULER173
      IF(TEM)320,330,320                                                MULER174
  320 TE3=(TE1*TE7+TE2*TE8)/TEM                                         MULER175
      TE4=(TE2*TE7-TE1*TE8)/TEM                                         MULER176
      GO TO 340                                                         MULER177
  330 TE3=RZERO                                                         MULER178
      TE4=RZERO                                                         MULER179
  340 AXR=ALP3R+TE3*TE5-TE4*TE6                                         MULER180
      AXI=ALP3I+TE3*TE6+TE4*TE5                                         MULER181
      ALP4R=AXR                                                         MULER182
      ALP4I=AXI                                                         MULER183
C                                   EVALUATE THE POLYNOMIAL AT THE NEW POINT.  MULER184
      M=4                                                               MULER185
      GO TO 680                                                         MULER186
  350 N6=1                                                              MULER187
C                                                                       MULER188
C  END OF THIS ITERATION OF MULLER-S METHOD.                            MULER189
```

```
C  TEST THE VALUE OF THE POLYNOMIAL FOR CONVERGENCE.                     MULER190
C  THE TEST FORMERLY USED HERE WAS      IF(ABS(HELL)+ABS(BELL)-RSMAL) .  MULER191
C  THIS IS AN ABSOLUTE TEST, AND CAN FAIL IF THE COEFFICIENTS ARE NOT    MULER192
C  PROPERLY SCALED.  USE A RELATIVE TEST INSTEAD....                     MULER193
C                                                                        MULER194
C                             IF(AMAX1(ABS(HELL),ABS(BELL))-RSMAL*TMAX)  MULER195
      AA=HELL                                                            MULER196
      IF(AA)360,370,370                                                  MULER197
  360 AA=-AA                                                             MULER198
  370 BB=BELL                                                            MULER199
      IF(BB)380,390,390                                                  MULER200
  380 BB=-BB                                                             MULER201
  390 IF(BB-AA)410,410,400                                               MULER202
  400 AA=BB                                                              MULER203
  410 IF(AA-RSMAL*TMAX)530,530,420                                       MULER204
C                                                                        MULER205
C                             TEST THE CHANGE IN Z FOR CONVERGENCE.      MULER206
C                                  THIS TEST IS RELATIVE.                MULER207
C                             TE7=ABS(ALP3R-AXR)+ABS(ALP3I-AXI)          MULER208
  420 AA=ALP3R-AXR                                                       MULER209
      IF(AA)430,440,440                                                  MULER210
  430 AA=-AA                                                             MULER211
  440 BB=ALP3I-AXI                                                       MULER212
      IF(BB)450,460,460                                                  MULER213
  450 BB=-BB                                                             MULER214
  460 TE7=AA+BB                                                          MULER215
C                             IF(TE7/(ABS(AXR)+ABS(AXI))-EPS)            MULER216
      AA=AXR                                                             MULER217
      IF(AA)470,480,480                                                  MULER218
  470 AA=-AA                                                             MULER219
  480 BB=AXI                                                             MULER220
      IF(BB)490,500,500                                                  MULER221
  490 BB=-BB                                                             MULER222
  500 IF(TE7-EPS*(AA+BB))530,530,510                                     MULER223
C                                                                        MULER224
C                             NO CONVERGENCE.  SHIFT THE ITERATES.       MULER225
  510 N3=N3+1                                                            MULER226
      ALP1R=ALP2R                                                        MULER227
      ALP1I=ALP2I                                                        MULER228
      ALP2R=ALP3R                                                        MULER229
      ALP2I=ALP3I                                                        MULER230
      ALP3R=ALP4R                                                        MULER231
      ALP3I=ALP4I                                                        MULER232
      BET1R=BET2R                                                        MULER233
      BET1I=BET2I                                                        MULER234
      BET2R=BET3R                                                        MULER235
      BET2I=BET3I                                                        MULER236
      BET3R=TEMR                                                         MULER237
      BET3I=TEMI                                                         MULER238
      IF(N3-ITMAX)160,160,520                                            MULER239
  520 NFAIL=3                                                            MULER240
C                             STORE THE ROOT.                            MULER241
  530 N4=N4+1                                                            MULER242
      ROOTR(N4)=ALP4R                                                    MULER243
      ROOTI(N4)=ALP4I                                                    MULER244
      N3=0                                                               MULER245
C                             HAVE WE FOUND ALL ROOTS....                MULER246
      IF(N4-N1)540,630,630                                               MULER247
C                             NO.  WAS THIS ROOT REAL ....               MULER248
C                             IF(ABS(ROOTI(N4))-CONSD*ABS(ROOTR(N4)))    MULER249
```

```
546      540 AA=ROOTI(N4)                                                  MULER250
547          IF(AA)550,560,560                                             MULER251
548      550 AA=-AA                                                        MULER252
549      560 ABT=ROOTR(N4)                                                 MULER253
550          IF(ABT)570,580,580                                            MULER254
551      570 ABT=-ABT                                                      MULER255
       C                             CHECK FOR THE -IQUIT- OPTION.         MULER256
552      580 IF(IQUIT)600,590,600                                          MULER257
       C                                                                   MULER258
553      590 IF(AA-CONSD*ABT)120,120,640                                   MULER259
554      600 IF(AA-CONSD*ABT)610,610,640                                   MULER260
555      610 IF(ROOTR(N4))120,120,620                                      MULER261
556      620 N1=N4                                                         MULER262
       C                             THIS IS THE ONLY RETURN STATEMENT.    MULER263
557      630 RETURN                                                        MULER264
       C                             IF ONE NON-REAL ROOT OF A CONJUGATE PAIR  MULER265
       C                                  HAS BEEN FOUND,                  MULER266
558      640 GO TO(650,120),L                                              MULER267
       C                                  START AT ITS COMPLEX CONJUGATE.  MULER268
559      650 AXR=ALP1R                                                     MULER269
560          AXI=-ALP1I                                                    MULER270
561          ALP1I=-ALP1I                                                  MULER271
562          M=5                                                           MULER272
563          GO TO 680                                                     MULER273
564      660 BET1R=TEMR                                                    MULER274
565          BET1I=TEMI                                                    MULER275
       C                             GET TWO OTHER ITERATES ALSO.          MULER276
566          AXR=ALP2R                                                     MULER277
567          AXI=-ALP2I                                                    MULER278
568          ALP2I=-ALP2I                                                  MULER279
569          M=6                                                           MULER280
570          GO TO 680                                                     MULER281
571      670 BET2R=TEMR                                                    MULER282
572          BET2I=TEMI                                                    MULER283
573          AXR=ALP3R                                                     MULER284
574          AXI=-ALP3I                                                    MULER285
575          ALP3I=-ALP3I                                                  MULER286
576          L=2                                                           MULER287
577          M=3                                                           MULER288
       C                                                                   MULER289
       C  THIS SECTION COMPUTES THE VALUE (TEMR,TEMI) OF THE POLYNOMIAL AT  MULER290
       C  THE POINT (AXR,AXI) USING HORNER-S RULE (NESTED MULTIPLICATION). MULER291
       C                                                                   MULER292
578      680 TEMR=COE(1)                                                   MULER293
579          TEMI=RZERO                                                    MULER294
580          TMAX=TEMR                                                     MULER295
581          IF(TMAX)690,700,700                                           MULER296
582      690 TMAX=-TMAX                                                    MULER297
583      700 DO 710 I=1,N1                                                 MULER298
584            TE1=TEMR*AXR-TEMI*AXI                                       MULER299
585            TEMI=TEMI*AXR+TEMR*AXI                                      MULER300
586      710   TEMR=TE1+COE(I+1)                                           MULER301
       C                             COMPUTE TMAX, THE MAGNITUDE OF THE LARGEST MULER302
       C                                  TERM IN THE POLYNOMIAL.          MULER303
587          TE1=RUNIT                                                     MULER304
588          TE2=RZERO                                                     MULER305
589          TMAX=COE(N1+1)                                                MULER306
590          IF(TMAX)720,730,730                                           MULER307
591      720 TMAX=-TMAX                                                    MULER308
592      730 DO 810 I=1,N1                                                 MULER309
```

```
      C                              (TE1,TE2) = Z**I                           MULER310
                TE3=AXR*TE1-AXI*TE2                                             MULER311
                TE2=AXR*TE2+AXI*TE1                                             MULER312
                TE1=TE3                                                         MULER313
      C                             TMAX=AMAX1(TMAX,ABS(COE(J)*TE1),            MULER314
      C                                             ABS(COE(J)*TE2))            MULER315
                J=N1+1-I                                                        MULER316
                TE3=COE(J)*TE1                                                  MULER317
                IF(TE3)740,770,750                                              MULER318
         740    TE3=-TE3                                                        MULER319
         750    IF(TE3-TMAX)770,770,760                                         MULER320
         760    TMAX=TE3                                                        MULER321
         770    TE3=COE(J)*TE2                                                  MULER322
                IF(TE3)780,810,790                                              MULER323
         780    TE3=-TE3                                                        MULER324
         790    IF(TE3-TMAX)810,810,800                                         MULER325
         800    TMAX=TE3                                                        MULER326
         810    CONTINUE                                                        MULER327
              HELL=TEMR                                                         MULER328
              BELL=TEMI                                                         MULER329
              IF(N4)820,860,820                                                 MULER330
      C                             -DEFLATE- NUMERICALLY ...  DIVIDE THE       MULER331
      C                                 POLYNOMIAL BY                           MULER332
      C                                 (Z-ROOT(1))...(Z-ROOT(N4)).             MULER333
         820 DO 850 I=1,N4                                                      MULER334
                TEM1=AXR-ROOTR(I)                                               MULER335
                TEM2=AXI-ROOTI(I)                                               MULER336
                TE1=TEM1*TEM1+TEM2*TEM2                                         MULER337
                IF(TE1)840,830,840                                              MULER338
         830    NFAIL=-N4                                                       MULER339
                GO TO 630                                                       MULER340
         840    TE2=(TEMR*TEM1+TEMI*TEM2)/TE1                                   MULER341
                TEMI=(TEMI*TEM1-TEMR*TEM2)/TE1                                  MULER342
                TEMR=TE2                                                        MULER343
         850    CONTINUE                                                        MULER344
      C                             -RETURN- TO THE APPROPRIATE POINT.          MULER345
         860 GO TO(130,140,150,350,660,670),M                                   MULER346
      C                                                                         MULER347
      C  END MULER.                                                             MULER348
              RETURN
              END                                                               MULER349

              SUBROUTINE INTPOL
      C  THE SUBROUTINE INTPOL WAS MODIFIED FROM A PROGRAM IN 'NUMERICAL
      C  COMPUTING:  AN INTRODUCTION',  SHAMPINE & ALLEN, SAUNDERS PUBLISHING
      C  CO., PG. 227-229.
      C
      C  INTPOL CONSTRUCTS AN INTERPOLATING POLYNOMIAL USING DIVIDED
      C  DIFFERENCES.  THE USER CAN EITHER SPECIFY THE DEGREE TO BE USED OR
      C  A TOLERANCE AND A MAXIMUM DEGREE.  IN THE LATTER CASE THE CODE USES
      C  THE LOWEST DEGREE POLYNOMIAL WHICH IT BELIEVES MEETS THE TOLERANCE.
      C
      C     N - NUMBER OF NODES.  N MUST BE AT LEAST 2.  THE CODE DOES NOT
      C         TEST FOR THIS.
      C     XN - ARRAY OF NODES.  MUST BE DISTINCT.  THE CODE DOES NOT TEST
      C         FOR THIS.
      C     FN - ARRAY OF FUNCTION VALUES CORRESPONDING TO NODES XN.
      C     XX - POINT AT WHICH INTERPOLATING POLYNOMIAL IS TO BE EVALUATED.
      C     ANS - VALUE OF THE INTERPOLATING POLYNOMIAL AT XX.
      C     ERROR - ESTIMATED ERROR OF ANS.  THE VALUE ANS+ERROR IS OFTEN
```

```
C           A MORE ACCURATE RESULT BUT NOT ALWAYS.
C        ABSERR - THE CODE TRIES TO CHOOSE THE DEGREE OF THE INTERPOLATING
C           POLYNOMIAL SO THAT ABS(ERROR).LE.ABSERR. TO SPECIFY THE
C           DEGREE.  SET ABSERR NEGATIVE AND USE MAXDEG AS DESCRBED BELOW.
C        MAXDEG - UPPER BOUND ON THE DEGREE OF THE INTERPOLATING POLYNOMIAL
C           IF TOLERANCE IS MET THE DEGREE OF THE POLYNOMIAL IS RETURNED
C           IN MAXDEG.  OTHERWISE, MAXDEG REMAINS AS AN INPUT AND THE
C           ERROR  TOLERANCE MAY NOT HAVE BEEN MET.  IN THIS CASE THE
C           USER SHOULD CHECK THE OUTPUT QUANTITY ERROR.  IF A POLYNOMIAL
C           OF SPECIFIED DEGREE IS DESIRED.  SET MAXDEG TO THE DESIRED
C           DEGREE AND ABSERR TO ANY NEGATIVE VALUE.  MAXDEG MUST BE LESS
C           THAN OR EQUAL TO N-2 (FOR DEGREE K A TOTAL OF K+2 POINTS ARE
C           REQUIRED TO EVALUATE THE POLYNOMIAL AND TO ESTIMATE THE ERROR.
C           IF IT EXCEEDS THIS VALUE IT IS SET EQUAL TO N-2.  SINCE MAXDEG
C           IS USED FOR OUTPUT AS WELL AS INPUT IT MUST BE A VARIABLE IN
C           THE CALLING PROGRAM.
         DOUBLE PRECISION  AA,    ABSERR,ANS,BB,CC,ERROR,FN(10),FREQ,PFE
         DOUBLE PRECISION  PFF,PFG,PFL,PHI,PROD,THET,TIME,V(10,10)
         DOUBLE PRECISION  XN(10),XX,TRISE,TSET,DRISE,DSET,SGN
         DOUBLE PRECISION  COE(20),ROOTR(20),ROOTI(20)
         DIMENSION  INDEX(10)
         COMMON  TRISE,TSET,DRISE,DSET
         COMMON  PFE,PFF,PFG,PFL,AA,BB,CC,FREQ,PHI,TIME,THET,SGN
         COMMON  COE,ROOTR,ROOTI
         COMMON  XN,FN,XX  ,ABSERR,ANS,ERROR
         COMMON  N1,NFAIL,IQUIT,N ,MAXDEG,JJ
C        INTPOL IS WRITTEN TO HANDLE PROBLEMS WITH UP TO 10 NODES.
C        IF MORE NODES ARE DESIRED THE DIMENSION STATEMENTS MUST BE
C        ALTERED TO HANDLE THE INCREASED NUMBER OF NOEDS.
         MAXDEG=MINO(MAXDEG,N-2)
         L=MAXDEG+2
         LIMIT=MINO(L,N-1)
C        DETERMINE AN ORDER FOR THE NODES XN(I) (STORED IN THE ARRAY
C        INDEX) SUCH THAT XN(INDEX(I)) IS THE NODE CLOSEST TO XX.
C        XN(INDEX(I)) IS THE SECOND CLOSEST, ETC.  THE ARRAY XN IS NOT
C        ALTERED.
C        EVALUATE THE INTERPOLATING POLYNOMIAL AT XX.
         DO 100 I=1,N
         V(I,1)=DABS(XN(I)-XX)
     100 INDEX(I)=I
         DO 120 I=1,LIMIT
         IP1=I+1
         DO 110 J=IP1,N
         II=INDEX(I)
         IJ=INDEX(J)
         IF(V(II,1).LE.V(IJ,1)) GO TO 110
         ITEMP=INDEX(I)
         INDEX(I)=INDEX(J)
         INDEX(J)=ITEMP
     110 CONTINUE
     120 CONTINUE
         PROD=1.0
         I1=INDEX(1)
         ANS=FN(I1)
         V(1,1)=FN(I1)
         DO 140 K=2,L
         IK=INDEX(K)
         V(K,1)=FN(IK)
         KM1=K-1
         DO 130 I=1,KM1
```

```
      II=INDEX(I)
  130 V(K,I+1)=(V(I,I)-V(K,I))/(XN(II)-XN(IK))
      IKM1=INDEX(KM1)
      PROD=(XX-XN(IKM1))*PROD
      ERROR=PROD*V(K,K)
      IF (DABS(ERROR).GT.ABSERR)  GO TO 140
      MAXDEG=K-2
      RETURN
  140 ANS=ANS+ERROR
      ANS=ANS-ERROR
      RETURN
      END

      SUBROUTINE  DISP
      DOUBLE PRECISION  AA,ABSERR,ANS,BB,CC,DISL,      ERROR,FN(10),FREQ
      DOUBLE PRECISION  PFE,PFG,PFL,PHI,POW1,POW2,POW3,THET,TIME
      DOUBLE PRECISION        XN(10),TRISE,TSET,PFF,DRISE,DSET,SGN
      DOUBLE PRECISION  COE(20),ROOTR(20),ROOTI(20)
      DOUBLE PRECISION  COF1,COF2,PSI
      COMMON  TRISE,TSET,DRISE,DSET
      COMMON  PFE,PFF,PFG,PFL,AA,BB,CC,FREQ,PHI,TIME,THET,SGN
      COMMON  COE,ROOTR,ROOTI
      COMMON  XN,FN,DISL,ABSERR,ANS,ERROR
      COMMON  N1,NFAIL,IQUIT,NN,MAXDEG,JJ
C  CALCULATE THET EXPONENT OF ONE REAL POLE.
      POW1=0.0
      IF ((DABS(AA*TIME)).GT.100.0)  GO TO 100
      POW1=DEXP(-1.0*AA*TIME)
C  CALCULATE THE EXPONENT OF ONE REAL POLE.
  100 POW2=0.0
      IF ((DABS(BB*TIME)).GT.100.0)  GO TO 110
      POW2=DEXP(-1.0*BB*TIME)
C  CALCULATE THE EXPONENT OF THE COMPLEX CONJUGATE PAIR.
  110 POW3=0.0
      IF  ((DABS(CC*TIME)).GT.100.0)  GO TO 150
      POW3=DEXP(-0.5*CC*TIME)
  150 COF1=0.0
      IF ((DABS(PFF)).LT.1.0D-10)  GO TO 130
      COF1=PFF
  130 COF2=0.0
      IF ((DABS(PFG)).LT.1.0D-10)  GO TO 140
      COF2=PFG
  140 PSI=0.0
      IF ((DABS(FREQ*TIME+PHI)).LT.1.0D-10)  GO TO 120
      PSI=FREQ*TIME+PHI
C  CALCULATE THE VALUE OF THE OUTPUT FOR A GIVEN VALUE OF TIME.
  120 THET=PFE+COF1*POW1+COF2*POW2+PFL*POW3*DSIN(PSI)
      RETURN
      END

      SUBROUTINE  ENVEL
      DOUBLE PRECISION  AA,ABSERR,ANS,BB,CC,DISL,ERROR,FN(10),FREQ,PFE
      DOUBLE PRECISION  PFF,PFG,PFL,PHI,POW1,POW2,POW3,THET,TIME
      DOUBLE PRECISION        XN(10),TSET,TRISE,DRISE,DSET,SGN
      DOUBLE PRECISION  COE(20),ROOTR(20),ROOTI(20)
      DOUBLE PRECISION  COF1,COF2
      COMMON  TRISE,TSET,DRISE,DSET
      COMMON  PFE,PFF,PFG,PFL,AA,BB,CC,FREQ,PHI,TIME,THET,SGN
      COMMON  COE,ROOTR,ROOTI
      COMMON  XN,FN,DISL,ABSERR,ANS,ERROR
```

```
      COMMON  N1,NFAIL,IQUIT,NN,MAXDEG,JJ
C  CALCULATE THE EXPONENT OF ONE REAL POLE.
      POW1=0.0
      IF ((DABS(AA*TIME)).GT.100.0)  GO TO 100
      POW1=DEXP(-1.0*AA*TIME)
C  CALCULATE THE EXPONENT OF ONE REAL POLE.
  100 POW2=0.0
      IF ((DABS(BB*TIME)).GT.100.0) GO TO 110
      POW2=DEXP(-1.0*BB*TIME)
C  CALCULATE THE EXPONENT OF THE COMPLEX CONJUGATE PAIR.
  110 POW3=0.0
      IF  ((DABS(CC*TIME)).GT.100.0)  GO TO 120
      POW3=DEXP(-0.5*CC*TIME)
  120 COF1=0.0
      IF  ((DABS(PFF)).LT.1.0D-10)  GO TO 130
      COF1=PFF
  130 COF2=0.0
      IF  ((DABS(PFG)).LT.1.0D-10) GO TO 140
      COF2=PFG
  140 THET=PFE+COF1*POW1+COF2*POW2+SGN*PFL*POW3
      RETURN
      END

$ENTRY
```

## APPENDIX F

## SYSTEM PARAMETERS IN STUDY

This appendix contains the parameters which were utilized in the linear and non-linear system simulations. The parameters were obtained from various sources. Some were from component manufacturers, others from experiments performed by the different classes which used the test set-up, and still others from textbooks on the subject.

The parameters utilized in this study can be found in Table V.

TABLE V

EXPERIMENTAL SYSTEM PARAMETERS

| Parameter | Value |
|---|---|
| Fluid bulk modulus | 150,000 psi |
| Orifice discharge coefficient | 0.625 |
| Actuator friction coefficient | 0.10 |
| Valve spool radial clearance | 5.0E-05 in |
| Actuator slip flow coefficient | 0.88E-08 |
| Actuator viscous drag coefficient | 160,000 |
| Steady state flow force velocity coefficient | 0.98 |
| Flapper nozzle control orifice diameter | 0.023 in |
| Valve spool diameter | 0.275 in |
| Fixed orifice diameter upstream from flapper nozzle | 0.012 in |
| Displacement of actuator | 1.512E-02 $in^3$/rad |
| Actuator rotary inertia | 2.16E-03 in $lb_f$ $sec^2$/rad |
| Torquemotor constant | 0.05 $lb_f$/ma |
| Torquemotor constant | 140 $lb_f$/in |
| Mechanical spring rate | 22.5 $lb_f$/in |
| Distance between ports for unsteady flow forces | 0.29 in |
| Spool length for viscous damping | 0.384 in |
| Valve spool mass | 3.2071E-05 $lb_f$ $sec^2$/in |
| Supply pressure | 1,100 psi |
| Exhaust pressure | 0 psi |
| Fluid density | 7.85E-05 $lb_f$ $sec^2/in^4$ |
| Fluid absolute viscosity | 2.0E-06 $lb_f$ $sec/in^2$ |
| External viscous drag | 0.018 in $lb_f$ sec |
| Flapper nozzle displacement at null | 0.0018 in |
| Length of oil volume each end of spool | 0.40 in |
| Volume under compression each oil line | 25.0 $in^3$ |

# VITA 2

Kenneth Joseph Stoss

Candidate for the Degree of

Master of Science

Thesis: SYNTHESIS TECHNIQUE FOR THE COMPENSATION OF ELECTRO-HYDRAULIC SERVOSYSTEMS WITH DYNAMIC PRESSURE FEEDBACK

Major Field: Mechanical Engineering

Biographical:

Personal Data: Born in Great Bend, Kansas, August 8, 1955, the son of Lester and Rita Stoss. Married Maureen Thielen, August 9, 1980, near Dubuque, Iowa.

Education: Attended Barton County Community College, Great Bend, Kansas completing pre-engineering curriculum May 1975; received Bachelor of Science in Mechanical Engineering from Kansas State University, December, 1977; enrolled in MBA program and attended the University of Northern Iowa from August, 1978 to December, 1979; completed requirements for Master of Science degree at Oklahoma State University, December, 1983.

Professional Experience: Research Engineer (summer training program), Amoco Research Center, Naperville, Illinois, 1977; Engineer, Cessna Fluid Power, Hutchinson, Kansas, 1978; Engineer, John Deere Product Engineering Center, Waterloo, Iowa, 1978-date; Graduate teaching assistant, School of Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, Oklahoma, 1981-1983; member of American Society of Mechanical Engineers and the Society of Automotive Engineers.