ornl

OAK RIDGE NATIONAL LABORATORY

LOCKHEED MARTIN

ORNL/TM-13300

RECEIVED
FEB 20 1997
OSTI

Hydraulic Manipulator Design, Analysis, and Control at Oak Ridge National Laboratory

R. L. Kress
J. F. Jansen
L. J. Love
A. M. Hasanul Basher

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

MANAGED AND OPERATED BY
LOCKHEED MARTIN ENERGY RESEARCH CORPORATION
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

ORNL-27 (3-96)

This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from the Office of Scientific and Technical Information, P. O. Box 62, Oak Ridge, TN 37831; prices available from (423) 576-8401, FTS 626-8401.

Available to the public from the National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Road, Springfield, VA 22161.

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government of any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

ORNL/TM-13300

HYDRAULIC MANIPULATOR DESIGN, ANALYSIS, AND CONTROL AT OAK RIDGE NATIONAL LABORATORY

R. L. Kress and J. F. Jansen
Robotics and Process Systems Division
Oak Ridge National Laboratory
P. O. Box 2008
Oak Ridge, Tennessee 37831-6426
Telephone: 423-574-2468
Facsimile: 423-576-2081

L. J. Love
Oak Ridge Institute For Science And Education
P. O. Box 2008
Oak Ridge, Tennessee 37831-6426
Telephone: 423-576-4630
Facsimile: 423-576-2081

A. M. Hasanul Basher
South Carolina State University
565 Elmwood Drive, N.E.
Orangeburg, South Carolina 29115

This report is the final version of an earlier report delivered to the Department of Energy's Office of Technology Development, Robotics Technology Development Program, Environmental Management Program, Cross-Cutting and Advanced Technology Modular Robotic Scoping Study for Fiscal Year 1993.

Date Published–September 1996

Prepared by the
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37831
managed by
LOCKHEED MARTIN ENERGY RESEARCH CORP.
for the
U.S. DEPARTMENT OF ENERGY
under contract DE-AC05-96OR22464

CONTENTS

Page

ABSTRACT V

1. INTRODUCTION 1
   1.1 GENERAL COMMENTS ON HYDRAULICS 1
   1.2 HISTORY OF HYDRAULIC MANIPULATORS AT OAK RIDGE NATIONAL LABORATORY 1

2. HYDRAULIC SYSTEM MODELING 7
   2.1 GENERAL HYDRAULIC MODELING AND ANALYSIS 7
   2.2 FLEXIBLE/PRISMATIC TEST STAND DESIGN AND MODELING 10
   2.3 HUMAN EXTENDER/AMPLIFIER MODELING AND ANALYSIS 11
      2.3.1 Description of Human Extender/Amplifier 11
      2.3.2 Human Extender/Amplifier Modeling 13
      2.3.3 Parameters for Human Amplifier Model 15
      2.3.4 Human Extender/Amplifier Control Objectives and Controller Design 18
      2.3.5 Stability Problems with the Human Extender/Amplifier System 19
   2.4 ANALYSIS OF PARAMETERS THAT VARY AS A FUNCTION OF TEMPERATURE 23

3. EXPERIMENTAL RESULTS 25
   3.1 FLEXIBLE/PRISMATIC CHARACTERIZATION 25
      3.1.1 Flexible/Prismatic Test Stand Mechanical Modeling 25
      3.1.2 Flexible/Prismatic Test Stand Sensors 27
   3.2 FLEXIBLE/PRISMATIC TEST STAND TEMPERATURE VARIATION EXPERIMENTS 27
   3.3 HUMAN EXTENDER/AMPLIFIER 31
   3.4 GENERAL EXPERIMENTAL COMMENTS AND OBSERVATIONS 31

4. FUTURE HYDRAULIC SYSTEMS 33
   4.1 GUNITE AND ASSOCIATED TANKS 33
   4.2 APPLICATION OF DAWM TO THE CP-5 REACTOR AT ANL 33
   4.3 SCHILLING 7F 33
   4.4 DEXTEROUS MANIPULATION OF HEAVY PAYLOADS 34

5. CONCLUSIONS 35

REFERENCES 37

APPENDIX A: LIST OF SYMBOLS 39

APPENDIX B: GOVERNING EQUATIONS FOR HYDRAULIC DYNAMICS 41

APPENDIX C: RAW DATA FOR SELECTED EXPERIMENTS FROM THE EXTENDED OPERATION EXPERIMENTS WITH THE FLEXIBLE/PRISMATIC TEST BED 47

APPENDIX D: END POINT POSITION SENSOR DATA FOR FLEXIBLE/PRISMATIC TEST BED.....59

APPENDIX E: VENDOR LITERATURE FOR HARDWARE TEST STANDS.....51

ABSTRACT

In the early developmental stages of robotics, hydraulics played an important role. Many of the early high-payload capacity manipulators were actuated by hydraulic cylinders and hydraulic rotary actuators. As the power-to-weight ratio of electric motors increased, they eventually came to be the preferred form of actuation for robotic manipulators because of the relative ease of operation, control, maintenance, and for general cleanliness. Recently, however, task requirements have dictated that manipulator payload capacity increase to accommodate greater payloads, greater manipulator length, and larger environmental-interaction forces. General tasks such as waste storage tank clean-up and facility dismantlement and decommissioning require manipulator lift capacities in the range of hundreds of pounds rather than tens of pounds. To meet the increased payload capacities demanded by present-day tasks, manipulator designers have turned once again to hydraulics as a means of actuation. Hydraulics have always been the actuator of choice when designing heavy-lift construction and mining equipment such as bulldozers, backhoes, and tunneling devices. In order to successfully design, build, and deploy a new hydraulic manipulator (or subsystem) sophisticated modeling, analysis, and control experiments are usually needed. To support the development and deployment of new hydraulic manipulators Oak Ridge National Laboratory (ORNL) has outfitted a significant experimental laboratory and has developed the software capability for research into hydraulic manipulators, hydraulic actuators, hydraulic systems, modeling of hydraulic systems, and hydraulic controls. The hydraulics laboratory at ORNL has three different manipulators. First is a 6-Degree-Of-Freedom (6-DoF), multi-planer, teleoperated, flexible controls test bed used for controls and teleoperator research. Second is a 2-DoF Flexible/Prismatic test bed used for the development of waste tank clean-up manipulator controls, thermal studies, system characterization, and manipulator tracking. Finally, is a human amplifier test bed used for the development of an entire new class of teleoperated systems. To compliment the hardware in the hydraulics laboratory, ORNL has developed a hydraulics simulation capability including a custom package to model the hydraulic systems and manipulators for performance studies and for controls development. This paper outlines the history of hydraulic manipulator developments at ORNL, describes the hydraulics laboratory, discusses the use of the equipment within the laboratory, and presents some of the initial results from experiments and modeling associated with these hydraulic manipulators. Included are some of the results from the development of the human amplifier/de-amplifier concepts, the characterization of the thermal sensitivity of hydraulic systems, and end-point tracking accuracy studies. Experimental and analytical results are included.

FIGURES

1. The Soldier Robot Interface Platform (SRIP) Vehicle
2. The Future Armor Rearm System (FARS)
3. The Telerobotic Small Emplacement Excavator (TSEE)
4. The Dual-Arm Work Module (DAWM)
5. The DAWM using a pipe cutting tool
6. The Schilling 7F, 6-D.O.F., multiplaner, teleoperated, flexible controls test bed
7. The flexible/prismatic test stand in the ORNL Hydraulics Laboratory
8. Bode plots for the transfer function models of Eqs. 2.1a and 2.1b
9. Root loci plot for transfer function models of Eqs. 2.1a and 2.1b with a proportional controller. The top figure is for $G_1(s)$ and the bottom for $G_2(s)$
10. Flexible/prismatic test bed hydraulic system model in Simulink
11. The ORNL 1-D.O.F. human amplifier
12. Free-body diagram of human extender test stand
13. Human extender/amplifier block diagram with controller
14. Position control with G_c+K_0(constant) = 1 and $E = A_p \sin(\omega t)$
15. Y/V(s) transfer function (A_p was set for 1 in. travel). X's mark actual data points, and the lines, represent the fit to the data
16. $F_{LC1}/E(s)$ transfer function's Bode Plots. X's mark actual data points, and the lines represents the fit to the data
17. Block diagram for the free-space stability problem for the human extender/amplifier
18. Bode plot for $G_c(s) = 1$ and $\alpha_F=1$ (the X's representing the actual data points have been removed to improve the quality of the figure)
19. Block diagram for the nonlinear contact problem #2 with environmental contact but no operator for the human extender/amplifier
20. Nichols plot for the contact stability problem 2 with environmental contact but no operator for the human extender/amplifier ($\alpha_F = 1$)
21. Analytical determination of the natural frequency as a function of length and payload for the flexible/prismatic test bed
22. Natural frequency as a function of length for 25-lb_f payload for the flexible/prismatic test bed
23. Temperature vs time for a flexible/prismatic test bed extender operation experiment
24. Minimum and maximum plant models for the rotary joint of the flexible/prismatic test bed extender operation experiment of Fig. 23
25. Plant variation of the rotary hydraulic joint as a function of temperature for the flexible/prismatic test bed extender operation experiment of Figs. 23 and 24
26. Tank waste removal hardware for the GAAT cleanup effort at ORNL

1. INTRODUCTION

1.1 GENERAL COMMENTS ON HYDRAULICS

Hydraulically actuated manipulators play an important role in real applications because of their high payload-to-mass ratios compared with conventional electrically actuated manipulators. Comparable power-to-mass ratios range from 3.3 kW/kg (2 hp/lb_m) for hydraulic systems to 50 W/kg (0.03 hp/lb_m) for electric systems.[1] The best electric motors are now getting as high as 1 kW/kg (0.6 hp/lb_m) power-to-mass ratios.

In many industrial applications where a high power capacity is desired, hydraulics is the general form of actuation. Hydraulics provides a number of unique features. First, the fluid provides a natural method of lubrication and cooling. There is no phenomenon in hydraulic components that is equivalent to the saturation and concomitant losses in magnetic materials as is associated with electric motors. Torque output from a hydraulic machine is limited only by safe stress levels. Furthermore, hydraulic actuators have a high stiffness compared to other drive devices. In addition, they have a higher speed of response as well as large torque-to-inertia ratios, providing high acceleration capacity. Hydraulic components may be operated in continuous, intermittent, reversing, and stalled conditions without damage. In addition, rotary and linear hydraulic actuators are available for many different sizes and power ranges.[2]

Hydraulics technology is not without its disadvantages. For example, hydraulic power is not as readily available in most industrial settings as is electric power and most stationary applications must have a hydraulic power supply installed. Hydraulic components are expensive. Hydraulic fluids are sometimes flammable and/or considered to be hazardous waste. Hydraulic systems almost always leak and are therefore considered messy. Hydraulic fluids must be filtered and in some cases filtered thoroughly for high-performance applications such as servovalves. Contaminated oil is one of the primary reasons for component failure in hydraulic systems. Hydraulic actuators are not generally as flexible and easy to use for low-power applications as are electric actuators.

1.2 HISTORY OF HYDRAULIC MANIPULATORS AT OAK RIDGE NATIONAL LABORATORY

The applied nature of the projects at Oak Ridge National Laboratory (ORNL) and the direct results of the value of hydraulics has made it the power source of choice in many ORNL projects. Consequently, ORNL has had considerable experience with hydraulic manipulators and movable systems over the past decade. This section briefly describes some of the hydraulic hardware systems developed at ORNL.

Except for some very early work with commercially available teleoperated manipulators, one of the first hydraulic systems developed at ORNL was the Soldier Robot Interface Platform (SRIP) vehicle shown in Fig. 1.[3]

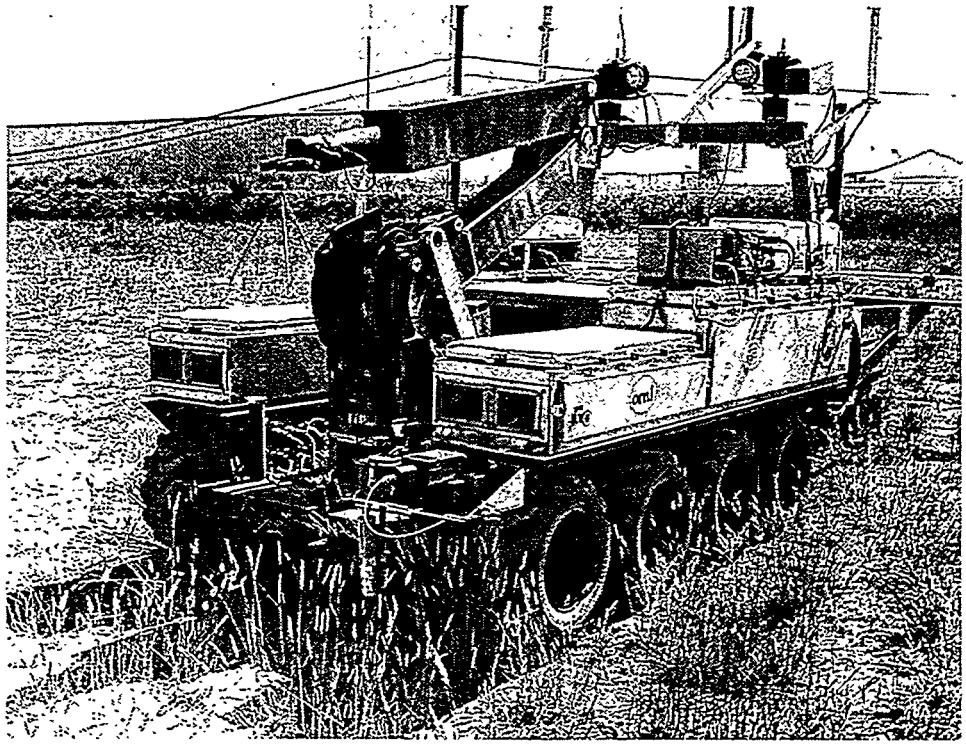


Fig. 1. The Soldier Robot Interface Platform (SRIP) vehicle.

The SRIP vehicle was designed and built with the assistance of the Tooele Army Depot in Utah. The SRIP has a hydrostatic-drive transmission and was fitted with an electric arm and numerous sensor systems. It was designed for two purposes: the military intended it as a platform for research into unexploded ordinance disposal, and the Department of Energy (DOE) supported it for buried waste site characterization and remediation. One of the particularly challenging aspects of the SRIP development was providing it with the ability to maintain an accurate trajectory necessary for complete coverage of an area during a waste characterization survey in spite of limited accuracy wheel position sensors, a natural tendency to slip on rought terrain and/or loose soil conditions, and a hydrostatic transmission. At the time of the development of SRIP, low-cost and accurate Global Positioning System (GPS) sensors were not readily available.

Another hydraulic system developed at ORNL was the Future Armor Rearm System (FARS), shown in Fig. 2.[4,5] The FARS was designed and built with the assistance of Tooele Army Depot. The FARS was developed to allow the Army to rearm its new M1A1 tanks without exposing the soldier to the hazards of the battlefield. The FARS hydraulically actuated arm served several purposes: (1) to dock with the empty tank; (2) to be a communication link between the tank and the FARS rearm vehicle; and (3) to transfer the ammunition between the tank and the FARS vehicle. Control of the FARS arm proved to be a challenge in achieving the fine motion required to dock with the tank rearm port. Management of the interaction forces between the arm and the docking port during contact was proven to be a hydraulic system challenge. Ultimately, interaction forces were controlled by developing a compliant rearm port on the tank. Flexibility and low natural frequency would be a problem with the FARS arm if it were moved quickly. To avoid exciting the arm's structural dynamics, however, the joints are moved slowly.

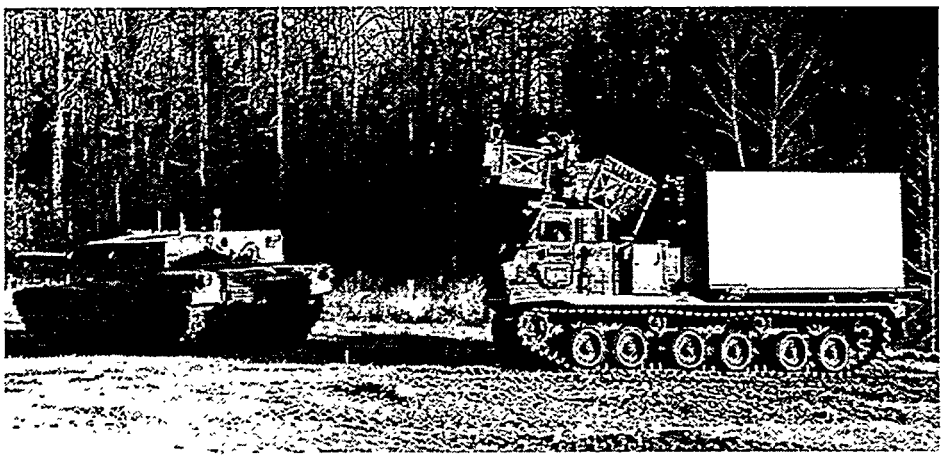


Fig. 2. The Future Armor Rearm System (FARS).

The next system developed by ORNL was the Telerobotic Small Emplacement Excavator (TSEE), shown in Fig. 3.[6,7] Like the SRIP, this system was developed as a dual-use system. The military supported development as a platform for research into unexploded ordinance disposal, and DOE wanted it for buried waste site characterization and remediation. One of the particularly challenging aspects of the TSEE development was providing it with the ability to be remotely operated as well as be operated as it was originally designed with no interference to the operator from the modifications for remote operation. In addition, the TSEE control panel was made to be portable with an intuitive operator interface. Arm and bucket motions were controlled by a single joystick where the direction of the joystick motion matched the physical direction and motion of the arm and bucket. The control system provided an adjustable dig floor to prevent the operator from inadvertently digging below a desired level. These intuitive features made the TSEE easy and simple to use; in fact, over half of the soldiers surveyed in an operational experiment comparing the use of the remotely operated TSEE with a conventionally operated version preferred the remotely operated TSEE. Low-cost and reliable proportional valves were used in the TSEE vehicle. The TSEE has also been operated over a computer internet link over distances of thousands of miles.

As part of the DOE Robotics Technology Development Program's support of decontamination and dismantlement (D&D) efforts, the Dual-Arm Work Module (DAWM) was developed by ORNL and RedZone robotics.[8,9] This system is the most current manipulator in the evolutionary development of telerobotic manipulators at ORNL.[10] The DAWM, shown in Fig. 4, features two 6-Degree-Of-Freedom (D.O.F.), hydraulically actuated, Schilling manipulators, and a 5-D.O.F., hydraulically actuated base. It was initially deployed off a 4-D.O.F. electrically actuated, gantry-like transporter and suspended from an overhead crane but could also be deployed off of a mobile platform.[11] Each of the Schilling arms is capable of lifting 240 lb fully extended. A similar system will be used to support the D&D activities at the CP-5 reactor at Argonne National Laboratory (ANL). The ORNL DAWM is used for support of the D&D effort at ANL in the areas of operator training, tool and fixture testing and development, control algorithm development and testing, cost/benefit experimental analysis, and operator interface design and evaluation. The DAWM is shown in a D&D mockup in Fig. 5.

Another hydraulic manipulator system at ORNL is the Schilling 7F, 6-D.O.F., multiplaner, teleoperated, flexible controls test bed used for controls and teleoperator research for hydraulically actuated, flexible-link manipulators. This system is shown in Fig. 6.

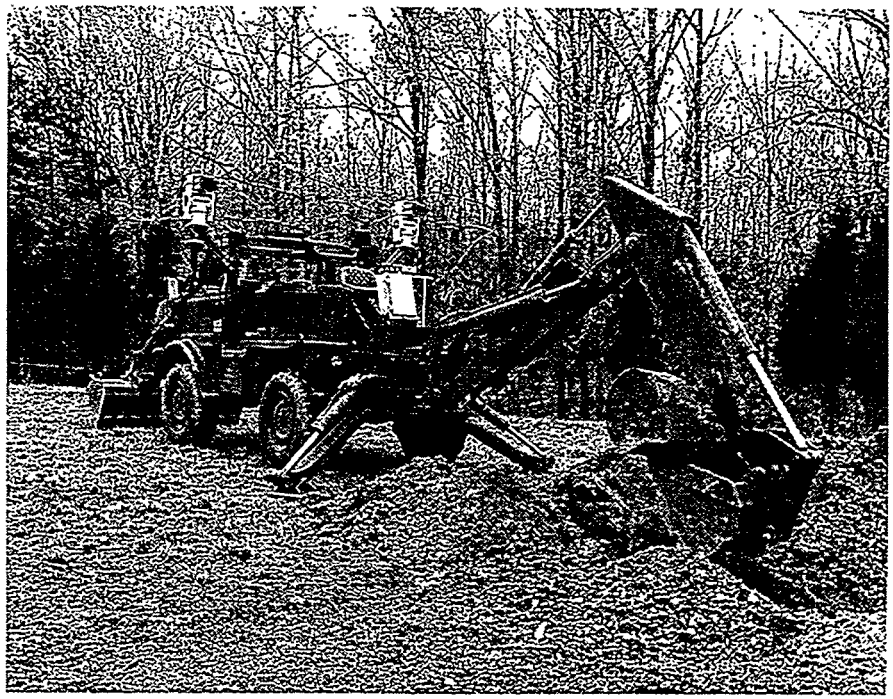


Fig. 3. The Telerobotic Small Emplacement Excavator (TSEE).

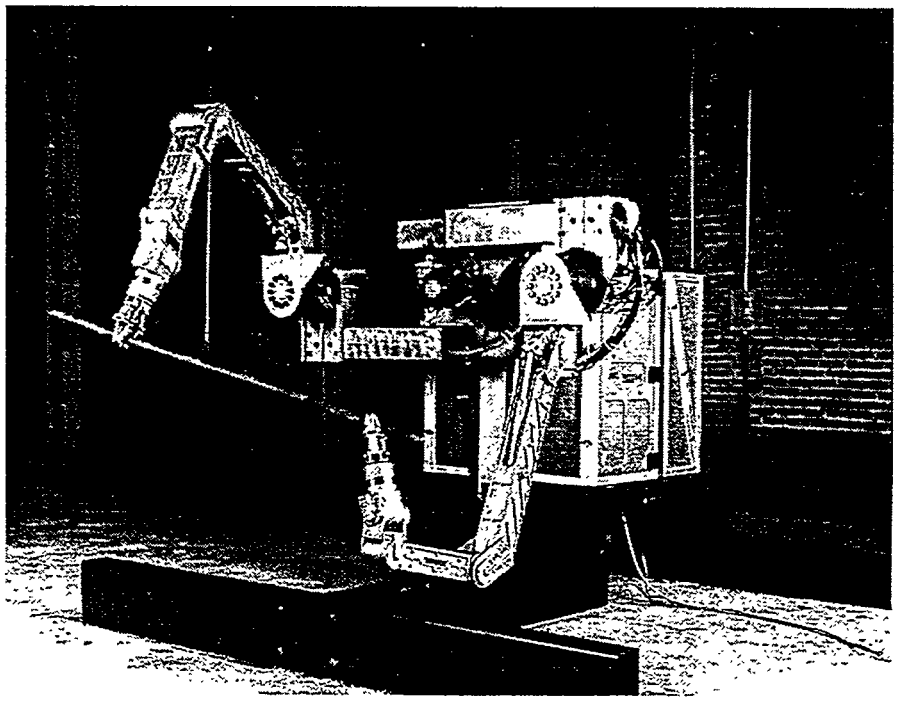


Fig. 4. The Dual-Arm Work Module (DAWM).

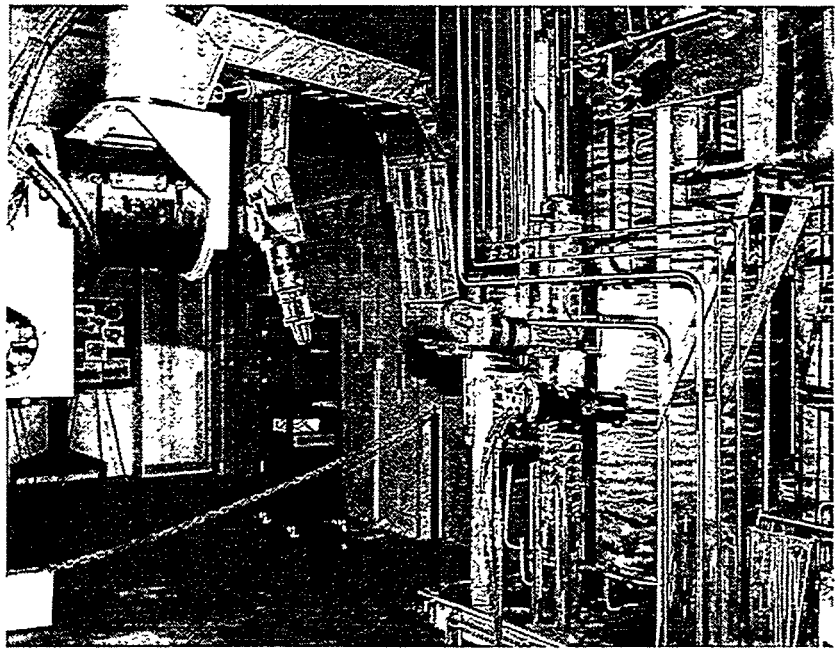


Fig. 5. The DAWM using a pipe cutting tool.

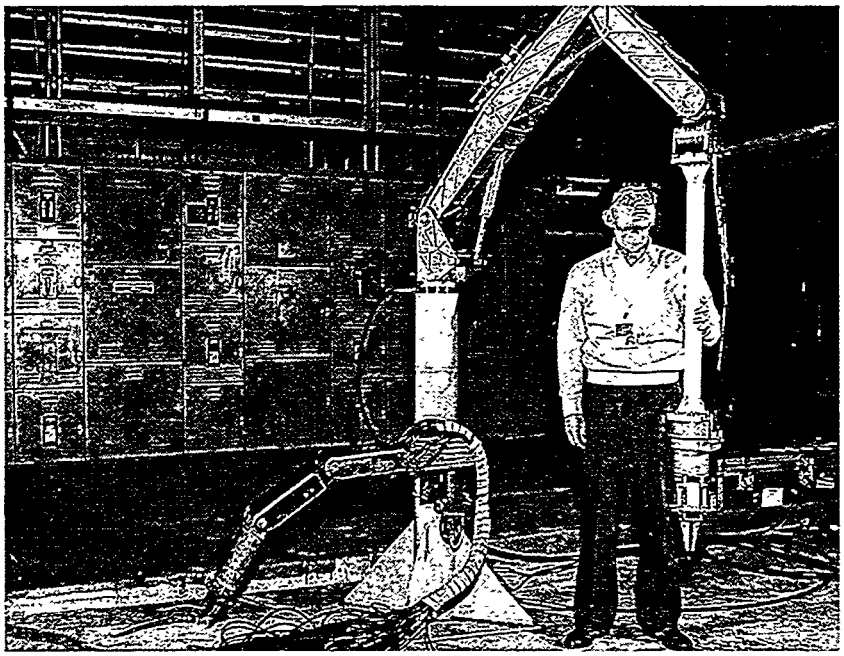


Fig. 6. The Schilling 7F, 6-D.O.F., multiplaner, teleoperated, flexible controls test bed.

This system is presently being used for event-based controller integration in support of ORNL's Gunite And Associated Tanks (GAAT) clean up effort.[12] This system has also been used for mock-up of larger arms and for preliminary demonstrations of hardware that is normally deployed off of larger manipulators. This system is normally teleoperated, but it has been converted to run robotically. Limited accuracy and low reliability remain a problem.

2. HYDRAULIC SYSTEM MODELING

2.1 GENERAL HYDRAULIC MODELING AND ANALYSIS

When considering the transfer function from voltage input to speed output electric motors respond as a simple lag device. In contrast, hydraulic actuator transfer functions between voltage and speed are basically quadratic in form with a relatively high natural frequency. This difference makes controller design and analysis much more difficult for hydraulic systems.

The flexible/prismatic test stand, shown in Fig. 7, was developed as a research manipulator for the ORNL Hydraulics Laboratory.

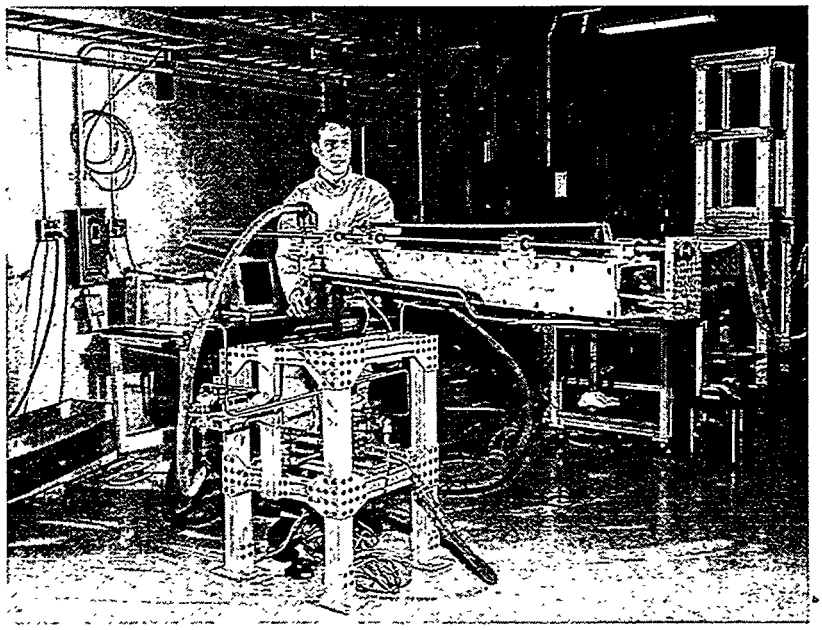


Fig. 7. The flexible/prismatic test stand in the ORNL Hydraulics Laboratory.

Its main objective is to serve as a research test bed for studying the effects of hydraulics and link flexibility on manipulator design and control. The goal of the design is to have a realistically sized actuator and payload. The mechanical details are described in greater detail in the Sect. 3, Experimental Results.

Consider two representations of the valve command (volts) to position output (in.) transfer function, G, for a typical hydraulic actuator shown in Eqs. 2.1a and 2.1b:

$$G_1(s) = \frac{437.6352}{s(s+37.53)}, \tag{2.1a}$$

$$G_2(s) = \frac{35782}{s(s^2 + 81.7618s + 2606.1)}, \tag{2.1b}$$

where s is the LaPlace transform variable.

These transfer functions are taken from actual fits to data obtained by using varying frequency sine waves as inputs to the prismatic joint servovalve of the ORNL flexible/prismatic test bed at the nominal operating temperature of ~25°C. Analytically derived transfer function representations from the valve command to the position output are presented in Merritt's book.[2] For a prismatic actuator with valve command in in., and the position output in in., one obtains:

$$y = \frac{\frac{K_{ql}}{A} x_v}{s\left(\frac{V_T M}{4\beta_e A^2} s^2 + \left(\frac{K_{PTp} M}{A^2} + \frac{B_p V_T}{4\beta_e A^2}\right) s + \left(1 + \frac{B_p K_{PTp}}{A^2}\right)\right)} \quad . \tag{2.2a}$$

Note that all of the symbols for all of the equations in this report are defined in Appendix A. For reference, the transfer function for the valve command in inches and the output position in radians for a rotary actuator is given as

$$\theta_m = \frac{\frac{K_{ql}}{D_m} x_v}{s\left(\frac{V_T J_T}{4\beta_e D_m^2} s^2 + \left(\frac{K_{PTm} J_T}{D_m^2} + \frac{B_m V_T}{4\beta_e D_m^2}\right) s + \left(1 + \frac{B_m K_{PTm}}{D_m^2}\right)\right)} \quad . \tag{2.2b}$$

Appendix B contains the governing equations for the derivation of the transfer functions of Eqs. 2.2a and 2.2b.

Examining Eqs. 2.1a and 2.1b in light of Eq. 2.2a, one can determine that the third-order transfer function of Eq. 2.1b is the one that is physically correct (second-order voltage to speed with an integrator); however, the second-order transfer function of Eq. 2.1a is the analytically equivalent electrical motor model for the hydraulic actuator. Equation 2.1a was obtained by fitting the nonphysical second-order transfer function to the same set of data used for the fit of Eq. 2.1b. A similar model can be obtained by developing a reduced-order transfer function from Eq. 2.1b. Note that the third-order fit has 50 times less error than the second-order fit. Bode plots from both models are shown in Fig. 8.

Notice that there is little difference between the frequency response of either representation except at the high frequencies; however, the phase angle plots are much different at frequencies greater than 10 rad/sec, as is expected because of the difference between the third-order vs the second-order transfer functions (second-order system with no constant term will go from -90° to -180°, whereas the third-order system phase will go from -90° to -270°). It is precisely this difference in the dynamic performance that makes hydraulic systems more difficult to control.

Consider the root loci of Fig. 9. These correspond to the transfer functions of Eqs. 2.1a and 2.1b and are for a proportional controller only.

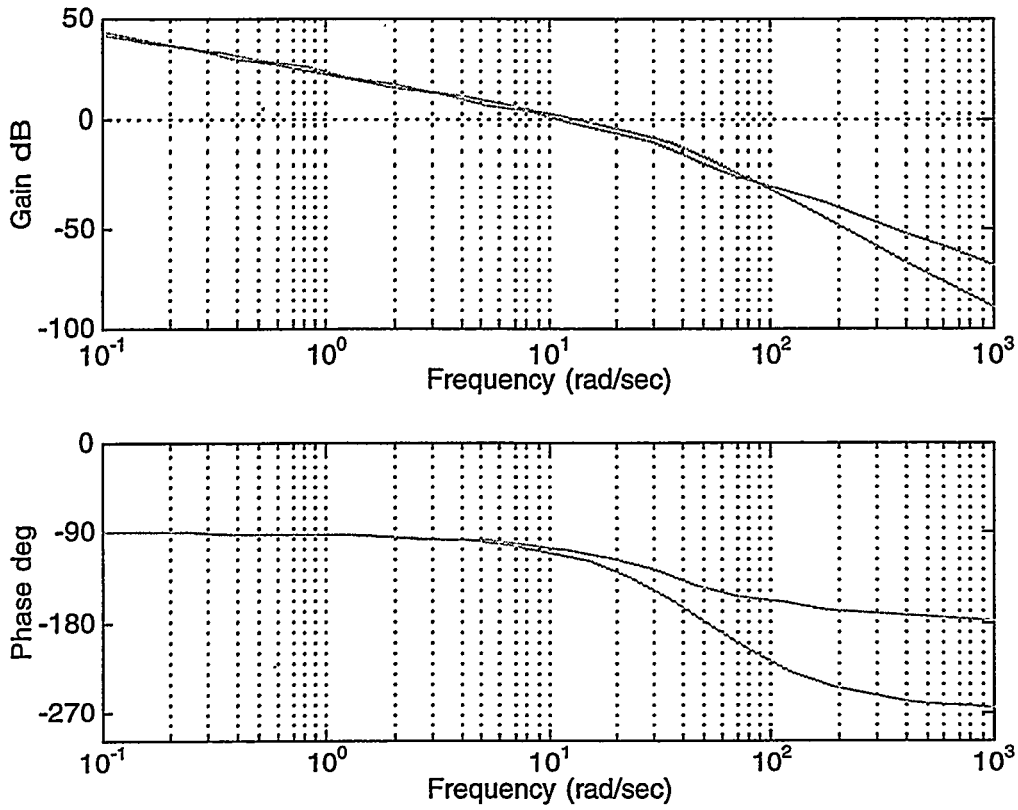


Fig. 8. Bode plots for the transfer function models of Eqs. 2.1a and 2.1b.

As the gain is increased in the root locus plot for the second-order transfer function, $G_1(s)$, the roots move from one on the negative real axis and one at the origin to both at the same point on the negative real axis and then along a line parallel to the Y-axis. However, at no time do the roots ever move into the right-half plane, indicating that (at least theoretically) this system remains stable for all values of the proportional gain. (Certainly, this is not true in reality since, as when the gain is increased, the system is no longer linear, and the transfer function does not adequately describe the system dynamics. In addition, noise becomes an important factor as the gain increases.) Considering the root locus for the third-order transfer function, $G_2(s)$, one observes that as the value of the proportional gain is increased, the real parts of the two complex conjugate roots move into the right-half plane. This indicates that there are values of the gain where the system becomes unstable regardless of nonlinearities and/or noise. This added complexity makes the development of controllers for hydraulically actuated manipulators more difficult than for electrically actuated manipulators and requires more extensive and more detailed modeling and analysis prior to hardware implementation. In addition, important hardware choices must be made in the initial design if a high-performance system is to be achieved. This includes the use of stiff hydraulic lines (e.g., metal as far as possible), the mounting of servovalves as close as possible to actuators, careful attention to the

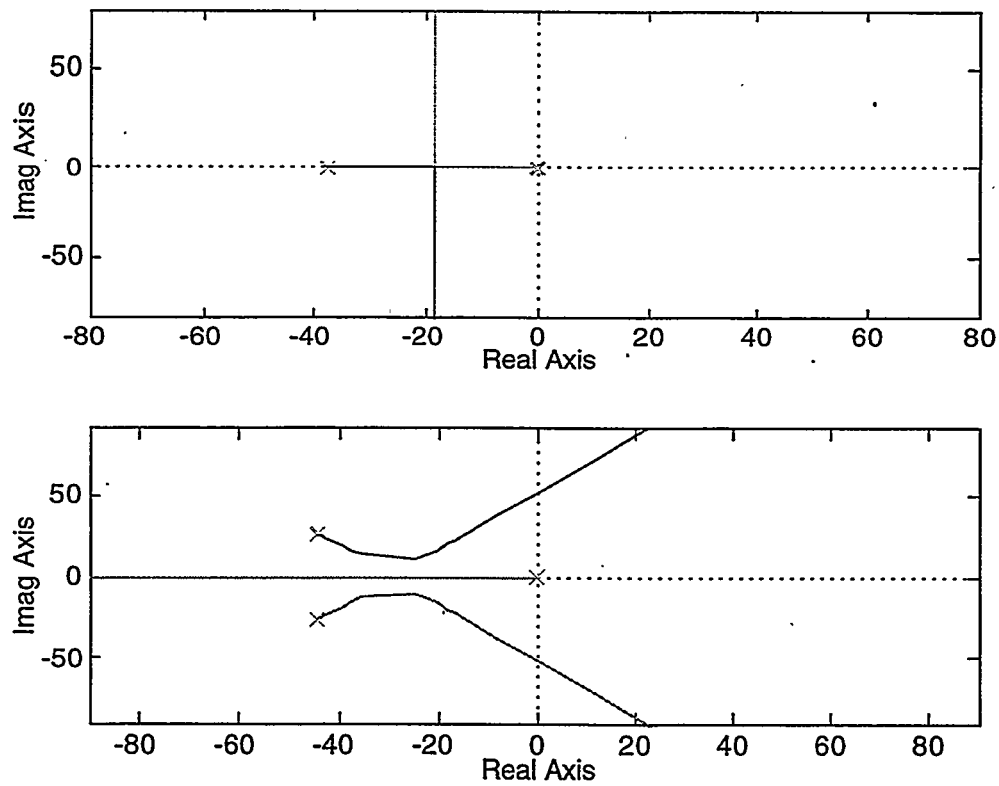


Fig. 9. Root loci plot for transfer function models of Eqs. 2.1a and 2.1b with a proportional controller. The top figure is for $G_1(s)$ and the bottom for $G_2(s)$.

removal of all entrapped air, and selection of high-performance components, including low-friction actuators and zero deadband valves. Because of the need for detailed modeling and analysis, the following sections describe models developed for some of the ORNL hydraulic systems that are used for controller design and evaluation.

2.2 FLEXIBLE/PRISMATIC TEST STAND DESIGN AND MODELING

Models were developed for the flexible/prismatic test bed mechanical design as well as for the hydraulic systems. Mechanical modeling was used in the test bed design and is discussed in Section 3.2.1. The hydraulic system of the flexible/prismatic test bed was modeled in detail using Simulink and Matlab. It was necessary to use a computer modeling package as opposed to relying completely on analytical models because of the significant set of nonlinear behaviors present in typical hydraulic systems. These nonlinearities include nonlinear friction, both stiction and Coulomb friction; nonlinear drive-train compliance; actuator saturation; mechanical backlash and deadband; servovalve nonlinear friction and deadband; entrapped air within the hydraulic fluid; nonlinear servovalve orifice effect; the effects of fluid contamination; pressure losses in piping, including bends and reductions; asymmetric actuator

areas and time-varying nonlinear effects. Not all of these nonlinearities were modeled, but several of the important ones were included in the formulation of the governing dynamics' equations. These equations are shown in Appendix B. An example of the Simulink model of the flexible/prismatic test bed is shown in Fig. 10.

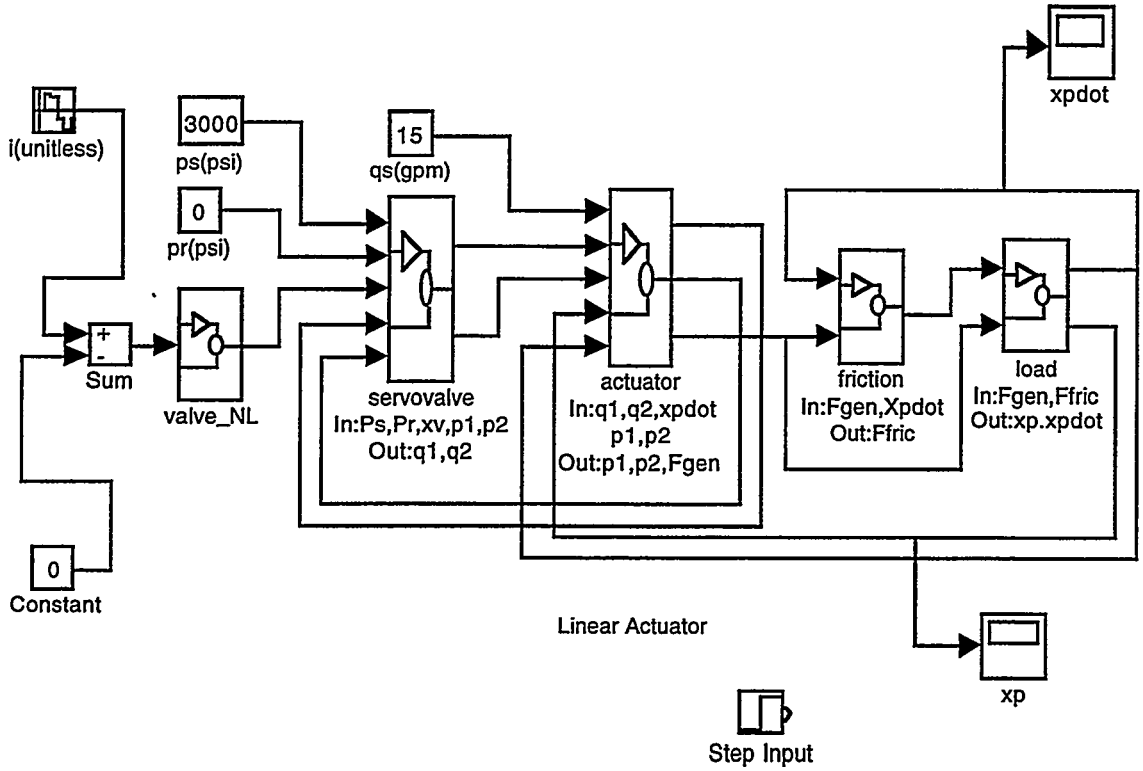


Fig. 10. Flexible/prismatic test bed hydraulic system model in Simulink.

2.3 HUMAN EXTENDER/AMPLIFIER MODELING AND ANALYSIS

2.3.1 Description of Human Extender/Amplifier

A human extender (also called a human amplifier system) is a device that amplifies the lifting capacity of a person and allows a preselected amount of force feedback to the operator (e.g., the operator can feel any desired portion of the load). This type of manipulator system is similar to a teleoperated system in that a human operator is coupled directly to the mechanical system; however, it is fundamentally different from the traditional teleoperated manipulator system in the sense that the master and slave manipulators are one integral unit. Applications of a human extender system include the following:

1. Material handler in an unstructured environment where manipulating and orienting large objects, while transmitting back to the operator a fraction of the object's dynamics (i.e., its weight, contact forces, slippage, etc.), could significantly enhance productivity, quality, and safety (e.g., missile loader and rearming tanks/heavy artillery for the military).
2. Rescue operations (e.g., Oklahoma bombing rescue, firefighting).
3. Material handler in the construction industry (i.e., taking large loads off trailers, as is typically done by a large crane; moving large pipes; putting up sheetrock; etc.).

4. Medical (e.g., patient manipulation).
5. Material handler in the mining industry.
6. Material handler in the forestry industry.

The human extender problem was first addressed in the 1960s by General Electric during the Hardyman project.[13] More recently, H. Kazerooni has been working on a scaled-down version of a similar concept.[14-16] These systems have always had difficulties because of profound stability issues associated with varying dynamics and gross nonlinearities in the fluid power system (e.g., nonlinear pressure-flow relationship, time-varying fluid properties, large quantities of nonlinear friction, and time-varying system dynamics).

The goals of the ORNL human amplifier system are to achieve high lift capacity (around 500 lb), force amplification (from 1 to 500), and tracking performances (submillimeter range). The effects of human dynamics must be minimized to achieve these goals.

Detailed hydraulic models that include most of the nonlinear fluid and mechanical dynamics have been generated for a 1-D.O.F. human extender test stand. Fundamental stability limits and how they relate to the mechanical device have been analytically developed and experimentally evaluated on the hydraulic test stand. The ORNL human extender/amplifier is shown in Fig. 11.

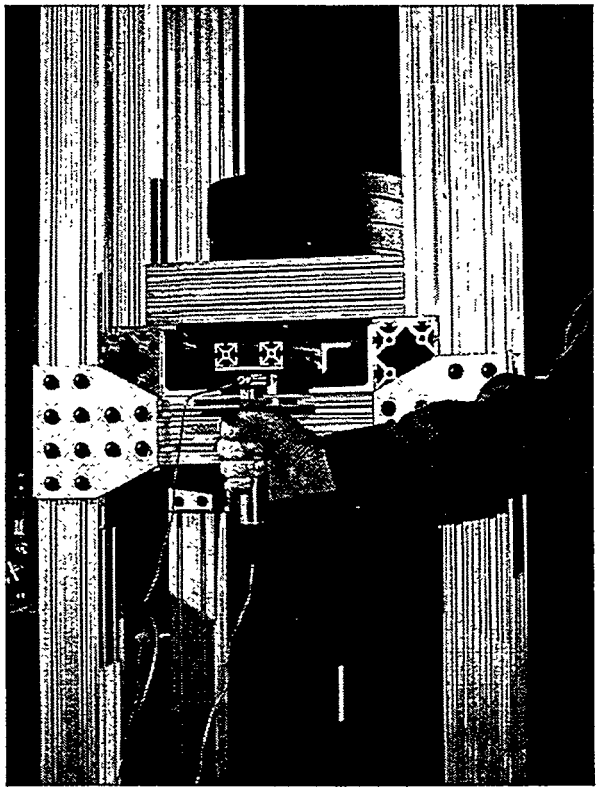


Fig. 11. The ORNL 1-D.O.F. human amplifier.

2.3.2 Human Extender/Amplifier Modeling

The governing linear equations for the human extender/amplifier will be derived in this section. The following derivation is valid for a symmetrical actuator with the servovalve spool at its center position. Using the center position as an operating point is a reasonable assumption for this system and its standard operating conditions. The human amplifier system at ORNL does not have a symmetrical actuator; however, this approximation is adequate to describe the dynamics of the system based on the experimental model verification demonstrated in the next section. Figure 12 is a free-body diagram of the human extender test stand.

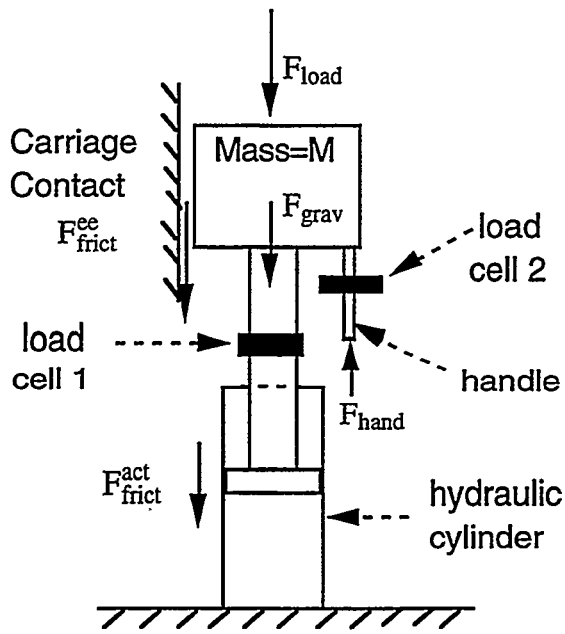


Fig. 12. Free-body diagram of human extender test stand.

The following is a linear analysis of the model of the electrohydraulic system shown in Fig. 12. (See Ref. 2 for details and the list of symbols in Appendix A for definitions of terms.) Flow to the actuator, Q_L, is related to command current to the servovalve, I, with load pressure, P_L (which is similar to back emf for dc-motors) reducing the total flow,

$$Q_L = K_q I - K_p P_L . \tag{2.3}$$

For a hydraulic cylinder, the time rate of change of the actuator position, $\dot{y}$, is related to the effective flow into the actuator, Q_L, minus the amount that the fluid compresses (proportional to the time rate of change of the load pressure P_L) and minus the amount leaked across the cylinder seals (proportional to the load pressure P_L),

$$A\,\dot{y} = Q_L - \frac{V_T}{4\,\beta_e}\frac{d\,P_L}{dt} - C_{tp}\,P_L . \tag{2.4}$$

The actuator force, F_{act}, is equal to the dynamic forces, frictional losses, external load forces (i.e., contact with an environment), gravity (i.e., weight of the hardware), minus the hand forces (since the operator is in contact with the frame),

$$F_{act} = A\,P_L = M\frac{d\,\dot{y}}{dt} + F_{frict} + F_{load} + F_{grav} - F_{hand}\,. \qquad (2.5)$$

Frictional forces can be divided into actuator frictional forces and everything else (i.e., friction associated with the carriage),

$$F_{frict} = F_{frict}^{act} + F_{frict}^{ee}\,. \qquad (2.6)$$

Load cell 1, placed between the cylinder and the frame, measures the cylinder actuator forces minus the actuator frictional forces with some sensor noise, N_{LC1},

$$F_{LC1} = F_{act} - F_{frict}^{act} + N_{LC1}\,. \qquad (2.7)$$

Load cell 2, placed between the operator handle and the frame, measures the hand forces with some sensor noise, N_{LC2},

$$F_{LC2} = F_{hand} + N_{LC2}\,. \qquad (2.8)$$

The total system is represented in the block diagram in Fig. 13.

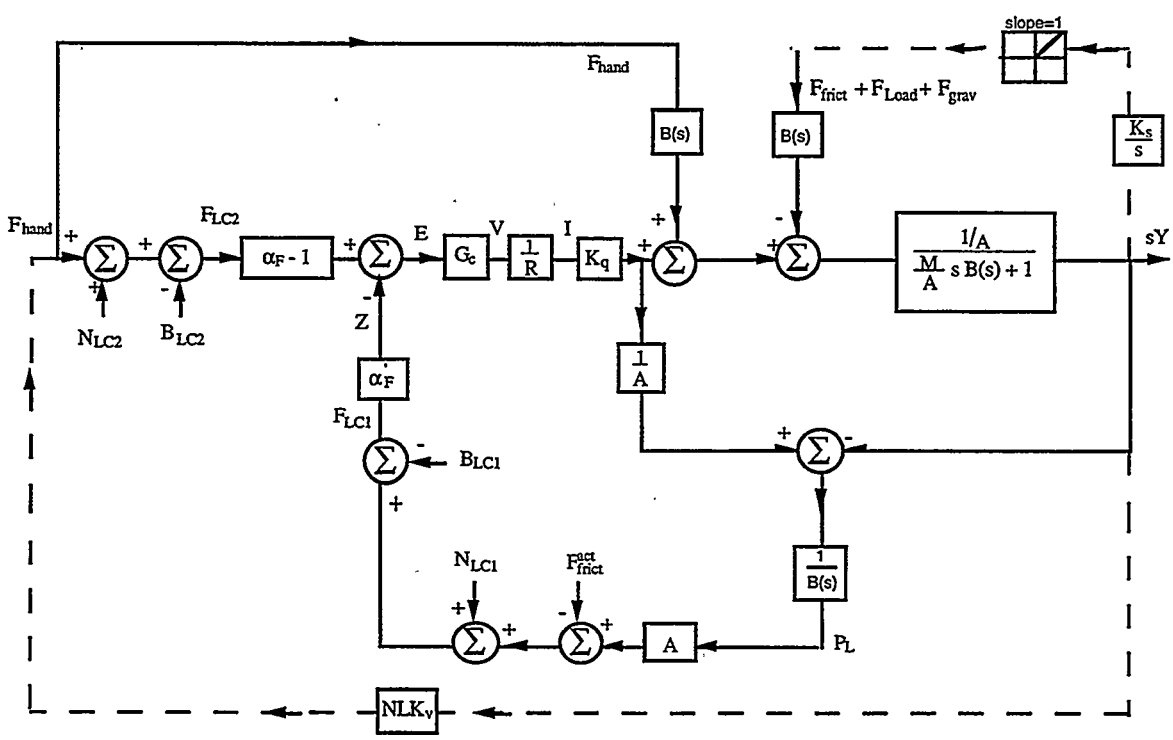


Fig. 13. Human extender/amplifier block diagram with controller.

Some of the blocks on the diagram require further description. The N and B terms that add to the force of the hand and the actuator prior to their respective load cell measurements represent noise and bias respectively. The nonlinearity in the upper-right corner represents the environment; forces are zero while moving in free space and then contact is made with an environment represented by a linear spring with rate K_S. G_C is a compensator that is to be

designed to stabilize and improve the performance of the human amplifier. B(s) is a transfer function containing valve flow pressure coefficient from Eq. 2.3 and leakage and compressibility term from Eq. 2.4. NLK_V is a nonlinear relationship between the amplifier frame motion and the human hand which will be discussed later. Note also that α_F is the primary force reflecting gain and α_F' is the auxiliary force reflecting gain.

2.3.3 Parameters for Human Amplifier Model

Two tests were conducted to determine values for the physical plant model in the block diagram of Fig. 13. For the first test, a sinusoidal test signal is injected at E in Fig. 14.

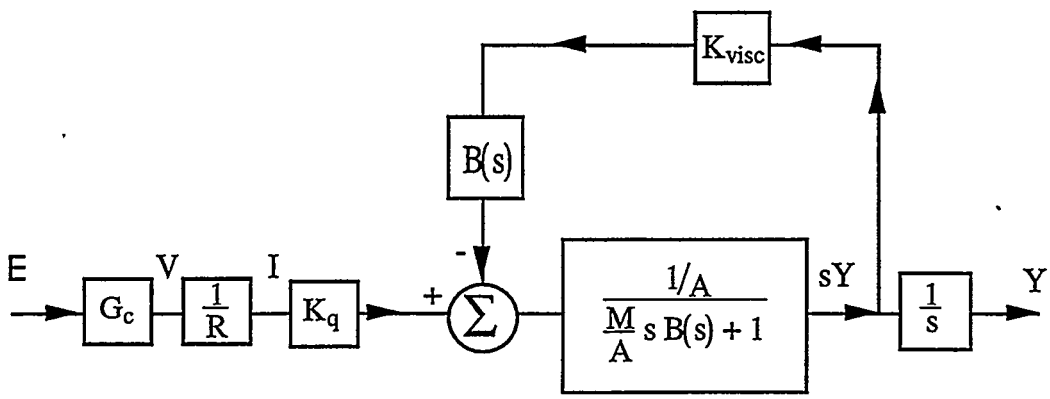


Fig. 14. Position control with $G_c = K_0$ (constant) = 1 and $E = A_p \sin(\omega t)$.

Note that Fig. 14 is the dynamic representation of operator-free, free space, unloaded motion from Fig. 13; consequently, only the viscous friction term is included in the velocity feedback loop. The transfer function between Y and E (or between Y and V since $G_c = 1$) is

$$\frac{Y}{V}(s) = \frac{\dfrac{K_q}{RA}}{s\left\{\dfrac{M V_T}{4\,\beta_e A^2}s^2 + s\left[\dfrac{M}{A^2}(K_p + C_{tp}) + \dfrac{K_{visc} V_T}{4\,\beta_e A^2}\right] + \left[1 + \dfrac{K_{visc}}{A^2}(K_p + C_{tp})\right]\right\}}. \tag{2.9}$$

Because of the low leakage and low viscous friction, $\frac{K_{visc}}{A^2}(K_p + C_{tp}) \cong 0$, Eq. (2.9) can be simplified to

$$\frac{Y}{V}(s) \cong \frac{\dfrac{K_q}{RA}}{s\left\{\dfrac{M V_T}{4\,\beta_e A^2}s^2 + s\left[\dfrac{M}{A^2}(K_p + C_{tp}) + \dfrac{K_{visc} V_T}{4\,\beta_e A^2}\right] + 1\right\}}. \tag{2.10}$$

The model of Eq. 2.10 taken from typical data is shown in Fig. 15. This Bode plot is only for a fixed A_p of 1 in.; the frequency was swept through the discrete values shown by the x-marks (from 0.6 rad/s to almost 200 rad/s). For different amplitudes A_p, the model changes because of the nonlinearities of the system (e.g., nonlinear orifice flow relationship in the servovalve, nonlinear friction, etc.); but the general trends are always the same.

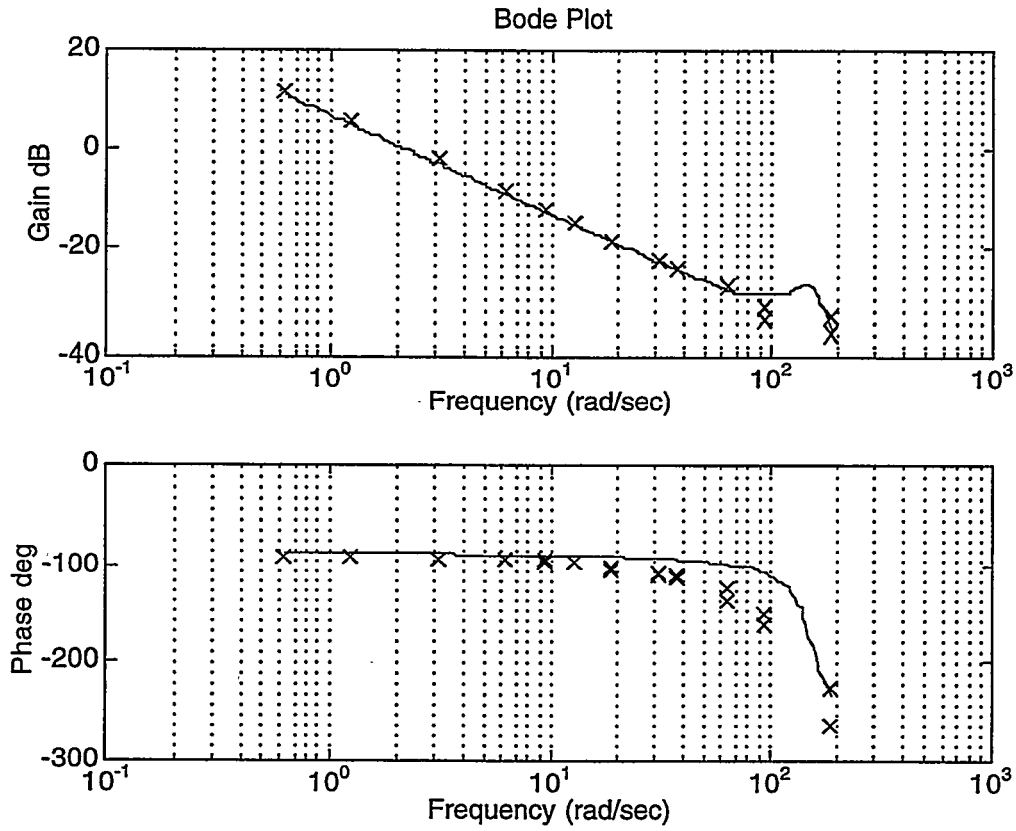


Fig. 15. Y/V(s) transfer function (A_p was set for 1 in. travel). X's mark actual data points, and the lines, represent the fit to the data.

The second test to determine values for the physical plant model in the block diagram of Fig. 13 is based on inserting a sinusoidal command signal into the error signal, E, and then measuring the force on the load cell between the actuator and the frame, F_{LC1}. All positional and force control loops were opened. The transfer function between F_{LC1} and E (ref. Fig. 13), assuming $\frac{K_{visc}}{A^2}(K_p + C_{tp}) \cong 0$ as in Eq. 2.10,

$$\frac{F_{LC1}}{E}(s) \cong \frac{\frac{K_q M}{R A}\left(s + \frac{K_{visc}}{M}\right)}{\frac{M V_T}{4 \beta_e A^2} s^2 + s\left[\frac{M}{A^2}(K_p + C_{tp}) + \frac{K_{visc} V_T}{4 \beta_e A^2}\right] + 1} . \tag{2.11}$$

An example of typical data collected for this transfer function is shown in Fig. 16.

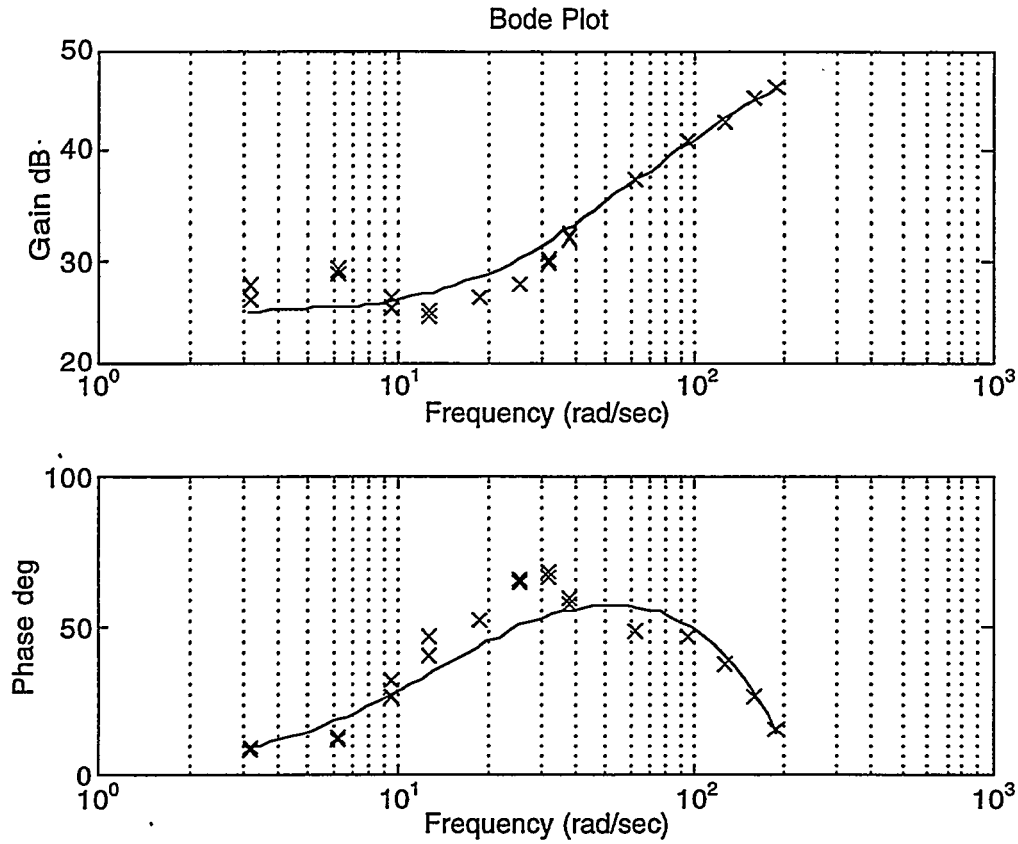


Fig. 16. $F_{LC1}/E(s)$ transfer function's Bode Plots. X's mark actual data points, and the lines represents the fit to the data.

The objective of these tests and the prior measurements is to determine the coefficients in the models of Eqs. 2.10 and 2.11 and to ultimately obtain an estimate of the amount of uncertainty is in these system models so that adequate margins could be inserted into the compensator design to accommodate these plant variations. Table 1 shows the values of similar coefficients taken from the models of Eqs. 2.10 and 2.11 and clearly illustrates the amount of variability resulting from the different models and measurements. These two model types were found to bound the data collected (as viewed from the Bode plots) for this test stand and will be used in the controller design.

Table 1. Coefficient values estimated for the models of Eqs. 2.10 and 2.11.

Model Type	$\frac{M V_T}{4 \beta_e A^2}$	$\frac{M}{A^2}(K_p + C_{tp}) + \frac{K_{visc} V_T}{4 \beta_e A^2}$	$\frac{K_q M}{R A}$
For $\frac{\dot{Y}}{V}(s)$ (Eq.2.10)	4.273E-5	2.189e-3	3.098
For $\frac{F_{LC1}}{E}(s)$ (Eq.2.11)	1.859E-5	5.093e-3	4.862

2.3.4 Human Extender/Amplifier Control Objectives and Controller Design

Typical controllers drive an error function toward zero; therefore, to determine the controller objectives for the human amplifier, the appropriate error signal must be formulated. The overall objective of the human extender/amplifier is to magnify the operator's hand force to overcome the dynamics of the mechanical system and load. This is represented by the following relationship:

$$\alpha_F F_{hand} = M\frac{d\dot{y}}{dt} + F^{ee}_{frict} + F_{load} + F_{grav} \ . \tag{2.12}$$

Combining the system dynamic and friction equations (Eqs. 2.5 and 2.6 respectively) one obtains the following:

$$M\frac{d\dot{y}}{dt} + F^{ee}_{frict} + F_{load} + F_{grav} = F_{act} + F_{hand} - F^{act}_{frict} \ . \tag{2.13}$$

Recognizing that load cell 1 (F_{LC1}) measures the actuator forces, F_{act}, minus the frictional losses, $Fact_{frict}$, and that load cell 2 (F_{LC2}) measures the hand forces, F_{hand}, both with added noise (neglecting sensor bias), then

$$M\frac{d\dot{y}}{dt} + F^{ee}_{frict} + F_{load} + F_{grav} = F_{LC1} + F_{LC2} + \text{Sensor Noise} \ . \tag{2.14}$$

If sensor noise is ignored then combining Eqs. 2.12 and 2.14 and recognizing that F_{hand} is also F_{LC2} produces

$$\alpha_F F_{hand} \cong \alpha_F F_{LC2} \cong F_{LC1} + F_{LC2} \ . \tag{2.15}$$

or

$$(\alpha_F - 1)\, F_{LC2} \cong F_{LC1} \ . \tag{2.16}$$

The error equation can then be formulated as

$$e = (\alpha_F - 1)\, F_{LC2} - F_{LC1} \ . \tag{2.17}$$

The controller's objective is to drive the error signal, e, to zero. Only forces are being utilized for a human extender. This is in contrast to teleoperated systems, where either a position-position or a position-force type control architecture is employed.[17] This error signal is shown in Fig. 13 as E. Next, it will be shown that it is desirable to set the controller gain to as high a value as possible to mitigate the effect of the large cylinder friction.

One can show that the velocity of the cylinder and the actuator friction (modeled as a disturbance) can be related, assuming no contact with the environment or the operator ($F_{hand} = 0$), as

$$\frac{\dot{Y}}{F^{act}_{frict}}(s) = \frac{-\frac{B}{A}}{\frac{M}{A}\, s\, B + 1 + G_c\, \alpha_F \frac{K_q M}{R A}} \ . \tag{2.18}$$

Equation 2.18 clearly shows that in the limit, as the compensator approaches an infinite gain ($G_c \rightarrow \infty$), the effect of friction due to the actuator will go to zero. This is the motivation for trying to keep the controller gain as high as possible over an acceptable range of frequencies.

2.3.5 Stability Problems with the Human Extender/Amplifier System

Three stability problems are encountered by the human extender system. The first is the free-space problem. This is where the device does not make contact with the environment or a human being. The second problem is the contact stability problem. For this condition, the device is making contact with the environment, but a human being is not attached to the device. The last stability problem is where contact is being made with the environment and the human being is attached to the device. The case where a human being is holding the handle and no contact is being made is a subset of the last problem and therefore will not be addressed.

1. Free-Space Problem (no contact with environment, either human or task):

   i. Set $F_{frict} + F_{Load} + F_{grav} = 0$, $B_{LC1} = B_{LC2} = 0$, $N_{LC1} = N_{LC2} = 0$, $NLK_s = 0$, $F_{hand} = 0$, and $F_{frict}^{act} = 0$.

The resulting block diagram is shown in Fig. 17.

   ii. The transfer function between the error, E, and Z (newly defined in the block diagram of Fig. 17) is

$$\frac{Z}{E}(s) = \frac{\alpha_F' \left(\frac{G_c K_q M}{R A} s \right)}{\frac{M}{A} s B + 1} \quad . \tag{2.19}$$

The Bode plot for Eq. 2.19 using the two sets of coefficients from the two different models in Table 1 is shown in Fig. 18.

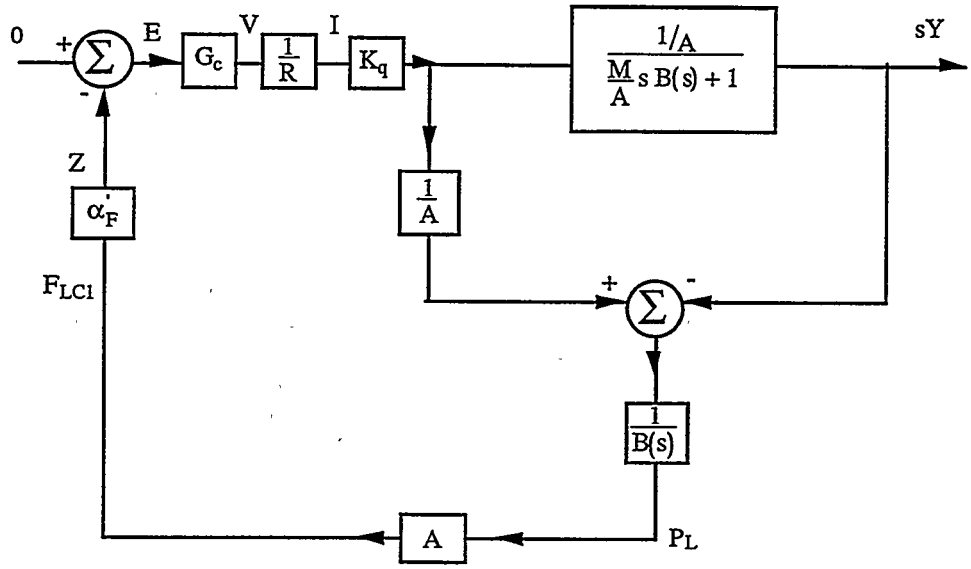


Fig. 17. Block diagram for the free-space stability problem for the human extender/amplifier.

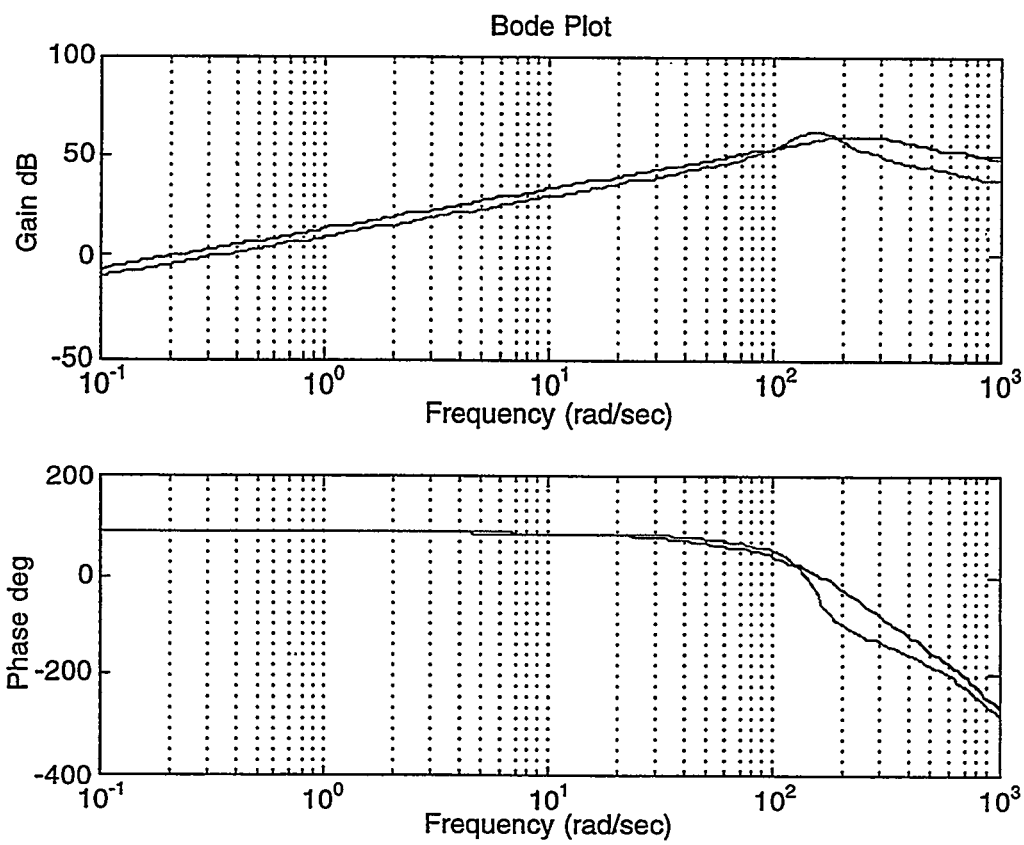


Fig. 18. Bode plot for $G_c(s) = 1$ and $\alpha_F'=1$ (the X's representing the actual data points have been removed to improve the quality of the figure).

Equation 2.19 is the open-loop transfer function. When the loop is closed, the transfer function between F_{hand} and $\dot{Y}$ is

$$\frac{\dot{Y}}{F_{hand}}(s) = \frac{\frac{G_c K_q}{R A}\left(\alpha_F + \alpha_F' - 1\right) + \frac{B}{A}}{\frac{M}{A} s B + \alpha_F'\left(\frac{G_c K_q M}{R A} s\right) + 1}. \tag{2.20}$$

2. Contact Stability Problem with $F_{hand} = 0$:
   i. Set $B_{LC1} = B_{LC2} = 0$, $N_{LC1} = N_{LC2} = 0$, $NLK_s = 0$, and $F_{frict}^{act} = 0$.
   ii. A nonlinear relationship between $\dot{Y}$ and an output F_{Load} (as shown by the dashed line in upper right corner of the block diagram in Fig. 13) represents the contact with the environment.
   iii. The transfer function between an input $\dot{Y}$ and an output F_{Load} is

$$\frac{\dot{Y}}{F_{Load}}(s) = \frac{-\left(\frac{B}{A} + \alpha_F' \frac{G_c K_q}{R A}\right)}{\frac{M}{A} s B + \alpha_F'\left(\frac{G_c K_q M}{R A} s\right) + 1}. \tag{2.21}$$

The closed-loop control problem is shown in Fig. 19.

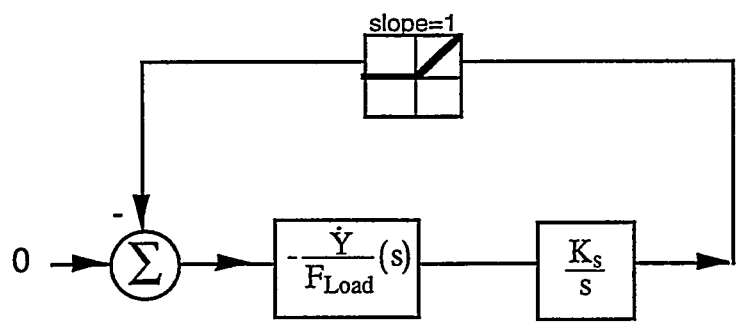


Fig. 19. Block diagram for the nonlinear contact problem #2 with environmental contact but no operator for the human extender/amplifier.

3. Contact Stability Problem with F_{hand} as an Input:
   i. Set $B_{LC1} = B_{LC2} = 0$, $N_{LC1} = N_{LC2} = 0$, and $F_{frict}^{act} = 0$.
   ii. A nonlinear relationship between $\dot{Y}$ and an output F_{Load} (as shown by the dashed line in upper-right corner of the block diagram in Fig. 13) represents the contact with the environment.
   iii. A nonlinear relationship exists between $\dot{Y}$ and F_{hand} and is represented by NLK_v. An assumption will be made that this relationship can be approximated by a linear relationship (e.g., linear damper and spring) as $F_{hand} = -G_v(s)\,\dot{Y}$.

iv. The transfer function between an input F_{Load} and an output $\dot{Y}$ is

$$\frac{\dot{Y}}{F_{Load}}(s) = \frac{-\left(\frac{B}{A} + \alpha_F' \frac{G_c K_q}{RA}\right)}{\frac{M}{A} s B + \alpha_F' \left(\frac{G_c K_q M}{RA} s\right) + 1 + G_v \left[\frac{G_c K_q}{RA}\left(\alpha_F + \alpha_F' - 1\right) + \frac{B}{A}\right]}. \tag{2.22}$$

This nonlinear problem is the same structurally as the one shown in Fig. 19.

To address problem 1, it is clear from Fig. 18 that in the low to mid frequency range the open loop system looks like a derivative operator (e.g., mass $d\dot{y}/dt$ = force). To achieve higher gains over this frequency range, either an integrator or lag compensator is needed in the controller compensator $G_c(s)$

$$G_c(s) = \frac{K_c}{s}. \tag{2.23}$$

In fact, it will be shown that an integrator for $G_c(s)$ will satisfy the next two problems as well. Setting the gain in Eq. 2.23, K_c, to 0.0028 provides at least a 25-dB-gain margin for problem 1. Integral compensation is not new to force control of manipulators;[18] however, the main contribution of this analysis is that the nonlinear contact stability problem is formulated and the stability bounds and performances of the human extender/amplifier are tied to the test stand design parameters that were experimentally measured.

The describing function approach was found to be inadequate to solve these types of nonlinear control problems because of a violation of the Filter Hypothesis,[19] that is, the output waveform is not dominated by the fundamental component; rather, it has a more extensive frequency content. Instead of using the describing function approach, the circle criterion[20] was employed. According to the circle criterion, the transfer function $\hat{g}(j\omega) = \left[\frac{-\dot{Y}}{F_{Load}}(s)\ \frac{K_s}{s}\right]\Big|_{s=j\omega}$ must lie in the half-plane $\{s: \text{Re } s > -1\}$ for stability to be guaranteed. It needs to be stressed that the circle criterion only states that if this region is avoided, then stability is assured; however, if this region is penetrated, then the circle criterion says nothing. In other words, it is sufficient to stay out of the region $\{s: \text{Re } s > -1\}$ to ensure stability, but it is not necessary. The constraint of avoiding $\{s: \text{Re } s > -1\}$ can be mapped to the Nichols chart, and this curve is shown in Fig. 20 as the open oval passing through the 0-dB, -180° point.

The load cell on the actuator has a stiffness of 2.8871×10^7 N/m (or 1.67×10^5 lb/in.). Since the load cell is more compliant than the steel or aluminum of the actuator and frame, it is the critical design parameter. The environmental stiffness K_s was selected to be 0.02 of the load cell compliance. Based on our experience, we could penetrate the circle criteria area and still maintain a stable system if we stayed away from the -1 point (e.g., 0-dB, -180°). Assuming a value of environmental stiffness of 0.02 of the load cell stiffness means that contact can only be made with relatively compliant objects such as rubber or cork. The obvious fix would be to add a compliant surface to the frame (i.e., end effector on a final design) such that hard contact with stiff objects such as metal is possible. This was the approach taken in the FARS project. The following discussion outlines an alternative solution for the hard contact problem (problem 3).

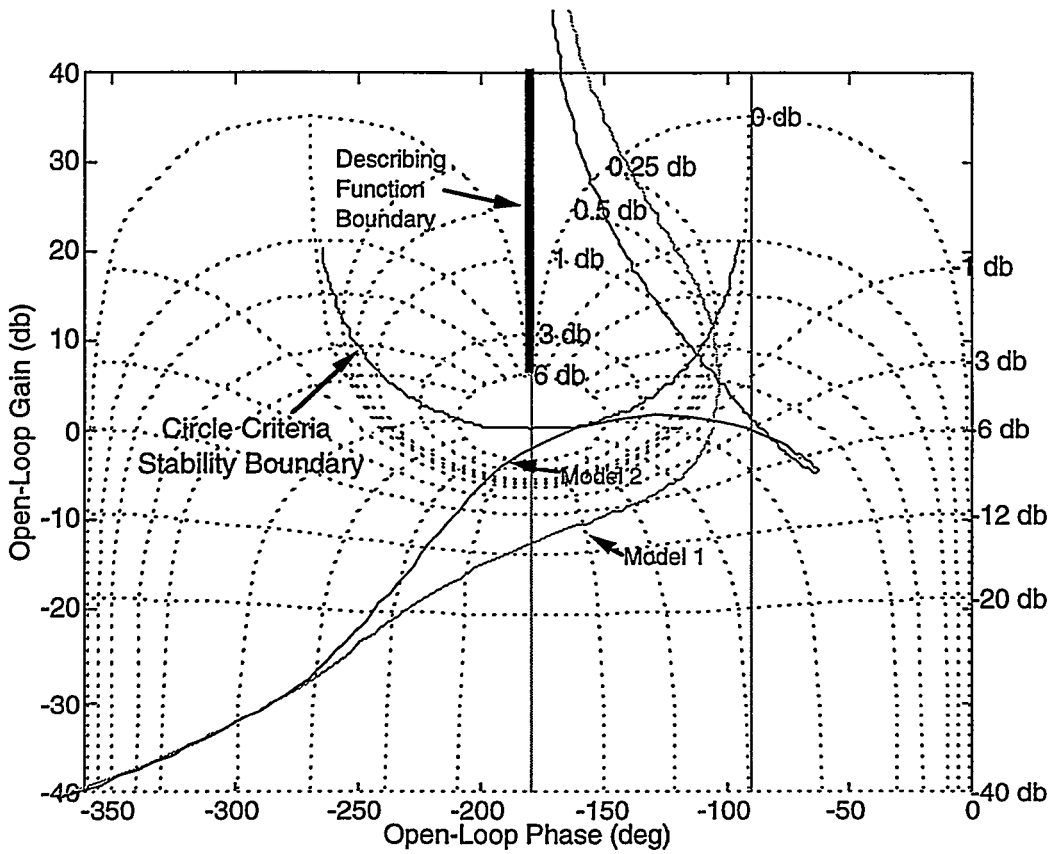


Fig. 20. Nichols plot for the contact stability problem 2 with environmental contact but no operator for the human extender/amplifier ($\alpha_F' = 1$).

To address problem 3 (environmental contact and operator contact), it can be shown that for higher force reflection gains, α_F, the model curves in Fig. 20 move lower along the gain axis. Moving the model curves down along the gain axis provides a greater margin between the model curves and the circle criteria stability boundary. This means that environments having larger stiffness can be accommodated. Mathematically, this becomes apparent by noting that α_F is in the denominator of Eq. 2.22. It can be shown that when the gains are around 200, hard contact with metals is possible without a compliant surface. Physically, this becomes apparent when one considers that with higher force gains the human operator can absorb more of the damping energy during contact. Once gains of up to 200 have been achieved, noise from the sensor becomes an issue and can no longer be neglected in the analysis. Section 3 will discuss an effective solution to the noise problem and other experimental observations.

2.4 ANALYSIS OF PARAMETERS THAT VARY AS A FUNCTION OF TEMPERATURE

When modeling hydraulic systems, it is important to recognize that the proper model is dependent upon the operating conditions, especially the temperature. Several of the fluid properties that influence the performance of a hydraulic manipulator vary as a function of temperature, and these are summarized in this section.

Density varies as a function of temperature and pressure, as is shown in Merritt:[2]

$$\rho = \rho_0 \left[1 + \frac{1}{\beta}(P - P_0) + \alpha(T - T_0) \right]. \tag{2.23}$$

Viscosity and bulk modulus vary with temperature, and examples are shown in Tables 2 and 3.

Table 2. Viscosity as a function of temperature for Houghto-Safe® hydraulic fluid.

Temperature (°C)	(°F)	Viscosity (Centistokes)*	Viscosity (Saybolt Universal Seconds)
-18	0	648	3000
21	70	78	361
38	100	42	200
54	130	24	117
66	150	17	89

The equation relating Centistokes to Saybolt Universal Seconds is Centistokes = 0.216SUS - 166/SUS.

Table 3. Bulk modulus as a function of temperature for a MIL-H-5606B hydraulic fluid[2]

Temperature (°C)	(°F)	Bulk Modulus (kN/m²)	(psi)
-18	0	2620	380,000
38	100	2413	350,000
93	200	2068	300,000
149	300	1379	200,000
204	400	620	90,000

Viscosity variation may be approximated by the following equation[2]:

$$\mu = \mu_0 e^{-\lambda(T-T_0)}, \qquad (2.24)$$

where λ is a fluid-dependent constant.

Pressure loss in bends, expansions, and entrances creates a rise in temperature and a resulting change in fluid properties. Temperature increases as a result of energy input through the pump but also through friction and energy loss through pipes and bends.

3. EXPERIMENTAL RESULTS

3.1 FLEXIBLE/PRISMATIC CHARACTERIZATION

3.1.1 Flexible/Prismatic Test Stand Mechanical Modeling

The flexible/prismatic test stand was designed to support ORNL's task cleanup and D&D activities. It was designed to carry realistic payloads with realistically sized hydraulic actuators. The prismatic joint of the flexible/prismatic test stand consists of a hardened steel tube with 1.0-in. OD and a 0.6 in. ID. This tube can extend from 12 to 60 in. In addition, the payload can vary from 10 to 75 pounds. With this range of payload and displacement, the arm can match the natural frequencies expected with the MLDUA. In addition, small displacements in the position of the prismatic actuator can provide dramatic variations in the natural frequency, as illustrated in Figs. 21 and 22. With the speed capacity of the prismatic and rotary joints, the natural frequency of the arm can vary by an order of magnitude over a very short range of motion in a very short time.

Figure 21 shows the variation of the flexible/prismatic test bed's link natural frequency as a function of payload and prismatic displacement. The stroke of the cylinder is 48 in. with the link length varying from 12 to 60 in.

Figure 22 shows the sensitivity of the flexible beam's natural frequency to variations in the prismatic displacement (in this case the payload is 25 lb_f).

The rotary actuator on the flexible-prismatic test bed is a Parker HTR30 hydraulic rack and pinion rotary actuator. At 2000 psi, this actuator has a maximum torque capacity of 20,000 in.-lb_f. The prismatic joint is powered by a Parker Series EH hydraulic cylinder with a 2 in. bore. Its force capacity at 2000 psi is 6280 lb_f. Moog 760 series valves control the fluid flow. These high-bandwidth valves are relatively popular in the aircraft and robotics community. The valve on the rotary joint has a rated flow of 5 gpm, while the prismatic joint's flow is rated at 15 gpm. Vendor literature for this hardware is included in Appendix E.

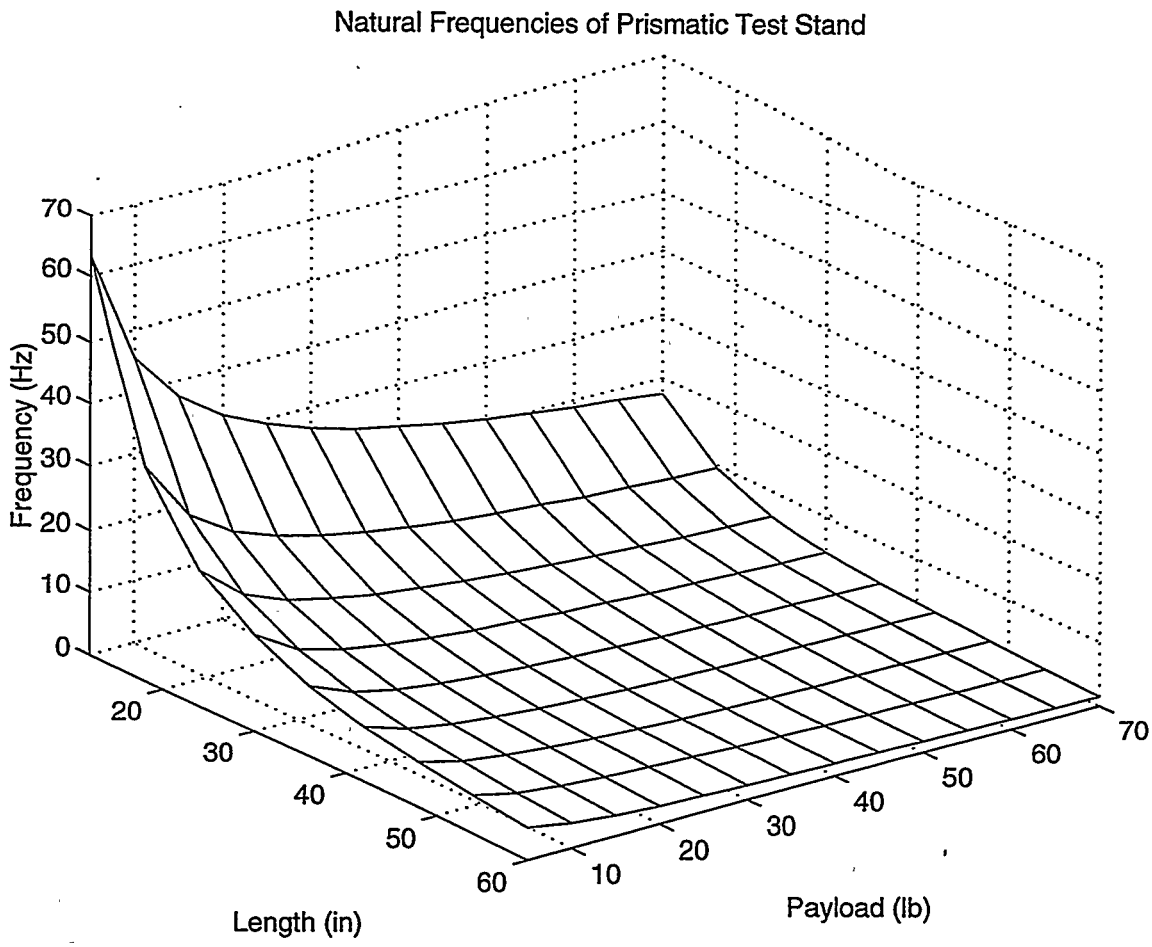


Fig. 21. Analytical determination of the natural frequency as a function of length and payload for the flexible/prismatic test bed.

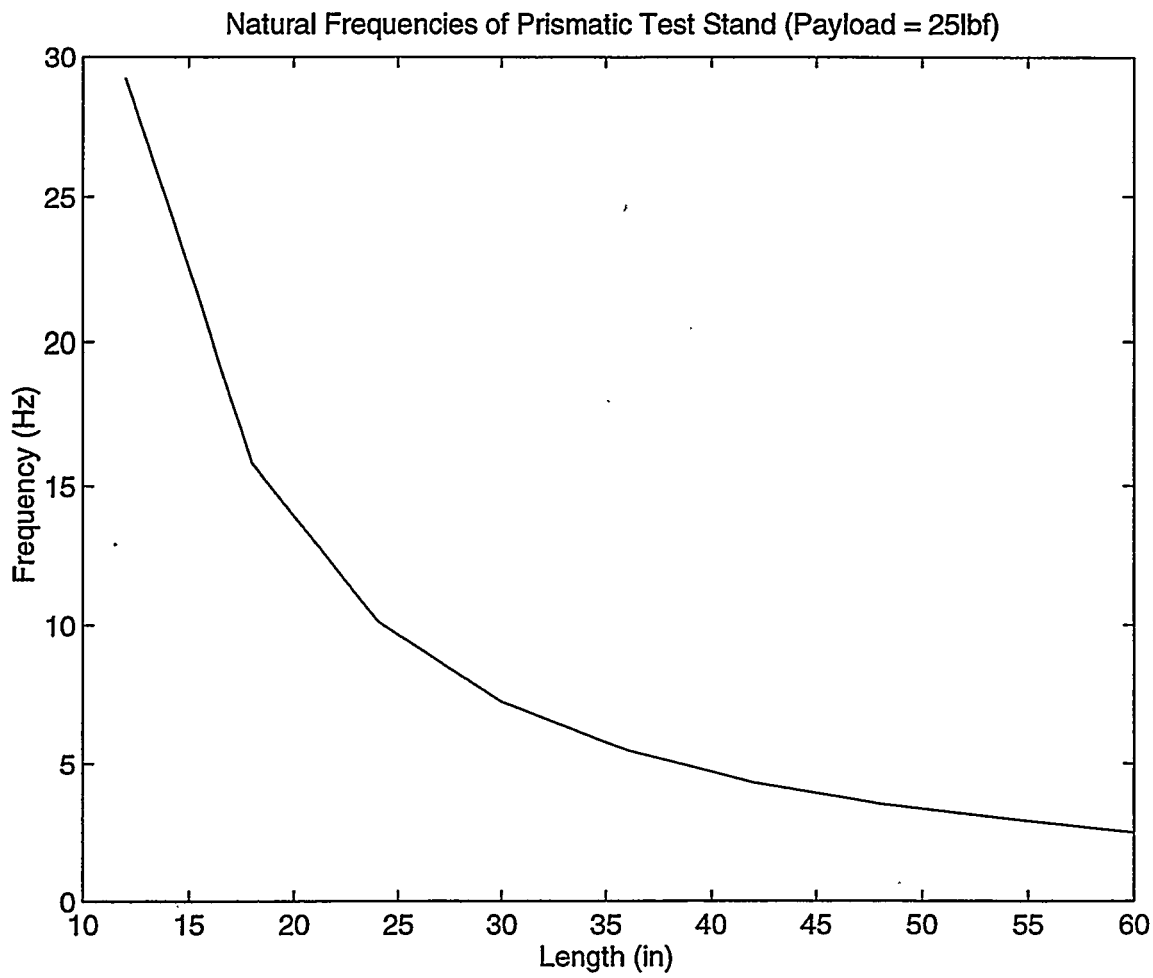


Fig. 22. Natural frequency as a function of length for 25-lb_f payload for the flexible/prismatic test bed.

3.1.2 Flexible/Prismatic Test Stand Sensors

Lateral effect photodiodes provide a relatively new approach to measuring link end point position. These systems mount a laser diode on one end of the flexible link and a detector on the other end. The detector can measure the relative displacement of the diode, in either a line or a plane, with respect to the detector. The detector selected for the flexible/prismatic test bed is a Graseby Model 272. This detector has a 9.9° field of view. Future experiments will focus on using this detector for both vibration measurement and control. The goal of the experiments is to establish the benefit of tip deformation or position measurements in the control of flexible/prismatic manipulators. Specifications for another type of end point sensor are presented in Appendix D.

3.2 FLEXIBLE/PRISMATIC TEST STAND TEMPERATURE VARIATION EXPERIMENTS

A set of experiments was performed to characterize the response of the flexible/prismatic test bed to extended operation. One of the primary goals was to determine the sensitivity of the plant dynamics to temperature increases encountered during system warm-up and extended operation. Figure 23 contains experimentally measured temperature data at the hydraulic reservoir and at the actuator supply (measured right at the servovalve). These data were taken from a single day-long experiment with the flexible/prismatic test bed.

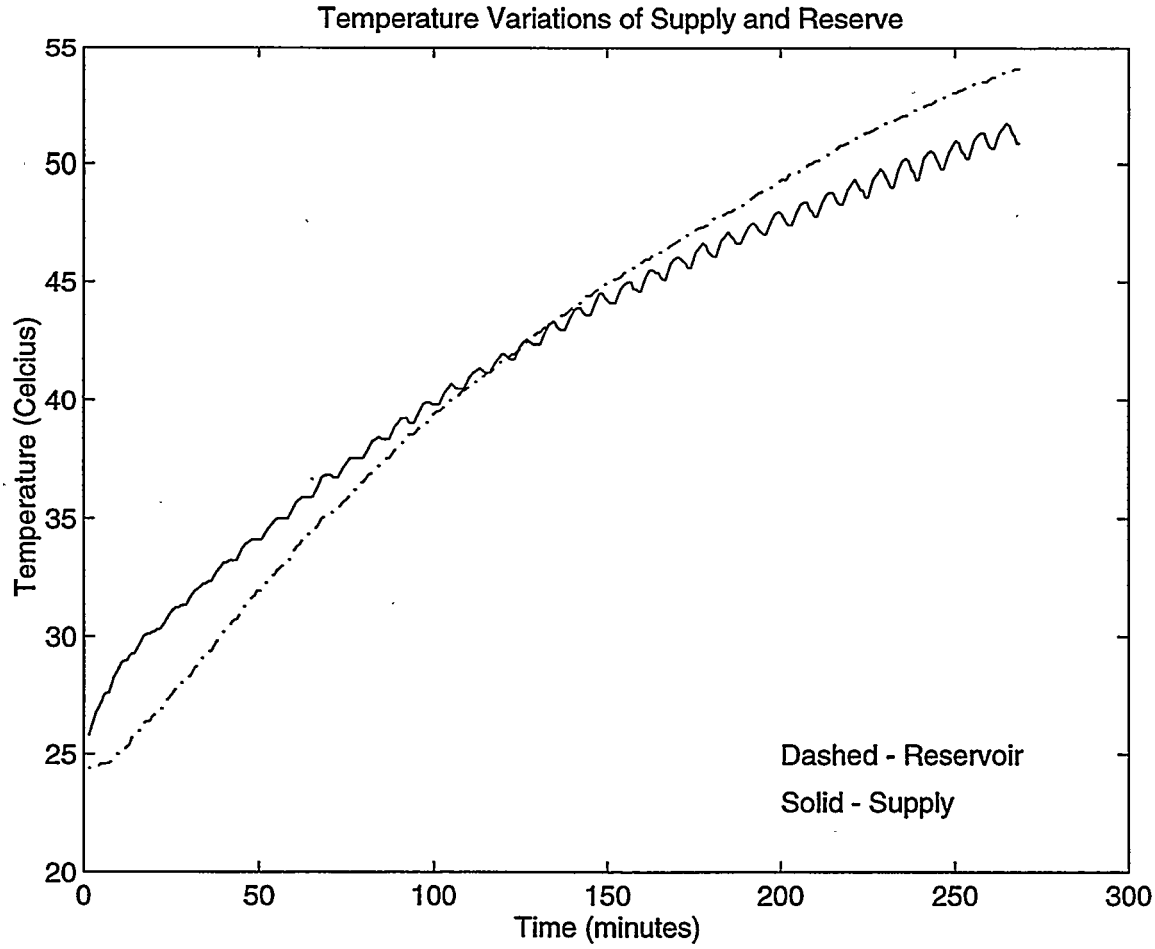


Fig. 23. Temperature vs time for a flexible/prismatic test bed extended operation experiment.

The figure shows the variation of reservoir and supply fluid temperature as a function of time. This increase in the temperature produced the effective variation in magnitude response evident in Figs. 24 and 25. The experiment consisted of running a consistent series of frequency response experiments. A program repetitively output 20 cycles of sine waves with frequencies of 0.4, 0.8, 1.0, 2.0, 5.0, 7.5, 10.0, and 12.5 Hz. During each cycle, the position response was measured, and the discrete fourier transform (DFT) of the frequency provided the corresponding magnitude and phase. Raw data for selected experiments are shown in Appendix C. The oscillation in the fluid temperature in the latter part of the experiment was due to a cooling effect in the hydraulic lines. The low-frequency moves are accompanied by significant motion of the actuator. As the frequencies of the commanded moves increase, the actuator response decreases in magnitude (refer to the Bode Plot of Fig. 24). This decrease in

actuator magnitude, which results from the natural inability of the dynamic system to track the high-frequency command, is associated with a concomitant decrease in the amount of fluid traveling through the lines from the reservoir to the valve and actuator. This lower flow is what causes the temperature to fall slowly since the fluid that rests in the hydraulic lines leading to the valve has time to cool and is not replaced by fresh, higher temperature fluid from the reservoir. Once the last high-frequency test of a particular series is completed, the lowest frequency test of the next set is conducted. The actuator can track this low-frequency command, resulting in large motion and large flow and accordingly, the temperature rises again. This explains the oscillating appearance of the supply temperature in Fig. 23.

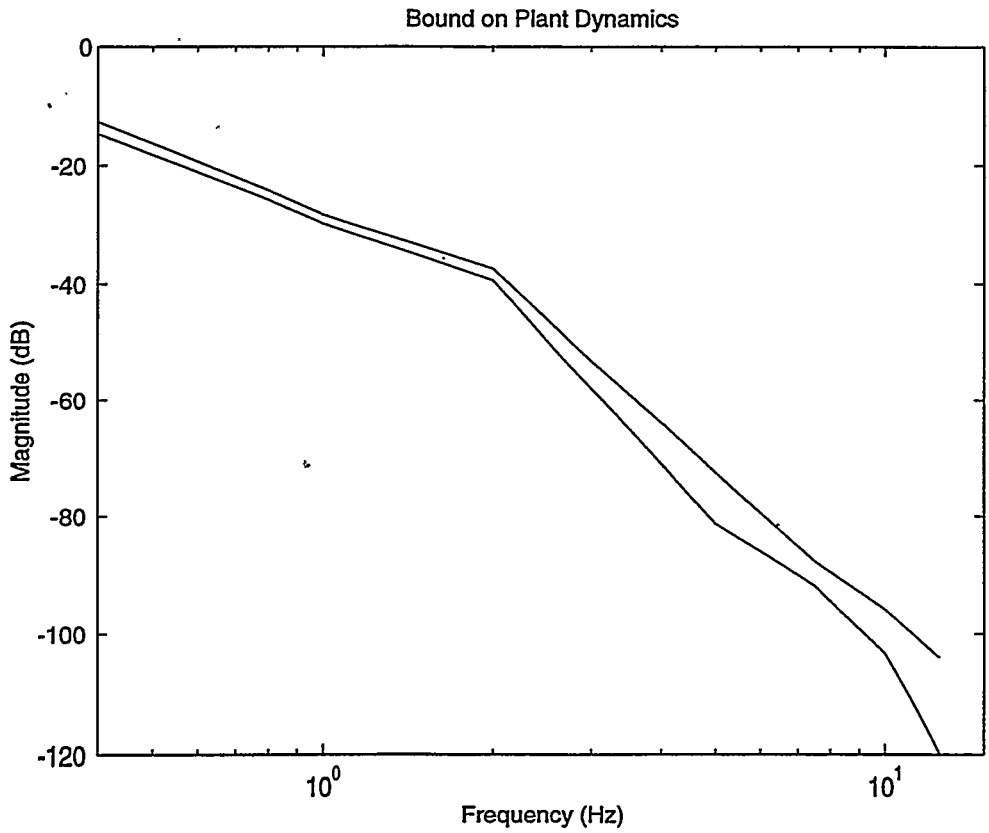


Fig. 24. Minimum and maximum plant models for the rotary joint of the flexible/prismatic test bed extended operation experiment of Fig. 23.

Figure 24 shows the variation of the rotary hydraulic cylinder's magnitude response as the temperature of the hydraulic fluid varied from 25°C to ~50°C. The variation in the magnitude response provides the motivation for robust controller designs.

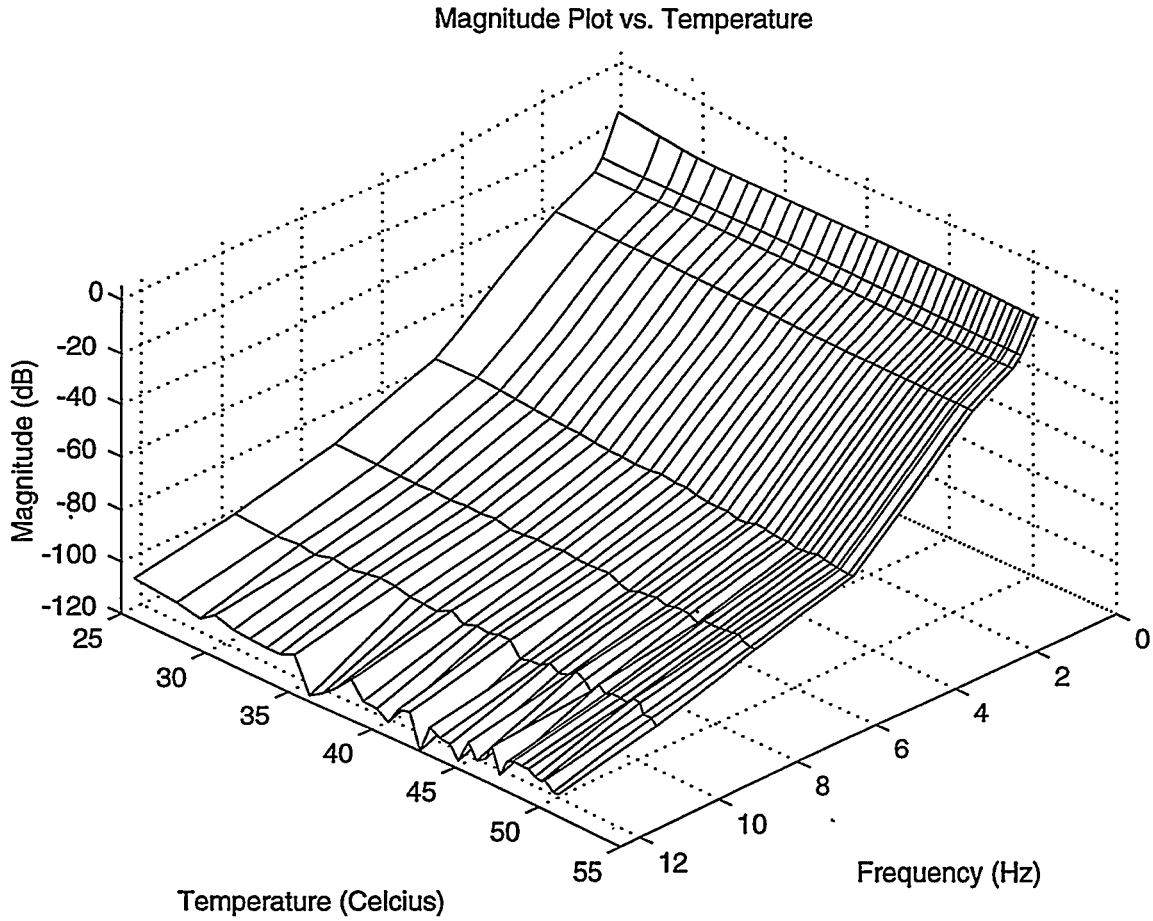


Fig. 25. Plant variation of the rotary hydraulic joint as a function of temperature for the flexible/prismatic test bed extended operation experiment of Figs. 23 and 24.

Figure 25 illustrates a three-dimensional plot of the magnitude response of the rotary hydraulic joint as a function of temperature and frequency. Appendix C includes the data and fits for selected temperature characterization experiments from this series of tests. Clearly, the main parameter that varied in the experiments is the magnitude of the response at low frequency. Referring to Eq. 2.4, one can see that the only parameter in the model that could vary as a function of temperature is the effective bulk modulus (also see Table 3). The conclusion is that the bulk modulus will decrease as a result of thermal expansion and that this change will reduce the stiffness and overall closed-loop bandwidth of a hydraulic servomechanism. When designing controllers for hydraulic manipulators, the variation of the plant with temperature (e.g., time) must be considered and appropriate measures taken. Cooling units can be added to the hydraulic power supply, and controllers should be formulated so that they are robust to temperature changes. In any event, adequate warm-up time should be allowed before operating, tuning, or calibrating a hydraulically actuated robot manipulator.

3.3 HUMAN EXTENDER/AMPLIFIER

The human extender/amplifier was experimentally evaluated in the laboratory and was shown to lift large loads with submillimeter accuracy. Force amplification ratios from 1 to 500 were achieved. More importantly, stable contact was achieved with environments of all stiffness ranging from free space, to foam, rubber and metal.

As the force amplification gain was increased to values above 100, it was observed that the noise level in the operator handle load cell (load cell 2) caused a noticeable random vibration. Typically, the auxiliary force reflection gain, α_F', was set to a value of 1, and the primary force reflecting gain, α_F, was set to the desired gain value. This was acceptable for total force reflecting gains (product of $1/\alpha_F'$ and α_F) under 100 where the handle load cell signal noise level did not cause a random vibration observable by the operator. If higher gains were required, a gain of 250 for example, α_F' would have to be lowered to 0.1 and α_F set to 25. This allowed higher operating ranges at the expense of lowering the effective actuator friction reduction.

Contact stability was tested by driving the system against various objects of different stiffness. This system was able to go beyond a stiffness of 2% of the load cell stiffness (up to 5%) by increasing the compensator gain. Once the noise problem was overcome using the auxiliary gain method, contact with surfaces of rubber to hard metal was easily obtained.

To test the force reflection of the system, weights of 25- and 35-lb_f increments were added to the system. A spring scale was placed at the handle. By varying the force reflection ratio, the spring scale static position would change. Accuracies to the limit of the spring scale resolution were observed.

3.4 GENERAL EXPERIMENTAL COMMENTS AND OBSERVATIONS

The use of environmentally safe fluids is appealing from the standpoint of safety and compliance. Environmentally safe fluids are difficult to control because of low viscosity and increased air entrapment.[21] ORNL has found Houghto-Safe® environmentally friendly fluid to be an acceptable compromise, and typical properties are provided in Table 4.

Table 4. Typical properties for Houghto-Safe® hydraulic fluid

Property	SI Units	English Units
Density	1072 kg/m^3	8.95 lb_m/gal (0.0387 $lb_m/in.^3$)
Specific gravity	1.074	1.074
Specific heat	2973 J/kg-°C	0.71 Btu/lb_m-°F
Thermal conductivity	0.45 W/m-°C	0.26 Btu/hr-ft-°F
Bulk modulus	1,965,000 N/m^2	285,000 $lb_f/in.^2$
Thermal expansion coefficient	0.00061 1/ °C	0.00034 1/ °F
Pour point	-54°C	-65°F
Viscosity (21°C or 70°F)	78 Centistokes	89 SUS
Refer to Table 2 for the complete viscosity data.		

4. FUTURE HYDRAULIC SYSTEMS

4.1 GUNITE AND ASSOCIATED TANKS

One of the largest problems facing DOE is the containment of hazardous waste stored in large underground nuclear waste storage facilities. Many of these facilities have contained hazardous materials for over 40 years. At the Hanford facility, there are 177 huge underground tanks of high-level nuclear waste, some of which have leaked or are building up heat or flammable gas.[22] Access to these tanks is limited to ports and risers, generally 40 in. in diameter, that have been used for monitoring and sampling of the waste. Because of the uncertain integrity of the tanks, it is desirable to establish a method of deploying robots in the tanks with minimal impact on the tank's structure. One option is to design a long, slender robot that can be deployed through the existing ports and risers.[23]

At ORNL, there are underground waste storage facilities as well. Fortunately, the number of tanks and their level of radiation is far below levels at the Hanford facility. A current project at the Robotics and Process Systems Division (RPSD) at ORNL consists of constructing a system that can go into an underground storage facility and safely extract waste for reprocessing and safer storage. This system consists of two long-reach manipulators deployed simultaneously in a tank. The first manipulator, the MLDUA, has 8-D.O.F. and a payload capacity of 250 lb. A second arm, the hose management arm (HMA), has 4-D.O.F. It carries a confined sluicing end-effector (CSEE) that acts as a cutting jet and vacuum for waste removal. The strategy is to have the MLDUA grasp the CSEE and move through the tank, extracting material. Figure 26 illustrates a conceptual image of the gunite tanks and the associated hardware. (Gunite is a concrete-like material.)

4.2 APPLICATION OF DAWM TO THE CP-5 REACTOR AT ANL

The most pressing application of the techniques described herein are for the accurate control of the DAWM described in the Introduction. This system is being deployed in the CP-5 reactor at ANL. The basic degrees-of-freedom of the DAWM arms are all hydraulically actuated and must be operated with some limited robotics capabilities. Accurate control of these joints will require a complete understanding of the hydraulic issues discussed in this report.

4.3 SCHILLING 7F

The 6-D.O.F., multiplaner, teleoperated, flexible controls Schilling 7F test bed has been transferred to the Cold Test Facility (CTF) at ORNL. This facility is a mockup of the ORNL Gunite tanks.[24] It includes a real control room, a mock tank, mock waste, vision systems, and other support hardware. The Schilling 7F test bed is being used for control system integration prior to receiving the Modified Light Duty Utility Arm (MLDUA)[25] that is to be inserted into the CTF and eventually into the Gunite tanks. One of the first tasks is to integrate path planning and control routines for waste cleanup and to integrate T. J. Tarn's newly developed event-based controllers.[26]

4.4 DEXTEROUS MANIPULATION OF HEAVY PAYLOADS

Dexterous manipulation of heavy payloads (DMHP) is a new application area for hydraulic manipulators. Basically, DMHP is defined as systems with payload/weight ratios greater than 1 and positioning accuracy/payload ratios less than 10^{-6} m/kg. These systems appear in nature as insects and as large animals (e.g., rhinoceroses and elephants) but they have no parallel in the machine world. Present manipulator systems have precision accuracy/payload ratios that range around 10^{-2} m/kg for electric robots to 10^{-1} m/kg for conventional machines and earthmoving equipment. Aiming for a goal of 10^{-6} m/kg for a DMHP machine will require careful analysis and design of hydraulic control systems. The design and modeling techniques developed herein will contribute to this goal.

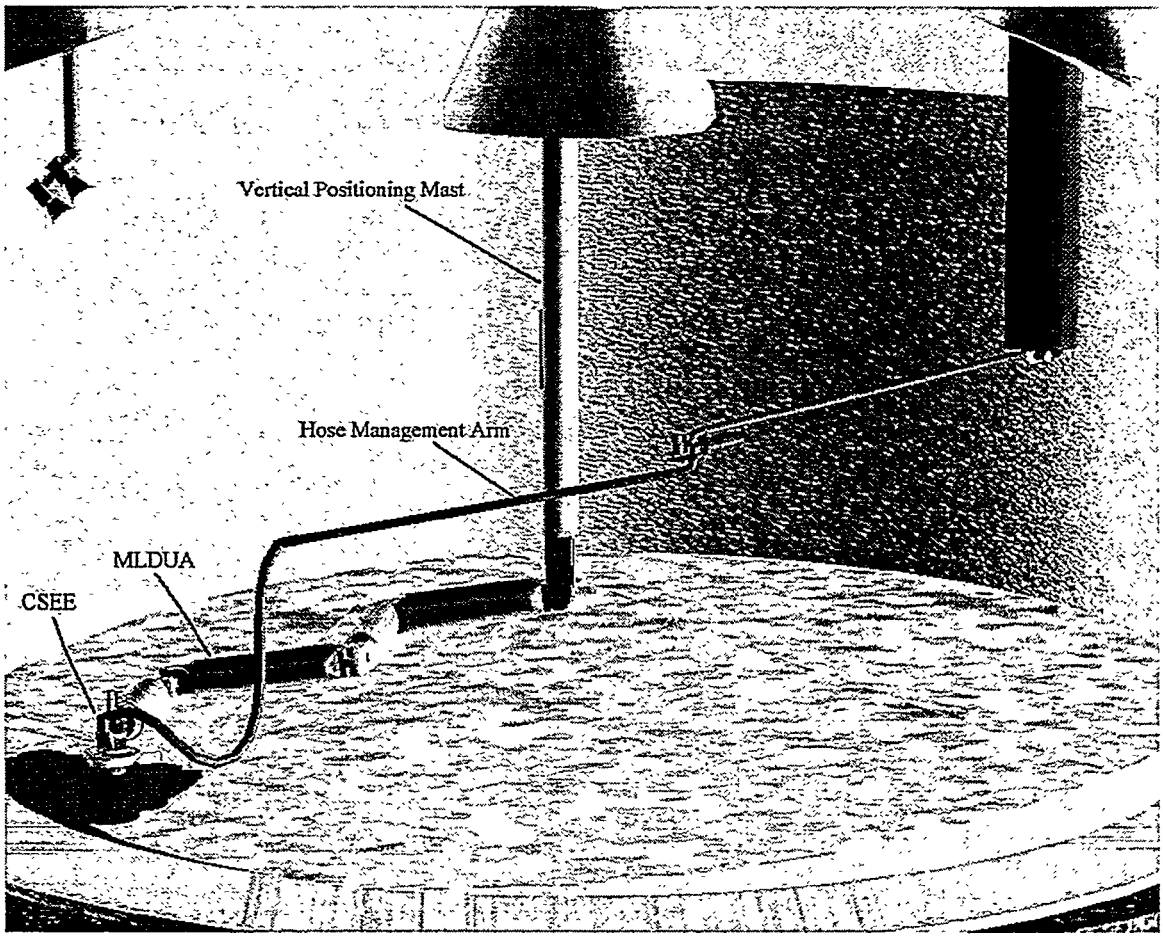


Fig. 26. Tank waste removal hardware for the GAAT cleanup effort at ORNL.

5. CONCLUSIONS

ORNL has extensive experience in the design, analysis, and control of hydraulic manipulators. ORNL has over a decade of experience in developing hydraulic systems for a number of applications including mobile robotics, robotic arms, and new concept systems. To support the development and deployment of the new hydraulic manipulators, ORNL has outfitted a significant experimental laboratory and has developed the software capability for research into hydraulic manipulators, hydraulic actuators, hydraulic systems, modeling of hydraulic systems, and hydraulic controls. The three manipulator systems in the Hydraulics Laboratory at ORNL and the hydraulics modeling capability have been used to extend ORNL's experience and hardware capability in hydraulics. The 6-D.O.F., multiplaner, teleoperated, flexible controls test bed has been used for basic controller development and as a systems integration tool for the gunite tank efforts. The 2-D.O.F. flexible/prismatic test bed has been used for the development of waste tank cleanup manipulator controls, thermal studies, system characterization, and manipulator tracking. One of the primary sets of tests with the 2-D.O.F. flexible/prismatic test bed was the extended operational experiments where the hydraulic system model was determined as a function of time (temperature). Finally, the human amplifier test bed was used for the development of an entirely new class of teleoperated systems, including not only amplifier systems but also deamplifier systems as well. These experiences and the existing hardware and software infrastructure give ORNL an unmatched capability in hydraulic manipulator design and analysis.

REFERENCES

1. T. Takamori, "Recent Trends in the Development of New Actuators," *J. of Robotics and Mechatronics*, 31, pp. 18-27.

2. H. E. Merritt, *Hydraulic Control Systems*, John Wiley and Sons, New York, 1967.

3. B. L. Burks, B. S. Richardson, G. A. Armstrong, W. R. Hamel, J. F. Jansen, S. M. Killough, and D. H. Thompson, "A Demonstration of Remote Survey and Characterization of a Buried Waste Site Using the SRIP Test bed," pp. 179-185 in *Proceedings of the Fourth ANS Topical Meeting on Robotics and Remote Systems*, Feb. 24-28, 1991, Albuquerque, New Mex.

4. R. L. Kress, H. M. Costello, R. L. Glassell, and F. Eldredge, "Control System Design for an Automated Shell Handling Vehicle," pp. 683-691 in *Proceedings of the Fourth ANS Topical Meeting on Robotics and Remote Systems*, Feb. 24-28, 1991, Albuquerque, New Mex.

5. R. L. Kress, H. M. Costello, R. L. Glassell, and S. March-Leuba, "Future Armor Rearm Control System Design," *Robotics and Autonomous Systems*, Special Issue on "Robotics and Intelligent Systems at Oak Ridge National Laboratory," Vol. 10, (1-2) 185-197 (1992).

6. D. H. Thompson, B. L. Burks, and S. M. Killough, "Remote Excavation Using the Telerobotic Small Emplacement Excavator," pp. 465-469 in *Proceedings of the Fifth ANS Topical Meeting on Robotics and Remote Systems*, April 26-29, 1993, Knoxville, Tenn.

7. B. L. Burks, S. M. Killough, D. H. Thompson, and M. A. Dinkins, *Explosive Ordinance Disposal Technology Demonstration Using the Telerobotic Small Emplacement Excavator*, ORNL/TM-12770, Oak Ridge National Laboratory, June 1994.

8. M. W. Noakes, W. R. Hamel, and W. E. Dixon, "Application of a Selective Equipment Removal System to D&D Tasks," pp. 115-122 in *Proceedings of the Sixth ANS Topical Meeting on Robotics and Remote Systems*, Feb. 5-10, 1995, Monterey, Calif.

9. Chan, T. F., Dubey, R. V., and Noakes, M. W., "Generalized Bilateral Control for Dissimilar Kinematic Teleoperators and Application to D&D Type Tasks," pp. 281-284 in *Proceedings of the Sixth ANS Topical Meeting on Robotics and Remote Systems*, Feb. 5-10, 1995, Monterey, Calif.

10. R. L. Kress, M. W. Noakes, and J. F. Jansen, "The Evolution of Telerobotic Manipulators at Oak Ridge National Laboratory," to appear in the *Proceedings of the Seventh ANS Topical Meeting on Robotics and Remote Systems.*

11. L. Conley, W. R. Hamel, M. W. Noakes, and B. R. Thompson, "Rosie: A Mobile Worksystem for Decontamination and Dismantlement Operations," pp. 231-238 in *Proceedings of the Sixth ANS Topical Meeting on Robotics and Remote Systems*, Feb. 5-10, 1995, Monterey, Calif.

12. D. D. Falter, S. M. Babcock, B. L. Burks, P. D. Lloyd, J. D. Randolph, and J. E. Rutenber, "Remote Systems for Waste Retrieval from the Oak Ridge National Laboratory Gunite Tanks," *Proceedings of the ANS 1995 Winter Meeting*, Oct. 29-Nov. 2, 1995, San Francisco, Calif.

13. E. G. Johnsen, and W. R. Corliss, Teleoperators and Human Augmentation, NASA SP-5047, NASA Office of Technology Utilization, December 1967, Washington, D.C.

14. H. Kazerooni, "Human/Robot Interaction via the Transfer of Power and Information, Part I: Dynamics and Control Analysis," pp. 1632-1640 in Proceedings of the IEEE Robotic and Automation Conference, Scottsdale, Ariz., 1989a.

15. H. Kazerooni, "Human/Robot Interaction via the Transfer of Power and Information, Part 2: An Experimental Analysis," pp. 1641-1647 in Proceedings of the IEEE Robotic and Automation Conference, Scottsdale, Ariz., 1989b.

16. H. Kazerooni and J. Guo, "Human Extenders", Journal of Dynamic Systems, Measurement, and Control, Vol. 115., pp. 281-290 (June 1993).

17. B. Hannaford, "A Design Framework for Teleoperators with Kinesthetic Feedback", Proceedings of the IEEE Transactions on Robotics and Automation, Vol., 5 (4), pp. 426-434, (August 1989).

18. R. Volpe, and P. Khosla, "A Theoretical and Experimental Investigation of Impact Control for Manipulators", The International Journal of Robotics Research, Vol. 12 (4), pp. 351-365, (August 1993).

19. A. Gelb, and W. E. V. Velde, Multiple-Input Describing Functions and Nonlinear Systems Design, McGraw Hill, New York, 1968.

20. M. Vidyasagar, Nonlinear Systems Analysis, Prentice Hall, Inc., New Jersey, 1978.

21. E. Trostmann, Water Hydraulics Control Technology, Marcel Dekker, Inc., New York, 1996.

22. G. Zorpette, 1996, "Hanford's Nuclear Wasteland," Scientific American, pp. 88-97, (May 1996).

23. K. Morris, A. Shattuck, and W. Covert, "Selected In-Tank Property Measurement Methods for Hanford Site Single-Shell Tanks," *Proceedings of the ANS Winter Annual Meeting*, Washington, D.C., 1990.

24. D. Van Hoesen, F. Bzorgi, A. Kelsey, C. Wiles, and K. Beeson, "Underground Radioactive Waste Tank Remote Inspection and Sampling," DOE EM Fact Sheet, Gunite and Associated Tanks Operable Unit at Waste Area Grouping 1 at Oak Ridge National Laboratory, The Department of Energy Environmental Program, Winter 1995.

25. H. Jones, B. Rubinger, P. Apperly, G. Battiston, Y. Fillion, and B. Fuller, "Development and Integration of the Light Duty Arm System," *Proceedings of the ANS 1996 Winter Meeting*, Feb. 25-29, 1996, Tucson, Ariz.

26. C. Guo, T-J. Tarn, N. Xi, and A. K. Bejczy, "Fusion of Human and Machine Intelligence for Telerobotic Systems," pp. 3110-3115 in Proceedings of 1995 IEEE Int. Conf. on Robotics and Automation, May 21-27, 1995, Nagoya, Japan.

APPENDIX A

LIST OF SYMBOLS

$A \equiv$ effective cylinder area (in.^{2})

$A_p \equiv$ amplitude (in.) for sinusoidal input functions

$B(s) \equiv \frac{K_p + C_{tp}}{A} + \frac{V_T}{4\,\beta_e\,A}\,s$ transfer function defined for compactness

$B_{LC1} \equiv$ bias voltage for load cell 1 (V)

$B_{LC2} \equiv$ bias voltage for load cell 2 (V)

$B_m \equiv$ motor viscous damping (in.-lb$_f$-s)

$B_p \equiv$ piston viscous damping (lb$_f$/in./s)

$C_{tm} \equiv$ total leakage coefficient for a rotary motor (in.^{3}/s/psi)

$C_{tp} \equiv$ total leakage coefficient for a piston (in.^{3}/s/psi)

$D_m \equiv$ volumetric displacement for a rotary motor (in.^{3}/rad)

$d/dt \equiv$ time derivative

E or e $\equiv$ error signal (V)

$F_{act} \equiv$ actuator force (lb$_f$)

$F_{frict} \equiv$ total friction force (lb$_f$)

$F_{frict}^{act} \equiv$ actuator friction (lb$_f$)

$F_{frict}^{ee} \equiv$ friction of carriage (lb$_f$)

$F_{grav} \equiv$ gravity force (lb$_f$)

$F_{hand} \equiv$ hand force (lb$_f$)

$F_{LC1} \equiv$ load cell 1 force (lb$_f$)

$F_{LC2} \equiv$ load cell 2 force (lb$_f$)

$F_{Load} \equiv$ load force (lb$_f$)

$g(jw) \equiv$ circle criterion transfer function

$G_1(s)$, $G_2(s) \equiv$ arbitrary transfer functions

$G_c(s) \equiv$ compensator transfer function

$G_v(s) \equiv$ linear transfer function modeling relationship between Operator and human extender/amplifier

$I \equiv$ electrical current (A)

$J_T \equiv$ motor and load interia reflected at the motor shaft (in.-lb$_f$-s^2)

$K_p \equiv$ valve flow pressure coefficient (in.^{3}/s/psi)

$K_{PTm} \equiv$ total flow pressure coefficient for a rotary motor (in.^{3}/s/psi)

$K_{PTp} \equiv$ total flow pressure coefficient for a piston (in.^{3}/s/psi)

$K_q \equiv$ valve flow gain with current as command (in.^{3}/s/A)

$K_{ql} \equiv$ valve flow gain with valve position as command (in.^{3}/s/in.)

$K_s \equiv$ environment stiffness (lb_f/in.)
K_{visc} = viscous coefficient (lb_f/in./s)
$M \equiv$ mass of load (lb_m)
$N_{LC1} \equiv$ noise voltage of load cell 1 (V)
$N_{LC2} \equiv$ noise voltage of load cell 2 (V)
$NLK_s \equiv$ nonlinear spring (lb_f/in.)
$NLK_v \equiv$ nonlinear block (lb_f/in./s)
$P_L \equiv$ load pressure (psi)
$Q_L \equiv$ effective flow (in.^{3}/s)
$R \equiv$ electrical resistance (Ω)
$s \equiv$ LaPlace transform variable
$T \equiv$ temperature (°F)
$T_o \equiv$ nominal temperature (°F)
$V_T \equiv$ total volume (in.^{3})
$V \equiv$ voltage (V)
$x_v \equiv$ valve position (in.)
Y or $y \equiv$ actuator position (in.)
$\dot{Y}$ or $\dot{y} \equiv$ actuator velocity (in./s)
$Z \equiv$ modified load cell force (lb_f)

$\alpha \equiv$ cubical expansion coefficient (1/°F)
$\alpha_F \equiv$ force gain
$\alpha_F' \equiv$ auxiliary force gain
$\beta \equiv$ bulk modulus (psi)
$\beta_e \equiv$ effective bulk modulus (psi)
$\theta_m \equiv$ rotary motor position (rad)
$\lambda \equiv$ viscosity temperature constant (1/°F)
$\rho \equiv$ density (lb_m/ft^3)
$\rho_o \equiv$ density at nominal condition (lb_m/ft^3)
$\mu \equiv$ absolute viscosity (lb_f-s/in.^{2})
$\mu_o \equiv$ absolute viscosity at T_o (lb_f-s/in.^{2})

APPENDIX B

GOVERNING EQUATIONS FOR HYDRAULIC DYNAMICS

B.1 INTRODUCTION

A typical linear actuator system is shown in Fig. B.1. It is basically a three-land four-way critical center spool valve that controls the flow of pressurized oil to and from the power cylinder. The power cylinder is solidly connected to a purely inertial load whose motion is opposed by nonlinear friction forces. In the case of the rotary actuator, the piston will have rotary motion, as opposed to linear motion and the rotary motor would generate a torque instead of a force.

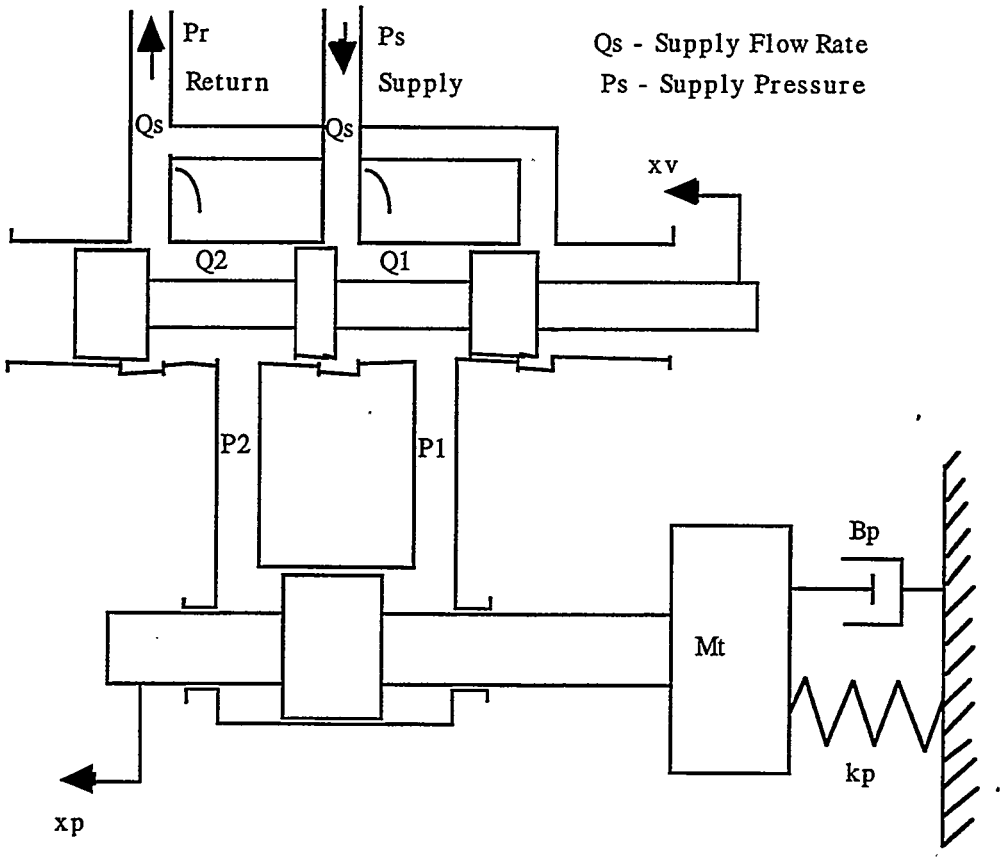


Fig. B.1. Model of hydraulic piston used for explanation of governing equations.

B.2 VALVE EQUATIONS

To develop the combined servovalve and actuator model, we will consider two cases: one where the servovalve spool position x_v is positive and the other where x_v is negative.

Case I: Extension Stroke ($x_v \geq 0$)

Assuming that there is symmetry in the construction of the servovalve and orifice geometries, the flow rates Q_1 and Q_2 are

$$Q_1 = C_d A \sqrt{\frac{2}{\rho}(P_s - P_1)}$$

$$Q_2 = C_d A \sqrt{\frac{2}{\rho}(P_2 - P_r)} \quad , \qquad \text{(B.1a \& B.1b)}$$

where C_d is the discharge coefficient (as defined by Merritt[2]) ρ is the fluid mass density, and A is the orifice area. With the assumption of linear valve displacement characteristics, Eqs. (B.1a) and (B.1b) can be rewritten in terms of spool position, x_v, as

$$Q_1 = C_d' x_v \sqrt{(P_s - P_1)}$$

$$Q_2 = C_d' x_v \sqrt{(P_2 - P_r)} \quad , \qquad \text{(B.2a \& B.2b)}$$

which are normalized to yield

$$q_1 = C_{deff} x_{vnorm} \sqrt{(1 - p_1)}$$

$$q_2 = C_{deff} x_{vnorm} \sqrt{(p_2 - p_r)}, \qquad \text{(B.3a \& B.3b)}$$

where

$$q_1 = \frac{Q_1}{Q_s}, \quad q_2 = \frac{Q_2}{Q_s}, \quad p_1 = \frac{P_1}{P_s}, \quad p_2 = \frac{P_2}{P_s}, \quad p_r = \frac{P_r}{P_s},$$

$$x_{vnorm} = \frac{x_v}{x_{v\max}}, \quad C_{deff} = C_d' \frac{\sqrt{P_s}}{Q_s} x_{v\max}.$$

Case II: Retraction Stroke ($x_v < 0$)

In this case, Eqs. (B.3a) and (B.3b) become

$$q_1 = C_{deff} x_{vnorm} \sqrt{(p_1 - p_r)}$$

$$q_2 = C_{deff} x_{vnorm} \sqrt{(1 - p_2)}. \qquad \text{(B.4a \& B.4b)}$$

In Eqs. (B.3) and (B.4), the sign of x_v determines the direction of flow. The flow saturation effect can be modeled as

$$x_{vnorm} = \begin{cases} x_{vnorm} & for \quad |x_{vnorm}| \langle 1 \\ x_{v\max} & for \quad |x_{vnorm}| \geq 1 \end{cases} . \qquad \text{(B.5)}$$

The servovalve dynamics are modeled as

$$\frac{d}{dt}\left(\frac{x_v}{x_{v\max}}\right) = \frac{x_{v\max}}{\tau}\left(\frac{x_{vi}}{x_{v\max}} - \frac{x_v}{x_{v\max}}\right), \qquad \text{(B.6)}$$

where x_{vi}, x_v are the desired and actual servovalve positions respectively.

B.3 LINEAR ACTUATOR EQUATIONS

The continuity equation to each piston chamber can be written as:

$$\dot{P}_1 = \frac{\beta_e}{V_1}\left[Q_1 - A_{p1}\dot{x}_p - C_{ip}(P_1 - P_2) - C_{ep}P_1\right]$$

$$\dot{P}_2 = \frac{\beta_e}{V_2}\left[-Q_2 - A_{p2}\dot{x}_p + C_{ip}(P_1 - P_2) - C_{ep}P_2\right], \qquad \text{(B.7a \& B.7b)}$$

where

V_1 = volume of forward chamber (includes valve, connecting line, and piston volume), in.^{3};

V_2 = volume of return chamber (includes valve, connecting lines, and piston volume), in.^{3};

C_{ip} = internal or cross-port leakage coefficient of piston, in^3/s-psi;

A_{p1} = area of piston on side 1, in.^{2};

A_{p2} = area of piston on side 2, in.^{2};

x_p = displacement of piston, in.

The volumes of the piston chambers may be written

$$V_1 = V_{01} + A_{p1}x_p$$

$$V_2 = V_{02} - A_{p2}x_p,$$

where V_{01} = is the initial volume of forward chamber in.^{3} and V_{02} = is the initial volume of return chamber, in.^{3}.

The Eqs. (B.7a) and (B.7b) are normalized to yield

$$\dot{p}_1 = \frac{\beta_e}{V_1}\left[q_1\left(\frac{Q_s}{P_s}\right) - \left(\frac{A_{p1}}{P_s}\right)\dot{x}_p - C_{ip}(p_1 - p_2) - C_{ep}p_1\right]$$

$$\dot{p}_2 = -\frac{\beta_e}{V_2}\left[q_2\left(\frac{Q_s}{P_s}\right) - \left(\frac{A_{p2}}{P_s}\right)\dot{x}_p - C_{ip}(p_1 - p_2) + C_{ep}p_2\right], \qquad \text{(B.8a \& B.8b)}$$

where $p's$ and $q's$ are as defined above.

The force generated or developed by the piston is

$$F_{gen} = A_{p1}P_1 - A_{p2}P_2 \ . \qquad \text{(B.9)}$$

The nonlinear friction model can be represented as shown in Fig. B.2, where theoretically, the switchover from the static friction F_S to dynamic friction F_C is considered to occur at $|\dot{x}_p| = 0^+$. However, this causes some instability when the model is to be solved by numerical techniques. Therefore, the switchover is assumed to take place at a certain velocity $|\dot{x}_p| = Dv$, which is chosen in order to give sufficient numerical stability.

The friction force F_f can be modeled as follows:

$$F_f = \begin{cases} F_{gen} & for \ |F_{gen}| \langle F_s \ and \ |\dot{x}_p| \le Dv \\ F_S sign(F_{gen}) & for \ |F_{gen}| \rangle F_s \ and \ |\dot{x}_p| \le Dv \\ F_C sign(F_{gen}) & for \ |\dot{x}_p| \rangle Dv \end{cases}, \qquad \text{(B.10)}$$

which is shown in Fig. B.2.

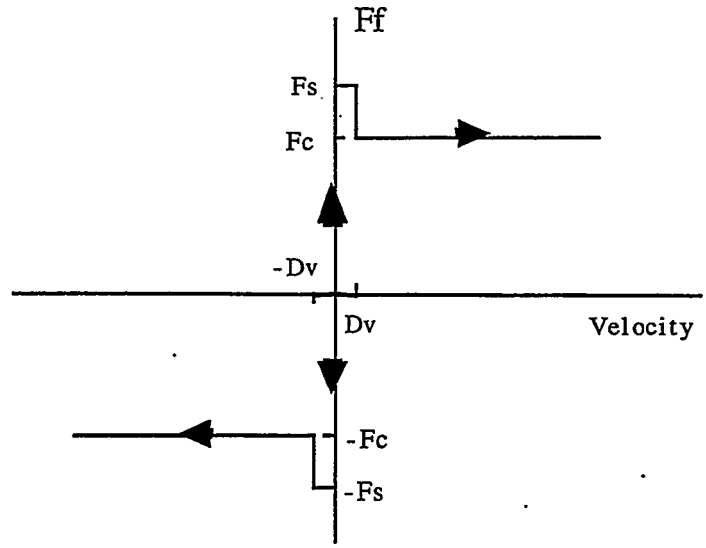


Fig. B.2. Model for nonlinear friction, including stiction and Coulomb friction.

Assuming inertia load and nonlinear friction, the force equation can be described by

$$F_{gen} = M_t \ddot{x}_p + B_p \dot{x}_p + F_f \quad , \tag{B.11}$$

where M_t is the total mass of piston and load referred to piston. (lb-s^2/in.) and B_p is the viscous damping coefficient of piston and load (lb$_f$-s/in).

For a bore diameter of D (in.), a piston rod diameter of d (in.), and a friction factor F_P, the coulomb friction force may be obtained from

$$F_C = 12d + 30F_P d + 6F_P D \quad . \tag{B.12}$$

The stiction force is calculated by applying a correction factor of F_{corr} to friction factor F_P. Hence the stiction force equation becomes

$$F_S = 12d + 30F_{corr}F_P d + 6F_{corr}F_P D \quad . \tag{B.13}$$

The correction factor for low-friction-type seals used in this actuator is 1.0.

B.4 ROTARY ACTUATOR EQUATIONS

The same servovalve models apply in the case of the rotary actuator. The continuity equations for piston motors are given by

$$\dot{p}_1 = \frac{\beta_e}{V_1}\left[q_1\left(\frac{Q_s}{P_s}\right) - \left(\frac{D_m}{P_s}\right)\omega_m - C_{im}(p_1 - p_2) - C_{em}p_1\right]$$

$$\dot{p}_2 = -\frac{\beta_e}{V_2}\left[q_2\left(\frac{Q_s}{P_s}\right) - \left(\frac{D_m}{P_s}\right)\omega_m - C_{im}(p_1 - p_2) + C_{em}p_2\right], \qquad \text{(B.14a \& B.14b)}$$

where D_m is the volumetric displacement of the motor (in.^{3}/rad) and ω_m is the angular velocity of the motor shaft (rad/s) and other variables are similar to those defined in the linear actuator case.

The two volumes may be expressed by

$$V_1 = V_0 + D_m\theta_m$$

$$V_2 = V_0 - D_m\theta_m,$$

where θ_m is the angular displacement of the motor. The torque generated by the motor becomes

$$T_{gen} = D_m(P_1 - P_2) \quad . \qquad \text{(B.15)}$$

The load equation can be written as

$$T_{gen} = J_t\ddot{\theta}_m + B_m\dot{\theta}_m + T_f \quad , \qquad \text{(B.16)}$$

where J_t is the load inertia, B_m is the viscous friction, and T_f is the nonlinear friction torque.

A similar model, as in the case of linear actuator, is assumed for the nonlinear friction torque. Nonlinear friction torques T_C and T_S are assumed equal and are calculated from the given breakaway force.

APPENDIX C

RAW DATA FOR SELECTED EXPERIMENTS FROM THE EXTENDED OPERATION EXPERIMENTS WITH THE FLEXIBLE/PRISMATIC TEST BED

Table C.1. Raw data from temperature variation study for the rotary actuator of the flexible/prismatic test bed
Column 1 is the input sine wave frequency (~1-inch position command), column 2 is the supply temperature, column 3 is the reservoir temperature, column 4 is the magnitude, and column 5 is the phase of the Bode plot model.

Freq (Hz)	Sup. Temp(°C)	Res. Temp(°C)	Mag.	Phase (deg)
0.4000	25.8000	24.4000	0.5371	-65.1316
0.8000	26.3000	24.4000	0.2976	-87.2921
1.0000	26.8000	24.5000	0.2429	-94.2214
2.0000	27.0000	24.5000	0.1542	-120.8883
5.0000	27.2000	24.6000	0.0266	-167.2850
7.5000	27.5000	24.6000	0.0123	169.8520
10.0000	27.6000	24.6000	0.0080	140.7889
12.5000	27.6000	24.6000	0.0055	125.8120
0.4000	29.7000	26.0000	0.5242	-66.0947
0.8000	30.0000	26.3000	0.2909	-87.9998
1.0000	30.1000	26.4000	0.2372	-94.3185
2.0000	30.1000	26.4000	0.1498	-121.0868
5.0000	30.2000	26.6000	0.0252	-168.4509
7.5000	30.2000	26.7000	0.0119	168.1162
10.0000	30.3000	26.8000	0.0078	139.5006
12.5000	30.3000	26.8000	0.0045	127.5730
0.4000	34.5000	32.3000	0.5114	-66.6169
0.8000	34.7000	32.5000	0.2828	-88.0043
1.0000	34.9000	32.6000	0.2311	-94.1931
2.0000	35.0000	32.8000	0.1483	-120.6683
5.0000	35.0000	32.9000	0.0224	-169.8434
7.5000	35.0000	33.0000	0.0120	170.0440
10.0000	35.0000	33.1000	0.0083	141.6796
12.5000	35.0000	33.1000	0.0047	130.6471
0.4000	40.3000	39.7000	0.4991	-67.3725
0.8000	40.5000	39.9000	0.2778	-88.1155
1.0000	40.7000	40.0000	0.2284	-93.9358
2.0000	40.6000	40.0000	0.1494	-120.5877
5.0000	40.5000	40.1000	0.0210	-169.9951
7.5000	40.5000	40.2000	0.0117	170.7045
10.000	40.5000	40.4000	0.0078	143.4931
12.5000	40.5000	40.4000	0.0039	134.5381
0.4000	45.2000	46.0000	0.4928	-67.9742
0.8000	45.5000	46.0000	0.2765	-88.1163
1.0000	45.5000	46.2000	0.2311	-93.6323
2.0000	45.4000	46.2000	0.1462	-120.1499

Table C.1 (cont.) Raw data from temperature variation study for the rotary actuator of the flexible/prismatic test bed

Column 1 is the input sine wave frequency (~1-inch position command), column 2 is the supply temperature, column 3 is the reservoir temperature, column 4 is the magnitude, and column 5 is the phase of the Bode plot model.

5.0000	45.4000	46.3000	0.0199	-169.1051
7.5000	45.2000	46.3000	0.0115	171.9688
10.0000	45.1000	46.4000	0.0066	146.5064
12.5000	45.1000	46.4000	0.0029	136.8753
0.4000	49.8000	51.8000	0.4865	-67.9928
0.8000	50.1000	51.9000	0.2776	-87.7229
1.0000	50.2000	51.9000	0.2275	-93.7230
2.0000	50.1000	52.0000	0.1409	-119.6014
5.0000	49.8000	52.0000	0.0180	-168.1353
7.5000	49.7000	52.1000	0.0115	172.5459
10.0000	49.4000	52.2000	0.0080	147.8011
12.5000	49.4000	52.2000	0.0042	141.0822

APPENDIX D

END POINT POSITION SENSOR DATA FOR FLEXIBLE/PRISMATIC TEST BED

This appendix includes some data on one commercially available 6-D.O.F. position sensor that might be applicable to the flexible/prismatic test stand.
Vendor: Ascension Technology
Product Name: Flock of Birds
Description: This is a 6-D.O.F. position sensor that is based upon a pulsed DC magnetic field. There is one transmitter (approximately a 12 inch cube) and up to 60 receivers (approximately a 1 inch cube). For example, one can track up to 60 of these sensors (relative to the big cube). Because of the pulsed field, the sensor is somewhat sensitive to metallic objects (it receives some interference due to eddy currents), but it is not as bad as ac field technology (e.g., Polhemus sensors).
Specifications:

Range: ± 3 ft. (translation) ± 180° (azimuth, roll) ± 90° (pitch)
Accuracy: 0.1 in. RMS (translation) 0.5 ° (rotation) Resolution: 0.03 in. (translation) 0.1°(rotation)
Cost: ~$2500

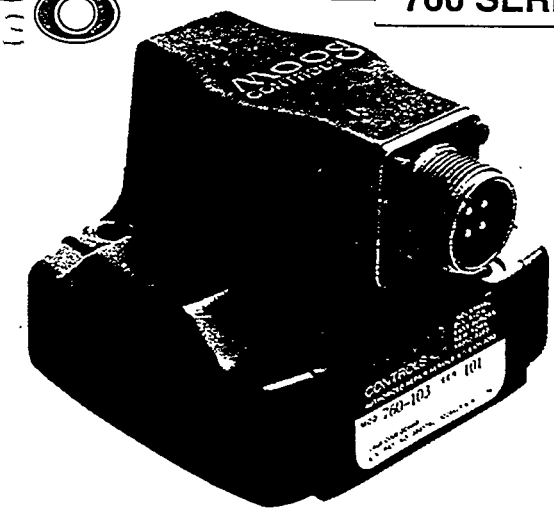
This sensor has extended range transmitters that go up to ± 12 ft. with a degradation in accuracy and resolution.

APPENDIX E

VENDOR LITERATURE FOR HARDWARE TEST STANDS

760 Series Servovalves

The 760 Series is a high performance, two-stage design that covers the range of rated flows from 1 to 15 gpm at 1000 psi valve drop. The output stage is a closed center, four-way, sliding spool. The pilot stage is a symmetrical double-nozzle and flapper, driven by a double air gap, dry torque motor. Mechanical feedback of spool position is provided by a cantilever spring. The valve design is simple and rugged for dependable, long life operation.



Specifications:

Fluid Supply: 760 Series Servovalves are intended to operate with constant supply pressure.

Supply Pressure:
Minimum: 200 psi (14 bar)
Maximum Standard: 3500 psi (245 bar)
Maximum (special order): 5000 psi (350 bar)

Proof Pressure:
150% of supply pressure at P port
100% of supply pressure at R port

Fluid:
Compatible with common hydraulic fluids
Recommended viscosity range:
60-450 SUS @ 100°F
(10-97 cSt @ 38°C)

Cleanliness Level:
ISO DIS 4406 code 16/13 max.
14/11 recommended

Operating Temperature:
Minimum: -40°F (-40°C)
(maximum fluid viscosity: 6000 SUS)
Maximum: +275°F (+135°C)

Rated Flow Tolerance: ±10%

Symmetry: < 10%

Hysteresis: < 3%

Threshold: < 1/2%

Null Shift:
with temperature, 100°F variation: < 2%
with acceleration to 10g: < 2%
with supply pressure 1000 psi change: < 2%
with back pressure 0 to 500 psi: < 2%

Frequency Response: Typical response characteristics for 760 Series Servovalves are shown in Figures 1, 2, and 3.

Step Response: Typical transient responses of 760 Series Servovalves are shown in Figure 4. The straight line portion of the response represents saturation flow from the pilot stage which will increase with higher supply pressures.

Model	Response	Rated Flow (1000 psi)		Internal Leakage (1000 psi)		Rated Current (parallel coils)	Coil Nom. Resistance
		gpm	lpm	gpm	lpm	mA	ohms
760-100A	Std.	1	3.8	<0.17	<0.66	15	200
760-910A	High.	1	3.8	<0.23	<0.92	40	80
760-101A	Std.	2.5	9.5	<0.22	<0.83	15	200
760-911A	High.	2.5	9.5	<0.30	<1.15	40	80
760-102A	Std.	5	19	<0.35	<1.33	15	200
760-912A	High.	5	19	<0.45	<1.78	40	80
760-103A	Std.	10	38	<0.35	<1.33	15	200
760-233A	High.	10	38	<0.45	<1.78	40	80
760-104A	Std.	15	57	<0.35	<1.33	40	80
760-814A	High	15	57	<0.45	<1.78	40	80

Optional designs are available with intrinsically safe coils (FM approved), and/or special spool/bushing lap configurations. Available seal materials: BUNA (Std.), VITON or EPR.

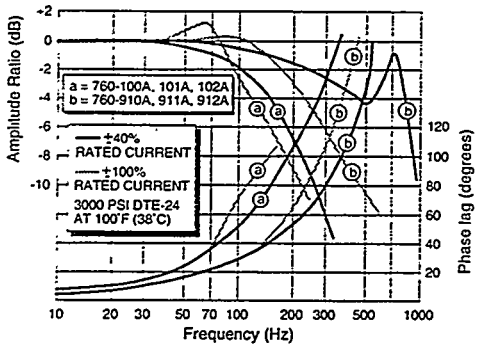


FIGURE 1 - FREQUENCY RESPONSE OF 1, 2-1/2 AND 5 GPM SERVOVALVES

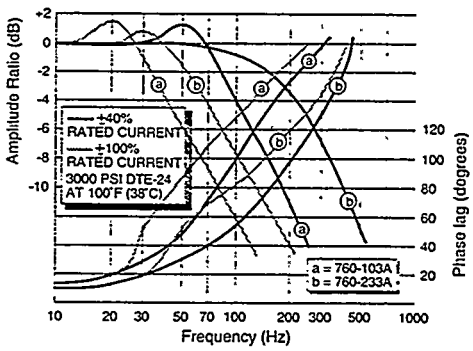


FIGURE 2 - FREQUENCY RESPONSE OF 10 GPM SERVOVALVES

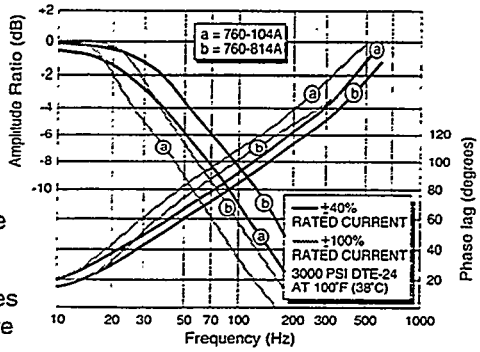


FIGURE 3 - FREQUENCY RESPONSE OF 15 GPM SERVOVALVES

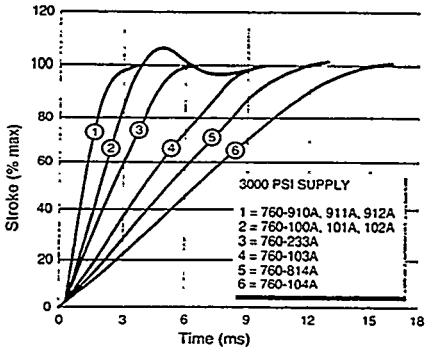


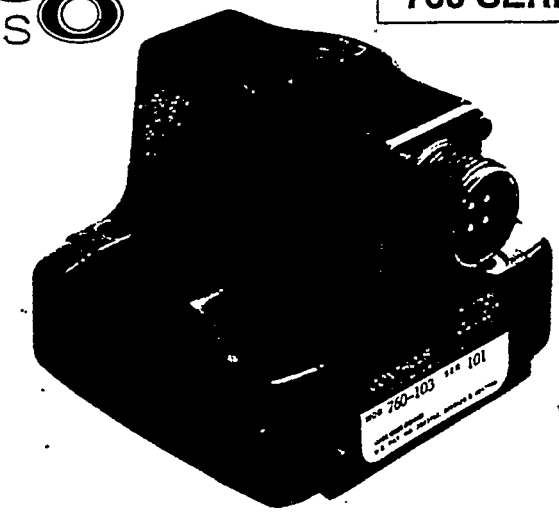
FIGURE 4 - STEP RESPONSES

Moog CONTROLS

760 SERIES

760 Series Servovalves

The 760 Series is a high performance, two-stage design that covers the range of rated flows from 1 to 15 gpm at 1000 psi valve drop. The output stage is a closed center, four-way, sliding spool. The pilot stage is a symmetrical double-nozzle and flapper, driven by a double air gap, dry torque motor. Mechanical feedback of spool position is provided by a cantilever spring. The valve design is simple and rugged for dependable, long life operation.



Specifications:

Fluid Supply: 760 Series Servovalves are intended to operate with constant supply pressure.

Supply Pressure:
Minimum: 200 psi (14 bar)
Maximum Standard: 3500 psi (245 bar)
Maximum (special order): 5000 psi (350 bar)

Proof Pressure:
150% of supply pressure at P port
100% of supply pressure at R port

Fluid:
Compatible with common hydraulic fluids
Recommended viscosity range:
60-450 SUS @ 100°F
(10-97 cSt @ 38°C)

Cleanliness Level:
ISO DIS 4406 code 16/13 max.
14/11 recommended

Operating Temperature:
Minimum: -40°F (-40°C)
(maximum fluid viscosity: 6000 SUS)
Maximum: +275°F (+135°C)

Model	Response	Rated Flow (1000 psi)		Internal Leakage (1000 psi)		Rated Current (parallel coils)	Coil Nom. Resistance
		gpm	lpm	gpm	lpm	mA	ohms
760-100A	Std.	1	3.8	<0.17	<0.66	15	200
760-910A	High	1	3.8	<0.23	<0.92	40	80
760-101A	Std.	2.5	9.5	<0.22	<0.83	15	200
760-911A	High	2.5	9.5	<0.30	<1.15	40	80
760-102A	Std.	5	19	<0.35	<1.33	15	200
760-912A	High	5	19	<0.45	<1.78	40	80
760-103A	Std.	10	38	<0.35	<1.33	15	200
760-233A	High	10	38	<0.45	<1.78	40	80
760-104A	Std.	15	57	<0.35	<1.33	40	80
760-814A	High	15	57	<0.45	<1.78	40	80

Optional designs are available with intrinsically safe coils (FM approved), and/or special spool/bushing lap configurations. Available seal materials: BUNA (Std.), VITON or EPR.

Rated Flow Tolerance: ±10%

Symmetry: < 10%

Hysteresis: < 3%

Threshold: < 1/2%

Null Shift:
with temperature, 100°F variation: < 2%
with acceleration to 10g: < 2%
with supply pressure 1000 psi change: < 2%
with back pressure 0 to 500 psi: < 2%

Frequency Response: Typical response characteristics for 760 Series Servovalves are shown in Figures 1, 2, and 3.

Step Response: Typical transient responses of 760 Series Servovalves are shown in Figure 4. The straight line portion of the response represents saturation flow from the pilot stage which will increase with higher supply pressures.

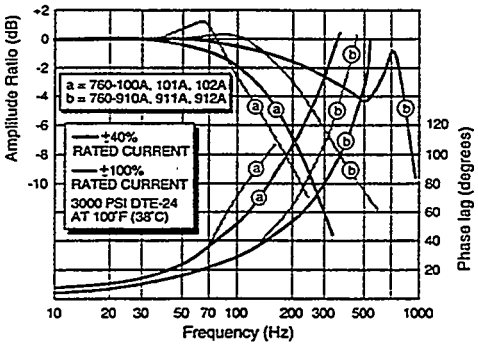


FIGURE 1 - FREQUENCY RESPONSE OF 1, 2-1/2 AND 5 GPM SERVOVALVES

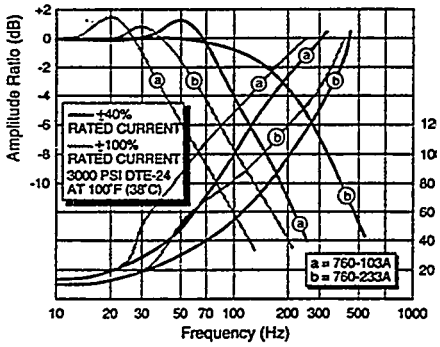


FIGURE 2 - FREQUENCY RESPONSE OF 10 GPM SERVOVALVES

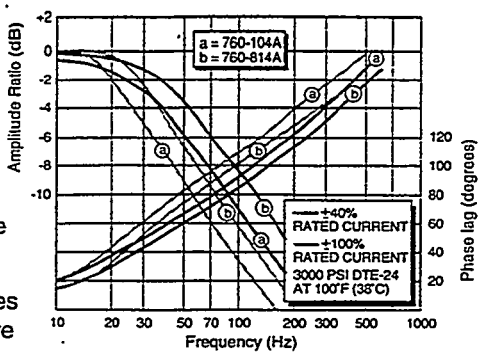


FIGURE 3 - FREQUENCY RESPONSE OF 15 GPM SERVOVALVES

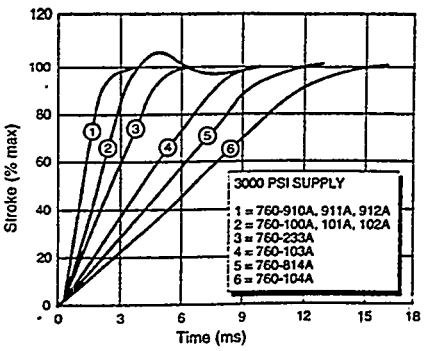
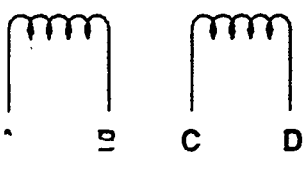
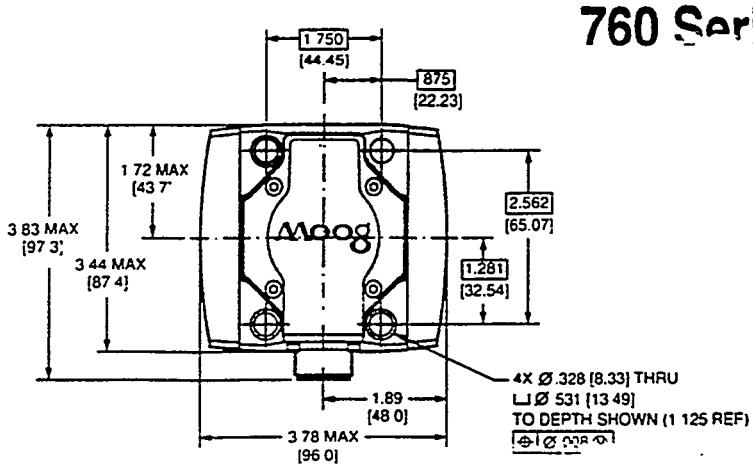


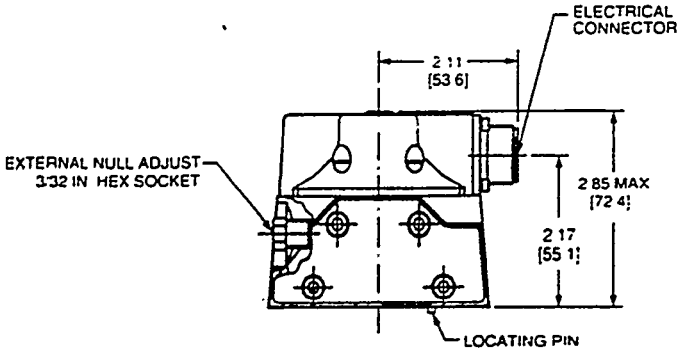
FIGURE 4 - STEP RESPONSES

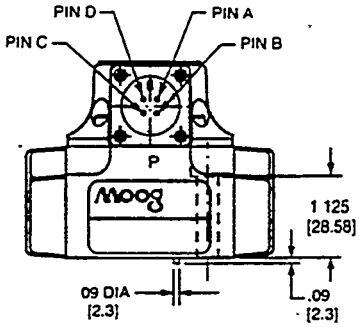
Standard Electrical Configuration



External connections and electrical polarity for flow out control port No. 2 are:
- single coil: A+, B-; or C+, D-
- series coils: tie B to C; A+, D-
- parallel coils: tie A to C and B to D; A & C+, B & D-







Accessories:

Flushing Block: PN 23718-1K1
Mating Electrical Connector:
PN 49054F14S2S (MS3106F14S-2S)
Suggested Mounting Bolts:
PN A31324-228B
5/16 - 18NC x 1-3/4 long
Socket Head Cap Screw

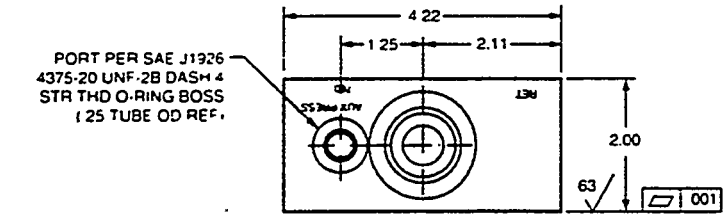
Subplate:

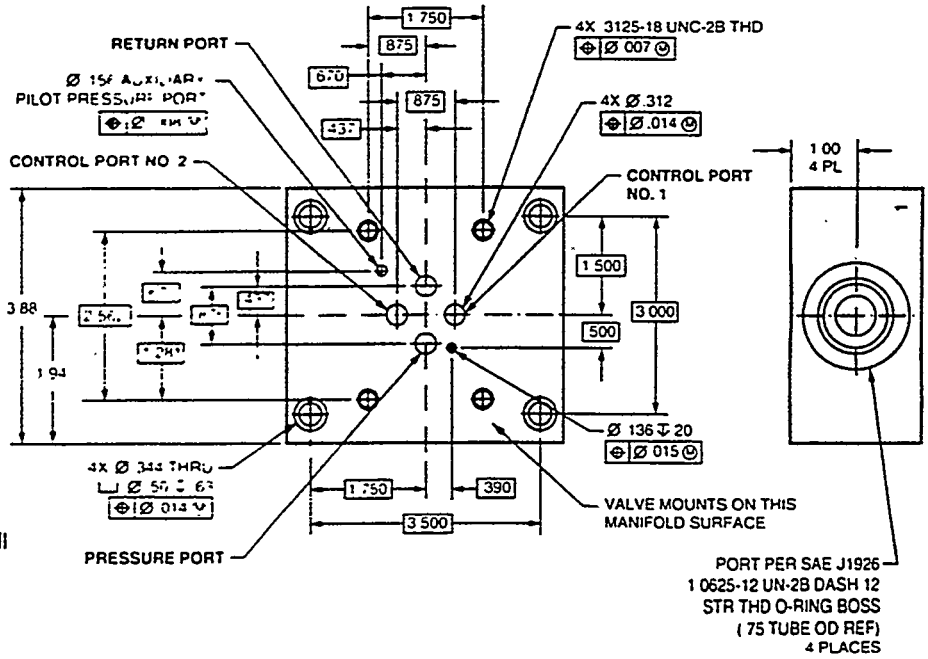
		3000 psi	5000 psi
4 port	PN 43586	-1	-5
5 port	PN 43586	-4	-7

System filters available. Call factory for details or see brochure #605.

Notes:

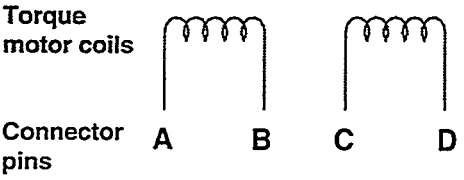
Valve Weight: 2.3 lb (1.03 kg)
Subplate O-Ring Size: 0.070 [1.78] sect. x 0.364 [9.25] I.D. (universal size -013)
Aux. Pilot Pressure Port O-Ring Size: 0.070 [1.78]sect. x 0.364 [9.25] I.D. (universal size -012)
Aux. Pilot Pressure Port: Optional
Null Adjust: Flow out of control port No. 2 will increase with clockwise rotation of null adjust screw (3/32 hex key).
Surface Finish: Surface to which valve is mounted requires 63 [ΔΔ] finish, flat within 0.001 [0.03] TIR.



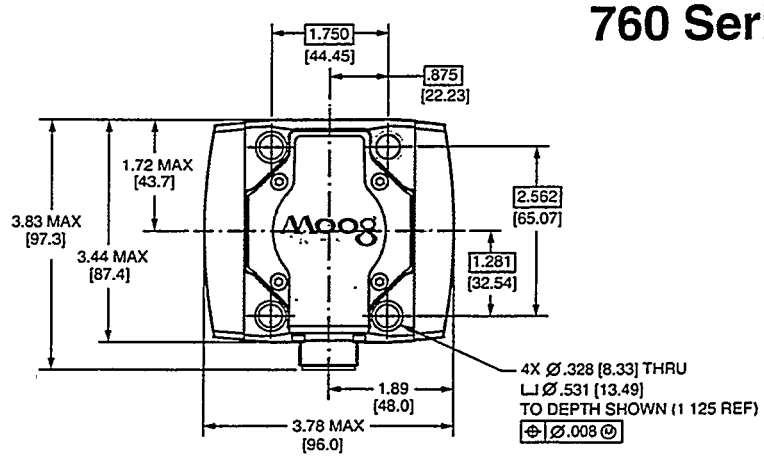


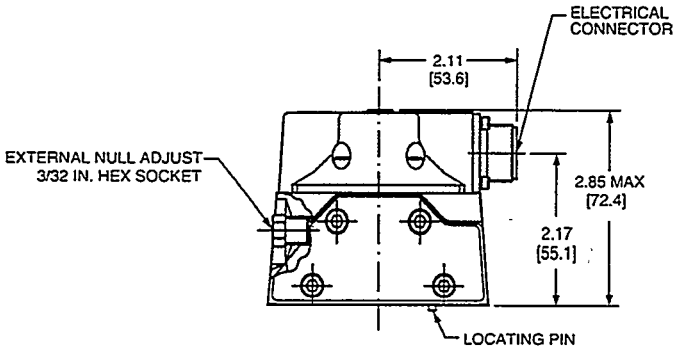
Typical Subplate Manifold

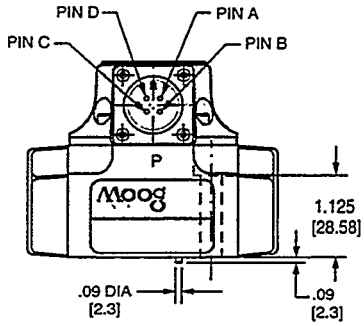
Standard Electrical Configuration



External connections and electrical polarity for flow out control port No. 2 are:
- single coil: A+, B-; or C+, D-
- series coils: tie B to C; A+, D-
- parallel coils: tie A to C and B to D; A & C+, B & D-







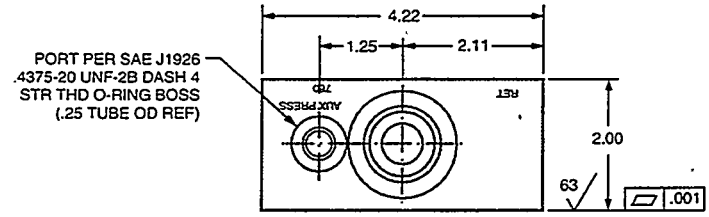
Accessories:

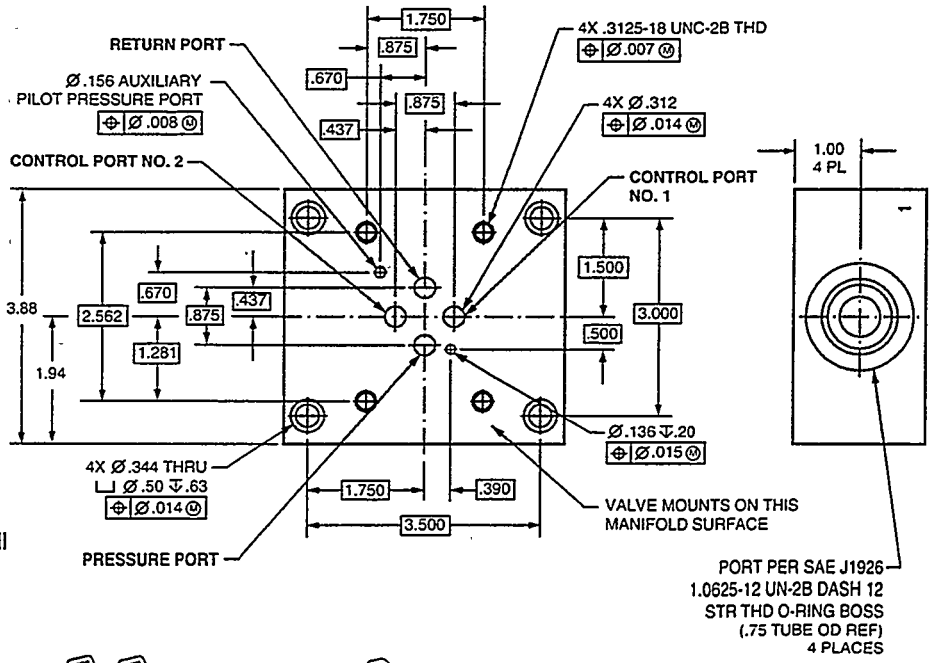
Flushing Block: PN 23718-1K1
Mating Electrical Connector:
PN 49054F14S2S (MS3106F14S-2S)
Suggested Mounting Bolts:
PN A31324-228B
5/16 - 18NC x 1-3/4 long
Socket Head Cap Screw

Subplate:

		3000 psi	5000 psi
4 port	PN 43586	-1	-5
5 port	PN 43586	-4	-7

System filters available. Call factory for details or see brochure #605.





Notes:

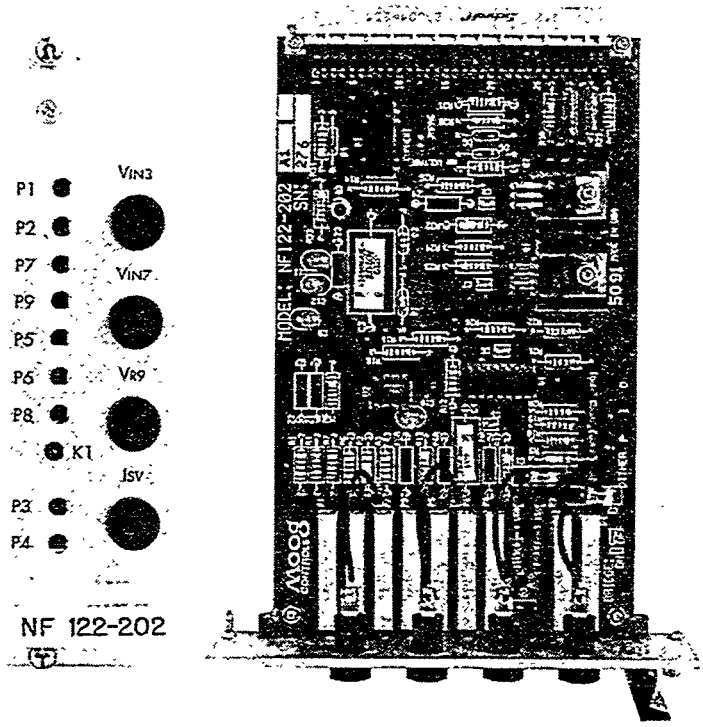
Valve Weight: 2.3 lb (1.03 kg)
Subplate O-Ring Size: 0.070 [1.78] sect. x 0.426 [10.82] I.D. (universal size -013)
Aux. Pilot Pressure Port O-Ring Size: 0.070 [1.78]sect. x 0.364 [9.25] I.D. (universal size -012)
Aux. Pilot Pressure Port: Special order
Null Adjust: Flow out of control port No. 2 will increase with clockwise rotation of null adjust screw (3/32 hex key).
Surface Finish: Surface to which valve is mounted requires 63 [VV] finish, flat within 0.001 [0.03] TIR.

Typical Subplate Manifold

Model NF122-202A1 Servoamplifier

This circuit card is designed to drive servovalves or proportional valves in closed-loop servosystems. It provides any combination of proportional, integral, and derivative control (PID). It may also be used as a simple voltage-to-current converter to drive servovalves.

The NF122-202A1 Servoamplifier is a form-fit-function replacement for the F122-202A001.



Features

PID Control
- Jumper-selectable proportional, integral, and derivative control
- Independent integral and derivative gain adjustments
- Adjustable low-pass filter on derivative control
- Easily-accessible integrator reset function

Error-Summing Input Stage
- Three standard inputs can be reconfigured for differential input
- Independent gain, bias, trim, and scale potentiometers
- Summed error easily accessible for monitoring

Front-Panel Adjustments
- Provide quick access to gains, scales, bias, and dither

Front-Panel Test Points
- Allow for fast & easy setup, test, & monitoring of parameters

Dither Generator
- Jumper-selectable with adjustable frequency and amplitude

Current or Voltage Drive
- Jumper-selectable with overcurrent protection

SPDT Relay Section
- Energized by high (5 to 15 Vdc) or low (0 Vdc) logic signal
- Used for integrator reset, signal switching, or other functions

Specifications

Proportional Gain	5 to 200 mA/V
Integral Gain	5 to 3,000 mA/V-sec
Derivative Gain	0.02 to 8 mA-sec/V
Current Output	to ± 50 mAdc into 130Ω coil
Voltage Output	to ± 10 Vdc nominal
Input Levels	to ± 15 Vdc on terminals 3 and 7 to ± 120 Vdc on terminal 9
Drift	≤ 0.1 mV/°C from 10°C to 50°C (with 100KΩ inputs and proportional gain at 50 mA/V)
Temperature Range	10°C to 50°C (50°F to 120°F)
Frequency Response	-3 db @ ≥ 800 Hz (1 Henry load)
Linearity	± 3% full scale
Dither	25 to 300 Hz, 0 to 20 mA p-p
Relay	contacts: ≤ 2 A/24 Vdc consumption: 15 mA from +24 Vdc line
Connector	DIN 41612 style C
Form Factor	Eurocard 100 x 160 mm, 7HP, 3U
Weight	0.36 lb (0.16 kg)

Adjustments

P1 **Bias** Changes bias voltage at input (summing) stage. Turn CW for positive input bias voltage. Adjust for desired offset between command and feedback.

P2 **Gain** Changes proportional gain of input (summing) stage. Also changes integral and/or derivative gains if jumpers I or D are on. Turn CW to increase gain. Adjust for system stability.

P3 **Dither Frequency** Turn CW to increase frequency. Adjust for a frequency appropriate for system dynamics.

P4 **Dither Amplitude** Turn CW to increase amplitude. Adjust for desired dither current amplitude, typically ± 10% of rated current. Note: jumper DITHER must be on.

P5 **Integral Gain** Changes integral gain if jumper I is on. Turn CW to increase gain. Adjust for system stability.

P6 **Filter Frequency** Changes corner frequency of low-pass filter on differentiator. Turn CW to increase frequency. Adjust to reduce excessive noise.

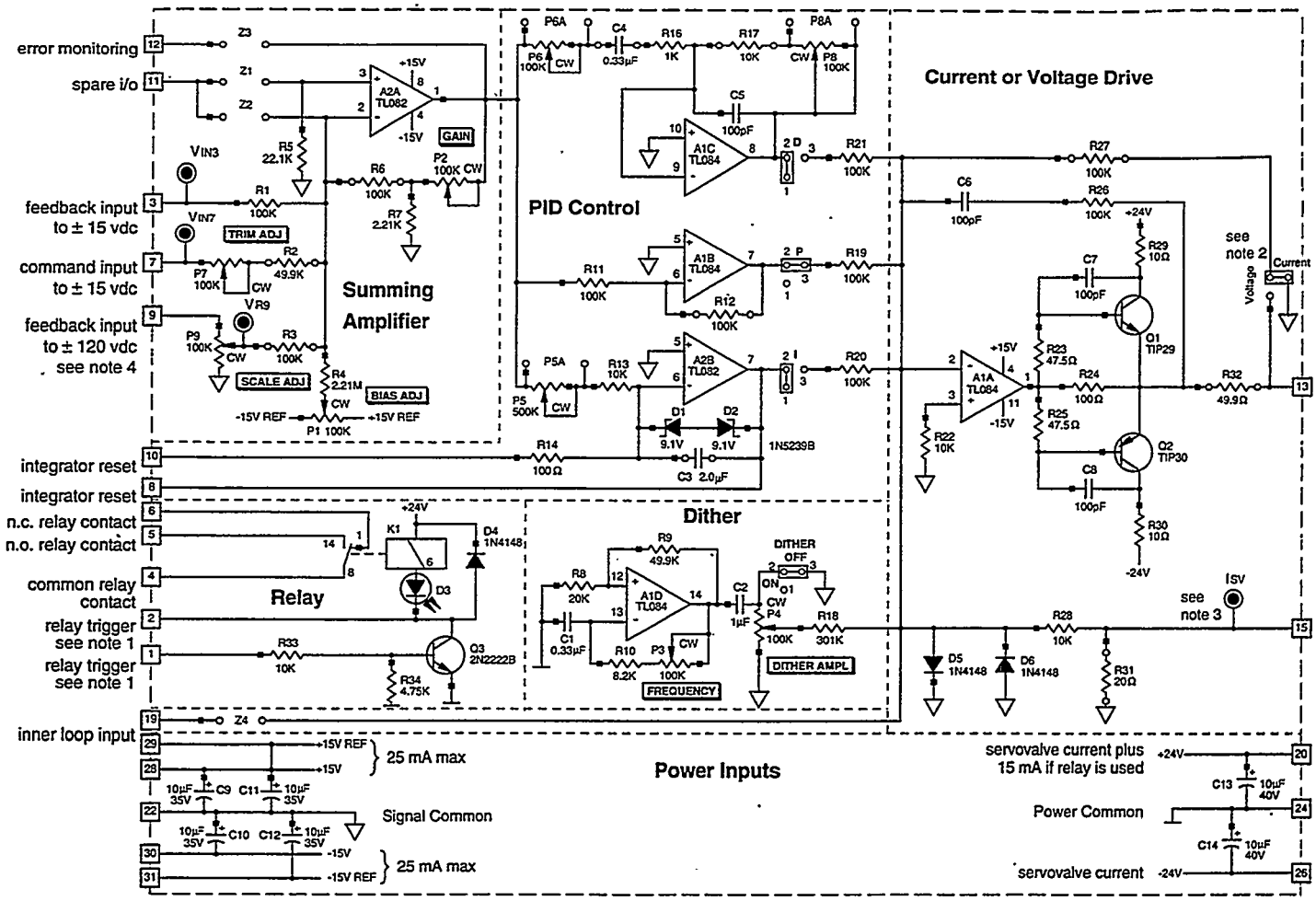
P7 **Trim** Changes authority of signal on terminal 7. Turn CW to reduce authority. Adjust to provide scaling of input at terminal 7.

P8 **Derivative Gain** Changes derivative gain if jumper D is on. Turn CW to decrease gain. Adjust to add phase lead.

P9 **Scale** Changes authority of signal on terminal 9. Turn CW to reduce authority. Adjust to provide scaling of input at terminal 9.

Servoamplifier Schematic

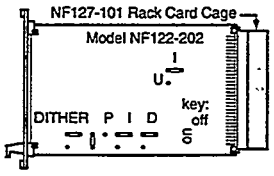
Model NF122-202A1



Note 1: To energize relay, connect terminal 2 to terminal 24 or connect +5 to +15 Vdc to terminal 1.
Note 2: Place I/U jumper in I position for current drive. Place I/U jumper in U position for voltage drive.
Note 3: Voltage at test point "Isv" represents current through valve coils with a scale factor of 50 mA/Volt.
Note 4: Voltage at test point "V_{RS}" must not exceed ± 15 Vdc.
Note 5: ■ = pin 1 (square pin).

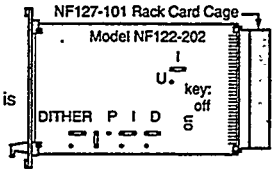
Typical Applications

Proportional Loops:
(current drive)
position, pressure, force, and torque control



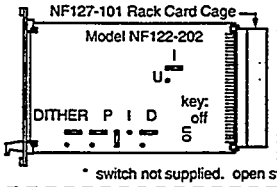
7 command signal
3 feedback signal
13 to servovalve coils
15 to servovalve coils

Voltage-to-Current Converter:
to drive servovalve when loop is closed with a digital controller



7 command signal
13 to servovalve coils
15 to servovalve coils

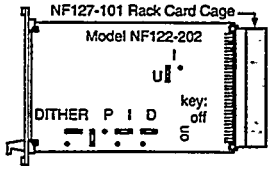
Integral Loops:
(current drive)
velocity and flow control



7 command signal
3 or 9 feedback signal
13 to servovalve coils
15 to servovalve coils
8 to 4
10 to 6
2
24

* switch not supplied. open switch to reset integrator

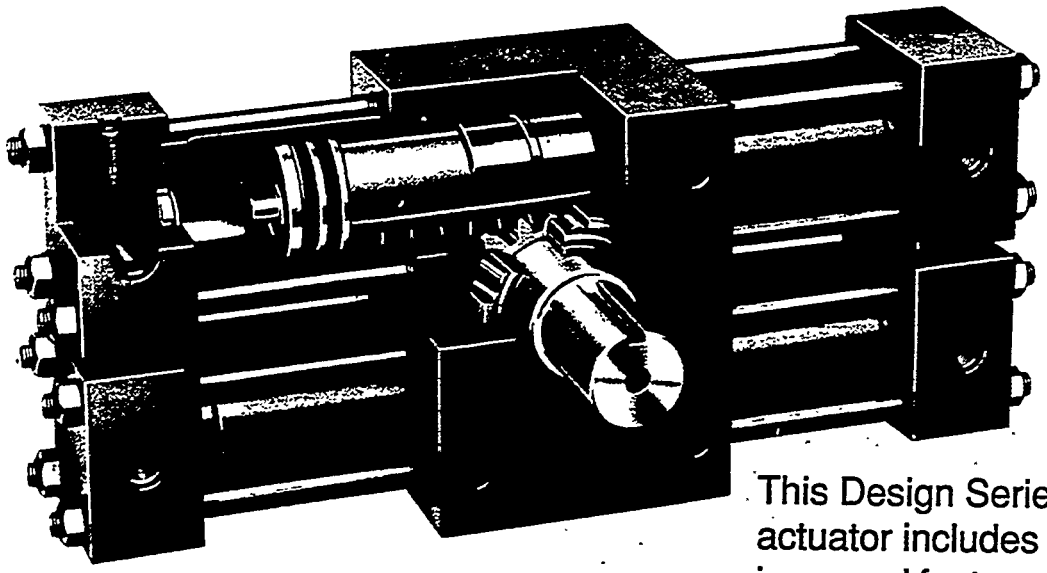
Voltage Driver for Proportional Valves:
position, pressure, force, and torque control



7 command signal
3 feedback signal
13 to valve input
22 to valve common

HTR Series

Hydraulic Rack & Pinion Rotary Actuator



This Design Series "C" actuator includes improved features to provide greater strength and service life under the most severe conditions.

Standard features of the HTR Series include:

■ TAPERED ROLLER BEARINGS

Large tapered roller bearings are used to support the pinion and output shaft, allowing the unit to support high external radial and thrust loads. These bearings ensure a minimum of downtime for high cycling applications, as well as reducing total system complexity.

■ DUCTILE IRON HOUSING

A high strength ductile iron housing is now standard on all size actuators. This rugged housing provides greater shock resistance, greater flexibility in mounting, and greater assurance of long unit life.

■ SAE SIDE PORTS

SAE side ports are used to provide a compact and leakfree actuator. The side ports can be specified in any one of four orientations to simplify plumbing and installation. SAE straight thread O-ring ports give accurate and tight fluid connections with guaranteed zero external leakage.

■ WEAR-PAK PISTONS AND SEALS

High strength steel pistons are piloted to the rack, ensuring concentricity. Filled PTFE wear bands eliminate metal to metal contact and scoring. Self energizing deep Molythane PolyPak seals reduce maintenance with virtually leakproof operation and long life.

■ CHROME ALLOY STEEL PINION AND RACK

Through hardened high quality chromium alloy steel is used for the rack and pinion. The through hardening provides maximum shock resistance and strength while minimizing wear, providing long, dependable service life.

TIE RODS
Medium alloy cold worked steel with precision rolled thread for greater strength and fatigue resistance.

BRONZE RACK BEARINGS
A large support area for the rack is provided by a bronze rack bearing, which reduces wear and gives long life. (Note: standard on HTR15 through HTR600.)

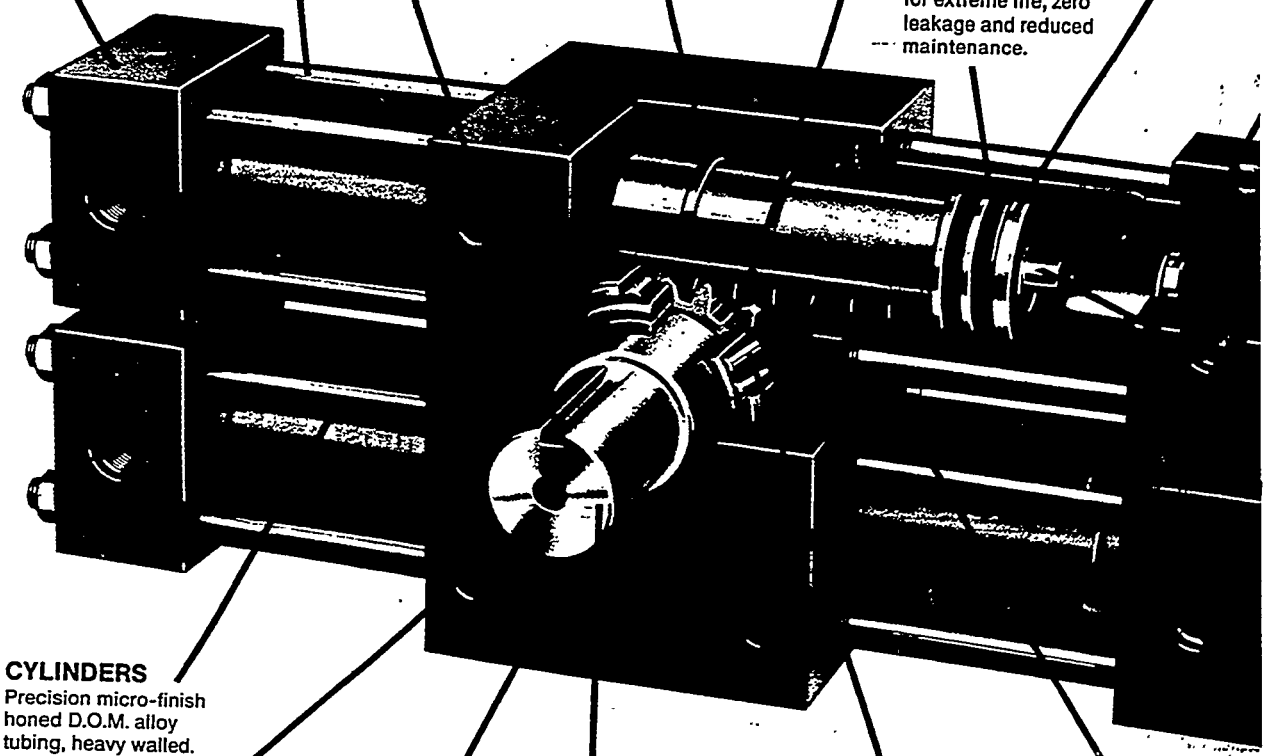
RACK & PINION
Heavy duty gear design is made from through hardened chrome alloy steel for maximum strength and shock resistance and long, dependable operation.

PISTON:
one piece
is piloted
ensuring
A heavy d
band elim.
to metal c
resultant
scoring, g
binding.

END CAPS
Machined from cold rolled steel plate, designed to NFPA standard cylinder bore sizes for increased flexibility of options and greater reliability.

KEYWAY
at 12:00 position at mid stroke of actuator.

PISTON SEALS
Heavy duty deep Molythane PolyPak seals are self compensating for extreme life, zero leakage and reduced maintenance.



CYLINDERS
Precision micro-finish honed D.O.M. alloy tubing, heavy walled.

THREADED MOUNTING HOLES
(4) each side offers choice of mounting surfaces for flexibility in design, optional mounting surfaces also available.

HOUSING
High strength ductile iron provides ultimate strength and resistance to shock loads.

STANDARD MALE KEYED SHAFT
is as large as possible to ensure superior strength; pinion and output shaft are one piece to provide long life.

THREADED BEARING RETAINER (FAR SIDE)
assures proper tapered roller bearing preload and resultant bearing load capacity by eliminating the inconvenience and guesswork of shims. Once properly torqued, a set screw securely locks the retainer in place, even under shock and high load conditions.

TAPERED ROLLER BEARINGS
provide substantial pinion and shaft support ensuring a rigid and long lasting assembly. The large bearings allow external radial and thrust loads to be applied as well, simplifying complexity and providing longer service life, even under high cycling applications.

LOW
RELIE
prevent
overpre
gear ch
possible
pinion s
provides
indicato
wear.

NOTE:
All pressure containing seals can be inspected or replaced without removing actuator from customer's shaft or disturbing actuator mounting.

For additional information – call your local Parker Fluidpower Motion & Control Distributor.

HTR Series

The HTR Series actuator incorporates significant features and improvements designed to give superior life under the most severe operating conditions. These features include high capacity tapered roller bearings, chrome alloy steel pinion and racks, and high strength ductile iron housing. The HTR Series is the most rugged and durable tie rod style actuator available today.

The HTR Series actuator will provide superior life on the most demanding applications, including machine tools, plastics and rubber machinery, process equipment, material handling equipment and in basic metals manufacturing.

The many options available on the HTR Series allows the actuators to be "designed" to the application, which can reduce overall system complexity and cost. For example, the optional hollow shaft and base mount can eliminate the need for external support bearings, as the actuator bearings themselves have additional load capacity. Or, specifying the optional cushions can eliminate the need for external deceleration devices, and reduce system shock and vibration.

The HTR Series actuator is also designed for ease of maintenance. The Wear-Pak pistons with self-energizing Molythane PolyPak piston seals can be inspected or serviced without removing the actuator from the mounting or shaft. The piston seals are the only dynamic seals that are pressurized, the pinion and shaft seals never see system pressure, eliminating the possibility of external leakage.

To guarantee delivery of only the highest quality products, all HTR Series actuators are 100% inspected and tested before shipment. These tests include a static proof test at 4500 psi, a breakaway pressure test, and a rigorous inspection to detect any external leakage, all ensuring shipment of the finest rotary actuator possible.

HTR SERIES SPECIFICATIONS

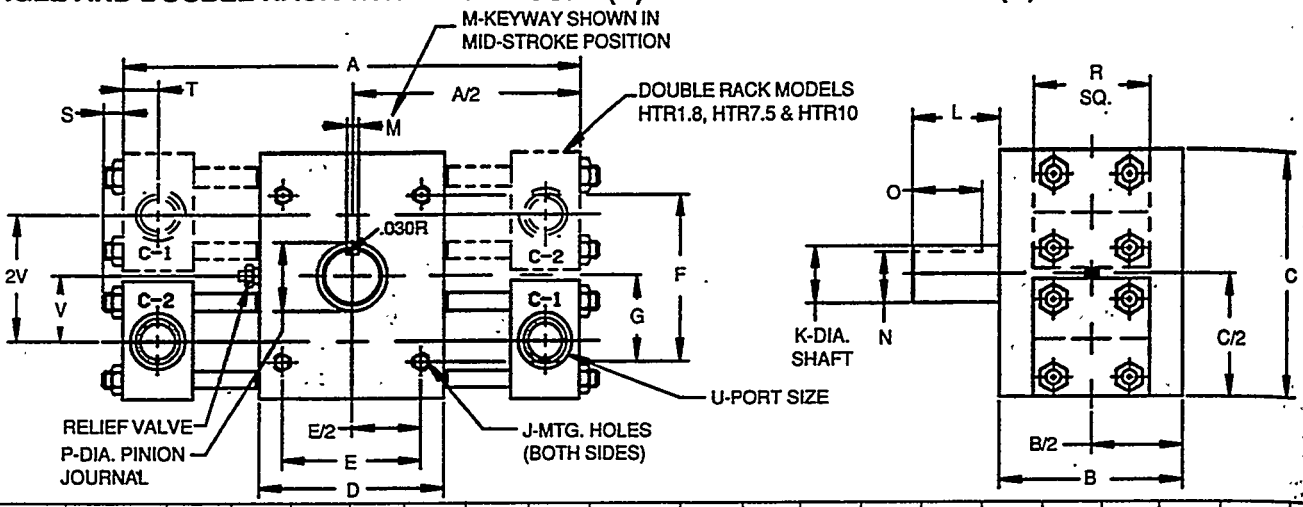
- Maximum Operating Pressure: 3000 psi non-shock, except HTR22/45: 2000 psi non-shock
- Proof Pressure: 4500 psi
- Standard Rotations: 90°, 180°, 360°
- Rotational Tolerance: -0°, +2°
- Output Torques @ 3000 psi: 900 lb-in to 600,000 lb-in
- Leakage - External: 0 cubic inches per minute
  - Internal: 0 cubic inches per minute
- Maximum Breakaway Pressure: 75 psi
- Operating Temperature (Standard Nitrile Seals): -40°F to 180°F
- Mounting Orientation: Unrestricted
- Timing: Keyway located at 12:00 position at midstroke of actuator

SPECIFICATION TABLE

Model		Maximum Pressure Rating, psi	Actual Output Torque, lb-in at Specified Pressure			Maximum Angular Backlash, Minutes	Standard Rotation Degrees	Displacement, Cubic Inches	Standard Unit Weight, lb
Single Rack	Double Rack		1000 psi	2000 psi	3000 psi				
HTR9		3,000	300	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	HTR1.8	3,000	600	1,200	1,800	30	90 180 360	1.13 2.27 4.53	16 19 25
HTR3.7		3,000	1,250	2,500	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	HTR7.5	3,000	2,500	5,000	7,500	30	90 180 360	4.86 9.71 19.4	35 41 53
HTR5		3,000	1,650	3,300	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	HTR10	3,000	3,300	6,600	10,000	30	90 180 360	6.25 12.5 25.0	45 54 66
HTR15		3,000	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	HTR30	3,000	10,000	20,000	30,000	15	90 180 360	17.8 35.5 71.0	89 97 117
HTR22		2,000	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	HTR45	2,000	15,000	30,000	—	15	90 180 360	27.8 55.5 111	98 108 134
HTR75		3,000	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	HTR150	3,000	50,000	100,000	150,000	15	90 180 360	92 184 368	321 367 454
HTR300		3,000	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	HTR600	3,000	200,000	400,000	600,000	10	90 180 360	355 711 1421	1110 1260 1560

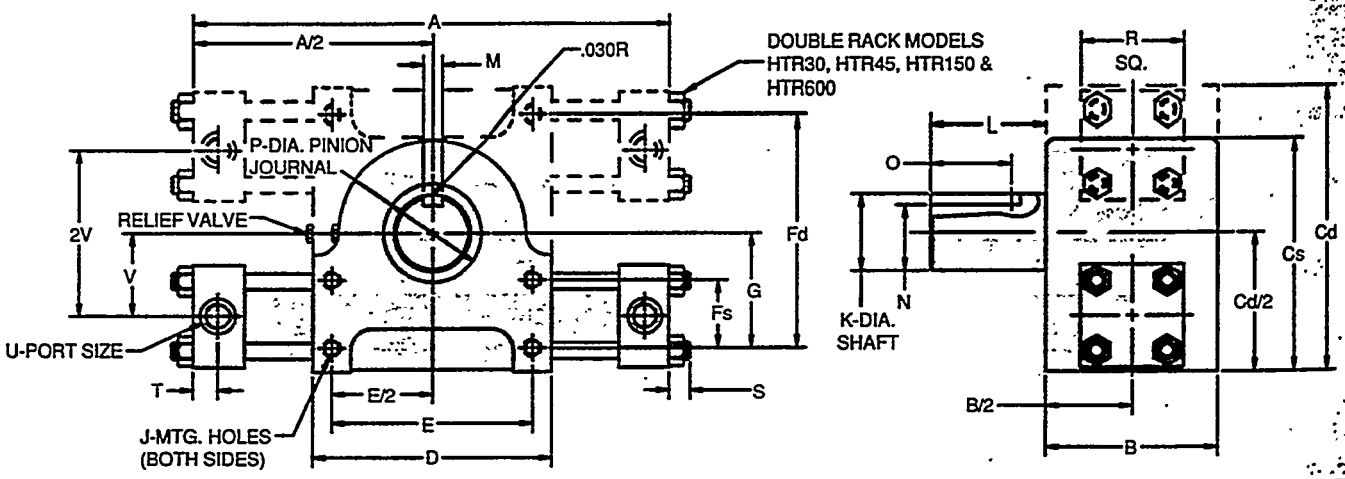
E

HTR.9 thru HTR10
SINGLE AND DOUBLE RACK WITH FACE MOUNT (A) AND MALE KEYED SHAFT (B)



Model	Rotation Degrees	A	B	C	D	E	F	G	J	K	L	M	N	O	P	R	S	T	U (SAE)	V
HTR.9 HTR1.8	90	7 1/4	3	3 5/8	3 1/2	2.625	2.375	1.188	5/16-18 x 1/2 DP	.875 .874	1 5/16	.250 .252	.732 .722	1	1	1 3/4	3/8	.50	9/16-18 (#6)	[illegible]
	180	9 1/8																		
	360	12 7/8																		
HTR3.7 HTR7.5	90	10 1/8	3 15/16	5 1/4	4	3.000	3.625	1.813	3/8-16 x 9/16 DP	1.250 1.249	1 7/8	.313 .315	1.074 1.064	1 1/2	1 1/2	2 1/2	1/2	.72	9/16-18 (#6)	1 3/8
	180	13 1/8																		
	360	18 1/4																		
HTR5 HTR10	90	11 3/8	3 15/16	6	4	3.000	5.000	2.500	3/8-16 x 9/16 DP	1.750 1.749	2 5/8	.438 .440	1.500 1.490	2	1 3/4	2 1/2	1/2	.72	9/16-18 (#6)	[illegible]
	180	14 5/8																		
	360	22 1/8																		

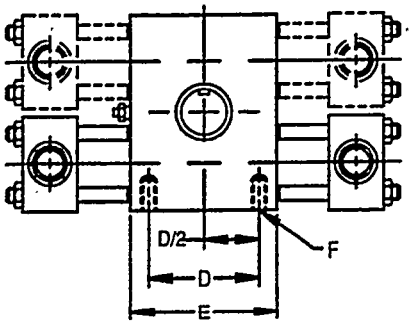
HTR15 thru HTR600
SINGLE AND DOUBLE RACK WITH FACE MOUNT (A) AND MALE KEYED SHAFT (B)

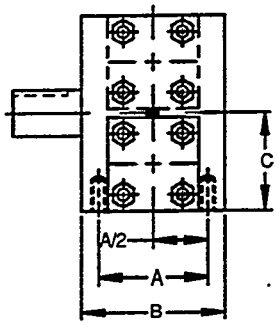


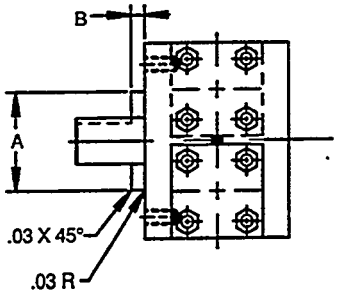
Model	Rotation Degrees	A	B	Cd	Cs	D	E	Fd	Fs	G	J	K	L	M	N	O	P	R	S	T	U (SAE)	V
HTR15	90	16	5	-	6 7/8	7	5.875	-	2.000	3.375	1/2-13 x 3/4 DP	2.250 2.249	3 3/8	.563 .565	1.933 1.928	2 3/8	2 7/8	3	5/8	.72	3/4-16 (#8)	2 [illegible]
	180	21 3/8																				
HTR30	360	32 1/8		8 1/8	-			6.750	-													
HTR22	90	16 3/8	5	-	6 15/16	7	5.875	-	2.000	3.375	1/2-13 x 3/4 DP	2.250 2.249	3 3/8	.563 .565	1.933 1.928	2 3/8	2 7/8	3 1/2	5/8	.72	3/4-16 (#8)	2 7/16
	180	21 3/4																				
HTR45	360	32 3/8		8 3/8	-			6.750	-													
HTR75	90	20 1/4	7 1/2	-	10	8 1/2	6.500	-	4.500	5.750	3/4-16 x 1 1/8 DP	3.000 2.999	4 1/2	.750 .752	2.577 2.567	3 3/8	3 3/4	5	1	.84	1 1/16-12 (#12)	[illegible]
	180	27 5/8																				
HTR150	360	42 1/4		13	-			11.500	-													
HTR300	90	31 1/4	12	-	15 1/4	15 7/8	13.000	-	5.000	7.625	1 1/4-7 x 1 7/8 DP	5.000 4.999	7 1/2	1.250 1.252	4.296 4.286	6	6 1/2	7 1/2	1 1/4	1.25	1 5/16-12 (#12)	5 1/8
	180	43 3/4																				
HTR600	360	68 7/8		18 3/4	-			13.500	-													

For additional information – call your local Parker Fluidpower Motion & Control Distributor

MOUNTING OPTIONS (B, P)







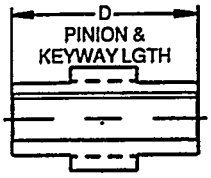
BASE MOUNTING (B)

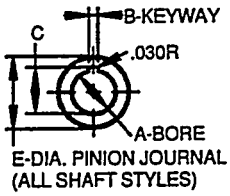
Model	A	B	C	D	E	F
HTR.9 HTR1.8	2.250	3	1.813	2.625	3 1/2	5/16-18 NC x 1/2 DP
HTR3.7 HTR7.5	3.000	3 15/16	2.625	3.000	4	3/8-16 NC x 9/16 DP
HTR5 HTR10	3.000	3 15/16	3.000	3.000	4	3/8-16 NC x 9/16 DP
HTR15/30 HTR22/45	3.875	5	4.063 4.188	5.875	7	1/2-13 NC x 3/4 DP
HTR75 HTR150	5.750	7 1/2	6.438	6.500	8 1/2	3/4-16 NF x 1 1/8 DP
HTR300 HTR600	9.500	12	9.375	13.000	15 7/8	1 1/4-7 NC x 1 7/8 DP

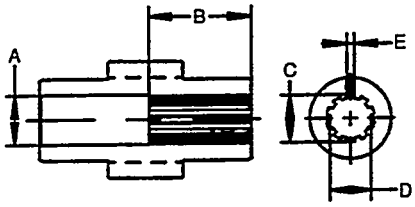
PILOT MOUNTING (P)

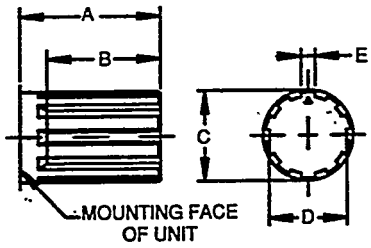
Model	A	B
HTR.9 HTR1.8	1.875 1.873	1/4
HTR3.7 HTR7.5	2.625 2.623	1/4
HTR5 HTR10	2.875 2.873	1/4
HTR15/30 HTR22/45	4.250 4.248	3/8
HTR75 HTR150	5.500 5.498	3/8
HTR300 HTR600	8.750 8.748	1/2

SHAFT OPTIONS All shaft options shown in mid stroke position.









FEMALE KEYED (A)

Model	A	B	C	D	E
HTR.9 HTR1.8	.625 .626	.187 .189	.709 .719	2 15/16	1
HTR3.7 HTR7.5	.875 .876	.187 .189	.964 .974	3 7/8	1 1/2
HTR5 HTR10	1.250 1.252	.250 .252	1.367 1.377	3 7/8	1 3/4
HTR15/30 HTR22/45	2.000 2.003	.500 .502	2.223 2.233	4 15/16	2 7/8
HTR75 HTR150	2.750 2.755	.750 .753	3.032 3.042	7 7/16	3 3/4
HTR300 HTR600	5.000 5.005	1.250 1.254	5.363 5.373	11 15/16	6 1/2

FEMALE SAE 10B SPLINE (D)

Model	A	B	C	D	E
HTR.9 HTR1.8	21/32	5/8	.625 .624	.538 .537	.098 .096
HTR3.7 HTR7.5	29/32	7/8	.875 .876	.753 .752	.137 .135
HTR5 HTR10	1 5/32	1 1/4	1.125 1.124	.968 .967	.176 .174
HTR15/30 HTR22/45	2 1/32	2	2.000 1.998	1.720 1.718	.312 .309
HTR75 HTR150	3 1/32	3	3.000 2.998	2.580 2.578	.468 .465
HTR300 HTR600	5 1/32	5	5.000 4.997	4.300 4.297	.780 .777

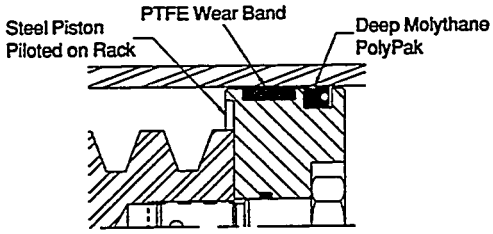
MALE SAE 10B SPLINE (E)

Model	A	B	C	D	E
HTR.9 HTR1.8	1 5/16	7/8	.873 .872	.747 .742	.134 .133
HTR3.7 HTR7.5	1 7/8	1 1/4	1.248 1.246	1.069 1.064	.192 .191
HTR5 HTR10	2 5/8	1 3/4	1.748 1.746	1.499 1.494	.270 .269
HTR15/30 HTR22/45	3 3/8	2 1/4	2.247 2.245	1.928 1.923	.347 .346
HTR75 HTR150	4 1/2	3	2.997 2.995	2.573 2.568	.464 .463
HTR300 HTR600	7 1/2	5	4.996 4.994	4.296 4.294	.776 .775

SEALS (V, W)

A durable Wear-Pak piston is standard on all units. This includes a deep Molythane PolyPak seal to eliminate leakage, and a filled PTFE wear ring to prevent metal to metal contact, wear, and contamination damage.

Operating Temperature Limits: -40° F to 180° F.
Recommended For: general purpose, petroleum based fluids.
Filtration Requirements: ISO class 17/14 cleanliness level.



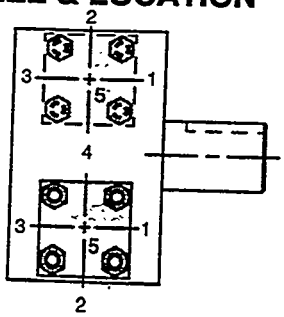
Optional seals:

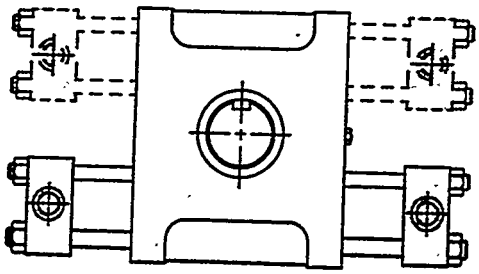
(V) Viton
Recommended for: High temperature and/or synthetic fluids.
Operating Temperature Limits: -20°F to 400°F

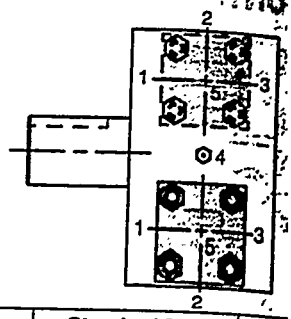
(W) Carboxilated Nitrile (Piston seals only)
Recommended for: Water glycol and high water content fluids.
Operating Temperature Limits: 30°F to 180°F

E

PORT SIZE & LOCATION







Model	Standard SAE Straight Thread (1)	NPT(2)
HTR.9 HTR1.8	9/16 - 18 (SAE 6)	1/4
HTR3.7 HTR7.5	9/16 - 18 (SAE 6)	1/4
HTR5 HTR10	9/16 - 18 (SAE 6)	1/4

Model	Standard SAE Straight Thread (1)	NPT
HTR15/22 HTR30/45	3/4 - 16 (SAE 8)	[illegible]
HTR75 HTR150	1 1/16 - 12 (SAE 12)	3/4
HTR300 HTR600	1 5/16 - 12 (SAE 16)	[illegible]

NOTE:

1. Port position 1 is standard.
2. Port position 2, 3 and 4 are standard options available at no additional cost.
3. Port position 5 is available at additional cost.

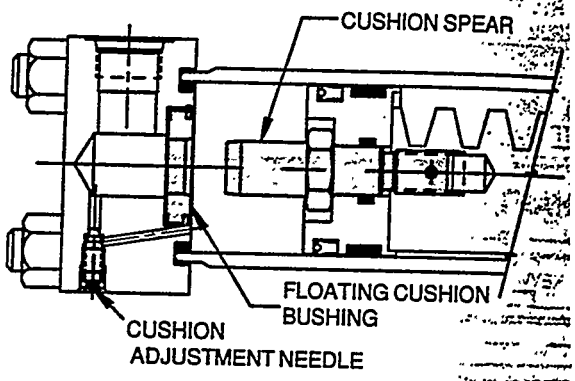
CUSHIONS (1, 2, 3, 4)

The standard cushion operates over the last 20° of rotation in either or both directions. A floating bushing ensures no binding of cushion spear. For severe operating conditions high performance cushions should be fitted on double rack units. All cushions are fully adjustable. On double rack units with only 2 cushions, cushions are located on upper cylinders.

STANDARD CUSHION NEEDLE LOCATIONS
(Reference above chart)

Port Position	Cushion Position
1	2
2	3
3	2
4*	3
5	2

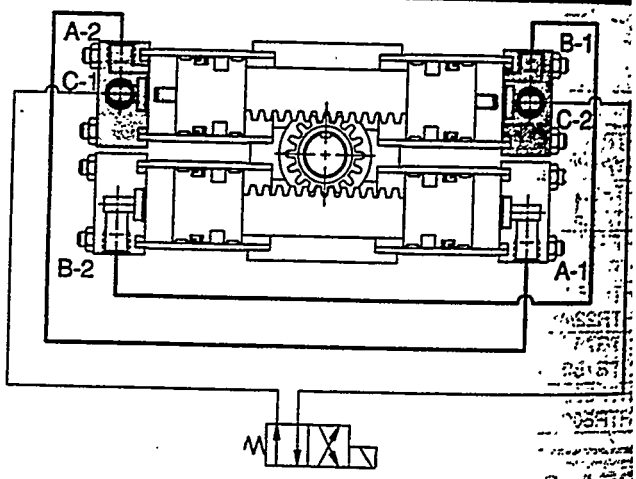
*Single rack only.



HIGH PERFORMANCE CUSHION (8)

(This option can be specified on double rack units only)

By combining the output/exhaust flow from two cylinders, then routing it through a single cushion bushing and cushion adjuster, cushion performance is enhanced. The increased cushion flow results in better control, doubles the cushioning torque, and eliminates dangerous pressure intensification. This unique circuit also eliminates two pipe or tubing tees.



OPERATION:

The work ports of a standard directional valve are plumbed to ports C-1 and C-2. Port A-1 is plumbed directly to A-2, and port B-1 is plumbed to B-2. When pressure is applied to port C-1 (clockwise shaft rotation), fluid is also directed through line A to the other rack. Exhaust flow from B-1 through B-2 is directed through the cushion bushing and cushion adjust ment. When the cushion spear closes off the main passage, total flow from both end caps is directed across one cushion adjustment needle, equalizing back pressure and improving performance. Alternatively, pressurizing C-2 and exhausting C-1 reverses the operation.

DIMENSIONAL INFORMATION:

Units are identical to standard double rack and pinion units, with the exception of porting location. This chart describes the location of the ports.

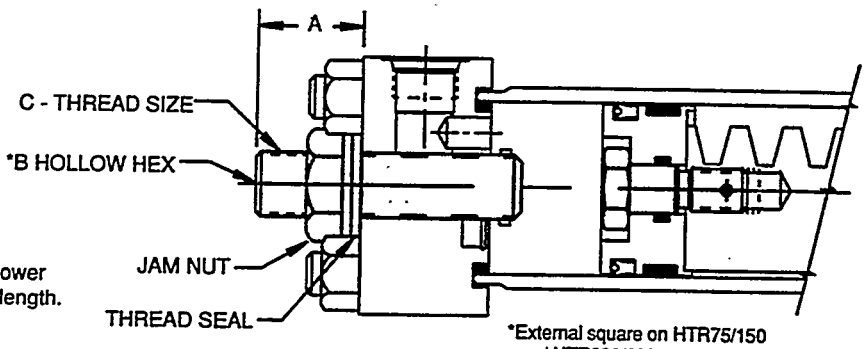
Work Ports C-1, C-2 Port Position	Cushion Adjustment Position	Connection Ports A-1, A-2, B-1, B-2 Port Position
1	2	4
2	3	1
3	2	4
5	2	3

For additional information – call your local Parker Fluidpower Motion & Control Distributor.

STROKE ADJUSTERS, 5° and 30°

NOTE:

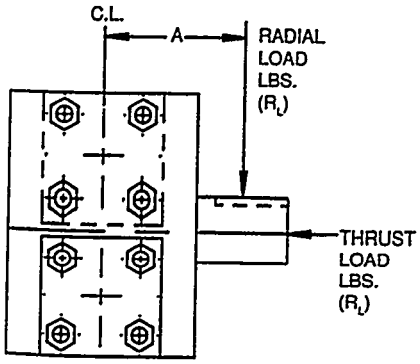
1. Maximum unit rotation is equal to rotation specified in model code. Adjusters allow rotational positioning equal to or less than the maximum rotation.
2. 5° stroke adjusters are available with or without cushions. Double rack units will have cushions on upper rack and stroke adjusters on lower rack. Single rack units will require additional "A" length.
3. 30° stroke adjusters not available with cushions.



*External square on HTR75/150 and HTR300/600

Model	(1) Turn Adj.	5° Adjustment w/o Cushioned End Cap A (Max)	5° Adjustment w/ Cushioned End Cap A (Max)	30° Adjustment w/o Cushioned End Cap A (Max)	B	C
HTR.9 HTR1.8	4.0°	.50	.88	.75	5/32	5/16 - 24 UNF
HTR3.7 HTR7.5	3.3°	.63	1.13	1.13	1/4	1/2 - 20 UNF
HTR5 HTR10	2.5°	.63	1.13	1.13	1/4	1/2 - 20 UNF
HTR15/22 HTR30/45	2.0°	.88	1.81	1.63	3/8	3/4 - 16 UNF
HTR75 HTR150	2.0°	2.56	3.75	3.56	*	1 1/2 - 12 UNF
HTR300 HTR600	1.2°	3.56	6.06	5.31	*	2 1/2 - 12 UNF

BEARING LOAD CAPACITIES



NOTE:

1. Static Bearing Load Capacities = Dynamic Values x 1.5
2. Values listed are "Bearing" moment capacities. Standard male shaft sizes do not provide 4:1 design factor at these operating conditions. Larger shaft sizes are available. Consult factory for further details.

Dynamic[1] Bearing Load Capacities vs. Operating Pressure									
Model	Radial Load (lbs.) R_L (Per Bearing) @			Thrust Load (lbs.) R_T @			Overhung Moment (lb-in) R_L x A @		
	1,000 psi	2,000 psi	3,000 psi	1,000 psi	2,000 psi	3,000 psi	1,000 psi	2,000 psi	3,000 psi
HTR.9	3,730	3,420	3,120	2,690	2,590	2,490	2,440	2,240	2,040
HTR1.8	4,030	4,030	4,030	2,790	2,790	2,790	2,640	2,640	2,640[2]
HTR3.7	5,920	4,980	4,100	3,610	3,360	3,120	6,250	5,250	4,325
HTR7.5	6,750	6,750	6,750	3,830	3,830	3,830	7,120	7,120	7,120[2]
HTR5	7,670	6,790	5,910	4,240	4,020	3,810	7,670	6,790	5,910
HTR10	8,560	8,560	8,560	4,460	4,460	4,460	8,560	8,560	8,560
HTR15	13,790	12,220	10,650	12,300	11,810	11,330	25,240	22,360	19,490
HTR30	15,360	15,360	15,360	12,780	12,780	12,780	28,110	28,110	28,110
HTR22	13,010	10,650	—	12,060	11,330	—	23,800	19,490	—
HTR45	15,360	15,360	—	12,780	12,780	—	28,110	28,110	—
HTR75	16,340	10,060	3,770	16,540	14,060	11,570	65,200	40,140	15,040
HTR150	22,620	22,620	22,620	19,020	19,020	19,020	90,250	90,250[2]	90,250[2]
HTR300	29,050	14,910	770	24,090	19,710	15,340	172,850	88,710	4,580
HTR600	43,190	43,190	43,190	28,460	28,460	28,460	256,980	256,980	256,980[2]

E

PROXIMITY SWITCHES

The inductive type proximity switch provides end of rotation indication. The non-contact probe senses the presence of the ferrous cushion spear and has no springs, plungers, cams or dynamic seals that can wear out or go out of adjustment. The switch is solid state and meets NEMA 1, 12 & 13 specifications. For ease of wiring the connector housing is rotatable through 360°. To rotate, lift the cover latch, position, and release.

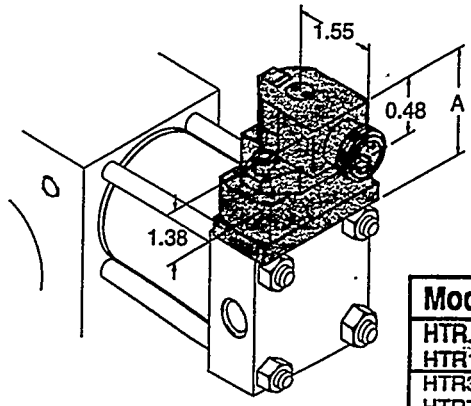
The standard proximity switch controls 20-230 VAC/DC loads from 5 to 500 mA. The low 1.7 mA off-state leakage current can allow use for direct PLC input. The standard short circuit protection (SCP) protects the switch from a short in the load or line upon sensing such a condition (5 amp or greater current) by assuming a non-conductive mode. The fault condition must be corrected and the power removed to reset the switch preventing automatic restarts.

The low voltage DC switch is also available for use with 10-30 VDC. This switch is in a non-rotatable housing, but does incorporate the short circuit protection.

Both switches are equipped with two LEDs, "Ready" and "Target". The "Ready" LED is lit when power is applied and the cushion spear is not present. The "Target" LED will light and the "Ready" LED will go out when the switch is closed, indicating the presence of the cushion spear. Both LEDs flashing indicates a short circuit condition.

NOTES:

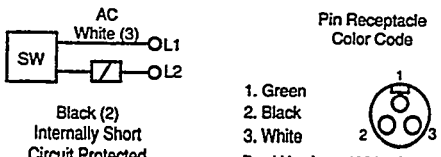
1. Available with or without cushions.
2. Not available with stroke adjusters.
3. Pressure rating: 1500 psi
4. Operating temperature: -4°F to 158°F
5. Specify switch type, orientation and voltage when ordering.
6. The low voltage DC switch is available in non-rotatable style only, consult representative for further information.



Model	A
HTR.9 HTR1.8	1.91
HTR3.7 HTR7.5	2.44
HTR5 HTR10	2.44
HTR15 HTR30	2.31
HTR22 HTR45	2.06
HTR75 HTR150	1.56
HTR300 HTR600	3.42

HIGH VOLTAGE SWITCH (20-230 VAC/DC)

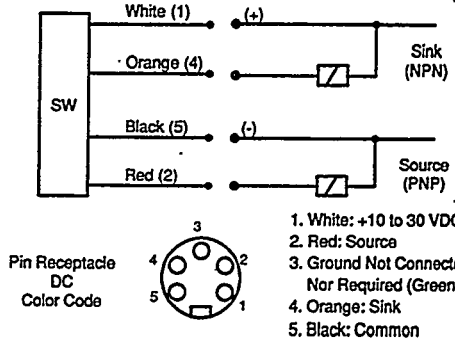
Wiring Diagram and Information



Brad Harrison 40909 Connector

LOW VOLTAGE DC SWITCH (10-30 VDC)

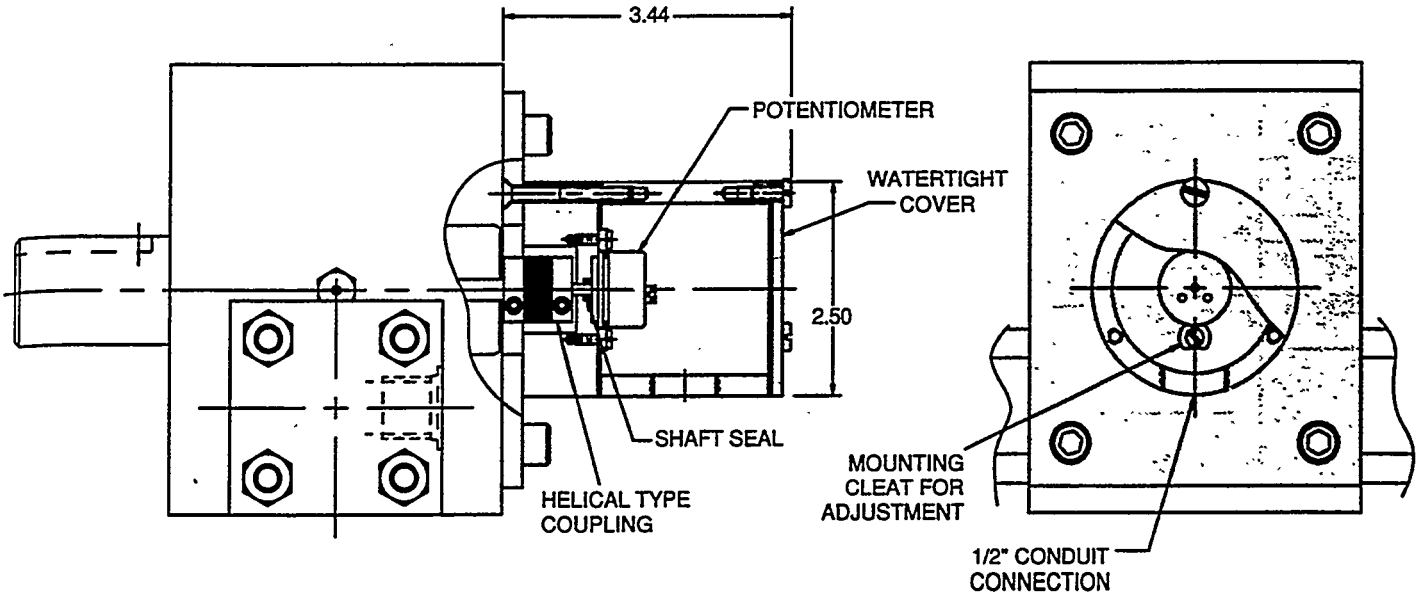
Wiring Diagram and Information



Brad Harrison 41310 Connector

NOTES:

PRECISION FEEDBACK POTENTIOMETER PACKAGE



APPLICATION:

The potentiometer option is an analog feedback device designed for use on a wide variety of applications that utilize closed-loop feedback to acheive accurate position, velocity, or motion control. The potentiometer option can also be used on open-loop systems as a continuous, infinite position monitoring device.

OPERATION:

A potentiometer is a variable resistor. There are three electrical terminals on the potentiometer - two on opposite ends of a fixed conductive plastic element (terminals 1 and 3, see electrical schematic on the right) and one attached to the "wiper" that moves along the conductive element as the shaft rotates (terminal 2). As the potentiometer shaft rotates, resistance between terminal 2 and terminals 1 and 3 charges.

Because the resistance of the potentiometer is linear (± .1%), when a fixed voltage level is supplied across terminals 1 and 3, shaft position can be determined by reading the output voltage at terminal 2. By measuring the rate of change in voltage at terminal 2, rotational velocity can be determined. To determine actuator position or velocity, the potentiometer shaft is connected to the actuator shaft via a flexible coupling. The potentiometer is enclosed in a water tight enclosure for resistance to dirty environments.

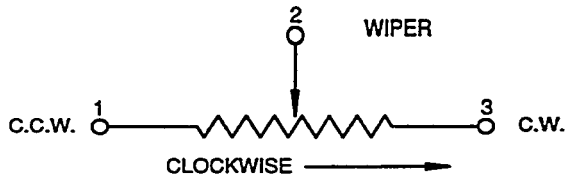
ORDERING INFORMATION:

Consult the factory or your local representative for specific application and ordering information.

POTENTIOMETER SPECIFICATIONS:

• Potentiometer:	7/8" diameter single turn precision servo-mount with conductive plastic element
• Resistance:	10K ohms
• Resistance Tolerance:	± 10%
• Linearity:	± 1%
• Resolution:	Essentially infinite
• Effective electrical angle:	340° ± 3%
• Power rating:	70°C - 1 watt 125°C - 0 watt
• Temperature Rating:	± 1°C to + 125°C
• Backlash:	0.1° max.
• Rotational life:	20,000,000 rev.
• Terminal:	Rear, turret style
• Available on:	
Rack & Pinion* -	PTR/LTR Series HTR Series M Series
Vane -	Standard Series

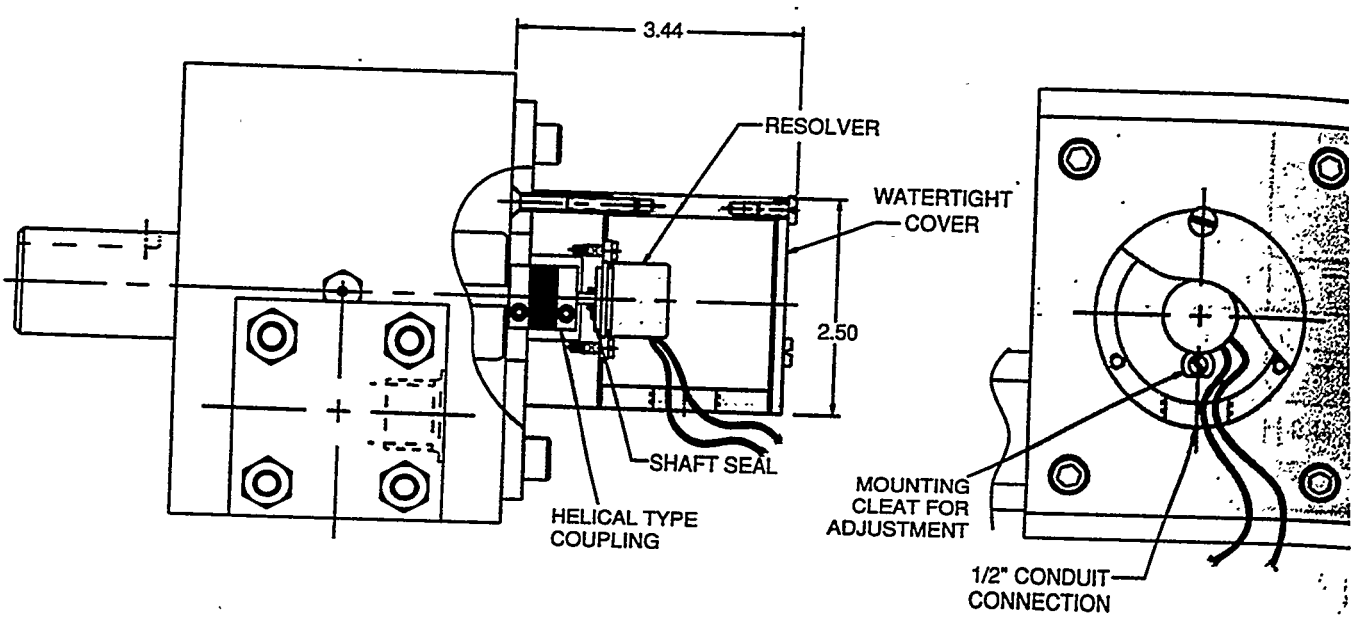
• Electrical Schematic:



NOTE:

*1. Rack and pinion actuators have a small amount of backlash- consider before applying.
2. Electrohydraulic options and operation may affect other actuator components such as seals, bearings, etc. Consult your local representative for additional information.

PRECISION RESOLVER FEEDBACK PACKAGE



APPLICATION:

The resolver option is a precision analog feedback device designed for use on a wide variety of applications that utilize closed-loop feedback to achieve accurate position, velocity, or motion control. The resolver option can also be used on open-loop systems as a continuous, infinite position monitoring device.

OPERATION:

A resolver is a brushless rotary sychronous transformer which eliminates wiping contact and provides higher response and accracy. A voltage and frequency is applied across the primary coil; changes in shaft position cause a proportional change in the secondary winding. Measuring the rate of change also provides velocity. To determine actuator position or velocity, the potentiometer shaft is connected to the actuator shaft via a flexible coupling. The potentiometer is enclosed in a water tight enclosure for resistance to dirty environments.

ORDERING INFORMATION:

Actuators can be supplied as complete packages, or with an enclosure for the mounting of a customer supplied resolver. Consult the factory or your local representative for specific application and ordering information.

RESOLVER SPECIFICATIONS:

• Resolver:	11D servo-mount, brushless synchro
• Input Voltage:	7.5 volts
• Input Frequency:	4000 Hz
• Input Current:	13.5 mA (max)
• Input Power:	60 mW
• Output Voltage:	4 volts
• Phase Shift:	0
• Sensitivity:	70 mV/°
• DC Rotor Resistance:	8 ohms
• DC Stator Resistance:	19 ohms
• Total Null Voltage:	10 mV
• Accuracy (Maximum error):	7 minutes*
• Leads:	#28 AWG, 12 inches long
• Available on:	
Rack & Pinion* -	PTR/LTR Series HTR Series M Series
Vane -	Standard Series

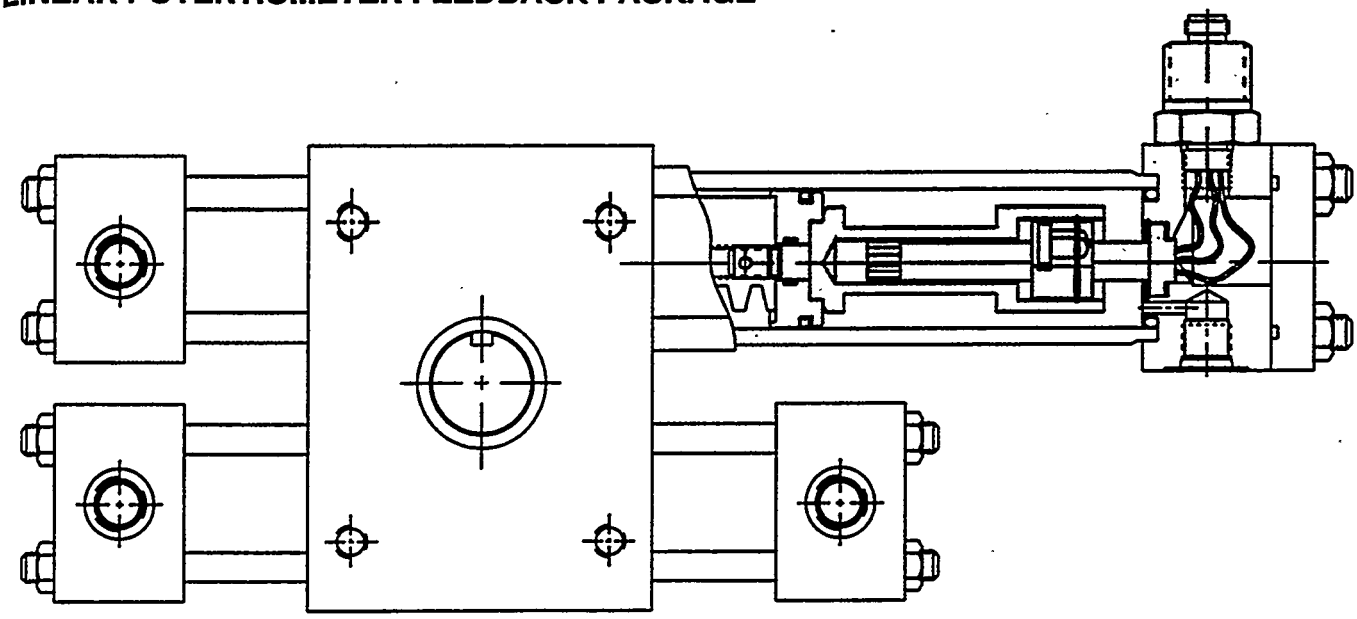
NOTE:

*1. Rack and pinion actuators have a small amount of backlash-consider before applying.

2. Electrohydraulic options and operation may affect other actuator components such as seals, bearings, etc. Consult your local representative for additional information.

For additional information – call your local Parker Fluidpower Motion & Control Distributor.

LINEAR POTENTIOMETER FEEDBACK PACKAGE



APPLICATION:

The linear potentiometer option is an analog feedback device typically used for rotary actuator rotations exceeding 340°, or where a standard rotary potentiometer cannot be connected to the actuator shaft. Applications for this device include closed loop feedback to achieve accurate position, velocity, or motion control. This option can also be used on open-loop systems as continuous, infinite position monitoring device.

OPERATION:

A potentiometer is a variable resistor. There are three electrical terminals on the potentiometer - two on opposite ends of a fixed conductive plastic element (terminals 1 and 3, see electrical schematic on the right) and one attached to the "wiper" that moves along the conductive element with the piston and rack. As the wiper moves along the element, resistance between terminal 2 and terminals 1 and 3 charges.

Because the resistance of the potentiometer is linear (± .1%), when a fixed voltage level is supplied across terminals 1 and 3, shaft position can be determined by reading the output voltage at terminal 2. By measuring the rate of change in voltage at terminal 2, rotational velocity can be determined. Since the linear potentiometer is enclosed in the actuator itself, it is safe for operation in dirty or wet environments.

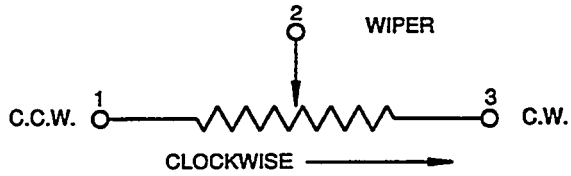
ORDERING INFORMATION:

Consult the factory or your local representative for specific application and ordering information.

LINEAR POTENTIOMETER SPECIFICATIONS:

• Potentiometer:	Calibrated conductive film with low resistance wiper carnage.
• Resistance:	Approx. 1 K ohm per inch of stroke
• ResistanceTolerance:	± 20%
• Linearity:	0.1% of full stroke
• Voltage:	5 - 50vdc
• Resolution:	Essentially infinite*
• Repeatability:	0.001*
• Power rating:	70°C - 1 watt/cm^2 125°C - 0 watt/cm^2
• Temperature Rating:	-55°C to 125°C
• Pressure Rating:	3,000 psi
• Terminal:	3 pin micro connector
• Available on:	
Rack & Pinion* -	PTR/LTR Series HTR Series M Series

• Electrical Schematic:

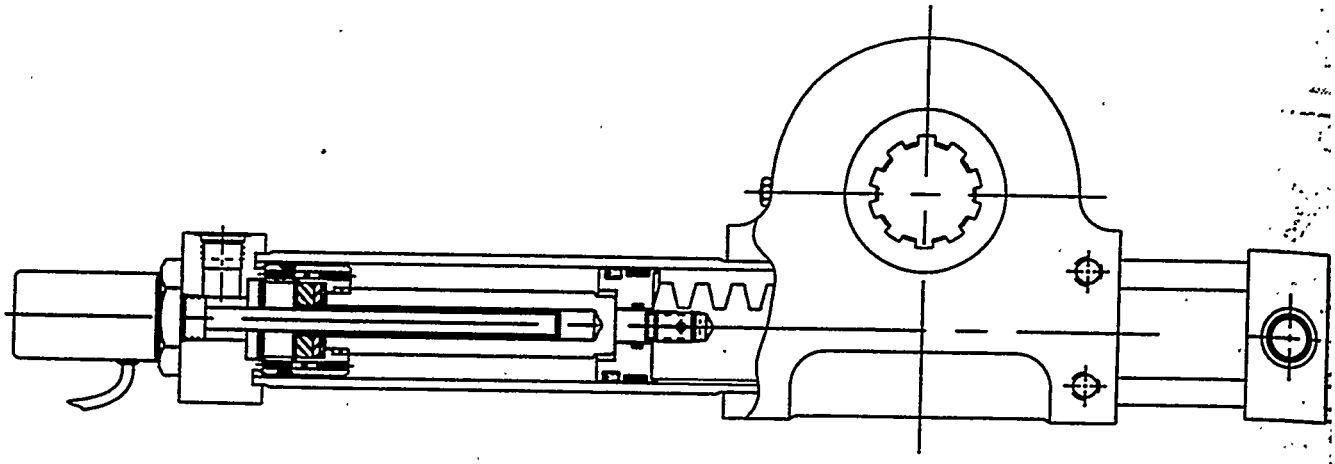


NOTE:

*1. Rack and pinion actuators have a small amount of backlash- consider before applying.
2. Electrohydraulic options and operation may affect other actuator components such as seals, bearings, etc. Consult your local representative for additional information.

E

LDT FEEDBACK PACKAGE



APPLICATION:

The linear displacement transducer (LDT) option provides digital feedback of the rack position via interaction between two magnetic fields. The LDT option is recommended for applications where rotation exceeds 340°, or where a standard rotary potentiometer or resolver cannot be connected to the actuator output shaft. Applications for this device include closed loop feedback to achieve accurate position, velocity, or motion control. This option can also be used on open-loop systems as a continuous, infinite position monitoring device.

OPERATION:

The LDT utilizes two magnets, a permanent one located on the piston, the other is a magnetic pulse generated by a current pulse along a wire inside a waveguide tube. The interaction between the fields produces a strain pulse which travels down the waveguide tube and is sensed by a coil at the end of the device. Position and velocity of the permanent magnet is pinpointed by measuring the lapsed time between the launching of the current pulse and the arrival of the strain pulse. An interface box converts this information to a usable output form, either digital or analog.

ORDERING INFORMATION:

Actuators can be provided as complete packages or with drilling and mounting for customers purchased LDTs. Consult the factory or your local representative for specific application and ordering information.

LDT SPECIFICATIONS:

- Linearity: ±.05% of full stroke (min. ±.002 i
- Repeatability: ±.0001 in.*
- Temperature Coefficient (probe): <10 ppm/°F
- Temperature Coefficient (electronics box):
  - Digital: <10 ppm/°F
  - Analog: 55 ppm/°F
- Resolution*:
  - Digital: .004, .002, .001 or .0005 availabl
  - Analog: Stepless continuous output*
- Output:
  - Digital: Absolute, TTL compatible, paralle or serial
  - Analog; -10 to +10 Vdc and 20mA range
- Operating Temperature:
  - Digital: 0° to 150°F
  - Analog: 35° to 150°F
- Hysteresis: .0008 in.*
- Available on:
  - Rack & Pinion* - HTR Series, M Series

NOTE:

*1. Rack and pinion actuators have a small amount of backlash- consider before applying.

2. Electrohydraulic options and operation may affect other actuator components such as seals, bearings, etc. Consult your local representative for additional information.

For additional information – call your local Parker Fluidpower Motion & Control Distributor.

ımple: HTR150-1803-AB11V-C23

HTR | 150 – 180 | 3 | C – A | B | 1 | 1 | V – CXX

ies
R - Heavy Duty Hydraulic Tie Rod Series

e
que Output in 1000 lb-in at 3000 psi

Single Rack Units	Double Rack Units
.9 = 900 lb-in	1.8 = 1,800 lb-in
3.7 = 3,700 lb-in	7.5 = 7,500 lb-in
5 = 5,000 lb-in	10 = 10,000 lb-in
15 = 15,000 lb-in	30 = 30,000 lb-in
22 = 15,000 lb-in*	45 = 30,000 lb-in*
75 = 75,000 lb-in	150 = 150,000 lb-in
00 = 300,000 lb-in	600 = 600,000 lb-in

grees Rotation
0 = 90°
0 = 180°
360 = 360°
Specify Other Rotations

ıshion
mit - None (STD)
CW Rotation**
CCW Rotation**
3 - Both Rotations
4 - Four Cushions***
8 - High Performance Cushion***
9 - Special

roke Adjusters (see Stroke Adjuster Table)
mit - None
0-5° CW Rotation**^
0-5° CCW Rotation**^
0-5° Both Rotation^
D - 0-30° CW Rotation**^^
E - 0-30° CCW Rotation**^^
F - 0-30° Both Rotations^^
X - Special

ounting Style
- Face (STD)
B - Base Mount
P - Pilot Mount

haft Configuration
- Female Keyed
- Single Male Keyed (STD)
- Double Male Keyed
D - Female 10B Spline
E - Single Male 10B Spline
F - Double Male 10B Spline
X - Special

ort Type
- SAE Straight Thread (STD)
- NPTF
3 - Flange
9 - Special

'ort Location (see Port Location Table)
- Side: Position 1 (STD)
- Side: Positon 2
3 - Side: Position 3
4 - Side: Position 4
5 - End
9 - Special

Seals
Omit - Molythane/Nitrile (STD)
V - Viton
W - Carboxilated Nitrile

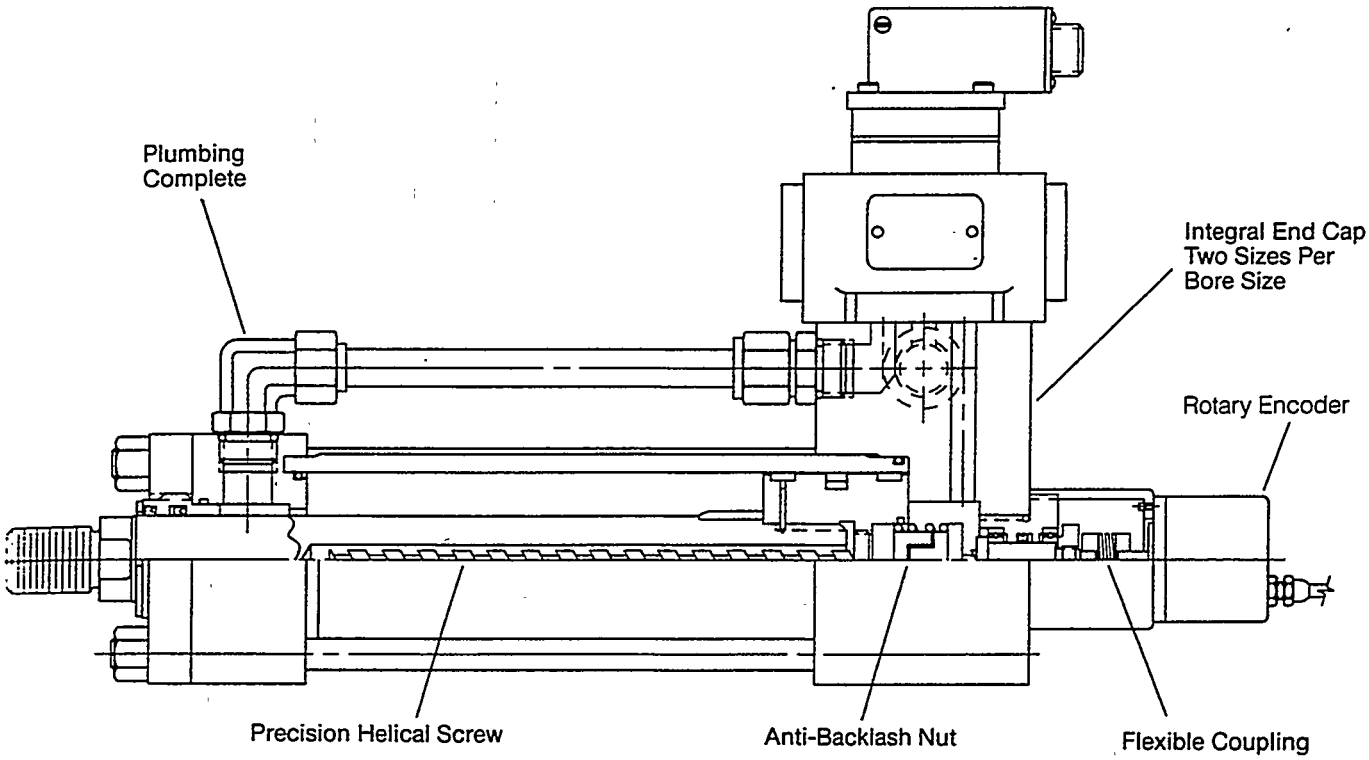
Special Options (Detail in Clear Text)
Air Bleeds
Proximity Switches
Feedback Potentiometer

Special Actuator Code
Omit - Standard Unit
(Two digit code assigned by factory and applies when any "X" or "9" appears in the model number or when special options or features are required.)

Design Series
Assigned by Factory

* HTR 22/45 rated to 2000 psi maximum
** Viewed from Shaft End
*** Double Rack Models Only. Use four cushions for existing applications only. For new applications, use option 8, High Performance Cushion. See options section for additional information.
^ Not available with End Ports. Standard 5° Stroke Adjusters and Standard Cushions available together on all units. Single rack units required additional "A" length.
^^ Not Available with End Ports or Cushions

Parker EH-IR Actuator
The Ultimate in Accurate, Dependable Linear Position



B

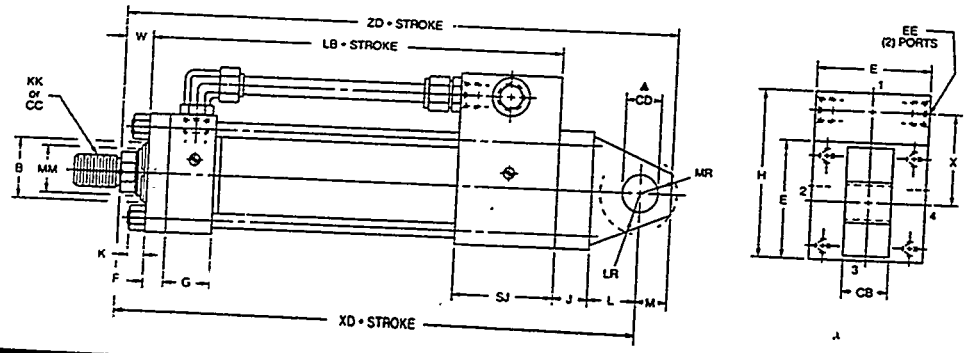
Here's How The Parker EH-IR Feeds Back Linear Position

The precision helical screw nut is attached to the cylinder piston. As the piston moves through its stroke, the nut "back-drives" the helical screw, rotating the screw within a set of precision thrust and radial bearings. The helical screw shaft is attached to the shaft of a rotary encoder with a flexible coupling. The linear motion of the cyclinder piston is converted to rotary motion by the precision helical screw.

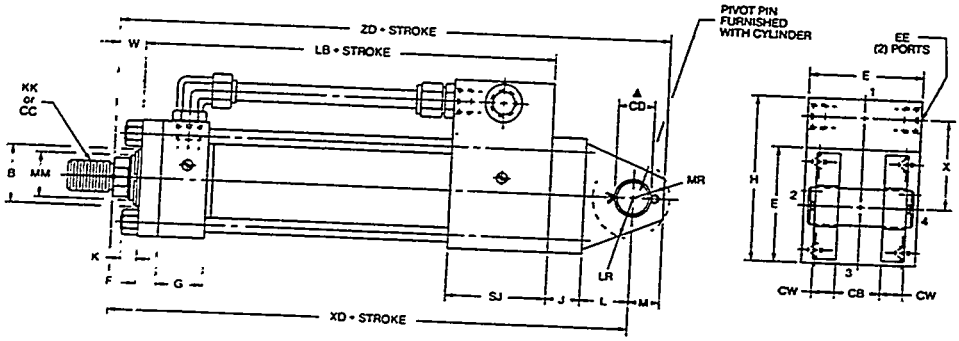
Standard Specifications and Features Include:

- Max. non-linearity (scale dependent) of .0001/inch
- Repeatability to ±.0005"
- Rotary output is one revolution per inch of stroke
- Mounting adapters to accept most rotary encoders, resolvers are available to provide an absolute or incremental feedback
- Complete factory testing of assembled unit-test report provided
- Unit flushed and certified to NAS-1638 particulate count Class 4
- Proven machine tool technology adapted to NFPA hydraulic cylinders for optimum performance and life
- Straight thread 'O'ring ports for "no leak" pressure and return hydraulic connections
- Encoder connector varies with the encoder order
- TTL compatible quadrature output is standard for position and velocity feedback
- Stroke length up to 48 inches
- Consult factory on velocities exceeding 20 inches per second

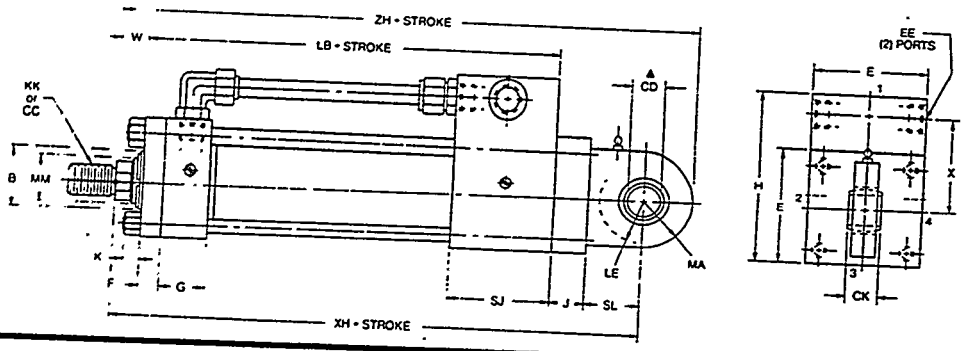
Cap Fixed Eye
Parker Style B with CLP Feedback Only



Cap Fixed Clevis
Parker Style BB with CLP Feedback Only



Cap Spherical Bearing
Parker Style SB with CLP Feedback only

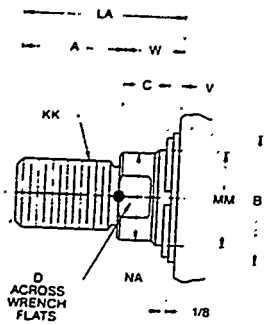


SB	PRESSURE RATING
2"	2000
2½"	1350
3¼"	1350
4"	1400
5"	1800
6"	1650

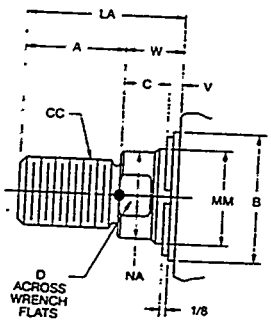
Pressure rating is for maximum life of cylinder and bearing based on dynamic load of commercial bearing.

Rod End Dimensions — See Table 2

Parker Thread Style 4



Parker Thread Style 8



"Special" THREAD STYLE 3

Special thread, extension, rod eye, blank, etc., are also available.

To order, specify "Style 3" and give desired dimensions for CC or KK, A and LA. If otherwise special, furnish dimensioned sketch. For female threads, see page 152.

For additional information – call your local Parker Fluidpower Motion & Control Distributor.

Parker Series EH
Electrohydraulic Actuator

Cap Fixed Eye
Cap Fixed Clevis
Cap Spherical Bearing
2"-8" Bore

TABLE 1 — ENVELOPE AND MOUNTING DIMENSIONS

BORE	E	SAE EE	F	G	H	J	K	L	M	X	LR	MR	CB	CW	+.000 CD▲ -.002	CK	MA	SL	LE	ADD STK. LB	SJ *	SJ **
2	3	#12	⅝	1¾	4⅞	1½	⅝	1¼	¾	2½	15/16	15/16	1¼	⅝	.751	21/32	1¼	1 3/16	⅞	6½	2⅞	NA
2½	3½	#12	⅝	1¾	5⅜	1½	⅝	1¼	¾	2¾	15/16	15/16	1¼	⅝	.751	21/32	1¼	1 3/16	⅞	6⅝	2⅞	4⅜
3¼	4½	#12	¾	2	6¾	1¾	¾	1½	1	3⅝	1 3/16	1 3/16	1½	¾	1.001	⅞	1½	1 9/16	1⅛	7¼	2⅞	4⅜
4	5	#12	⅞	2⅛	7¼	1¾	¾	2⅛	1⅜	3⅞	1¾	1 11/16	2	1	1.376	1 3/16	1¾	1 15/16	1½	7⅝	2⅞	4⅜
5	6½	#12	⅞	2⅛	8¾	1⅞	1	2¼	1¾	4⅝	1 13/16	2⅛	2½	1¼	1.751	1 17/32	2¼	2 7/16	1 15/16	7½	2⅞	4⅜
6	7½	#16	1	2¼	9¾	2¼	1⅛	2½	2	5	2 1/16	2½	2½	1¼	2.001	1¾	2¾	2 11/16	2 3/16	8⅜	2⅞	4⅜
7	8½	#16	1	3⅛	11	2⅝	1¼	3	2½	5¾	2 9/16	3	3	1½	2.501	—	—	—	—	9⅛	2⅞	4⅜
8	9½	#16	1	3½	12	3	1½	3¼	2¾	6¼	2⅞	3¼	3	1½	3.001	—	—	—	—	9¾	2⅞	4⅜

▲Dimension CD is pin diameter.

* For lower flow servo valves — see page 153, Group A, B, C, G.

** For higher flow servo valves — see page 153, Group D, E, F, H. Velocity of CLP actuators must not exceed 40 ips.

TABLE 2 — ROD END AND ENVELOPE DIMENSIONS

BORE	ROD NO.	ROD DIA. MM	THREAD CC	THREAD KK	A	+.000 B -.002	C	D	LA	NA	V	W	ADD STROKE XD	ADD STROKE XH	ADD STROKE ZD	ADD STROKE ZH
2	2	1⅜	1¼-12	1-14	1⅝	1.999	⅝	1⅛	2⅝	1 5/16	⅜	1	10¼	10 1/16	11	11 7/16
2½	2	1¾	1½-12	1¼-12	2	2.374	¾	1½	3¼	1 11/16	½	1¼	10⅝	11 9/16	11⅜	11 13/16
	3	1⅜	1¼-12	1-14	1⅝	1.999	⅝	1⅛	2⅝	1 5/16	⅜	1	10⅜	10 3/16	11⅛	11 9/16
3¼	1	1⅜	1¼-12	1-14	1⅝	1.999	⅝	1⅛	2½	1 5/16	¼	⅞	11⅜	11 7/16	12⅜	12 15/16
	2	2	1¾-12	1½-12	2¼	2.624	⅞	1 11/16	3½	1 15/16	⅜	1¼	11¾	11 13/16	12¾	13 5/16
	3	1¾	1½-12	1¼-12	2	2.374	¾	1½	3⅛	1 11/16	⅜	1⅛	11⅝	11 11/16	12⅝	13 3/16
4	1	1¾	1½-12	1¼-12	2	2.374	¾	1½	3	1 11/16	¼	1	12½	12 5/16	13⅞	14 1/16
	2	2½	2¼-12	1⅞-12	3	3.124	1	2 1/16	4⅜	2⅜	⅜	1⅜	12⅞	12 11/16	14¼	14 7/16
	3	2	1¾-12	1½-12	2¼	2.624	⅞	1 11/16	3⅜	1 15/16	¼	1⅛	12⅝	12 7/16	14	14 3/16
5	1	2	1¾-12	1½-12	2¼	2.624	⅞	1 11/16	3⅜	1 15/16	¼	1⅛	12¾	12 15/16	14½	15 3/16
	2	3½	3¼-12	2½-12	3½	4.249	1	3	4⅞	3⅜	⅜	1⅜	13	13 3/16	14¾	15 7/16
	3	2½	2¼-12	1⅞-12	3	3.124	1	2 1/16	4⅜	2⅜	⅜	1⅜	13	13 3/16	14¾	15 7/16
	4	3	2¾-12	2¼-12	3½	3.749	1	2⅝	4⅞	2⅞	⅜	1⅜	13	13 3/16	14¾	15 7/16
6	1	2½	2¼-12	1⅞-12	3	3.124	1	2 1/16	4¼	2⅜	¼	1¼	14⅜	14 9/16	16⅜	16 13/16
	2	4	3¾-12	3-12	4	4.749	1	3⅜	5¼	3⅞	¼	1¼	14⅜	14 9/16	16⅜	16 13/16
	3	3	2¾-12	2¼-12	3½	3.749	1	2⅝	4¾	2⅞	¼	1¼	14⅜	14 9/16	16⅜	16 13/16
	4	3½	3¼-12	2½-12	3½	4.249	1	3	4¾	3⅜	¼	1¼	14⅜	14 9/16	15⅜	16 13/16
7	1	3	2¾-12	2¼-12	3½	3.749	1	2⅝	4¾	2⅞	¼	1¼	16	—	18½	—
	2	5	4¾-12	3½-12	5	5.749	1	4¼	6¼	4⅞	¼	1¼	16	—	18½	—
	3	3½	3¼-12	2½-12	3½	4.249	1	3	4¾	3⅜	¼	1¼	16	—	18½	—
	4	4	3¾-12	3-12	4	4.749	1	3⅜	5¼	3⅞	¼	1¼	16	—	18½	—
	5	4½	4¼-12	3¼-12	4½	5.249	1	3⅞	5¾	4⅜	¼	1¼	16	—	18½	—
8	1	3½	3¼-12	2½-12	3½	4.249	1	3	4¾	3⅜	¼	1¼	17¼	—	20	—
	2	5½	5¼-12	4-12	5½	6.249	1	4⅜	6¾	5⅜	¼	1¼	17¼	—	20	—
	3	4	3¾-12	3-12	4	4.749	1	3⅜	5¼	3⅞	¼	1¼	17¼	—	20	—
	4	4½	4¼-12	3¼-12	4½	5.249	1	3⅞	5¾	4⅜	¼	1¼	17¼	—	20	—
	5	5	4¾-12	3½-12	5	5.749	1	4¼	6¼	4⅞	¼	1¼	17¼	—	20	—

Note: Electrical port or connector will be provided at position 1 of rear cap.

Cylinder Division
500 So. Wolf Road
Des Plaines, IL 60016
708/298-2400

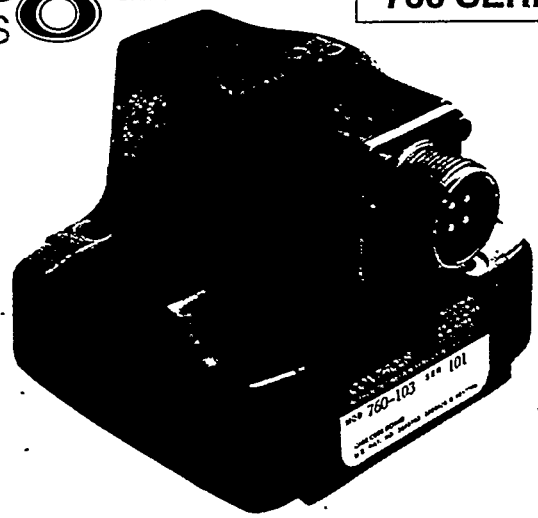
Regional Plants
- Santa Fe Springs, CA 310/698-0985
- Enfield, CT 203/749-2215
- Goodland, IN 219/297-3182
- Plymouth, MI 313/455-1700
- Hillsborough, NC (Accum) 919/732-9371
- Hillsborough, NC 919/732-9371
- Akron, OH 216/253-4500
- Portland, OR
- Owen Sound, Ont. Can. 519/371-2662
- Toronto, Ont. Can. 416/255-4567
- Lachine, Quebec, Can.

Moog CONTROLS

760 SERIES

760 Series Servovalves

The 760 Series is a high performance, two-stage design that covers the range of rated flows from 1 to 15 gpm at 1000 psi valve drop. The output stage is a closed center, four-way, sliding spool. The pilot stage is a symmetrical double-nozzle and flapper, driven by a double air gap, dry torque motor. Mechanical feedback of spool position is provided by a cantilever spring. The valve design is simple and rugged for dependable, long life operation.



Specifications:

Fluid Supply: 760 Series Servovalves are intended to operate with constant supply pressure.

Supply Pressure:
Minimum: 200 psi (14 bar)
Maximum Standard: 3500 psi (245 bar)
Maximum (special order): 5000 psi (350 bar)

Proof Pressure:
150% of supply pressure at P port
100% of supply pressure at R port

Fluid:
Compatible with common hydraulic fluids
Recommended viscosity range:
60-450 SUS @ 100°F
(10-97 cSt @ 38°C)

Cleanliness Level:
ISO DIS 4406 code 16/13 max.
14/11 recommended

Operating Temperature:
Minimum: -40°F (-40°C)
(maximum fluid viscosity: 6000 SUS)
Maximum: +275°F (+135°C)

Rated Flow Tolerance: ±10%

Symmetry: < 10%

Hysteresis: < 3%

Threshold: < 1/2%

Null Shift:
with temperature, 100°F variation: < 2%
with acceleration to 10g: < 2%
with supply pressure 1000 psi change: < 2%
with back pressure 0 to 500 psi: < 2%

Frequency Response: Typical response characteristics for 760 Series Servovalves are shown in Figures 1, 2, and 3.

Step Response: Typical transient responses of 760 Series Servovalves are shown in Figure 4. The straight line portion of the response represents saturation flow from the pilot stage which will increase with higher supply pressures.

Model	Response	Rated Flow (1000 psi)		Internal Leakage (1000 psi)		Rated Current (parallel coils)	Coil Nom. Resistance
		gpm	lpm	gpm	lpm	mA	ohms
760-100A	Std.	1	3.8	<0.17	<0.66	15	200
760-910A	High	1	3.8	<0.23	<0.92	40	80
760-101A	Std.	2.5	9.5	<0.22	<0.83	15	200
760-911A	High	2.5	9.5	<0.30	<1.15	40	80
760-102A	Std.	5	19	<0.35	<1.33	15	200
760-912A	High	5	19	<0.45	<1.78	40	80
760-103A	Std.	10	38	<0.35	<1.33	15	200
760-233A	High	10	38	<0.45	<1.78	40	80
760-104A	Std.	15	57	<0.35	<1.33	40	80
760-814A	High	15	57	<0.45	<1.78	40	80

Optional designs are available with intrinsically safe coils (FM approved), and/or special spool/bushing lap configurations. Available seal materials: BUNA (Std.), VITON or EPR.

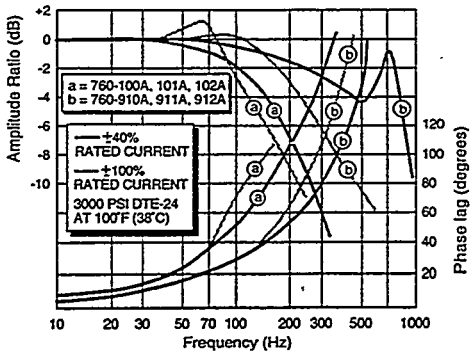


FIGURE 1 - FREQUENCY RESPONSE OF 1, 2-1/2 AND 5 GPM SERVOVALVES

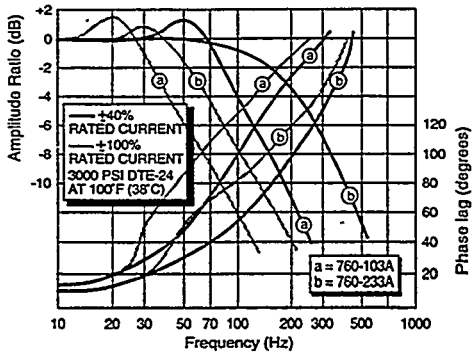


FIGURE 2 - FREQUENCY RESPONSE OF 10 GPM SERVOVALVES

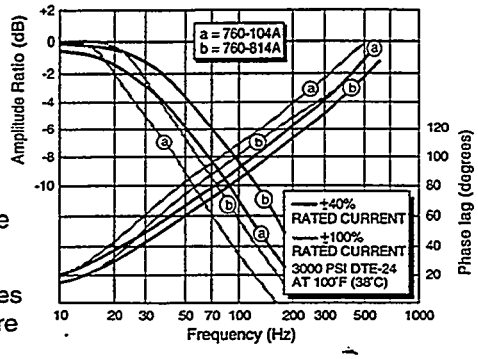


FIGURE 3 - FREQUENCY RESPONSE OF 15 GPM SERVOVALVES

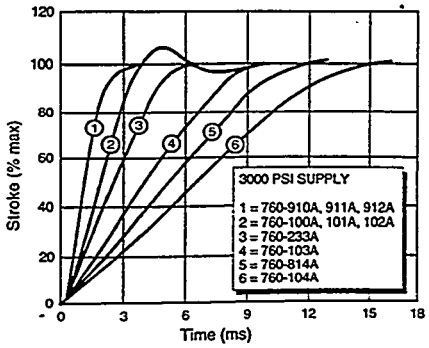


FIGURE 4 - STEP RESPONSES

A QUOTATION FROM

Hoyt N. Payne Company

301 East Depot Avenue
P.O. Box 338
Knoxville, TN 37901

Knoxville (615) 523-5184
Johnson City (615) 283-4244
Fax # (615) 523-1651

STOCKING DISTRIBUTOR OF MECHANICAL · FLUID POWER, MATERIAL HANDLING & ELECTRICAL PRODUCTS

Quotation # 1212

TO MARTIN MARIETTA ENERGY SYSTEMS
ROBOTICS & PROCESS CONTROL
ORNL
OAK RIDGE, TN 37830
ATTN: BRETT KARAUS

REF. NUMBER

DATE OF QUOTE 6-13-95

EXPIRATION OF QUOTE

F.O.B. SHIPPING POINT

TERMS NET 30 DAYS

QUANTITY	MATERIAL DESCRIPTION	PRICE
	WE ARE PLEASED TO QUOTE THE FOLLOWING FOR YOUR CONSIDERATION:	
1	PARKER HYDRAULIC CYLINDER MODEL 2"TD2HXKTS13A X 55" 2" BORE 1" ROD HIGH LOAD PISTON 7" STOP TUBE NET STROKE 48" LOW FRICTION ROD GLAND BOLT ON MANIFOLD FOR MOOG 760 PARKER GROUP A HEAD END PLUMBED TO CAP LA = 28" A = 1 1/8" KK = 3/4-16 MALE	$1167.80 EA.
1	PARKER HYDRAULIC CYLINDER MODEL 2"TD2HXKTS23A X 55" 2" BORE 1 3/8" ROD HIGH LOAD PISTON 7" STOP TUBE NET STROKE 48" LOW FRICTION ROD GLAND BOLT ON MANIFOLD FOR MOOG 760 PARKER GROUP A HEAD END PLUMBED TO CAP LA = 28" A = 1 5/8" KK = 1-14 MALE LDT WITH APM - 10VDC TO + VDC POSITION ONLY	3537.20 EA.
1	BALLUFF LINEAR DISPLACEMENT TRANSDUCER MODEL BTL-2-G21-1270-F-S50	781.00 EA.
1	BK-S50-05 CONNECTOR WITH 15 FT. CABLE	121.00 EA.

SHIPPING PROMISE
CYLINDER WITH TRANSDUCER 5 - 6 WEEKS
CYLINDER WITHOUT TRANSDUCER 3 - 4 WEEKS
TRANSDUCER 5 - 10 DAYS

HOYT N. PAYNE COMPANY

BY Jack Campbell
JACK CAMPBELL

1. This and other information from Hoyt N. Payne Co. Provide product or system options for further investigation by users having technical expertise before you select or use any product or system. It is important that you analyze all aspects of your application and review the information concerning the product in the current product catalog. The user, through its own analysis and testing, is solely responsible for making the final selection of the system and components and assuring that all performance safety and warning requirements of the application are met.
2. This quotation is made subject to the terms and conditions listed herein and the standard terms and conditions of the suppliers of the materials quoted. Copies available on request.
3. Sellers liability with respect to any item not of sellers manufacture shall be limited to that of the Vendor thereof.
4. Unless otherwise specified, prices quote herein are subject to revision to the prices which are in effect at the time shipment is made.
5. If an order is received for the items quoted herein, it will not be subject to cancellation without indemnifying both the suppliers of the materials quoted and the Hoyt N. Payne Co. against loss.
6. Unless specifically stated otherwise, in no event shall seller be liable here under or otherwise for loss of profits, special, incidental or consequential damages.

General Design

Balluff magnetostrictive linear displacement transducers offer non-contact, wearfree, and accurate linear positioning. They are designed for rugged industrial environments wherever linear motion must be controlled.

Balluff offers three housing configurations to suit most any application. The captive and floating magnet profile housings are designed to the same formats as linear potentiometers. The rod style is designed for hydraulic cylinder positioning or external mounting.

Linear stroke lengths range from 50 to 3606mm (2 in. to 142 in.).

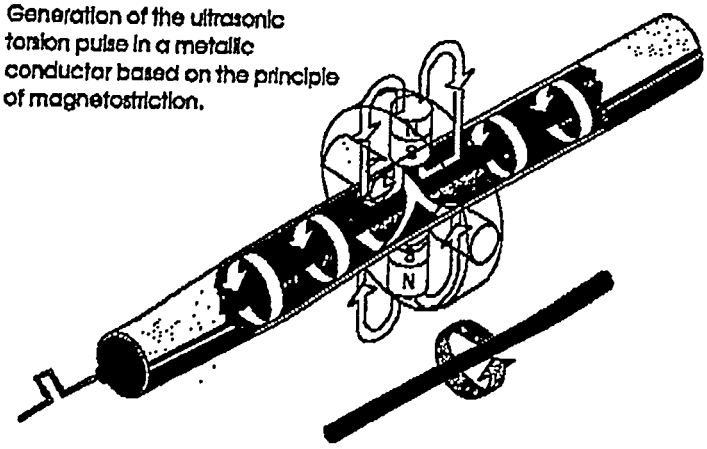
Available output signals range from analog DC voltage or current to pulse signal timing.

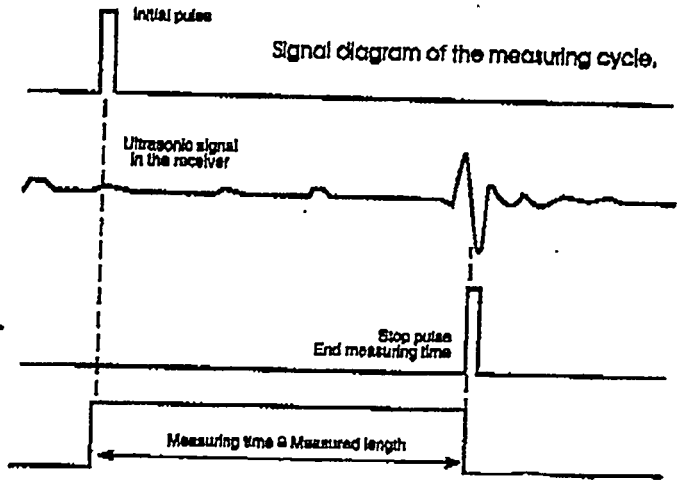
All BTL-2 transducers contain integrated surface mount electronics designed for high reliability and accuracy.

Principles of Operation

Although variously referred to as an "LVDT", an "LDT", or even a "pot", the accurate term for the Balluff BTL is a "magnetostrictive transducer." Magnetostriction, a physical effect first identified by the 19th century British physicist J.P. Joule, is defined as the cross-sectional and longitudinal deformation of a ferromagnetic material in the presence of a defined magnetic field. This property has been exploited in apparatus for generating ultrasound as well as devices for measuring quickly changing forces, such as strain gauges.

Generation of the ultrasonic torsion pulse in a metallic conductor based on the principle of magnetostriction.





Signal diagram of the measuring cycle.

In the BTL, the magnetostrictive element is an extremely small diameter (I.D. <0.0125") Ni-Fe alloy tube held in place inside a protective outer tube. This so-called waveguide runs the length of the transducer. To initiate a measurement for position update, a circuit in the housing pulses a current on a conductor wire which has been threaded coaxially through the waveguide. During the short time that this pulse is on, a rotating electromagnetic field surrounds the waveguide. At the same time, lines of field from magnets in the marker ring or block which has been attached externally to the moving member of the user's equipment come to focus on the waveguide. The effect of these two fields is to generate a magnetostrictive strain wave just below the magnets, which ripples back down the waveguide to a receiver in the transducer head. This mechanical pulse is converted into an electrical signal through a Balluff patented process. A high-speed clock or an integrator measures the time between launching of the current pulse and arrival of the torsional wave. Since the velocity of the torsion pulse (typ. 2850 m/s, or 8.9 µs/in.) is known as a material constant, and time has been determined by the clock or integrator, we now know distance. In essence, a sonic (or ultrasonic) delay line has been created which allows us to know the position of the magnet ring to a resolution of 2.5 µm (0.0001") depending on output signal type and stroke length. The user may specify an already scaled analog output signal, or just the raw timing pulses for processing the the controller.

2.3 Spring Loading or Tensioning

The transducer rod (flexible or rigid) can be spring loaded or tensioned using a stationary weight. Attach a spring mechanism or weight to the dead zone of the transducer rod with a clamping device which will not deform the transducer rod. The maximum weight or spring tension is 5 to 7 lbs.

2.4 Cylinder Installation

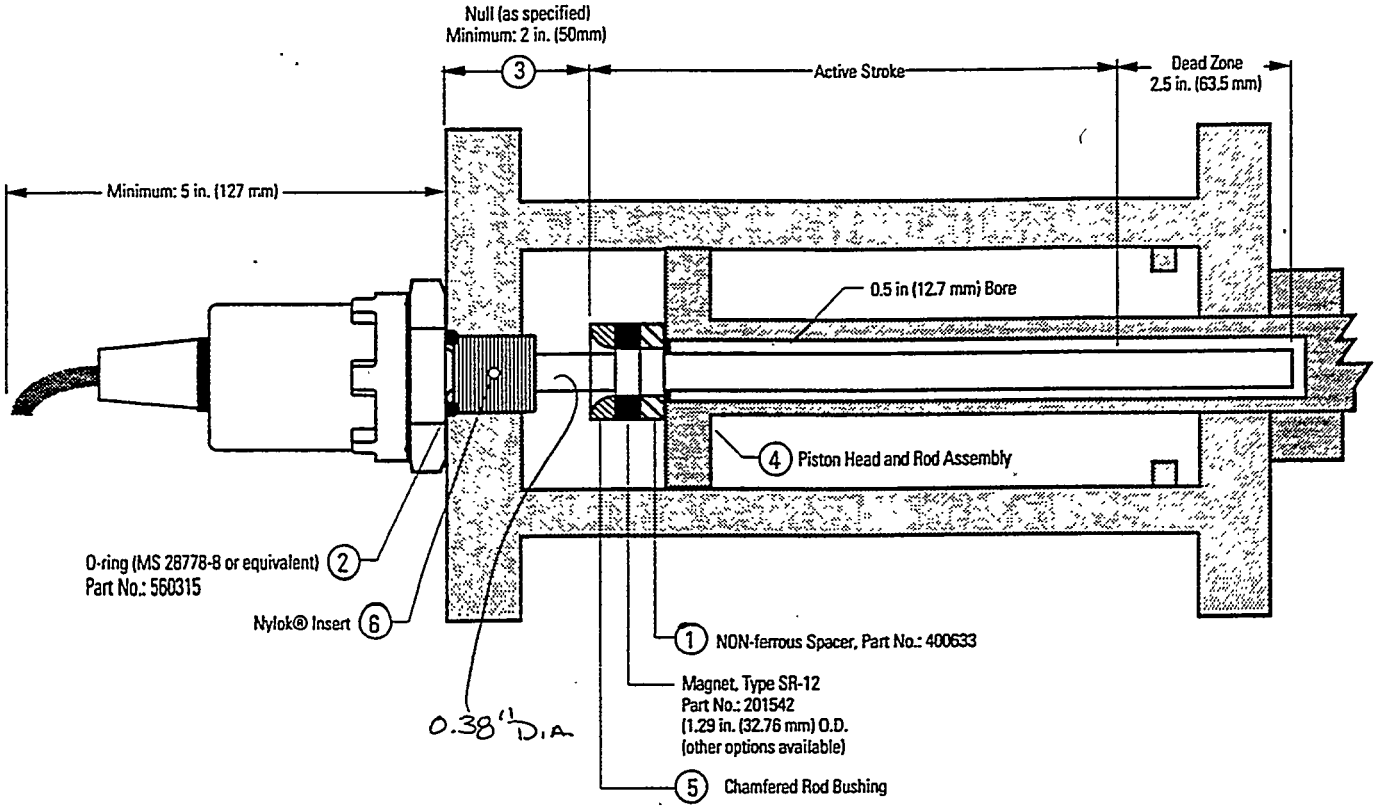
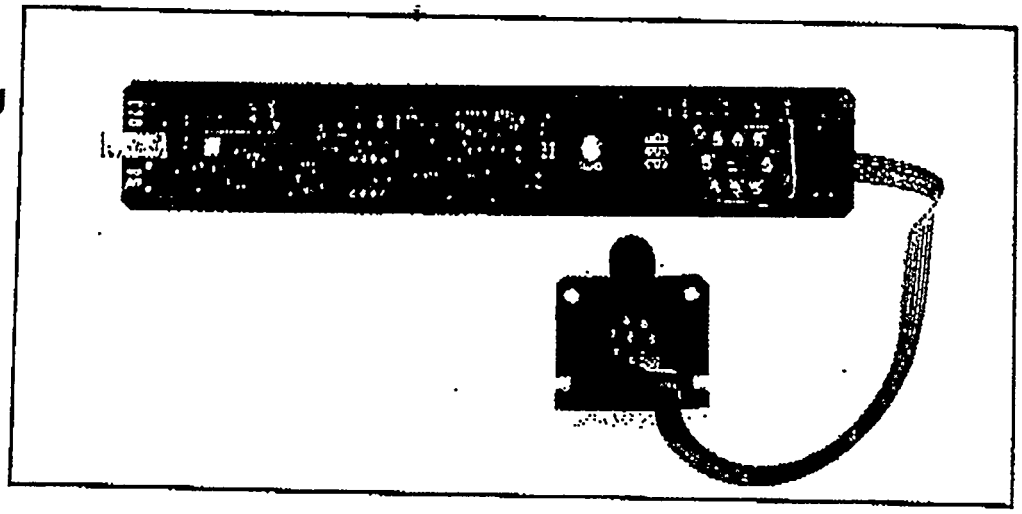


Figure 2-8
Typical Cylinder Installation

Figure 2-8 shows a typical cylinder installation. Review the following before attempting this type of installation.

- Use a non-ferrous (plastic, brass, Teflon®, etc.) spacer [1] to provide 1/8 inch (32 mm) minimum space between the magnet and the piston.
- An O-ring groove [2] is provided at the base of the transducer hex head for pressure sealing. MTS uses mil-standard MS33514 for the O-ring groove. Refer to mil-standard MS33649 or SAE J514 for machining of mating surfaces.
- The null space [3] is specified according to the installation design and cylinder dimensions. The analog output module provides a null adjustment. Make sure that the magnet can be mounted at the proper null position.
- The piston head [4] shown is typical. For some installations, depending on the clearances, it may be desired to countersink the magnet.

Highest Standards of Quality through Modern Manufacturing and Testing



The entire signal processing circuitry of the BTL Linear Displacment Transducer uses SMD technology on printed circuit boards.

The actual waveguide is shock mounted.

In the rod style the outer tube and the mounting flange are laser welded, and the stainless steel outer tube is rated to 8,700 PSI (600 bar).

Each and every BTL Linear Displacement Transducer undergoes a strict testing procedure after final assembly.

The following criteria are checked by the computer control:

- temperature characteristics
- temperature cycles from -20°C ...+60°C
- short circuit protection of the output drivers
- linearity of the output signal
- polarity reversal protection of the supply voltage
- overvoltage protection to 40 V
- dielectric strength between electronics and housing of 1 kV

The Advantages

Non-contact, reliable linear positioning
IP 67 and NEMA 6 industrial rated
Rod version rated to 600 bar (8700 PSI)
Various outputs - analog/pulse signals
Interface cards for various signal processing

Full Line of Accessories Available

Floats
Connectors
Special Magnets
Custom Mounts
Special connectors/cables
(see pages G42 - G48 for details)

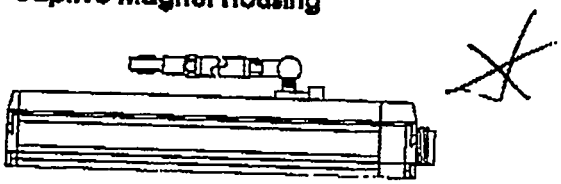
BALLUFF

Linear Displacement Transducer Series BTL-2

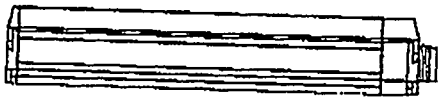
System Overview Analog Output

Housing Styles

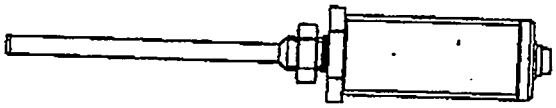
BTL-2- . . . F
Captive Magnet Housing



BTL-2- . . . P
Floating Magnet Housing



BTL-2- . . . Z
Rod Style Housing



Analog Output Types

'A' Analog Output (V)
Voltage Output Signal
0 ... 10 V
10 ... 0 V
• resolution = 0.1 mV
• total steps = 100,000

'B' Analog Output (V)
Voltage Output Signal
+5 ... -5 V
-5 ... +5 V
(0 V at center)
• resolution = 0.1 mV
• total steps = 100,000*

'G' Analog Output (V)
Voltage Output Signal
-10 ... +10 V
+10 ... -10 V
• resolution = 0.1 mV
• total steps = 100,000*

'C' Analog Output (I)
Current Output Signal
0 ... 20 mA
20 ... 0 mA
• resolution = < 1.0 µA
• total steps = 100,000*

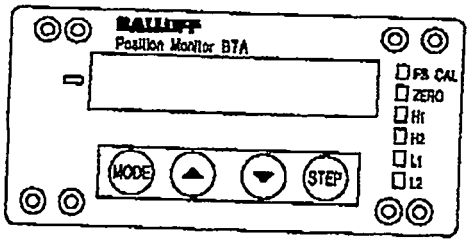
'E' Analog Output (I)
Current Output Signal
4 ... 20 mA
20 ... 4 mA
• resolution = < 1.0 µA
• total steps = 100,000*

*Positional accuracy = $\frac{\text{Full Scale Length}}{\text{total steps}}$

Example calculation = $\frac{610\text{ mm}}{100{,}000}$ = 6.1 µm = 0.00024 in

System Design

Balluff offers three BTL-2 housing configurations to suit most any application. All three housings can output any of the four analog signals shown. Internal SMT electronics convert absolute magnet positions to voltage or current output signals. Simple interfacing to many controls.



BTA-LIN

- 0...10 Volt or 4...20 mA Analog Input
- 5 digit, LED display
- Optional 4 setpoint
- 5 A relay output

BTL-2 Analog Output Series

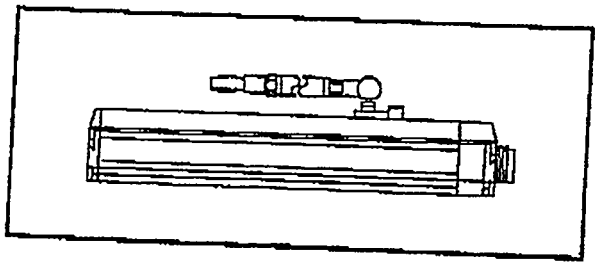
General Description

Balluff Linear Displacement Transducers offer non-contact, wearfree, and accurate linear positioning for industrial environments wherever linear motion must be controlled. The sensing element is enclosed in either an anodized stainless steel rod or an extruded aluminum housing. The absolute position is detected by a non-contact magnet that is attached to the moving member of the machine. The output is an analog signal, with no need to re-home after a power interruption.

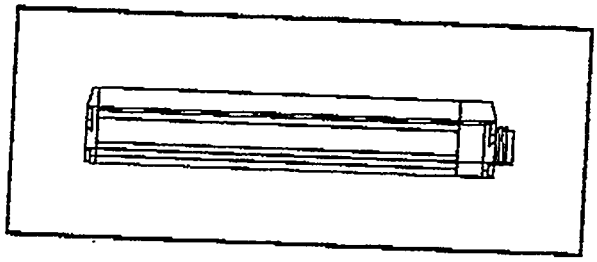
Housing Configurations

Shown below are the housing configurations available for the BTL-2 transducers. The BTL-2-...-Z "Rod Style" version is designed for installations in hydraulic and pneumatic cylinder applications. The BTL-2-...-F "Captive Magnet" and the BTL-2-...-P "Floating Magnet" versions are designed to the same formats as linear potentiometers for use in general positioning control.

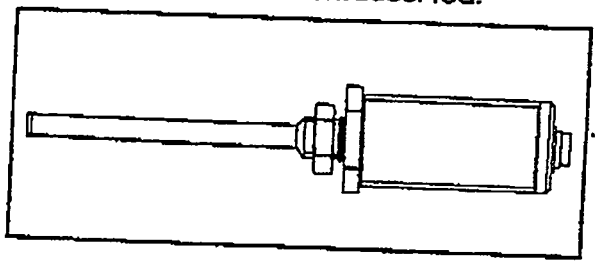
BTL-2-...-F: "Captive Magnet" version transducers are designed to be used when the mechanics of the system require a sliding motion for the point of contact for positioning. The magnet slides along the top of the transducer housing on special guides and connects to the system via a swivel ball joint.



BTL-2-...-P: "Floating Magnet" version transducers are designed to be used where the positioning mechanism can glide along a path above the transducer at a close range.



BTL-2-...-Z: "Rod Style" version transducers are designed to be used in hydraulic cylinder position control. The magnet (ring style) is attached directly to the piston face sliding over the transducer rod.



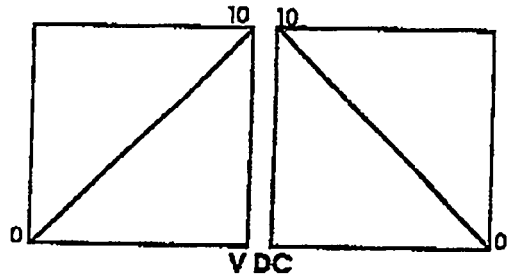
Output Signal Descriptions

Linear Displacement Transducer Series BTL-2

BALLUFF

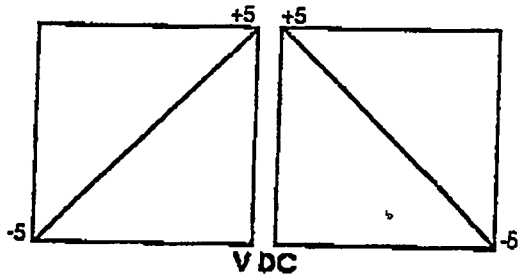
BTL2-A__ Signal Description

Dual analog voltage output signals are available with this version as : 0 ... 10, and 10 ... 0 V DC. Resolution is 0.001 % of Full Scale (10V) which gives the smallest measureable signal change as 0.1 mV DC. The output signals are valid across the length denoted by: "xxxx". Maximum current loading is 10 mA.



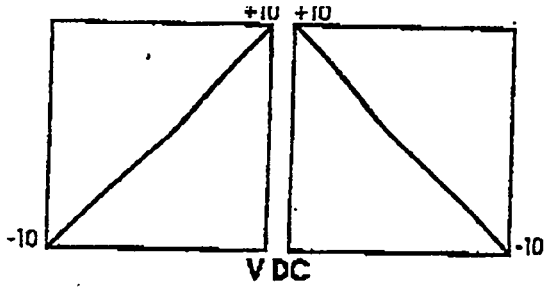
BTL2-B__

Dual analog voltage output signals are available with this version as: -5 ... +5, and +5 ... -5 V DC. Resolution is 0.001 % of Full Scale (10 V) which gives the smallest measurable signal change as 0.1 mV DC. The output signals are valid across the length denoted by: "XXXX". Maximum current loading is 10 mA.



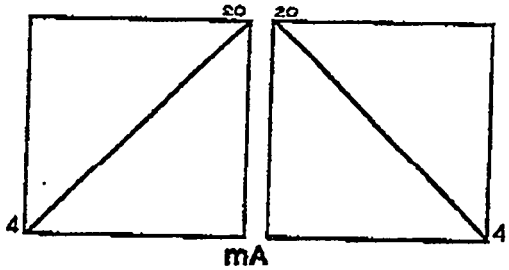
BTL2-G__

Dual analog voltage output signals are available with this version as:. -10 ... +10, and +10... -10 V DC. Resolution is 0.001 % of Full Scale (10 V) which gives the smallest measurable signal change as 0.1 mV DC. The output signals are valid across the length denoted by: "XXXX". Maximum current loading is 10 mA.



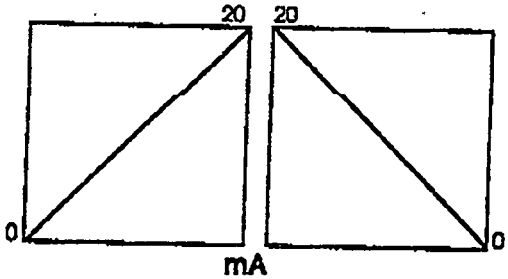
BTL2-C__

Analog current output signals are available with this version as either: 0 ... 20, or 20 ... 0 mA DC. Resolution is 0.001 % of Full Scale (20 mA) which gives the smallest measureable signal change as <1.0 µA DC. The output signals are valid across the length denoted by: "xxxx". Maximum current loading is 500 ohms.



BTL2-E__

Analog current output signals are available with this version as either: 4 ... 20, or 20 ... 4 mA DC. Resolution is 0.001 % of Full Scale (20 mA) which gives the smallest measureable signal change as <1.0 µA DC. The output signals are valid across the length denoted by: "xxxx". Maximum current loading is 500 ohms.



BALLUFF

Linear Displacement Transducer Series BTL-2

Analog Output Transducers Technical Data

Technical Specifications

pply Voltage (-__11) ...+24 V DC ± 10%
pply Voltage (-__21) ...±15 V DC ± 2%

urrent Draws:
L-2-__11-... ..... 130 mA (24 V DC)
-2-__21-... ..... 100/50 mA (±15 V DC)

n-linearity:
oke ≤500 mm) .....±150 µm (±0.01 in.)
oke >500 mm) .....±0.03% of Full Scale

solution (all types)

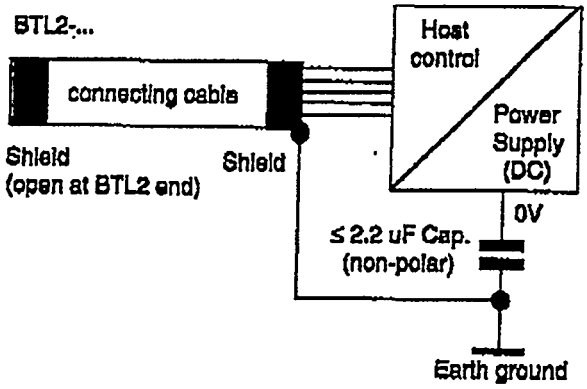
_(0...10 V types) .....=0.1 mV DC
_(-5...+5 V types).....=0.1 mV DC
_(0...20 mA types).....=<1.0 µA DC)
_(4...20 mA types).....=<1.0 µA DC)
teresis .....2µm (0.00008 in.)
peat accuracy .....Resolution + Hysteresis
mpling Rate ..... 1.0 kHz (per update)
erating Temp. .....-25...+80°C
rage Temp. .....-25...+125°C

Interface Information

The BTL-2 analog output series transducers interface with either analog voltage or current input control systems depending on model type. To retain the high resolution analog signals possible with the BTL-2 transducers, care should be taken when specifying a DC power supply and routing electrical cables. **The BTL-2 current output transducers (-C and -E types) offer higher noise immunity over the voltage output types.** The analog current output signals can be converted to a DC voltage signal by placing a load resistance of 500 ohms across the signal output at the control system's input terminals. This 500 ohm load would convert the 0...20 mA signal into 0...10 VDC at the control input end.

ble Shielding and Grounding

luff's standard cable is 22 /G multiconductor type with overall braided shield. To luce external noise, the cable eld must be properly used., nnect the cable shield to the ntrol's system GND. The cable eld is not connected at the -2 transducer end. Use a 2.2 non-polarized type capacitor suppress any noise that pears on signal lines. Always serve proper grounding hniques such as single point unding and isolating high lage (i.e. 120/240 VAC) from voltage (24 VDC) cables. diagram at right.



S50 Connector Pinouts for Analog Output Transducers - "Profile" Housing Style

For: BTL-2-A/B/G11 (+24V supply), and BTL-2-A/B/G21 (±15V supply)

Cable Wire Color	S50 Pin No.	Function	Function
		Voltage	Current
yellow	1	Not used	N/A
grey	2	Signal GND	N/A
pink	3	Vout falling	N/A
blue	4	Supply GND	N/A
brown	5	(+) 24V or 15V	N/A
green	6	Vout rising	N/A
white	center	(-) 15V	N/A

For: BTL-2-C/E10 (+24V supply), and BTL-2-C/E20 (±15V supply)

Cable Wire Color	S50 Pin No.	Function	Function
		Voltage	Current
yellow	1	N/A	0-20 or 4-20mA*
grey	2	N/A	Signal GND
pink	3	N/A	not used
blue	4	N/A	Supply GND
brown	5	N/A	(+) 24V or 15 V
green	6	N/A	not used
white.	center	N/A	(-) 15V

**rising only*

For: BTL-2-C/E17 (+24V supply), and BTL-2-C/E27 (±15V supply)

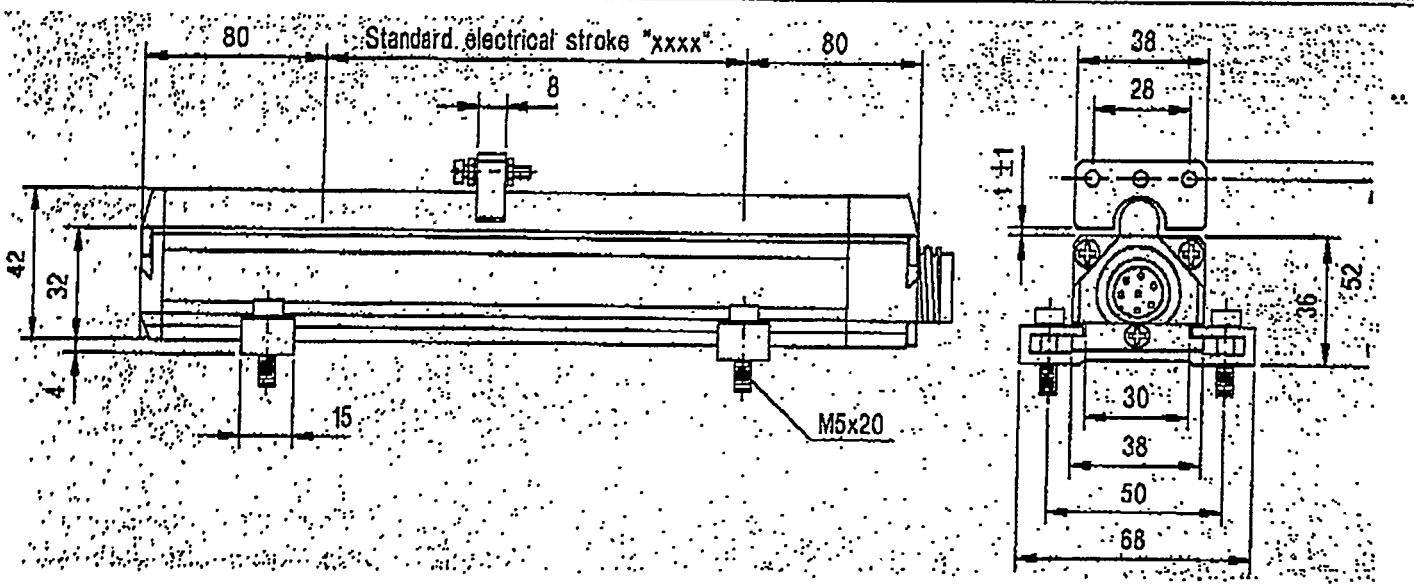
Cable Wire Color	S50 Pin No.	Function	Function
		Voltage	Current
yellow	1	N/A	20-0 or 20-4 mA*
grey	2	N/A	Signal GND
pink	3	N/A	not used
blue	4	N/A	Supply GND
brown	5	N/A	(+) 24V or 15V
green	6	N/A	not used
white	center	N/A	(-) 15V

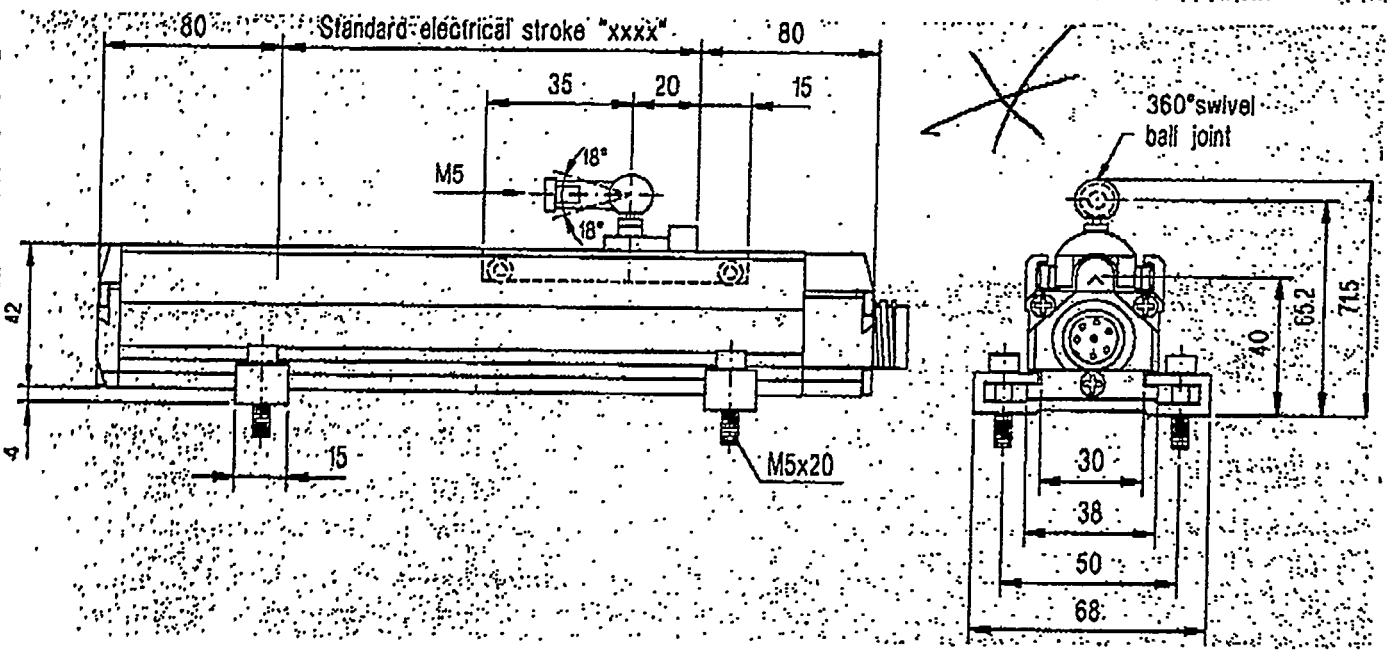
**falling only*

BALLUFF

Linear Displacement Transducer Series BTL-2

Analog Output Transducers Mechanical Specifications





Profile Styles* Standard Lengths (All Outputs)

Electrical Stroke (mm)	Electrical Stroke (in.)	Physical (mm)	Physical (in.)
0102	4.0	261.6	10.3
0152	6.0	312.4	12.3
0230	9.0	388.6	15.3
0254	10.0	414.0	16.3
0305	12.0	464.8	18.3
0381	15	541.0	21.3
0407	16.0	566.4	22.3
0457	18.0	617.2	24.3
0508	20.0	668.0	26.3
0610	24.0	769.6	30.3
0762	30.0	922.0	36.3
0914	36.0	1074.4	42.3

Electrical Stroke (mm)	Electrical Stroke (in.)	Physical (mm)	Physical (in.)
1016	40.0	1176.0	46.3
1270	50.0	1430.0	56.3
1524	60.0	1684.0	66.3
1778	70.0	1938.0	76.3
2032	80.0	2192.0	86.3
2286	90.0	2446.0	96.3
2540	100.0	2700.0	106.3
2794	110.0	2954.0	116.3
3048	120.0	3208.0	126.3
3302	130.0	3462.0	136.3
3606	142	3766.8	148.3

*(Captive and Floating Magnet Styles)

Analog Output Transducers Ordering Code

Linear Displacement Transducer Series BTL-2

BALLUFF

Ordering Code for Analog Output Transducers

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
B	T	L	–	2	–	A	1	1	–	0	3	0	5	–	Z	–	S	5	0		
																–	S	3	2		
																–	K	A	0	5	

Balluff - Transducer - Linear

Generation 2

Output Signal

A = 0...10 V
B = –5...+5 V
C = 0...20 mA
E = 4...20 mA
G = –10...+10 V

Supply Voltage

1 = 24 V ±10%
2 = ±15 V ±2%

Output Signal

If A, B, or G in position [7] :

1 = Vmin or Vmax at connector end, i.e. user selectable rising or falling*
0 = Vmin at connector end only, i.e. rising only**
7 = Vmax at connector end only, i.e. falling only**

If C or E in position [7] :

0 = Imin at connector end (rising towards opposite end)
7 = Imax at connector end (falling towards opposite end)

Nominal stroke in mm

0305 = 305 mm active electrical stroke

Housing geometry

B = Rod style with M18x1.5 thread and **reference point 30 mm from flange** (European standard)
F = Profile extrusion with captive magnet
P = Profile extrusion for floating magnet
Z = Rod style with 3/4"-16 UNF thread and **reference point 2 in. from flange** (USA standard)

Connection type

S	5	0	
S	3	2	
K	A	0	5

= connector with pin contacts, 7-pole, form 50 (not available for analog rod-style)
= connector with pin contacts, 8-pole, form 32 (not available on "F" or "P" style)
= integral axial cable (with 5 m cable)
only if 'B' or 'Z' in position [16], not available in "F" or "P" style

Notes:

*** Use "1" for new applications only, or consult factory if used as replacement**
**** Use "0" or "7" for exact pin compatibility with older BTL models. Examples:**
Exact replacement for BTL-A16-xxxx-Z-S32 is BTL-2-A10-xxxx-Z-S32
Exact replacement for BTL-A18-xxxx-Z-S32 is BTL-2-A17-xxxx-Z-S32
Exact replacement for BTL-E16-xxxx-Z-S32 is BTL-2-E10-xxxx-Z-S32
Exact replacement for BTL-E28-xxxx-Z-KA02 is BTL-2-E27-xxxx-Z-KA02

Magnet rings or blocks must be ordered separately, except "F" style profile, where Balluff p/n BTL-Z-2-F-2413-3 slide magnet assembly is included at no charge and does not have to be ordered separately.

Consult factory for special part numbers denoting outputs such as 0...5V, special null points, etc.

BALLUFF

Linear Displacement Transducer Series BTL-2

Analog Output Transducers Ordering Code Worksheet

Worksheet - Create your BTL-2 Part Number (for digital pulse output, see pages G28 and G29).

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
B	T	L	-	2	-				-					-		-					

Output Signal
Fill in A, B, C, E or G
This specifies the output range only, not whether increasing or decreasing with direction of motion. Position 9 will be used to define this specification.

Supply Voltage
Fill in 1 or 2
This specifies the supply voltage required.

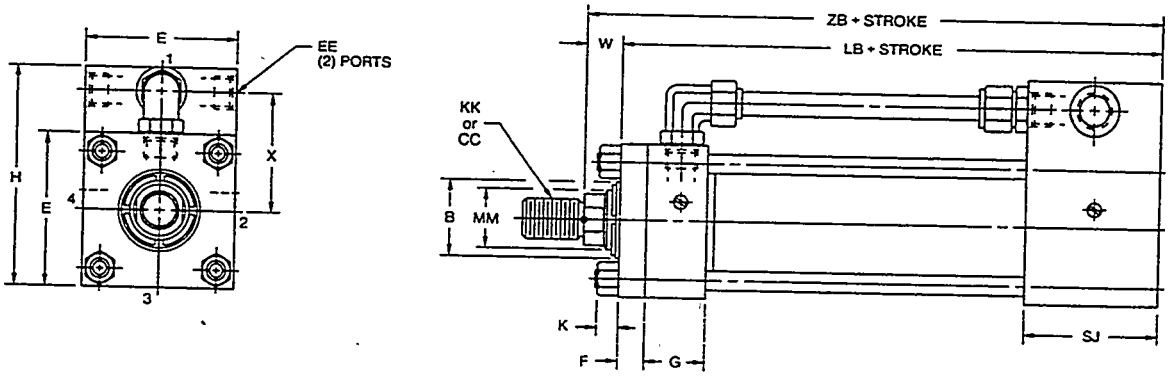
Output Orientation
Fill in 1, 0 or 7.
A "1" is only allowed for voltage output (A, B or G output). The "1" means that two pins (leads) have a position output signal; one carries the voltage increasing in value as the magnet moves away from the connector/cable end of the transducer, and the other carries the voltage decreasing in value as the magnet moves away from the connector/cable end. The user may use either output, but not both at the same time.
A "0" or "7" <u>must</u> be used to designate direction for current outputs (C or E), and <u>may</u> be used with voltage outputs if the BTL2 is replacing an older model (see notes previous page). If "0" or "7" is specified with voltage output, only one output signal is available (increasing or decreasing).

Nominal Stroke in mm
Fill in stroke length in mm, using leading zero if necessary.
See pages G16 and G17 for list of standard lengths available for both rod and profile style housings. The nominal stroke represents the electrical stroke, that is the travel distance over which the full output range is measured. For example, 0305 means that 0...10 V, or 4...20 mA etc. is output over a distance of 305 mm (12 inches). On either side of this nominal stroke area are zones where the magnet is normally not allowed to travel.

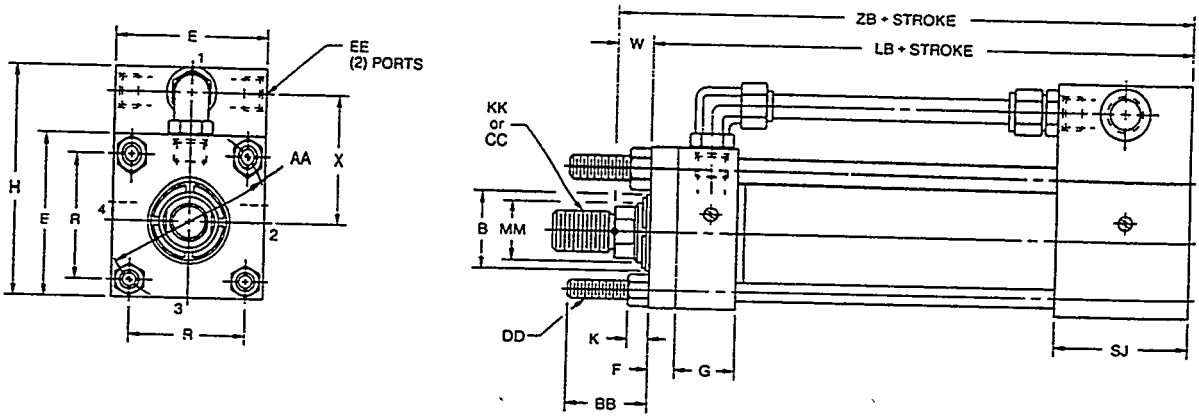
Housing Geometry
Fill in B, F, P or Z.
This specifies the housing style or mounting thread size. "B" and "Z" mean rod-style housing, M18 or 3/4"-16 UNF respectively. Standard (stocked) thread is 3/4-inch. "F" and "P" refer to "profile"-style housing, "F" means captive magnet version and "P" means floating magnet version.

Connection Type
Fill in S50, S32 or KAxx.
This specifies the connection method, either with quick-disconnect axial connector, or integral cable. For analog output, the rod-style is available with S32 connector or integral cable only (S50 not available). Digital pulse output rod-style is available with S50 or S32 connector or integral cable. Profile housing (digital pulse or analog) is available with S50 connector only (integral cable not available).

No Mount
Parker Style T — All Feedback Types

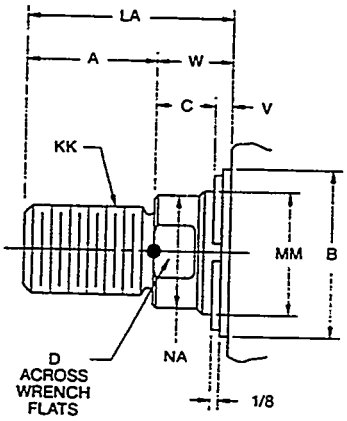


Tie-Rods Extended Head End
Parker Style TB — All Feedback Types

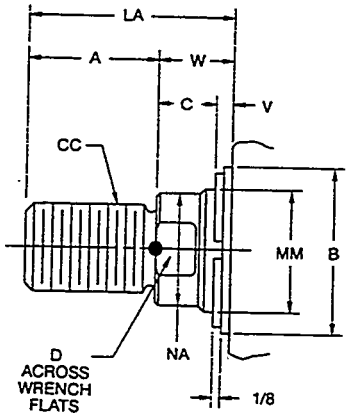


Rod End Dimensions — See Table 2

Parker Thread Style 4 (NFPA Style 5 M)



Parker Thread Style 8 (NFPA Style 1 M)



"Special" THREAD STYLE 3

Special thread, extension, rod eye, blank, etc., are also available.

To order, specify "Style 3" and give desired dimensions for CC or KK, A and LA. If otherwise special, furnish dimensioned sketch. For female threads, see page 152.

Hoyt N. Payne Company

301 East Depot Avenue
P.O. Box ~~XXX~~ 338
Knoxville, TN 37901

Knoxville (615) 523-5184
Johnson City (615) 283-4244
Fax # (615) 523-1561

STOCKING DISTRIBUTOR OF MECHANICAL - FLUID POWER, MATERIAL HANDLING & ELECTRICAL PRODUCTS.

SINCE 1936

Quotation # 1213

TO MARTIN MARIETTA ENERGY SYSTEMS
ORNL
ROBOTICS & PROCESS CONTROL
OAK RIDGE, TN 37830
ATTN: BRETT KARAUS

REF. NUMBER

DATE OF QUOTE 6-15-95

EXPIRATION OF QUOTE

F.O.B. SHIPPING POINT

TERMS NET 30 DAYS

QUANTITY	MATERIAL DESCRIPTION	PRICE
	WE ARE PLEASED TO QUOTE THE FOLLOWING FOR YOUR CONSIDERATION:	
1	PARKER HYDRAULIC ROTARY ACTUATOR MODEL #HTR30-360-AB11WX X = LDT FEED BACK PACKAGE CONSISTING OF TEMPOSONICS II LDT WITH APM -10VDC TO +10VDC POSITION ONLY. ALSO INCLUDES MATING CONNECTOR WITH 15 FT. CABLE FOR TRANSDUCER	$3637.00 EA.

SHIPPING PROMISE 4 TO 5 WEEKS

HOYT N. PAYNE COMPANY

BY JACK CAMPBELL

1. This and other information from Hoyt N. Payne Co. Provide product or system options for further investigation by users having technical expertise before you select or use any product or system. It is important that you analyze all aspects of your application and review the information concerning the product in the current product catalog. The user, through its own analysis and testing, is solely responsible for making the final selection of the system and components and assuring that all performance safety and warning requirements of the application are met.
2. This quotation is made subject to the terms and conditions listed herein and the standard terms and conditions of the suppliers of the materials quoted. Copies available on request.
3. Sellers liability with respect to any item not of sellers manufacture shall be limited to that of the Vendor thereof.
4. Unless otherwise specified, prices quote herein are subject to revision to the prices which are in effect at the time shipment is made.
5. If an order is received for the items quoted herein, it will not be subject to cancellation without indemnifying both the suppliers of the materials quoted and the Hoyt N. Payne Co. against loss.
6. Unless specifically stated otherwise, in no event shall seller be liable here under or otherwise for loss of profits, special, incidental or consequential damages.

Hoyt N. Payne Company

STOCKING DISTRIBUTOR OF MECHANICAL-FLUID POWER, MATERIAL HANDLING & ELECTRICAL PRODUCTS.

301 East Depot Avenue
P.O. Box 338
Knoxville, TN 37901

Telephone (615) 523-5184
FAX # (615) 523-1561

FAX TRANSMISSION

TO: ORNL
Brett Karaus

FROM: Jack Campbell
H.N.P. CO.

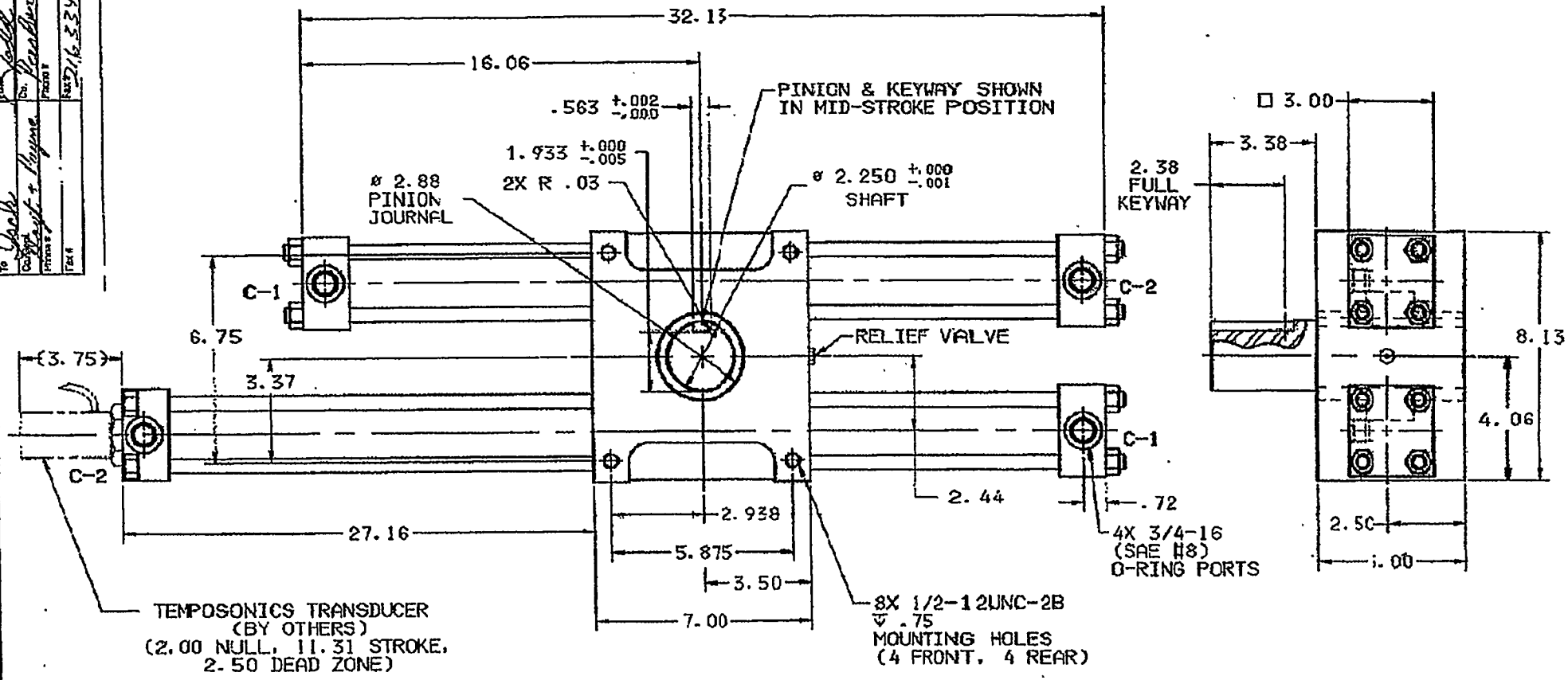
DATE: 7-24-95

TIME:

NO. OF PAGES 1 SENT
not including this sheet
if transmission is faulty,
please call.

Brett, Here is a Drawing on your rotary Actuator
Thought you may want it before it shows up

Jack



NOTES:

1. MAXIMUM OPERATING PRESSURE: 3,000 PSI (NON-SHOCK).
2. OUTPUT TORQUE @ 3,000 PSI: 30,000 IN-LBS.
3. ROTATION: 360° PLUS 2° MINUS 0°.
4. DISPLACEMENT: 71.0 IN³ PER 360° ROTATION.
5. PRESSURE ON C-1 PORTS: CLOCKWISE ROTATION.
   PRESSURE ON C-2 PORTS: COUNTERCLOCKWISE ROTATION.
6. ALL MOVING PARTS PRELUBRICATED.
7. UNIT EQUIPPED WITH TAPERED ROLLER BEARINGS.
8. UNIT EQUIPPED WITH CARBOXILATED NITRILE SEALS.

CUSTOMER: U.S. DEPT. OF ENERGY
P.O. NO.: 21100
A.A.D. Order No.: 041044
Sales Representative: HOYT + PAYNE Co.
Dwg. Certified By: [illegible] Date: 7-21-95

TITLE: Parker Series HTR Hydraulic Rotary Actuators

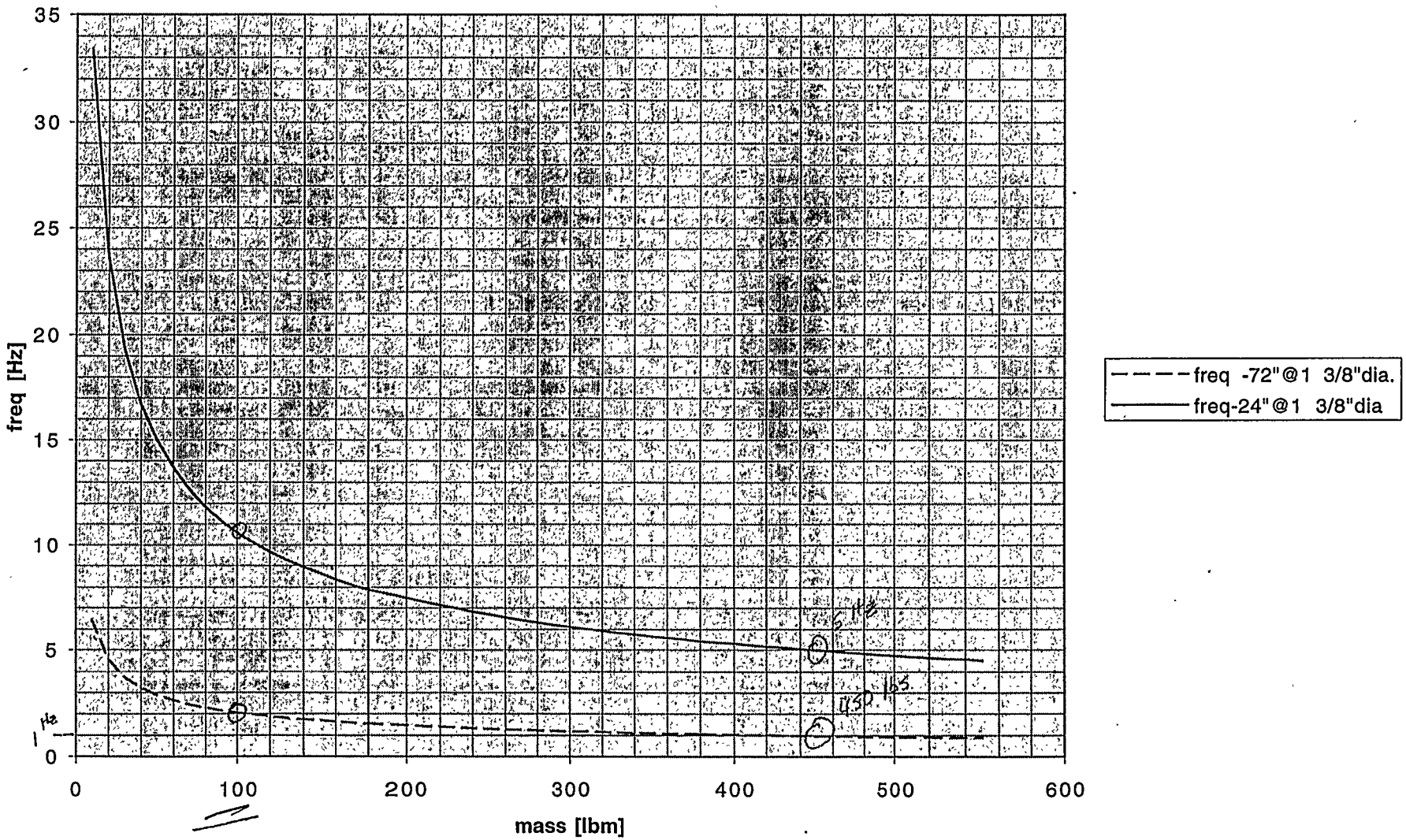
Parker — AUTOMATION ACTUATOR DIVISION, WADSWORTH, OHIO 44281

DRAWN BY: JBP
CHECKED BY: [illegible] 7-21-95
MATERIAL: N/A
SCALE: NONE
DRAWING NUMBER: HTR30-360-1B11W-C69
REV: —

brett 1 Chart 2

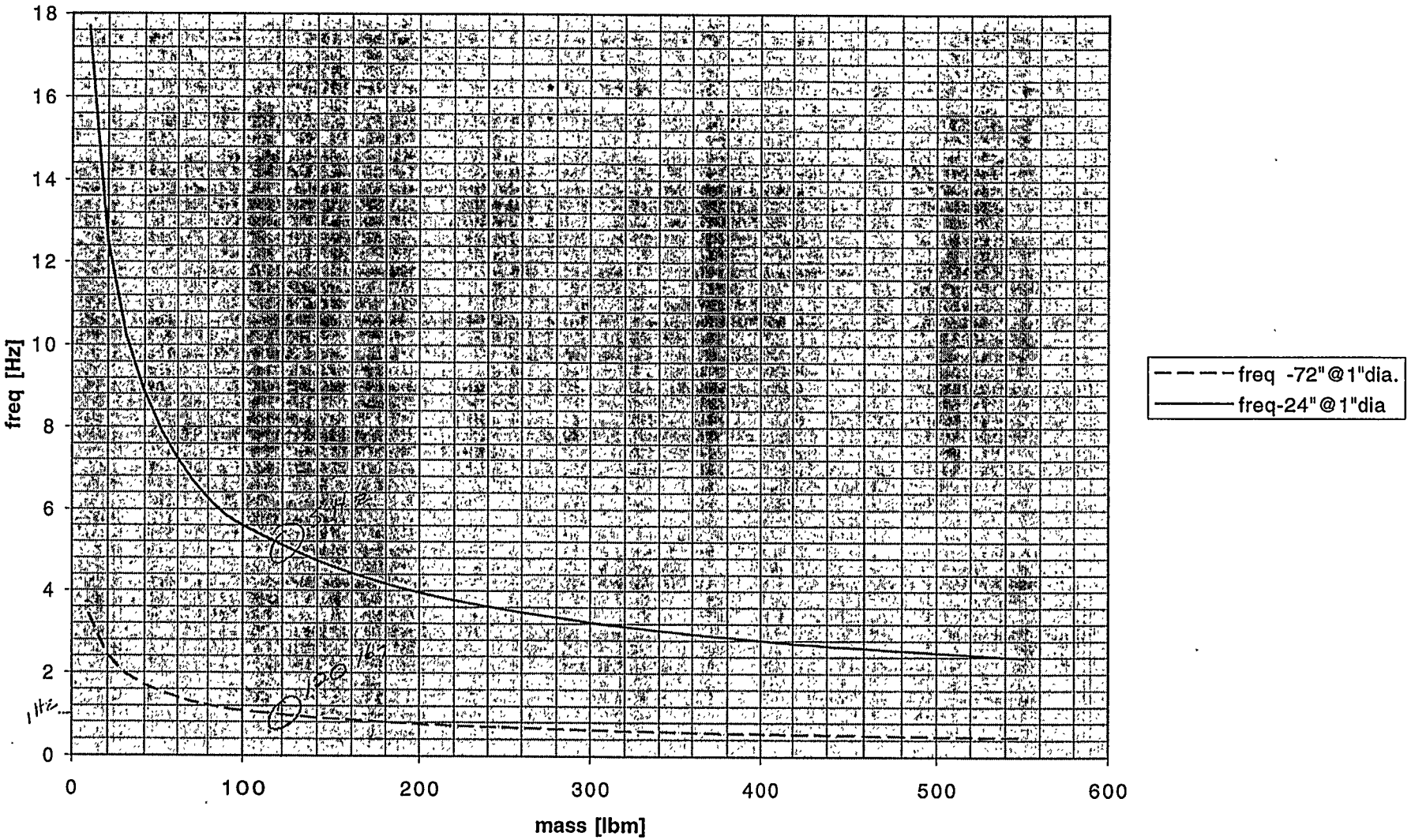
1 3/8" dia.

freq v/s end weight



1" dia

end mass v/s freq.



	density [lbm/i	0.28		extend lgth [in	72
	length [in]	96		E-beam [lb/in2	3.00E+07
	diameter [in]	1		Icsa [in4]	0.04908739
	mass [lbm]	21.1115026		end mass [lbm	120
	Ixx [lbm-in2]	2.63893783		K-beam [lb/in]	11.8362715
	tot move lgth	48		w [rad/sec]	6.17355605
	up time [s]	0.25		w [Hz]	0.98255196
	const time [s]	0.5		end deflect [in	10.1383278
	dwn time [s]	0.25			
	tot run time [s	1			
	Vmax [in/s]	64			
	rod accel [in/s	256		extend lgth [in	24
	force reqd [lb	5404.54467		E-beam [lb/in2	3.00E+07
	force req [lbf]	13.9869169		Icsa [in4]	0.04908739
				end mass [lbm	120
				K-beam [lb/in]	319.579331
				w [rad/sec]	32.0787382
				w [Hz]	5.10548975
				end deflect [in	0.37549362
	density [lbm/i	0.28		extend lgth [in	72
	length [in]	96		E-beam [lb/in2	3.00E+07
	diameter [in]	1.375		Icsa [in4]	0.17546104
	mass [lbm]	39.9139347		end mass [lbm	10
	Ixx [lbm-in2]	9.43278534		K-beam [lb/in]	42.3083133
	tot move lgth	48		w [rad/sec]	40.4325763
	up time [s]	0.25		w [Hz]	6.43504438
	const time [s]	0.5		end deflect [in	0.23636017
	dwn time [s]	0.25			
	tot run time [s	1			
	Vmax [in/s]	64			
	rod accel [in/s	256		extend lgth [in	24
	force reqd [lb	10217.9673		E-beam [lb/in2	3.00E+07
	force req [lbf]	26.4440147		Icsa [in4]	0.17546104
				end mass [lbm	10
				K-beam [lb/in]	1142.32446
				w [rad/sec]	210.093829
				w [Hz]	33.4374714
				end deflect [in	0.00875408

-3910

HOUGHTO - SAFE® 620

fire-resistant water-glycol hydraulic fluid gives long pump life

There is a hazard wherever heat, molten metal or flame are near hydraulic equipment. If a line or fitting ruptures, it can send a combustible spray of oil into the heat source. Severe injury and property damage may result.

To prevent this danger, fire-resistant fluids were developed and widely applied early in the 1950's. Houghton, a pioneer in fortifying petroleum hydraulic oils, was among the first to perfect fire-resistant fluids for hydraulic use.

Houghto-Safe 620 was developed as a maximum safety medium for use, under pressure, near open flame or high temperature areas.

Houghto-Safe 620 is a red, extremely "oily" hydraulic fluid that protects against corrosion and has additives to give long fluid life and excellent pump wear characteristics.

Houghto-Safe 620 gives maximum safety. In plant use it has prevented hundreds of fires and saved lives of workers. 620 is approved by Factory Mutual Engineering Laboratories as a less hazardous hydraulic fluid.

Houghto-Safe 620 provides high hydraulic efficiency through its exceptionally high viscosity index and is stable under high shearing stresses. Houghto-Safes are also available in other viscosities.

In accordance with accepted practice, it is recommended that bulk fluid temperatures be maintained below 120°F.

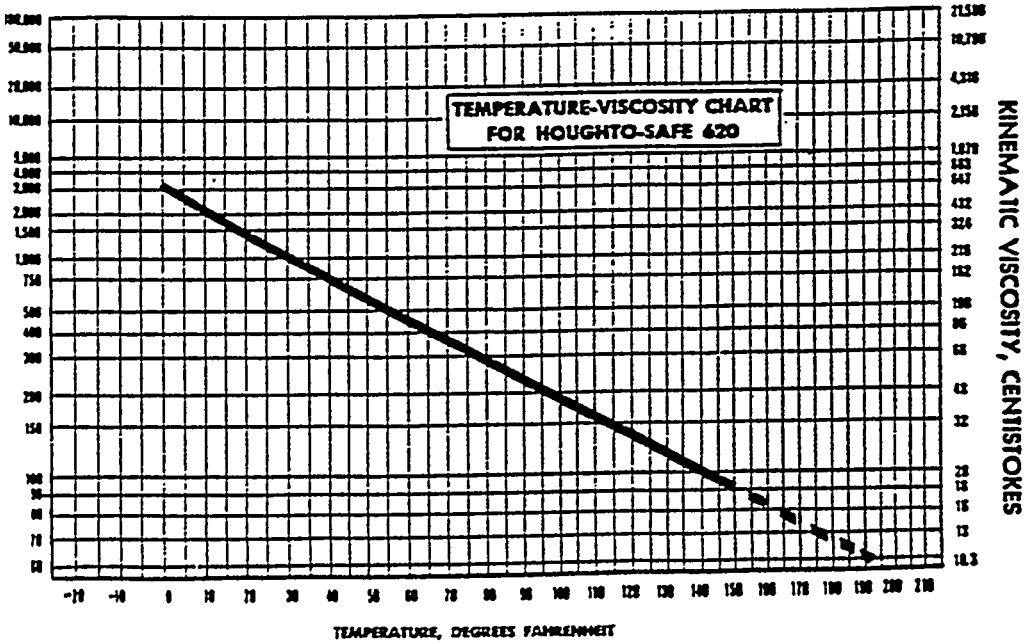
Applications

Houghto-Safe 620 is used in virtually every type of hydraulic equipment to help prevent the danger of fire:

- Die Casting Machines
- Foundry Machines
- Coke Oven Door Jacks and Pushers
- Furnance Chargers and Dischargers
- Centrifugal Pipe Casting Machines
- Forging Presses and Extrusion Presses
- Glass Feeder and Forming Machines
- Ingot Manipulators
- Fork Lift Trucks
- Clamping Fixutres on Automatic Welders
- Scarfing Machines
- Flying Shears
- Cranes, Hoists and Elevators
- Rod and Strip Coilers
- Rod Mill, Tube Mill, and Hot Strip Mill
- Crushers
- Planetary Mills
- Screw Down Controls
- Ladle Stoppers
- Pelletizing Mill Equipment
- Electric Furnace-Roof Lift, Tilt Mechanisms and Electrode Control
- Dolomite Machines
- Steel Strip Coil Unloaders
- Permanent Mold Machines
- Hydraulic Regulators and Controls
- Autopour Units
- Trimmers
- Furnace Door Controls
- Coupling Tighteners
- Roll Jacks and Roll Balance Controls
- Lift and Transfer Tables
- Plating Machines
- Grid Machines
- Straddle Trucks

Typical Physical Properties

Viscosity, SUS at	0°F.	3000
	70°F.	361
	100°F.	200
	130°F.	117
	150°F.	89
Viscosity Index		
Old Method (ASTM D567)		154
New Method (ASTM D2270)		201
Pour Point, °F.		–65
pH		8 to 10
Specific Gravity at 60°F.		1.074
Pounds per Gallon at 60°F.		8.95
Flash Point (ASTM)		Not applicable
Fire Point (ASTM)		Not applicable
Color		Red
Specific Heat at 60°F.		0.71
Thermal Conductivity B.T.U. per hr. per sq. ft. per °F. per ft.		0.26



Lubricity and Pump Life

The high lubricity of Houghto-Safe 620 is shown in these typical pump tests in widely used vane type and intravane type pumps.

Test conditions	*Vane type pump*	*Intravane type pump*
Test period, hours	500	500
Pressure, psi	1000	2000
Sump temp., °F.	115	115
Pump speed, rpm	1200	1200
Test results after 500 hours		
Wear on ring, grams	0.050	0.080
Wear on vanes gms	0.045	0.095
Wear on intravane set, grams		0.009
Fluid	No significant change	
Wear	Less than 0.1%	Less than 0.025%

This is very low wear! Many years of service can be expected under typical operating conditions.

Flex. Pris. Test Stand

Bertelkamp Automation, Inc.

Industrial Automation Specialists

P.O. Box 11643 / KNOXVILLE, TN 37939-1643
615-588-7691 / 800-251-9134
FAX: 615-588-9845

Quotation

Date: 22-Aug-95
Time: 09:35 AM

Quotation No.: ORNL0822
Revision:
Your inquiry:

Reference: Update of ORNL0816

To: **Brett Karaus**
ORNL
Oak Ridge, TN

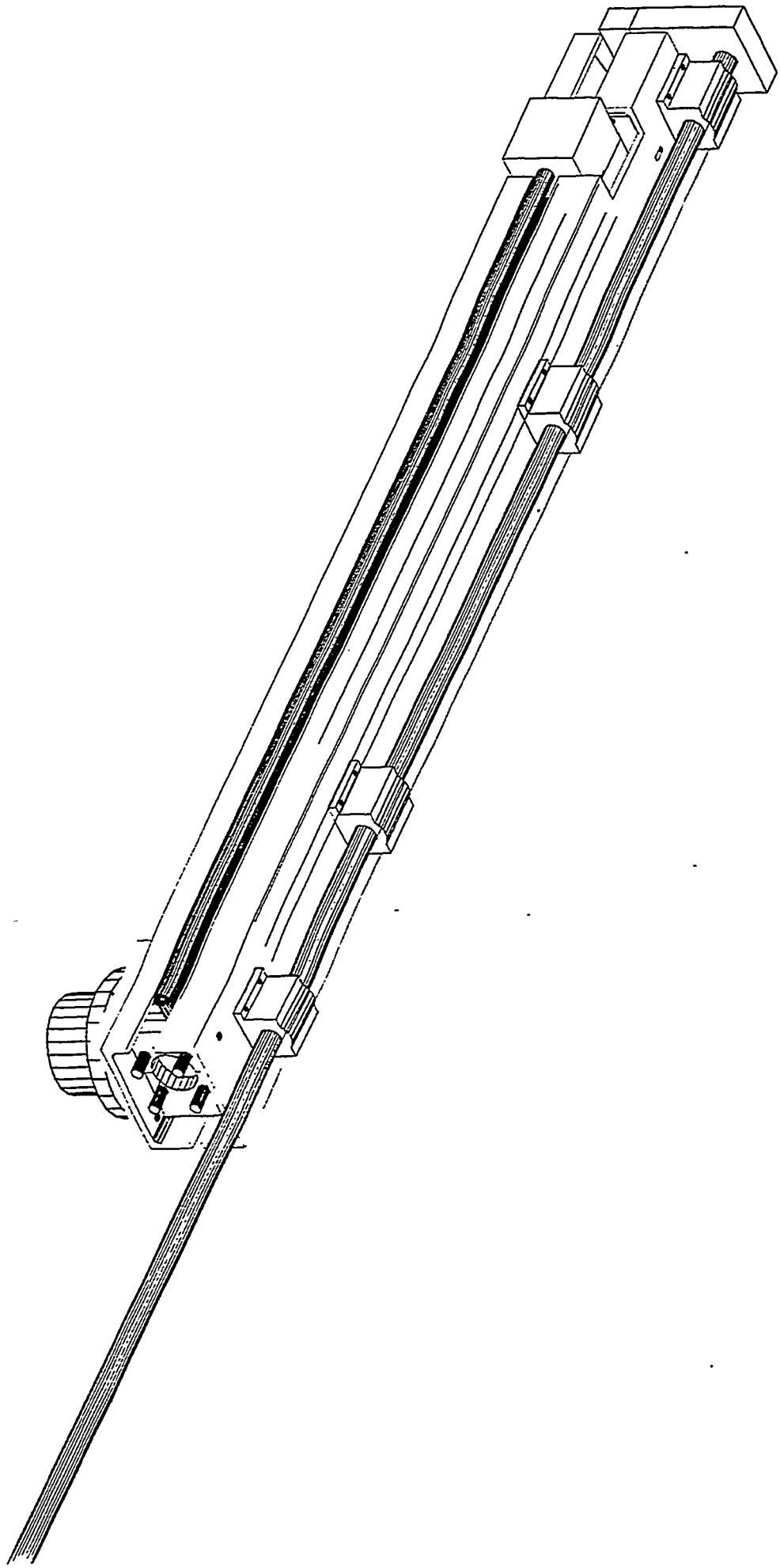
Attn:
cc: Gary Dalby

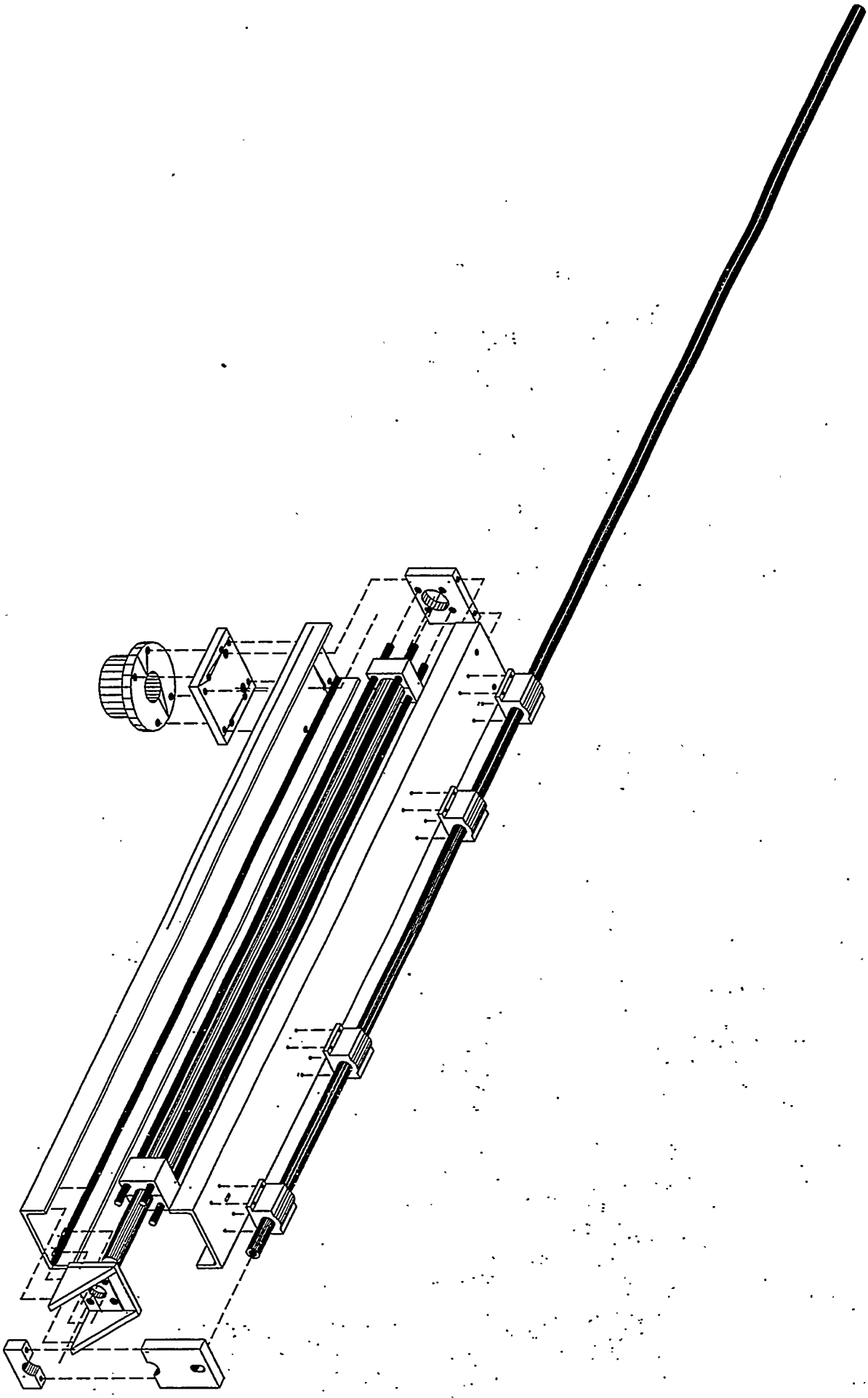
Item	Quantity	Part #	Description	Unit price	Ext price
1	144	3030	3" X 3" Heavy Wall T-Slotted Extrusion (4) at 36"	$1.39	$200.16
2	228	1530	1.5" X 3" Heavy Wall T-Slotted Extrusion (8) at 18" (4) at 21"	$0.88	$200.64
3	4	7030	Cut To Length 3030	$2.45	$9.80
4	12	7020	Cut To Length 1530 and 1530-Lite	$2.30	$27.60
5	4	2410	3030 Floor Mount Base Plate	$35.60	$142.40
6	8	4328	12 Hole 90 Degree Joining Plate	$12.25	$98.00
7	8	4325	12 Hole Tee Joining Plate	$12.25	$98.00
8	2	4355	8 Hole Tee Joining Plate	$9.05	$18.10
9	6	4366	6 Hole Joining Plate	$5.55	$33.30
10	4	4301	4 Hole Inside Corner Bracket	$4.05	$16.20
11	146	3355	5/16-18 X 5/8" Flanged BHSCS, Double Economy T-Nut	$1.39	$202.94
12	24	4375	6 Hole Inside Corner Bracket	$6.10	$146.40
13	96	3320	5/16-18 X 5/8" Flanged BHSCS & Economy T-Nut	$0.57	$54.72
14	30	1010	1" X 1" T-Slotted Extrusion (1) at 30"	$0.21	$6.30
15	1	7005	Cut To Length 1010	$1.85	$1.85
16	2	2380	1010 Floor Mount Base Plate	$21.10	$42.20
17	4	3356	1/4-20 X 1/2" Flanged BHSCS,Double Economy T-Nut	$0.90	$3.60

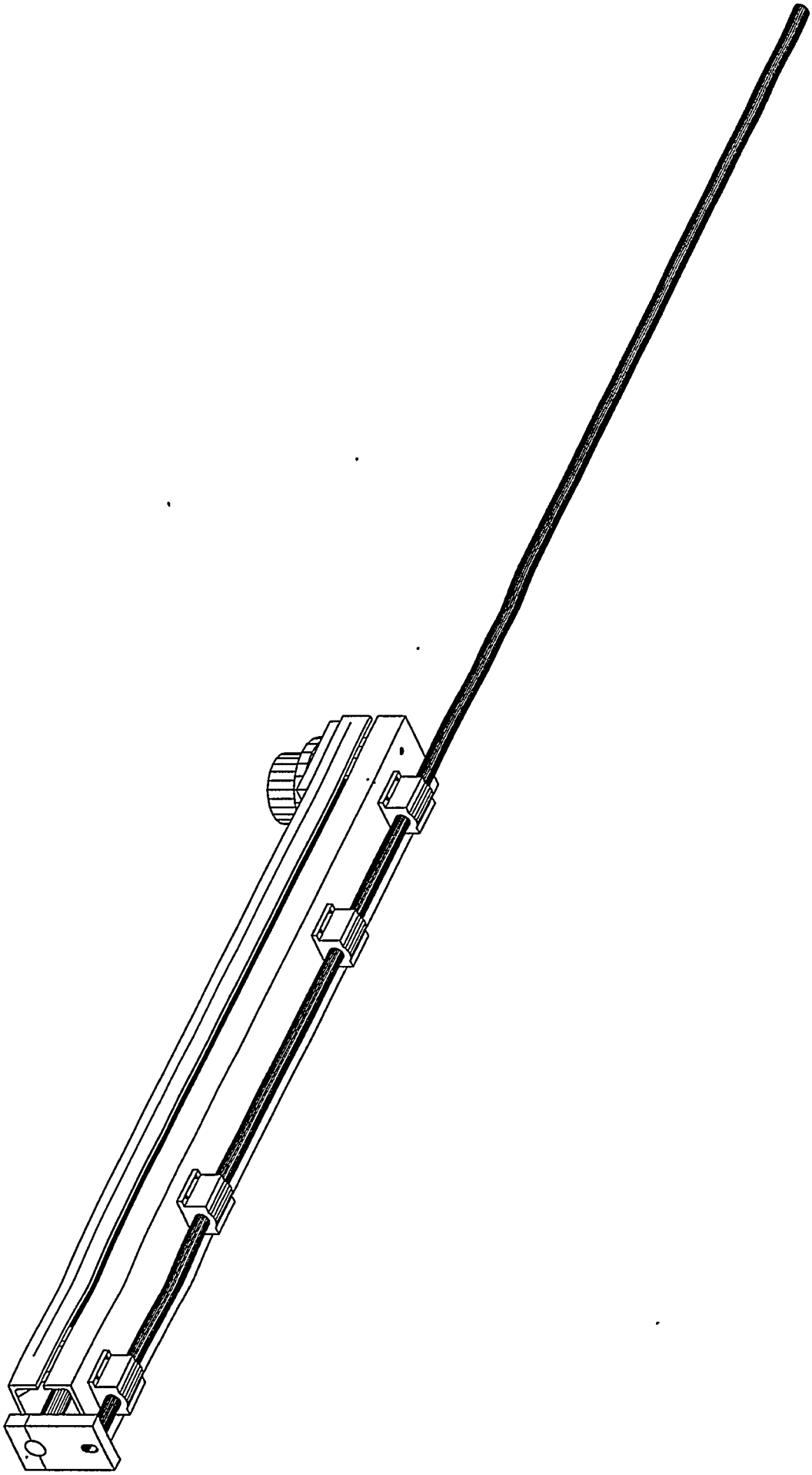
Total: $1,302.21

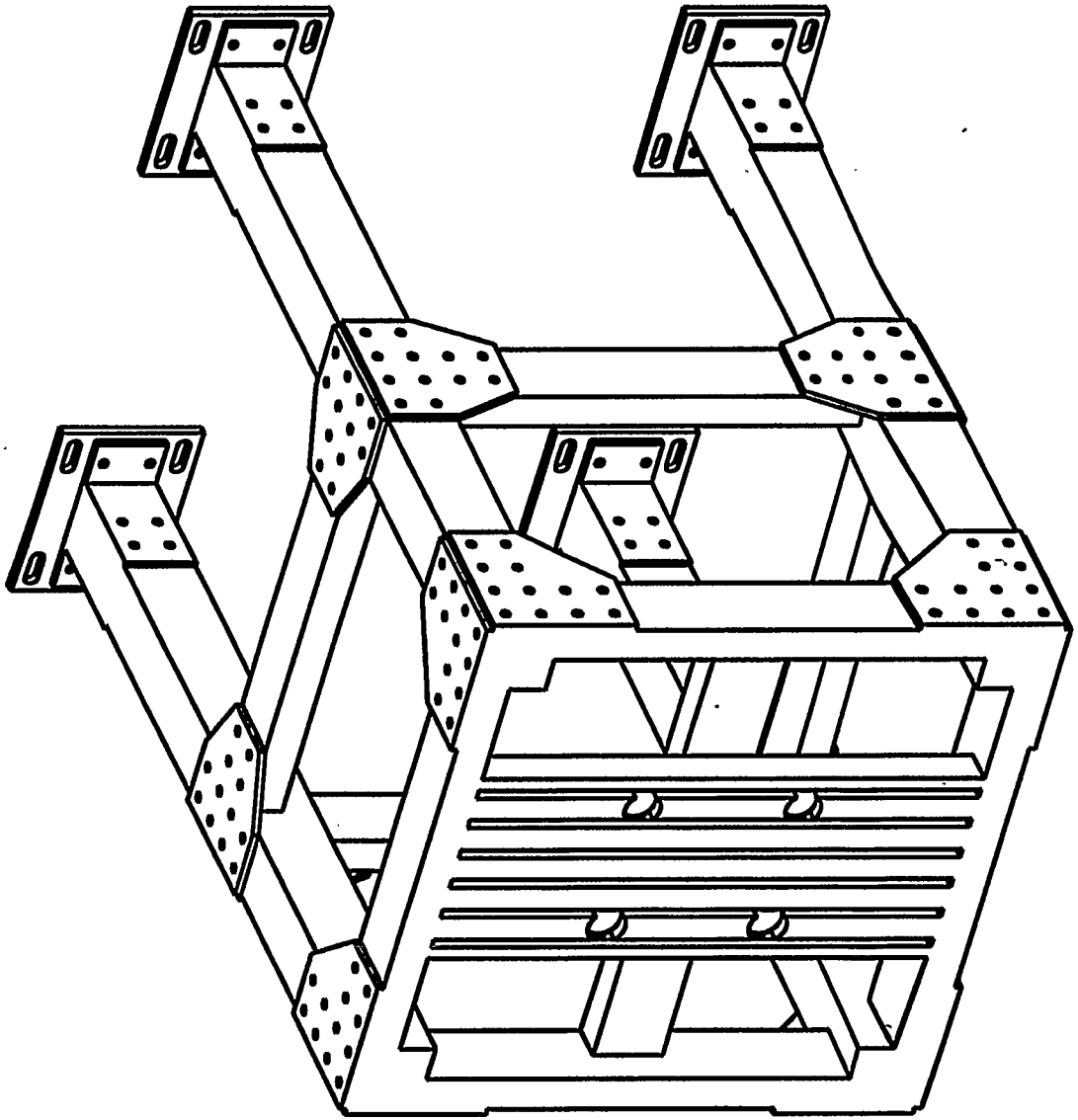
Quotations are firm for a period of 30 days unless otherwise specified.
F.O.B. Shipping point
Terms: 1% 10, net 30

BY: Mike Little









```
#include "filter.hh"

extern "C"
{
  void filmain(void);
}

void filmain(void)
{
  // Create filter parameter vectors
  RowVector b(5);
  RowVector a(4);

  b[0]=0.0048;
  b[1]=0.0193;
  b[2]=0.0289;
  b[3]=0.0193;
  b[4]=0.0048;

  a[0]=2.3695;
  a[1]=-2.314;
  a[2]=1.0547;
  a[3]=-0.1874;

  // Build Filter object Butter
  Filter Butter(b,a);

  // Run the filter on some data
  double sign=-1;
  for(int j=0;j<10;j++)
  {
    sign=-1*sign;
    for(int i=0;i<50;i++)
    {
      double u=10*sign;
      double y=Butter.Update(u);
      printf("\rInput = %lf and Output = %lf",u,y);
    }
  }
};
```

```
#include "filter.hh"

Filter::Filter() : theta(1), phi(1)
{
  FilterOrder=theta.GetVecSize()+phi.GetVecSize();
  NumOrder=theta.GetVecSize();
  DenOrder=phi.GetVecSize();
};

Filter::Filter(Filter &tmpFilter)
                : theta(tmpFilter.theta), phi(tmpFilter.GetFilterOrder())
{
        FilterOrder=tmpFilter.GetFilterOrder();
        NumOrder=tmpFilter.GetNumOrder();
        DenOrder=tmpFilter.GetDenOrder();
}

Filter::Filter(RowVector numtmp, RowVector dentmp)
                : theta(numtmp,dentmp), phi(numtmp.GetVecSize()+dentmp.GetVecSiz
e())
{
   FilterOrder=numtmp.GetVecSize()+dentmp.GetVecSize();
   NumOrder=numtmp.GetVecSize();
   DenOrder=dentmp.GetVecSize();
}

Filter::Filter(int numorder, int denorder)
                : theta(numorder+denorder), phi(numorder+denorder)
{
  FilterOrder=numorder+denorder;
  NumOrder=numorder;
  DenOrder=denorder;
}

Filter & Filter::operator = (Filter & filter1)
{
  if((filter1.NumOrder == NumOrder) & (filter1.DenOrder == DenOrder))
  {
    theta = filter1.theta;
    phi = filter1.phi;
  }
  else
  {
    printf("\nFilters are incompatable in Filter class!\n");
    exit(0);
  }
  return *this;
}

void Filter::print(void)
{
   printf("\nOrder of Num is %d ",NumOrder);
   printf("and Order of Den is %d",DenOrder+1);
   printf("\nParm Vector = ");
   theta.print();
}

// Actual filtering process; u is input stream
double Filter::Update(double u)
{
  double y;
  for(int i=NumOrder-1;i>0;i--)
    phi[i]=phi[i-1];
  phi[0]=u;

  y = theta * phi;

  for(i=(DenOrder+NumOrder-1);i>NumOrder;i--)
    phi[i]=phi[i-1];
  phi[NumOrder]=y;

  return(y);
}
```

```
// Filter Class
//
// General Form  Y(z)    B0 + B1 z^-1 + .. + Bnz z^-nz
//               ---- = -------------------------------
//               U(z)    1 - A1 z^-1 - ... - Anp z^-np
//
// Constructor is passed 2 vectors, num and den.
// num gives the vector parameters of the numerator coef. of the filter
// den gives the vector parameters of the denomenator coef.  Note the form
// above.  The filter
//
//                     2 + 3z^-1
//         G(z) = -------------------
//                1 + .25z^-1 + 3z^-2
//
//         would have num=[2 3] and den=[-.25 -3];
//
// Parameter Vector, theta, is a row vector, [a b ...]
// and State Vector, phi, is a column vector, [x y z ...]'
// output of filter is y[k]=theta*phi
//

#if !defined(__FILTER_HH)
#define __FILTER_HH 1

#include "matrix.hh"

class Filter
{
  public:
    Filter();
    ~Filter() { Delete(); };
    Filter(RowVector numparms, RowVector denparms);
    Filter(int numord, int denord);
    Filter(Filter &tmpFilter);
    Filter & operator = (Filter &Filter1);
    void print(void);
    double Update(double u);
    int GetFilterOrder(void){ return FilterOrder; }
    int GetNumOrder(void){ return NumOrder; }
    int GetDenOrder(void){ return DenOrder; }

  protected:
    int NumOrder;
    int DenOrder;
    int FilterOrder;
    RowVector theta;
    ColVector phi;
    void Delete (void)
    {
       NumOrder=0;
       DenOrder=0;
    };
};

#endif
```

```
cd "/u0/love/control"
ld < filtest.V
```

```
/*********************************************************************\
 *                    Oak Ridge National Laboratory
 *                 Robotics and Process Systems Division
 *                  U. S. Government internal use only
 *
 * FILE:          Bode.C
 * AUTHOR:        Lonnie J. Love
 * VERSION:       1.4
 * MODIFIED:      12 Jun 1996 09:00:18
 * DESCRIPTION:
 *
 * HISTORY:
 *   #      Date      By                    Comment
 * ------ ---------- ----- ----------------------------------------
 *  1.5    06/03/96   LJL  -Changed to FlexController Class
 *                          Broke program into controller and new Robot class
 *  1.4    08/09/95   PMR  -Removed HUMAN_MODELING_TEST stuff
 *  1.3    08/09/95   PMR  -Added load cell bias voltage monitor
 *                         -Passed analog input tests
 *  1.2    07/05/95   PMR  -Full humanAmp capabilities demonstrated in
 *                          both NORMAL and BODE modes
 *                         -HUMAN_MODELING_TEST not finished yet
 *  1.1    06/15/95   PMR  -Checked into SCCS.
 *  1.0    06/12/95   PMR  -Initial file creation.
\*********************************************************************/
//  Embedded sccs id string:
static char sccs_Bode_C[]= "@(#)Bode.C  1.4";

//  C includes
extern "C"
{
  #include  <vxWorks.h>   // For OK/ERROR
  #include  <logLib.h>    // For logMsg()
  #include  <stdio.h>     // For printf()
  #include  <sysLib.h>    // For sysClkRateSet()
  #include  <taskLib.h>   // For taskDelay(), taskLock(), taskUnlock()
  #include  <semLib.h>    // For semaphore calls
  #include  <math.h>      // For ldexp()
  #include  <fioLib.h>    // For fioRdString
  #include  <vme.h>
  #include  <time.h>      // For ctime

  //  Command-line-callable functions
  void  startBode_vf(void);
  void  stopBode_vf( void );
  int   BodeTask_if( void );
};

//  C++ includes
#include  "Robot.H"
#include  "traj.H"
#include  "Bode.H"
#include  <readDataFile.H>

//  Local macros
FILE      *fname;
int       FREQ_COMPLETE = TRUE;
int       FIRST = TRUE;

/*********************************************************************
  FUNCTION NAME:  startBode_vf ( )
  DESCRIPTION:    Kick off the task to create and maintain the Bode
                  controller object
*********************************************************************/
void  startBode_vf(void)
{
  if( startTaskId_i || terminationSem_a )
    logMsg( "Controller already running\n", 0,0,0,0,0,0 );
  else
    startTaskId_i = taskSpawn( "tHAStart",
                               9,
                               0, //VX_FP_TASK,
                               5000,
                               (FUNCPTR) BodeTask_if,
                               0, 0, 0, 0, 0, 0, 0, 0, 0, 0 );
}


/*********************************************************************
  FUNCTION NAME:  BodeTask_if ( )
  DESCRIPTION:    Task which creates and maintains the Bode controller
*********************************************************************/
int  BodeTask_if( void )
{
  time_t bintime;
  time(&bintime);

  logMsg( "BodeTask_if: Starting Bode controller\n", 0,0,0,0,0,0 );

  //  Create the semaphore used to terminate this task
  terminationSem_a = semBCreate(0,SEM_EMPTY);

  //  Create the instance of the Bode controller
  BodeController  controller_a;
  printf("Built Bode Controller\n");

  //  Wait for the termination sem to be given by the stopBode_vf func.
  semTake( terminationSem_a, WAIT_FOREVER );
  semDelete( terminationSem_a );
  terminationSem_a = (SEM_ID) NULL;
  startTaskId_i = 0;
  fclose(fname);
  printf( "\nBodeTask_if: Stopping Bode controller\n", 0,0,0,0,0,0 );

  return(OK);
}


/*********************************************************************
  FUNCTION NAME:  stopBode_vf ( )
  DESCRIPTION:    Kill off the controller
*********************************************************************/
void  stopBode_vf( void )
{
  if( startTaskId_i && terminationSem_a )
    semGive( terminationSem_a );
}


/*********************************************************************
  METHOD NAME:  BodeController ( )
```

```
  DESCRIPTION:  Constructor method
*********************************************************************/
BodeController::BodeController(void ) :
                    processClass(       "tHACntrl",
                                        100,  // Doesn't matter. Overridden.
                                        100,
                                        0,
                                        5000,
                                        MANUAL_startMeth,
                                        OFF_clockErr ), FlexArm(),
                                        initialJointPosition(2),
                                        Zero(2),
                                        desJtPosition(2),
                                        jointCommand(2),
                                        jointPosition(2),
                                        controlOut(2),
                                        error(2),
                                        Frequency(10)
{
  int  i,
       keepTrying_i = TRUE,
       retVal_i;
  double temp;

  if((fname=fopen("/u0/projects/Prismatic/dataFiles/BodeExp.dat","w+"))==NULL)
    printf("fopen failed\n");

  desJtPosition[ROTARY]=0;
  desJtPosition[PRISMATIC]=0;

  //  Create the compensator object
  compensatorR_ap = new Comp( "../../setupFiles/ServoRComp.dat", 0);
  compensatorP_ap = new Comp( "../../setupFiles/ServoPComp.dat", 0);

  if ( (!compensatorR_ap)||(!compensatorP_ap) )
    exit( ERROR );

  //  Read the sample frequency from the compensator for local use
  sampleFreq_i = compensatorR_ap->SampleFreq_if();

  //  Set the rate of the CPU's system clock (ticks per second)
  if( sysClkRateSet( sampleFreq_i ) == ERROR )
    exit( ERROR );

  //  Set the frequency of controller looping
  if( changeProcessFrequency_if( sampleFreq_i ) == ERROR )
    exit( ERROR );

  if( sampleFreq_i )
    del_time_d =  1.0/((double) sampleFreq_i);
  else
    del_time_d = 0.0;

  //  Initialize Robot object by passing base address
  FlexArm.Init();

  //  Initialize other class members
  tick_ul                 = 0;
  double elapsedRunTime_d      = 0.0;

  //  Read in data related to the DFT
  readDataFile setup_a( "/u0/projects/Prismatic/setupFiles/Bode.dat" );

  setup_a.readNextParmAsInt_if( &BodeJoint,1 );
  setup_a.readNextParmAsDouble_if( &A0, 1);
  setup_a.readNextParmAsInt_if( &NumOfFreq, 1);
  for(i=0;i<NumOfFreq;i++)
  {
    setup_a.readNextParmAsDouble_if( &temp, 1);
    Frequency[i]=temp;
  }

  //  Start the controller
  if( start_if() == ERROR )
    exit( ERROR );

  //  All is done.  Print a message for the operator
  taskLock();
  logMsg ( "\n\n\n", 0,0,0,0,0,0 );
  logMsg ( "|-----------------------------------------------------|\n",
           0,0,0,0,0,0 );
  logMsg ( "|||  Construction complete.  Controller is running  |||\n",
           0,0,0,0,0,0 );
  logMsg ( "|-----------------------------------------------------|\n\n\n",
           0,0,0,0,0,0 );
  taskUnlock();
}


/*********************************************************************
  METHOD NAME:  doit_vf ( )
  DESCRIPTION:  Method to be called at a fixed period to perform the
                synchronous control.
*********************************************************************/
void  BodeController::doit_vf( void )
{
  int           i,
                j;
  double        Ts,
                decay;

  Ts = del_time_d;
  double sampleFreq_d = 1/Ts;

  //  Update loop related variables
  elapsedRunTime_d = del_time_d * tick_ul;  // tick_ul is in processClass

  //  Read the analog inputs
  jointPosition = FlexArm.GetJointPosition();

  //-------------------------------------------------------------------
  //  Start off by moving to home position
  //-------------------------------------------------------------------
  if( elapsedRunTime_d < timeHome_Cd )
  {
    // Hold at initial position for 1st 10 sec.
    if ( elapsedRunTime_d < 0.5*timeHome_Cd)
    {
      error = initialJointPosition-jointPosition;
    }
    else
    {
```

```
      decay=1.0-exp(-(elapsedRunTime_d-0.5*timeHome_Cd)/7.5);
      error=(initialJointPosition+
            decay*(desJtPosition-initialJointPosition) )
           -jointPosition;
    }
  }
  //-----------------------------------------------------------------------
  //  Now start doing Bode stuff
  //-----------------------------------------------------------------------
  else
  {
    if(FREQ_COMPLETE == TRUE)
    {
      FREQ_COMPLETE = FALSE;

      if(FIRST == TRUE)
      {
        FIRST=FALSE;
        FreqIndex=0;
      }
      else
      {
        // Compute the magnitude and phase before reseting constants
        Magnitude = (2/NumDataPoints)*sqrt(B0*B0+C0*C0)/A0;
        Phase = 180.0*atan2(C0,B0)/3.14159265;

        // Write results to a file
        fprintf(fname,"%6.4lf %3.1lf %6.4lf %6.4lf %7.4lf\n",
          A0, Frequency[FreqIndex], Magnitude, Phase, elapsedRunTime_d);

        printf("Freq=%3.1lf Mag=%6.4lf Phase=%6.4lf Time=%7.4lf\n",
          Frequency[FreqIndex], Magnitude, Phase, elapsedRunTime_d);

        // Increment to the next frequency and cycle back to first freq. if
        // the index exceeds the number of frequencies investigated
        FreqIndex++;
        if(FreqIndex >= NumOfFreq)
          FreqIndex=0;
      }
      // Compute the number of data points necessary to create ten cycles
      // at the selected frequency and convert to rad/sec.
      NumDataPoints=int(10*(sampleFreq_d)/Frequency[FreqIndex]);
      omega=2*3.14159265*Frequency[FreqIndex]/sampleFreq_d;

      // Reset all necessary counters
      DataPoint=0;
      tick_ul=0;
      elapsedRunTime_d=0;
      B0 = 0;
      C0 = 0;
    }
    error[0]=0;
    error[1]=0;

    // Compute the command to the desired joint
    error[BodeJoint] = A0*sin(omega*(elapsedRunTime_d-timeHome_Cd))-
      jointPosition[BodeJoint];

    // Compute the DFT at the specified frequency
    DFTCount=2*3.14159265*Frequency[FreqIndex]/sampleFreq_d*(double(DataPoint));
    DataPoint++;

      B0 += jointPosition[BodeJoint]*cos(omega*(DFTCount));
      C0 += jointPosition[BodeJoint]*sin(omega*(DFTCount));

      if(DataPoint >= NumDataPoints)
        FREQ_COMPLETE=TRUE;
    }

    if((compensatorR_ap->Next(error[ROTARY],&controlOut[ROTARY])!=1)||
       (compensatorP_ap->Next(-1.0*error[PRISMATIC],&controlOut[PRISMATIC])!= 1))
        jointCommand=Zero;
    else
        jointCommand=controlOut;

    //  Write the analog outputs
    FlexArm.SetCommand(jointCommand);
  }
```

```
/*********************************************************************\
 *                      Oak Ridge National Laboratory
 *                   Robotics and Process Systems Division
 *                    U. S. Government internal use only
 *
 * FILE:          Bode.H
 * AUTHOR:        Lonnie J Love
 * VERSION:       1.0
 * MODIFIED:      19 Jan 1996 09:01:34
 * DESCRIPTION:
 *
 * HISTORY:
 *   #      Date       By                    Comment
 * ------ ---------- ----- ----------------------------------------
 *  1.3    08/09/95   PMR  -Added load cell bias voltage monitor
 *                         -Passed analog input tests
 *  1.2    07/05/95   PMR  -Full humanAmp capabilities demonstrated in
 *                          both NORMAL and BODE modes
 *                         -HUMAN_MODELING_TEST not finished yet
 *  1.1    06/15/95   PMR  -Checked into SCCS.
 *  1.0    06/12/95   PMR  -Initial file creation.
\*********************************************************************/
//  Embedded sccs id string:
static char sccs_Bode_H[]= "@(#)Bode.H  1.1";

#ifndef  BODE_H
#define  BODE_H

  #define MAX_NUMBER_OF_POINTS  10;  // maximum number of points for a path

  int     startTaskId_i     = 0;
  SEM_ID  terminationSem_a = (SEM_ID) NULL;

  //  C++ includes
  #include  <IP-padc.H>
  #include  <IP-dac.H>
  #include  <IP-dig24.H>
  #include  <processClass.H>
  #include  "compensator.H"
  #include  "Robot.H"
  #include  "matrix.H"

  class BodeController : processClass
  {
    public:
      BodeController( void );
      ~BodeController( void )
      {
        if( compensatorP_ap)  delete compensatorP_ap;
        if( compensatorR_ap)  delete compensatorR_ap;
        if( startTaskId_i )  startTaskId_i = 0;
        if( terminationSem_a )
        {
          semDelete( terminationSem_a );
          terminationSem_a = (SEM_ID) NULL;
        }
        printf("\nBodeController Destroyed..\n");
      }

    private:
      Robot                 FlexArm;
      double                del_time_d,
                            omega,
                            Magnitude,
                            Phase,
                            A0,
                            B0,
                            C0,
                            DFTCount,
                            sampleFreq_d,
                            elapsedRunTime_d;

      const double          timeHome_Cd          = 40.0;  // Seconds
      const double          biasCalcTime_Cd      = 5.0;   // Seconds

      int                   sampleFreq_i,
                            FreqIndex,
                            NumOfFreq,
                            BodeJoint,
                            DataPoint,
                            NumDataPoints;


      // Path and Control Vectors
      ColVector             initialJointPosition;
      ColVector             Zero;
      ColVector             desJtPosition;
      ColVector             jointCommand;
      ColVector             jointPosition;
      ColVector             error;
      ColVector             controlOut;
      ColVector             Frequency;

      //. Pointer to a compensator object
      Comp *                compensatorR_ap; // jfjansen added 8-29-95
      Comp *                compensatorP_ap; // jfjansen added 8-29-95

      // Methods
      void  doit_vf( void );
  };
#endif
```

```
/*********************************************************************\
 *                    Oak Ridge National Laboratory
 *                 Robotics and Process Systems Division
 *                  U. S. Government internal use only
 *
 * FILE:          Bode.C
 * AUTHOR:        Lonnie J. Love
 * VERSION:       1.4
 * MODIFIED:      12 Jun 1996 09:00:18
 * DESCRIPTION:
 *
 * HISTORY:
 *   #      Date      By                    Comment
 * ------ ---------- ----- ----------------------------------------
 *  1.5    06/03/96   LJL  -Changed to FlexController Class
 *                          Broke program into controller and new Robot class
 *  1.4    08/09/95   PMR  -Removed HUMAN_MODELING_TEST stuff
 *  1.3    08/09/95   PMR  -Added load cell bias voltage monitor
 *                         -Passed analog input tests
 *  1.2    07/05/95   PMR  -Full humanAmp capabilities demonstrated in
 *                          both NORMAL and BODE modes
 *                         -HUMAN_MODELING_TEST not finished yet
 *  1.1    06/15/95   PMR  -Checked into SCCS.
 *  1.0    06/12/95   PMR  -Initial file creation.
\*********************************************************************/
//  Embedded sccs id string:
static char sccs_Bode_C[]= "@(#)Bode.C  1.4";

//  C includes
extern "C"
{
  #include  <vxWorks.h>   // For OK/ERROR
  #include  <logLib.h>    // For logMsg()
  #include  <stdio.h>     // For printf()
  #include  <sysLib.h>    // For sysClkRateSet()
  #include  <taskLib.h>   // For taskDelay(), taskLock(), taskUnlock()
  #include  <semLib.h>    // For semaphore calls
  #include  <math.h>      // For ldexp()
  #include  <fioLib.h>    // For fioRdString
  #include  <vme.h>
  #include  <time.h>      // For ctime

  //  Command-line-callable functions
  void  startBode_vf(void);
  void  stopBode_vf( void );
  int   BodeTask_if( void );
};

//  C++ includes
#include  "Robot.H"
#include  "traj.H"
#include  "Bode.H"
#include  <readDataFile.H>

//  Local macros
FILE      *fname;
int       FREQ_COMPLETE = TRUE;
int       FIRST = TRUE;

/*********************************************************************
  FUNCTION NAME:  startBode_vf ( )
  DESCRIPTION:    Kick off the task to create and maintain the Bode
                  controller object
*********************************************************************/
void  startBode_vf(void)
{
  if( startTaskId_i || terminationSem_a )
    logMsg( "Controller already running\n", 0,0,0,0,0,0 );
  else
    startTaskId_i = taskSpawn( "tHAStart",
                               9,
                               0, //VX_FP_TASK,
                               5000,
                               (FUNCPTR) BodeTask_if,
                               0, 0, 0, 0, 0, 0, 0, 0, 0, 0 );
}


/*********************************************************************
  FUNCTION NAME:  BodeTask_if ( )
  DESCRIPTION:    Task which creates and maintains the Bode controller
*********************************************************************/
int  BodeTask_if( void )
{
  time_t bintime;
  time(&bintime);

  logMsg( "BodeTask_if: Starting Bode controller\n", 0,0,0,0,0,0 );

  //  Create the semaphore used to terminate this task
  terminationSem_a = semBCreate(0,SEM_EMPTY);

  //  Create the instance of the Bode controller
  BodeController  controller_a;
  printf("Built Bode Controller\n");

  //  Wait for the termination sem to be given by the stopBode_vf func.
  semTake( terminationSem_a, WAIT_FOREVER );
  semDelete( terminationSem_a );
  terminationSem_a = (SEM_ID) NULL;
  startTaskId_i = 0;
  fclose(fname);
  printf( "\nBodeTask_if: Stopping Bode controller\n", 0,0,0,0,0,0 );

  return(OK);
}


/*********************************************************************
  FUNCTION NAME:  stopBode_vf ( )
  DESCRIPTION:    Kill off the controller
*********************************************************************/
void  stopBode_vf( void )
{
  if( startTaskId_i && terminationSem_a )
    semGive( terminationSem_a );
}


/*********************************************************************
  METHOD NAME:  BodeController ( )
```

```
  DESCRIPTION:  Constructor method
*********************************************************************/
BodeController::BodeController(void ) :
                    processClass(       "tHACntrl",
                                        100;  // Doesn't matter. Overridden.
                                        100,
                                        0,
                                        5000,
                                        MANUAL_startMeth,
                                        OFF_clockErr ), FlexArm(),
                                        initialJointPosition(2),
                                        Zero(2),
                                        desJtPosition(2),
                                        jointCommand(2),
                                        jointPosition(2),
                                        controlOut(2),
                                        error(2),
                                        Frequency(10)
{
  int  i,
       keepTrying_i = TRUE,
       retVal_i;
  double temp;

  if((fname=fopen("/u0/projects/Prismatic/dataFiles/BodeExp.dat","w+"))==NULL)
    printf("fopen failed\n");

  desJtPosition[ROTARY]=0;
  desJtPosition[PRISMATIC]=0;

  //  Create the compensator object
  compensatorR_ap = new Comp( "../../setupFiles/BodeServoComp.dat", 0);
  compensatorP_ap = new Comp( "../../setupFiles/BodeServoComp.dat", 0);

  if ( (!compensatorR_ap)||(!compensatorP_ap) )
    exit( ERROR );

  //  Read the sample frequency from the compensator for local use
  sampleFreq_i = compensatorR_ap->SampleFreq_if();

  //  Set the rate of the CPU's system clock (ticks per second)
  if( sysClkRateSet( sampleFreq_i ) == ERROR )
    exit( ERROR );

  //  Set the frequency of controller looping
  if( changeProcessFrequency_if( sampleFreq_i ) == ERROR )
    exit( ERROR );

  if( sampleFreq_i )
    del_time_d =  1.0/((double) sampleFreq_i);
  else
    del_time_d = 0.0;

  //  Initialize Robot object by passing base address
  FlexArm.Init();

  //  Initialize other class members
  tick_ul                  = 0;
  double elapsedRunTime_d      = 0.0;
  B0 = 0.0;
  C0 = 0.0;

  //  Read in data related to the DFT
  readDataFile setup_a( "/u0/projects/Prismatic/setupFiles/Bode.dat" );

  setup_a.readNextParmAsInt_if( &BodeJoint,1 );
  setup_a.readNextParmAsDouble_if( &A0, 1);
  setup_a.readNextParmAsInt_if( &NumOfFreq, 1);
  for(i=0;i<NumOfFreq;i++)
  {
    setup_a.readNextParmAsDouble_if( &temp, 1);
    Frequency[i]=temp;
  }

  //  Start the controller
  if( start_if() == ERROR )
    exit( ERROR );

  //  All is done.  Print a message for the operator
  taskLock();
  logMsg ( "\n\n\n", 0,0,0,0,0,0 );
  logMsg ( "|-------------------------------------------------------|\n",
           0,0,0,0,0,0 );
  logMsg ( "|||  Construction complete.  Controller is running  |||\n",
           0,0,0,0,0,0 );
  logMsg ( "|-------------------------------------------------------|\n\n\n",
           0,0,0,0,0,0 );
  taskUnlock();
}


/*********************************************************************
  METHOD NAME:  doit_vf ( )
  DESCRIPTION:  Method to be called at a fixed period to perform the
                synchronous control.
*********************************************************************/
void  BodeController::doit_vf( void )
{
  int           i,
                j;
  double        Ts,
                decay;

  Ts = del_time_d;
  double sampleFreq_d = 1/Ts;

  //  Update loop related variables
  elapsedRunTime_d = del_time_d * tick_ul;  // tick_ul is in processClass

  //  Read the analog inputs
  jointPosition = FlexArm.GetJointPosition();

  if(tick_ul == 1)
    initialJointPosition=jointPosition;

  //-------------------------------------------------------------------
  //  Start off by moving to home position
  //-------------------------------------------------------------------
  if( elapsedRunTime_d < timeHome_Cd )
  {
    if ( elapsedRunTime_d < 0.25*timeHome_Cd)
      error = initialJointPosition-jointPosition;
```

```
    else
      decay=1.0-exp(-(elapsedRunTime_d-0.25*timeHome_Cd)/5.0);
      error=(initialJointPosition+
            decay*(desJtPosition-initialJointPosition) )
           -jointPosition;
  }
  else
  {
    if(FREQ_COMPLETE == TRUE)
    {
      FREQ_COMPLETE = FALSE;
      if(FIRST == TRUE)
      {
        FIRST=FALSE;
        FreqIndex=0;
      }
      else
      {
        // Compute the magnitude and phase before reseting constants
        Magnitude = (2/double(NumDataPoints))*sqrt(B0*B0+C0*C0)/A0;
        Phase = 180.0*atan2(C0,B0)/3.14159265;

        // Write results to a file
        fprintf(fname,"%6.4lf %3.1lf %6.4lf %6.4lf %7.4lf\n",
          A0, Frequency[FreqIndex], Magnitude, Phase, elapsedRunTime_d);

        printf("\nFreq=%3.1lf Mag=%6.4lf Phase=%6.4lf Time=%4.2lf\n",
          Frequency[FreqIndex], Magnitude, Phase,elapsedRunTime_d);

        // Increment to the next frequency and cycle back to first freq. if
        // the index exceeds the number of frequencies investigated
        FreqIndex++;
        if(FreqIndex >= NumOfFreq)
          FreqIndex=0;
      }
      // Compute the number of data points necessary to create ten cycles
      // at the selected frequency and convert to rad/sec.
      NumDataPoints=int(20*(sampleFreq_d)/Frequency[FreqIndex]);

      // Reset all necessary counters
      DataPoint=0;
      tick_ul=0;
      elapsedRunTime_d=0;
      B0 = 0;
      C0 = 0;

      // Reset the initial joint position
      initialJointPosition = FlexArm.GetJointPosition();
    }

    // First set command to servo about desJointPosition
    error = desJtPosition - jointPosition;

    DataPoint++;
    DFTCount=2*3.14159265*Frequency[FreqIndex]/sampleFreq_d*(double(DataPoint));

    // Reset the target joint to have a sinusoidal command
    error[BodeJoint] = desJtPosition[BodeJoint] + A0*sin(DFTCount) -
                       jointPosition[BodeJoint];

    B0 += jointPosition[BodeJoint]*cos(DFTCount);
    C0 += jointPosition[BodeJoint]*sin(DFTCount);

    if(DataPoint >= NumDataPoints)
      FREQ_COMPLETE=TRUE;
  }
  if( !(tick_ul %50) )
    printf("error=%6.4lf %6.4lf B0=%6.4lf C0=%6.4lf joint=%ld time=%4.1lf\r",
      error[ROTARY],error[PRISMATIC],B0,C0,BodeJoint,elapsedRunTime_d);

  if((compensatorR_ap->Next(error[ROTARY],&controlOut[ROTARY])!=1)||
     (compensatorP_ap->Next(-1.0*error[PRISMATIC],&controlOut[PRISMATIC])!= 1))
      jointCommand=Zero;
  else
      jointCommand=controlOut;

  jointCommand[ROTARY] = 0.02*error[ROTARY];
  jointCommand[PRISMATIC]= -0.02*error[PRISMATIC];

  //  Write the analog outputs
  FlexArm.SetCommand(jointCommand);
}
```

```
/*********************************************************************\
 *                      Oak Ridge National Laboratory
 *                  Robotics and Process Systems Division
 *                   U. S. Government internal use only
 *
 * FILE:          Bode.H
 * AUTHOR:        Lonnie J Love
 * VERSION:       1.0
 * MODIFIED:      19 Jan 1996 09:01:34
 * DESCRIPTION:
 *
 * HISTORY:
 *   #      Date      By                    Comment
 * ------ ---------- ----- ----------------------------------------
 *  1.3    08/09/95   PMR  -Added load cell bias voltage monitor
 *                         -Passed analog input tests
 *  1.2    07/05/95   PMR  -Full humanAmp capabilities demonstrated in
 *                          both NORMAL and BODE modes
 *                         -HUMAN_MODELING_TEST not finished yet
 *  1.1    06/15/95   PMR  -Checked into SCCS.
 *  1.0    06/12/95   PMR  -Initial file creation.
\*********************************************************************/
//  Embedded sccs id string:
static char sccs_Bode_H[]= "@(#)Bode.H  1.1";

#ifndef  BODE_H
#define  BODE_H

  #define MAX_NUMBER_OF_POINTS  10;  // maximum number of points for a path

  int     startTaskId_i    = 0;
  SEM_ID  terminationSem_a = (SEM_ID) NULL;

  //  C++ includes
  #include  <IP-padc.H>
  #include  <IP-dac.H>
  #include  <IP-dig24.H>
  #include  <processClass.H>
  #include  "compensator.H"
  #include  "Robot.H"
  #include  "matrix.H"

  class BodeController : processClass
  {
    public:
      BodeController( void );
      ~BodeController( void )
      {
        if( compensatorP_ap)   delete compensatorP_ap;
        if( compensatorR_ap)   delete compensatorR_ap;
        if( startTaskId_i )   startTaskId_i = 0;
        if( terminationSem_a )
        {
          semDelete( terminationSem_a );
          terminationSem_a = (SEM_ID) NULL;
        }
        printf("\nBodeController Destroyed..\n");
      }

    private:
      Robot                 FlexArm;
      double                del_time_d,
                            omega,
                            Magnitude,
                            Phase,
                            A0,
                            B0,
                            C0,
                            DFTCount,
                            sampleFreq_d,
                            elapsedRunTime_d;

      const double          timeHome_Cd          = 40.0;  // Seconds
      const double          biasCalcTime_Cd      = 5.0;   // Seconds

      int                   sampleFreq_i,
                            FreqIndex,
                            NumOfFreq,
                            BodeJoint,
                            DataPoint,
                            NumDataPoints;


      // Path and Control Vectors
      ColVector             initialJointPosition;
      ColVector             Zero;
      ColVector             desJtPosition;
      ColVector             jointCommand;
      ColVector             jointPosition;
      ColVector             error;
      ColVector             controlOut;
      ColVector             Frequency;

      //  Pointer to a compensator object
      Comp *                compensatorR_ap; // jfjansen added 8-29-95
      Comp *                compensatorP_ap; // jfjansen added 8-29-95

      // Methods
      void  doit_vf( void );
  };
#endif
```

```
/*************************************************************************\
 *                        Oak Ridge National Laboratory
 *                     Robotics and Process Systems Division
 *                      U. S. Government internal use only
 *
 * FILE:  Robot.H
 * AUTHOR: Lonnie Love
 * VERSION: %I%
 * Modified: %G% %U%
 *
 * Description:
 *
 *
 * HISTORY:
 *   #      Date       By                    Comment
 * ------ ---------- ----- ----------------------------------------
 *  1.0    05/10/96   love  Initial file creation.
\*************************************************************************/
#ifndef  ROBOT_H
#define  ROBOT_H

enum actuators
{
  ROTARY,
  PRISMATIC,
  NUM_JOINTS
};

/* Embedded sccs id string: */
static char sccs_Robot_H[]= "%W%";

// C++ includes
#include "IP-padc.H"
#include "IP-dac.H"
#include "IP-dig24.H"
#include "processClass.H"
#include "matrix.H"
#include "vector.H"

class Robot
{
  public:
    Robot(void);
    ~Robot()
    {
      // Set the DAC outputs to zero...
      for (int i=0; i<(int)NUM_ANALOG_OUTPUTS_a; i++)
        dac_1_p->write_analogval( anOutChanMap_aa[i], analogDAC_BIAS_Cd[i] );
      if( dig24_1_p     )  delete dig24_1_p;
      if( padc_1_p  )  delete padc_1_p;
      if( dac_1_p   )  delete dac_1_p;
      printf( "Robot: destroyed Robot object\n");
    };
    void Init(void);
    ColVector & GetJointPosition(void);
    ColVector & GetTipDeflection(void);
    ColVector & GetTrueTipPosition(void);
    ColVector & CalcTipPosition(const ColVector & jtPosition);
    ColVector & CalcDesJointPosition(const ColVector & tipPosition);
    void SetCommand(const ColVector & cmd);

  private:
    ColVector JointPosition;
    ColVector DesJointPosition;
    ColVector TipPosition;
    ColVector TrueTipPosition;
    ColVector TipDeflection;
    ColVector Command;
    const int DOF = 2;
    const double pi = 3.1415926;
    const double PI = 3.1415926;
    enum analogInputs
    {
      HUMAN_FORCE_CHAN_a,     // first 3 channels used by John's manAmp
      LOAD_FORCE_CHAN_a,
      LOAD_POSITION_CHAN_a,      // Next 2 channels are joint position
      FLEXIBLE_ROT_a,
      FLEXIBLE_PRIS_a,
      NUM_ANALOG_INPUTS_a
    };
    enum analogOutputs
    {
        LOAD_COMMAND_FORCE_CHAN_a,
        FLEXIBLE_ROT_DRIVE,
        FLEXIBLE_PRIS_DRIVE,
        NUM_ANALOG_OUTPUTS_a
    };
    enum digitalInputs
    {
      DEADMAN_SWITCH_CHAN_a,
      NUM_DIGITAL_INPUTS_a
    };
    enum digitalOutputs
    {
      ENABLE_SERVO_VALVE_CHAN_a,
      DEBUG_TIMER_CHAN_a,
      NUM_DIGITAL_OUTPUTS_a
    };

    // Pointers to I/O objects
    PADC_class *          padc_1_p;
    DAC_class *           dac_1_p;
    DIG24_class *         dig24_1_p;

    // Arrays for mapping A/D I/O channels
    PADC_class::CHNL      anInChanMap_aa[NUM_ANALOG_INPUTS_a];
    DAC_class::chan_num   anOutChanMap_aa[NUM_ANALOG_OUTPUTS_a];
    DIG24_class::IO_LINE  digInChanMap_aa[NUM_DIGITAL_INPUTS_a];
    DIG24_class::IO_LINE  digOutChanMap_aa[NUM_DIGITAL_OUTPUTS_a];

    double                anInVals_da[NUM_ANALOG_INPUTS_a];
    double                anOutVals_da[NUM_ANALOG_OUTPUTS_a];
    double                analogInScaleFactor_da[NUM_ANALOG_INPUTS_a];
    double                analogDAC_BIAS_Cd[NUM_ANALOG_INPUTS_a];
    int                   digInVals_ia[NUM_DIGITAL_INPUTS_a];
    int                   digOutVals_ia[NUM_DIGITAL_OUTPUTS_a];
};

#endif /*  ROBOT_H */
```

```
/*********************************************************************\
 *                      Oak Ridge National Laboratory
 *                   Robotics and Process Systems Division
 *                    U. S. Government internal use only
 *
 * FILE:  Robot.C
 * AUTHOR: Lonnie Love
 * VERSION: %I%
 * Modified: %G% %U%
 *
 * Description:
 *
 *
 * HISTORY:
 *   #      Date       By                    Comment
 * ------ ---------- ----- ----------------------------------------
 *  1.0    05/10/96   love  Initial file creation.
\*********************************************************************/
/* Embedded sccs id string: */
static char sccs_Robot_C[]= "%W%";

#include "Robot.H"
#define VIPC610_VMEBUS_BASE_ADDR (0x6000)

extern "C"
{
  #include  <vxWorks.h>   // For OK/ERROR
  #include  <logLib.h>    // For logMsg()
  #include  <sysLib.h>    // For sysClkRateSet()
  #include  <taskLib.h>   // For taskDelay(), taskLock(), taskUnlock()
  #include  <semLib.h>    // For semaphore calls
  #include  <math.h>      // For ldexp()
  #include  <fioLib.h>    // For fioRdString
  #include  <vme.h>
}

/*********************************************************************
        METHOD NAME:     Robot ()
        DESCRIPTION:     Constructor method
*********************************************************************/
Robot::Robot(void) : JointPosition(2), DesJointPosition(2), TipPosition(2),
  TrueTipPosition(2), TipDeflection(2)
{
  logMsg( "Robot: Constructed Robot object\n",0,0,0,0,0,0);
}


/*********************************************************************
        METHOD NAME:     Init ()
        DESCRIPTION:     Initializes all drivers
*********************************************************************/
void Robot::Init(void)
{
  int retVal_i;
  char * local_base_addr_cp;

  retVal_i = sysBusToLocalAdrs( VME_AM_SUP_SHORT_IO,
                                (char *) VIPC610_VMEBUS_BASE_ADDR,
                                &local_base_addr_cp );
  if( retVal_i == ERROR )
    exit( ERROR );

  //  Create and setup the IP-DAC object
  dac_1_p = new  DAC_class( (int *) (local_base_addr_cp + IP_A_IO_OFFSET),
                            (unsigned char)  0x41 );
  if( !dac_1_p )
    exit( ERROR );

  //  Map the DAC channels to analogOutputs for conveinent access
  anOutChanMap_aa[ (int) LOAD_COMMAND_FORCE_CHAN_a ] = DAC_class::chan_1;
  anOutChanMap_aa[ (int) FLEXIBLE_ROT_DRIVE ] = DAC_class::chan_2;
  anOutChanMap_aa[ (int) FLEXIBLE_PRIS_DRIVE ] = DAC_class::chan_3;

  //  Initialize DAC bias values
  analogDAC_BIAS_Cd[FLEXIBLE_ROT_DRIVE]  = 0.000;
  analogDAC_BIAS_Cd[FLEXIBLE_PRIS_DRIVE] =-0.316;

  //  Setup the DAC channels.
  for(int i=0; i<NUM_ANALOG_OUTPUTS_a; i++ )
      dac_1_p->set_chnl( anOutChanMap_aa[i],
                         DAC_class::minus10_to_plus10 );

  //  Create and setup IP-PADC object
  padc_1_p = new PADC_class( (int *) (local_base_addr_cp+IP_B_IO_OFFSET) );
  if( !padc_1_p )
    exit( ERROR );

  //  Map the PADC channels to the analogInputs array for conveinent access
  anInChanMap_aa [ (int) HUMAN_FORCE_CHAN_a ]   = PADC_class::CH1;
  anInChanMap_aa [ (int) LOAD_FORCE_CHAN_a ]    = PADC_class::CH2;
  anInChanMap_aa [ (int) LOAD_POSITION_CHAN_a ]     = PADC_class::CH3;
  anInChanMap_aa [ (int) FLEXIBLE_ROT_a ] = PADC_class::CH4;
  anInChanMap_aa [ (int) FLEXIBLE_PRIS_a ] = PADC_class::CH5;

  //  Setup the PADC channels.  No need to call calibrate_padc_vf,
  //  since it is called by it's constructor already
  for( i=0; i<(int)NUM_ANALOG_INPUTS_a; i++ )
  {
    padc_1_p->set_chnl_vf( anInChanMap_aa[i],    //
                           PADC_class::X1,       // +-5 Volts
                           PADC_class::DIF );    // Differential
    switch( i )
    {
      case  HUMAN_FORCE_CHAN_a:
            analogInScaleFactor_da[i] = 10.0;    // Lbs/Volt
            break;

      case  LOAD_FORCE_CHAN_a:
            analogInScaleFactor_da[i] = 100.0;   // Lbs/Volt
            break;

      case  LOAD_POSITION_CHAN_a:
          analogInScaleFactor_da[i] = 2.1;       // inches/volt
          break;

      case  FLEXIBLE_PRIS_a:
          analogInScaleFactor_da[i] = 6.0;
          break;

      case  FLEXIBLE_ROT_a:
          analogInScaleFactor_da[i] = 45.0; //jfj
```

```
        break;

      default:
          analogInScaleFactor_da[i] =1.0; //jfj
            break;
    }
  }

  //  Create and setup IP-DIG24 object
  dig24_1_p = new  DIG24_class( (int *)(local_base_addr_cp + IP_C_IO_OFFSET),
                                (unsigned char)  0xb1 );
  if( !dig24_1_p )
    exit( ERROR );

  //  Digital Inputs
  //  Map the DIG24 channels to digInChanMap_aa for conveinent access
  digInChanMap_aa[ (int) DEADMAN_SWITCH_CHAN_a ] = DIG24_class::LINE_1;

  //  Digital Outputs
  //  Map the DIG24 channels to digOutChanMap_aa for conveinent access
  digOutChanMap_aa[ (int) ENABLE_SERVO_VALVE_CHAN_a] = DIG24_class::LINE_13;
  digOutChanMap_aa[ (int) DEBUG_TIMER_CHAN_a       ] = DIG24_class::LINE_24;

  logMsg( "Robot: Initializeded Robot object\n",0,0,0,0,0,0);
}


/*************************************************************************
        METHOD NAME:    GetJointPosition
        DESCRIPTION:    Reads A/D for joint data and puts in JointPosition
                        vector as well as returns the results in a vector
*************************************************************************/
ColVector & Robot::GetJointPosition(void)
{
  for(int i=0; i<(int)NUM_ANALOG_INPUTS_a; i++)
  {
    padc_1_p->read_analog_val_vf( anInVals_da+i, anInChanMap_aa[i] );
    anInVals_da[i] *= analogInScaleFactor_da[i];
  }
  // Map to JointPosition vector
  JointPosition[PRISMATIC]=anInVals_da[FLEXIBLE_PRIS_a];
  JointPosition[ROTARY]=anInVals_da[FLEXIBLE_ROT_a];
  return( JointPosition );
}


/*************************************************************************
        METHOD NAME:    GetTipDeflection
        DESCRIPTION:    Reads the data from the lateral effect photo diodes
                        and returns the deformation of the tip.
*************************************************************************/
ColVector & Robot::GetTipDeflection(void)
{
  return(TipDeflection);
}


/*************************************************************************
        METHOD NAME:    GetTrueTipPosition
        DESCRIPTION:    Uses joint and deformation information to estiamte
                        the true tip position of the robot
*************************************************************************/
ColVector & Robot::GetTrueTipPosition(void)
{
  return(TrueTipPosition);
}


/*************************************************************************
        METHOD NAME:    CalcTipPosition
        DESCRIPTION:    Does forward kinematics on joint data to estimate
                        rigid body tip position.  The home position is at
                        the axis of rotation.  There is an inner radius of
                        twelve inches plus the zero position of the cylinder
                        is at the center of the cylinder.  In addition, the
                        polarity of the prismatic joint is reversed.
*************************************************************************/
ColVector & Robot::CalcTipPosition(const ColVector &jtPosition)
{
  double r = 36 - jtPosition[PRISMATIC];
  double theta =  -(PI/180)*jtPosition[ROTARY];

  TipPosition[0] = r*sin(theta);
  TipPosition[1] = r*cos(theta);

  return(TipPosition);
}


/*************************************************************************
        METHOD NAME:    CalcJointPosition
        DESCRIPTION:    Does the inverse kinematics on the tip position
                        passed to the function.
*************************************************************************/
ColVector & Robot::CalcDesJointPosition(const ColVector & tipPosition)
{
  double x = tipPosition[0];
  double y = tipPosition[1];

  // In inches
  DesJointPosition[PRISMATIC] = 36.0 - sqrt(x*x+y*y);
  // In degrees
  DesJointPosition[ROTARY] = -(180/PI)*atan2(x, y);

  return(DesJointPosition);
}


/*************************************************************************
        METHOD NAME:    Command
        DESCRIPTION:    Sends the command out the D/A converter
*************************************************************************/
void Robot::SetCommand(const ColVector & cmd)
{
  int j;
  // 3 D/A wired, only using #2 and #3 (manAmp uses #1)
  for(int i=0; i<(int)NUM_ANALOG_OUTPUTS_a-1; i++)
  {
    j=i+1;  // offset to skip 1st D/A (used by manAmp)
    dac_1_p->write_analogval(anOutChanMap_aa[j],cmd[i]+analogDAC_BIAS_Cd[j]);
  }
}
```

```
/*********************************************************************\
 *                    Oak Ridge National Laboratory
 *                 Robotics and Process Systems Division
 *                  U. S. Government internal use only
 *
 * FILE:          FlexController.H
 * AUTHOR:        Lonnie J Love
 * VERSION:       1.0
 * MODIFIED:      19 Jan 1996 09:01:34
 * DESCRIPTION:
 *
 * HISTORY:
 *   #      Date      By                    Comment
 * ------ ---------- ----- ----------------------------------------
 *  1.3    08/09/95   PMR  -Added load cell bias voltage monitor
 *                         -Passed analog input tests
 *  1.2    07/05/95   PMR  -Full humanAmp capabilities demonstrated in
 *                          both NORMAL and BODE modes
 *                         -HUMAN_MODELING_TEST not finished yet
 *  1.1    06/15/95   PMR  -Checked into SCCS.
 *  1.0    06/12/95   PMR  -Initial file creation.
\*********************************************************************/
//  Embedded sccs id string:
static char sccs_FlexController_H[]= "@(#)FlexController.H      1.1";

#ifndef  FLEXCONTROLLER_H
#define  FLEXCONTROLLER_H

  #define MAX_NUMBER_OF_POINTS  10;  // maximum number of points for a path
  int     startTaskId_i    = 0;
  SEM_ID  terminationSem_a = (SEM_ID) NULL;

  //  C++ includes
  #include  <IP-padc.H>
  #include  <IP-dac.H>
  #include  <IP-dig24.H>
  #include  <processClass.H>
  #include  "compensator.H"
  #include  "Robot.H"
  #include  "matrix.H"

  class FlexController : processClass
  {
    public:
      FlexController( void );
      ~FlexController( void )
      {
        if( compensator_ap )  delete compensator_ap;
        if( compensatorP_ap)  delete compensatorP_ap;
        if( compensatorR_ap)  delete compensatorR_ap;
        if( compensatorFlexRot_ap) delete compensatorFlexRot_ap;
        if( compensatorFlexPris_ap) delete compensatorFlexPris_ap;
        if( startTaskId_i )  startTaskId_i = 0;
        if( terminationSem_a )
        {
          semDelete( terminationSem_a );
          terminationSem_a = (SEM_ID) NULL;
        }
        printf("\nFlexController Destroyed..\n");
      }
      int getPath(void);

    private:
      Robot                 FlexArm;
      double                del_time_d,
                            pathRunTime_d,
                            elapsedRunTime_d;
      int                   goneHomeMsgFlag_i,
                            deadmanSwitchTimer_i,
                            biasCalcMsgFlag_i,
                            operatingMsgFlag_i,
                            sampleFreq_i;

      const double          deadmanDelay_Cd      = 1.0;   // Seconds
      const double          timeHome_Cd          = 40.0;  // Seconds
      const double          runTime_Cd           = 20.0;  // Seconds
      const double          biasCalcTime_Cd      = 5.0;   // Seconds
      const double          pathPause            = 2.0;  // Seconds
      const double          startupTime_Cd       = 5.0;   // Seconds
      double                stopTime_Cd;

      // Path and Control Vectors
      ColVector             initialJointPosition;
      ColVector             Zero;
      ColVector             desJtPosition;
      ColVector             desTipPosition;
      ColVector             jointCommand;
      ColVector             jointPosition;
      ColVector             tipPosition;
      ColVector             initialTipPosition;
      ColVector             finalTipPosition;
      ColVector             Xdes;
      ColVector             error;
      ColVector             controlOut;
      ColVector             XFiltdes;
      Matrix                Path;
      int                   pathNumberOfPoints,
                            NumPoints,
                            LineIndex,
                            PointIndex;

      //  Pointer to a compensator object
      Comp *                compensator_ap;
      Comp *                compensatorF_ap; // jfjansen added 8-22-95
      Comp *                compensatorV_ap; // jfjansen added 8-29-95
      Comp *                compensatorR_ap; // jfjansen added 8-29-95
      Comp *                compensatorP_ap; // jfjansen added 8-29-95
      Comp *                compensatorFlexRot_ap;  // ljl added 4-17-96
      Comp *                compensatorFlexPris_ap;  // ljl added 4-17-96

      // Methods
      void  setControllerInitialConditions_vf( void );
      void  doit_vf( void );
  };
#endif
```

```
/*********************************************************************\
 *                    Oak Ridge National Laboratory
 *                 Robotics and Process Systems Division
 *                  U. S. Government internal use only
 *
 * FILE:         FlexController.C
 * AUTHOR:       Lonnie J. Love
 * VERSION:      1.4
 * MODIFIED:     19 Jan 1996 09:00:18
 * DESCRIPTION:
 *
 * HISTORY:
 *   #      Date       By                    Comment
 * ------ ---------- ----- ----------------------------------------
 *  1.5    06/03/96   LJL  -Changed to FlexController Class
 *                          Broke program into controller and new Robot class
 *  1.4    08/09/95   PMR  -Removed HUMAN_MODELING_TEST stuff
 *  1.3    08/09/95   PMR  -Added load cell bias voltage monitor
 *                         -Passed analog input tests
 *  1.2    07/05/95   PMR  -Full humanAmp capabilities demonstrated in
 *                          both NORMAL and BODE modes
 *                         -HUMAN_MODELING_TEST not finished yet
 *  1.1    06/15/95   PMR  -Checked into SCCS.
 *  1.0    06/12/95   PMR  -Initial file creation.
\*********************************************************************/
//  Embedded sccs id string:
static char sccs_FlexController_C[]= "@(#)FlexController.C      1.4";


//  C includes
extern "C"
{
  #include  <vxWorks.h>   // For OK/ERROR
  #include  <logLib.h>    // For logMsg()
  #include  <stdio.h>     // For printf()
  #include  <sysLib.h>    // For sysClkRateSet()
  #include  <taskLib.h>   // For taskDelay(), taskLock(), taskUnlock()
  #include  <semLib.h>    // For semaphore calls
  #include  <math.h>      // For ldexp()
  #include  <fioLib.h>    // For fioRdString
  #include  <vme.h>
  #include  <time.h>      // For ctime

  //  Command-line-callable functions
  void  startFlex_vf(void);
  void  stopFlex_vf( void );
  int   FlexTask_if( void );
  int   getPath_if( void );
};

//  C++ includes
#include  "FlexController.H"
#include  "Robot.H"
#include  "traj.H"
#include  <readDataFile.H>

//  Local macros
int       PATH_SET = TRUE;
int       PATH_COMPLETE_MESSAGE = FALSE;
int       PATH_COMPLETE = FALSE;
int       LINE_COMPLETE = TRUE;
FILE      *fname;


/*********************************************************************
  FUNCTION NAME:  startFlex_vf ( )
  DESCRIPTION:    Kick off the task to create and maintain the human amp
                  controller object
*********************************************************************/
void  startFlex_vf(void)
{
  if( startTaskId_i || terminationSem_a )
    logMsg( "Controller already running\n", 0,0,0,0,0,0 );
  else
    startTaskId_i = taskSpawn( "tHAStart",
                               9,
                               0, //VX_FP_TASK,
                               5000,
                               (FUNCPTR) FlexTask_if,
                               0, 0, 0, 0, 0, 0, 0, 0, 0, 0 );
  PATH_COMPLETE = FALSE;
  LINE_COMPLETE = TRUE;
}


/*********************************************************************
  FUNCTION NAME:  getPath_if ( )
  DESCRIPTION:    Task which is entered by user and reads in Path.dat
*********************************************************************/
int getPath_if( void )
{
  // Set path flag. This will cause FlexController::getPath() method to
  // execute in FlexController::doit()
  PATH_SET = TRUE;
  return( OK );
}


/*********************************************************************
  FUNCTION NAME:  FlexTask_if ( )
  DESCRIPTION:    Task which creates and maintains the Flexure controller
*********************************************************************/
int  FlexTask_if( void )
{
  time_t bintime;
  time(&bintime);

  logMsg( "FlexTask_if: Starting flexure controller\n", 0,0,0,0,0,0 );

  //  Create the semaphore used to terminate this task
  terminationSem_a = semBCreate(0,SEM_EMPTY);

  //  Create the instance of the human amplifier controller
  FlexController  controller_a;
  printf("Built FlexController\n");

  //  Wait for the termination sem to be given by the stopFlex_vf func.
  //  When it is given, delete the sem and clear the task ID buffer, then
  //  return (which terminates the controller)
  semTake( terminationSem_a, WAIT_FOREVER );
  semDelete( terminationSem_a );
```

```
  terminationSem_a = (SEM_ID) NULL;
  startTaskId_i = 0;
  printf( "\nFlexTask_if: Stopping Flexure controller\n", 0,0,0,0,0,0 );

  return(OK);
}


/*************************************************************************
  FUNCTION NAME:  stopFlex_vf ( )
  DESCRIPTION:    Kill off the controller
*************************************************************************/
void  stopFlex_vf( void )
{
  if( startTaskId_i && terminationSem_a )
    semGive( terminationSem_a );
}


/*************************************************************************
  METHOD NAME:  FlexController ( )
  DESCRIPTION:  Constructor method
*************************************************************************/
FlexController::FlexController(void ) :
                    processClass(       "tHACntrl",
                                        100,  // Doesn't matter. Overridden.
                                        100,
                                        0,
                                        5000,
                                        MANUAL_startMeth,
                                        OFF_clockErr ), FlexArm(),
                                        initialJointPosition(2),
                                        Zero(2),
                                        desJtPosition(2),
                                        desTipPosition(2),
                                        jointCommand(2),
                                        jointPosition(2),
                                        tipPosition(2),
                                        initialTipPosition(2),
                                        finalTipPosition(2),
                                        Xdes(2),
                                        XFiltdes(2),
                                        controlOut(2),
                                        error(2),
                                        Path(10,4)
{
  int  i,
       keepTrying_i = TRUE,
       retVal_i;

  //  Create the compensator object
  compensatorR_ap=new Comp("/u0/projects/Prismatic/setupFiles/ServoRComp.dat", 0
);
  compensatorP_ap=new Comp("/u0/projects/Prismatic/setupFiles/ServoPComp.dat", 0
);
  compensator_ap = new Comp( "/u0/projects/Prismatic/setupFiles/ServoComp.dat",
0);
  compensatorFlexRot_ap=new Comp("/u0/projects/Prismatic/setupFiles/FlexComp.dat
", 0);
  compensatorFlexPris_ap=new Comp("/u0/projects/Prismatic/setupFiles/FlexComp.da
t", 0);

  if ((!compensator_ap)||(!compensatorFlexRot_ap)||(!compensatorFlexPris_ap)||
      (!compensatorR_ap)|| (!compensatorP_ap) )
    exit( ERROR );

  //  Read the sample frequency from the compensator for local use
  sampleFreq_i = compensator_ap->SampleFreq_if();

  //  Set the rate of the CPU's system clock (ticks per second)
  if( sysClkRateSet( sampleFreq_i ) == ERROR )
    exit( ERROR );

  //  Set the frequency of controller looping
  if( changeProcessFrequency_if( sampleFreq_i ) == ERROR )
    exit( ERROR );

  if( sampleFreq_i )
    del_time_d =  1.0/((double) sampleFreq_i);
  else
    del_time_d = 0.0;

  //  Initialize Robot object by passing base address
  FlexArm.Init();

  //  Initialize other class members
  tick_ul                  = 0;
  deadmanSwitchTimer_i     = (int) deadmanDelay_Cd * sampleFreq_i;
  double elapsedRunTime_d       = 0.0;

  //  Load path variables if appropriate (added by ljl - 4/16/96)
  if(PATH_SET == TRUE)
  {
    taskLock();
    getPath();

    // print out results
    printf("\n\nPath Parameters...\n");
    printf("------------------\n");
    for(i=0;i<pathNumberOfPoints;i++)
      printf("Xf=[%4.2lf %4.2lf] Vmax=%4.2lf Amax=%4.2lf\n",
        Path[i][0],Path[i][1],Path[i][2],Path[i][3]);
    taskUnlock();
    PATH_SET = FALSE;
  }

  //  Start the controller
  if( start_if() == ERROR )
    exit( ERROR );

  //  All is done.  Print a message for the operator
  taskLock();
  logMsg ( "\n\n\n", 0,0,0,0,0,0 );
  logMsg ( "|-------------------------------------------------------|\n",
           0,0,0,0,0,0 );
  logMsg ( "|||  Construction complete.  Controller is running  |||\n",
           0,0,0,0,0,0 );
  logMsg ( "|-------------------------------------------------------|\n\n\n",
           0,0,0,0,0,0 );
  taskUnlock();
}
```

```
/*************************************************************************
  METHOD NAME:  getPath ( )
  DESCRIPTION:  Method to read Path.dat file and store results in Path matrix
*************************************************************************/
int FlexController::getPath(void)
{
  double temp;

  // open up new file for saving data (closed at end of path)
  if((fname=fopen("/u0/projects/Prismatic/dataFiles/PrismExp.dat","w+"))==NULL)
    printf("fopen failed\n");

  readDataFile setup_a( "/u0/projects/Prismatic/setupFiles/Path.dat");
  setup_a.readNextParmAsInt_if( &pathNumberOfPoints, 0);
  printf("\nNumber of Points %ld\n",pathNumberOfPoints);

  for(int i=0;i<pathNumberOfPoints;i++)
  {
    setup_a.readNextParmAsDouble_if( &temp, 0);
    Path[i][0]=temp;  // final X
    setup_a.readNextParmAsDouble_if( &temp, 0);
    Path[i][1]=temp;  // final Y
    setup_a.readNextParmAsDouble_if( &temp, 0);
    Path[i][2]=temp;  // max velocity
    setup_a.readNextParmAsDouble_if( &temp, 0);
    Path[i][3]=temp;  // max acceleration
  }
  setup_a.closeSetupFile_if();

  // Initialize the starting point at the current arm position
  initialJointPosition=FlexArm.GetJointPosition();

  LineIndex=0;
  PointIndex=0;
  tick_ul=0;
  PATH_COMPLETE = FALSE;
  LINE_COMPLETE = TRUE;

  return(OK);
}


/*************************************************************************
  METHOD NAME:  doit_vf ( )
  DESCRIPTION:  Method to be called at a fixed period to perform the
                synchronous control.
*************************************************************************/
void  FlexController::doit_vf( void )
{
  int           i,
                j,
                retVal_i;
  double        desVel,
                maxAcc,
                Ts,
                decay;
  ColVector     XFiltdes(2);

  static MinTimeTrajectory Line(2);  // use minimum time trajectories on path
  Ts = del_time_d;

  //  Check to see if a new path is requested
  if( (PATH_SET==TRUE) & (PATH_COMPLETE == TRUE)  )
  {
    taskLock();
    printf("\nset Path in doit()\n");
    getPath();
    taskUnlock();
    PATH_SET = FALSE;
  }

  //  Update loop related variables
  elapsedRunTime_d = del_time_d * tick_ul;  //  tick_ul is in processClass

  //  Read the analog inputs
  jointPosition = FlexArm.GetJointPosition();
  tipPosition = FlexArm.CalcTipPosition(jointPosition);

  //  Calculate the error signal
  if( elapsedRunTime_d < timeHome_Cd )
  {
    // Hold at initial position for 1st 10 sec.
    if ( elapsedRunTime_d < 0.5*timeHome_Cd)
    {
      error = initialJointPosition-jointPosition;
    }
    else
    {
      decay=1.0-exp(-(elapsedRunTime_d-0.5*timeHome_Cd)/5);
      error=(initialJointPosition+
            decay*(desJtPosition-initialJointPosition) )
           -jointPosition;
    }
    if( !(tick_ul %10) )
    {
      printf("desJt=%4.2lf %4.2lf error=%4.2lf  %4.2lf decay=%4.2lf\r",
         desJtPosition[ROTARY],desJtPosition[PRISMATIC],
         error[ROTARY],error[PRISMATIC],decay);
    }
  }
  else if( elapsedRunTime_d < (biasCalcTime_Cd  +  timeHome_Cd) )
  {
    desTipPosition[0]=Path[0][0];
    desTipPosition[1]=Path[0][1];
    desJtPosition=FlexArm.CalcDesJointPosition(desTipPosition);
    error=desJtPosition-jointPosition;
    if( !(tick_ul %10) )
    {
      printf("desJt=%4.2lf %4.2lf error=%4.2lf  %4.2lf\r",
         desJtPosition[ROTARY],desJtPosition[PRISMATIC],
         error[ROTARY],error[PRISMATIC]);
    }
  }
  else
  {
    // Path planner
    if(PATH_COMPLETE == FALSE)
    {
      if(LINE_COMPLETE == TRUE)
      {
        initialTipPosition[0]=Path[LineIndex][0];
```

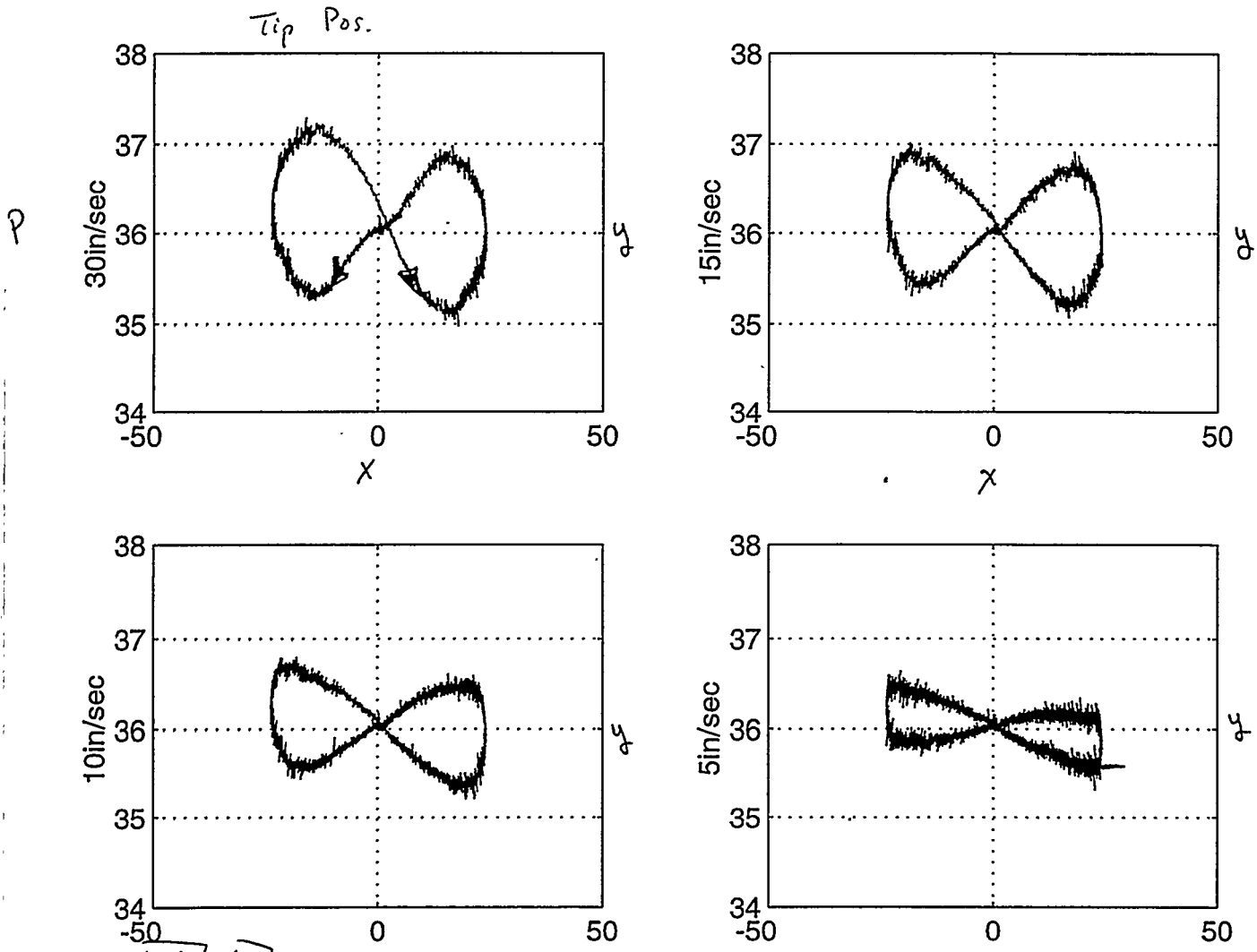
```
      finalTipPosition[0]=Path[LineIndex+1][0];
      initialTipPosition[1]=Path[LineIndex][1];
      finalTipPosition[1]=Path[LineIndex+1][1];
      desVel=Path[LineIndex+1][2];
      maxAcc=Path[LineIndex+1][3];
      Line.Reset(initialTipPosition,finalTipPosition,desVel,maxAcc,Ts);
      NumPoints=Line.GetT();
      PointIndex=0;
      LINE_COMPLETE=FALSE;
      printf("\n\nLineIndex=%d NumPoints=%d\n",LineIndex,NumPoints);

    }
    desTipPosition=Line.Update(PointIndex);
    PointIndex++;

    if(PointIndex==NumPoints-1)
    {
      LineIndex++;
      LINE_COMPLETE=TRUE;
    }

    if(LineIndex>=pathNumberOfPoints-1)
    {
      stopTime_Cd = elapsedRunTime_d;
      printf("\n");
      PATH_COMPLETE=TRUE;
      PATH_COMPLETE_MESSAGE=TRUE;
    }
    // Inverse Kinematics to get desired joint values
    Xdes=FlexArm.CalcDesJointPosition(desTipPosition);

    // Run the filter here to flush out any transients.
    compensatorFlexRot_ap->Next( Xdes[ROTARY],  &XFiltdes[ROTARY] );
    compensatorFlexPris_ap->Next( Xdes[PRISMATIC], &XFiltdes[PRISMATIC] );

    // Not running filter yet
    error=Xdes-jointPosition;

    if( !(tick_ul %10) )
    {
      printf("Tip=%4.2lf %4.2lf Joint=%4.2lf %4.2lf\r",
          tipPosition[0],tipPosition[1],jointPosition[ROTARY],
          jointPosition[PRISMATIC]);
    }
    fprintf(fname,
          "%6.4lf %6.4lf %6.4lf %6.4lf %6.4lf %6.4lf %6.4lf %6.4lf %5.3lf\n",
          jointPosition[0], jointPosition[1],
          Xdes[0], Xdes[1],
          tipPosition[0],tipPosition[1],
          desTipPosition[0],desTipPosition[1],elapsedRunTime_d);
  }
  else
  {
    if(elapsedRunTime_d < stopTime_Cd+pathPause)
    {
      error=Xdes-jointPosition;
      fprintf(fname,
          "%6.4lf %6.4lf %6.4lf %6.4lf %6.4lf %6.4lf %6.4lf %6.4lf %5.3lf\n",
          jointPosition[ROTARY], jointPosition[PRISMATIC],
          Xdes[ROTARY], Xdes[PRISMATIC],
          tipPosition[0],tipPosition[1],
          desTipPosition[0],desTipPosition[1],elapsedRunTime_d);
      }
      else
      {
        if( PATH_COMPLETE_MESSAGE == TRUE )
        {
          printf("\nPath is over...\n");
          PATH_COMPLETE_MESSAGE = FALSE;
          fclose(fname);
        }
        error=Xdes-jointPosition;
      }
    }
  }

  //  Apply the control filters
  if ( elapsedRunTime_d < (biasCalcTime_Cd + timeHome_Cd) )
  {
    if((compensatorR_ap->Next(error[ROTARY],&controlOut[ROTARY])!=1)||
    (compensatorP_ap->Next(-1.0*error[PRISMATIC],&controlOut[PRISMATIC])!= 1))
    {
      printf("error in jointCommand...\r");
      jointCommand=Zero;
    }
    else
      jointCommand=controlOut;
  }
  else
  {
   if((compensatorP_ap->Next(-1.0*error[PRISMATIC],&controlOut[PRISMATIC])!=1)||
      (compensatorR_ap->Next( error[ROTARY], &controlOut[ROTARY]) != 1 ) )
    {
      printf("error in jointCommand...\r");
      jointCommand=Zero;
    }
   else
      jointCommand=controlOut;
  }

  //  Write the analog outputs
  FlexArm.SetCommand(jointCommand);
}


/*************************************************************************
  METHOD NAME:  setControllerInitialConditions_vf ( )
  DESCRIPTION:  Method to reset the controller's initial conditions.
*************************************************************************/
void  FlexController::setControllerInitialConditions_vf( void )
{
  return;
}
```

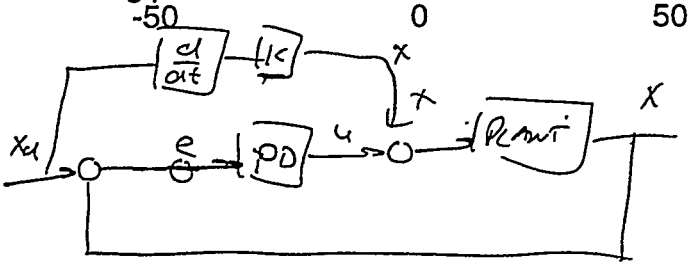
Dynamic Accuracy Test

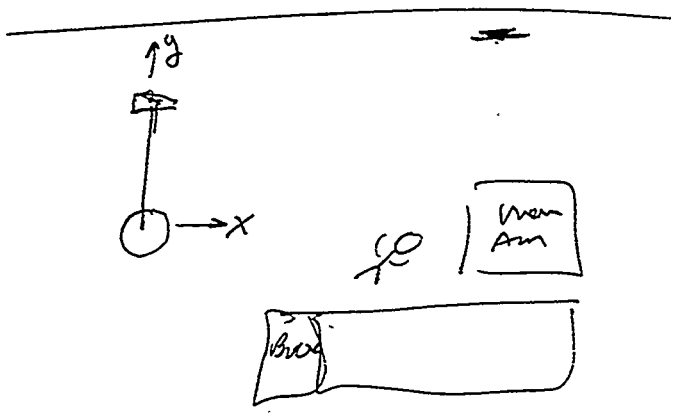
Dyn. Accuracy

St. Line 0 → 36 x
36 → -36 x
-36 → 0 x

P - Calculated from kinematics w/ Θ_{Actual}

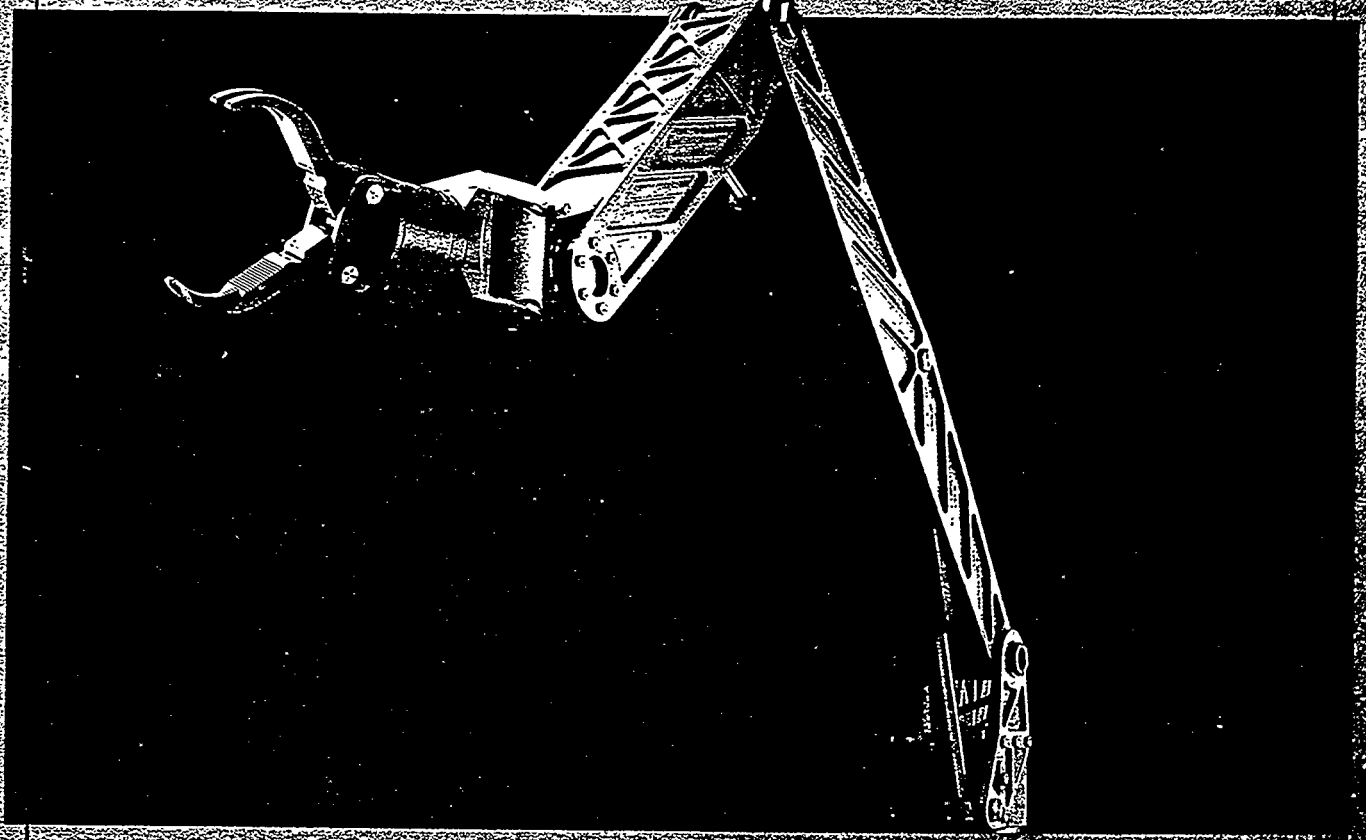






TITAN 7F and GAMMA 7F

REMOTE MANIPULATOR SYSTEMS



Schilling Development, Inc. meets user requirements for seven-function remote manipulators with two highly dexterous and powerful servo-hydraulic, Master/Slave systems. The TITAN 7F and GAMMA 7F. Each are constructed primarily of 6-4 titanium and employ advanced electronic and mechanical design for use whenever manipulative tasks must be performed in locations or environments where man cannot safely or practically venture. Applications range from undersea to radioactive environments.

The TITAN 7F has acquired an impeccable reputation since its introduction in 1987 and is seeing service for a variety of commercial, scientific and military users. Typical tasks range from undersea salvage, maintenance and construction to ordnance handling and toxic cleanup.

Building upon the TITAN 7F's established record of reliability and performance, the GAMMA 7F provides for operation in radioactive environments. Through careful selection of radiation resistant materials, the GAMMA 7F is designed to tolerate an accumulated exposure of 10^7 RAD gamma radiation. Now a standard, commercially proven product is available that eliminates costly development, rework costs and reliability concerns associated with custom designed equipment.

The TITAN 7F and GAMMA 7F are controlled by compact master arms that provide for comfortable and intuitive manipulator control. Advanced hydraulic and control systems technology combine smooth and fluid operation with an extremely tight control loop giving the manipulators human-like speed and accuracy.

FEATURES:

- Available for undersea and terrestrial applications.
- Microprocessor, servo controlled.
- Dexterous and powerful.
- 250 lb. payload at full arm extension.
- Radiation hardened to 10^7 R.
- Oil hydraulic and silicon base fluid compatible.
- Titanium construction.
- Portable master controllers that are simple to learn, easy and comfortable to operate and require little space.
- Compact and powerful three-axis wrist assembly.

SCHILLING

GENERAL DESCRIPTION

Mode of Operation ..... Spatially Correspondent
Input Device ..... Compact Master Control Arm
Number of Functions ..... Seven
Power System ..... (TITAN 7F) Oil Hydraulic
(GAMMA 7F) Multi Fluid Compatible

DIMENSIONS AND SPECIFICATIONS

SLAVE ARM

Maximum Reach ..... 78 inches
Lift capacity at Full Extension ..... 250 lbs.
Jaw Capacity ..... 4.0 inches (standard)
Jaw Closure Force ..... 350 lbs. max.
Weight in Air ..... 147.0 lbs.
Weight in Water ..... 113.0 lbs.

MASTER CONSOLETTE

Height ..... 10.0 inches
Width ..... 6.0 inches
Length ..... 19.0 inches
Weight ..... 10.0 lbs.

SLAVE CONTROLLER ASSEMBLY

Height ..... 4.0 inches
Width ..... 7.5 inches
Length ..... 16.0 inches
Weight in Air ..... 27.0 lbs.
Weight in Water ..... 14.5 lbs.

PERFORMANCE

	Hardware Range	Max. Slew Rate
Waist Yaw	270°	90°/sec.
Shoulder Pitch	90°	90°/sec.
Elbow Pitch	120°	90°/sec.
Wrist Pitch	180°	400°/sec.
Wrist Yaw	180°	400°/sec.
Wrist Rotate		
Slaved		270°
Continuous		0 to 55 rpm
Wrist Torque		70 ft. lbs. (peak)

HYDRAULIC REQUIREMENTS

3000 psi - 3.0 gpm nominal

ELECTRICAL REQUIREMENTS

25 watts nominal powered by 120/240 VAC or 20-30 VDC

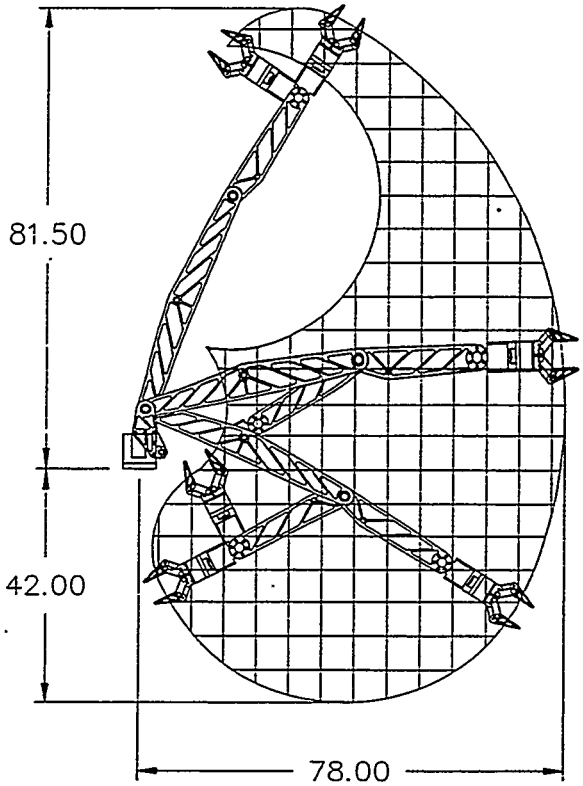
TELEMETRY REQUIREMENTS

RS-422 type media - Single twisted wire pair;
RG-108 or equivalent

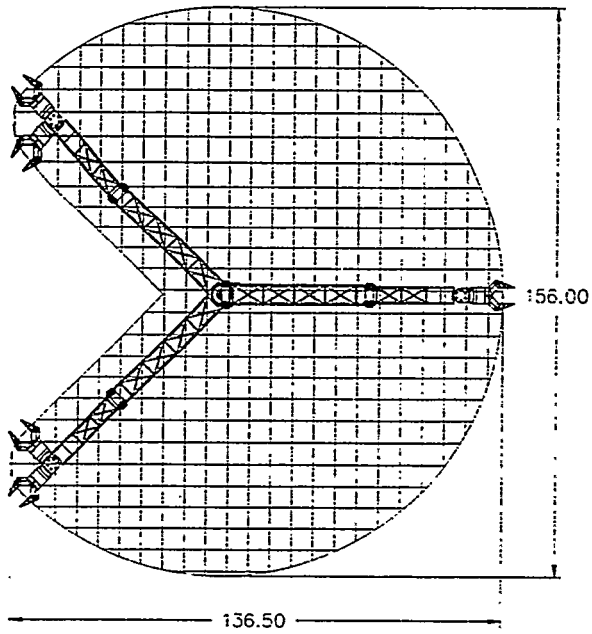
OPTIONS

Contact Schilling Development, Inc. for details.
Available in single and dual manipulator configurations.

Side Elevation View



Plan View



SCHILLING DEVELOPMENT
1632 DA VINCI CT., DAVIS, CA 95616
(916) 753-6718 • FAX 753-8092

Description and specifications are subject to change without notice. Contact Schilling Development for latest information.

1.2 TITAN 7F Description

The Schilling Development TITAN 7F is a high velocity, compact and lightweight, spatially correspondent manipulator system specifically designed for remote manipulation. The TITAN 7F has six degrees of freedom plus grip and is controlled by a master controller. Movements introduced at the master control arm by the operator are duplicated by the slave arm, maintaining the spatial correspondence between the master and the slave.

For the purposes of this chapter, the system can be viewed as three components: 1) the slave arm; 2) the slave controller; and 3) the master consolette, containing the master control arm and the master control electronics. The manipulator system is an engineered integration of mechanical assemblies, servo-hydraulics, and a microprocessor based position control system. Because the TITAN 7F is an integrated system, it is not possible to completely segregate the functions of the subsystems. Frequent cross-references will be made to the subsystems in the discussion that follows. See Figure

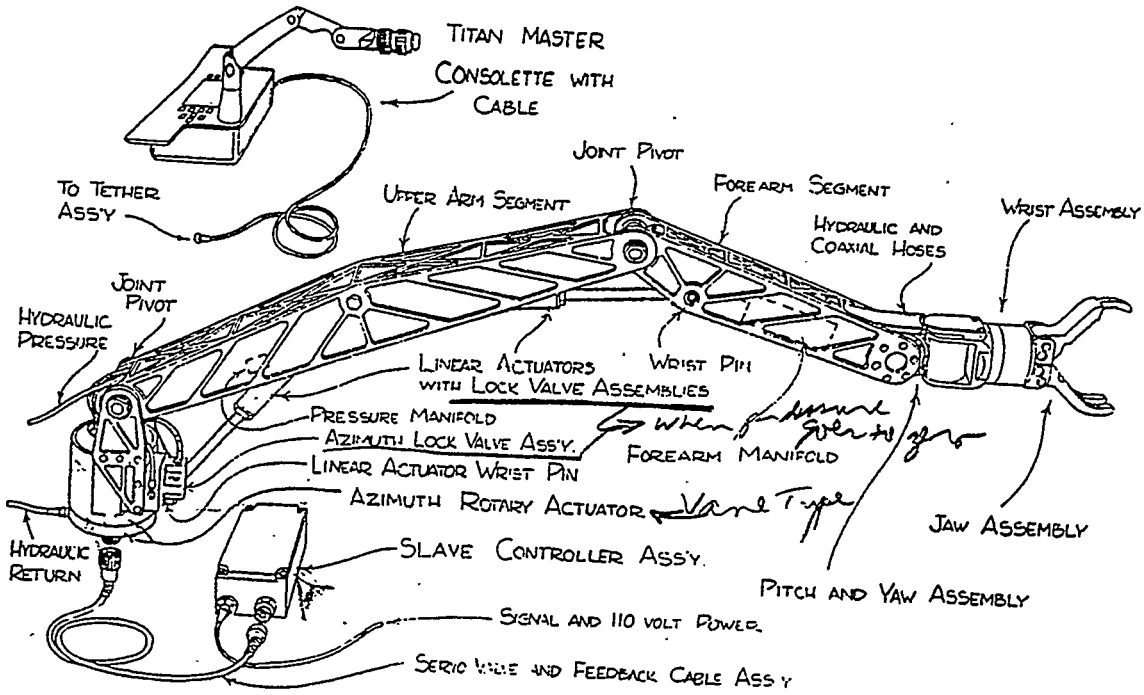
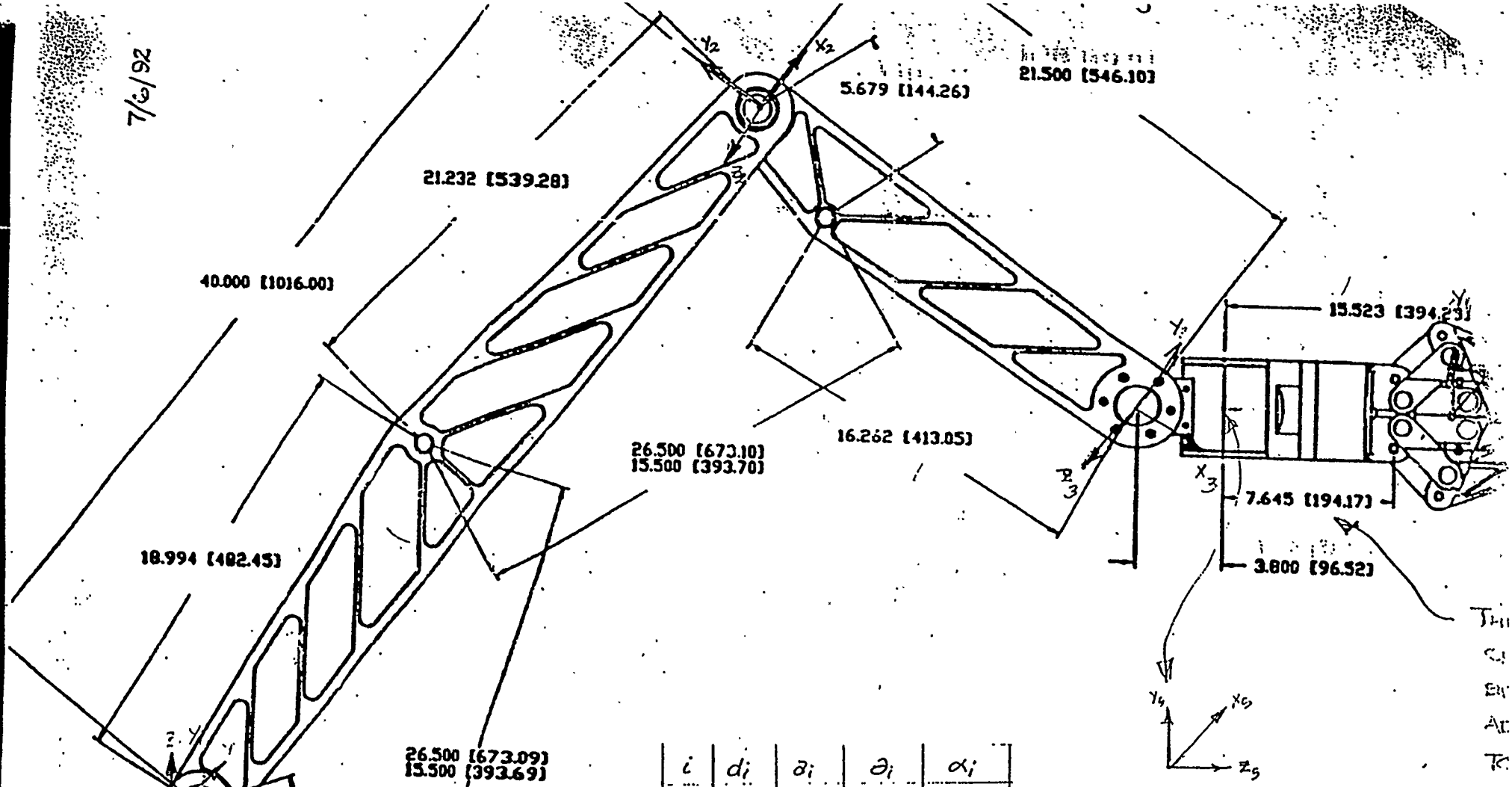


Figure 1.1-Titan 7F Manipulator System and Major Assemblies

7/9/92



i	d_i	a_i	θ_i	α_i
1	0	1.085"	θ_1	90°
2	0	40"	θ_2	0
3	0	21.5"	θ_3	0
4	0	3.8"	θ_4	-90°
5	0	0	θ_5	90°
6	16"	0	θ_6	0

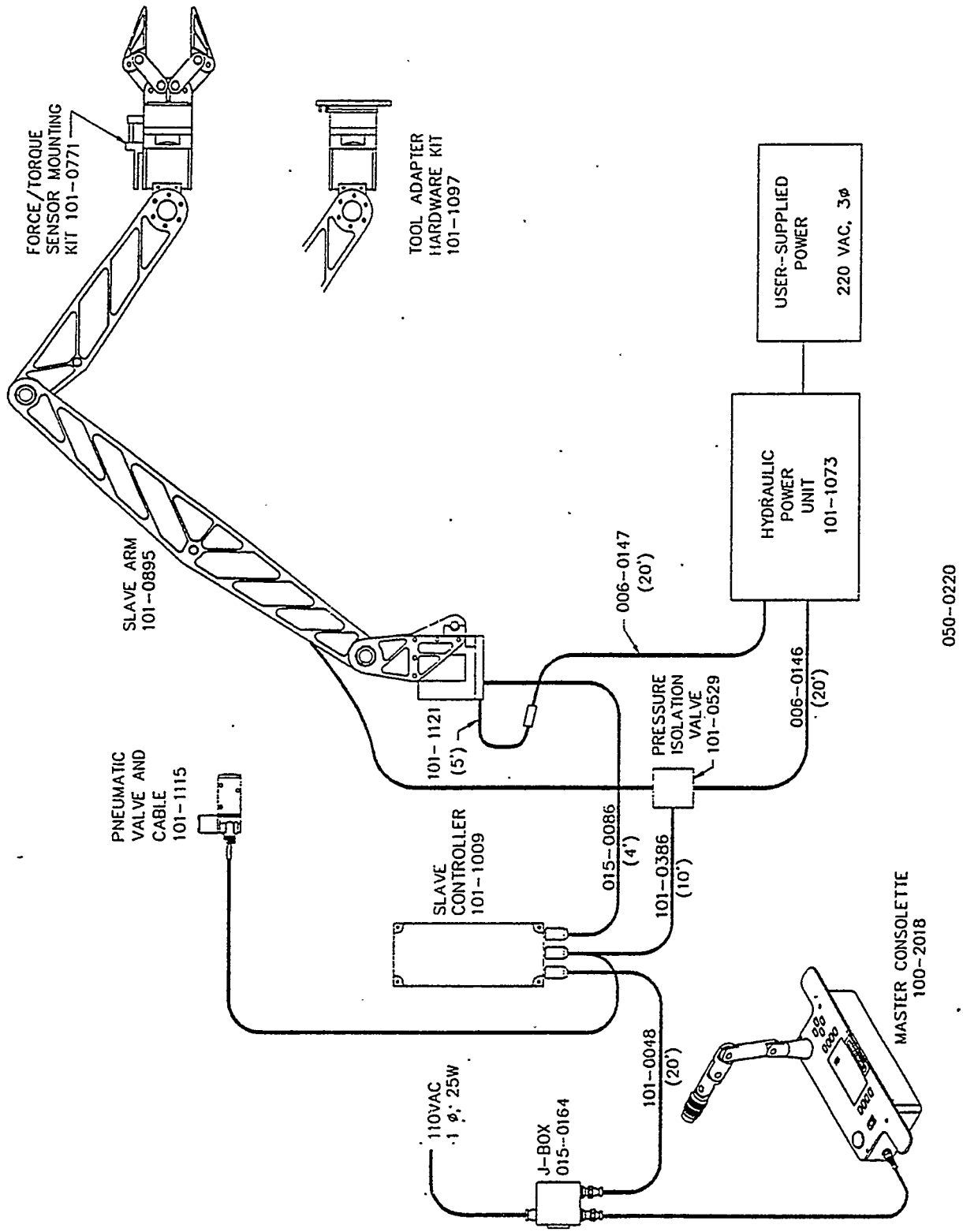


Figure 8.1 System Overview

ler electronics to determine the positioning of the master control arm. The Master Control Arm Freeze button is located on the tip of the control arm. Deflecting the button toggles the FREEZE switch between OFF and ON settings. In this version of the master controller, the function of the FREEZE button is user defineable. The 6 DOF master control arm is shown in Figure 1.3.

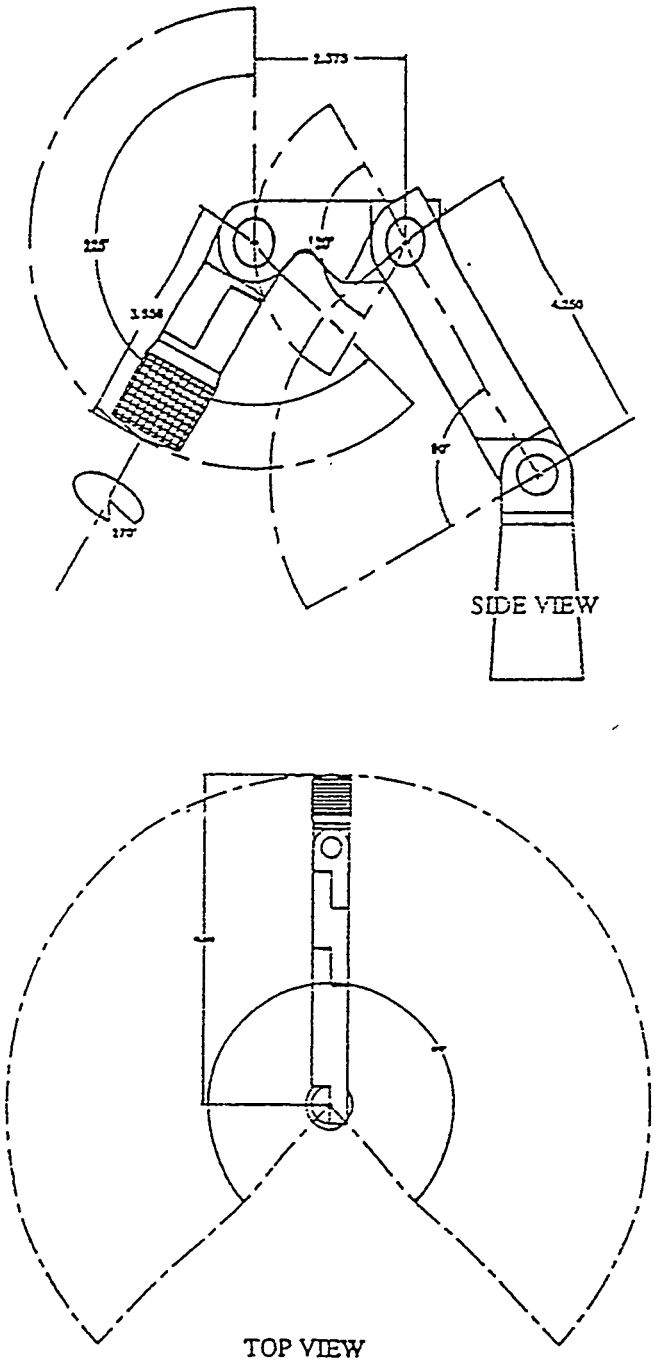


Figure 1.3. 6 DOF Master Control Arm