# The Design and Qualification of a Hydraulic Hardware-in-the-Loop Simulator

A Thesis
Presented to
The Academic Faculty

by

**Scott Driscoll**

In Partial Fulfillment
Of the Requirements for the Degree
Masters of Science in Mechanical Engineering

School of Mechanical Engineering
Georgia Institute of Technology
August 2005

# The Design and Qualification of a Hydraulic Hardware-in-the-Loop Simulator

Approved by:

Dr. Wayne Book, Chair
School of Mechanical Engineering
*Georgia Institute of Technology*

Dr. Chris Paredis
School of Mechanical Engineering
*Georgia Institute of Technology*

Dr. Nader Sadegh
School of Mechanical Engineering
*Georgia Institute of Technology*

Date Approved: 5/19/2005

# ACKNOWLEDGEMENTS

Special thanks go to my advisor, Dr. Book, for his support and guidance throughout this project and to my committee members Dr. Sadegh and Dr. Paredis. Our lab research engineer, James D. Huggins, was instrumental to getting a working system and teaching me the ways of the hydraulic world. This work would not have been possible without the generous support of many companies involved with the Fluid Power and Motion Control Center. Notable contributions for this project came from Siemens, which donated the electric motors, John Deere, Sauer Danfoss, and Hydac. Finally, I would like to thank my sage officemates, Amir Shenuda and Matt Kontz, for the never ending consultations the shared office obliged them to give.

# TABLE OF CONTENTS

# LIST OF FIGURES

# SUMMARY

The goal of this work was to design and evaluate a hydraulic Hardware-in-the-Loop (HIL) simulation system based around electric and hydraulic motors. The idea behind HIL simulation is to install real hardware within a physically emulated environment, so that genuine performance can be assessed without the expense of final assembly testing. In this case, coupled electric and hydraulic motors were used to create the physical environment emulation by imparting flows and pressures on test hardware. Typically, servo-valves are used for this type of hydraulic emulation, and one of the main purposes of this work was to compare the effectiveness of using motors instead of the somewhat standard servo-valve. Towards this end, a case study involving a Sauer Danfoss proportional valve and emulation of a John Deere backhoe cylinder was undertaken. The design of speed and pressure controllers used in this emulation is presented, and results are compared to data from a real John Deere backhoe and proportional valve. While motors have a substantially lower bandwidth than servo-valves due to their inertia, they have the ability to control pressure at zero and near-zero flows, which is fundamentally impossible for valves. The limitations and unique capabilities of motors are discussed with respect to characteristics of real hydraulic systems.

# CHAPTER I

# INTRODUCTION

In the design of new devices and controllers, oftentimes there is a need for middle ground between computer simulation and full scale testing. Simulation provides inexpensive and fast insight into the performance of new designs, but its reliability is limited by the accuracy of the models and parameters being used. In many cases, systems are too complex to be modeled adequately, exact parameters are unknown, or small changes in environmental and initial conditions produce large changes in output. All these factors mean that a single real-world test can thwart predictions from the best modeling efforts. On the other hand, real-world testing faces its own challenges, which include large costs, time expenditures, and safety hazards. A compromise between these two methods is a technique called Hardware-in-the-Loop (HIL) Simulation. The idea is to use real hardware that is central to the design question, and place it in a physically emulated environment. Thus, the flexibility, speed, and safety of simulation are combined with the reliability of using the real physical hardware itself. However, just as with complete simulation, the reliability of this method is limited by the emulation system's ability to recreate an accurate physical environment for the hardware in question.

In order to create a realistic physical emulation of an environment, both an adequate model of the environment, and actuators that can produce the flows and pressures demanded by that model are needed. Typically, hydraulic emulation systems employ servo-valves as actuators, and while these have high bandwidths, they are ill-

equipped for dealing with low or zero flow situations. This thesis will examine an alternative actuator, an electric motor coupled to a hydraulic motor, and in the end, compare its capabilities to those of servo-valve based systems. To aid in the evaluation of the motors, a case study involving a proportional flow valve as the "hardware," and a John Deere backhoe as the emulated environment was undertaken. Results from the case study emulation are compared to those from a real backhoe and valve setup to demonstrate the effectiveness of using motors in hydraulic HIL simulation.

# CHAPTER II

# Literature Review

Hardware-in-the-Loop (HIL) Simulation has been used extensively in a variety of fields as a method for dealing with the inadequacies of pure simulation in a cost effective manner. Often referred to as load simulation or emulation, perhaps the most familiar applications of HIL simulation can be found in the aerospace and automotive industry[15-17], [4],[5],[19-20]. Typical examples include flight simulators (where a human is the "hardware") and dynamometers for testing new engine designs. The technique is by no means limited to those industries, however, and can be found almost anywhere that realistic tests of system components are needed prior to final construction. For instance, a Minimally Invasive Surgical system using Haptic operations [21], hard drive control system[22], earth moving power train simulator[2-3], and software traffic simulator [14] are all examples from other industries. Oftentimes the hardware under question is an embedded controller and the environment can be emulated purely electrically [18]. Some of the latest approaches can be found by perusing the websites of companies that produce control systems products, such at The Mathworks, dSpace, and National Instruments, for customer case studies [23-25].

One observation is that there does not seem to be a common approach or significant study of generic HIL issues in the literature. Most systems are developed within industry or academia to test a specific component. The goal of this research, alternatively, is to evaluate the use of an electric and hydraulic motor pair as a general purpose hydraulic environment emulation tool, and give recommendations for the types

of environments it is most suitable for emulating. Although extensive work was done by Ramden et al [11-13] towards the design and verification of a generic hydraulic environment simulator, as noted the majority of other HIL systems have been designed for specific applications. Nevertheless, a review of several of these more specific versions will reveal insight into many of the problems faced by a general purpose system.

## *Selected Example HIL Systems*

### Electric Motor Dynamometer

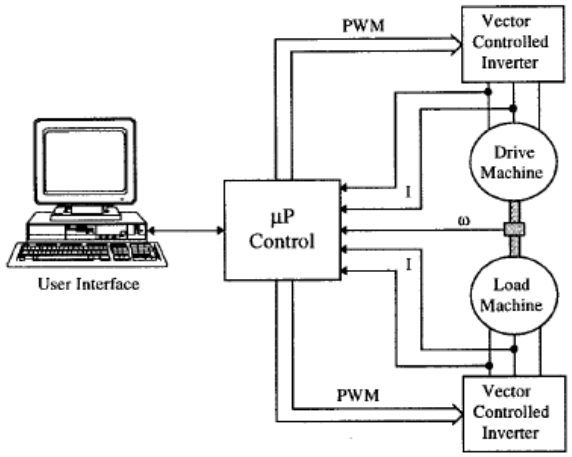


**Figure 1:** Electric Motor Dynamometer [10]

First, an example is given of a non-hydraulic system involving the emulation of non-linear rotational loads with vector-controlled induction motors [10]. A picture of Akpolat et al's experimental rig is shown in Figure 1. The goal of their work was to create a dynamometer capable of simulating non-linear loads in order to provide a more realistic test bed for new adaptive and robust control schemes. Also, the system could

greatly aid commissioning engineers by giving them off-site testing abilities for challenging applications such as high-stiction, impact, and underhauling / overhauling loads. Although no comparison was made with an actual load, the load machine was able to track the speed of several nonlinear models very closely. The authors briefly mention that comparisons with a real load would only serve to validate the model being emulated, and not necessarily prove that the system was able to accurately recreate the model. This is actually a fundamental problem that all HIL systems face in verification. In emulating a particular environment, differences between the emulation and real results could be from two sources: the emulation system's failure to physically produce what the model commands, and the model's failure to accurately represent the real environment. All one can do to show the correctness of the emulation system is show that the controlled variables align with the desired ones. Illustrating that the emulation provides a useful replication of real-world loads depends on the fidelity of the load model, and this is another issue in itself. Of course, one of the purposes of HIL simulation is to avoid modeling complex systems, so ideally the less predictable parts of a given system could be kept in the hardware portion of the loop. The torque controllers used in the inverters were controllable at 200-250 Hz, which led the authors to suggest that control at 50-100 Hz should be possible, although no experimental results on bandwidth were given.

Akpolat et al also mention another difficulty concerning causality that is common to many other HIL simulation systems. They point out that many previous efforts have relied on inverse dynamics, measuring quantities such as speed and position, and then inverting the dynamics to calculate an appropriate torque to display. While theoretically valid, practical implementation requires differentiation and filtering that limits actual

performance. An analogous limitation will be discussed in many of the hydraulics simulators mentioned below, as well as the system studied in this thesis. Because of this limitation, the authors chose to operate their system in an opposite manner by measuring torque and then displaying speed. A simple example can illustrate the causality problem of using inverse dynamics. Consider a target emulation load that has a lower static friction than the system being used as the emulator. If the emulation is measuring speed and outputting torque, it will have to wait for a speed change to react. But in a static starting position, the speed will not change until enough torque is developed to overcome the emulation system's static friction, thus imparting too high a load on the test drive. However, if torque was being measured and speed controlled, the emulation could provide a speed change at the correct level of torque from the test / drive system.

## Hydraulic Cylinder Load Emulation

An analogous problem was addressed by Qinghe et al [7] in a similarly structured hydraulic load emulator. They focused on a typical load simulator that consists of two cylinders rigidly connected, where one is driven by a test valve and the other is driven by a load control valve. In this case, a change in flow from the test valve creates an undesirable pressure / torque in the load as the system adjusts to a new flow rate. They label this a "disturbance torque," and suggest implementing a leakage port between the test and load chambers to counter the effect. This, although effective in reducing the disturbance torque, produces a corresponding limitation on the load emulation.

## Injection Molding Load Emulation

Guerrier and Edge [9] also address a cylinder-based load simulator, although theirs was designed specifically to replicate loads found in injection molding machines. A diagram illustrating the hydraulic circuit is shown in Figure 2. Originally designed as a test bed for new control schemes, this system provides a valuable opportunity to compare emulation results to those of real injection molding machines. The PQ valve shown is typical of injection molding machines, and the Load Valve is a MOOG servovalve that regulates load pressure by restricting the flow of oil out of the load side of the cylinder. Since injection molding is a meter-in process, the emulator needed only to provide resistance to flow, and not generate reverse action or over-running loads. Excessive noise below 50 Hz in the pressure feedback signal led to difficulties and instabilities which required extensive tuning as well as a gain-scheduled controller to achieve adequate performance. Once again, no experimental mention of control bandwidth is given, although they do mention that the packing phase of the process is too stiff for their system to replicate. The system was driven with a constant PQ valve opening, so it is presumed that the resulting semi-constant velocity avoided any disturbance torque effects. Also, the emulator was designed to share as many physical characteristics with real injection machines as possible (including mass); so much of the emulation was pre-made into the test setup.

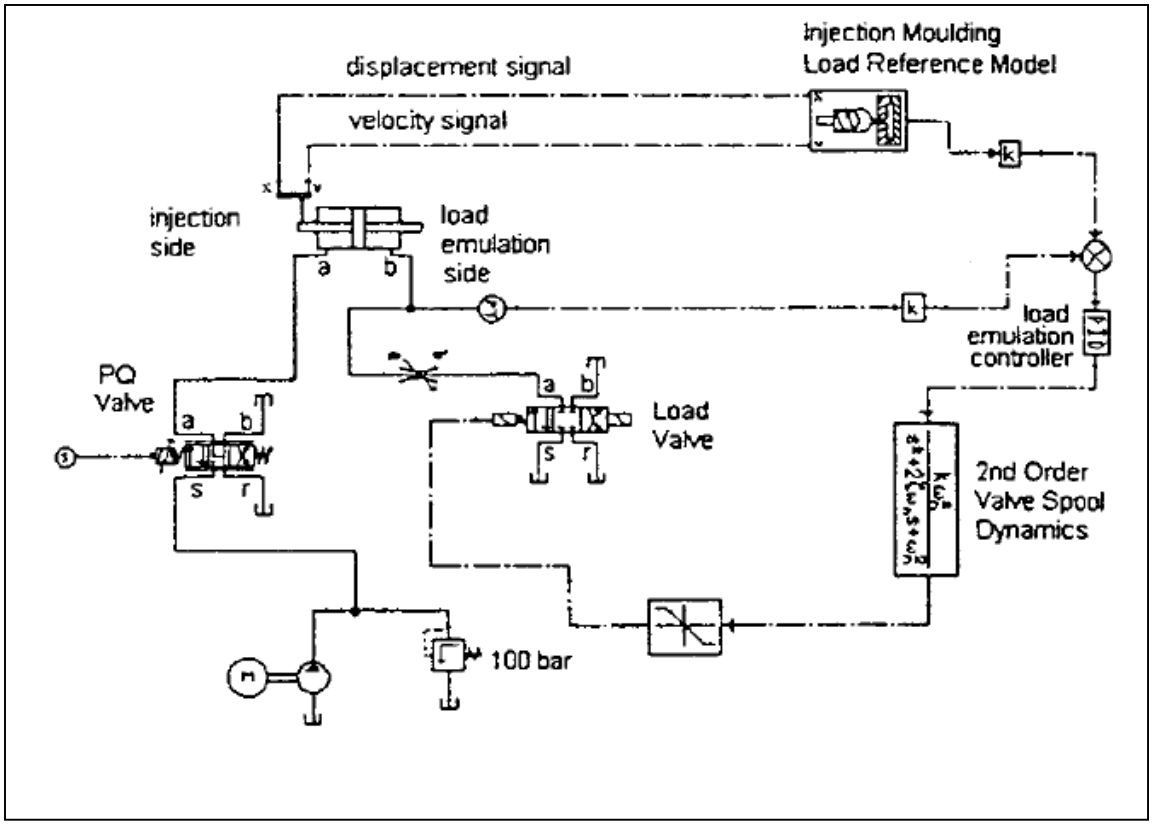


**Figure 2:** Injection Molding Machine Emulation [9]

## Earthmoving Vehicle Powertrain Emulation

Other examples of hydraulic emulation systems designed for rather specific uses include a road simulator for heavy vehicle testing [4] and an earthmoving powertrain simulator [2-3]. Gardner et al [4] demonstrate the need for tuning of servo loops contained within road simulators, and is mentionable mainly because the target application of simulating 20,000 lb trucks would be unsuitable for electrical actuators, and illustrates the need for hydraulic emulation. A copy of the simulation scheme developed by Zhang et al [2-3] for the earth moving powertrain simulator is shown in Figure 3. The objective of their research was to optimize control over the various

interacting loads in a typical earthmoving machine for the purpose of energy savings. An emulation system was developed to help verify the effectiveness of the resulting controller. In a real earthmoving vehicle, driving would be accomplished by opening a valve to a hydraulic motor attached to the wheels. In order to simulate this and other loads, hydraulic motors were coupled to pumps with restricting servovalves on their output ports (see Figure 3). In this manner, resistance would be added to the hydraulic motor's shaft by closing the valve on the pump's outlet. The design of the emulation was not the main subject of either of the two papers referenced, so it is not immediately obvious why the restricting valve on the pump output could not have been used by itself instead of the combined hydraulic motor, pump, and valve. In any case, it seems that the emulation is presumed to work for the types of loads under consideration, and no mention is given about the available emulation control bandwidth.

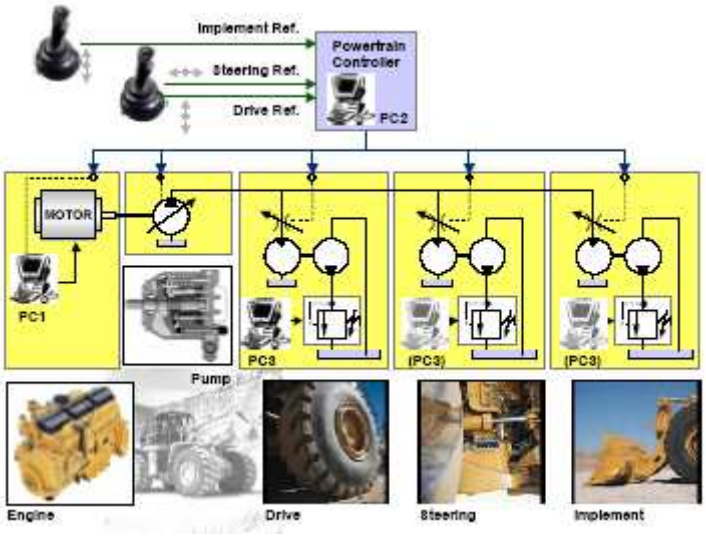


**Figure 3:** Earthmoving Powertrain Emulator [2-3]

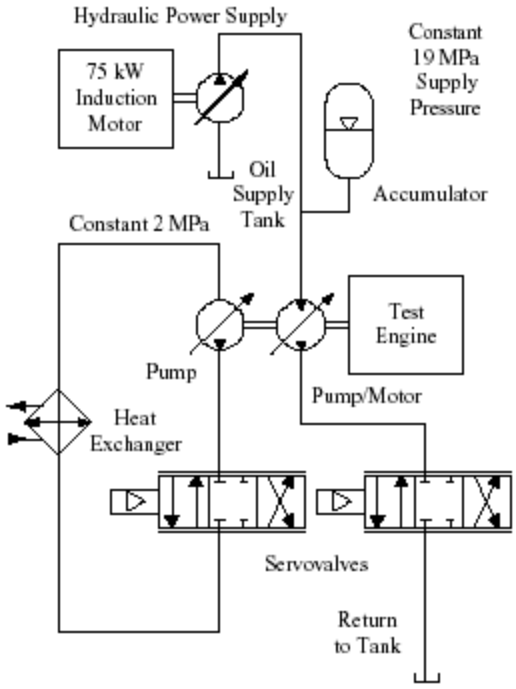


**Figure 4:** High bandwidth transient dynamometer [5]

## High-bandwidth Transient Dynamometer

A somewhat similar configuration of motor and pump is discussed in the design of a high-bandwidth transient dynamometer [5]. This configuration, however, is able to add and subtract torque from the test motor, not just resist flow, allowing the emulation of over-running loads. The circuit diagram of their system is shown in Figure 4. The dynamometer consists of a hydraulic pump and hydraulic motor attached to the shaft of the test engine. By controlling the Moog valves downstream, either the hydraulic motor can add torque to the engine or the pump can subtract torque, resisting its motion. The bandwidth of the valves used was 120 Hz, which allowed the group to simulate the torque variations from simulated cylinder combustion events. This method offers excellent performance characteristics, but is also very specific in its application. And, like many of the other examples, can only operate in one direction.

## Servo-valve General Purpose Simulator

The most relevant work towards the creation of a general purpose hydraulic environment simulator has been done by Ramden et al [11-13]. The authors present the design, analysis, and verification of a servo-valve based hydraulic emulation system, targeted mainly towards the testing of other valves. A schematic of their system is reproduced in Figure 5.

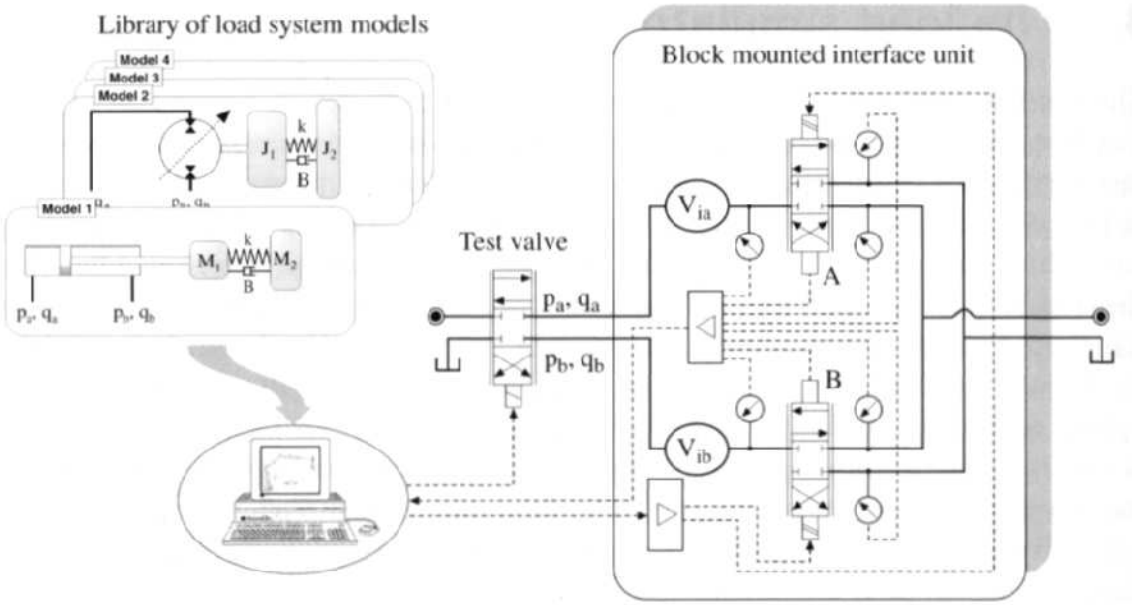


**Figure 5:** Servo-valve based hydraulic environment simulator [11-13]

The dual layout of MOOG high performance servovalves allows the pressure or flow at each port of the test valve to be controlled independently, with flow going in either direction. This has the advantage of being able to simulate single-ended cylinders with different piston areas and corresponding flow rates. Also, over-running and regenerative loads can be accommodated. The MOOG valves in the simulator have a 100% command change bandwidth (-90 deg def.) of 80Hz, so the system is more than

capable of emulating a wide variety of hydraulic systems, especially considering that most hydraulic systems have a natural frequency less than 10Hz. There are, however, many instances that hydraulic systems undergo frequencies higher than 10Hz, such as impacts in excavators, or end of stroke reactions, so the extra bandwidth is certainly useful. Another advantage of their system is that it does not suffer from a disturbance torque that many of the previously mentioned emulators encountered. This is simply because no moving mass is associated with the flow rates. The only inertia the valves have to overcome is that of the oil itself. The simulator can be operated in two modes, measuring pressure and controlling flow, or measuring flow and controlling pressure.

The only apparent limitation is that pressure errors occur at low flows (less than 5 l/min) in steady state conditions. This is primarily due to the method by which flows are measured in the device. Ramden lists many references alluding to the difficulty of measuring a large variety of flow rates with fast response times. [26-27] Positive displacement type flow meters are highly accurate, and insensitive to changes in velocity profile or viscosity, but have slow response times and are mostly limited to steady-state measurements. Alternatively, many other measurement techniques, including hot-film anemometry, laser Doppler anemometry (for local flow), the two microphone method, and sharp-edged orifices are ideal for dynamic measurements, but fail in steady state. Ramden chose to use the orifice in the MOOG valve as a measurement device, using a lookup table that related spool position and pressure drop to flow. The advantages of this approach are that no extra pressure drops were introduced, and a large variety of flow rates could be measured. The two microphone method was used to verify accuracy between 10 and 100 Hz. Unfortunately, as the flow rate through the valve drops below 5

l/min, significant errors in measured flow rate occur. This issue appears to be the only limitation to this simulation method, and the problems only occur within a small part of a valve's operating conditions. Ramden points out that flow detection and pressure control are accurate even for sinusoidal flows with zero mean (based on experimental results); it is only steady-state low flow conditions that give rise to errors. And furthermore, emulated loads have been shown to remain stable in these conditions, the only artifact being an offset in pressures.

Ramden et al have extensively analyzed and compared their simulation method to real-world loads, and found correlation within 5%. This small remaining error was shown to mostly be the result of modeling errors, and not the system's failure to correctly emulate a load.

# CHAPTER III

## Motivation and Research Question Development

Initial work using the electric motors studied in this thesis for HIL Simulation was done by Sooyong Jung and Young Lee [6]. A test case was created using one electric motor / pump pair as an emulated internal combustion system, and another pair as an emulated excavator arm lifting under gravity. While the results of that test showed basic functionality of the system, the general capabilities were unknown. It is the goal of the current research to define the strengths and limits of an electric motor / hydraulic motor pair as a tool for HIL Simulation. To aid in this evaluation, a case study was designed using a backhoe cylinder and valve as a target system. While quantitative measures will be given for the specific hardware used in the case study described below, qualitative remarks will be made about HIL systems in general that are based around motors. It is hoped that future researchers wishing to create emulated hydraulic environments may then be able to make a more informed decision about what tool to use for a particular environment.

In almost all of the previous examples, the HIL system was developed for a specific task, but in this case, the objective was to create a system that would expose some limitations of the electric motor, and at the same time highlight its unique strengths. One of the obvious limitations of an electric motor driven hydraulic motor, compared to using valves, is that it must overcome its own inertia to make changes to flow and pressure. It is unrealistic to expect the bandwidth of an electric motor based system to approach that of a servo valve based system. On the other hand, most hydraulic systems

have a natural frequency less than 10Hz, so the apparent disadvantage might not be as limiting as it seems. Many of the above systems are only able to emulate flows going in one direction, and some of those can only provide resistance, making the simulation of over-running loads unfeasible. While Ramden's work can simulate perhaps some of the most generic environments (high dynamic content, over running loads), it falters when flow nears zero in semi-steady state situations, such as a cylinder slowly starting or stopping. In fact, it would seem that all of the valve based controls from above lose controllability as flow approaches zero (and measurement ability). Without any flow, there is no pressure drop across the valve, regardless of the valve opening. It is in this situation that a hydraulic motor may have a substantial advantage. Although pressure dependent leakage and static friction effects occur at low to zero flows, the hydraulic motor still has the potential to control pressure by slowly rotating to accommodate lost fluid. Emulators that use cylinders as part of the actual hardware have no issue calculating exact flow from position, but are limited in applicability to environments with similar cylinders, and would not be able to simulate a hydraulic motor, for example.

Given all of these constraints, it seemed that a cylinder presented the most general challenge for a simulator to replicate. Depending on other parameters like mass, friction, and applied forces, a cylinder environment could be made to have a variety of natural frequencies, and also highly nonlinear loads. Also, due to the start / stop nature of its operation, flow rates would approach zero often throughout operation. Finally, a single-rod cylinder, due to its unequal areas and different flow rates, can cause unpredictable behavior in valves tuned for symmetrical flow. It was felt that a single rod cylinder was one of the most general challenges a hydraulic environment simulator would have to face,

and is also one of the most common endpoints of hydraulic power. For these reasons, a single rod ended cylinder was chosen as an environmental case study for evaluating the motors' capabilities.

A Sauer Danfoss proportional valve with load sensing was chosen as the test object. This was done partially because the load-sensing feature of the valve leads to a somewhat unpredictable set of dynamics and also because the lab had access to a real John Deere backhoe with an identical valve. The load-sensing feature enables the valve to compensate for changes in load pressure in an effort to maintain the same flow for a given command input. A diagram of the internal workings of the valve is shown in Figure 6. Modeling this valve would be very involved due to the complexity of the internal hydro-mechanical feedback mechanisms. The dynamics of the internal workings are further complicated by changes in temperature, corresponding changes in viscosity, and bulk modulus, which may be hard to predict for a given application since it can vary greatly depending on entrapped air levels. Finally, since the valve was designed to control a variety of different actuators, including cylinders of varying sizes and configurations (double, single rod) and hydraulic motors, its exact performance is also dependent on the final attachment. In the case of a single-rod cylinder, flow in one port is different from the other, leading to unequal flow forces on the spool. All of these factors mean that creating an accurate model for a variety of loads would be a substantial and difficult effort, highlighting the value of using HIL simulation to verify controller design for a variety of applications.

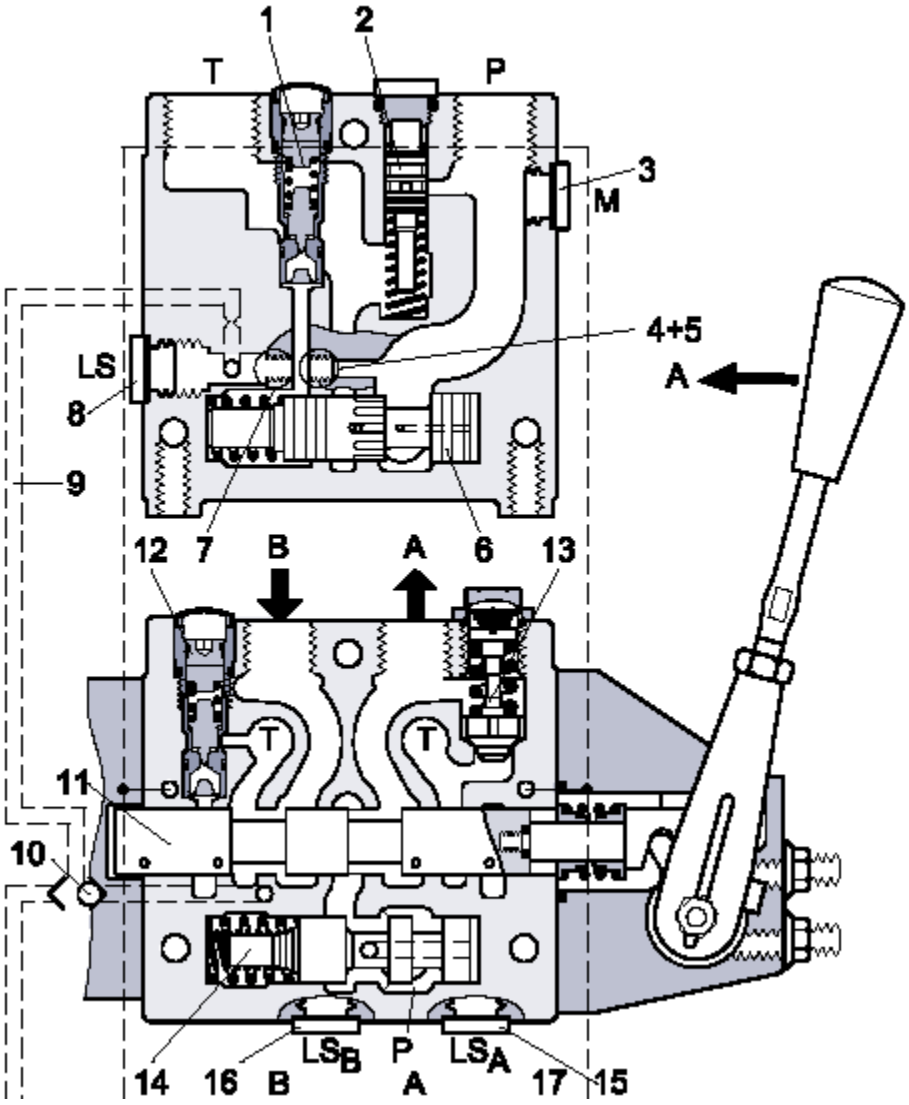


**Figure 6:** Sauer Danfoss proportional flow valve internal schematic

Although results from an actual backhoe and valve will be supplied for comparison, it is not the goal of this work to achieve a perfect match between the emulation and real system. This is mainly because an exact model of the backhoe and target cylinder would be necessary to replicate that environment, and creating a high fidelity model was not in the scope of this work. Rather, the purpose of the comparison

will be to show that many of the behaviors and features of the real environment are indeed present in the emulation, and also reveal which of those behaviors are suitable for motor based emulation, and which ones pose difficulties. Another limitation to point out is that many of the performance metrics will be germane only to the specific hardware and controllers used in this case study, but some qualitative remarks about motor-based systems in general will be made in the conclusion.

Two input and output variable pairings will be used as interconnections between the model and physical system. In one case, flows will be recorded and used within a model to calculate desired pressures, which are then fed to a pressure controller. In the alternate configuration, pressures are measured, and desired flows are sent to a flow controller. The flow-to-pressure case requires calculation of inverse dynamics, and is typical of many emulation systems, whereas the pressure-to-flow method preserves the causality of physical systems in which motion is the output variable response to a driving torque or force. The effectiveness of both methods will be compared with respect to the particular hardware in use, which necessitates the creation of two controllers: one for flow and one for pressure. The controllers will be evaluated using standard control metrics, including step-response, steady-state error, and bandwidth. Also, since low flow rates present a potential advantage for motors, the effect of low to zero flows on the above metrics will be discussed. The final evaluation will be based on how well the motors can emulate required flows and pressures under a variety of circumstances, and how beneficial those capabilities are for creating useful HIL simulation.

# CHAPTER IV

## System Development

The following hydraulic circuit was designed in order to implement an emulation of a single-rod cylinder environment for a proportional valve on a backhoe. A picture showing the actual system is shown in Figure 8.

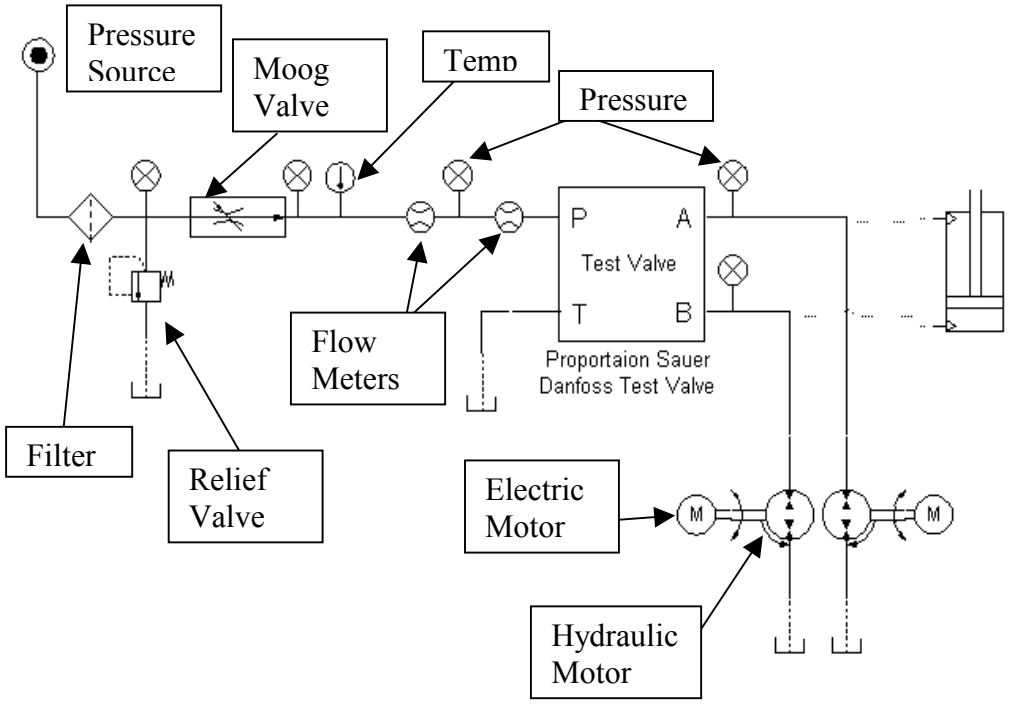


**Figure 7:** Hydraulic Circuit Diagram of HIL Simulation System. Motors are controlled to simulate single-rod cylinder as shown with dotted lines.

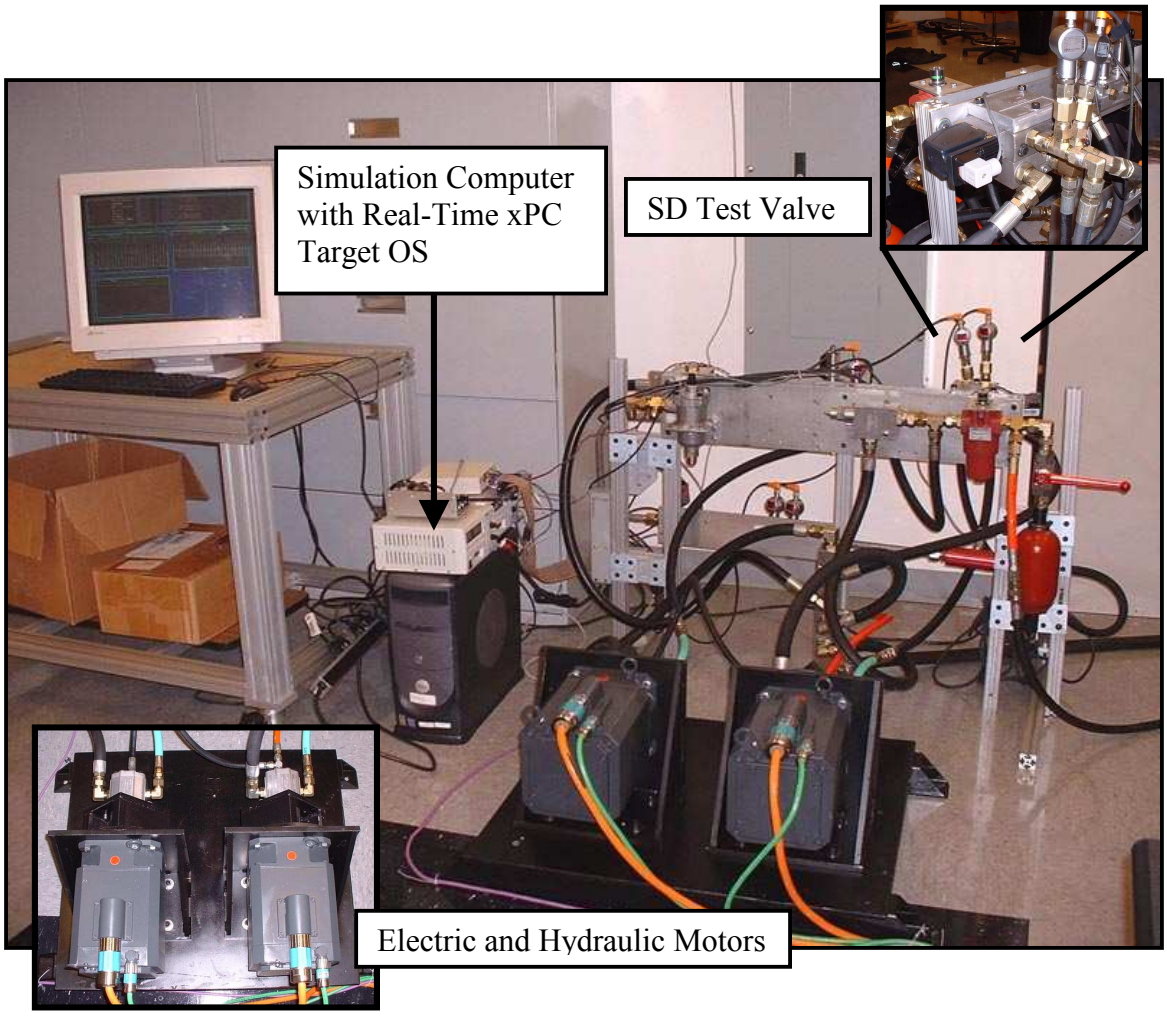


**Figure 8:** Actual HIL System

An electric motor and hydraulic motor pair was attached to each work port of the Sauer Danfoss proportional valve, allowing complete independent control for each connection. The main advantage of this method is that any arbitrary cylinder area ratio could be implemented. One difficulty of this design was that the fluid source on the actual backhoe was a constant flow source, and our lab did not have a matching constant flow pump. Instead, a constant pressure source was used in combination with a Moog valve to emulate a constant flow source. This led to the additional control problem of input flow regulation, but also has the potential in the future to allow simulation of engine

models. As described above, there are two directions of causality that could be implemented: measuring pressure, and controlling flow, or vice versa. It was desired to experiment with both, so two motor controllers were needed: one to control pressure and one to control flow. Since the input flow to the proportional valve was also being simulated, an additional flow controller for the Moog valve placed upstream was needed. To differentiate between the two controllers, and because the flow going through the motors is primarily a function of motor speed, the motor flow control is referred to as "speed control" below, while the input flow control for the Moog valve will be referred to as "flow control." These three main controller designs are detailed below.

There were also several other implementation tasks, that while time consuming and challenging, are somewhat ancillary to the main research question, and are only briefly described here.

### Profibus PCI Card Driver

The two most challenging issues were: communication with the electric motors and confining the electric motor noise from other electrical networks (sensors, analog control outputs to valves). An obvious consequence of creating a physical emulation system is that it must be run in real-time. This presents a challenge for windows-based computers since a fast, constant simulation step time cannot always be guaranteed due to unpredictable interruptions from the operating system. Mathwork's xPC target real-time operating system was chosen as a platform for implementing the real-time simulation and control. This product is basically a stripped down operating system whose only task is to run a given simulation, and can be loaded on any common PC by booting from a floppy disk. Another computer, commonly called the "host" computer, is used to create a

simulation in Simulink, compile it to real-time code using "Real-time Workshop," and then send it over Ethernet to the real-time dedicated machine, known as the "target." Parameters can even be updated in real-time from the host machine, allowing for extremely convenient control prototyping. Unfortunately, the real-time operating system ("xPC Target") did not support drivers for the PCI communication card necessary to talk to the Siemens motors, which use a common European fieldbus communication standard known as Profibus. Custom software drivers were written to allow the xPC Target real-time (OS) operating system to use a PCI Profibus communication card. No in-depth discussion of this process will be presented in this thesis, but the major steps are listed below:

- Learning how to implement custom c-code for simulations run on xPC
- Learning how the xPC Target OS communicates with PCI cards
- Learning how to write custom drivers for the xPC OS
- Learning how to write custom software applications for the Profibus PCI card
- Learning how to write custom drivers for unsupported OS's on the PCI card
- Figuring out how to transfer binary firmware to the xPC Target
- Implementation

One particularly challenging aspect of this process was the transfer of a binary firmware file to the xPC Target operating system. Mathworks provides ample explanation for implementing custom c-code within real-time simulations, but no obvious method for transferring a large binary firmware file, which needed to be copied into the onboard

memory of the PCI card. Binary files cannot simply be "copied" into c-code. In fact, firmware files for other supported PCI cards were actually built into the xPC target OS itself. A "hack" of sorts was developed that converted binary files into large integer arrays, which could then be copied into c-code, downloaded to the xPC target, and finally re-converted back into binary to be copied to the PCI card.

## Noise Control

The other major implementation challenge was confining the noise from the electric motors away from other circuitry. High powered motors are often controlled via PWM signals, which involve high frequency switching of power. This switching action, and the motors themselves, are extremely noisy and troublesome to deal with. Particularly troublesome, is that the switching frequency and its harmonics are well beyond the Nyquist frequency of the sampling equipment, so the noise is aliased into much lower frequencies which are impossible to filter digitally. A picture of this effect is shown in Figure 9.

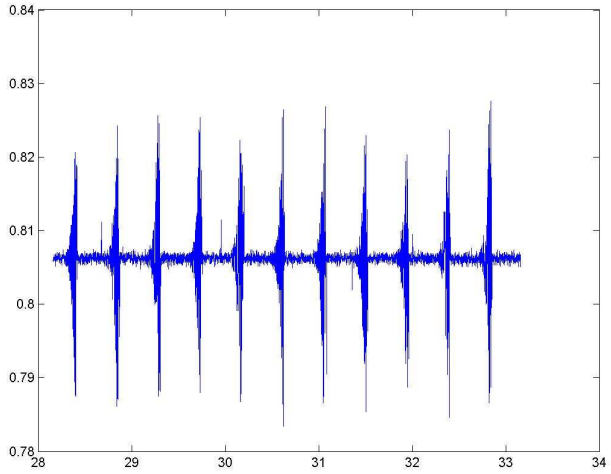


**Figure 9:** Qualitative picture of noise spikes

The scale of the picture is not given, but the spikes are higher than many actual signal levels, and they occur at about 2 Hz, which is impossible to eliminate through digital filtering without substantially affecting actual signals. Usually, noise like this can be filtered before it reaches the final sampling point. However, in our situation, it was not necessarily clear how the noise was being transferred. Some of it was undoubtedly affecting the system ground, and bypassing the low pass filters we had setup. Further complicating the situation, it was not initially possible to isolate the motors from the test-stand and sensor circuitry since the hydraulic hosing contained a metal mesh. It was believed that noise was traveling through the air, signal wires, and ground. While a final explanation for how the noise got into our system is unknown, the following efforts were somehow able to reduce it to less than 1% of maximum signal level (down from 20%).

- Using non-referenced (NRSE) signal connections [28]

- Connecting all grounds together *against* NI recommendations, including signal ground, supplied ground, and casing
- Low pass filtering input signals with RC circuit (cutoff ~220Hz)
- Use short wires where possible
- Installing ferrite beads for ultra high frequency reduction on signal lines
- Moving sensors system as far away physically as possible from amplifiers
- Installing amplifiers in grounded metal cabinet

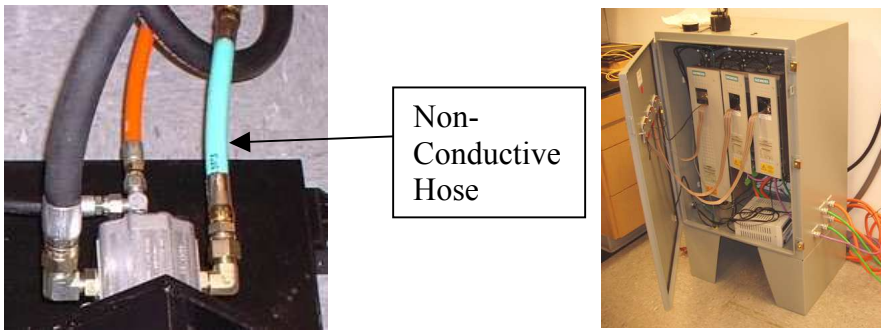


**Figure 10:** Left: Non-conductive hosing. Right: Electrical isolation cabinet.

- Isolating motors electrically from sensor networks using non-conductive hosing
- Isolating sensor networks and test stand from constant pressure pump using non-conductive hosing—very helpful
- NOT connecting the test stand in any way to grounded objects

The above list contains many counter intuitive actions that go against published recommendations. Noise control is an art-form and careful consideration of this issue is recommended before attempting to use high powered motors.

Other hardware contributions include the design and creation of an orifice based flow meter, specification of hydraulic motors and couplings (see Appendix), and implementation of a voltage-to-current amplifier for the Moog valve.

# CHAPTER IV

# Controller Design

## Introduction

The controller design consists of three main components: a pressure controller, speed controller, and flow regulator for emulating a constant supply flow. A schematic view of the interactions between the computer simulation and physical components is shown in Figure 11. Two directions of causality and interface were investigated. In the direction shown in Figure 11, pressures from the physical system are measured and fed into the model, which then calculates desired flows for the motors to track. Oppositely, flows could be measured and desired pressures calculated. As can be seen, the direction of causality has a direct effect on the input / output variables for the model.

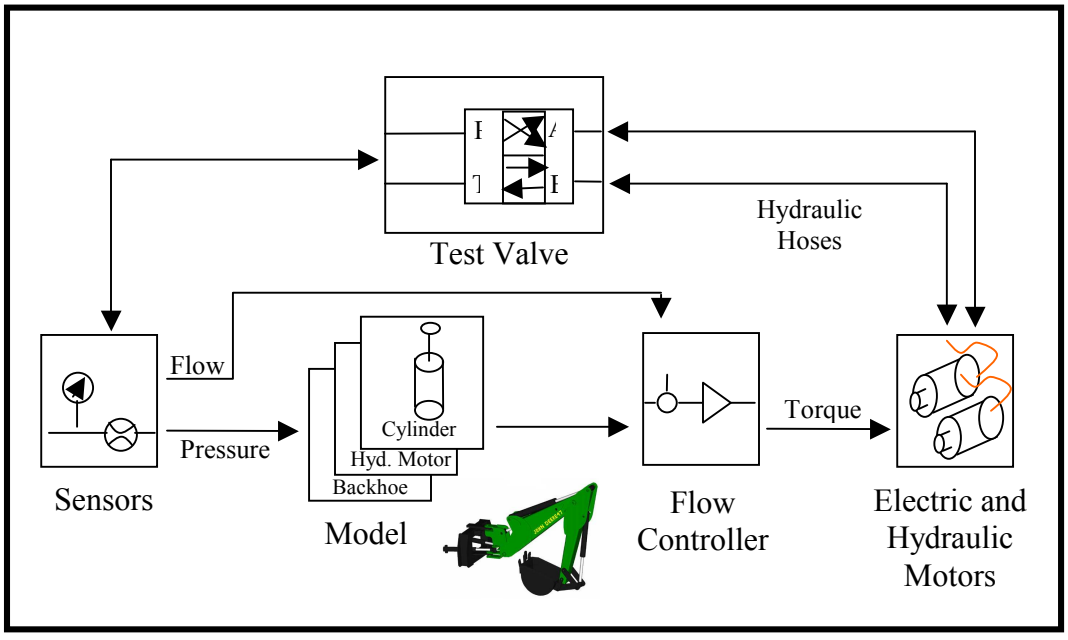


**Figure 11:** Interaction between model simulation and physical components.

The case where flow is measured and pressure controlled is the direction commonly used in load emulation systems. This, however, has the inherent side effect of disturbance torques and difficulties in physical realization since signals must be differentiated, as noted in the review of previous work. Differentiation is necessary in that case because acceleration has to be calculated from a measured velocity within the model. The special case of low flow and its effect on the pressure control will also be discussed.

## Pressure Controller

The overall structure of the pressure controller can be seen in Figure 13. The main components include a feedforward term, PD control over pressure error, and compensation adjustment for low speeds, which were found experimentally to require more torque, most likely due to static friction effects.

The first attempt at pressure control was a simple PID control by itself attempting to drive pressure error to zero. A diagram showing the hydraulic circuit created for testing purposes is shown in Figure 12. Flow was controlled with a Moog valve set to a constant value so that approximately 3-4 GPM was sent through the hydraulic motor at a supply pressure of 1000 psi and 200 psi pressure drop across the motor. Despite extensive experimentation with the PID gains, results were sluggish, and even more worrisome, asymmetrical. One of the more effective parameter settings yielded the result shown in Figure 14, which shows a 200 psi step response going from 500 psi to 300 psi and vice versa. The 300 ms settling time in the step up and the 400 ms response in the step down are notably slow given the capabilities of the electric motors and the inertia involved. It is suspected that nonlinearities relating to the pressure drop across various

orifices, including the Moog valve and also the flow meter, led to the asymmetrical behavior in the pressure response. As will be shown later, the steady state pressure across the hydraulic motor is almost completely linear with torque, so it seems unlikely that this relation caused the irregularities in the pressure responses.

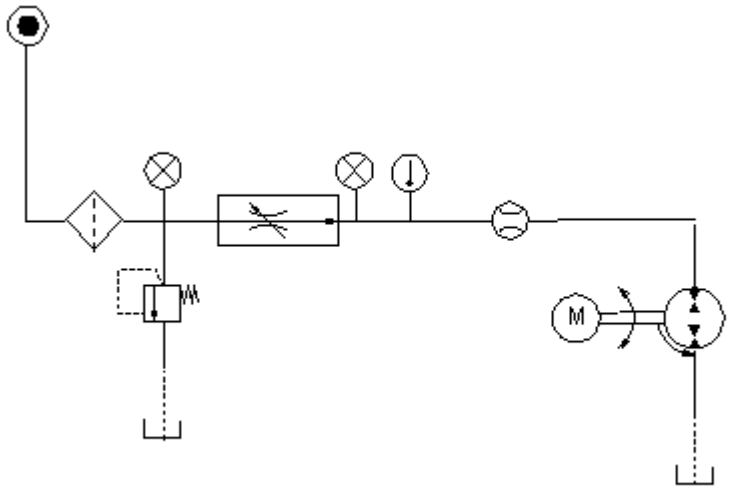


**Figure 12:** Simplified hydraulic circuit for pressure control development.

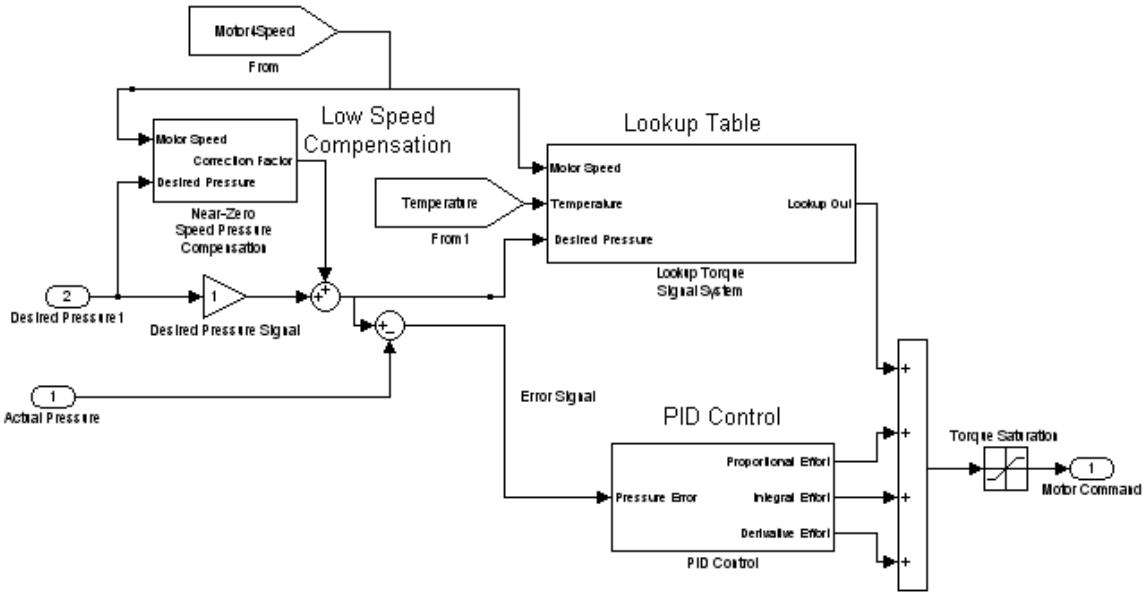


**Figure 13:** Pressure Control Block Diagram

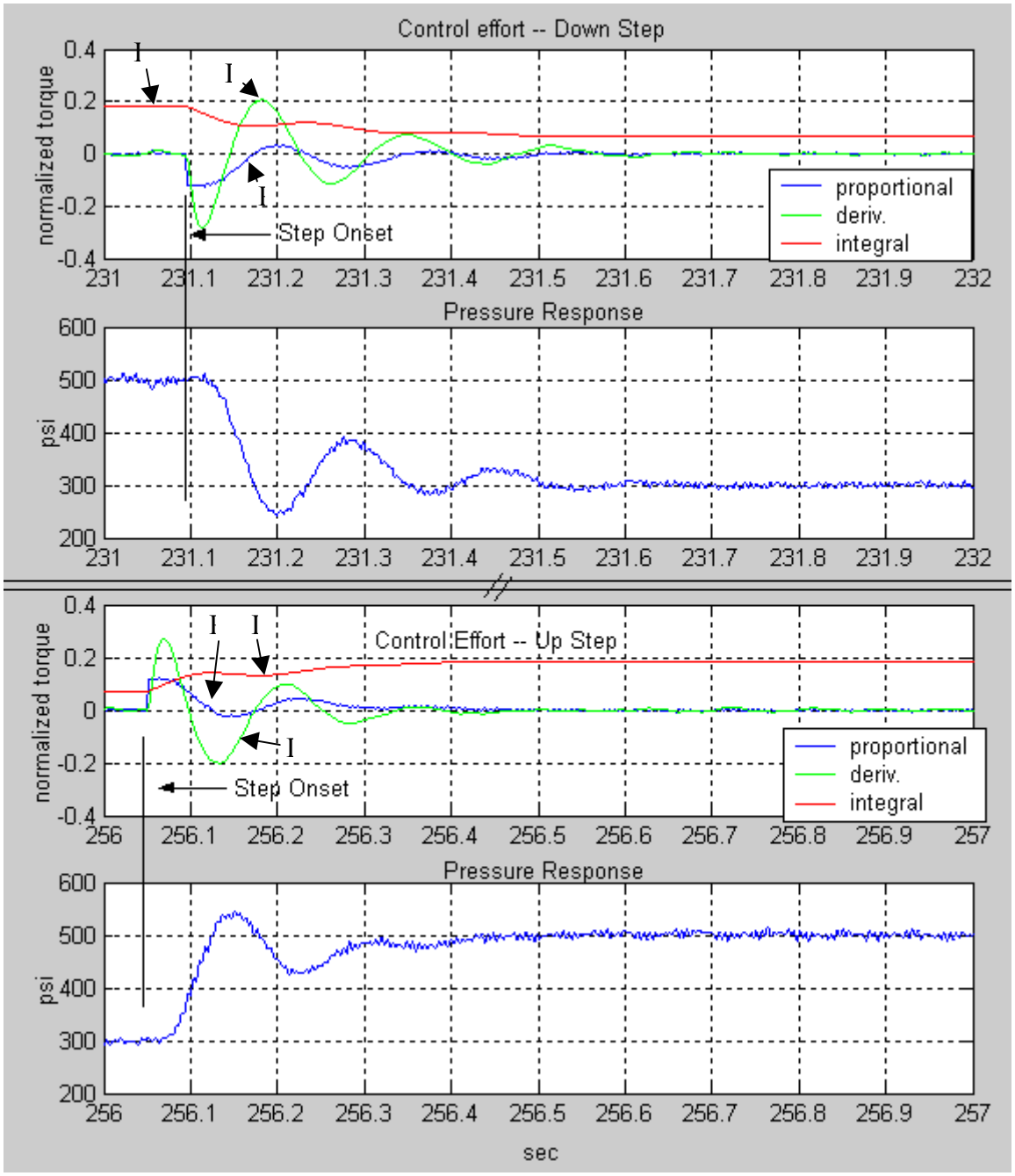


**Figure 14:** Control effort and pressure response to step down input (top), step up (bottom).

The next iteration in the pressure control design included a lookup-table based feedforward term. This was added to reduce the controller's dependence on the relatively slow responding integral term, and instead use the desired pressure setpoint to feed a torque that would yield that pressure in steady state. It was found that pressure was approximately linear with applied torque, as long as motor speed was fast enough to avoid static friction effects. A graph showing the response to the same 500 to 300 psi command input from before is shown in Figure 5, where the controller only consists of a lookup table (mostly linear) and derivative term. The derivative term helped improve a stand-alone open loop lookup-table controller by adding extra control effort at the onset of a fast command change, and also by reducing overshoot.

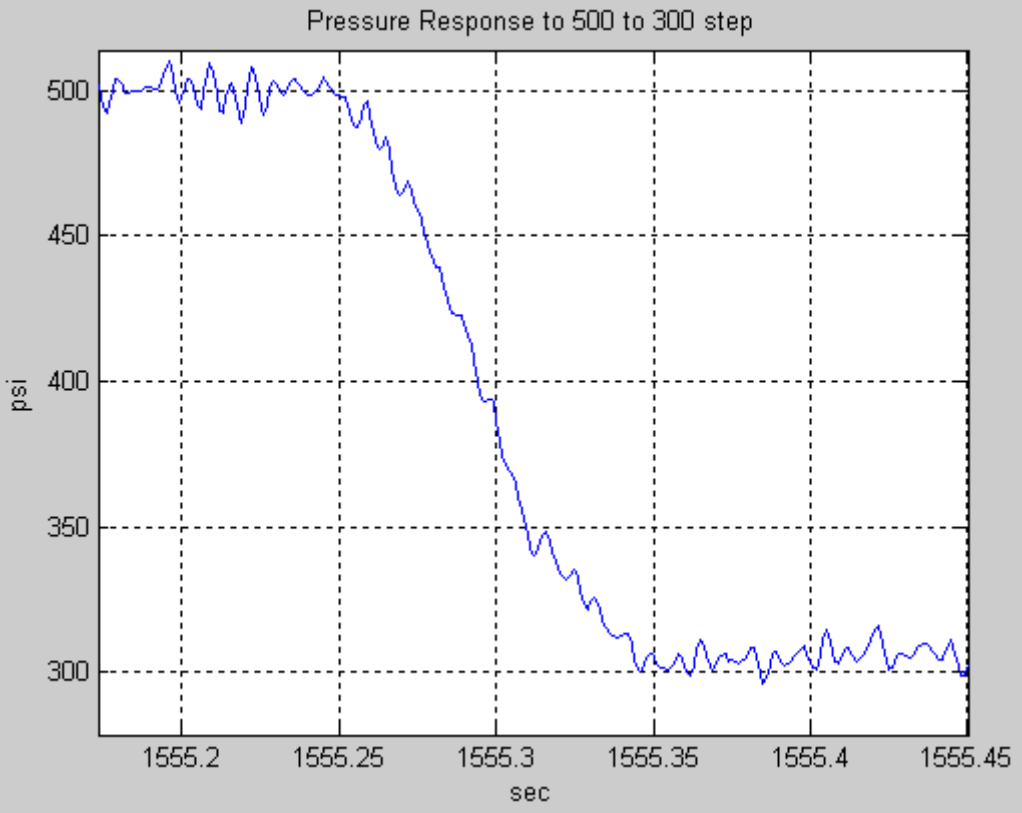


**Figure 15:** Pressure step response with Lookup + Derivative controller

Although no reference signal is available, note that there is no appreciable overshoot when compared to the noise in the signal, and that the time for the pressure to drop from 500 to 300 psi is less than 100 ms, a major improvement over the PID controller. In fact, this controller, although only a feedforward open loop command plus derivative feedback, performed better in terms of settling time and overshoot than a controller with proportional feedback. Later versions of the controller involved a proportional term, but this was added to improve steady state performance and actually detracted from the transient response. Refinements to the lookup table were made and detailed later, but first, the importance and severe dependency on filtering will be discussed.

Although the electrical noise from the motors was reduced to levels close to background noise, the remaining noise is likely the limiting factor in the performance of the pressure controller due to its dependence on a derivative. The noise in the system, despite being less than 1% of the full signal range, is a higher percentage of the pressure range actually used, which is approximately 0 to 1500 psi. Since the full range was 3000 psi, this means the effective noise is doubled. Still, 2% is a very small portion, but unfortunately the character of that remaining noise made it extremely difficult to remove with filtering. As discussed earlier, due to aliasing, much of the electrical noise occurred at frequencies (post sampling) well within the operating range of pressure dynamics. Also, although it would be hard to determine which part is which, some of the low amplitude pressure oscillations were undoubtedly due to actual variations in pressure. These could be due to a number of sources including the PWM actuation frequency in the

Sauer Danfoss valve (found in later hydraulic circuit configurations), pressure oscillations from the hydraulic motor gear teeth, and small resonances between various hydraulic components, hoses, and actuators. In any case, the remaining noise, although small compared to the sensor's overall range, was quite problematic to filter. A picture of the raw noise with associated DFT is shown in Figure 16.

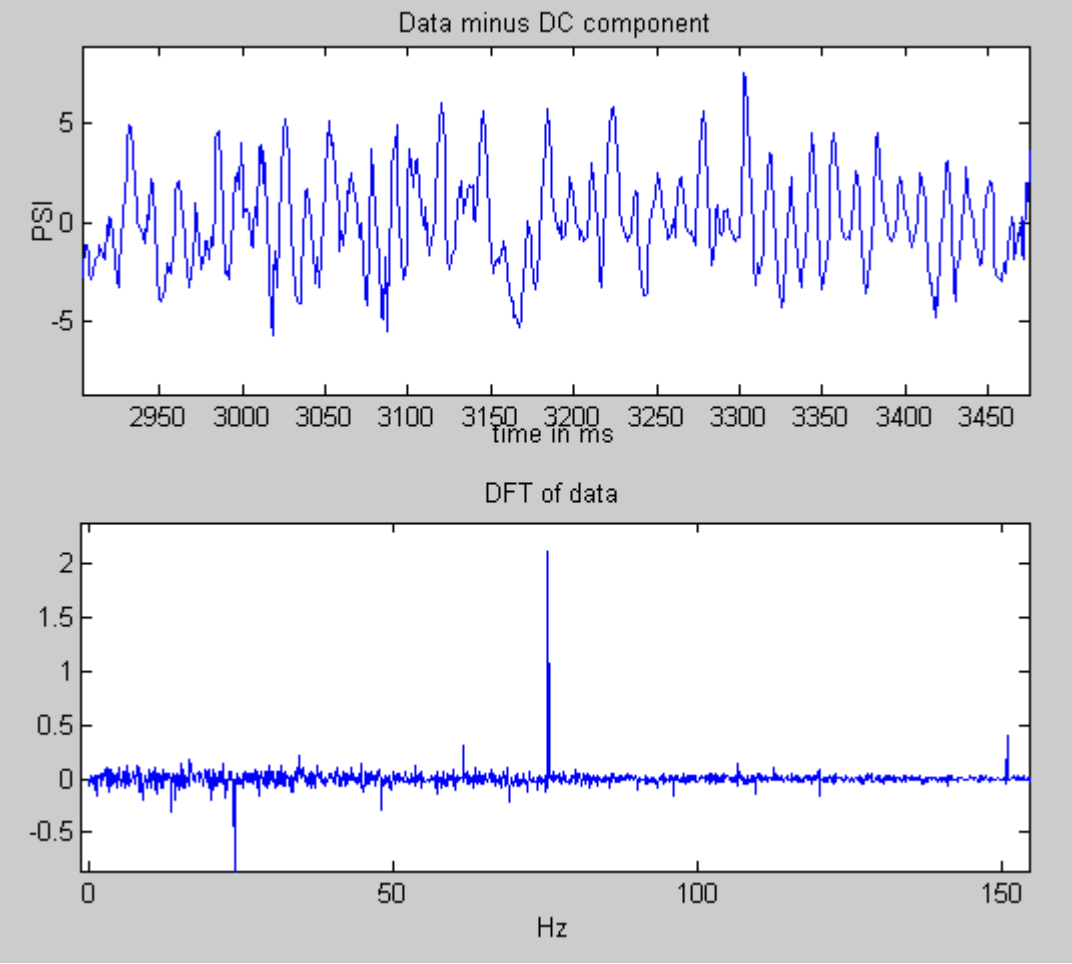


**Figure 16:** DFT of pressure noise.

The spikes at 24 and 75 Hz are interesting to note. However, notch filtering those spikes out of the signal proved to have little benefit overall to the controller's performance. To further highlight the problem, another graph of noise and DFT is shown

in Figure 17. This graph is from the orifice based flow meter (to be described below), in which the flow calculation involves the difference of two pressure signals. Note that the noise does not cancel after subtraction by any means, and is actually compounded.

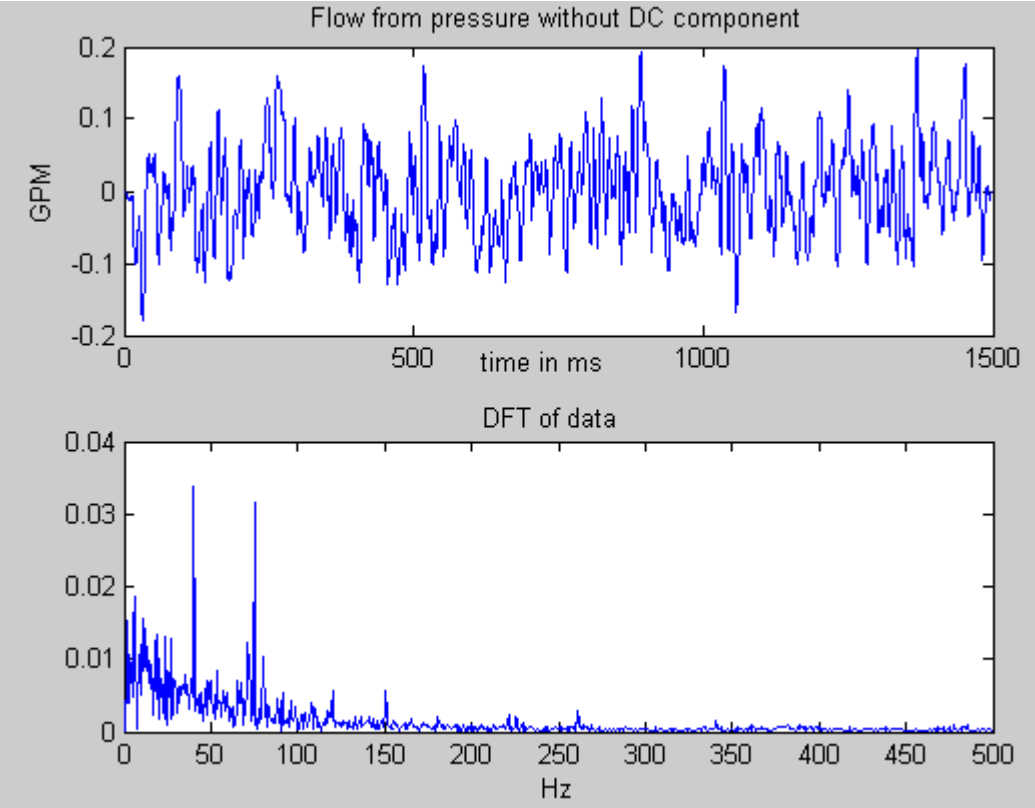


**Figure 17:** DFT of flow noise from orifice flow meter

Here, again there are two main spikes in noise energy, but the relative magnitude of the surrounding frequencies is much higher. The increase in magnitude as the frequency approaches zero Hz is particularly troublesome for filtering.

The usefulness of a derivative in a controller depends on how much the derivative signal is indicative of actual signal changes, and how much is from low amplitude noise. Unfortunately, in both the pressure controller and flow controller to be discussed later,

much of the noise was well within the range of usable pressure signals. The consequence of this was that any filter that could effectively remove the noise would also delay the signal, creating an automatic phase delay in the control signal. The final filter used for the pressure controller ended up being a 40Hz first order Butterworth lowpass filter. The type, order, and cutoff frequency were extensively experimented with, only to find that any higher order filter or lower cutoff frequency delayed the control signal to the extent that the benefit of the derivative control was negated. Increasing the cutoff frequency caused the portion of the resulting derivative from noise to approach that of actual signal changes. The end result of the overall magnitude and its frequency character of the noise was a limitation on the gain of the derivative portion of the controller. It is felt that better performance could be achieved if the true derivative of the signal was available.

In an attempt to find optimal gains for the original PID controller without lookup table, a linearized analysis was undertaken, and the resulting system was analyzed via Root Locus techniques. Besides some gain in intuition for the system, this analysis provided almost no benefit in terms of finding actual numerical gains to use in the experimental setup, and is therefore omitted from this thesis. The analysis was also limited by the number of non-linearities in the system and uncertainty of many parameters (bulk modulus in particular). The linearized model also did not take into account any limitations from noise.

After the final hydraulic system was configured, the lookup table mentioned above was refined to take into account flow rates and temperature variations. The resulting controller was able to achieve almost zero steady state error without any integral feedback and only a small amount of proportional. The diagram of the final HIL

hydraulic circuit is shown again below for reference in Figure 8. Note that the pressure transducers have been mounted close to the Sauer Danfoss work ports, and that there is about 3 feet of hosing between the sensors and hydraulic motors. The result of this is that there is a pressure drop due to flow going through the hose and fittings connecting the sensors to motors.

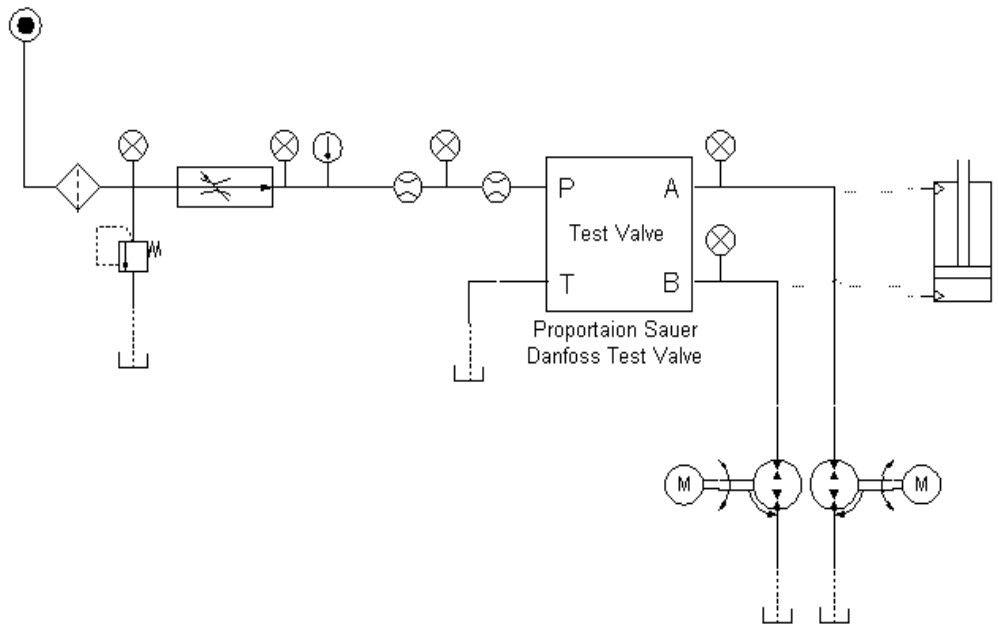


**Figure 18:** Hydraulic circuit for HIL Simulation system

The pressure drop appears to depend mostly on the standard orifice equation, but is also a function of temperature.

$$Q = k\sqrt{P2 - P1}$$

A plot of the necessary torque adjustments to achieve a constant pressure at different flow rates (or motors speeds) and temperatures is show in Figure 19.

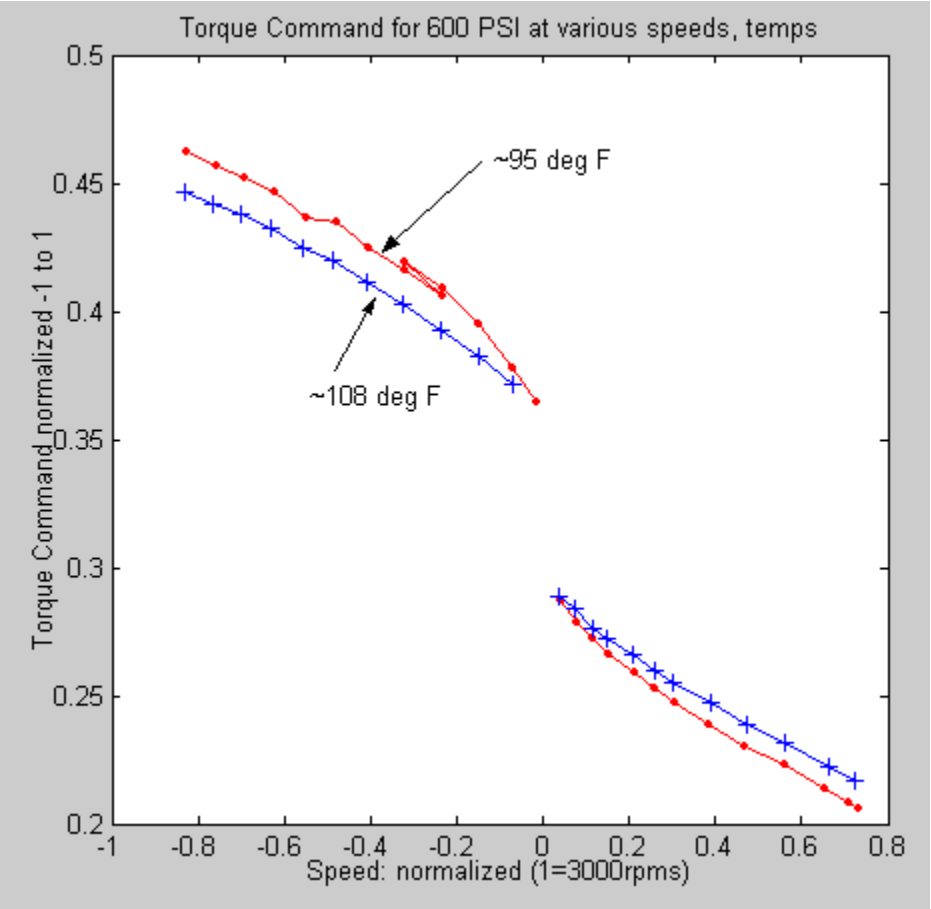


**Figure 19:** Flow and temperature effect on necessary torque

In the negative speed region, flow is being pumped from the motor into the valve, so additional torque is needed to overcome the pressure drops through the orifices and hydraulic lines. In positive speed, flow is coming from the valve and the pressure drop works in the favor of the motor, so a comparable reduction in torque is necessary. Two temperature curves show that at higher temperatures, which lead to a lower viscosity and resistance, less adjustment in torque is necessary to achieve the same pressure. In other

words, the effect of the orifices and hosing is reduced. The above data was sampled into a linearly interpolated map, and then installed in a lookup table system that would take as input desired pressure, motor speed, and temperature, and output necessary steady state torque. The lookup table system is shown in Figure 20.

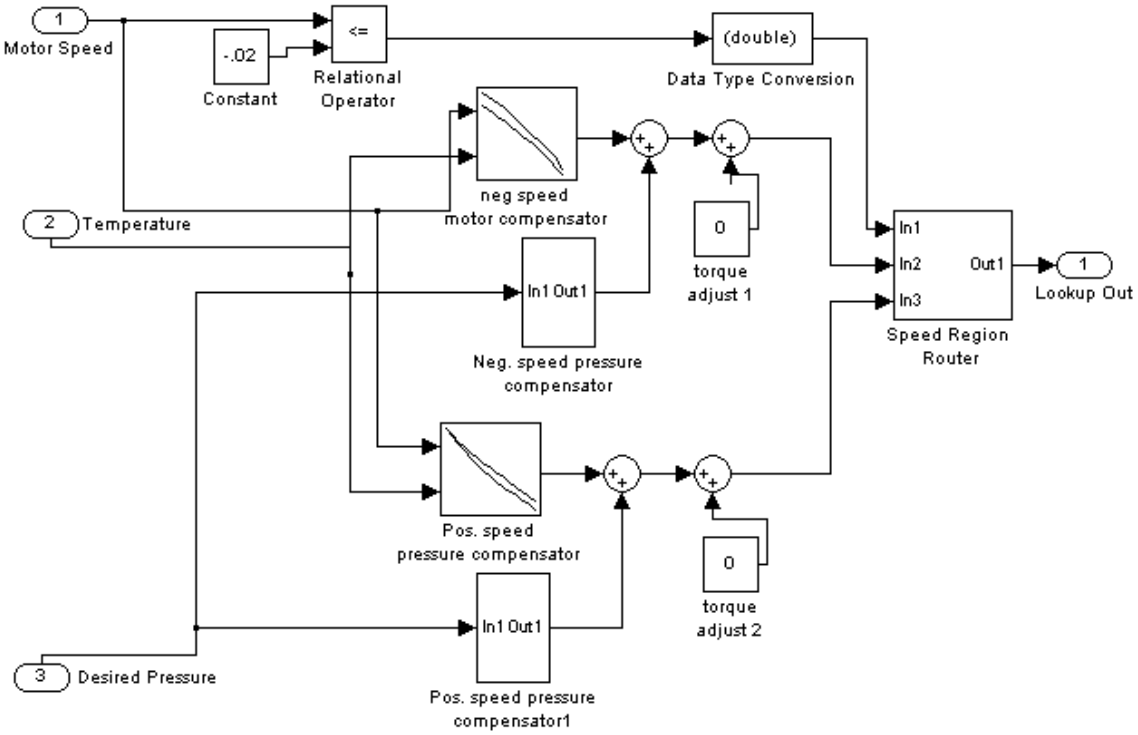


**Figure 20:** Pressure control lookup table system

The basic format of the above lookup table system is a linear interpolated map between desired pressure and torque, to which an adjustment based on temperature and flow rate is added. A different linear map and adjustment scheme is used for positive flows than the one used for negative flows. The addition of the temperature and flow dependent lookup table with a PD controller allowed for fast and precise pressure control with near zero steady state error as long as the speed remained above a certain threshold.

A final addition to the pressure controller involved compensation for zero or near-zero motor speeds. This was implemented by adding an adjustment to the desired pressure before it entered the lookup table and PD controller, shown in Figure 21.

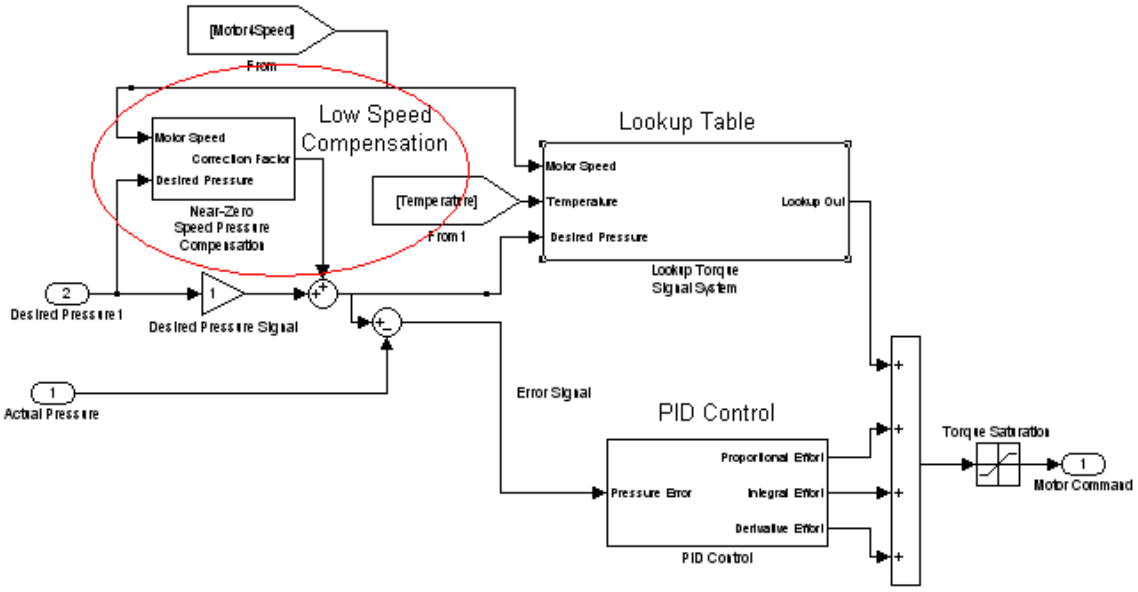


**Figure 21:** Low speed pressure compensation

The increased torque at low speeds is most likely due to static friction effects in the hydraulic and electric motor. The necessary adjustment was found by recording the increased input pressure necessary to achieve a certain desired pressure. This information was then installed into a lookup table which added pressure adjustments when speed was below a specified threshold.

Several plots which display the pressure controller's performance in terms of step response speed and frequency response bandwidth will now be discussed. Afterwards, some special considerations for near zero flows are covered.

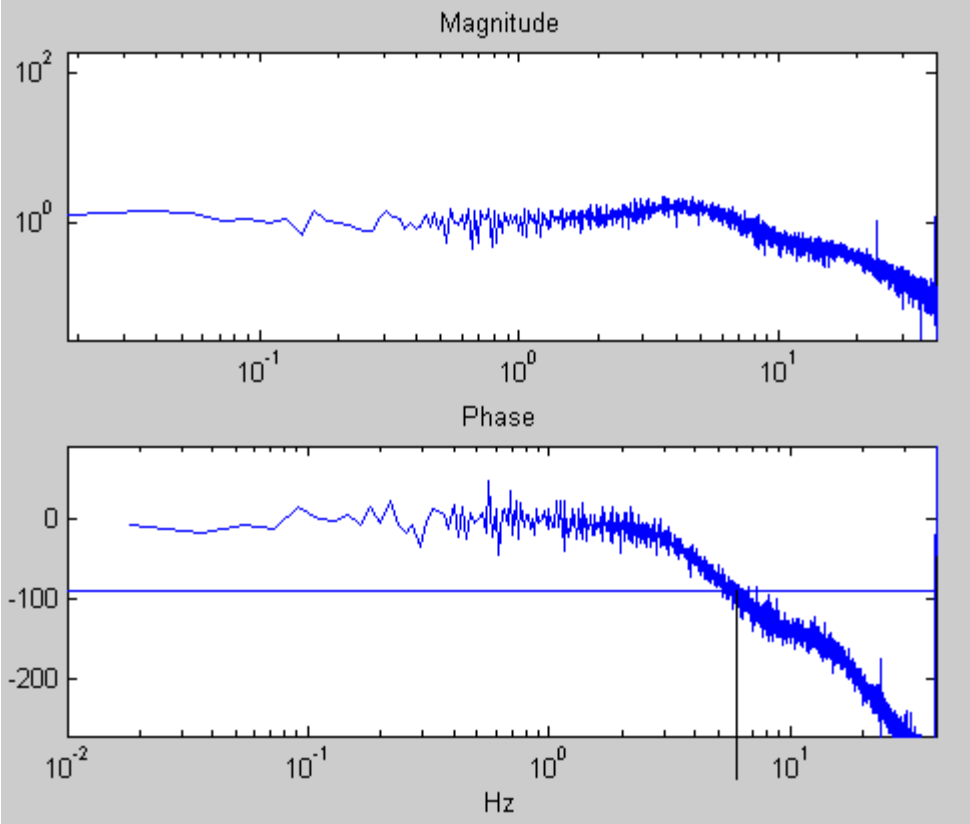


**Figure 22:** Pressure control frequency response at 2.1 GPM with 50 psi oscillation

This plot (Figure 22) shows the closed-loop bandwidth (using -90 deg. phase definition) of the pressure controller to be about 6 Hz. The plot was obtained by sending an 50 psi oscillating pressure about 300 psi to the pressure controller while a nominal 2.1 GPM was being sent through the hydraulic motor. A similar result was obtained with a 100 psi oscillation in pressure. In the 100 psi case, the small deviation from -40 dB/decade in magnitude at about 10 Hz was less apparent. An important caveat to this bandwidth is that oscillations much higher than 100 psi would probably exhibit a much smaller bandwidth, as the electric motor was nearly exerting its maximum torque during the 100 psi oscillations. It is also unlikely that 6 Hz would be achievable if the DC pressure was set much higher than 300 psi due to clipping again. Gains were

experimentally optimized to achieve a good balance between bandwidth, step response, and disturbance rejection throughout the entire range of operation. This bandwidth should be interpreted as a maximum response of the system, and not necessarily typical for the entire pressure range.

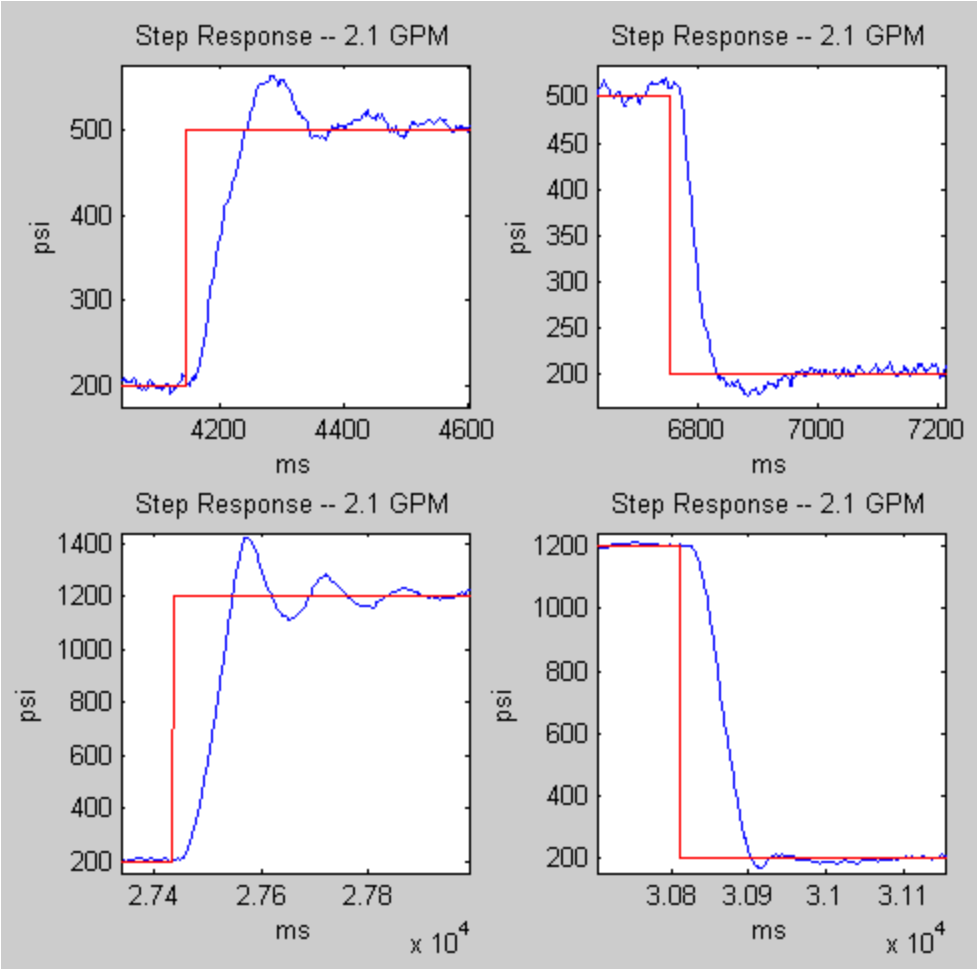


**Figure 23:** Final pressure controller step responses

The step responses show a 250-350 ms settling time when going up (depending on magnitude) and about 100-150 ms settling when a step down command is issued. There is also more overshoot and oscillation on the upward step, which is a little

troubling, but is probably due to some of the many nonlinearities, as well as interaction with the Sauer Danfoss valve's hydro-mechanical flow regulator. The lookup table + D control without any proportional effort was actually able to obtain symmetrical 100 ms settling times in both step directions. Adding proportional control effort was done to improve steady state performance at zero speed situations, where it helped counter disturbances from the slow moving gear teeth, despite having a negative effect on the transient response.

Figure 24 shows a plot of several pressure steps throughout the entire range of pressure control at 2.1 GPM. As can be seen, the steady state accuracy is very good independent of pressure despite a lack of integral control action. The plot does not show pressures below 200 psi or above 1700 psi, although the motor is capable of exceeding those bounds. At 1700 psi and 2.1 GPM, the motor is exerting about 90% of available torque, so pressures exceeding 1800 psi could be displayed although this is dependent on the flow rate. At zero flow, the max pressure the motors can reach is approximately 1600 psi, so the extra pressure it is able to achieve with flow is due to pressure losses through orifices and hosing. The higher the flow rate, the higher the pressure the motor resisting the flow can display. However, when flow goes the other direction, the motor is acting as a pump, and it must overcome the line losses, so maximum pressure would be about 1400 psi (based on pressure loss at 2.5 GPM) depending on flow.

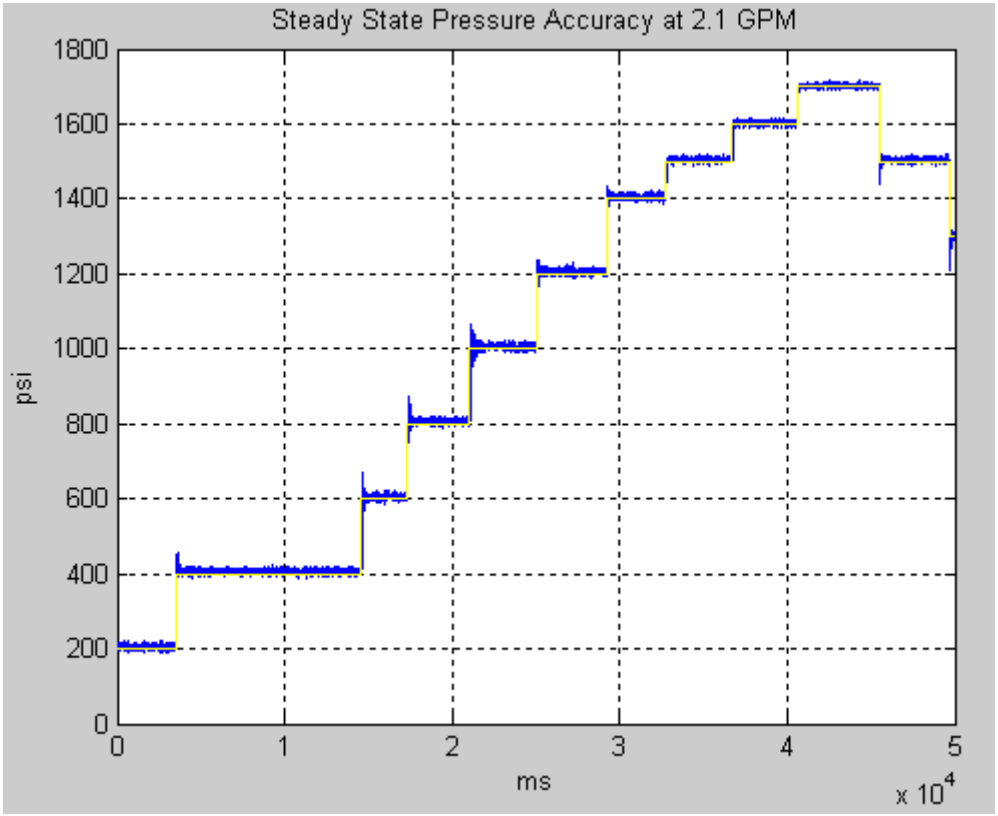


**Figure 24:** Steady state pressure tracking at 2.1 GPM.

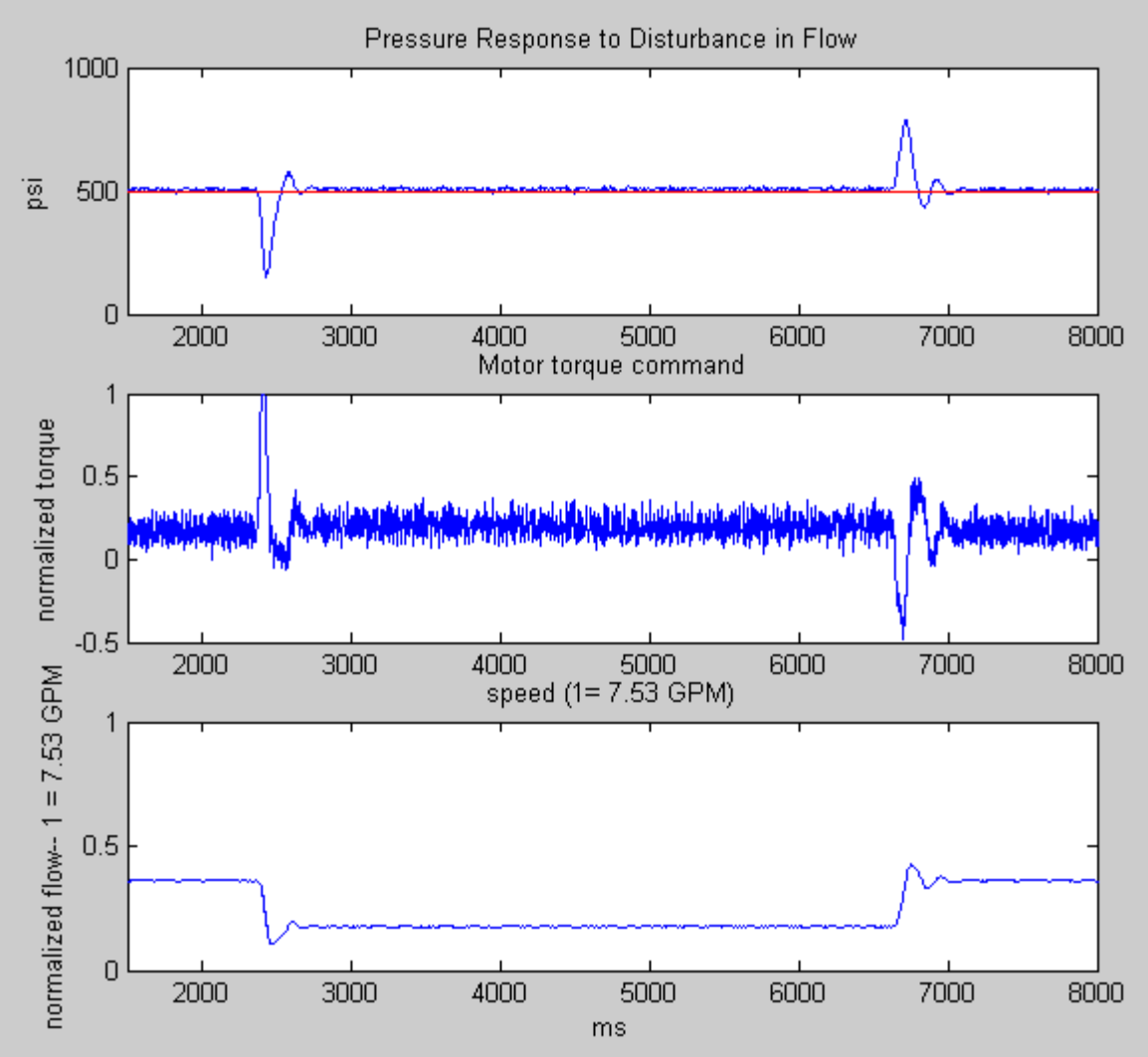


**Figure 25:** Pressure response to flow disturbance. The third plot shows motor speed, which can be approximately related to flow rates since leakage relative to this scale of speeds is small. A speed of 1 corresponds to approximately 7.53 GPM.

A final plot shows the pressure regulator's ability to respond to changes in flow. The change in flow is about 1 GPM. A perfect pressure controller would be able to maintain pressure completely independent of flow, but since some pressure must be developed to overcome the inertia of the motor, this is impossible with this system. Either a zero inertia electric and hydraulic motor would be needed, or the ability to see into the future, which may actually be conceivable in some simulations. This effect

represents one of the fundamental limitations to using a pressure control scheme: a disturbance pressure is inevitable when changes in flow rate occur.

## Low Flow Pressure Control

The next set of plots will cover the special zero flow cases. As hypothesized earlier, one of the main benefits to using motors for hydraulic environment emulation is that they can control pressure in zero flow situations. However, several complications come into play at near zero speeds, including the effects of nonlinear gear teeth pressure, and unpredictable static friction. A detailed model of the flow ripple in gear pumps is developed by Manring [1]. Figure 26 shows that the control bandwidth at zero speed is somewhat higher than the bandwidth at higher speeds. Also, the system fits much closer to that of a linear $2^{nd}$ order system. The closer fit to a $2^{nd}$ order system and higher bandwidth are probably because interactions with the hydro-mechanical flow control in the Sauer Danfoss valve are avoided.

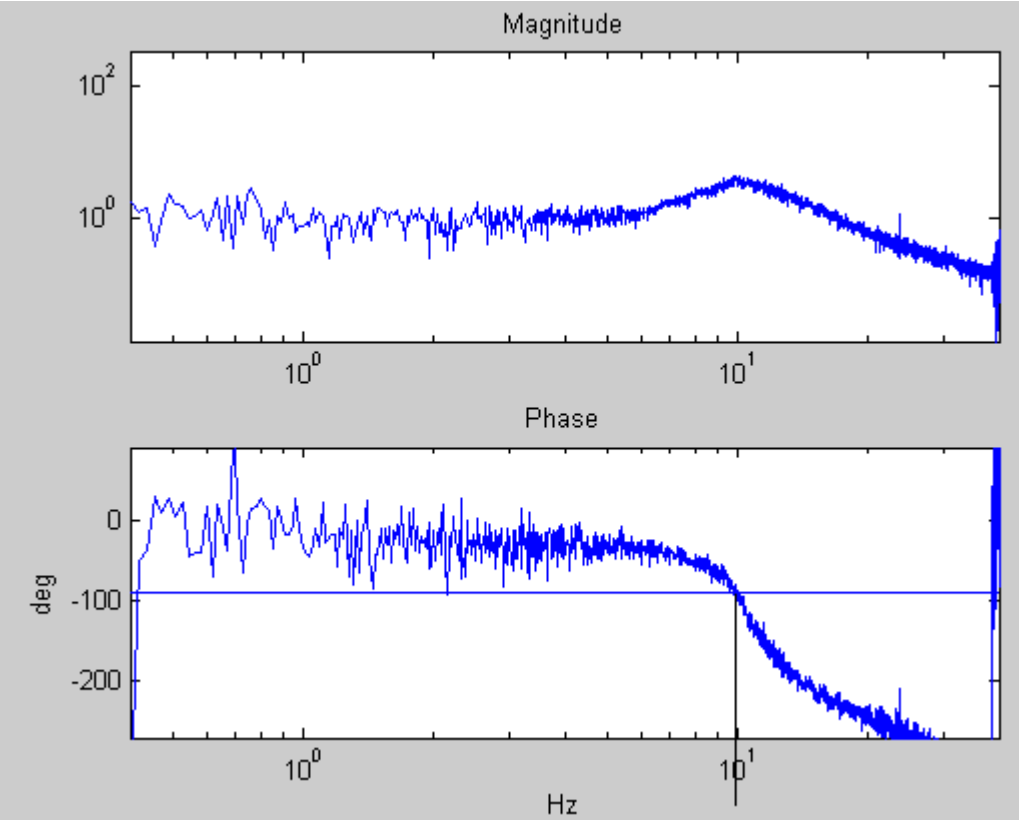


**Figure 26:** Pressure frequency response at zero flow.

All of the following plots were obtained by closing the Sauer Danfoss valve completely, and using the motor to regulate pressure in the hose leading to the valve. Figure 17 shows several step response times. All responses settle to within 5% by about 300 ms. The difference between these responses and those at non-zero speeds seems to be that overshoot occurs both in up and down steps.

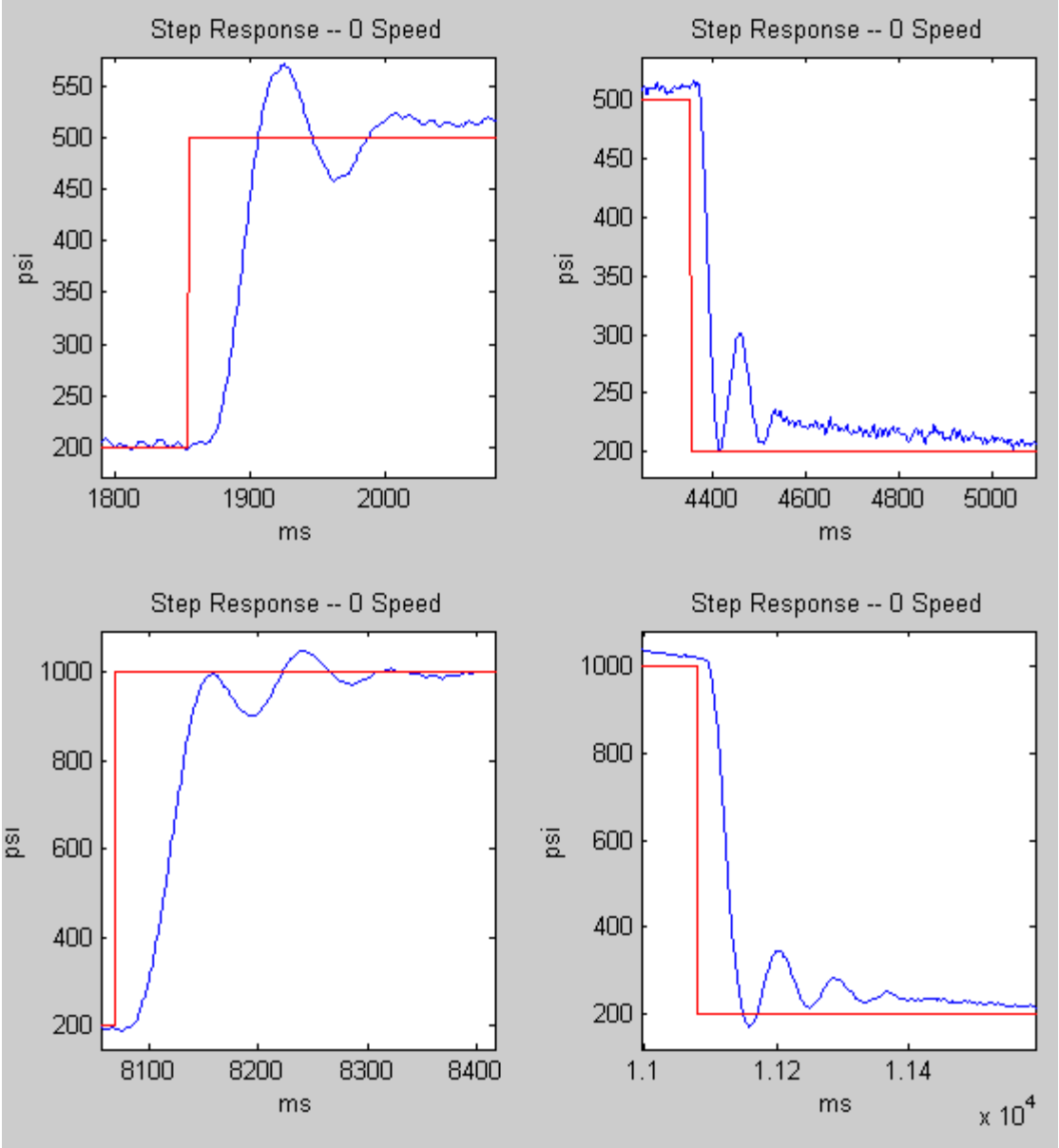


**Figure 27:** Pressure step responses at zero flow.

The following figure shows the non-linear effect that the slow moving gear teeth have. As the gear in the hydraulic motor is turned, the teeth exert a pressure on the fluid that is by no means constant. The data seems to indicate that at certain positions, almost no pressure can be exerted and leakage flow increases dramatically, causing the motor to jerk and create a spike in pressure. The higher the pressure, the more leakage occurs in

the hydraulic motor, and the faster it must spin in order to compensate. This explains why the spikes are more frequent at higher pressures. The low-speed compensator discussed above does a relatively good job at maintaining the correct average steady state pressure, despite occasional spikes and slow moving dips in pressure. As can be seen, 1400 psi is about the limit for effective maintenance of pressure, and beyond that variations approach 200 psi.

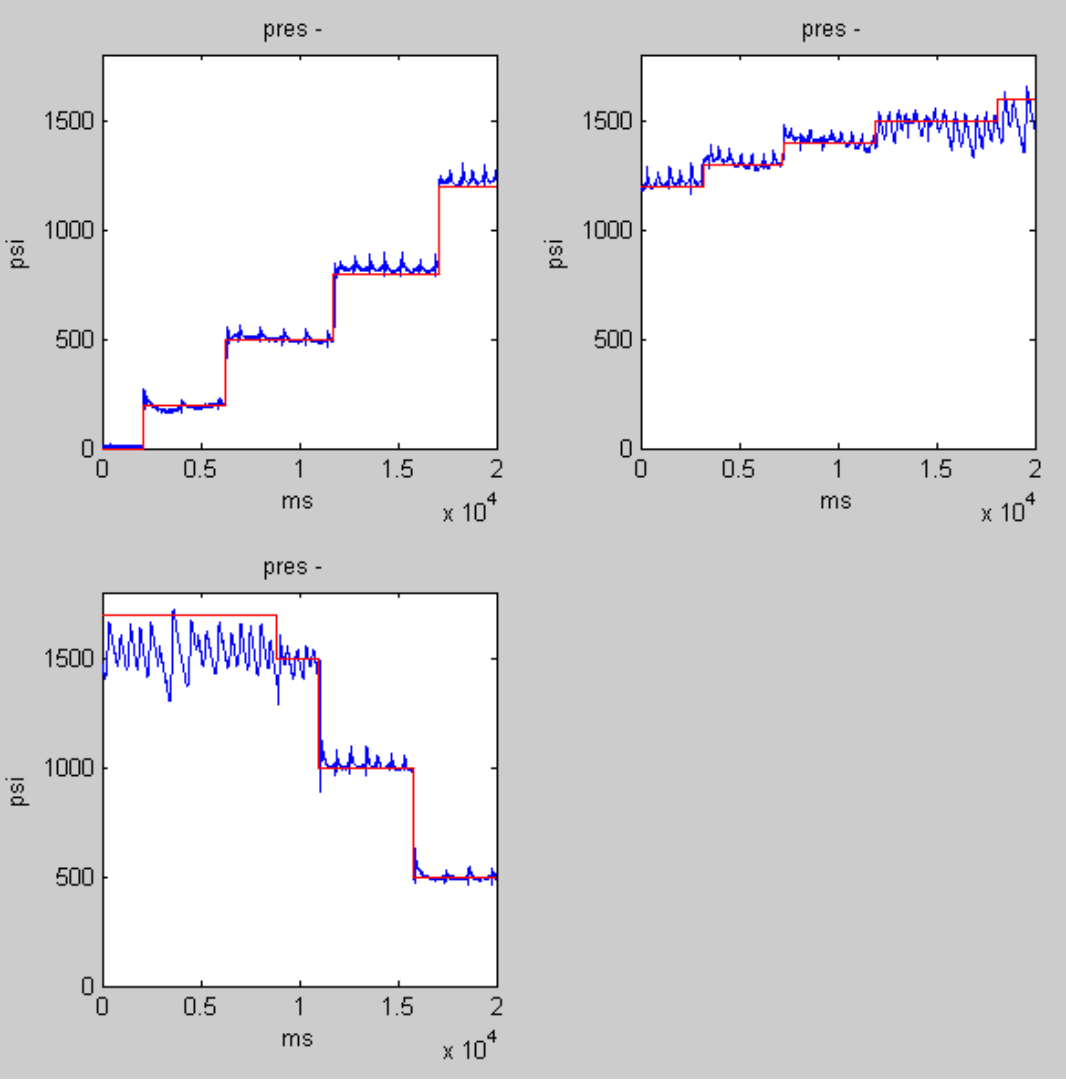


**Figure 28:** Steady-state pressure control at zero flow.

The next picture (Figure 29) gives a clearer picture of gear-teeth effect. The spikes are notably smaller at 200 psi than 1000 psi. Figure 30 shows that another effect at low speed seems to be a slower overall oscillation in pressure, possibly related to the absolute angle of the gear, although this is surprising since the gears are thought to be highly symmetrical.

Despite being left out of the final design, a pulse control was experimented with to combat the spike effect. The basic idea was to add a pulse of negative torque to the motors when a spike in pressure was detected to keep the main controller from overcompensating and causing a jump in pressure. This was done by monitoring the derivative of the pressure and activating the torque pulse if the derivative crossed a specified threshold. While the results were fairly successful (see Figure 31), in practice it would not be possible to tell the difference between a sudden spike in pressure error due to the gear teeth or one due to a sharp command change. It is possible that additional logic could be implemented to help decide when to turn on the pulse control, but this was not implemented in this work. Also, since the test valve would be closed in zero flow situations, it is somewhat unlikely that the spikes would have much influence on the interactive hydraulic environment. So, while a key advantage of using hydraulic motors is zero flow pressure control, this advantage is somewhat limited by the pulsing pressure from the gear teeth at low flows, which worsens as pressures increase.

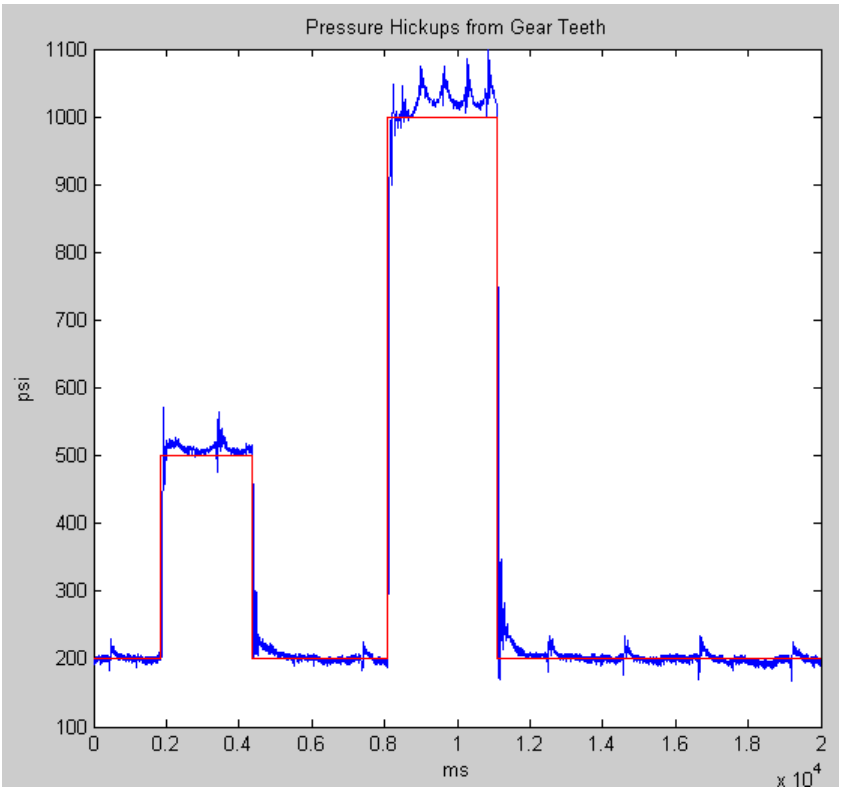


**Figure 29:** Zero flow pressure spikes from non-linear gear teeth pressure profile.

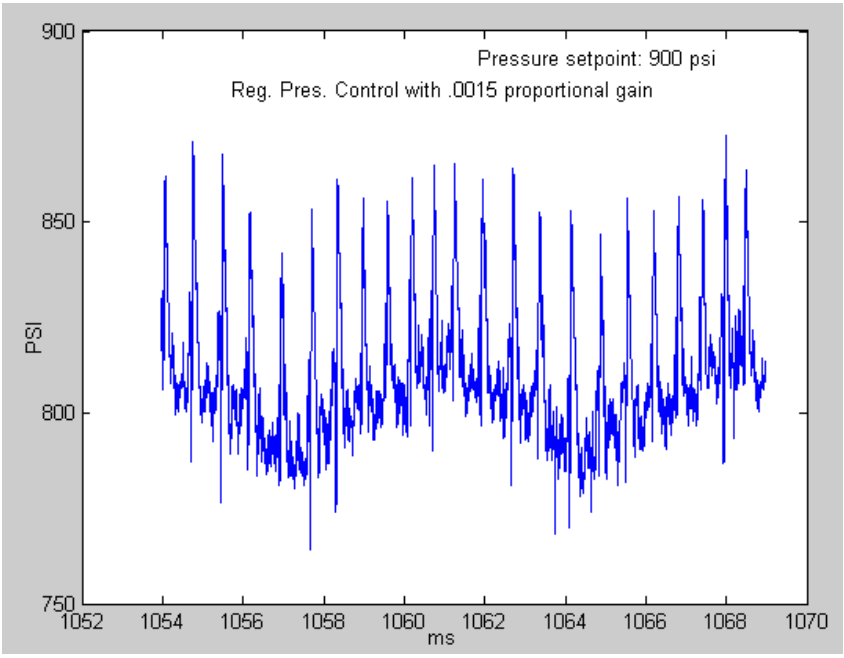


**Figure 30:** Zero flow pressure variation.

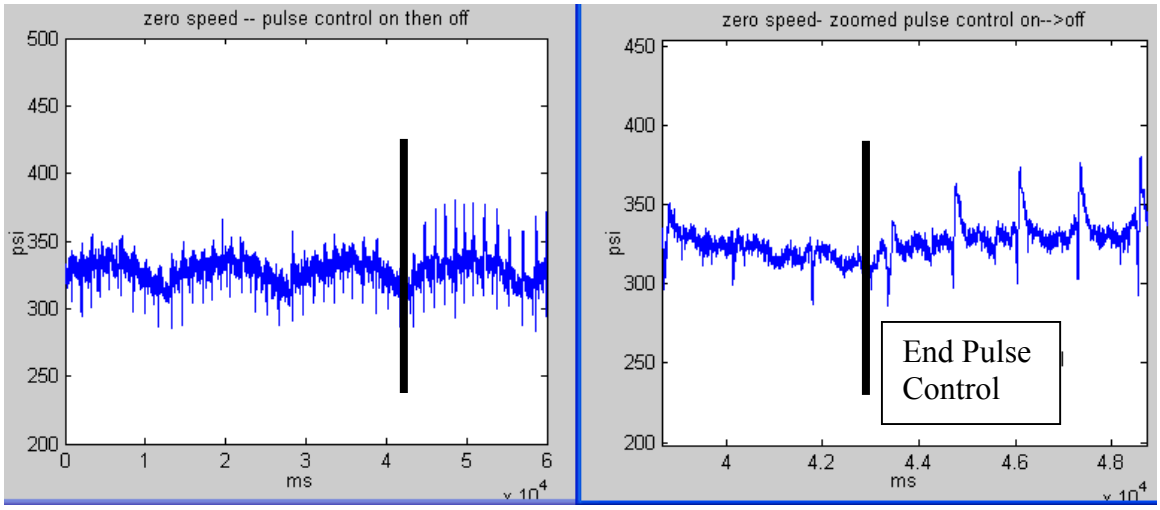


**Figure 31:** Zero flow pressure control with pulse compensation, on then off.

## Speed Controller

In comparison to the pressure controller, the speed controller is relatively simple. The final controller merely consists of a PID feedback controller with pressure based compensation for leakage. The compensator should probably include compensation for temperature, but changes in motor leakage due to temperature variations were found to be relatively small, so this was left out in the current design. The PID control and its integration with the surrounding system are shown in Figure 32.

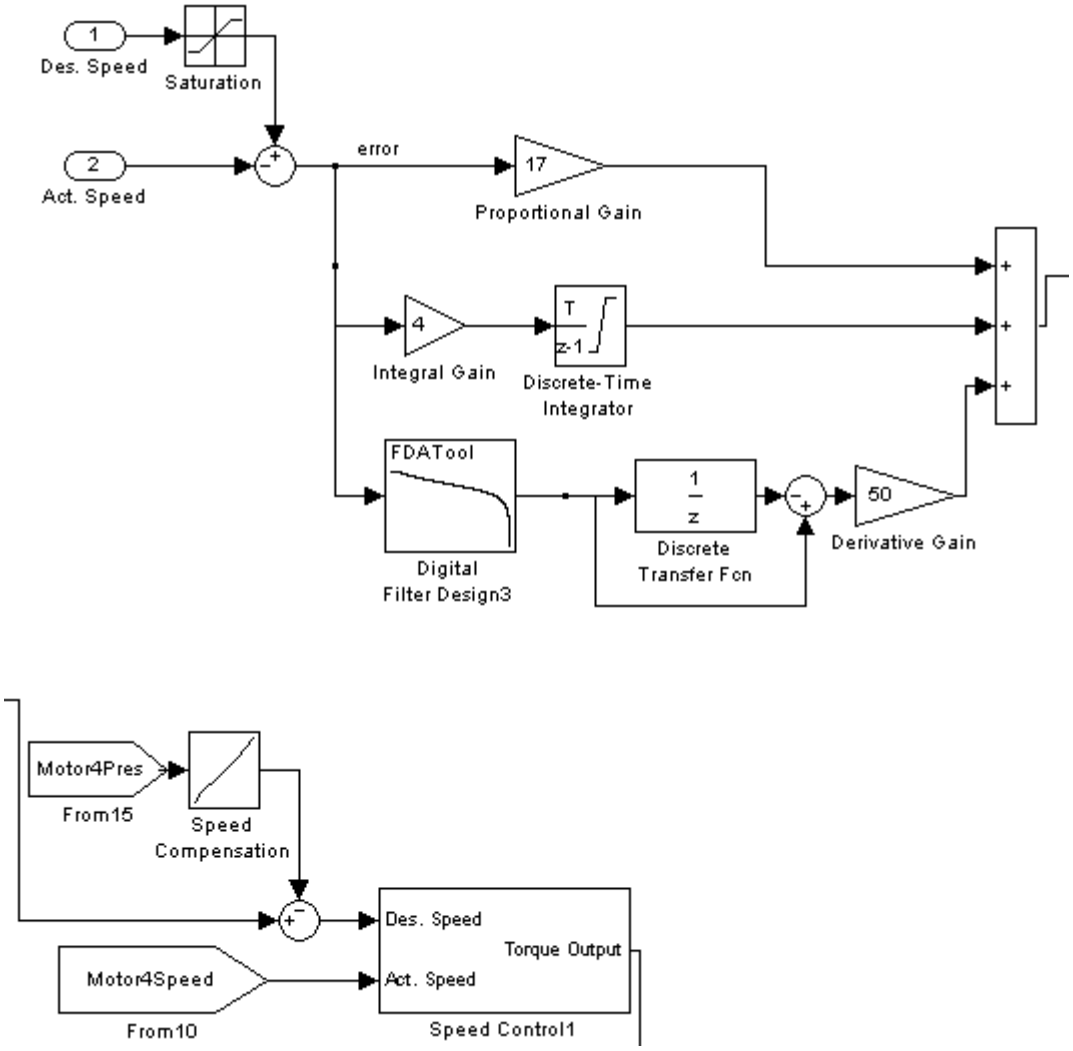


**Figure 32: Speed** control block diagram (top) and system integration (bottom).

Leakage flow was found to be mostly linear with pressure, and was determined by recording the average speed necessary to maintain a certain pressure at zero flow. This data was also correlated with comparisons between motor speed, pressure, and flow measured from a standard turbine based flow meter in steady state. Figure 33 is actually a 3D plot with pressure in the hidden dimension, and it shows how little motor speed varies with actual flow over a 1500 psi pressure range. The thickness of the line (graph on right) is an indication of the variation in actual flow for a particular motor speed. Note

also that the speed compensations are very small (<.005) when compared to the max speed (1.0).

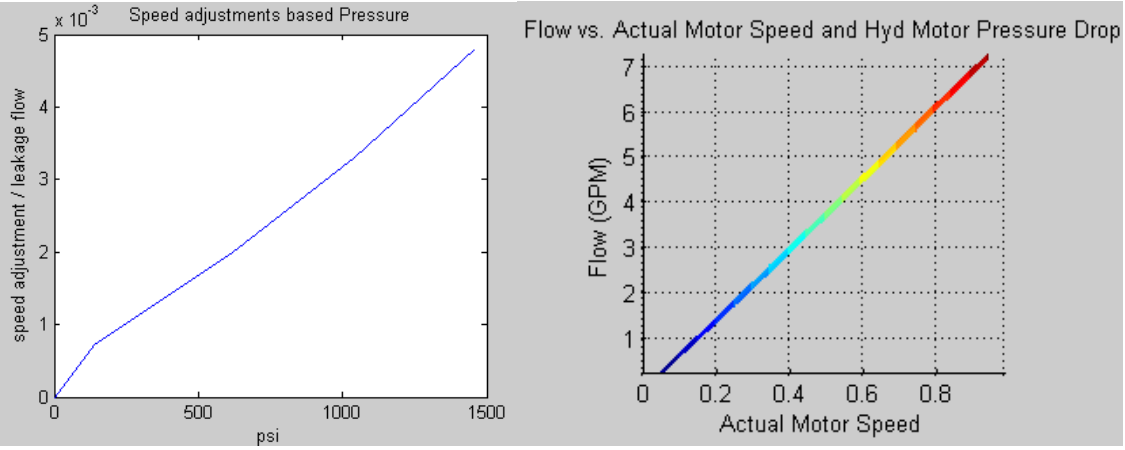


**Figure 33:** Speed adjustment to compensate for leakage at various pressures (left). Actual flow vs. Motor speed for various pressures, where pressure is the hidden dimension (right).

The PID controller was tuned by experimentally adjusting gains until a good combination of bandwidth, step response speed, low overshoot, and performance at high and low pressures was achieved. A first order filter with cutoff frequency of 50 Hz gave a good balance between noise rejection and signal delay. The speed controller had the advantage over the pressure controller that speed information from the motors was transferred completely digitally, meaning that the only noise came from the sampled zero-order-hold effect. This made it possible to increase derivative gain substantially higher than the pressure control case, which in turn allowed for higher proportional gain. Plots of the frequency response and step responses follow.

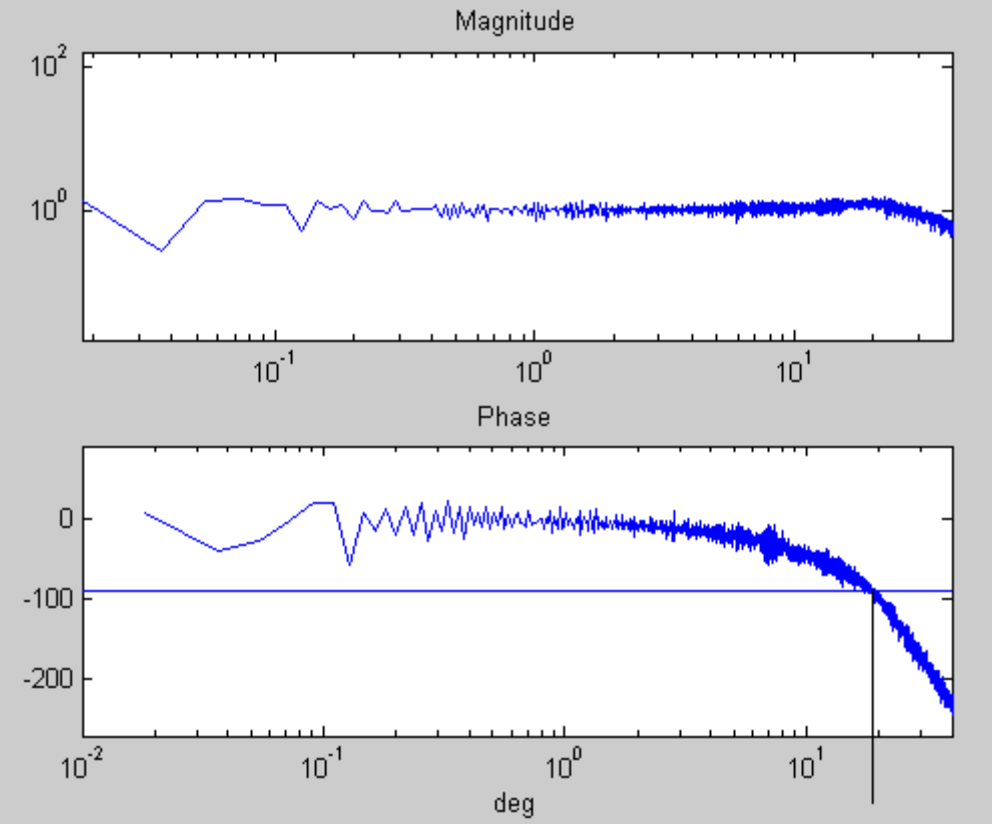


**Figure 34:** Speed control frequency response for .03 magnitude oscillation about .1 normalized speed.

A bandwidth (-90 deg. def.) of about 11 Hz is shown when the input frequency was given at a .03 magnitude oscillation about .1 normalized speed (3000 rpms = 1). Again, it is important to note that this bandwidth represents a maximum response frequency for the system. If the center point of speed or the magnitude of oscillation were increased very much, the torque command being sent to the motors would saturate. The plot below (Figure 35) shows the control effort over the course of the chirp signal used to generate the above frequency response. Notice that at bandwidth (-90 deg phase) frequency, the controller is already using about 60% of the torque available to the motors. In fact, in actual use, much of the time the controller is commanding the motor at its full

torque in one direction or the other. The figure (Figure 36) below showing step response illustrates the timing as well as control effort.

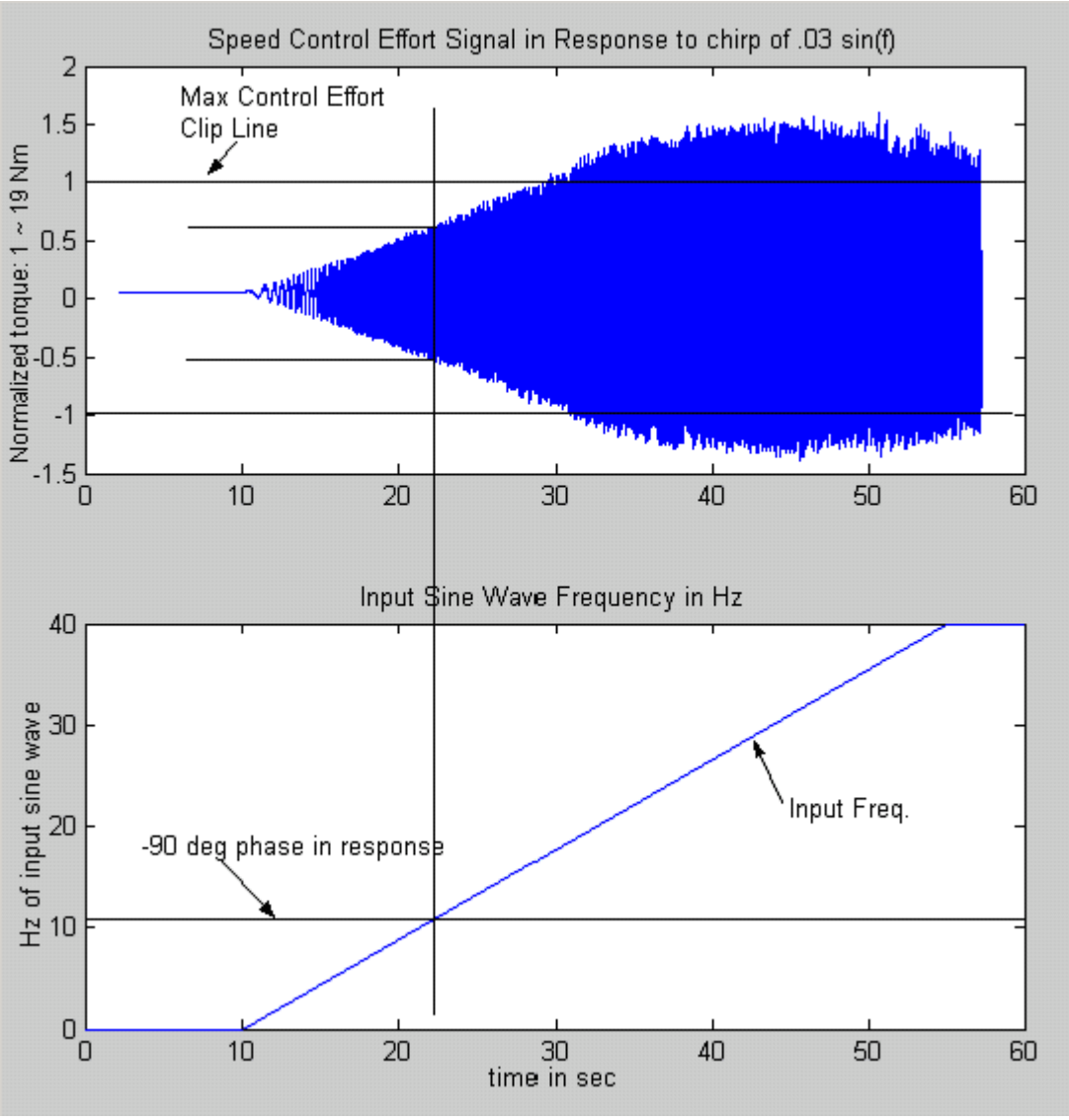


**Figure 35:** Control effort as a function of frequency during chirp.

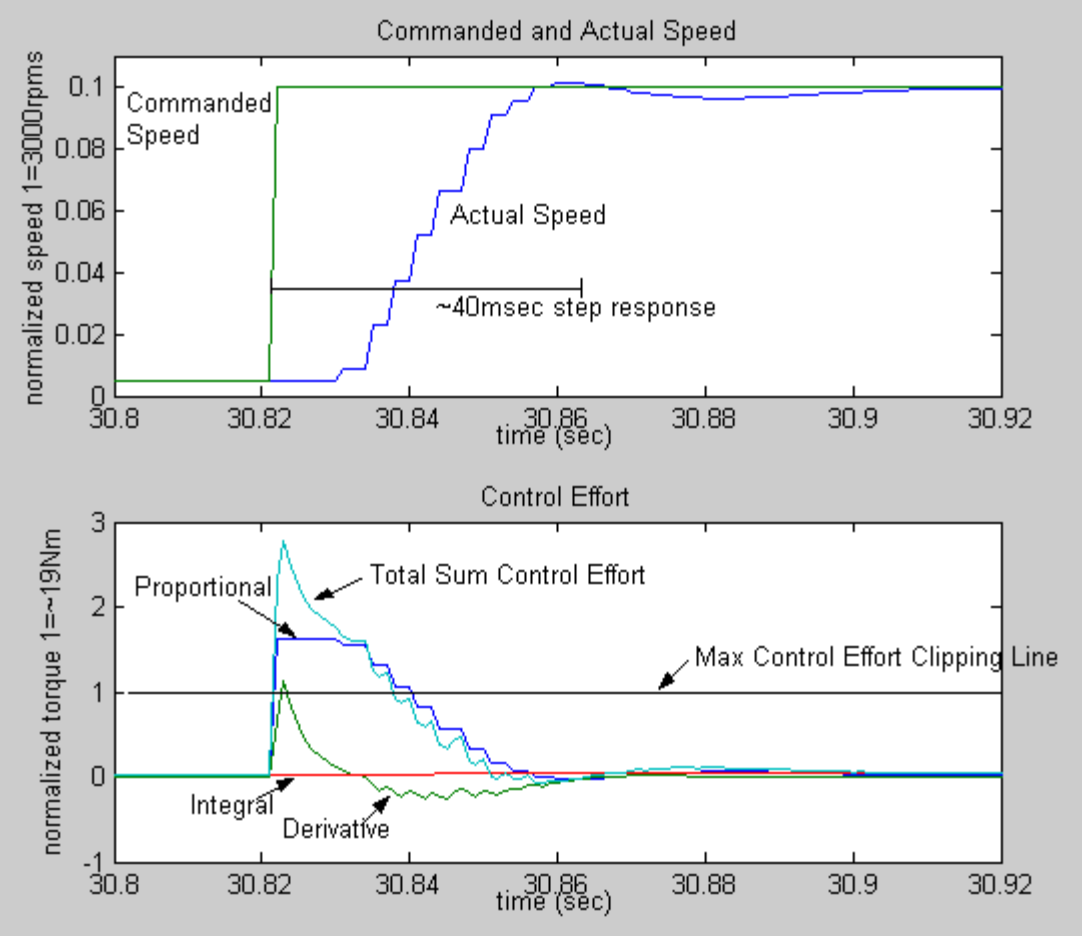


**Figure 36:** Speed control step response with controller effort.

The step response shows an approximate 40 ms settling time, much faster than the 100-300 ms time for pressure controlled responses. At this level of speed, the delay in the motor communication is a significant factor. The plot below (Figure 27) shows an enlarged view from beginning of the step response which highlights the motor delay. The simulation has a time step of 1 ms, and the motor amplifiers operate at a 4 ms clock cycle, which explains the 9ms delay before a change in speed is detected after a full torque command is issued. It takes 4ms from when the amplifiers receive a command to when they apply torque, and also another 4ms for when a speed is measured to when it is

sent back to the simulation computer. This also shows that delays (other than timing) between the motor being issued a command, and the amplifier actually generating current are negligible.

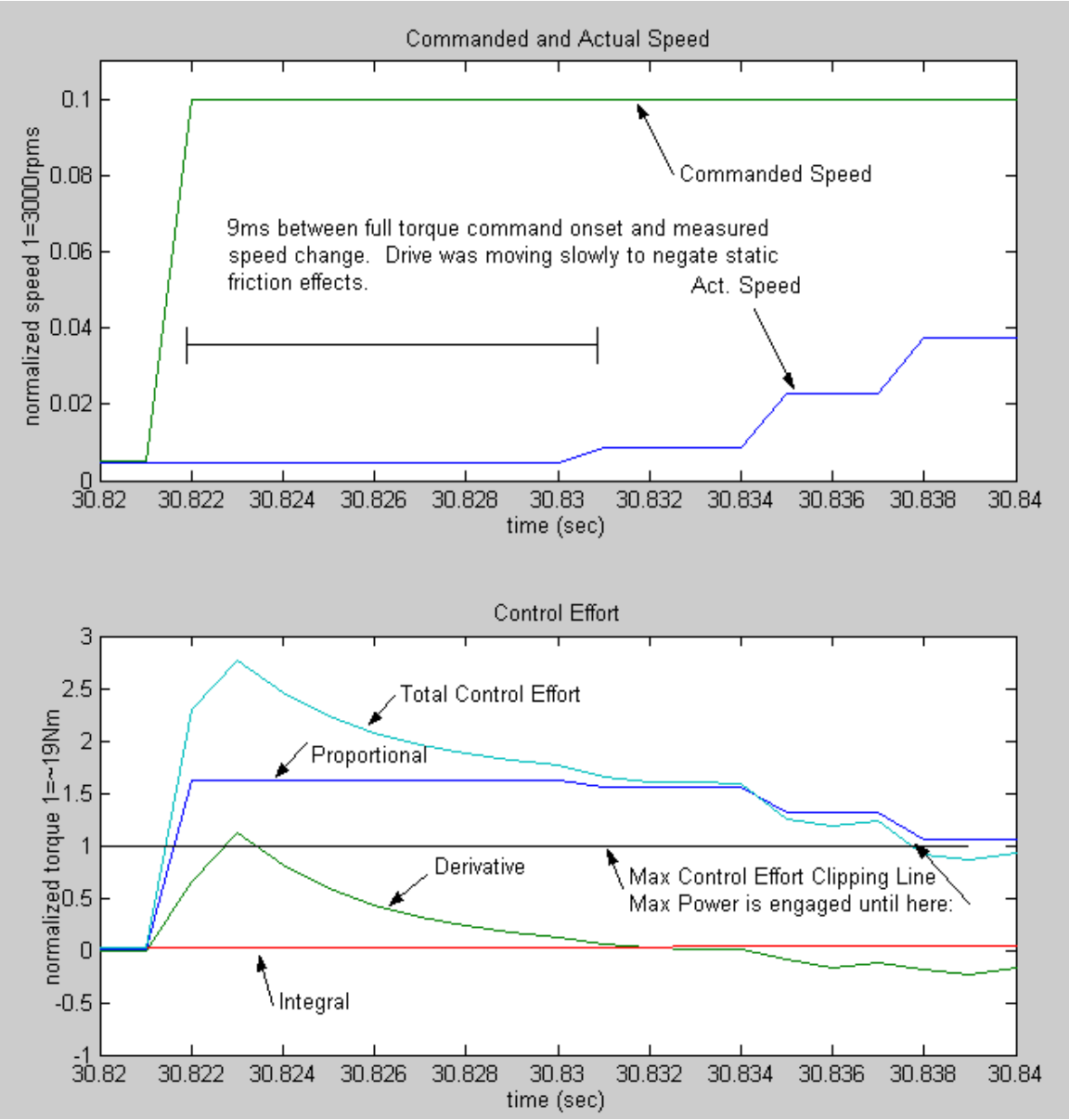


**Figure 37:** Speed control step response magnification showing communication and motor delay.

## Flow Control

The third and final controller developed for the HIL simulation was a flow regulation controller. The Sauer Danfoss valve is an open-centered valve, which means it uses constant flow as an input and then creates pressure by restricting this flow. To have more flexibility, and also because a constant displacement pump and motor was unavailable, flow from a constant pressure supply was regulated using a high speed Moog servo valve to simulate constant flow. The main components of the flow controller were a lookup table feedforward term and PI feedback. The lookup table was based on the Moog command necessary in steady state to effect a flow of 6 GPM given a certain pressure drop. Implementing the PI portion was more complicated due to the inability of a standard turbine based flow meter to respond faster than 50 ms. Proportional signals from the standard flow meter were found to be too slow to be of use, and actually contributed to instability in the system.

To solve this problem, an orifice based flow meter was developed that used pressure drop across the orifice to calculate instantaneous flow rates. It has been shown that the effects of inertia in the oil can be neglected up to 100Hz in this type of measurement, [11-13] meaning this method should be very much acceptable for the HIL system. Flow meters of this nature are usually problematic because an orifice must be small enough to give enough pressure change and resolution for small changes in flow, and also must be big enough that the surrounding system can tolerate the energy loss and heat generated by the orifice. These two competing factors make it difficult for one orifice to accommodate both large and slow flows. Luckily, flow only needed to be

supplied at a constant rate, making this system an ideal application. Unfortunately, noise in the pressure sensors required that somewhat large variations in pressure were needed to discriminate between actual flow changes and noise. This, combined with the fact that the goal of the controller was to limit flow changes, meant the orifice had to be designed fairly small to reveal small flow changes. On the other hand, since the HIL system had no active cooling, the orifice had to be big enough so that experiments could be run at semi-constant temperatures. The final orifice was sized experimentally to give about 150 – 300 psi pressure difference for typical flows, and allow the system to be run for about 10 minutes before increasing the oil temperature much higher than temperatures that occurred during test runs of the comparison John Deere backhoe (110 degrees F or 43 degrees C). Regrettably, the signal to noise ratio from the pressure transducers was too low to permit the signal to be differentiated. Any filter that could reduce noise to an acceptable level would have also added so much delay that the original turbine based flow meter would actually respond faster. The flow signal generated from the orifice flow meter was still noise free enough to provide an effective proportional feedback term, though.

In summary, the flow controller consists of a lookup table based on the pressure drop across the Moog valve, proportional feedback term based on flow from an orifice based flow meter, and integral feedback term based on flow from a standard turbine based flow meter. The hydraulic circuit is shown in Figure 38 and the Simulink control diagram is shown in Figure 39.

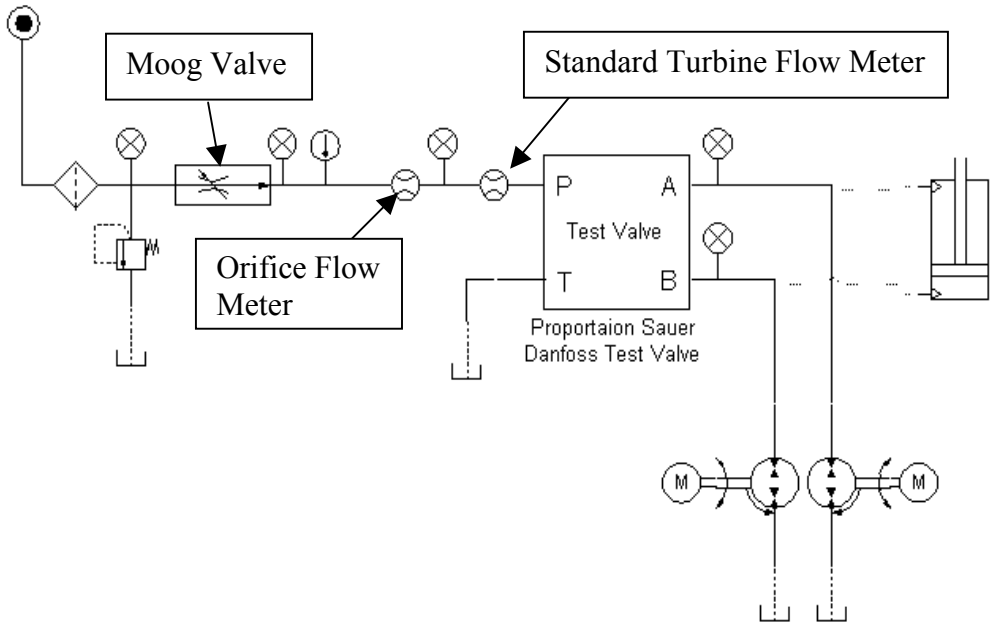


**Figure 38:** Hydraulic circuit for flow control.

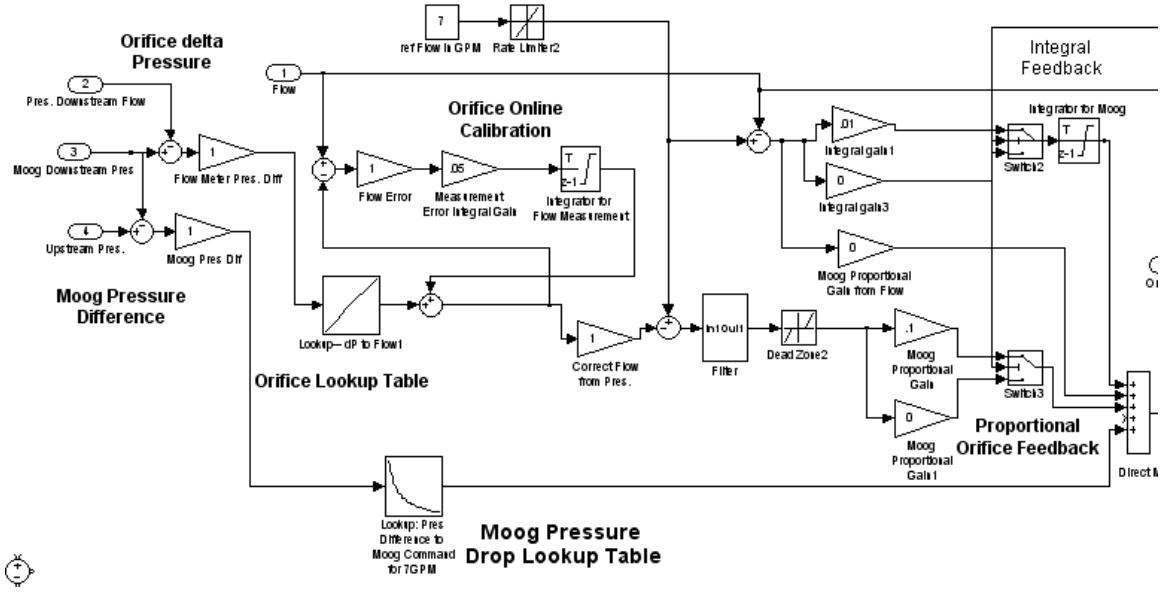


**Figure 39:** Flow control block diagram.

The integral feedback was meant mostly as a slow temperature compensator, so the gain was set very low and had little impact on transient changes in flow. The error between the orifice flow meter and traditional flow meter was also integrated and added to the predicted flow from the orifice lookup table, so that the orifice flow prediction would also adjust slowly to temperature changes over time.

Figure 31 shows the response of the flow meter to a pressure transient of about 600 psi. There is approximately a 50 ms delay between the peaks of the two signals, and according to the orifice flow, the real flow is almost completely restored to its regulated value by the time the turbine flow meter peaks, illustrating the importance and success of the orifice flow meter.

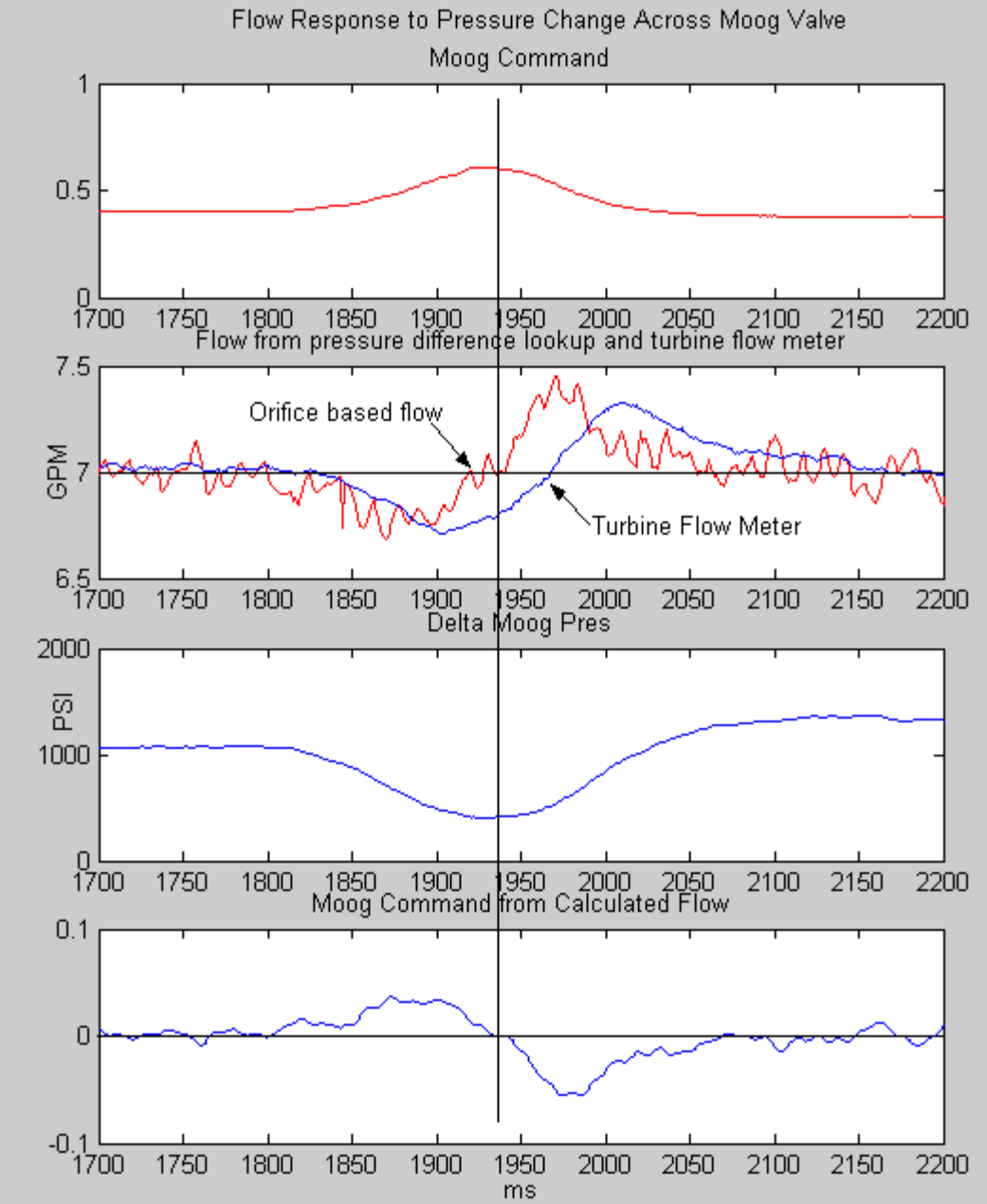


**Figure 40:** Flow control disturbance response.

## Flow Estimation

A final note will be made about estimating the actual flow being delivered through the motors. As plots above showed, there was very little leakage even at large pressure drops, but a compensator was designed nevertheless to account for this leakage and its dependence on temperature. The compensator consisted of a lookup table that interpolated between points on a hyper-surface relating pressure, motor speed, and temperature to actual flow. The hyper-surface was generated by taking samples over the entire workspace of flow, pressure and temperature, and then interpolating into a linear grid that could be installed in a lookup table.

# CHAPTER V

# RESULTS

## Introduction

Two models are implemented on the Hardware-in-the-Loop hydraulic simulator: an infinite cylinder with viscous friction and boom cylinder of a John Deere backhoe. Both flow-to-pressure (flow measured, pressure controlled) and pressure-to-flow versions of the cylinder model were implemented, with very limited results for the flow-to-pressure scheme (pressure control). For this reason, only a flow control model of the backhoe was used. In all cases results are shown for several different parameter configurations, illustrating the effectiveness, limitations, and flexibility of the electric motor based HIL system.

## Infinite Cylinder Simulation

Figure 42 shows the model for an infinitely long cylinder and its integration into the surrounding simulation system can be seen in Figure 43. This version implements a flow-to-pressure scheme, where motor speeds and their implied flow rates are used as input to the model, and pressure is the output.

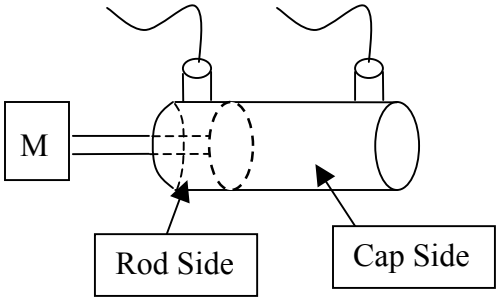


**Figure 41:** Single-rod cylinder schematic.

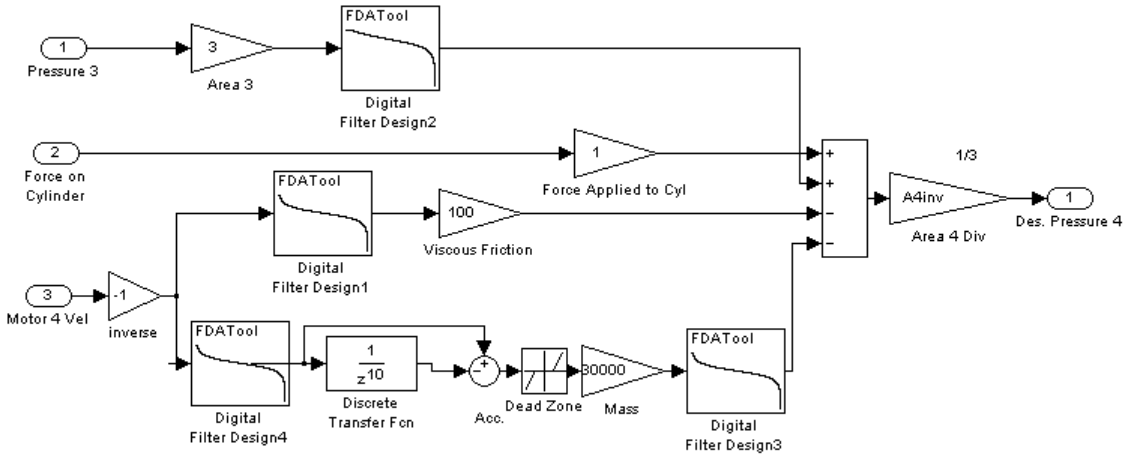


**Figure 42:** Infinite cylinder model for flow-to-pressure control scheme.

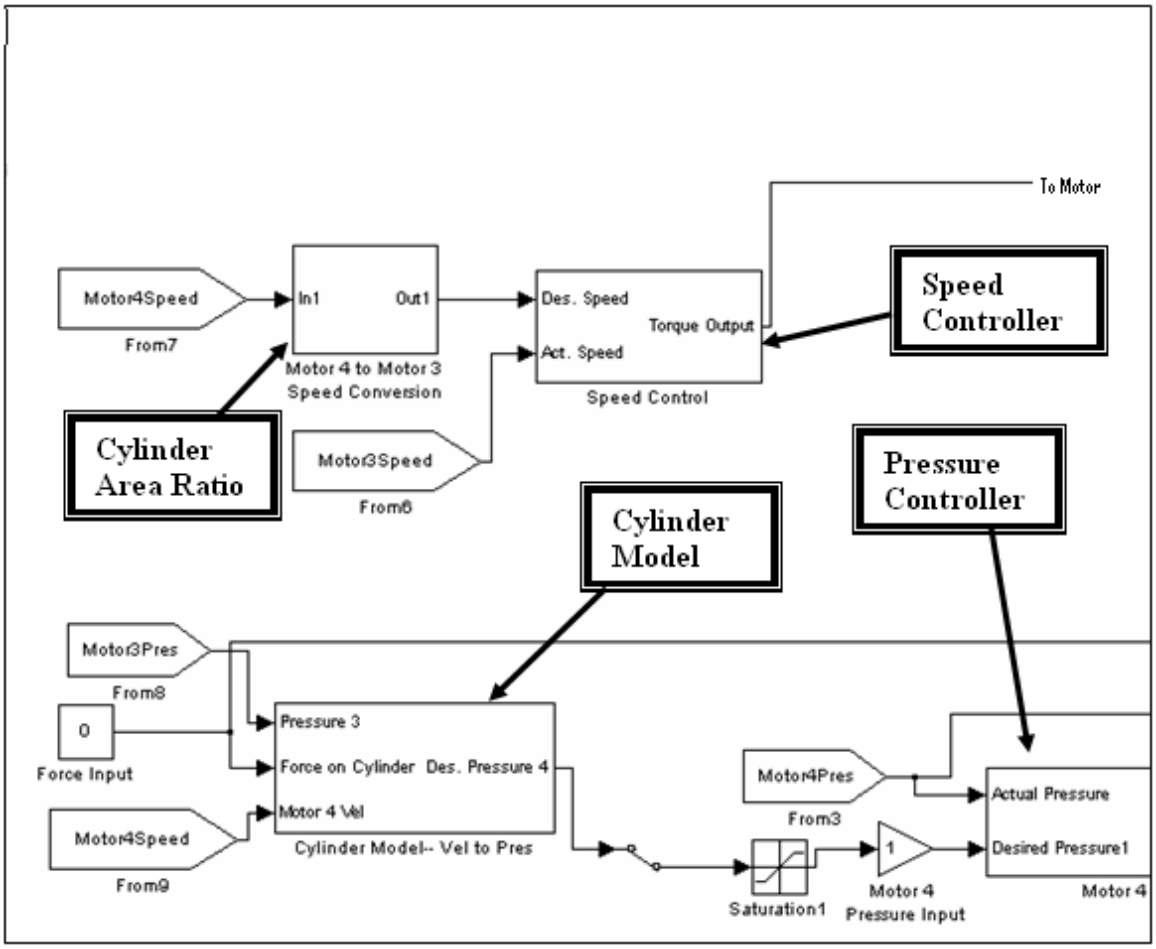


**Figure 43:** Cylinder model system integration for flow-to-pressure control.

The idea behind the flow-to-pressure control scheme is to measure the velocity, and then use that to calculate a desired pressure, which is then fed into the pressure controller. This is for only one side of the cylinder, however. The other side is placed under velocity control in order to maintain the continuity constraint:

$$\frac{Q_{rod\_in}}{Area_{rod}} = cylinder\_vel = \frac{Q_{cap\_out}}{Area_{cap}}$$

Later, the configuration where both motors are under speed control will be discussed. In both configurations, at least one motor is required to be under speed control, because the continuity constraint is fundamentally a flow constraint, and not a pressure one.
The equation for the cylinder, represented in block diagram form in Figure 42, follows.

$$M\ddot{x} = P_{rod}A_{rod} - P_{cap}A_{cap} - \dot{x}k_{fric}$$

One of the major drawbacks of using pressure control is that pressure signals end up being differentiated twice, once in the cylinder model, and then again in the pressure controller. To combat the inevitable amplification of noise that results, extensive filtering was used within the cylinder model to smooth signals before they were sent to the pressure controller. However, as will be seen in the following diagrams, the filters were not able to limit the noise to reasonable levels without adding a detrimental amount of delay. An additional effort to combat the noise was to subtract the current sample from one delayed 10 units instead of 1 in the differentiator. The idea was to look at a longer time frame so that signal changes occurring over a larger time could be differentiated from smaller short-term noise oscillations. This technique was helpful, but not effective enough as can be seen in the following graph. Figure 44 shows the response to the infinite cylinder model in the presence of a step up and down in flow driven by the Sauer Danfoss valve.

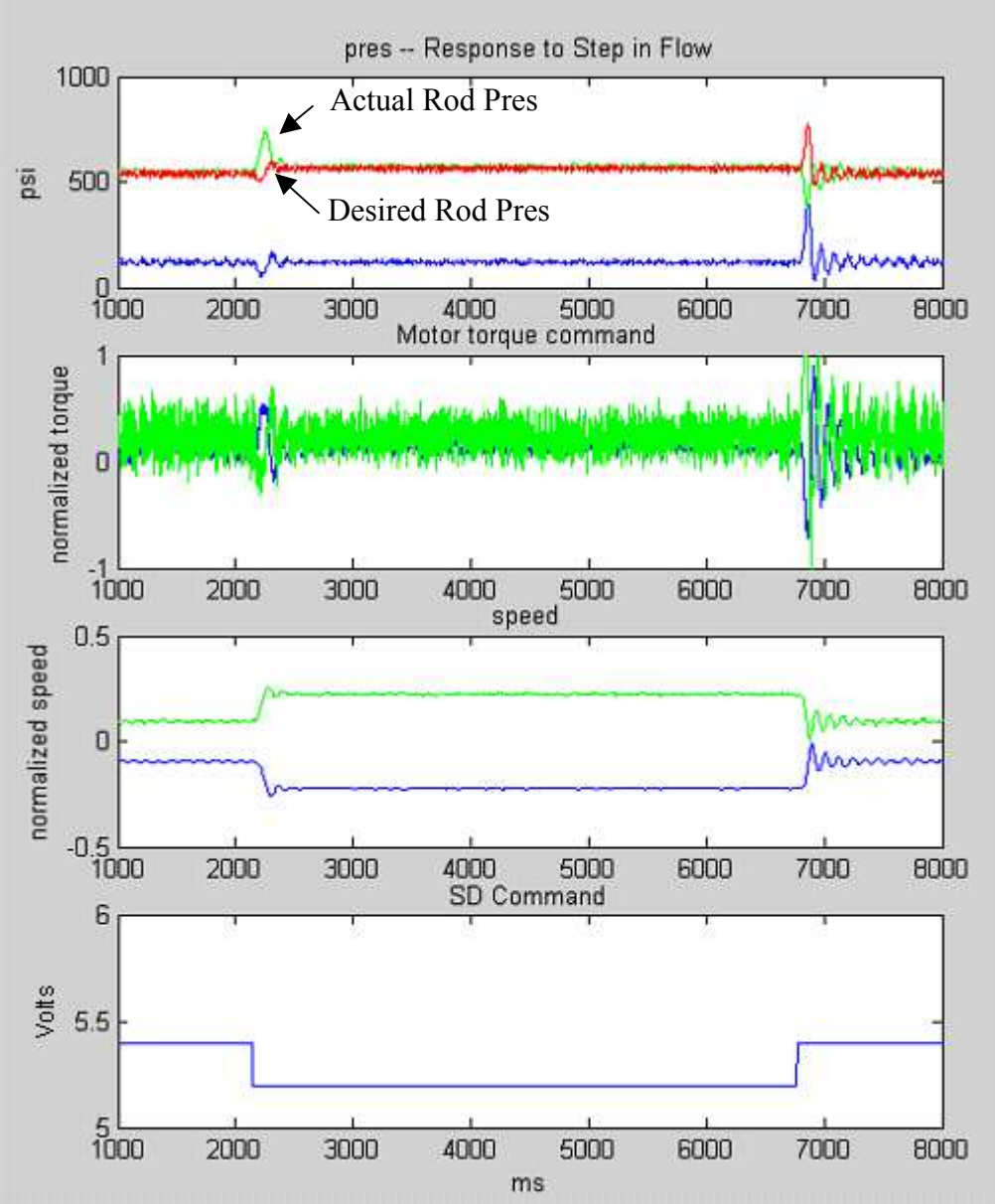


**Figure 44:** Cylinder response to speed step with flow-to-pressure model.

Parameters include 30,000 lbs for virtual mass, and a viscous gain of 600 lbs/(unit speed) for the friction parameter (The model is not meant to relate to physical units, but merely show qualitative behavior). Note that during the speed change in both directions, the controller is unable to follow the desired trajectory. Also, the second graph shows the

torque signal being sent to the motors. Even in steady conditions the noise in the commanded torque takes up nearly 25% of the available command. This was evident during the experimental run by excessive vibration and noise. The noise has the effect of causing the signal to reach its clipping values much faster, and also supplies the hydraulic system with medium frequency vibrations which on occasion find resonances within the system. One successful aspect of the simulation was the speed controller, which controlled the cap side of the cylinder to have an equal and opposite speed, shown as the bottom line in the third graph down in Figure 44. The cap motor speed is effectively controlled to be opposite of the rod. When the mass was increased to 70,000 virtual lbs, the noise problem was even more evident, as seen in Figure 45.

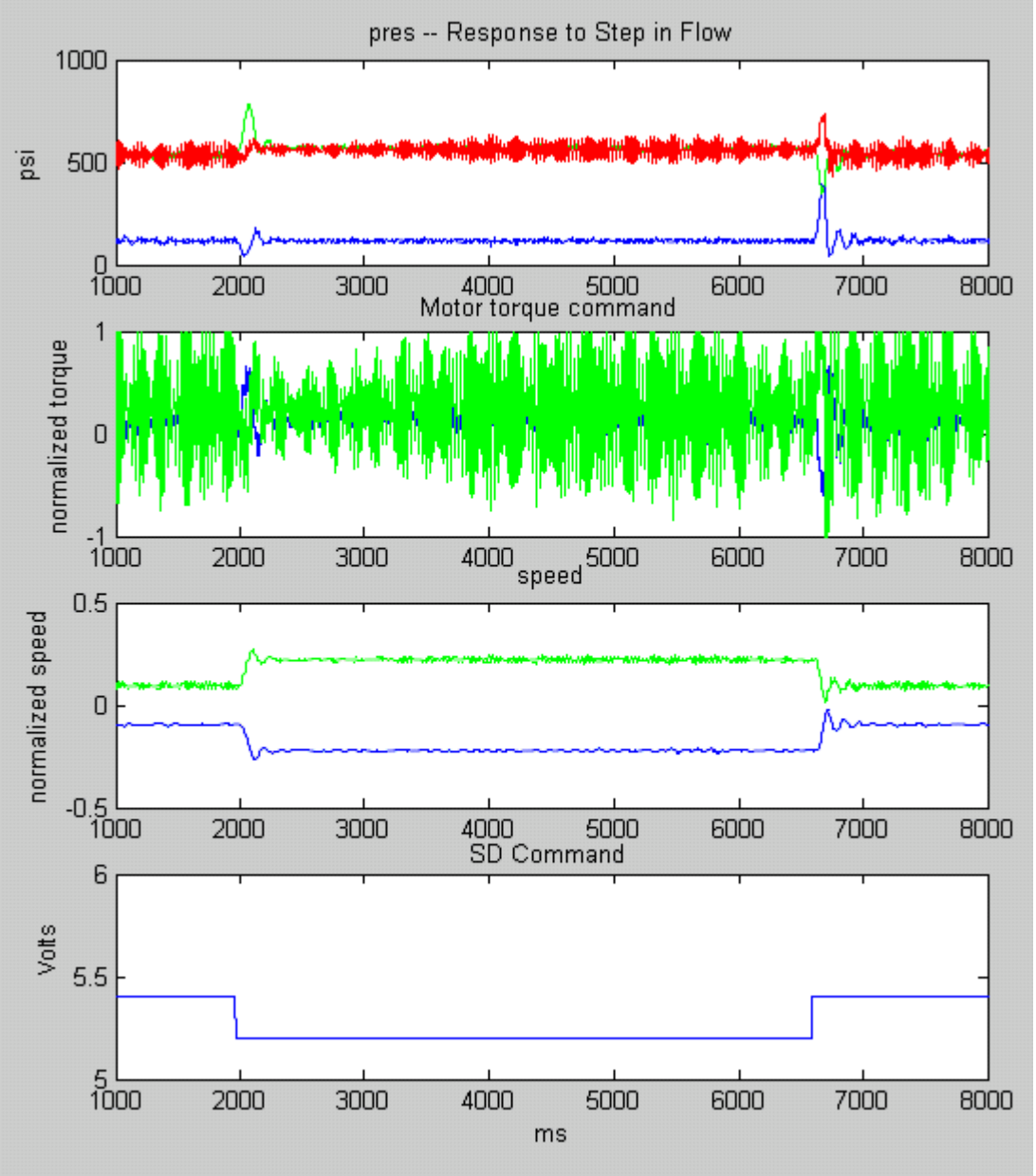


**Figure 45:** Cylinder response to speed step with higher virtual mass.

In this case, the noise consumed almost the entire available torque, once again introducing a medium frequency noise into the pressure signal, which only exacerbated the noise, since the pressure signal was being fed back and differentiated twice again. In addition to the problematic noise, the large increase in mass produced hardly any

difference in behavior. One would expect a much larger pressure spike as the Sauer Danfoss attempts to change the speed of the cylinder, and also a slower response in speed. Neither of these effects is readily visible. Despite extensive experimentation with filtering and adjustment of model parameters, the pressure control scheme was largely ineffective at emulating the environment of an infinite cylinder. Little if any expected behavior was apparent. Although successful pressure control was achieved when the input was a constant, step, or noise-free sine wave, a noisy command input seems to have overwhelmed the capabilities of the controller. Furthermore, noise in the pressure command led to physical oscillations in the actual pressure, which only fed back into the system and increased the problem. Filtering was unsuccessful since the frequency content of the noise was well within the range of signal frequencies.

## Speed Control Scheme

The speed controller was far more successful than the pressure controller, yielding results that fit well with how a 2$^{nd}$ order system would react. As a side note, we are unable to predict exactly what the actual behavior of such an infinite cylinder attached to a Sauer Danfoss valve would be, as no accurate model of the valve is available and no infinite cylinder *with simple viscous friction* is available to test in real-world conditions. However, the emulated behaviors compare qualitatively, if not quantitatively in some measures, to what one would expect from a 2$^{nd}$ order system.

Figure 46 and Figure 47 show the cylinder model and to its interconnection with the surrounding system, respectively.

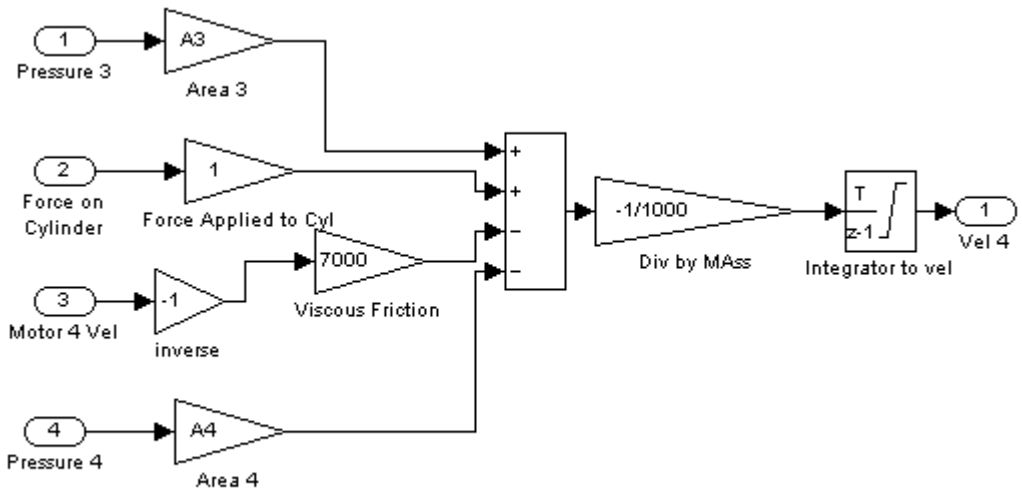


**Figure 46:** Cylinder model for pressure-to-speed control scheme.

Once again, the speed of the cap side of the cylinder is proportional to the rod size, where the proportionality constant is the ratio of the respective areas.

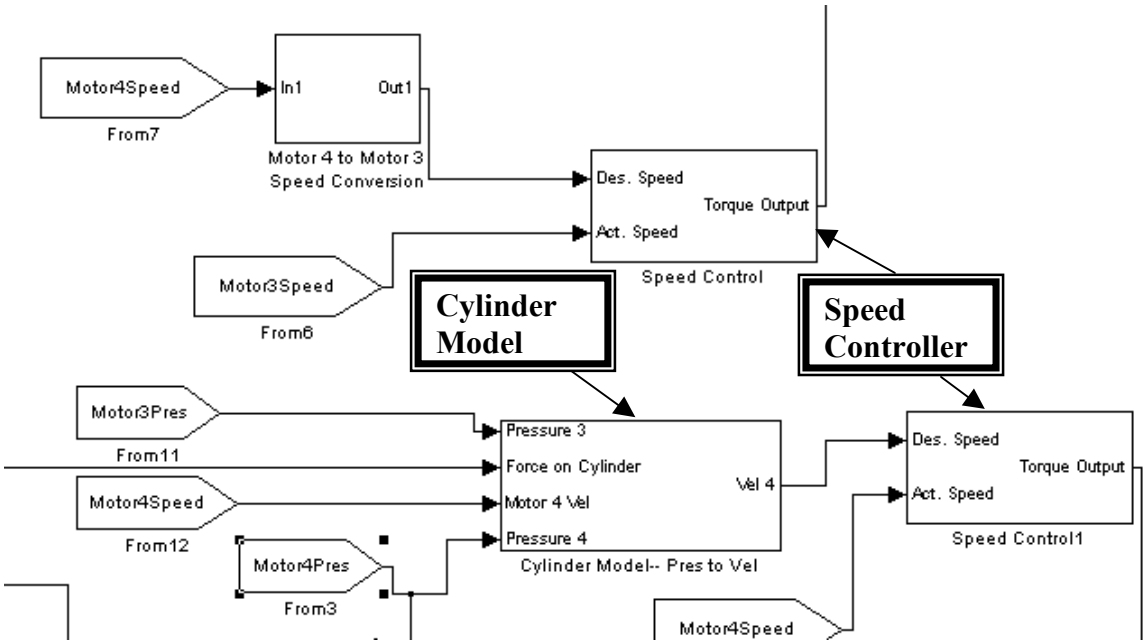


**Figure 47:** Cylinder model system integration for pressure-to-speed control.

First, the result of a model with the same parameters used in the pressure control case (viscous friction = 600 lbs/(normalized speed unit), mass = 30,000 lbs) is shown in Figure 48. The speed control case result is dramatically different from that of the pressure control. The torque, pressure, and speeds are smooth and relatively noise free. This is due to two reasons: the speed input is transferred through a completely digital path, and also the signal is being integrated instead of differentiated. The reason integration is needed instead of differentiation is because pressures are being measured, which yield a force balance in the model. The force balance is converted to acceleration by dividing by mass, and then integrated to find desired flow rates.

A small description of the Sauer Danfoss valve operation will be necessary to fairly judge the results of the simulation. It is known in the hydraulics world to be a "proportional flow valve," which means that it tries to supply flow proportional to an input command, independent of the load pressure. This, in turn, means that regardless of mass and friction parameters, or even the structure of the hydraulic environment emulation, the SD valve will work to produce the same flow barring pressure limitations. However, the transient pressure and speed, as well as the final speed, are completely determined by the emulation. To better understand the graphs, note that a command of 6 to the SD (Sauer Danfoss) valve is neutral, 3 is full flow into the rod side, and 9 is full flow into the cap side. So, when the SD command changes from 5.25 to 5.2, an increase in flow has been issued to the rod side of the cylinder. As would be expected, the pressure is shown to spike upwards as the SD valve tries to increase the speed of mass on the cylinder. A similar effect is shown when the SD valve tries to reduce the flow, and hence reduce the speed of the mass. The rod pressure dives down to zero psi as the mass

tries to continue onward at the same speed even as the flow of oil is reduced. This is what is known as an over-running load.

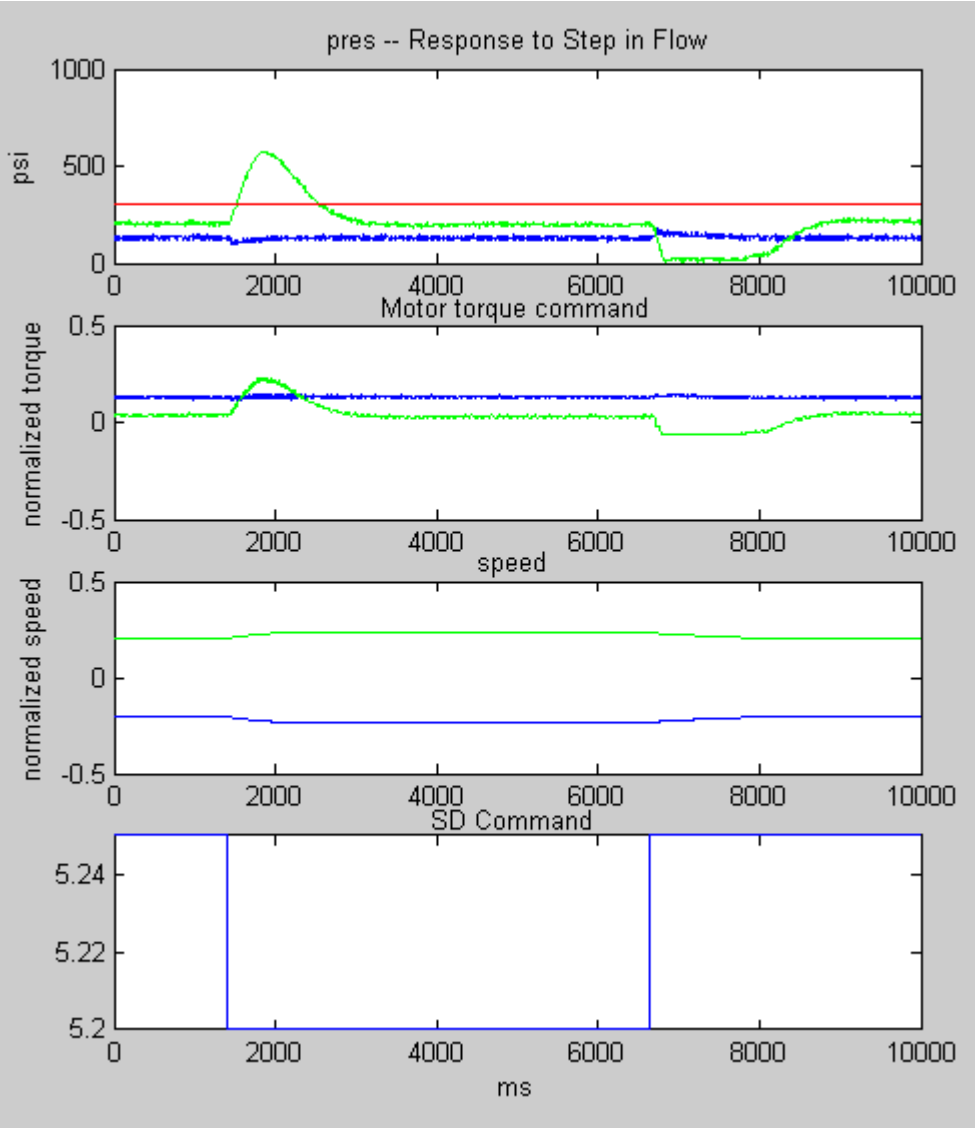


**Figure 48:** Cylinder response to speed change with pressure-to-flow model.

Two other graphs, Figure 49 and Figure 50 are shown to demonstrate that changing the mass and friction of the emulation model does indeed produce expected results. Figure 49 shows the response to the same SD commands as before, but with a

1000 lb virtual mass instead of 30,000 lbs. As expected, the effort needed to increase the speed of the mass is less, which leads to a smaller spike in pressure as the flow changes. Also, the natural frequency of the system has increased as can be seen by the larger number of peaks in pressure at the speed changes.

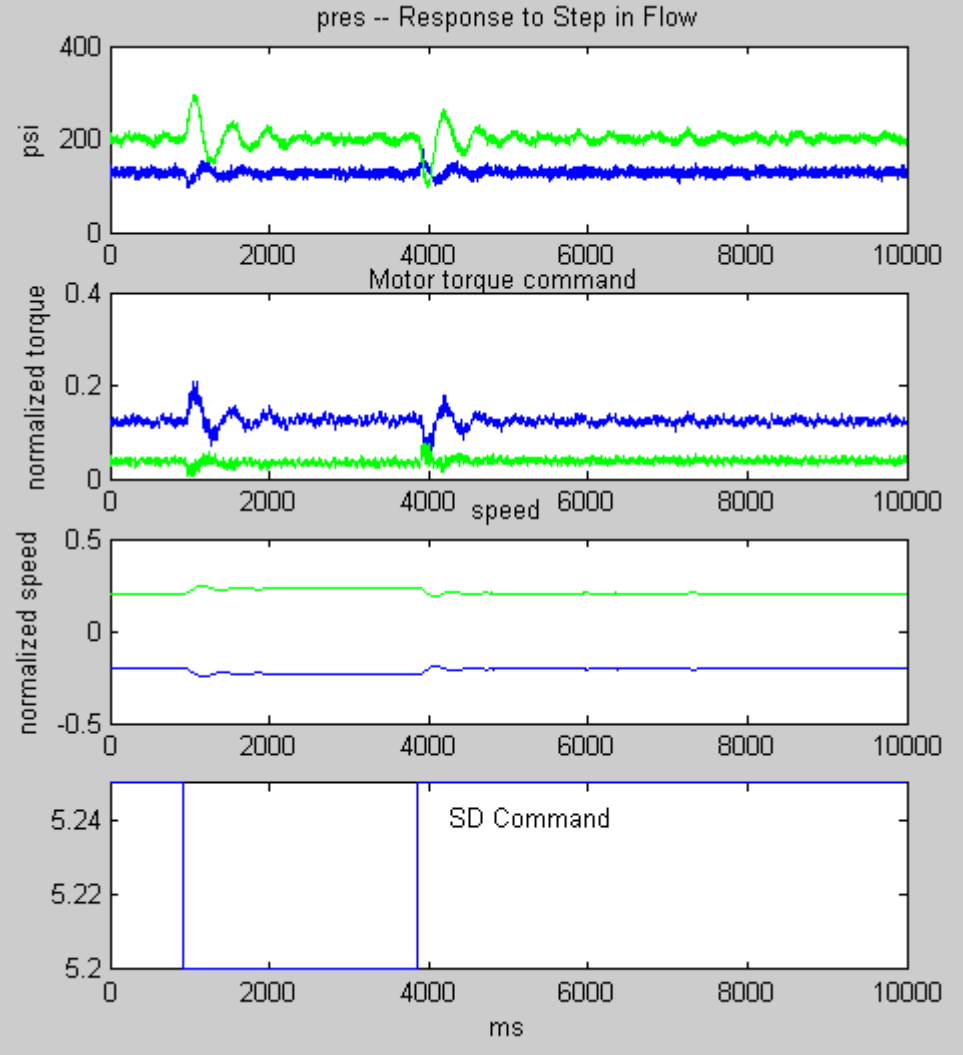


**Figure 49:** Cylinder response to speed change with pressure-to-flow model illustrating effect of smaller virtual mass.

The friction in this model is negligible in comparison with the pressure forces on the valve, so most of the dampening is coming from the SD valve itself. It has hydro-

mechanical feedback mechanisms that counter oscillations in flow to produce one that is proportional to input. An interesting capability of the computer model is that a negative friction can be introduced to overcome the natural damping effect of the valve. The following Figure 50 shows the results of a viscous friction gain equal to -2700 lbs / (normalized speed unit). Note the significantly longer decay time of the oscillations.

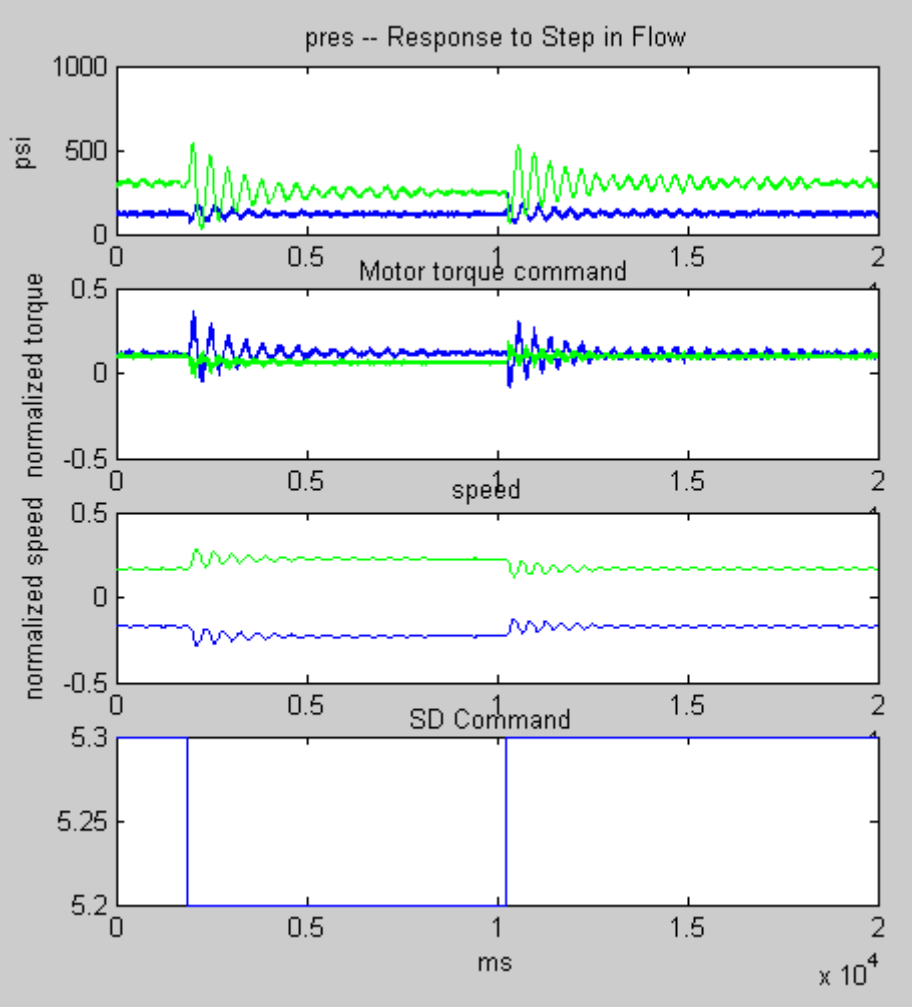


**Figure 50:** Cylinder response to speed change with pressure-to-flow model illustrating effect of negative virtual friction.

## Backhoe Cylinder Emulation

Given the dismal performance of the pressure controller scheme in the simple infinite cylinder simulation, results were only obtained for the speed controlled method. A brief description of the backhoe model will be given, followed by simulation results, and finally comparisons to data from an actual John Deere backhoe will be made. Figure 51 shows the integration of the backhoe model within the larger simulation system. An overview of the backhoe model is shown in Figure 52. Appreciation is given to Matt Kontz for developing the kinematics and dynamics of the boom arm. A schematic graphic of the boom and cylinder is shown in Figure 53.

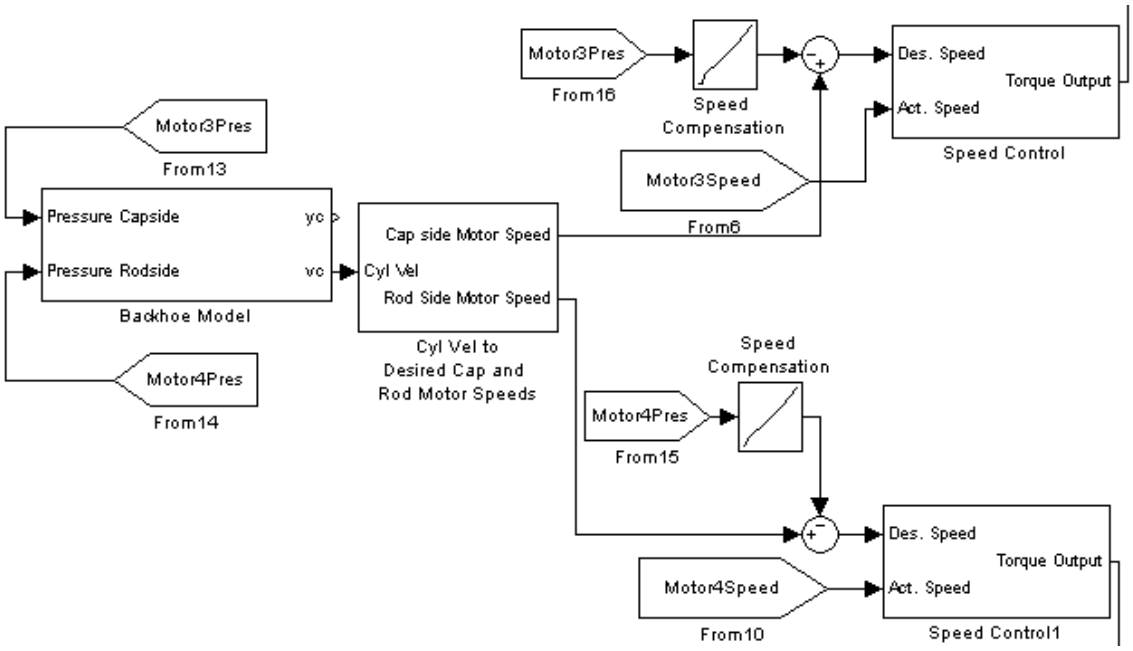


**Figure 51:** Backhoe model system integration.

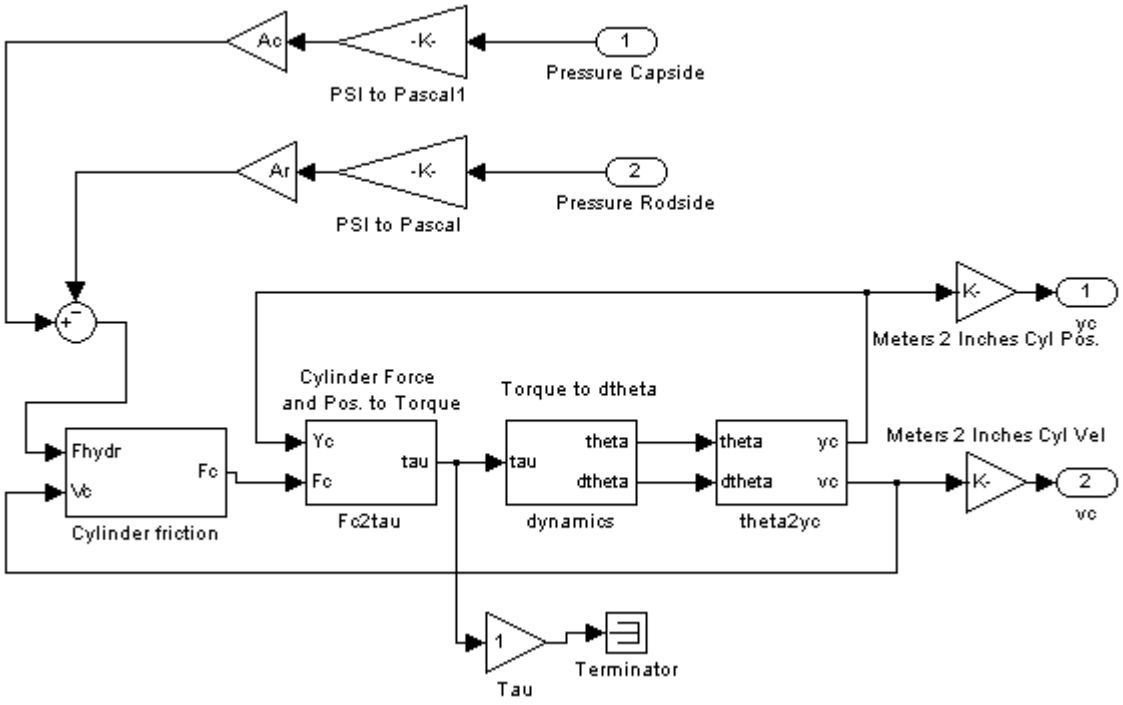


**Figure 52:** Backhoe model block diagram—credit Matt Kontz.

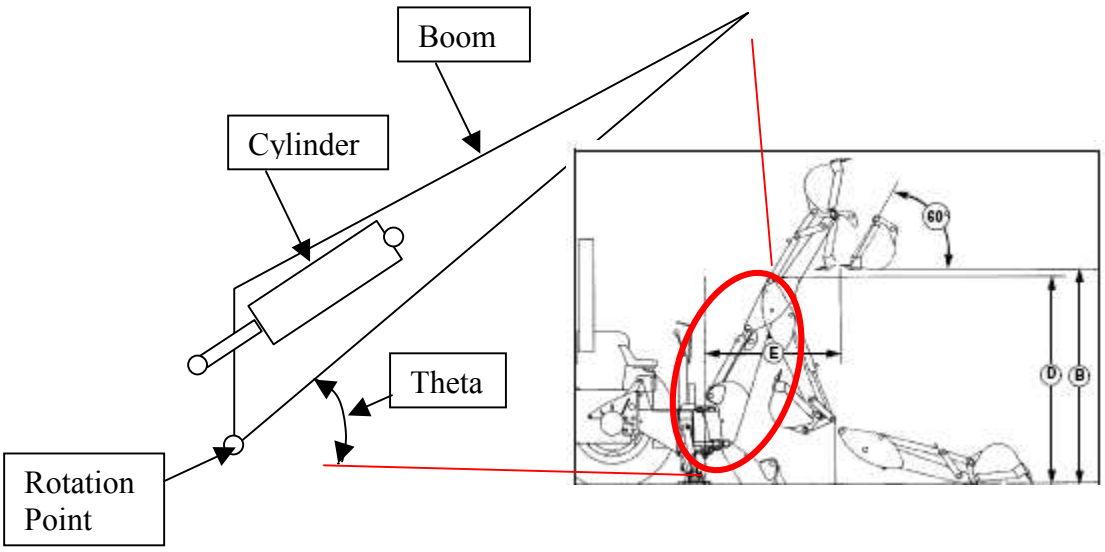


**Figure 53:** Backhoe boom and cylinder schematic.

The configuration in the simulation will have all links fully extended, so that maximum torque is necessary for lifting. The above Figure 53, shows only the main boom link of the backhoe, and its relative geometric relation to the cylinder that drives it. Since the rod side of the cylinder is pointing down, an increase in cylinder length corresponds to a decrease in theta, the angle of the boom. The basic flow of signals in the model starts with the hydraulic pressure difference on the cylinder, which is then combined with a frictional force, and fed into a block that calculates boom torque from geometry. This torque is then used with gravity and inertia considerations to calculate boom angle acceleration, velocity, and position. The angular velocity is finally used to calculate cylinder velocity, which is then multiplied by the respective rod or cap area, and fed to the motor speed controllers. The following set of figures demonstrates the performance of the emulation under various motions and parameter settings.

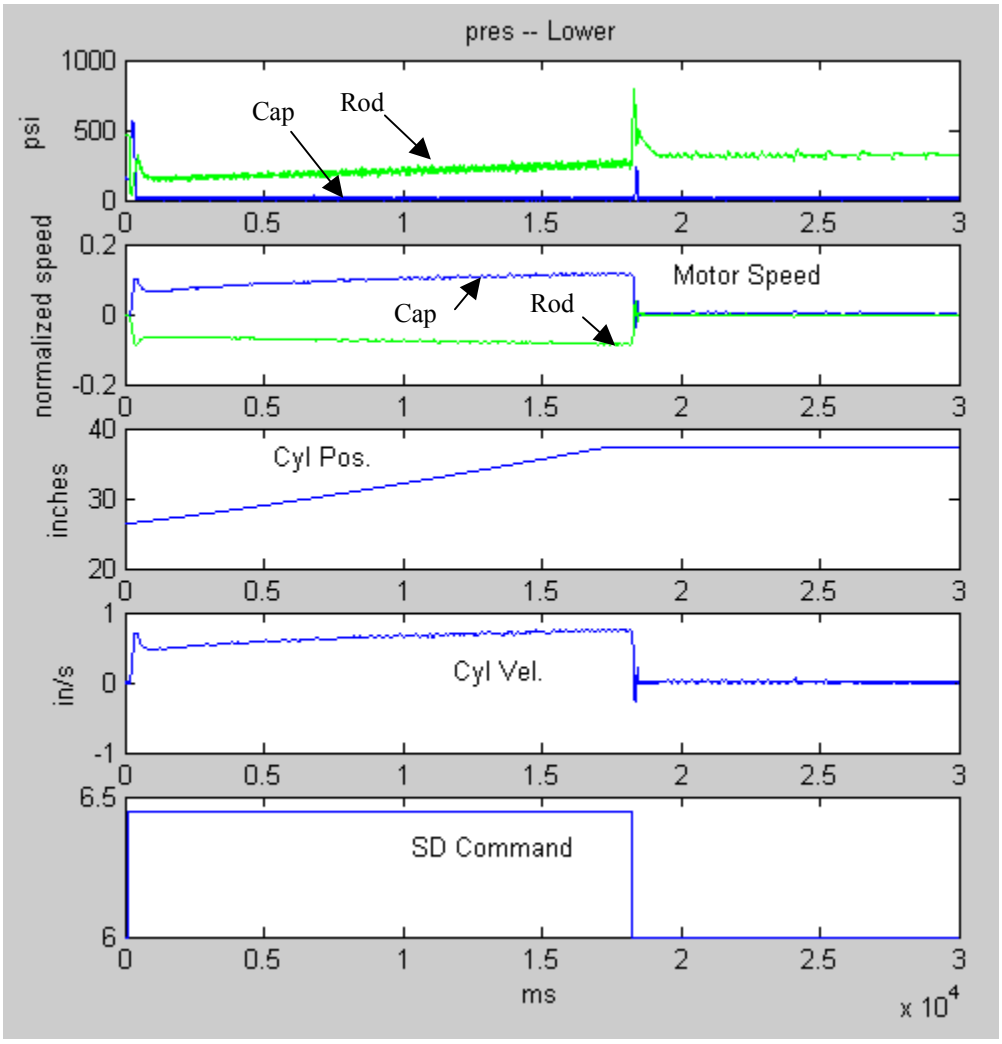


**Figure 54:** Backhoe emulation response to boom lowering command.

Figure 54 shows the simulated backhoe boom lowering. In this mode, flow is fed from the valve into the cap side of the cylinder, and out of the rod side back to the valve. Although the valve is forcing flow into the cap side, the pressure remains near zero since most of the lowering effort comes from gravity. The rod side remains pressurized as the valve is regulating the amount of the fluid that can leave that chamber. As expected, the pressure rises slowly as the boom is lowered, due to the increase in leverage gravity has

as the boom nears a horizontal position. The increase in torque from gravity is somewhat offset by the increased moment arm the cylinder has on the boom as it lowers. In the completely raised position, the line of force of the cylinder is almost parallel with the boom arm. But as the boom lowers, the line of force becomes increasingly perpendicular to the boom. Figure 55 shows this relationship.

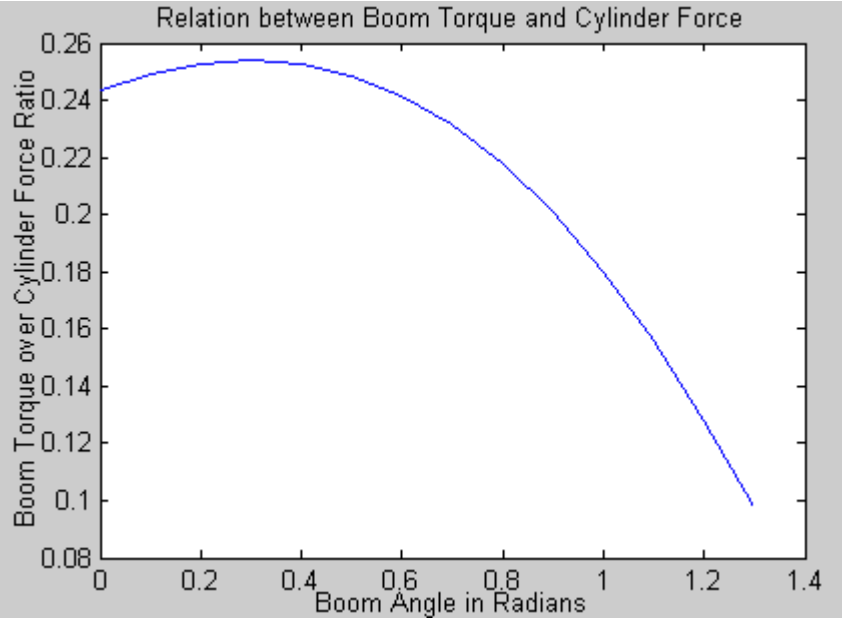


**Figure 55:** At 1.2 radians, when the boom is almost completely raised, boom torque is only 10% of cylinder force, whereas at .2 radians (completely lowered) boom torque is 20% of cylinder force.

Figure 56 shows the corresponding response from raising the boom. Many of the same features are present here that were present in lowering. Rod pressure slowly lowers as the boom is raised, as the effect of gravity is lessoned. Also note that pressure in the cap side is much higher during raising, despite the fact the valve is forcing flow into the

rod side. This is due to the unequal area ratio of the cylinder. For every unit of flow forced into the rod side, 1.37 units are forced out of the cap side. This flow must then go through the valve orifice to tank, which is why a pressure increase is visible.

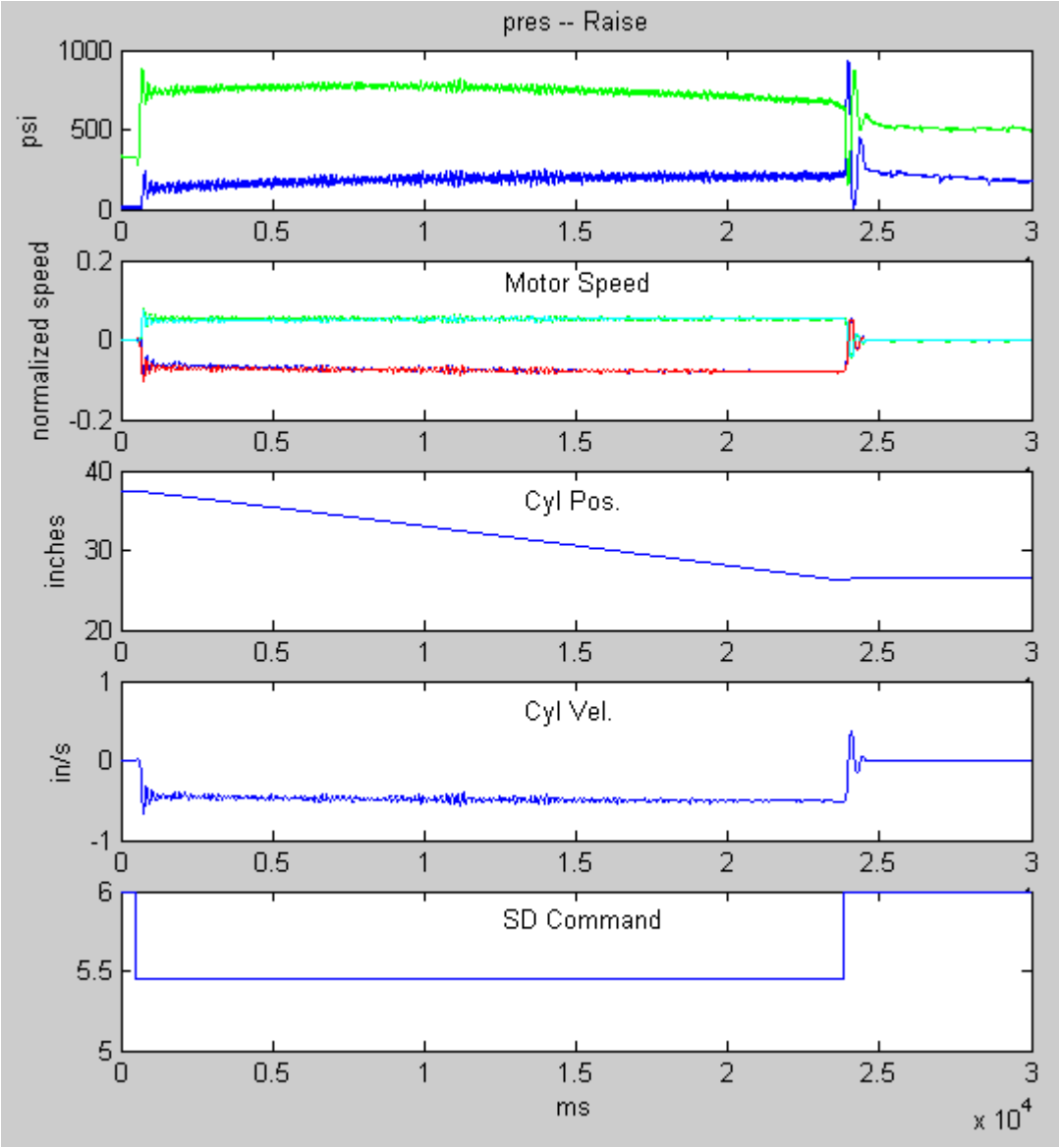


**Figure 56:** Backhoe emulation response to boom raising command.

The following figure shows actual pressure data from a real John Deere backhoe undergoing similar lowering and raising actions.

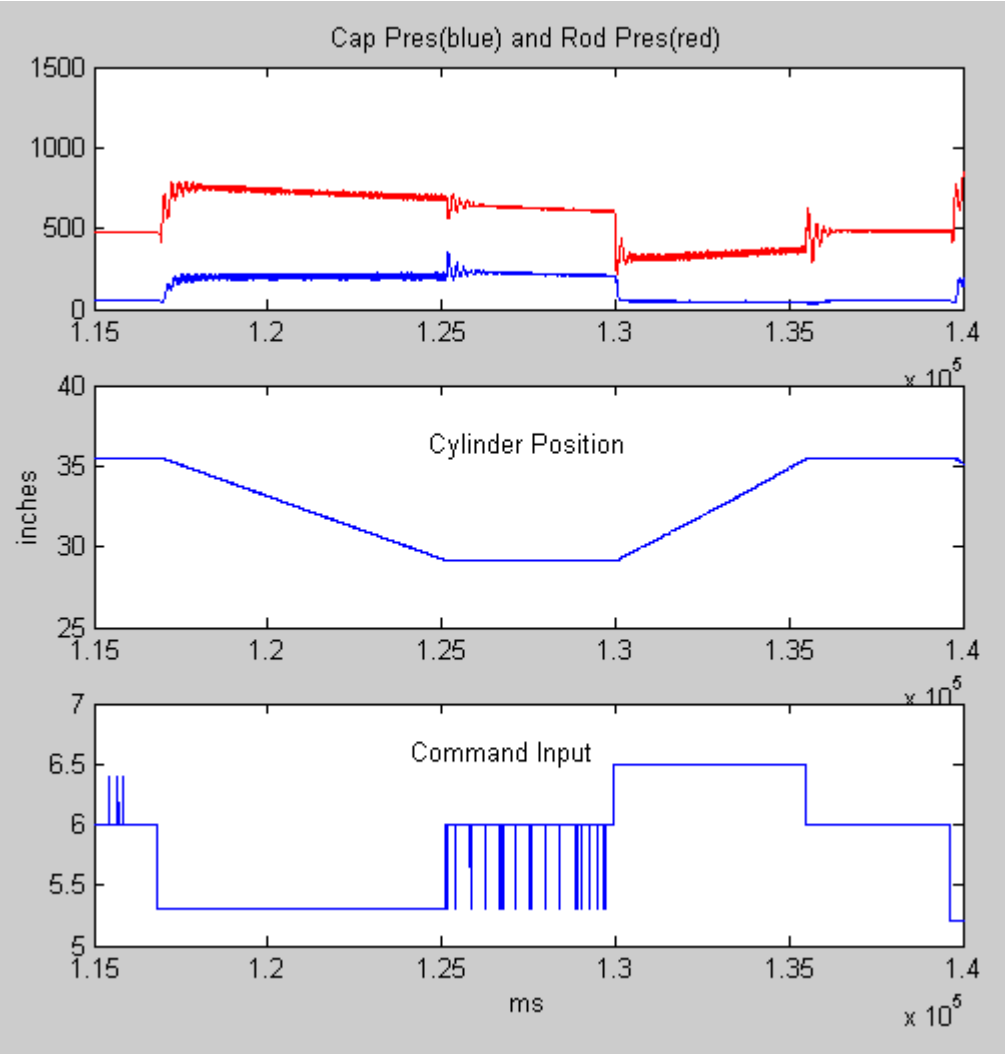


**Figure 57:** Real backhoe response while the boom is raising and then while the boom is lowering.

As the boom is raised, there is again a corresponding drop in pressure, and vice versa when the boom is lowered. Also, the cap pressure remains near zero during lowering and comes up to about the same pressure range during lifting as the emulation.

The corresponding rod pressure also compares well with the emulation pressure. It should be noted at this point that the Sauer Danfoss valve used in emulation, while the same type, is under no guarantee to have the same input-output relation as the one installed on the John Deere backhoe. For example, the edges of the deadband are different for the two valves, as flow does not begin until a command of 5.4 is issued to the backhoe, while 5.5 started flow in valve used in the emulation system. In order to evaluate whether a true match in emulation has been made, it would be necessary to complete testing on the exact same valve used on the backhoe, and this was not done in this research.

Unfortunately, the above plots represent the best case comparisons between the actual backhoe data and emulation results. Figure 58 shows the actual backhoe response to a sequence of lifting and lowering actions, each given a subsequently higher velocity command input.

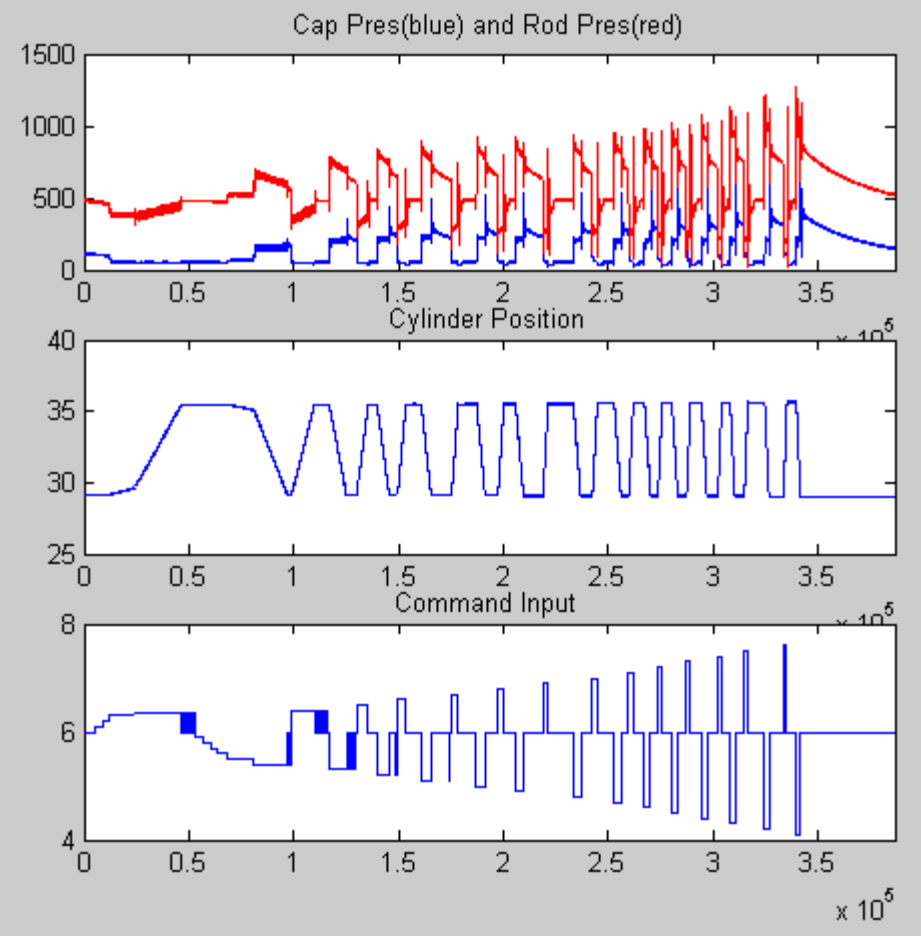


**Figure 58:** Real backhoe response while boom is being raised and lowered with increasing velocity. Note time is in ms, and the graph covers several minutes. This illustrates that pressures rise as higher velocities occur, but always stay below 1500 psi.

The best comparisons occurred at relatively small command inputs. Abnormally high pressures were observed in the emulation when larger input velocities were issued to the valve. For instance, in the real backhoe data, the cap side pressure never exceeds 500 psi, but as shown in Figure 59, the emulation encountered a significant spike at the end of a raising operation. The cap pressure peaks at 2900 psi as the boom is brought to a halt at the end of a lifting operation. Because of these irregularities, tests were not done for extremely high input commands to the valve. A magnification of the motor speed at the

time of the large pressure spike reveals a significant difference between the desired speed and actual resulting speed. In lifting action, the cap side motor is pumping fluid into the valve at a fairly high speed, so when the valve suddenly closes the orifice flow was being pumped into, the inertia of the motor, if uncontrolled, would cause a pressure spike. The desired speed command was much less than the actual, and this seems to indicate that the controller was unable to overcome the inertia of the motor and slow the speed down fast enough. This situation highlights the main limitation to using motors for hydraulic emulation. Not only must the controller be capable of controlling speed at high bandwidth, it also must be strong enough to counter the effects of the inertia in the electric motor, coupling, and hydraulic motor. Even though the electric motor and hydraulic motor pair is capable of producing 1500 psi, it does not have enough torque to handle the fast change in speeds required by the valve and model in this simulation. The design requirements of such a simulation system should take into account the maximum acceleration required by a certain application and model, and not just maximum pressures.

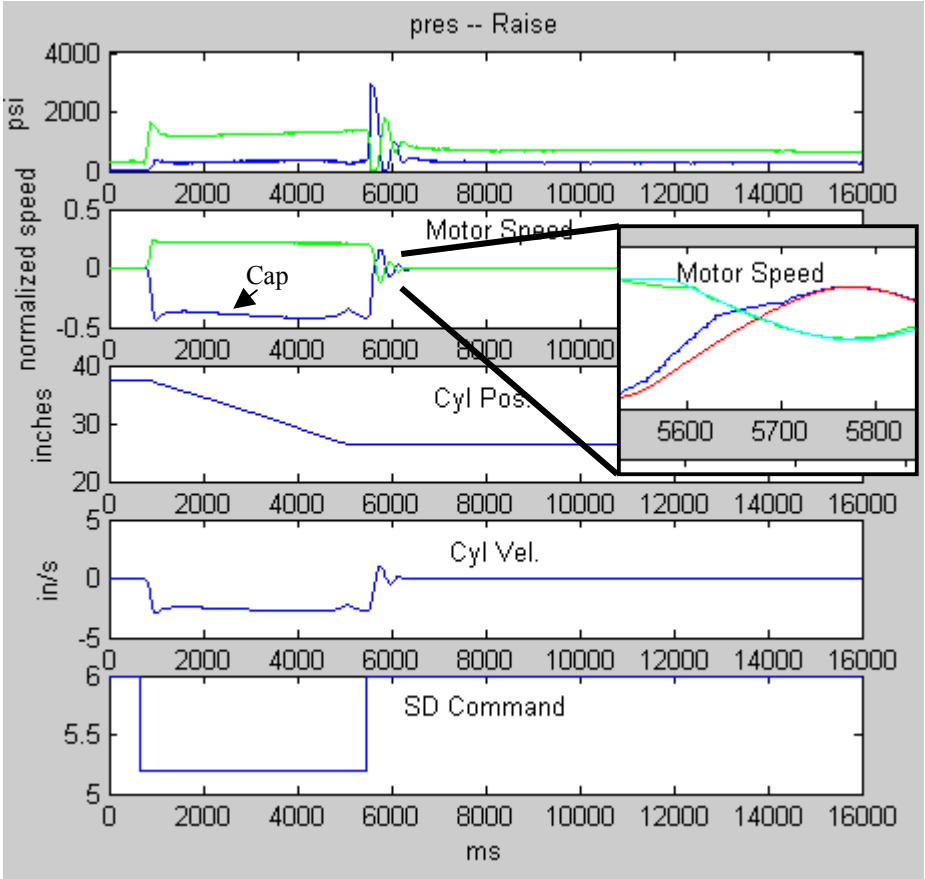


**Figure 59:** Cap-side pressure spike (in emulation) from stopping while raising the boom quickly. Also, the inability for the motor to track desired speed is shown.

## Cylinder Friction

One drawback to using a backhoe cylinder as a comparison base for the HIL simulation system is that an accurate model of cylinder friction is essential to achieving a comparable emulation. The results above were acquired using an inertia for the boom about twice as large as would be expected based on the geometry and mass distribution. This was required because unstable oscillations occurred at lower inertias unless so much viscous friction was added that pressures were twice those in the real comparison data. It

is felt that the excessive pressures mentioned above for higher command velocities were in part because unrealistically high viscous friction coefficients were used. Yet, due to the lack of an accurate friction model, those viscous coefficients were necessary to stabilize the system. It was not in the scope of this work to create and analyze an accurate friction model of a cylinder, but some comments will be made regarding some of the simpler models typically used. A simple coulomb viscous model, as noted in [29], does not take into account changes in friction due to pressure in the cylinder. Cylinder seal designs are able to prevent leakage at high pressures in part because the amount the seal presses against the rod is a function of chamber pressure. In [29], the authors developed a friction model as a function of pressure, and showed that it was mostly independent of velocity.

$$F_f = x_1 e^{x_2 v} + x_3(p_1 - p_2) + x_4 p_2 + x_5 v$$

The above equation represents the final form for their model, where x's are coefficients to be determined. The first term is meant to capture the effect of increased friction towards the ends of cylinders, where the chamber has been less polished. Their test setup consisted of a cylinder with no load driven by a standard servo valve. They mention that the last velocity dependent term has a negligible effect on friction. Their setup was a no-load situation, so the validity of the model in loaded situations is questionable. Regardless, based on their work, a pressure dependent term was added to the backhoe friction model, which seemed to improve results.

The final model used in the backhoe HIL emulation is shown in Figure 60. It incorporates a standard coulomb and viscous effect, as well as a term proportional to rod pressure. The end effect is that the coulomb portion of friction is increased as pressure increases.

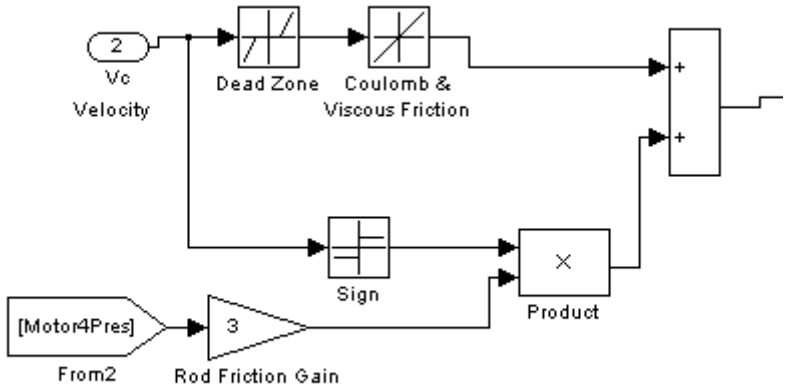


**Figure 60: Friction** Model showing viscous, coulomb, and pressure dependent terms.

## Model Flexibility

A final note will be made about the flexibility of the HIL simulation system. As mentioned above, higher than expected inertias were used (1000 kg/m^2) when the expected value was between 200 and 300 kg/m^2. This was done primarily to keep the system stable without necessitating extremely large amounts of viscous friction. However, changes in the inertia produced expected results in terms of natural frequency oscillations, although friction was increased substantially to obtain the following graphs. Figure 61 shows the transients recorded when the valve was issued a stop command

while raising the boom. As the inertia is reduced, the natural frequency of the resulting vibrations increases as expected. And in what seems to be one of the better correlations to real data for the entire emulation, as the inertia is lowered down closer to 250 kg/m^2 (which is the approximate value expected from mass distribution), the frequency of the oscillations is nearly equal to the oscillations that occur in the real backhoe data during the same operation, 4.7Hz. At 100 kg/m^2, the oscillation frequency in the emulation is about 5.8 Hz. Figure 62 shows a plot for the response at J=200 kg/m^2 with actual and desired motor speeds. As can be seen, the controller tracks the desired speeds well, which is expected since the frequency is well below the 12 Hz bandwidth and the motors are not at maximum torque.

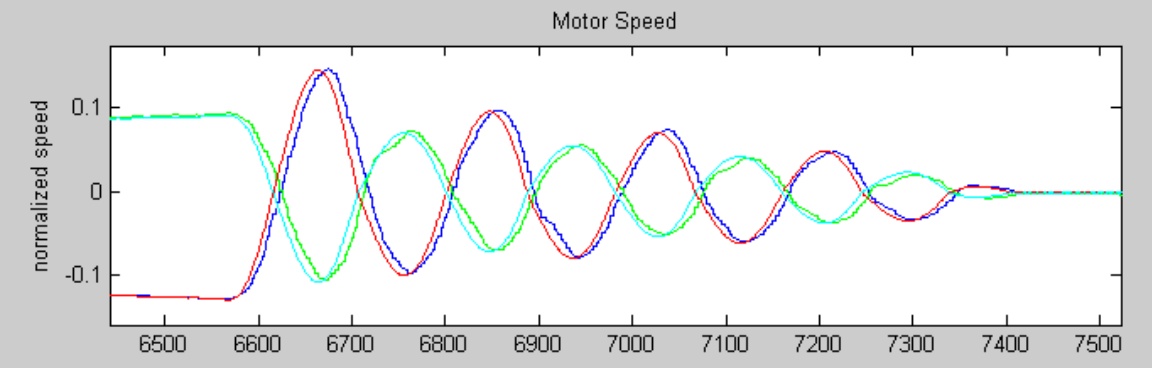


**Figure 61:** Speed tracking when boom was stopped after being raised for inertia of 200 kg/m^2.

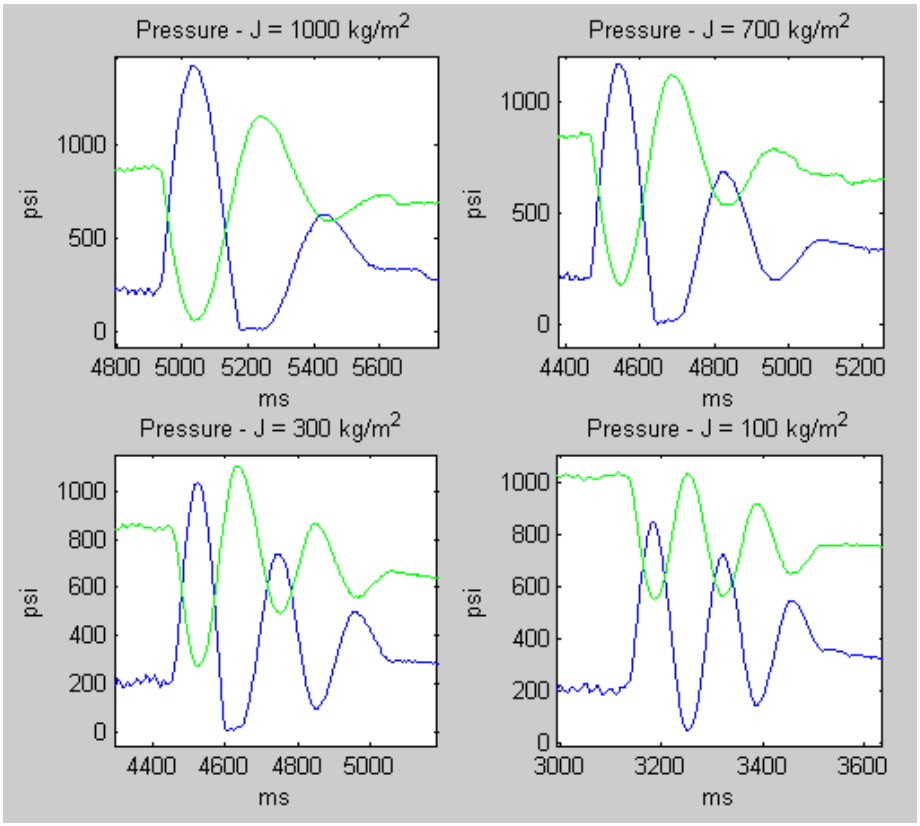


**Figure 62:** Pressure oscillations after boom was stopped for various inertias.

# CHAPTER VI

# CONCLUSION

The goal of this work was to evaluate the effectiveness of using an electric motor coupled to a hydraulic motor for Hardware-in-the-Loop Simulation. Towards this end, the boom cylinder on a backhoe was physically emulated using the motors, so that reactions with a proportional valve could be compared to those from a real backhoe. The basic layout of the HIL system is shown in Figure 63.

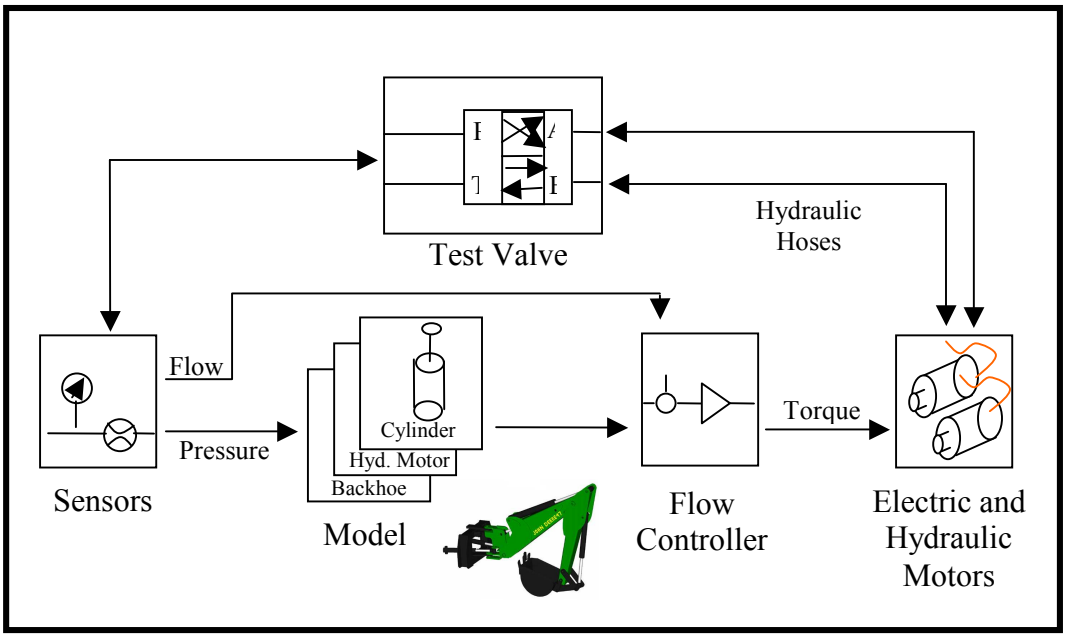


**Figure 63:** Hydraulic HIL Simulation diagram

The main disadvantage of using electric and hydraulic motors instead of valves for replicating hydraulic environments is that the inertia of the motors must be overcome, so control bandwidth is inherently lower. The bandwidth of high end servo valves can exceed 100 Hz, whereas the bandwidth of the motors used in this work was 11 Hz for small signal changes in the speed control mode. Even so, most hydraulic systems have a natural frequency less than 10 Hz, so this bandwidth may be adequate for simulating many systems.

Two versions of causality in the model were experimented with: measuring flow rates and then calculating desired pressures, and oppositely, measuring pressures and deriving desired flow rates. The flow-to-pressure scheme proved to be problematic since states had to be differentiated, and many of the system signals were too noisy for this method to be effective. Another problem is that flow changes lead to disturbance pressures (or disturbance torque), an issue encountered in many of the other research efforts discussed previously that used inverse dynamics. The pressure-to-flow scheme requires integration instead of differentiation, and also naturally avoids the disturbance pressure effect, so it proved to be a much better emulation method. A more subtle difference is that the connecting hoses or tubing become part of the "hardware" (in this case the hardware included the valve and connecting hoses), so no compensation has be done to account for hosing compliance.

Despite the inherent disadvantage of slower responses due to inertia, using motors instead of valves does provide certain advantages at low and zero flows. At zero flow, a valve inherently has difficulty controlling the pressure drop across its orifice, but a hydraulic motor can still apply torque and pressure. Hydraulic motors have their own set

of problems at low flows, though, such as higher leakage and non-linear pressure from slow moving gear teeth. Furthermore, the pressure variations from slow moving teeth worsen with higher pressures. A controller similar to the pulse control discussed above could potentially be designed to lesson this effect, so it is felt that motors still have a valuable advantage in near-zero flow situations. In a related note, measuring flows when using valves is extremely complicated at best, and no simple solution is available that can accurately measure a large variety of flows, including steady state, highly dynamic, and low flow situations. Since the hydraulic motor is a displacement device, flow can be ascertained directly from motor speed as long as leakage is accounted for, which is typically relatively small.

Results from the real backhoe compared favorably to the emulation in a qualitative sense, in that many of the behaviors evident in the real backhoe also appeared in the simulation. Also, the effect of changing mass and friction parameters in the model produced intuitively predictable results. A quantitative match between emulation and real backhoe data over the entire range of the Sauer Danfoss valve was not produced, however. While some pressure data corresponded closely between the two systems, at many times significantly higher pressures occurred in the emulation, especially for larger command inputs to the valve. Also, instabilities were found under some parameter settings that were expected to be realizable. This does not necessarily mean that the hydraulic motors cannot perform successful environment emulation, though. The effectiveness of a HIL system to create an accurate physical environment depends on two things: its ability to display a given model hydraulically, and also how closely that model reflects a physical system. It was not the goal of this work to create a perfect model of a

John Deere backhoe cylinder, so some differences between real data and emulation results are expected, and are simply due to inaccuracies in the model. To evaluate how well a particular system can create a hydraulic environment, the best one can do is to note that controlled speeds or pressures track desired ones closely. And this was found to be the case for many of the smaller Sauer Danfoss input commands. When more demanding commands were given, the motors were not able to track desired trajectories as closely because of torque limitations. So, the differences between real data and emulation are probably due to a combination of motor power limits and model deficiencies, which are most likely related to the friction component of the model.

Supposing electric motors coupled to hydraulic gear motors are chosen as appropriate actuators for an emulation system, a few recommendations can be made about considerations to take into account when designing a new system. If a given emulation environment is known, a few defining parameters can to a large extent be used to specify motor and controller requirements. Those parameters include maximum flow ($q_{max}$), maximum pressure ($p_{max}$), and maximum delta flow ($\dot{q}_{max}$), which can be derived from the required bandwidth of the environment. In a typical hydraulic motor, flow is related to rotational speed by a hydraulic motor constant, $k_d$ which has units of volume per revolution. The relation is: $Q = k_d * \dot{\theta}$, where $\dot{\theta}$ is the rotational speed of the motor, and Q is flow. The same constant can be used to relate pressure to torque: Pressure = Torque / $k_d$. Using these two relations, the connection between motor properties and environmental properties can be summarized by the following equations:

$$Torque_{required\_from\_pres} = p_{max} * k_d$$

$$\dot{\theta}_{max} = q_{max} / k_d$$

$$\ddot{\theta}_{max} = \dot{q}_{max} / k_d$$

Using a simple linear model for a motor, the $\dot{q}_{max}$ requirement can also be converted into a torque requirement:

$$\text{Torque}_{required\_from\_delta\_flow} = J\frac{\dot{q}_{max}}{k_d} + b\frac{q_{max}}{k_d} = \frac{1}{k_d} * (J\dot{q}_{max} + bq_{max})$$

As can be seen, by making $k_d$ smaller, a corresponding smaller torque is required for a given maximum pressure, but this has the effect of increasing the torque necessary to achieve a given change in flow. While an optimum value can be found for the hydraulic motor constant using these equations, one must also take into account the controller's ability to meet the bandwidth requirement, as well as availability in electric motor selection with respect to inertia and torque. An important consideration to keep in mind is that the inertia will be the combination of the coupling, electric motor, and hydraulic motor's inertia, which can be substantially higher than that of the motor alone. Approximate values for the inertias used in this study can be found in the appendix. Some other considerations to keep in mind are the motor leakage, simulation step time and its effect on controller performance, and the limitation noise may have on performance.

In summary, the electric and hydraulic motors used in this case study have been shown to effectively replicate many of the behaviors found in a real backhoe cylinder and valve interaction. Notably, similar pressures to the actual system were found in low and zero flow situations, which a region of flow that is inherently difficult for valve-based

systems to control. Another caveat to valve-based systems is that they require an additional pressure or flow supply to simulate over-running loads, whereas an electric and hydraulic motor can do this independently. Still, due to their inertia, motors will naturally have a lower response bandwidth than valves, so careful consideration should be given to torque limitations, inertia, and required bandwidth when selecting a device for hydraulic HIL simulation.

# CHAPTER VII

# FUTURE OUTLOOK

A better assessment for the capabilities of using electric and hydraulic motors would most likely be found by using a simpler test valve and environment than the Sauer Danfoss valve and John Deere backhoe used in this work. While they provide interesting non-linear effects and demonstrate the value for such an emulation system, these difficulties can make initial evaluations somewhat difficult. Also, in many of the works referenced above, nonlinear and linearized models were developed to aid in understanding of the system limitations and their causes. This would probably be the next step taken if work was continued on this topic.

There are undoubtedly improvements that could be made to the pressure, speed, and flow controllers in this work, as fairly simple strategies such as lookup tables and PID control comprised the majority of control techniques. Improvements in control would likely lead to increased bandwidth, and expanded capabilities for simulating stiffer systems with higher natural frequencies. However, the fact that motors must fight their own inertia means that a simulation system based on valves will always be better suited for emulating environments with faster dynamics.

The long term goal of this work would be the creation of a framework for choosing the proper actuator for a given HIL simulation, and also perhaps deciding when HIL simulation is beneficial. The main idea of HIL simulation is to assess the realistic performance of a piece of hardware in situations where pure computer simulation would be unreliable. This, of course, is only possible when one is able to sufficiently emulate

the environment surrounding the device in question. If salient features of the environment are also too difficult or unreliable to model, then results would be from a real piece of hardware in a fictitious world, and perhaps more useless than total simulation. There are, however, many situations in which many of the important modeling complexities are contained within a particular piece of hardware, such as a flow control valve, and much knowledge can be achieved by placing it in a variety of different environments that would have been prohibitively expensive to test with full scale systems. Deciding which of those cases in which HIL simulation adds value is a much larger question, and depends on the complexity needed in the environment, and the relative unreliability of a pure simulation effort. Assuming that HIL simulation is found to be beneficial, the next step is designing a system and choosing actuators that can recreate the necessary environment. The design would be based on many factors, such as types of flow, pressures, stiffness, necessary connections, etc., finally guiding choices between different motor sizes, or valves, and controllers, among many other details. While creating a general-purpose framework for making all of these decisions may be unnecessary, some effort in this direction would provide valuable improvement to current HIL design, which has primarily been ad hoc in nature for the majority of systems.

# APPENDIX

## *HIL System Specifications*

**Electric Motors:**
(2) Siemens 3 phase brushless servomotor: 1FT6102-8AF71-3AK1
Nominal torque: 19.5 Nm
Nominal speed rating: 3000 rpms
Inertia: 9.9e-3 $kgm^2$

**Hydraulic Motors:**
(2) Sauer Danfoss gear motors: SNM2/95 C106 MER1/1D
Displacement: .58 cubic inches / rev
Inertia: 35e-6 $kgm^2$
Coupling Inertia: .0181 $in\text{-}lbs\text{-}s^2$

**Proportional Valve:**
Sauer Danfoss Proportional Valve: PVG 32 with PVES electrical actuator

**Backhoe:**
John Deere 4410 series tractor and model 47 backhoe

**Simulation Software:**
The Mathworks xPC Target real-time operating system

**I/O cards:**
National Instruments 6052E A/D
Profibus CP5613 communication PCI card (for communicating with motor amps)

## *HIL System General Capabilities*

| | |
|---|---|
| Pressure: | 1200-1700 psi depending on flow |
| Flow: | 7.5 GPM ~ 3000 rpms, higher torques may limit speed |
| Simulation Freq: | 1000-1500 Hz. Max Time-of-Execution recorded was .00049 sec, but system usually overruns when run at 2000 Hz. |
| Bandwidth: | 11 Hz for small amplitude oscillations |

# REFERENCES


[1] N. D. Manring, S. B. Kasaragadda. The theoretical flow ripple of an external gear pump. In *Proc ASME International Mechanical Engineering Congress & Exposition*, pp 19-27, New Orleans, Louisiana, November 2002,.

[2] R. Zhang, D. E. Carter, A. G. Alleyne. Multivariable control of an earthmoving vehicle powertrain experimentally validated in an emulated working cycle. In *Proc ASME International Mechanical Engineering Congress & Exposition,* Washington, D.C., November, 2003.

[3] E. A. Prasetiawan, R. Zhang, A. G. Alleyne, Tsu-Shin Tsao. Modeling and control design of a powertrain simulation testbed for earthmoving vehicles. *Journal of Fluid Power Systems and Technology, Transactions of the ASME*, 6:99-105, 1999.

[4] J. F. Gardner, B.T. Kulakowski, S. A. Gore, D.A. Streit. Modeling and tuning of hydraulic road simulators for heavy vehicle testing. *Journal of Fluid Power Systems and Technology, Transactions of the ASME*, 2:99-105, 1995.

[5] J. L. Lahti, S. J. Andrasko, J. J. Moskwa. A transient hydrostatic dynamometer for single cylinder engine research with real time multi-cylinder dynamic simulation. In *Proc ASME International Mechanical Engineering Congress & Exposition,* pp 223-230. Washington, D.C., November, 2003.

[6] S. Jung, Y. J. Lee, W. J. Book. Hardware-in-the-loop simulation for control development in EHPV applications. In *Proc ASME International Mechanical Engineering Congress & Exposition,* Washington, D.C., November, 2003.

[7] L. Qinghe, P. Zongcai, W. Shenglin. A study on producing mechanism and eliminating methods of the disturbance torque in electrohydraulic load simulator. In *Fourth JHPS International Symposium on Fluid Power*, Tokyo, November 1999.

[8] H. Ohuchi, H. Ikai. Control of a load simulator. In *First JHPS International Symposium on Fluid Power*, Tokyo, 1989.

[9] P. K. Guerrier, K. A. Edge. Hydraulic Emulation of injection moulding. In *Proc ASME International Mechanical Engineering Congress & Exposition,* New York, NY, 2001.

[10] Z. H. Akpolat, G. M. Asher, J. C. Clare. Dynamic emulation of mechanical loads using a vector-controlled induction motor-generator set. In *IEEE Transactions on Industrial Electronics*, Vol. 46, No. 2, pp 370-379, 1999.

[11] C. Ramden, A. Jansson, J. O. Palmberg. Design and analysis of a load simulator for testing hydraulic valves. In C. R. Burrows and K. A. Edge, editors, *Fluid Power Engineering: Challenges and Solutions, Tenth Bath International Fluid Power Workshop*. Research Studies Press Ltd., Baldock, Hertfordshire, England, 1998.

[12] C. Ramden, J. O. Palmberg. Load simulation—an effective tool for testing hydraulic valves. In *Proceedings of The Sixth Scandinavian International Conference on Fluid Power*, Tampere, Finland, May 1999.

[13] C. Ramden, J. O. Palmberg. Experimental investigation of the accuracy of a load simulator for testing fluid power valves. *Technical Report* LiTH-IKP-R-1040, Division of Fluid and Mechanical Engineering Systems, Department of Mechanical Engineering, Linkoping University, SE-581 83 Linkopoing, 1999.

[14] Zhen Li, M. Kyte, B. Johnson. Hardware-in-the-loop real-time simulation interface software design. In *Proc of The 7th International IEEE Conference on Intelligent Transportation Systems*, pp. 1012-1017, October 2004.

[15] J. Leitner. Space technology using hardware in the loop simulation. In *Proc of IEEE Aerospace Applications Conference*, 2:303-311, Aspen, CO, February 1996.

[16] D. L. Martin, L. Turner III. Role of hardware-in-the-loop simulation in fly-by-light system development. In *Proc of 11th IEEE/AIAA Digital Avionics Systems Conference*, pp. 19-24, Seattle, WA, October 1992.

[17] M. Rimer, M. Falco, M. J. Solan. Systems integration test laboratory, application & experiences. In *Proc of the eighth symposium on space and nuclear power systems*, pp. 1181-1193. The American Institute of Physics, 1991.

[18] W. K. N. Anakwa, H. P. Roca, J. Lopez, A. Malinowski. Environments for rapid implementation of control algorithms and hardware-in-the-loop simulation. In *IEEE 28th Annual Conference of the Industrial Electronics Society*, 3:2288-2293, November 2002.

[19] M. Yamazaki, M. Brackx. Analysis of automatically generated vehicle system control software in a HIL environment. In *Proc. Of the American Control Conference*, pp. 3135-3140, 2002.

[20] J. Schaffnit. Hardware-in-the-loop simulation and rapid prototyping for the development of control functions for diesel engines. In *Proc. 1st IFAC Conf. Mechatronic Systems*, 2000.

[21] J. Kim, M. A. Srinivasan. Computationally efficient technique for real time surgical simulation with force feedback. In *Proc. 10th Sym. On Haptic Interfaces For Virtual Environment & Teleoperator Systems*, 2002.

[22] Y. Senta, E. Okamura. HIL simulation system for hdd servo firmware. *IEEE Transactions on Magnetics*, 38 (5), pp. 2204-2207, 2002.

[23] The Mathworks. http://www.mathworks.com/company/user_stories/

[24] dSpace. http://www.dspace.de/ww/en/pub/company/press.htm

[25] National Instruments. http://www.ni.com/realtime/

[26] A. Toft Fensvig. *On Dynamic Modeling of Hydraulic Control Systems and the Development of a Novel Static/Dynamic Flowmeter*. PhD thesis, Control Engineering Laboratory, Technical University of Denmark, DK-2800 Lyngb, Denmark, December 1975.

[27] R. W. Miller. Flow *Measurement Engineering Handbook*. McGraw-Hill Publishing Company, 3rd edition, 1990.

[28] National Instruments. *SCB-68 68-Pin Shielded Connector Block User Manual*. December 2002 Edition, Part Number 320745B-01.

[29] A. Bonchis, P. I. Corke, D. C. Rye. Pressure-based, velocity independent, friction model for asymmetric hydraulic cylinders. In *Proc. IEEE International Conference on Robotics and Automation*, 3:1746-1751, Detroit, MI, May 1999.