Bradley University
Department of Electrical and Computer Engineering

Active Suspension Test Platform - Controls

Project Report

By
Jerry L. Campbell

Advisor
Mr. Steven D. Gutschlag

May 11, 2004

Abstract

The purpose of this two team project was to design and build a reliable test platform to simulate the effects of various inputs expected by active suspension systems during normal operation (e.g., automotive suspensions and operator platforms for agricultural equipment). This portion of the project was to initially determine a mathematical model for a linear actuator, then design, simulate, and construct an analog controller to ensure the platform motion correctly follows the desired movement regardless of the load placed on the platform. A second ECE team responsible for hardware selection will utilize the knowledge gained from this project to design a simple micro-controller based digital controller. The completed platform controller will be utilized to provide simulated inputs to test future active suspension design projects for the Bradley University ECE department.

Table of Contents

Project Overview:

The overall purpose of the project is to design and build a reliable test platform to simulate the effects of various inputs expected by active suspension systems during normal operation (e.g., automotive suspensions and operator platforms for agricultural equipment). The platform will be utilized to provide simulated inputs to test future active suspension design projects for the Bradley University ECE department.

The initial focus of this project was to develop a mathematical model for a given linear actuator and motor assembly. The project then developed a closed loop feedback model of the platform, actuator, and motor. The closed loop model provided an adjustable pulse width modulated (PWM) signal to ensure the platform provides the desired input regardless of the load applied to the platform. Matlab and Simulink were used to design and simulate the required feedback system, and the actual feedback control system will eventually be implemented utilizing an EMAC MicroPac 535 micro-controller based development system. Another project team will select the proper linear actuator for the system and provide the power electronics required for the interface with the EMAC MicroPac based feedback control system. Figure 1 shows a basic system block diagram.

System Block Diagram:

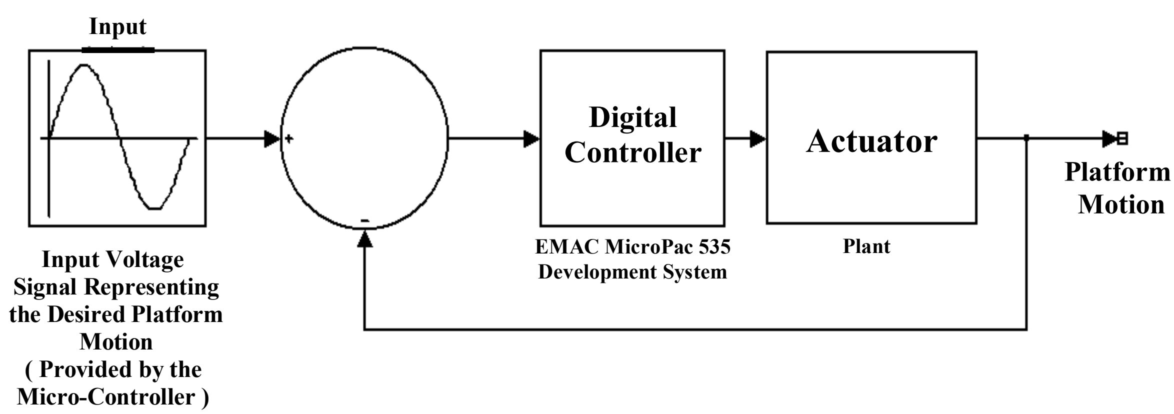


Figure 1 - System Block Diagram

Notes: 1) For the system as defined in the figure above, only one global input and output exist. There are, however, numerous other signals present in the system's interior.

2) In the system above, the input voltage signal, controller, and the summing junction used to generate the error signal are all calculated inside the EMAC MicroPac 535 development board.

Actuator Block Diagram:

An error signal is generated by comparing the actual platform position and the desired platform motion selected by the user via the keypad. The error signal is converted to a PWM controller voltage signal which is applied to the actuator as shown in figure 2 below. The actuator will provide the desired platform motion, and a position signal will be returned to the controller for continuous feedback control.

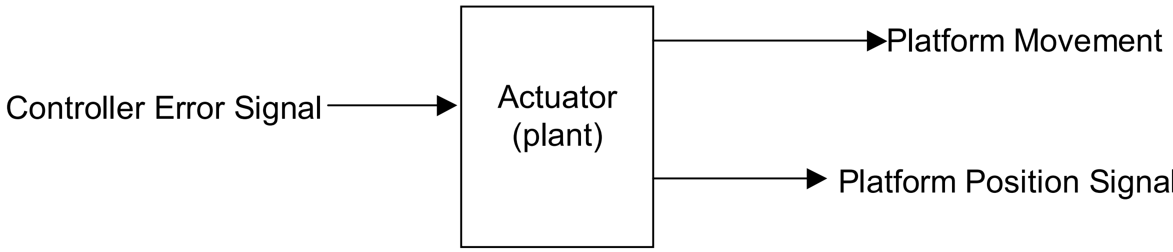


Figure 2 - Actuator Block Diagram

Controller Block Diagram:

The EMAC MicroPac 535 micro-controller based development system will serve as the system controller (figure 3). The controller will be software based and receive inputs from the operator and the platform. The user will enter the platform's desired motion via the keypad, and the controller will receive a feedback position signal from the platform. The controller will perform necessary calculations and produce an error signal that will be sent to the actuator. The LCD will also be updated by the controller to inform the user of the mode of operation.

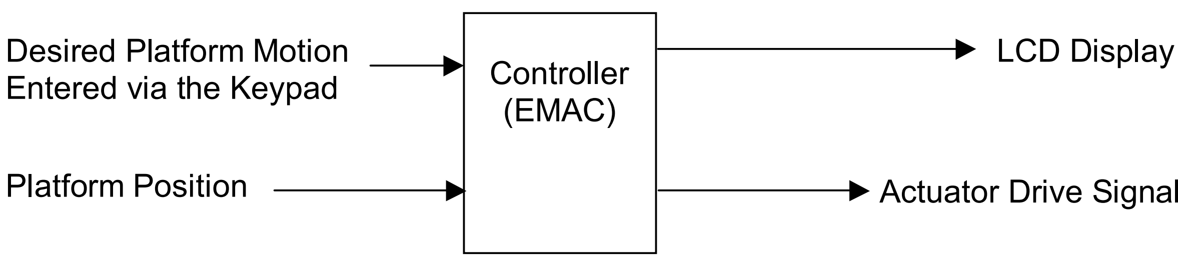


Figure 3 - Controller Block Diagram

Controller Software:

The initial software design will consist of three main modules: motion, calculations, and interaction. The motion module contains the code used to provide the linear actuator with the necessary commands to move the platform. The calculation module collects data entered via the keypad and data from the position sensor. Collected data is used to calculate the distance moved, acceleration, and velocity of the platform. These calculations are used to create the error signal. The interaction module will control the LCD and interpret the user's input to the keypad (figure 4).

The user will be prompted to enter the desired motion of the platform. The user will enter the wave shape of the platforms motion, frequency if applicable, and amplitude of the motion. The user's data will be interpreted by the calculation module and then used to control the platform. After the initial control calculations are complete, a prompt command will appear on the LCD display. Platform motion will not begin until the user has verified he/she is ready for the platform to move.

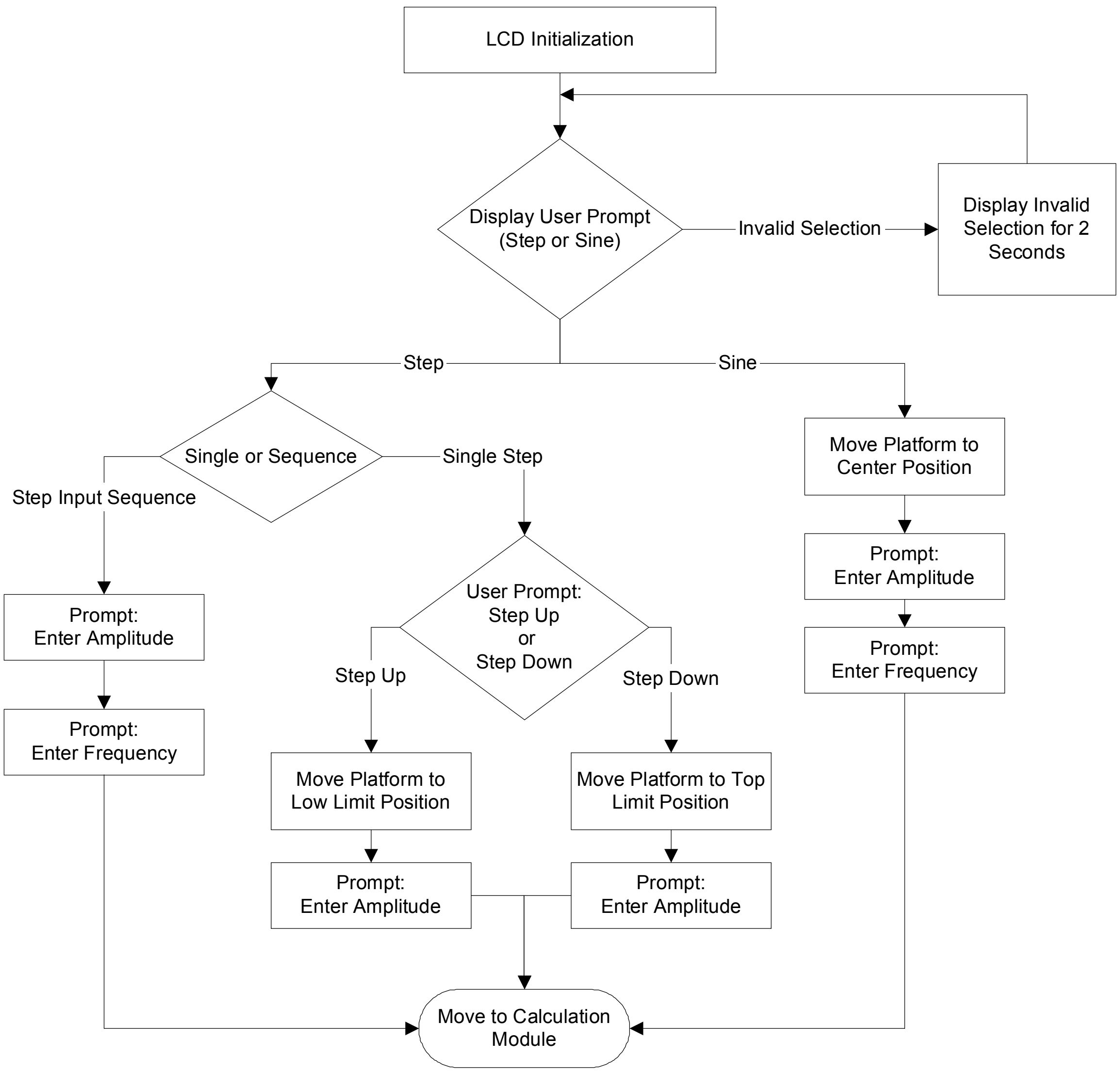


Figure 4 Software Flow Chart for Interaction Module

If the selections entered via the keypad are valid, a motion module is chosen. If the platform is unable to perform the requested motion an error message is displayed. The software will follow the same sequence of commands regardless of the input selected, but the platform will perform different motions. If a sinusoidal input is selected, the platform will move in a continuous up and down motion in relation to the input data entered. A step input will make a single step up or down, whereas a square wave will make steps up and down in relation to the amplitude and frequency selected.

The sensor, which is an integral part of with the actuator hardware, will provide a signal proportional to the position of the platform. This value will be used in the calculation module to compute the acceleration, velocity, and displacement of the platform. When the microcontroller is initialized, the platform will start at an initial position. The platform will follow the desired motion as specified by the user. When another desired input is selected, the platform will first return to the initial position before carrying out new movements. The starting position will depend on the type of motion selected.

Another software module will calculate velocity. This module will take the platforms displacement and divide it by the time that the platform took to move. Again dividing incremental changes in velocity by time will yield acceleration since acceleration is the derivative of velocity.

System Inputs and Outputs:

The only true input to the system is the desired platform motion as selected from the keypad. The output of the system will be the platform's motion which precisely follows the desired motion.

System

Inputs	Outputs
Desired Platform Motion (R)	Actual Platform Motion (C)

Note: Desired system response is C=R, or C/R = 1.0

EMAC MicroPac 535 micro-controller based development board (Controller)

Inputs	Outputs
Keypad (Desired Platform Motion)	Actuator Drive Signal
Platform Position	LCD Display

Actuator (plant)

Inputs	Outputs
Error Signal from Controller	Platform Movement
Disturbance Force (load)	Position Signal

Figure 5 System Input/Output Table

Modes of Operation:

The operator will be prompted to specify a desired input via the keypad of the EMAC MicroPac 535 micro-controller based development board to select the mode of operation. The various types of inputs will permit the user to select sinusoidal, step, and triangular platform motions. The operator will then be asked to select the desired frequency and amplitude of the platform’s motion. The selections will be limited by the software timing and selected hardware. Limitations will be determined once initial research has been conducted and a final hardware configuration has been determined. The system will control the platform position to ensure the system output correctly follows the desired movement regardless of the load placed on the platform (within the constraints of the selected hardware).

System Identification:

The first step of the project was to determine an accurate mathematical model for the DC linear actuator that was to be used for the project. An accurate model is required before any practical controller design can begin. A detailed specification sheet was not available for the device being used for this project. If a detailed specification sheet was available for the device, there are mathematical and analytical steps that can be followed to determine an approximate transfer function to describe the plant. These methods are discussed and practiced extensively in the EE431 Controls course. Since the specifications needed for these calculations were not available to the project, the required parameters were obtained by means of frequency response and various load versus speed measurements.

Several parameters must be obtained to create an accurate mathematical model to describe the linear actuator (figure 6). The key parameters needed to create a mathematical description are the equivalent inertia and damping, the torque and back EMF constants, gear ratio, and screw lead. There are many other parameters involved in system identification that are usually omitted from the initial model due to complexity. Undergraduate control students work with systems that are assumed to be linear or linear approximation models. Affects such as dead band and backlash are frequently omitted from initial designs. From the experimental data obtained for the actuator used in this project it is quickly apparent that these parameters cannot be ignored.

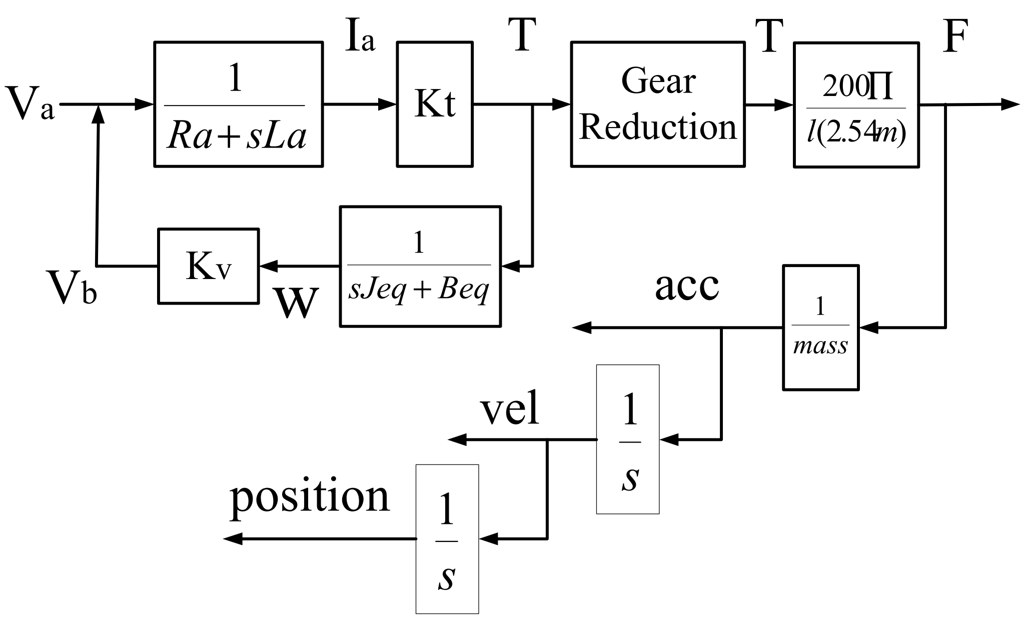


Figure 6 Block Diagram of a Simple DC Machine (Open Loop)

Frequency Response:

The frequency response represents the system's response to sinusoidal inputs at varying frequencies. The output of a linear system to a sinusoidal input is a sinusoid of the same frequency, but with a different magnitude and phase. The frequency response is defined in terms of the magnitude and phase variations between the input and output sinusoids. Figure 7 shows a data sample used to create the frequency response plots.

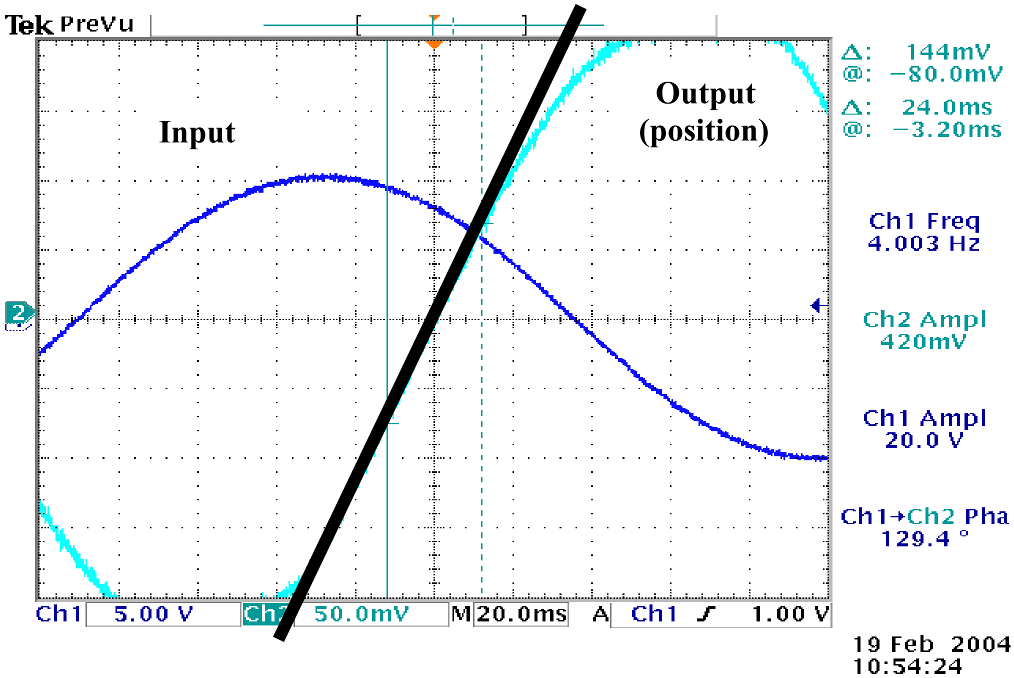


Figure 7 Tektronix Scope Plot of Frequency Response Data Sample

The usable frequency range of the system was determined to be 400mHz to 10Hz. This was due to the fact that if the operating frequency dropped below 400mHz, the position reached the upper and/or lower physical limit of the actuator. At frequencies above 10Hz, the actuator did not move because the applied signal changed too rapidly. A data table was created with fixed measurement points over the operating frequency range. For each data point, a sine wave of constant amplitude was applied to the actuator, and an oscilloscope plot similar to figure 7 was obtained. From each plot, two measurements were obtained and recorded. First, the phase difference between the applied signal and the position signal was recorded. Next, the slope of the position signal at the inflection point was measured. Measuring the slope of the position signal yields a velocity value, and hence, measuring the slope of the position at the inflection point yields maximum velocity. This measurement was obtained by two methods. First, the data was collected using the cursors on the TDS3012B oscilloscope. The data was again obtained by drawing a line tangent to the inflection point with a straight edge and measuring the slope by hand. This was done because the oscilloscope was measuring small amplitudes over very small time differences, and the accuracy of the data was in question.

Once this data was collected, the values were entered into Matlab to plot a piecewise linear plot of the recorded frequency response data of the system over the range of 400mHz to 10Hz. Figure 8 shows the magnitude curve of the frequency response of the system. The first important piece of information that was obtained from this data was that the system appeared to be a first order system. The system did not appear to contain integration because the plot was created by velocity measurements. A –20 dB/decade slope reference line was drawn to clearly show the system appears to be a first order system with assumed time delay characteristics. This plot shown in figure 8 also shows that a system pole (-3 dB point) appears at approximately 42 rad/sec.

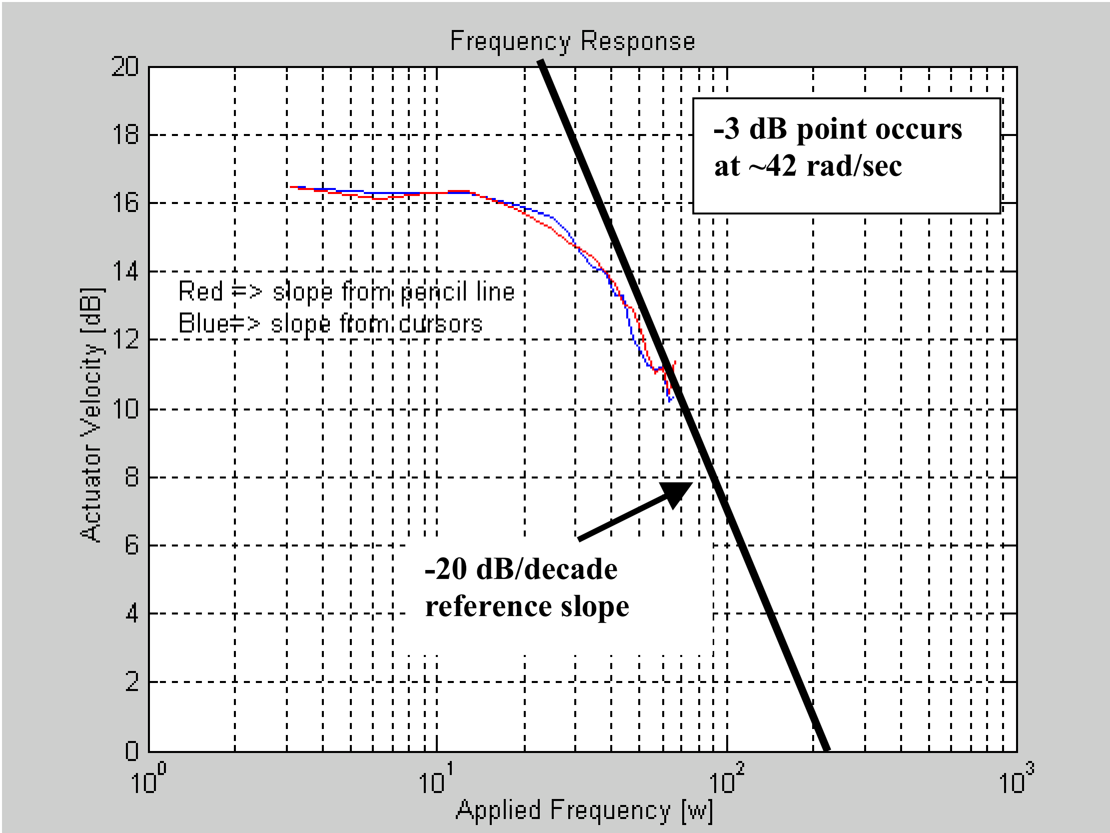


Figure 8 Magnitude Plot of Frequency Response Data

The next step was to enter the phase data that was collected into Matlab and create a piecewise linear plot of the phase response of the system. In a first order system without time delay or additional phase shift due to non-linearities, the –45 degree point will match the –3 dB point from the magnitude curve. Figure 9 shows the Matlab phase plot of the system. The –45 degree point is at approximately 28 rad/sec. This location occurs at a lower frequency than the pole location from the magnitude curve. This is a clear indication of non-linear effects apparent in the system. Non-linearities add phase lag to the system which slows the response of the system.

A pure time delay will not affect the magnitude curve of a frequency response. A pure time delay will only introduce phase lag and therefore move the –45 degree point (pole location) of the phase response. Due to the small frequency range of sample data collected, I was unable to calculate the actual time delay approximation from the frequency response plots. It was also becoming clear that the affects of non-linearities are significant for the actuator under consideration.

It was decided that an initial model should be used that could correctly model the system in a manner that included the effects of all the non-linearities. Simulink was used to create an initial working model of the system. Simulink was a very useful tool because it allows the user to place output scopes throughout the system to ensure the model is behaving in a manner consistent with the actual device.

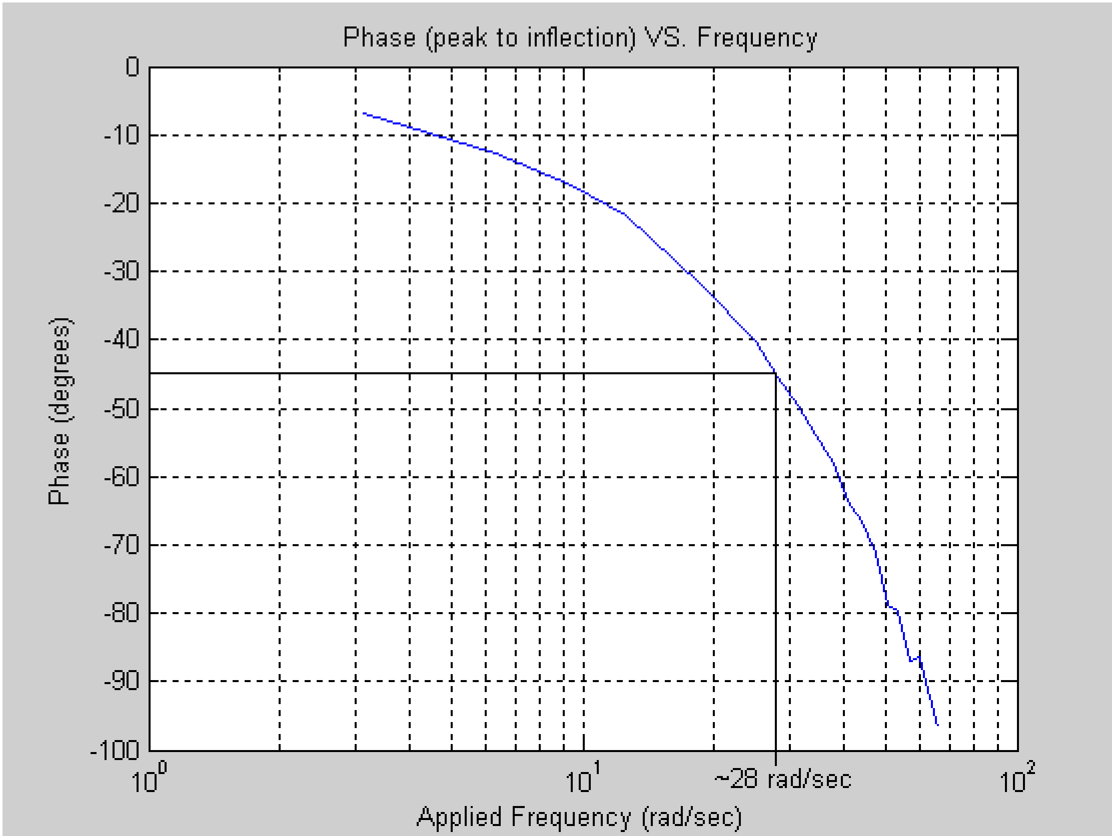


Figure 9 Phase Plot of Frequency Response Data

The Simulink model shown in figure 10 is the initial model used to include the effects of all the nonlinear parameters included (full page of model is included in the Appendix on page 1). This model has many advantages and disadvantages.

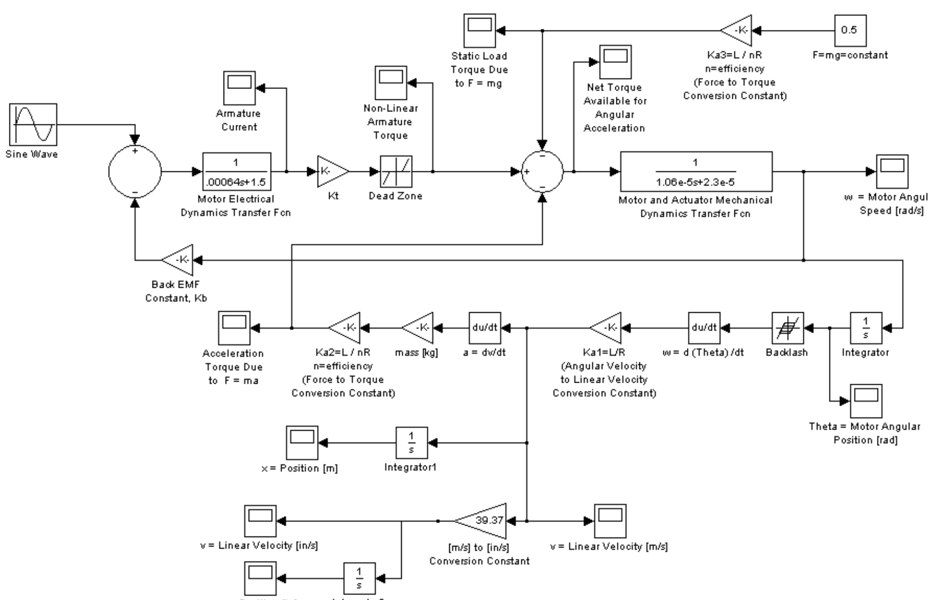


Figure 10 Preliminary Simulink Model for the Warner Linear Actuator Including Non-Linear Effects

A Simulink model of this complexity has many advantages. Constants and parameter values can be changed easily, and scopes can be placed internal to the device being modeled. Scopes allow the user to view the signals present at any given location within the system. This type of model allows the user to see inside the plant and make quick tuning adjustments to the model to ensure the plant is accurately represented. The main disadvantage to a Simulink model of this type is the fact that it cannot be easily placed into a simple traditional control system block diagram. This makes it virtually impossible to use the model for mathematical hand calculations of controller components.

The initial Simulink model must be simplified before most undergraduate control students can begin to design a controller for the system. This model only served as a beginning point to acknowledge the non-linearities. The main non-linear component present in this system is backlash. Backlash is a term used to refer to the play in the internal gear train of the system. When the gears are meshed and turning in a given direction, they must disengage from their present direction, travel through any play in the gear teeth, engage the gears in the opposite direction and begin to move when the applied voltage changes polarity. These effects, shown in figure 11, are highly non-linear in nature.

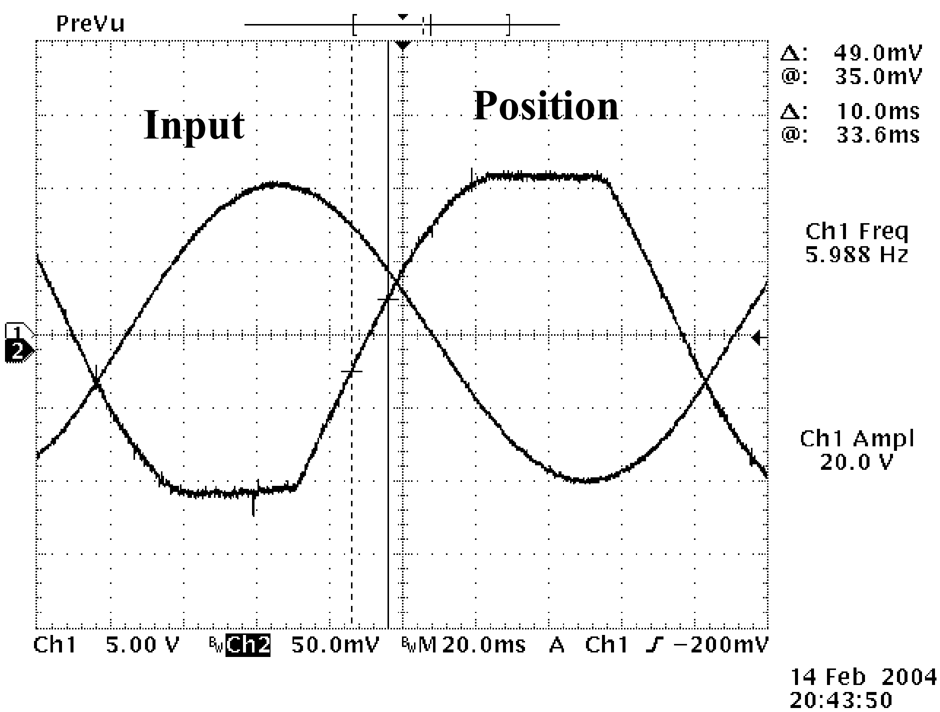


Figure 11 TDS Oscilloscope Plot Displaying Backlash Effects

The Simulink model shown in figure 10 and again in the appendix must be simplified before a controller can be mathematically designed using control methods. Many assumptions must be made to simplify the model. The system must first be assumed linear. Undergraduate control classes at Bradley University are taught on this assumption. The backlash and dead band must also be assumed to be non-existent. These assumptions certainly introduce error into the model, but are required to obtain a linear approximation of the model. However, it is not unrealistic to assume a linear model for the actuator, since once the system is operated in a closed loop configuration, a high loop gain can be used to minimize the effects of the non-linearities present in the system.

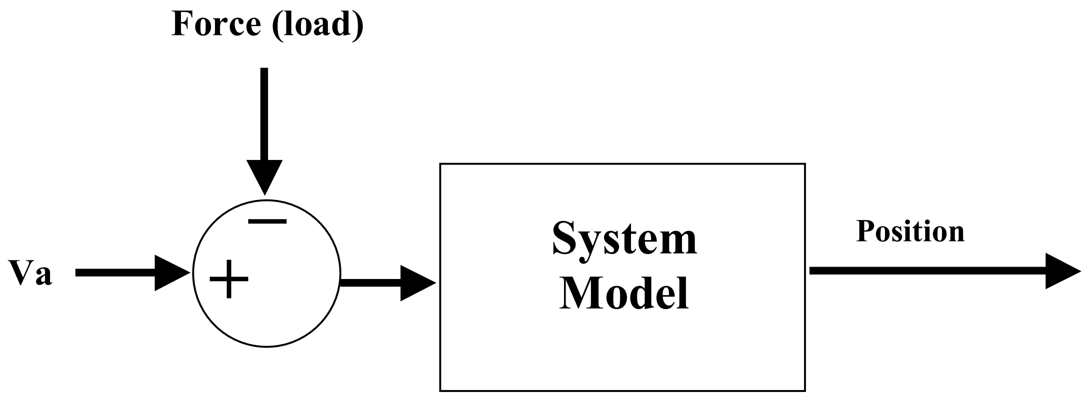


Figure 12 Simplified System Model

The model needed to be simplified to a form that could easily be placed into the system diagram shown in figure 12. A model was needed that could show, for a given load or disturbance force on the platform, that a known applied voltage will produce a predicable platform position or change in position. Through many algebra steps and block diagram reductions, a simplified Simulink model was obtained. This simplified model is shown in figure 13 and again as a full page diagram in the appendix on page 2.

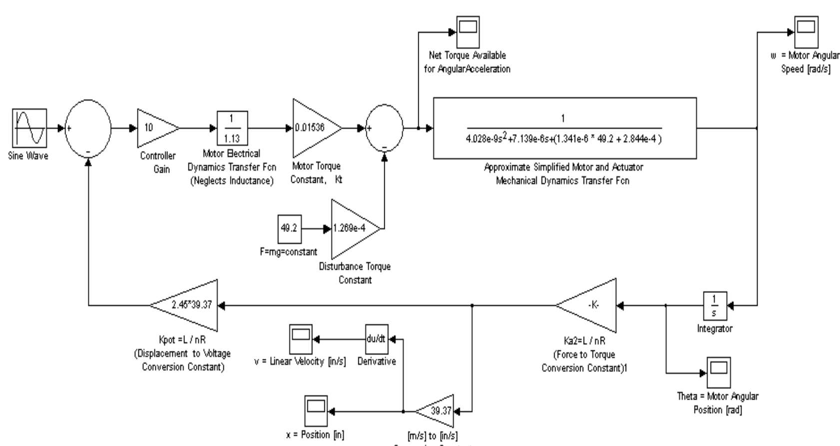


Figure 13 Simplified System Simulink Model
(Enlarged view is available in the appendix on page 21)

<u>Phase Margin Determination:</u>

Phase Margin was determined to be the most important design specification for this system. Through conversations with people with more practical control experience than the project members, it was determined that the phase margin for a system of this type should be held above 60 degrees. The project team was also informed that a proportional controller gain of 100 is a realistic practical beginning point. Using the simplified Simulink model from figure 13, the loop was closed and a proportional controller of gain 100 was added. To determine the phase margin of the system in this configuration, the feedback loop was opened and a constant zero was placed at the negative terminal of the summing junction as shown in figure 14 (and expanded in the appendix on page 22).

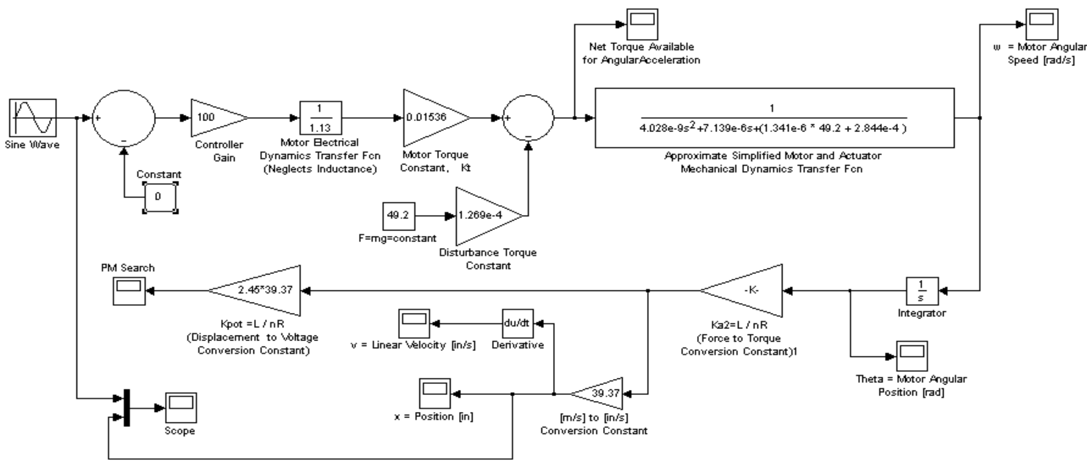


Figure 14 Simplified Simulink Model Configured to Determine Phase Margin

(Enlarged view is available in the appendix on page 22)

To determine the phase margin of the system in this configuration, the frequency of the applied input was varied until the peak to peak magnitude of the applied signal matched the peak to peak magnitude of the signal in the feedback loop. At this frequency, the loop gain of the system is one (or 0 dB), which by definition is the gain crossover frequency. The phase margin is measured at the gain crossover frequency, and as figure 15 shows, the gain crossover frequency for the model represented by figure 14 is at 6 Hz. Ignoring the transients, the magnitude of the open loop path signal matches the applied signal at 2 [V] peak to peak. Using this data plot, the phase angle of the plant was calculated as the phase difference between the two curves. The phase angle subtracted from 180 degrees resulted in a phase margin of approximately 78 degrees when the proportional controller gain is 100.

The gain of the proportional controller was changed to 10 and the phase margin calculation was performed again. With a proportional controller gain of 10, the phase margin was calculated to be approximately 55 degrees. This value was lower than the 60 degree mark previously set, but the value was tolerable. It was now known that a proportional gain controller of values 10 to 100 would produce a phase margin within reasonable limits. The concern now moved to designing a controller that would allow the system to track a desired input at a reasonable frequency. Unfortunately the system Identification consumed about four fifths of the semester.

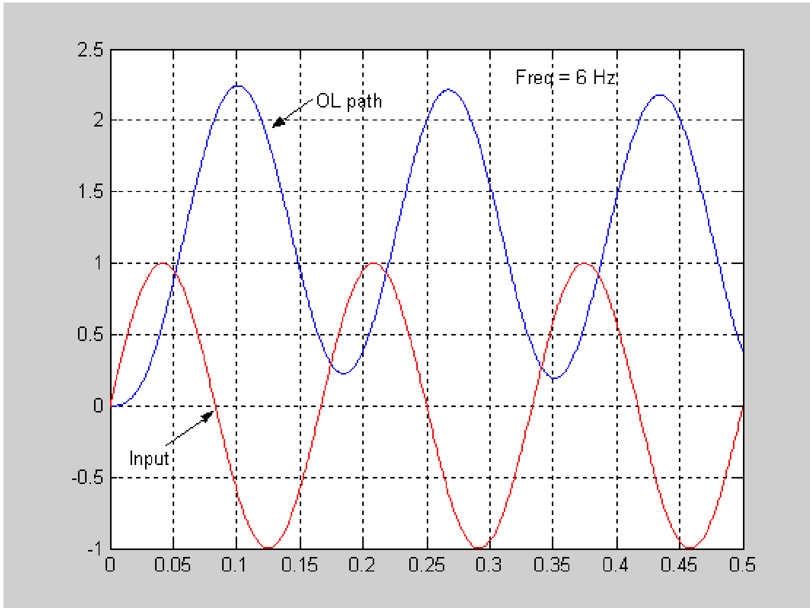


Figure 15 Matlab Plot Displaying Phase Margin Calculation Data

System Performance:

In an open loop configuration, unless a DC offset is applied, the platform will slowly approach the lower physical limit of the actuator as shown in figure 16. When the system operates in a closed loop configuration with a basic proportional gain controller, the platform motion accurately tracks an applied sinusoidal input signal at low frequency.

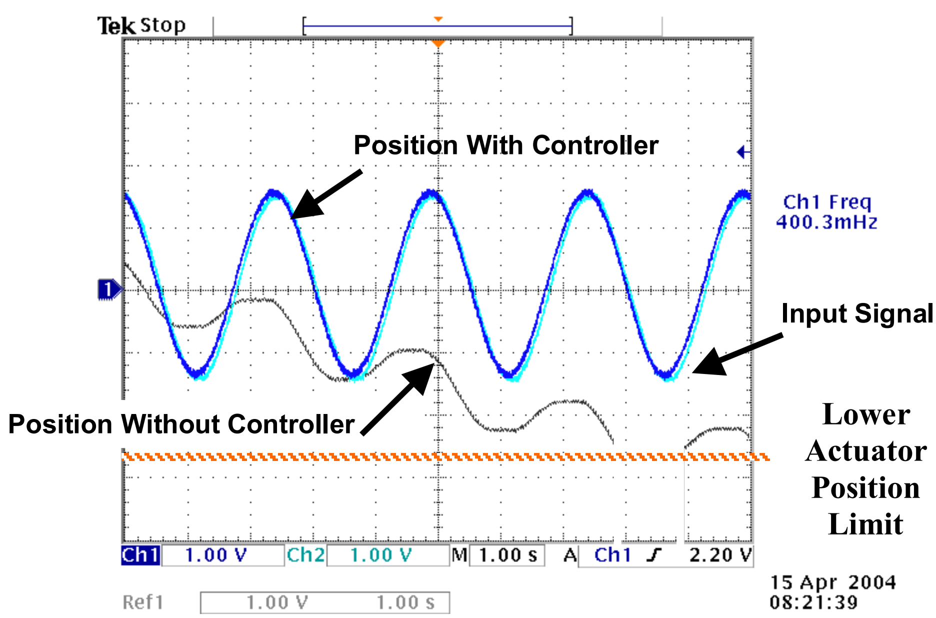
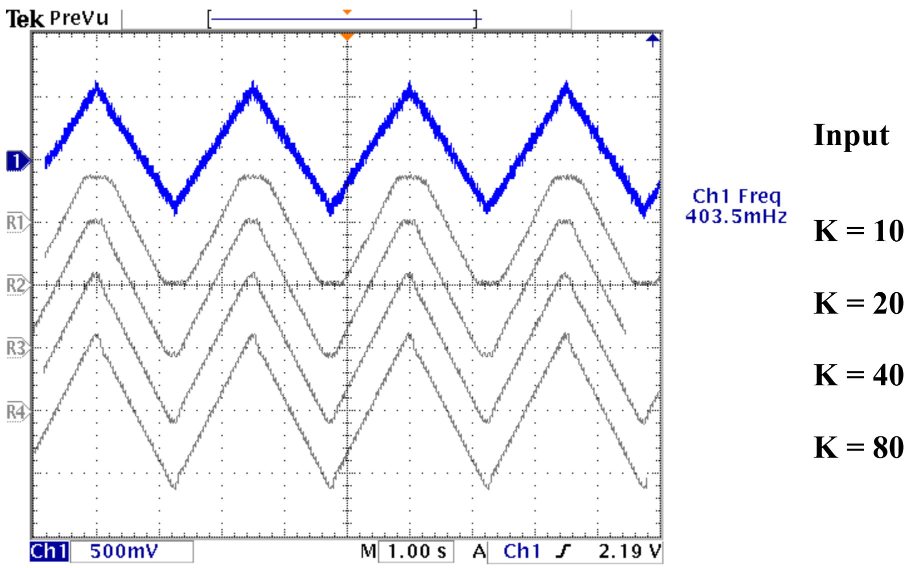


Figure 16 System Performance Comparison (Open and Closed Loop)

A proportional controller gain as low as 10 will allow the platform to accurately follow a desired input at low frequency with a load applied to the platform. More sophisticated controllers need to be implemented to accurately track complex inputs at higher operating frequencies. The project only had time to test and implement a basic analog proportional controller. The project was able to show that the non-linear effects such as backlash could be minimized by increasing the loop gain of the system. Figure 17 shows how the backlash effects present while tracking a triangle wave are reduced as the proportional controller gain and thus the loop gain are increased.



Note: Backlash Effects Minimized as Loop Gain Increases

Figure 17 System Performance as Loop Gain Increases

The implemented controlled worked well in tracking slow signals, but as seen in figure 18, the system is very slow to track rapid input changes. With a proportional

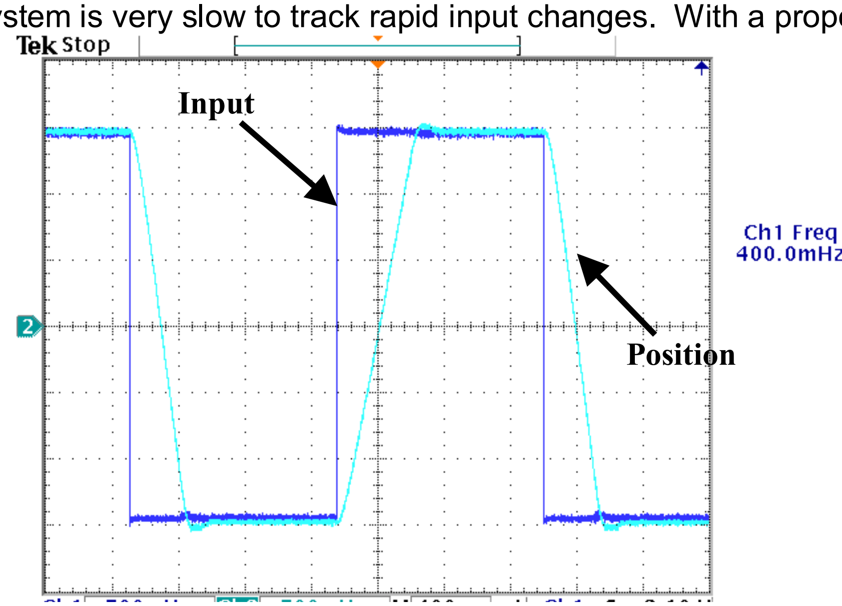


Figure 18 Closed Loop Step Response (K = 40)

controller gain of 40, the system takes approximately 500ms to reach the same level as an applied step input. Even if the controller gain is doubled to provide a gain of 80, the system still requires 350 to 400 ms to match the step input. Practical implementation of this system with a simple proportional gain controller is not possible without restricting the rates of the applied input signals. A faster system is needed to build a practical test platform.

Controller Options:

The project's initial design plan was to implement a PI controller. This type of controller would create a Type 1 system and theoretically ensure the steady state error of the system would be zero for a step input. However, a PI controller was not implemented because the integration would add 90 degrees of phase lag to the system. Our current system only had 55 to 78 degrees of phase margin, so the system became unstable when the integration was added. This was quickly observed in the Simulink modeling of the system, and although lead networks could be added to stabilize the system, time did not allow this alternative to be pursued.

Problems Encountered:

The system identification portion of the project was initially allotted 3-4 weeks of time to complete. Deriving an accurate model of the plant, however, took 3-4 times longer than anticipated. Obtaining an accurate system model requires a very meticulous approach, and is a very time-consuming process. The collection of frequency response data was completed twice due to inaccurate first-run data. The initial data collected was determined unusable because the quality of the waveforms collected was poor. The poor data was due to inadequate power supplies used to drive the actuator. Eight power supplies were eventually used to drive the device to ensure the actuator was not forcing the voltage source to current-limit during high load conditions. The excess time used to collect data kept the project from reaching the digital implementation phase of the project.

Future Work:

The actuator used had a maximum load capacity of 25 pounds. This device was well suited for initial project research, but a higher capacity device should be obtained before actual platform construction begins. With system identification complete, advanced controller design ideas can be researched, simulated, and implemented. Assembly language code to run on the EMAC MicroPac 535 micro-controller development system, including code to generate the various applied input signals is still awaiting development. The digital implementation of various system controllers is also a key task awaiting completion.

Analysis / Conclusions:

System identification is not a task to be taken lightly. To complete an accurate frequency response of a given plant required many steps and an abundance of time. The data collection is tedious and the results must be interpreted properly to yield accurate results. It was quickly noted that undergraduate control classes work on many linear assumptions. These assumptions work well with some systems, but must be modified to adequately model other systems. Assuming plant non-linearities are not present is a risky assumption, especially considering the fact that non-linearities can render a system unstable. Non-linear parameters such as backlash severely effect system performance, but this project succeeded in demonstrating that increasing loop gain is one of many ways to minimize the effects of non-linearities in some systems.

Appendix

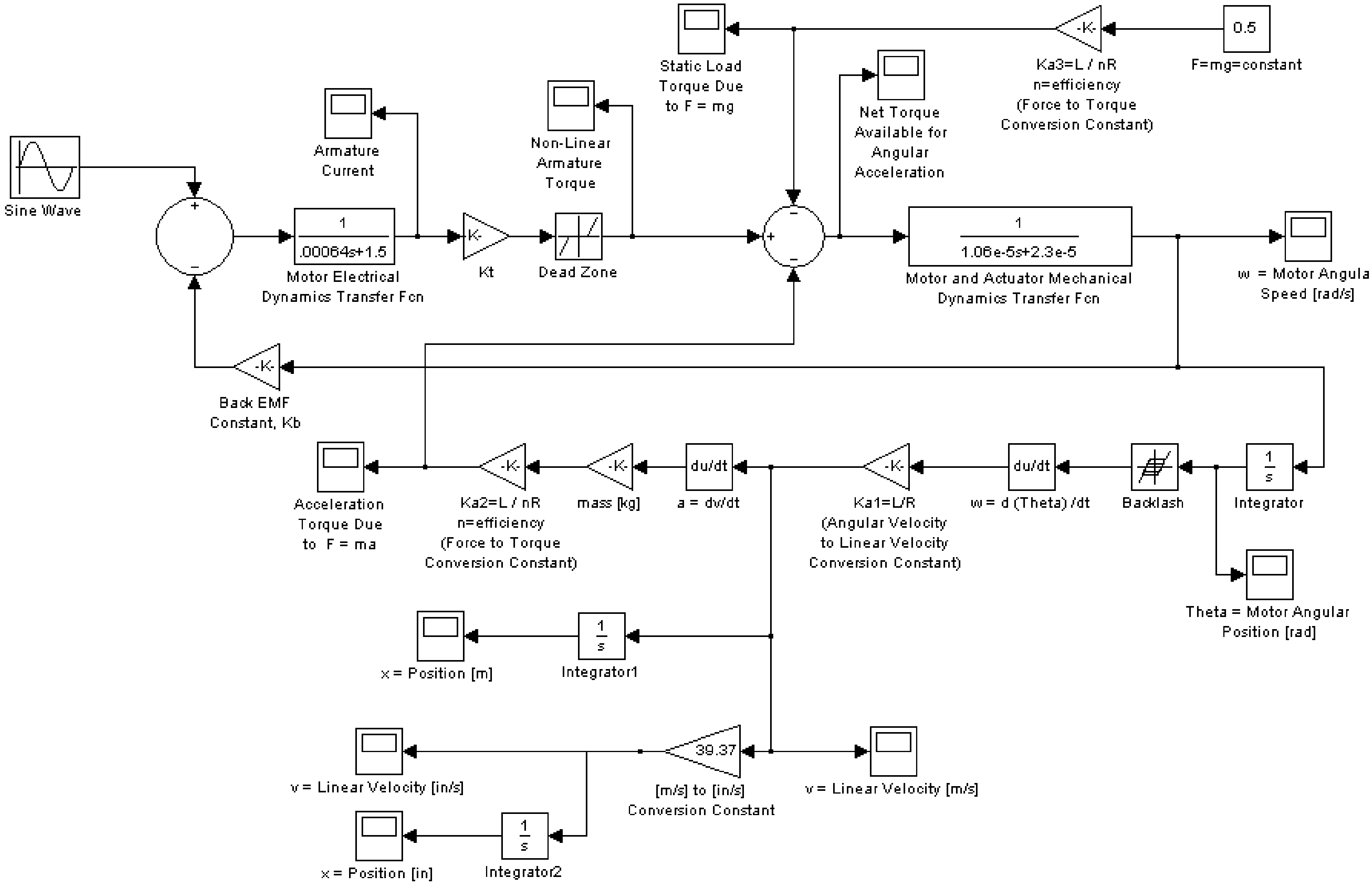


Figure 10 (expanded) Preliminary Simulink Model for the Warner Linear Actuator Including Non-Linear Effects

Figure 13 (expanded) Simplified System Simulink Model

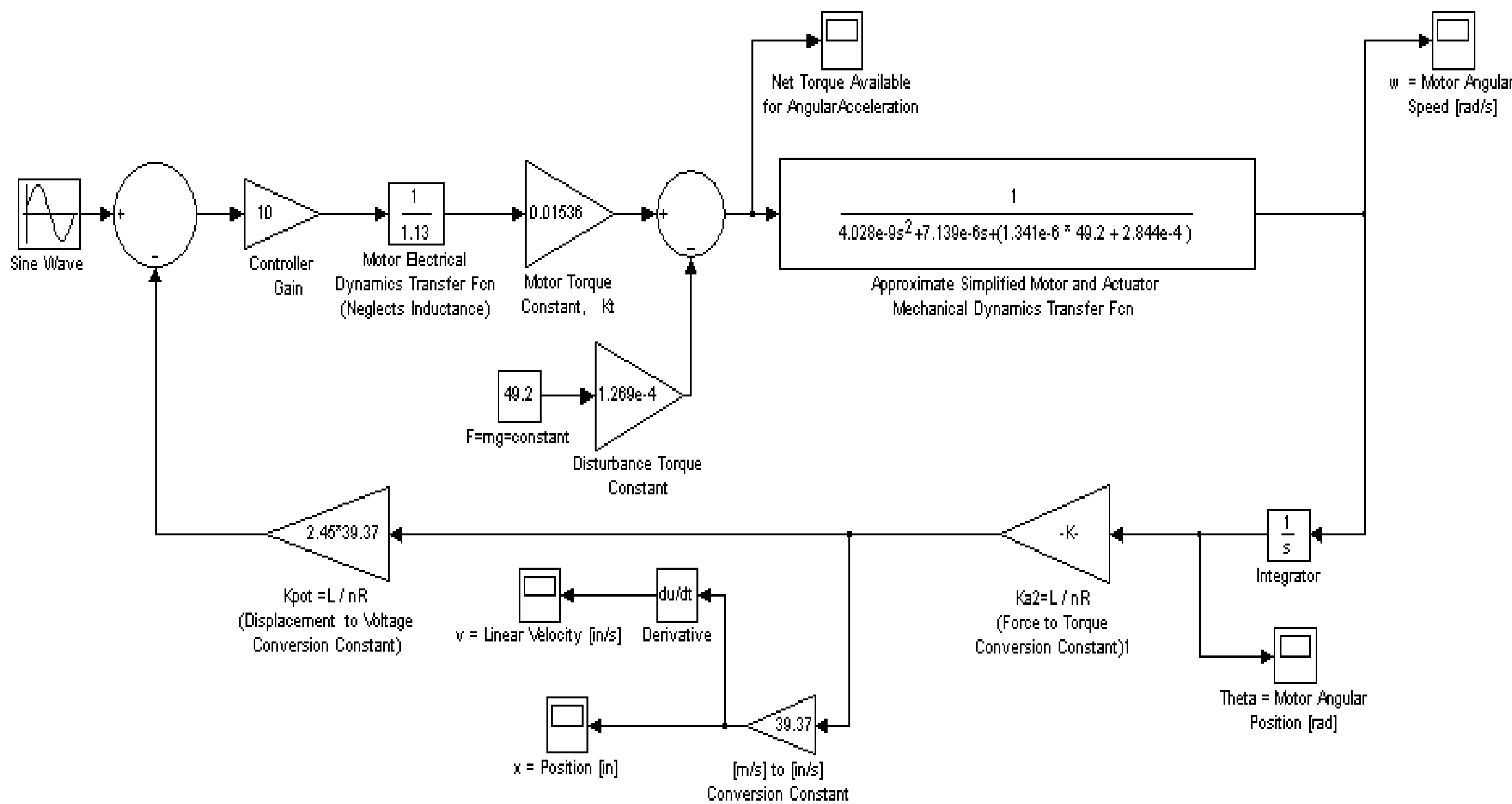


Figure 14 (expanded) Simplified Simulink Model Configured to Determine Phase Margin