Design and Control of a Long Stroke Fast Tool Servo

by

Marten F. Byl

B.S., Mechanical Engineering, University of California - Davis (1993)
M.S., Mechanical Engineering, University of California - Davis (1996)

Submitted to the Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2005

Author .....
Department of Mechanical Engineering
May 10, 2005

Certified by .....
David L. Trumper
Professor of Mechanical Engineering
Thesis Supervisor

Accepted by .....
Lallit Anand
Chairman, Department Committee on Graduate Students

Design and Control of a Long Stroke Fast Tool Servo

by

Marten F. Byl

Submitted to the Department of Mechanical Engineering
on May 10, 2005, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

In this thesis, I detail the design and control of a linear long stroke fast tool servo (FTS) with integral balance mass. The long stroke fast tool servo consists of an air bearing stage driven by a unique three phase oil cooled linear motor. The linear FTS has a travel range of 25 mm and is capable of 100 m/s^2 accelerations. The FTS is mounted to a T-base diamond turning machine (DTM). The FTS is attached to a hydrostatic bearing supported in-feed stage which is driven by a second linear motor. The in-feed stage is allowed to move in response to the FTS actuation forces and thus acts as an integral balance mass.

We have developed a unique control structure to control the position of both the FTS and the reaction mass. The FTS controller employs a conventional lead-lag inner loop, an adaptive feedforward cancelation (AFC) outer loop, and command pre-shifting. For the FTS controller, the AFC resonators are placed in the forward path which creates infinite gain at the resonator frequency. The controller for the hydrostatic stage consists of a conventional lead-lag control inner-loop and a base acceleration feedback controller. The acceleration feedback controller consists of a high-pass filter, a double integrator for phase compensation, and an array of AFC resonators. For the base acceleration controller, the AFC resonators are placed in the feedback path and thus act as narrow-frequency notch filters. The notch filters allow the hydrostatic stage/balance mass to move freely at the commanded trajectory harmonics thus attenuating the forces introduced into the DTM. The AFC control loops are designed using a new loop shaping perspective for AFC control.

In this thesis, we present two extensions to AFC control. The first extension called Oscillator Amplitude Control (OAC) is used to approximate the convergence characteristics of an AFC controller. The second extension termed Amplitude Modulated Adaptive Feedforward Cancelation (AMAFC) is designed to exactly cancel disturbances with a time varying amplitude.

Thesis Supervisor: David L. Trumper
Title: Professor of Mechanical Engineering

Acknowledgments

First, I would like to thank my advisor Professor David L. Trumper. Through my entire tenure here at M.I.T., Prof. Trumper has challenged me to be a more thoughtful researcher, student, and teacher. His clear insights into system theory, classical control systems, magnetic circuit design, and teaching are what made the successful completion of my doctoral studies possible. In particular the benchmark for teaching performance he set while instructing 2.003 is one that I do not expect to be exceeded.

I would also like to thank the members of my committee Prof. Martin Culpepper and Prof. Samir Nayfeh. While I spent far too little time interacting with Prof. Culpepper during my time here, his insightful comments on my research had a significant impact on the mechanical design of the FTS and the final form of the result presented here. I interacted with Prof. Nayfeh on a far more regular basis and would like to thank him for his encouragement and support throughout my tenure in the basement of Bldg. 35. I would never have made it through without your advice.

Special thanks go to Prof. David Gossard and Prof. David Hardt for allowing me to TA 2.003 the terms that they ran the class and for their cheerful and ongoing support. Similarly, I would like to thank Prof. Neville Hogan from whom I learned more than I ever thought I would want to know about modeling and analysis. I greatly enjoyed my time as a student in your classes and as a fellow instructor in 2.003 lab.

I have had the good fortune of having had my work supported by a great number of people. Dr. Stephen Ludwick's rotary fast tool servo and suggestion for future work formed the foundation on which my research is built. Joseph Calzaretta served as my mentor my first year here at M.I.T. and his background work with AFC control contributed greatly to the development of our loop shaping approach. Joseph Cattell provided most of the heavy lifting in the development of the OAC perspective of AFC control. Michael Liebman who in addition to being my tennis partner also designed and built the oil cooled linear motor used to actuate the FTS. David Otten supplied much of the C code used to implement the serial interface encoder.

Over the years many people have passed through the Precision Motion Control Laboratory and each has had an impact on me either providing useful advice or simply good companionship. They have included Aaron Mazzeo, Larry Hawe, David Cuff, Andrew Stein, Katie Lilienkamp, David Chargin, Augusto Barton, and Ming-chih Weng. Special thanks go to our Dutch visitors Mart and Twan who even in their short stay provided a great deal of entertainment. This entire experience has been shared with my fellow doctoral students Richard Montesanti and Xiaodong Lu, who have provided timely advice for tackling both technical and administrative hurdles. David Rodriguera, Maggie Sullivan, Denise Moody, and Laura Zaganjori have cheerfully addressed all the administrative needs of the Lab. Thank you to the rest of the basement 35 crowd who are too numerous to name.

A personal thanks goes out to my M.I.T. friends who occasionally managed to pry me out of lab or off of my couch. Marty Vona has been my workout, cooking, and hiking buddy in addition to providing help when ever I need programming or electronics advice. Paul Konkola and Carl Gang Chen did their level best to make sure I got a sunburn at least once a year.

Finally, I would like to thank my family. My sister Carolyn and her husband Chris, my sister Jennifer and her husband and children Harry, Emily, Carrie, and James, and my parents Fred and Nancy. I can only hope that someday I can support all of you as you have supported me.

This research was partially funded by NSF grant DMI-9908325

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

This thesis describes the design, development, and control of a unique long stroke fast tool servo with integral balance mass. This project started for me in August 1999, when I joined Joseph Calzaretta in his efforts to optimize and improve the rotary fast tool servo equipped diamond turning machine developed by Stephen Ludwick for turning asymmetric spectacle lenses. From August 1999 to June 2000, we conducted a large number of cutting studies aimed at producing a lens that would be ready for coating with no additional fining required. The summer of 2000, I worked to develop an on-machine fining station that would lightly polish a cut lens so that all the required machining operations could be incorporated in a single machine. From June 2000 to December 2000, I worked on designs for a second generation rotary fast tool servo. In late 2000, Prof. Trumper purchased a Moore Nanotechnologies Nanotech 350 machine base to be the machining platform for all of his diamond turning research. The machine arrived early in 2001, at which point I spent the spring and summer of 2001 constructing and testing the machine wiring harness, dSpace instrumentation interface box, and establishing basic control of the machine spindle and slideways. During this time, we made the decision to switch to a linear fast tool servo design. In late 2001, we began to work in earnest on our loop shaping approach for controllers with Adaptive Feedforward Cancelation. From Sept. 2001 to May 2002, I worked

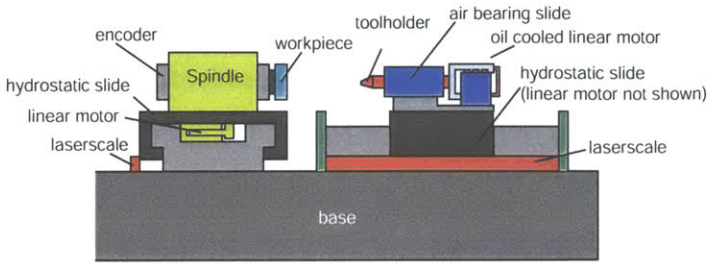


Figure 1-1: Schematic of the linear long stroke fast tool servo.

on the development of a unique liquid cooled voice coil motor design to actuate our proposed linear fast tool servo. In May 2002, the decision was made to progress with a less aggressive FST design that could be constructed using off the self components. During the summer of 2002, a prototype linear FST was designed and constructed using existing components. From Sept. 2002 to August 2003, I worked with Joseph Cattell to test our loop shaping techniques for tuning AFC resonator and develop an oscillator amplitude control perspective of AFC control. Sept. 2003 to May 2004 saw the redesign of our prototype linear fast tool servo. Construction of the redesigned FTS occurred from May 2004 through July 2004. The encoder serial interface was developed in August and September 2004. Basic control of the FTS and reaction mass stage occurred in October 2004. Feedforward and AFC control were applied to the FTS in Nov. 2004. December 2004 saw the implementation of base acceleration feedback on the reaction mass. Testing of AMAFC control occurred in Jan 2005. Thesis preparation occurred from Jan. 2005 through April 2005.

1.2 Thesis Overview

In this section, we will provide a brief overview the key work and contributions in this thesis.

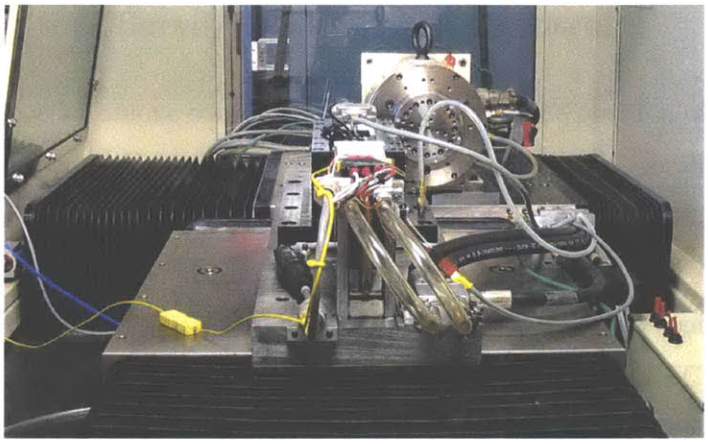


Figure 1-2: Rear view photo the prototype linear fast tool servo mounted on the Moore Nanotechnologies machine base.

1.2.1 Linear Long Stroke Fast Tool Servo with Integrated Balance Mass

Our linear long stroke fast tool servo (FTS) is specifically designed to produce asymmetric optics. The innovative portion of our design is the use of an integral balance mass to attenuate reaction force in the machine base. The prototype FTS, shown in schematic in Figure 1-1 and in a photo in Figure 1-2, consists of a 2"x2" air bearing stage actuated by an oil cooled linear motor designed and built by Michael Liebman [46]. The fast tool servo has 25 mm travel and is capable of 100 m/s^2 accelerations. The FTS is mounted on the in-feed axis of a T-base diamond turning machine (DTM). The in-feed axis of the DTM is allowed to move in response to the FTS actuation forces, thus reducing the reaction forces in the base. The in-feed axis consists of a hydrostatic stage actuated by a conventional linear motor. The T-based diamond turning machine is completed by a second linear motor driven hydrostatic stage carrying an air bearing spindle. Position feedback is provided by glass-scale encoders on all of the linear axis and an incremental encoder on the spindle. Since the position of the FTS tooltip is a function of both the measured FTS position and the reaction mass position, it is critical to accurately control both the FTS and the reaction mass.

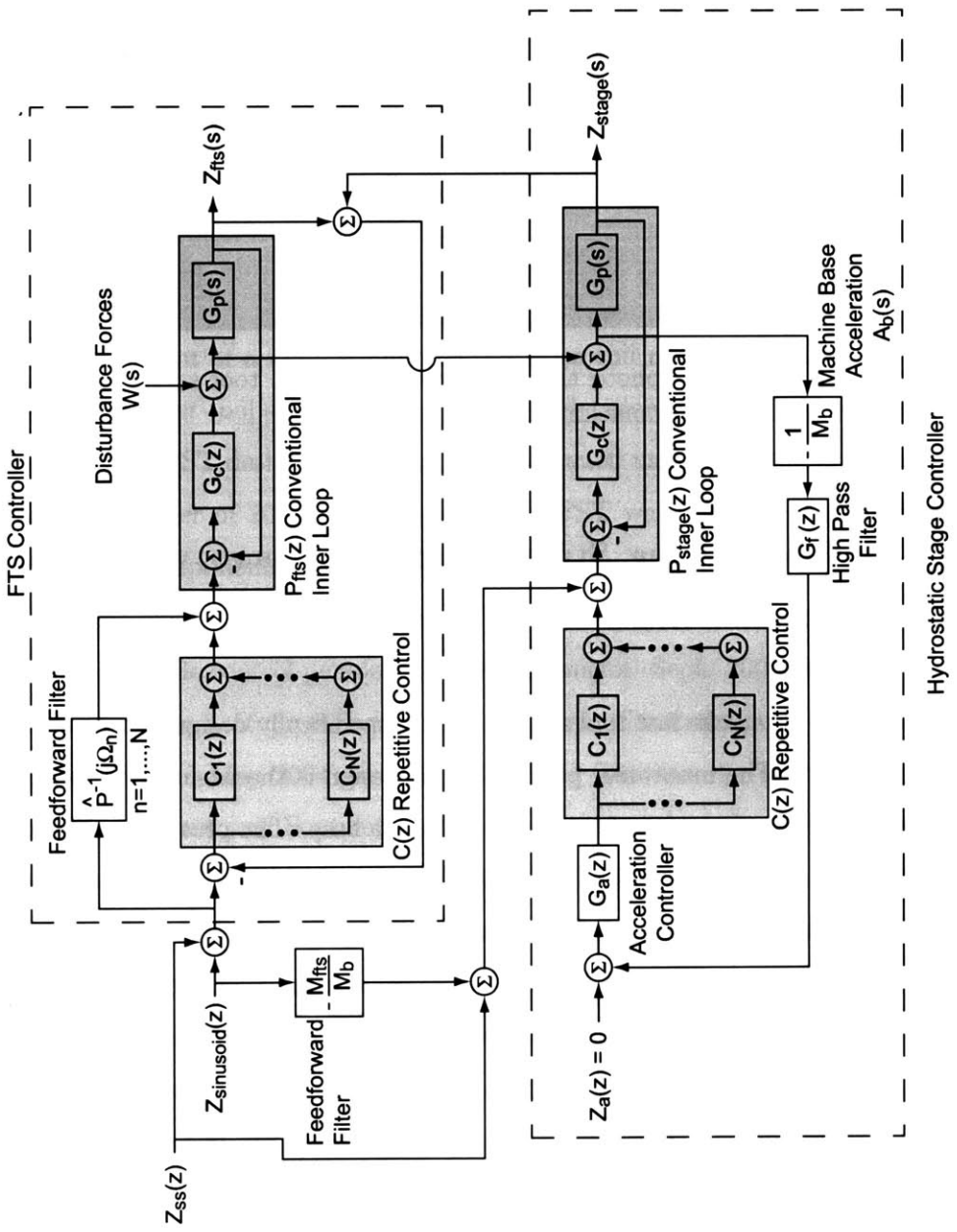


Figure 1-3: Block diagram of the overall control structure for our prototype fast tool servo with integral balance mass.

1.2.2 Fast tool Servo/Reaction Mass control

We have developed a unique control structure to control the position of both the FTS and the reaction mass. Figure 1-3 shows a block diagram of the overall FTS/reaction mass control structure. The FTS controller employs a conventional lead-lag inner loop, an adaptive feedforward cancelation (AFC) outer loop, and command preshifting. The FTS controller is coupled to the hydrostatic stage controller through the mixing of the measured positions. For the FTS controller, the AFC resonators are place in the forward path which creates infinite gain, thus perfect command following, at the resonator frequency (note: most asymmetric shapes can be described by a summation of sinusoids and thus the FTS trajectory becomes a function of the harmonics of the DTM spindle speed). The controller for the hydrostatic stage consists of a conventional lead-lag position control inner-loop and a base acceleration feedback controller. The acceleration feedback controller consists of a high-pass filter, a double integrator for phase compensation, and an array of AFC resonators. For the base acceleration controller, the AFC resonators are place in the feedback path for the position controller and thus act as narrow frequency notch filters. The hydrostatic stage is disturbed by the FTS actuation forces. The notch filters allow the hydrostatic stage/balance mass to move freely at the commanded trajectory harmonics thus attenuating the forces introduced into the DTM. The conventional lead-lag controllers are tuned using classical loop shaping techniques, while the AFC control loops are designed using our new loop shaping perspective for AFC control.

1.2.3 Adaptive Feedforward Cancelation from a Loop Shaping Perspective

Figure 1-4 shows the canonical form of an AFC resonator in continuous time. This structure is equivalent to the following linear time invariant (LTI) transfer function:

$$\frac{Y(s)}{X(s)} \equiv C_n(s) = g_n \frac{s\cos\phi_n + \omega_n \sin\phi_n}{s^2 + \omega_n^2}. \tag{1.1}$$

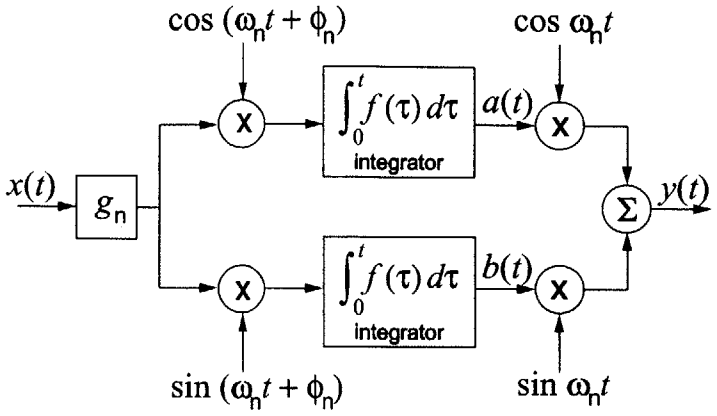


Figure 1-4: Resonator structure which forms the core of the AFC controller.

This transfer function places a single zero along the real axis and a pair of poles directly on the imaginary axis of the complex plane. With the poles directly on the imaginary axis, this transfer function has infinite magnitude and an instantaneous -180^o phase drop when $\omega = \omega_n$. From a loop shaping perspective, the phase margin of any system including an AFC resonator is maximized when the -180^o phase drop is centered on 0^o. This is accomplished when

$$\phi_n = \angle P(j\omega_n) \tag{1.2}$$

where $P(s)$ is the transfer function of all other elements in the system loop transmission. The controller gain can similarly be selected from a loop shaping perspective by ensuring that the magnitude of the loop transmission is kept below 1 (0 dB) between the AFC resonant peaks. We extend this direct approach to systems with multiple AFC resonators. In this thesis, we will present a comprehensive summary of our loop shaping perspective. There are two weakness of this approach. First, this analysis does not predict how long it takes an AFC compensated system to cancel a disturbance. Second, this AFC structure cannot completely cancel out disturbances with a time varying component.

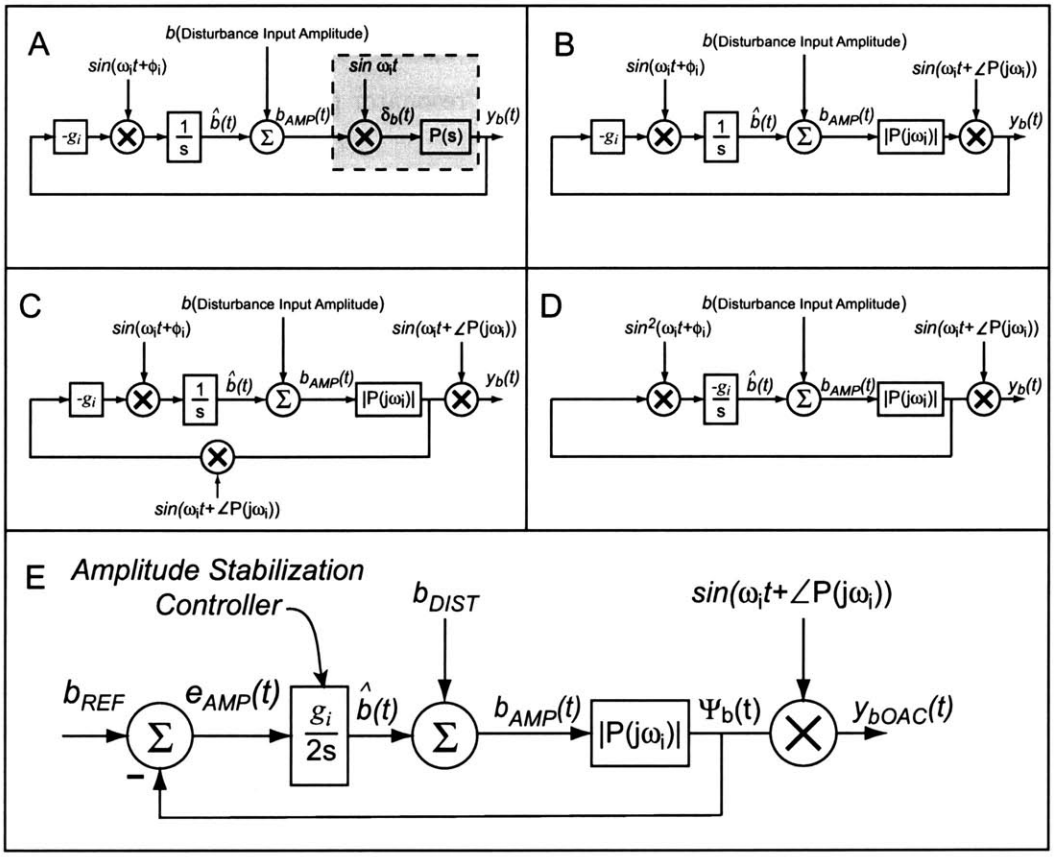


Figure 1-5: Simplification of the closed-loop block diagram for the sine channel of the single resonator AFC system. The reference signal $r_b(t)$ has been removed from frames A-D for simplicity.

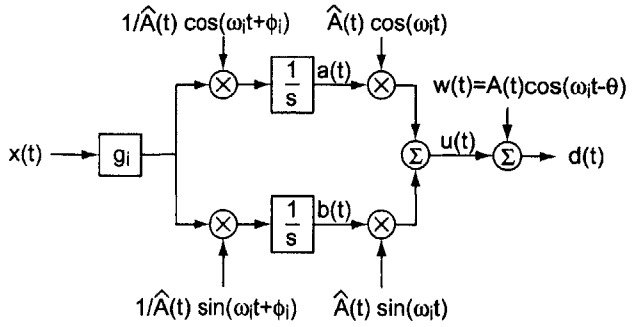


Figure 1-6: Block diagram of an AMAFC resonator plus disturbance subsystem.

1.2.4 OAC approximation and AMAFC control

In order to determine the convergence characteristics of an AFC resonator, we apply a classic oscillator stabilization technique called Oscillator Amplitude Control (OAC). Following up on the thesis work of Joseph Cattell [15], we can view an AFC resonator as a control structure which detects the amplitude component of a disturbance at the frequency ω_n. Once the amplitude is estimated, the AFC resonator produces a signal to exactly cancel the disturbance. This makes an AFC resonator, an oscillator with a control amplitude output. To determine, the stability and convergence characteristics of the amplitude controller, we approximate that the sine and cosine channels are independent and reduce the AFC system block diagram as shown in Figure 1-5. This simplified system has an amplitude convergence time constant of

$$\tau_i = \frac{2}{g_i|P(j\omega_i)|} \text{ sec.} \tag{1.3}$$

The OAC approximation shows how important the proper selection of g_i can be on system performance.

While an AFC controller can exactly cancel out a disturbance with a constant amplitude, it will always have some residual error if the amplitude of the disturbance varies with time (note: this error is very small if the magnitude varies slowly). We have applied the Internal Model Principle (IMP) to develop an extension of AFC control which we call Amplitude Modulated Adaptive Feedforward Cancela-

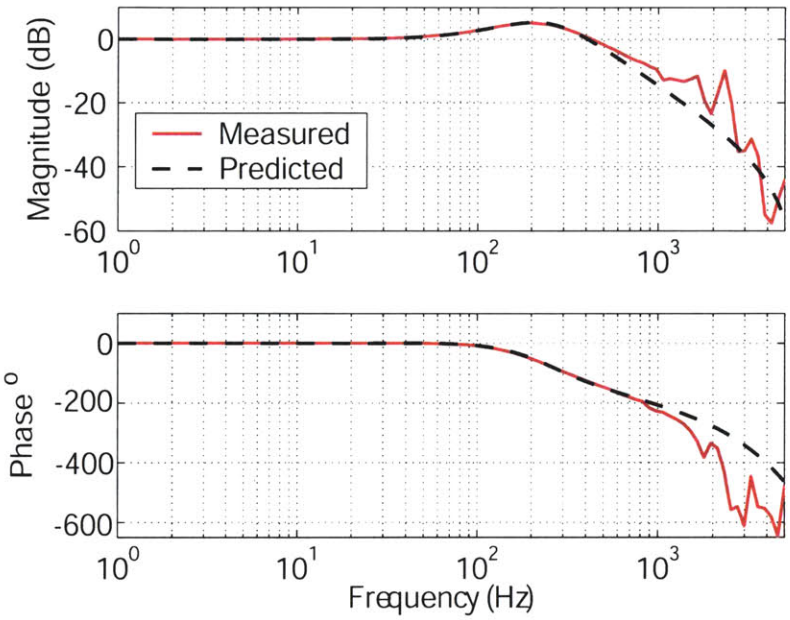


Figure 1-7: Experimental conventional fast tool servo frequency response. The dashed line indicates the predicted response.

tion (AMAFC). For AMAFC control, we make the assumption that we have some knowledge about the time varying characteristics of the disturbance signal. If the magnitude of the disturbance has the form $A(t)$, we can build this knowledge into our AFC resonator structure as an estimate $\hat{A}(t)$. This is shown in Figure 1-6. If $\hat{A}(t) = A(t)$, the AMAFC controller will completely cancel the disturbance.

1.2.5 Overview of Results

Our prototype fast tool servo achieves an exceptional closed-loop bandwidth of 540 Hz (this is twice the bandwidth of comparable long stroke FTS systems). The FTS bandwidth is limited by the first structural resonance at 1600 Hz and the two unit delay associated with the position measurement. Figure 1-7 shows the measured and predicted closed-loop frequency response.

Using just the conventional inner loop controller, the FTS has a maximum following error of 42%. The introduction of command pre-shifting and AFC control greatly reduce the following error by incorporating knowledge of the plant frequency

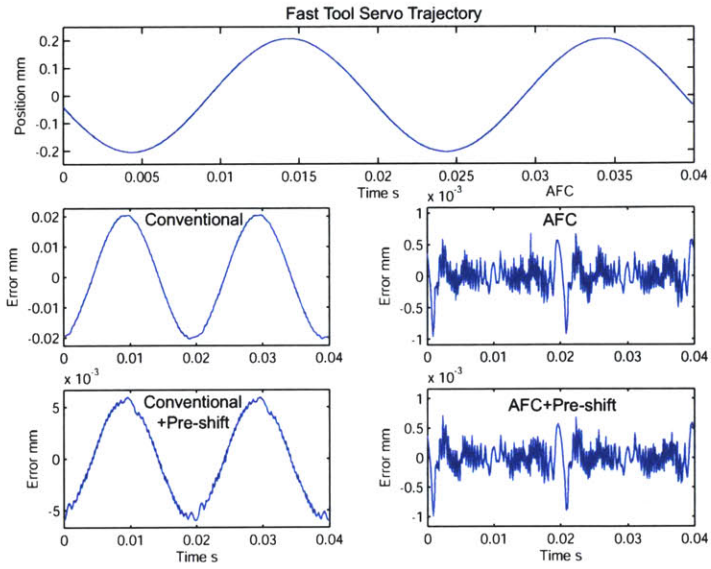


Figure 1-8: Measured fast tool servo following error to a single 50 Hz 0.4 mm pk-pk position command under conventional control, conventional control with command pre-shifting, AFC control, and AFC control with pre-shifting.

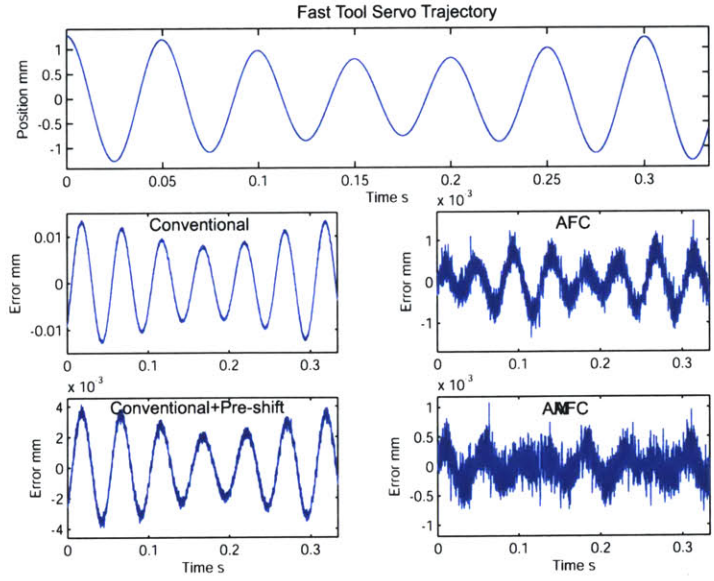


Figure 1-9: Measured fast tool servo following error to a single 20 Hz position command with 3 Hz modulation under conventional control, conventional control with command pre-shifting, AFC control, and AMAFC control.

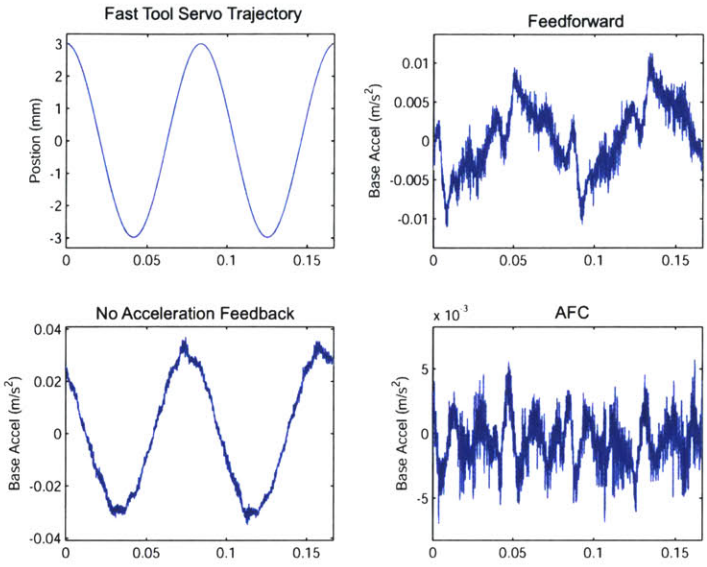


Figure 1-10: Measured base acceleration with no feedback, feedforward, and single resonator AFC control. The fast tool servo trajectory is a single harmonic 12 Hz 6 mm pk-pk sinusoid.

response and information from previous passes to correct the trajectory input to the conventional inner loop. Figure 1-8 shows the measured FTS following error to a 50 Hz 0.4 mm peak-to-peak sinusoidal trajectory. Command pre-shifting reduces the peak-to-peak following error from 103 μm to 31 μm. The introduction of AFC control further reduces the following error to 1.7 μm pk-pk. Similarly the rms following error for conventional control, command pre-shifting, and AFC control is 14.4 μm, 4 μm, and 0.25 μm respectively. Our proposed AMAFC controller was less successful than anticipated offering only a small reduction in the overall following error. Figure 1-9 shows the measured following error to 20 Hz trajectory with a 3 Hz amplitude modulation. The FTS has a peak-to-peak following error of 15.2 μm under conventional control, 3 μm under AFC control, and 2.1 μm under AMAFC control.

The integral balance mass successfully reduced the disturbance forces introduced into the DTM base. Figure 1-10 shows the measured DTM base acceleration with no feedback, feedforward control, and AFC control. The fast tool servo trajectory is a single harmonic 12 Hz 6 mm pk-pk sinusoid with a peak acceleration of 1.72 g.

The peak-to-peak base acceleration is 0.072 m/s^2 without acceleration feedback, 0.025 m/s^2 with feedforward control, and 0.015 m/s^2 with AFC control. More importantly the acceleration component at 12 Hz is 0.028 m/s^2 pk-pk without feedback, 0.0049 m/s^2 with feedforward control, and 0.000327 m/s^2 with AFC control. Thus the use of a balance mass has reduced the base reaction force introduced by the FTS at the commanded frequency by a factor of approximately 100.

1.3 Thesis Contributions

The primary contributions of this thesis are:

- The development of a long stroke fast tool servo with integral balance mass.
- The development of a unique control structure combining the fast tool servo position control, balance mass position control, and base acceleration feedback.
- The development of a novel loop shaping perspective for tuning control systems with multiple adaptive feedforward cancelation resonators.
- The application and demonstration of our loop shaping perspective for tuning adaptive feedforward cancelation control systems to both the fast tool servo position control and base acceleration feedback.
- Correctly applying the oscillator amplitude control perspective to both channels of an adaptive feedforward cancelation resonator and demonstrating the limitations of this approach.
- The development of an extension to adaptive feedforward cancelation called amplitude modulated adaptive feedforward cancelation meant to improve tracking performance for sinusoidal trajectories with time varying amplitude.
- The application and demonstration of amplitude modulated adaptive feedforward cancelation.

1.4 Thesis Organization

The chapters in this thesis are organized as follows:

- Chapter 1 contains the background of this thesis, an overview of the thesis contents and results, and reviews the design of existing fast tool servos and related actuators.
- Chapter 2 reviews our fast tool servo design process and contains a detailed description of our prototype fast tool servo.
- Chapter 3 describes and reviews the control structures used to control fast tool servos and reject periodic disturbance forces.
- Chapter 4 is a detailed presentation of our loop shaping perspective for tuning controllers with AFC control. Chapter 4 also presents our proposed AMAFC control structure.
- Chapter 5 includes a detailed analysis of AFC control from an oscillator amplitude control perspective.
- Chapter 6 describes the application of the developed control theory to the prototype diamond turning machine.
- Chapter 7 validates the performance of the developed control theory by experimental results.
- Chapter 8 concludes this thesis with a summary of our results and suggestions for further work.

1.5 Asymmetric Turning

There are a wide array of products which require the precision production of asymmetric parts. These products include pistons, cylinder bores, crank shafts, engine camshafts, and most relevant for this project, the production of asymmetric optics.

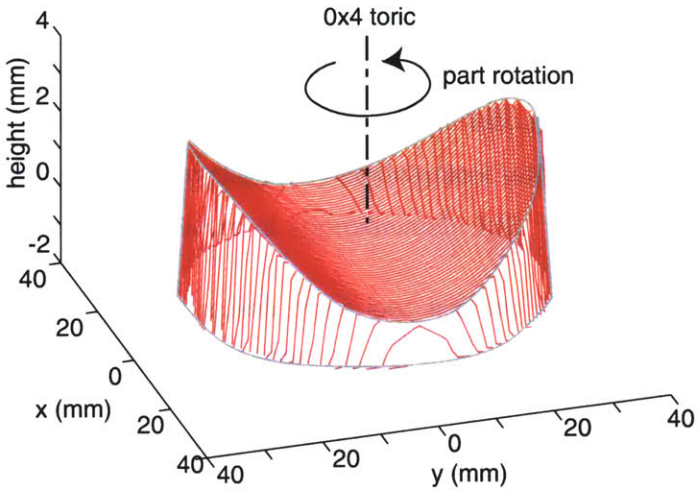


Figure 1-11: Typical toric spectacle lens.

Figure 1-11 shows the shape of a typical toric lens used to correct astigmatism. In general, these parts are either cast/molded directly to form (formed parts may require some fining operations) or machined into the desired shape using a combination of turning, grinding, and polishing.

In the case of spectacle lenses (the original motivation for this project), the traditional fabrication method for an asymmetric lens was to rough (typically in a grinding operation) the asymmetric shape into a rough shaped lens blank and then progressively polish the lens to the desired finish using a hard lap formed to the desired asymmetric shape. This lens forming method has several production drawbacks. First, to produce only the most commonly required asymmetric optics, the production facility must have a dedicated tooling library of several hundred hard laps. Second, the polishing process is a multi-stage, time intensive process. Lastly, the polishing process inevitably produces a lens with some form error, the magnitude of the error growing with the size of the lens asymmetry. These production drawbacks have driven the optical industry to explore alternate methods to form the asymmetric lens directly with the ultimate goal of completely eliminating the polishing process.

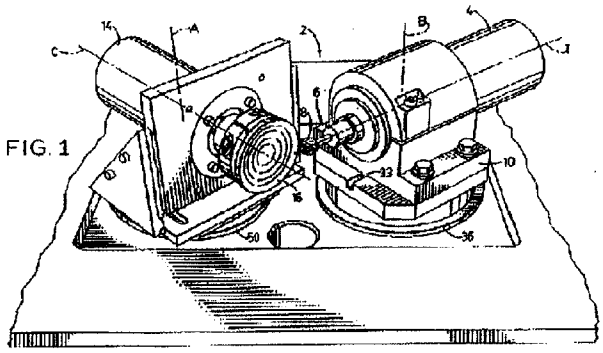


Figure 1-12: Diagram of LOH Optical Machinery, Inc. computer controlled lens surfacer from U.S. Patent 5,231,587.

A general shift occurred in the consumer optics industry with the introduction of a new machine by LOH Optical Machinery, Inc [48]. A diagram of the machine concept from U.S. Patent 5,231,587 is shown in Figure 1-12. This machine incorporates a tool carriage with a rapidly revolving tool (10,000+ rpm), a spindle which slowly revolves ($\approx$ 60 rpm) the workpiece, and a machine carriage capable of moving the tool in three dimensions relative to the workpiece. This machine, commercially available as the LOH V75 with improved tool geometry, was the first machine capable of directly producing lenses ready for coating at a commercially viable rate. The LOH V75 is capable of producing lenses with up to 10 mm of asymmetry at a nominal production rate of 60 lens/hr. A similar machine configuration is used in the Gerber-Coburn SGX family of surface generators [32]. While the SGX family of machines quickly produces good quality rough lenses, these lenses are not ready for coating and thus require a small amount of polishing. To eliminate the need for a library of hard laps, Gerber-Coburn produce disposable polystyrene laps in parallel with the rough lens.

The machines from LOH Optical Machinery, Inc. and Gerber-Coburn Inc. are optimized for the spectacle industry and cannot produce parts of the shape and quality desired by other branches of the optical industry. The most common method used to generate complex optical shapes is to diamond turn the optics on an FTS-equipped

diamond turning machine.

1.5.1 Fast tool servos

There are two main reasons to implement an FTS on a diamond turning machine. The first is to cancel out repetitive errors that are introduced into a part during the machining process (potential sources of error are external disturbances, resonances in the turning machine structure, spindle/part imbalance, or bearing noise). The second main reason to use an FTS on a diamond turning machine is to machine complex geometries into a workpiece. We categorize FTS for this purpose into three categories:

1. Short stroke (displacements less than 100 μm)
2. Intermediate Stroke (displacements between 100 μm and 1 mm)
3. Long stoke (displacement greater than 1 mm).

Table 1.1 list the associated organization, actuator type, displacement and bandwidth for a number of error cancellation, short stroke, intermediate stroke, and long stroke FTSs. As a general rule, FTSs trade displacement for bandwidth.

Short Stroke Fast Tool Servos

As we can see from Table 1.1, the short stroke fast tool servo category is dominated by piezoelectric driven fast tool servos. There are a number of reasons for this. First, piezoelectric stacks offer high stiffness, high bandwidth, high acceleration and nanometer resolution position control. Secondly since, good piezoelectric stacks are available commercially, researchers can focus their efforts on the mechanical design of the FTS, the FTS controller strategy, and the trajectory generation as opposed to specialized actuator design. Typically for piezoelectric FTS, a capacitance gauge is used to measure tool displacement. Most piezoelectric FTS use flexure bearings.

Figure 1-13 shows a number of common configurations for piezoelectric fast tool servos. In Figure 1-13-A, we see a configuration (adapted from [69]) where the piezoelectric axis and the FTS axis are parallel but not collinear. This configuration has the advantage that capacitance sensor may easily be placed collinear with the FTS

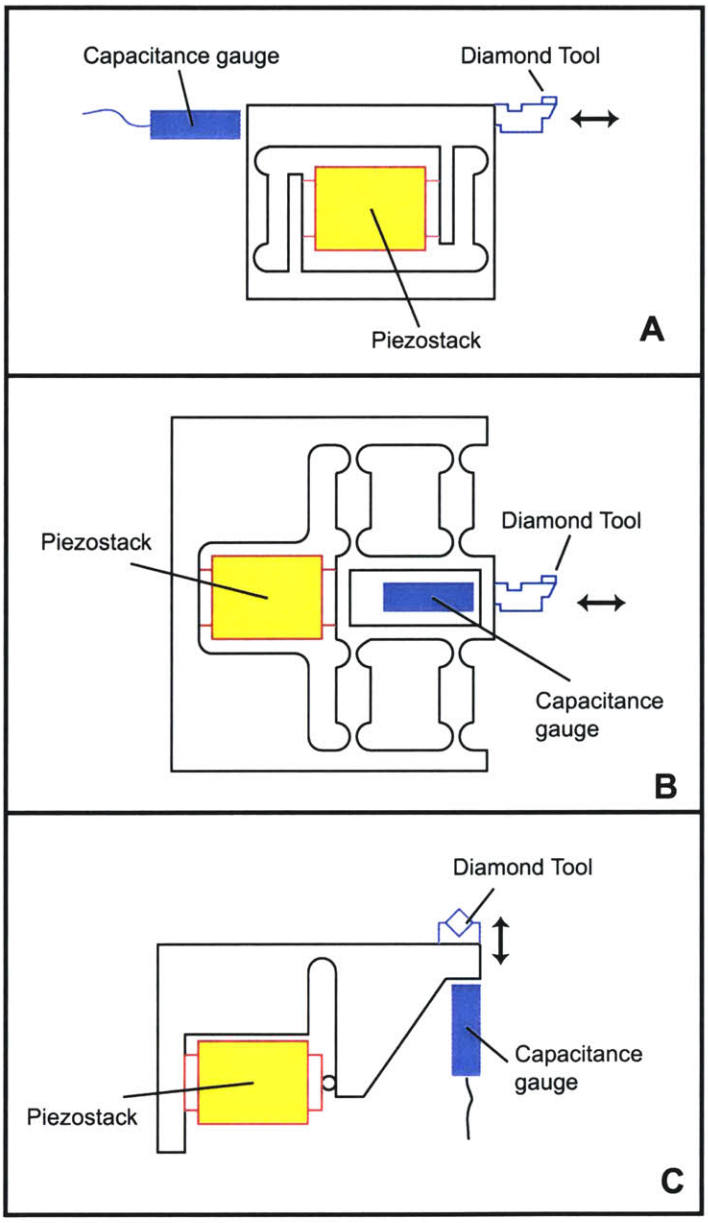


Figure 1-13: Typical configurations for piezoelectric fast tool servos. A adapted from [69], B adapted from [102], and C adapted from [74]

Error Cancellation				
Organization/Name	**Actuator**	**Stroke (μm)**	**Bandwidth**	**Ref.**
Lawrence Livermore Lab., LODTM	Piezo	1.27	100 Hz	[23] [72]
Georgia Tech.	Piezo	60	350 Hz	[20]
University of British Columbia	Piezo	40	N/A	[104]
Short Stroke				
Hitachi	Piezo	20	N/A	[65]
Agency of Industrial Science and Technology, Japan	Piezo	5	200	[69]
North Carolina State	Piezo	20	2 kHz	[24]
Fraunhofer-Institute, FTS 1	Piezo	35	950 Hz	[98]
Fraunhofer-Institute, FTS 2	Piezo	30	1.5-2 kHz	[98]
University of Illinois Urbana-Champaign	Piezo	50	200 Hz	[74]
University of Illinois Chicago, 3D Stage	Piezo	3.5x3	200 Hz x3	[86]
M.I.T., Short stroke Rotary FTS	Lorentz	50	2 kHz	[60]
M.I.T., Electomagnetic FTS	Var. Reluct.	50	10 kHz	[51], [52]
University of British Columbia	Piezo	38	200 Hz	[102]
Int. for Adv. Engineering, So. Korea	Piezo	7.5	100 Hz	[42]
Intermediate Stroke				
North Carolina State, Raleigh, MAC-100	Piezo	100	100 Hz	[21], [25]
Kinetic Ceramics, Precision Lathe PZT	Piezo	100	1 kHz	[43]
Stanford University, hydraulic FTS	hydraulic	180	3 kHz	[93], [94]
North Carolina State, Raleigh	magnetic servo	240/1000	10/300 Hz	[36], [87]
Long Stroke				
University of Illinois Urbana-Champaign	VCM	6.5 mm (min.)	? Hz	[75], [7]
University of Illinois Urbana-Champaign	Hydraulic	50 mm (min.)	100 Hz	[95]
University of Illinois Urbana-Champaign	Electro-hydraulic	10 mm (min.)	100 Hz	[41]
Fraunhofer-Institute, Hybrid FTS	Piezo & Linear	2 mm	85 Hz	[100]
Gerber-Coburn, Lensmaker XRT	Servo	25 mm	50 Hz	[32]
M.I.T., Rotary FTS	Servo	25 mm	230 Hz	[54]
Fraunhofer-Institute, Aerostatic FTS	Linear	16 mm	240 Hz	[99], [98]

Table 1.1: Table of error cancellation, short stroke, and intermediate FTS.

tool tip but offsetting the piezoelectric stack results in some internal torques. An additional advantage of this structure is with proper placement of the piezo stack, the designer can amplify or reduce the displacement of the piezo stack. Figure 1-13-B shows a configuration where the tool tip, capacitance sensor, and piezoelectric actuator are all collinear (adapted from [102]). The advantage of this structure is that is minimizes the internal torques and places the capacitance gauge behind the tool tip in most designs but properly fixturing the capacitance gauge can be a challenge. Figure 1-13-C shows a configuration where the piezoelectric stack and tool tip are neither collinear nor parallel (adapted from [74]. This type of structure is typically used when space in the direction of the tool tip motion is limited (boring bars for example). The capacitance gauge may be placed either inline with the tool tip or the piezoelectric stack. This structure has the disadvantage that the tool tip has a rotary motion. Since by definition these are short stroke FTS, the lateral component of the

motion is ignored.

Two of the more innovative short stroke fast tool servo designs have been developed here at M.I.T. in the Precision Motion Control Laboratory. Figure 1-14 shows Richard Montesanti's short stroke rotary fast tool servo [61], [60], and [62]. The fast tool servo axis is guided by eight over-constrained flexure, four above the tool and four below. Items **106** and **108** in Figure 1-15 are two of the eight over-constrained flexures. The rotary axis is driven by a commercially available moving magnet Lorentz actuator (**256** Figure 1-14). The FTS position is measured using two capacitance gauges (**280** Figure 1-14). This FTS has a full stroke of 50 μm and a bandwidth of 2 kHz. The performance of this FTS is limited by the first torsional resonance of the tool axis, the same resonance which limited the performance of Stephen Ludwick's long stroke rotary fast tool servo [54] discussed later.

One of the limitations of piezoelectric actuators is that when they undergo deformation there are significant mechanical and electrical hysteresis losses which heat the actuator. In high bandwidth applications this is a significant issue. In addition to stack heating, piezoelectric FTS are limited by the structural resonance of the piezoelectric stack. To avoid these issues Xiaodong Lu designed a high bandwith linear short stroke FTS which is driven by a normal-stress variable reluctance actuator [51] and [52]. Figure 1-16 shows a schematic of this FTS. This FTS has a 50 μm stroke up to 1 kHz and a closed loop -3 dB bandwidth of 10 kHz. The maximum acceleration is 160 g's measured at 3 kHz.

Intermediate Stroke Fast Tool Servos

Intermediate stroke fast tool servos are essentially transition designs. In the case of the piezoelectric fast tool servos, one from North Carolina State [21], [25] and one from Kinetic Ceramics [43], the FTS designs are essentially short stroke fast tool servos pushed to their size limits. The MAC-100 is a linear design of the style shown in Figure 1-13-B, with a large piezoelectric stack. The MAC-100 design uses a single O-ring for the FTS bearing. The Precision Lathe PZT [43] uses a design of the form shown in Figure 1-13-C. The Precision Lathe PZT mechanism is designed such that the tool tip travel is greater than the piezoelectric stack displacement.

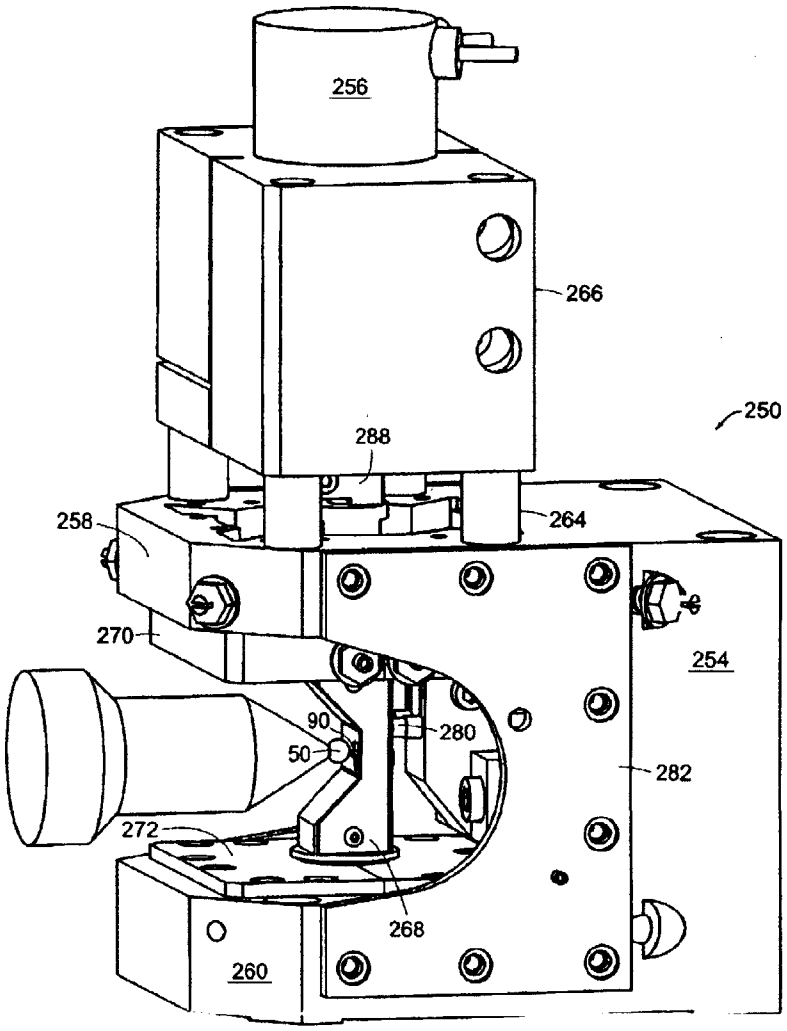


Figure 1-14: High bandwidth short stroke rotary fast tool servo with lorenz actuator (from U.S. Patent application #20040035266 [62]).

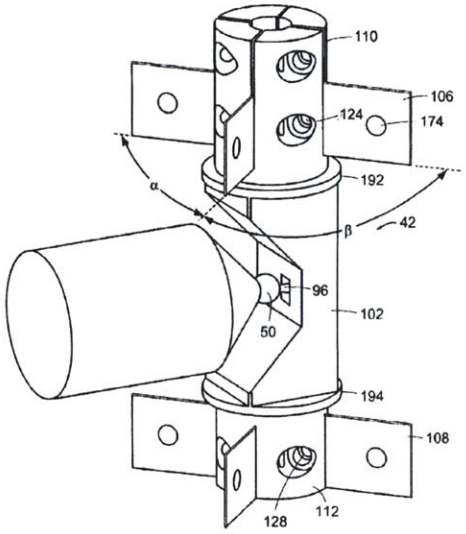


Figure 1-15: Tool axis for the high bandwidth short stroke rotary fast tool servo with Lorentz actuator (from U.S. Patent application #20040035266 [62]).

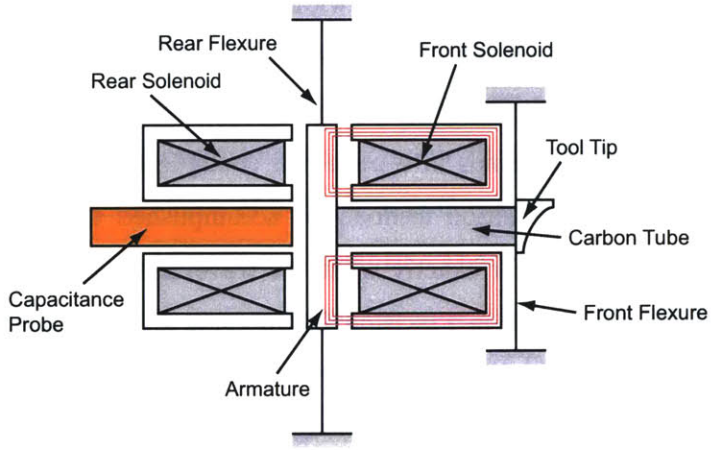


Figure 1-16: Schematic of a short stroke variable reluctance fast tool servo (adapted from [51] and [52]).

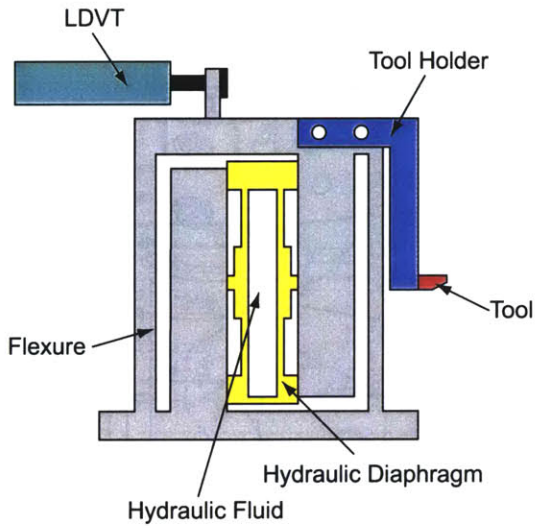


Figure 1-17: Schematic of intermediate stroke hydraulic fast tool servo (adapted from [94]).

Figure 1-17 shows a schematic of the Stanford hydraulic FTS. As we can see, this fast tool servo is of the same style as Figure 1-13-A, where the piezoelectric actuator has been replaced by a hydraulic diaphragm and the short range capacitance gauge has been replace with a Linear Variable Differential Transformer (LVDT). This actuator has a 180 μm stroke with a quoted bandwidth of 3 kHz but since the LDVT is quoted to have a bandwidth of 100 Hz this seems unlikely [94] and [93].

The magnetically levitated fast tool servo from North Carolina State is driven by a total of four E-coil solenoids in a push-pull configuration. The FTS is supported by a pair of flexures. Position feedback is accomplished using a heterodyne laser interferometer. It is unclear exactly what the travel and bandwidth of the fast tool servo are. In a brief paper in Precision Engineering [87], the travel and bandwidth are quoted to be 1 mm and 300 Hz respectively. In [36], the travel and bandwidth are stated to be 240 μm and 10 Hz although it appears that the travel limitation is due to the linearized model used in the FTS control loop. Lastly, while [35] does not provide travel data, it claims a fast tool servo bandwidth of 2.5 kHz.

Long Stroke Fast Tool Servos

For the purposes of this research, the FTS in the long stroke category are of the

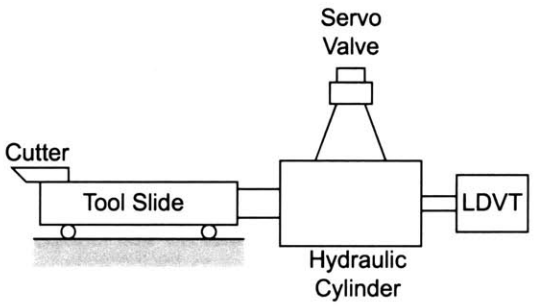


Figure 1-18: Schematic of a long stroke hydraulic fast tool servo (adapted from [95]).

most interest. There is limited data available for the two hydraulic FTS from the University of Illinois. Figure 1-18 shows a schematic of the 50 mm, 100 Hz hydraulic fast tool servo [95]. This is linear fast tool servo design with the tool axis running on roller bearings with a hydraulic actuator. Position feedback is accomplished using an LDVT. It is unclear if the second hydraulic fast tool servo is a different actuator or not, since no design details are provided in the cited reference [41]. No mention is made of reaction force management for either of these FTS which might explain the relatively large tracking error of 26 μm max and 6.8 μm RMS despite the use of feedforward and repetitive control.

More details are available for the two long stroke fast tool servos from the Fraunhofer-Institute. The Fraunhofer hybrid FTS combines a short stroke piezoelectric actuator (40 μm/1000 Hz) with a long stroke linear motor (2 mm/40 Hz). The piezoelectric actuator has a maximum force of 2400 N, a stiffness of 50 N/μm and a resonance frequency with a loaded mass close to 2000 Hz. The long stroke actuator consists of three moving magnet linear motors connected in series. The moving mass guides are parallel springs (flexures). The linear motor has a peak force of 900 N. The total tool movement is measured using a laser interferometer. The position of the short stroke piezoelectric actuator is monitored using a capacitance gauge. The speed of the linear motor is measured using a linear velocity transducer. Both the piezoelectric actuator and linear motor coils are water cooled. This is the first fast tool servo design which directly addresses the need to manage the fast tool servo reaction forces. In this fast

tool servo, the linear motor coils are also mounted on springs and allowed to move in reaction to the FTS actuation forces. This strategy was only partially successful since the 68 Hz resonance of the motor coil/spring system cannot be attenuated using closed loop control which resulted in a significant resonant peak at 68 Hz for any control loop with a bandwidth greater than 50 Hz (loops were closed as high as 84 Hz) [100].

The second long stroke fast tool servo from the Fraunhofer Institute has 16 mm of travel and a closed loop bandwidth of 100 Hz. The FTS is driven by a linear motor with two stationary coils and two moving magnets. The linear motor has a significant non-linear force constant dependent on the position of the magnets. The moving components are mounted on a synthetic fiber carriage. The FTS is supported by a 12 pad air bearing. The bearing surfaces on the moving mass are nickel coated aluminium bonded to the synthetic fiber frame. For position feedback, Weck tested two different linear scales supplied by Heidenhain. The first was the LIP 382 with a signal period of 128 nm and a resolution of 0.13 nm and a maximum speed of 0.06 m/s. With the LIP 382, the FTS displayed 1.4 nm of error but the low maximum speed limited the dynamic performance. The second linear scale was the LIP 403 with a signal period of 2 μm, a resolution of 2 nm, and a maximum velocity of 0.8 m/s (the desired maximum velocity was 2 m/s). With the LIP 403, the system noise rises to 10 nm. There is no attempt to manage the reaction forces in the fast tool servo design but Weck notes that the reaction forces must be attenuated for high acceleration profiles. Weck proposed that to manage reaction forces one could mount a second FTS on the machine operating in the opposite direction [99].

Figure 1-19 shows a schematic diagram of the voice coil actuated FTS from the University of Illinois at Urbana-Champaign. The FTS consist of a moving coil single phase linear motor directly driving the FTS tool axis. The FTS is supported by a Rulon-LR bearing and uses a laser position sensor with a resolution of 0.618 μm for position feedback. The voice coil motor is rated at 1670 N peak and 320 N continuous. The moving mass is 1.5 kg, which result in a maximum FTS acceleration of 100 g's. Neither reference for this FTS gives a maximum travel range but the travel is at

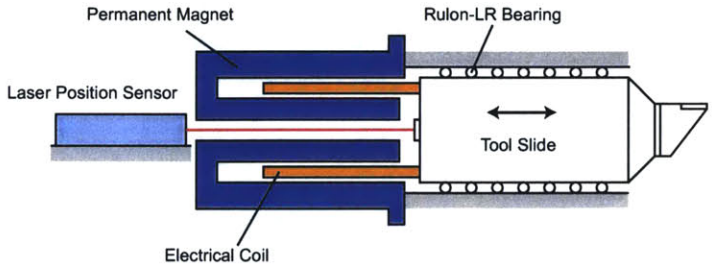


Figure 1-19: Schematic diagram of voice coil driven fast tool servo (adapted from [75]).

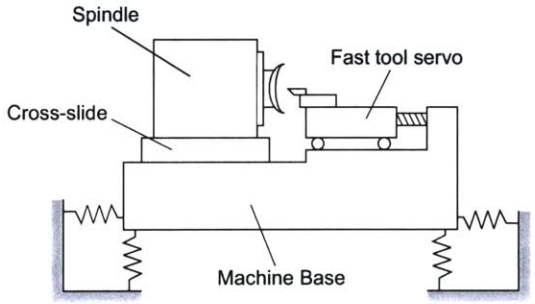


Figure 1-20: Schematic of the Lensmaker XRT.

least 6.5 mm, the magnitude of the cam trajectory used to generate data. The servo bandwidth is assumed to be 500 Hz or less based upon the 5 kHz controller sampling frequency. This FTS has a 100 μm pk-pk error following a cam trajectory with a tool travel of 6.5 mm, a peak acceleration of 9.8 m/s^2, and a maximum velocity of 0.6 m/s. No mention is made of compensating for the FTS reaction force. It should be noted that since the position controller for this FTS contained only proportional and derivative control, the steady state cutting force (31 N) resulted in a significant steady state following error [75] and [7].

The Lensmaker XRT, from Gerber-Coburn Optical [32], is a conventional turning machine equipped with a 25 mm travel 50 Hz bandwidth FTS. Figure 1-20 shows a simple schematic of the Lensmaker. The FTS on the Lensmaker is driven by a high-lead ballscrew and supported by a conventional roller bearing slideway. The Lensmaker spring mounts the machine frame to allow it to serve as a reaction mass. Thus

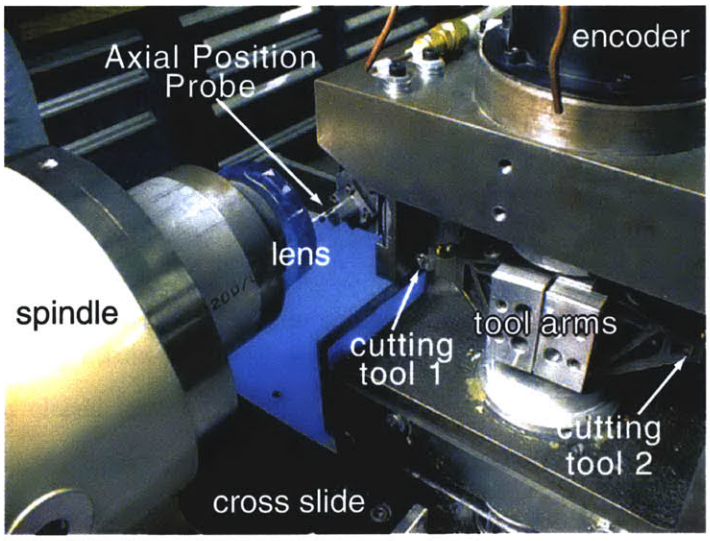


Figure 1-21: Photograph of the rotary fast tool servo.

in operation, the FTS introduces large reaction forces and motions in the machine base. The Lensmaker XRT is capable of producing small amplitude toric spectacle lens which require light polishing at a rate of 60 lens/hr, but production rate and part quality quickly degrade as the amplitude of the asymmetry increases. The Lensmaker XRT has been replaced by the DTL Generator. The DTL generator incorporates a 1000 lb granite base and a 6 g tool actuator. Further details of this machine are not available.

Stephen Ludwick, building on his experience of tuning and testing a Lensmaker XRT FTS, arrived at the unique solution of managing reaction forces by designing a rotary long stroke FTS. A picture of the rotary FTS is shown in Figure 1-21 and Figure 1-22 shows a detailed cross section of the FTS. The lens is mounted on a Professional Instruments [73] Model 4R Twin Mount air bearing spindle which has an integrated motor and 10,000 count/rev encoder. The spindle is carried on a New Way [68] air bearing cross-slide which is driven by a linear motor on the basis of linear encoder feedback.

In this design, the diamond tool is mounted at the tip of an arm which pivots on a rotary axis on two sets of angular contact bearings. The rotary axis is driven by an

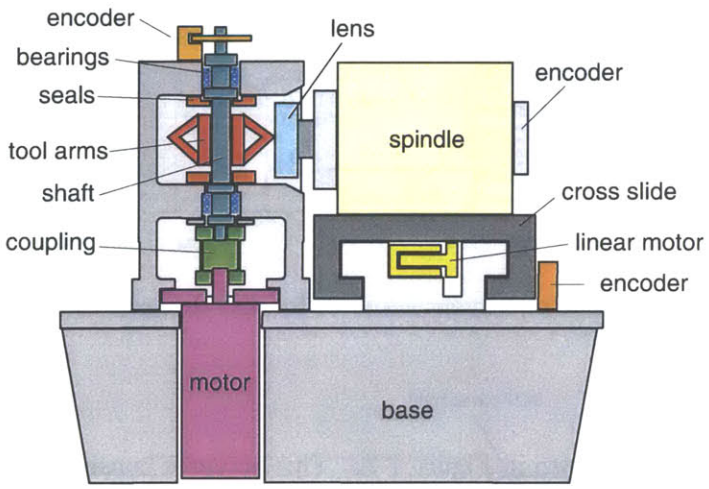


Figure 1-22: Schematic of the rotary fast tool servo.

Aerotech [3] model BM1400 brushless servo motor capable of 10 Nm torque in steady state. This axis is driven on the basis of feedback from a 60 million count/rev MircoE [59] rotary encoder. By using a balanced rotary arm, motor actuation inputs a pure torque into the arm structure and thereby creates only a pure reaction torque on the machine base. Because moment of inertia scales as a function of radius squared for each increment of mass, the moment of inertia of the machine base is about 10^5 larger than the moment of inertia of the servo axis. Further, the plate structure of the machine base is extremely stiff in shear, and thus experiences little deflection under the motor reaction torques. This large inertia ratio and machine stiffness means that base vibrations due to the fast tool servo reaction torques are insignificant. This is a compelling advantage over linear motion fast tool servos. The design and development of this rotary FTS is detailed in [54], [16], [55], [53], and [14].

1.6 Other Actuators of Interest

In the course of our literature search, we came across a few actuators and mechanisms that were not fast tool servos but were of interest to our FTS development. One is a small voice coil driven linear actuator for disk drive track following [64]. A schematic

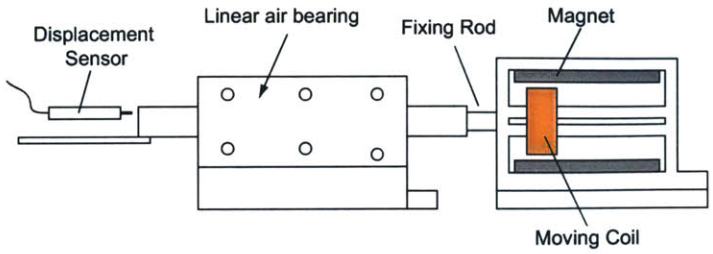


Figure 1-23: Schematic of a linear actuator for disk drive servo track writing (adapted from [64]).

of this actuator is shown in Figure 1-23. This actuator consist of a traditional orifice compensated air bearing system made of extra-super duralumin (Al-Zn-Mg-Cu alloy A7075) driven by a custom voice coil motor. Two different voice coil motors were tested. Figure 1-24 shows a schematic cross-section of the two voice coil motors. In the MC1 version, the moving coil support frame consist an oval shell on which the coils are wound and a single horizontal rib to which the fixing rod is attached. The MC1 actuator assembly has a moving mass of 13.72 grams and the entire actuator has a resonant mode at 6.9 kHz with this coil design. The MC2 version has a coil frame consisting of an oval shell with ribs in both the horizontal and vertical directions. The MC2 actuator assembly has a moving mass of 16.44 grams with a resonant mode at 14.4 kHz. The VCM force constant for both coil designs were within 4% of each other. With a maximum motor thrust of 4 N, the maximum actuator acceleration was 29 g's with the MC1 coil and 24 g's with the MC2 coil. Position feedback was accomplished with two different sensors. Sensor-A is a MTI-2000 photonic sensor with a resolution of 17 nm and a working range of 10 μm. Sensor-B is an ATOPS ATP-A30 with a resolution of 0.8 nm and a working range of 5.12 μm. No information is given for the total travel range of the actuator but from the design it is clearly greater than the working ranges of the sensors. With the MC1 coil a bandwidth of 2.2 kHz was achieved, while with the MC2 coil a bandwidth of 4 kHz was achieved. Tracking error with this actuator was limited by the sensor resolution [64].

Another actuator of interest is the high speed, long travel, dual voice coil actuator developed by ASM Assembly Automation Ltd. for the use in wire bonders and die

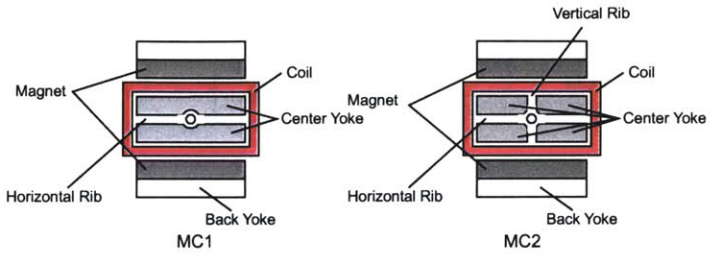


Figure 1-24: Cross section of voice coil motor (adapted from [64]).

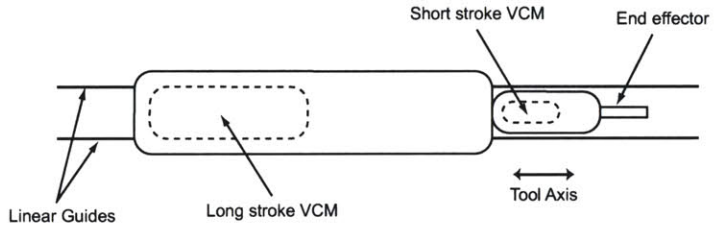


Figure 1-25: Schematic of a dual VCM actuator (adapted from [17]).

bonders [17]. Figure 1-25 shows a schematic of this actuator. The small VCM portion of this actuator is capable of 5 g accelerations and has an accuracy of 0.5 μm. We are interested in both the mechanical design of this structure and the controller design for this actuator since the position of our FTS will be the compounded position of the FTS and reaction mass stage. The designers note that because the position of the end effector is a function of large and small VCM position that the two controllers are coupled (Figure 1-26). Specifically, they note that the movement of the large VCM significantly disturbs the small VCM controller. Unfortunately the reference [17] goes into very little detail on the actual mechanical design and the actual controller used to decouple to motion of the two VCM.

Another interesting multiple stage actuator is the the ultra-precision aerostatic table developed at the Precision and Intelligence Laboratory, Tokyo Institute of Technology [82]. Figure 1-27 shows a top view schematic of this high speed nanometer positioning stage and Figure 1-28 shows a cross-section view. This stage consist of a coarsely positioned linear motor magnet track and a finely positioned aerostatic stage/linear motor coil assembly. By using a short travel linear motor to actuate the

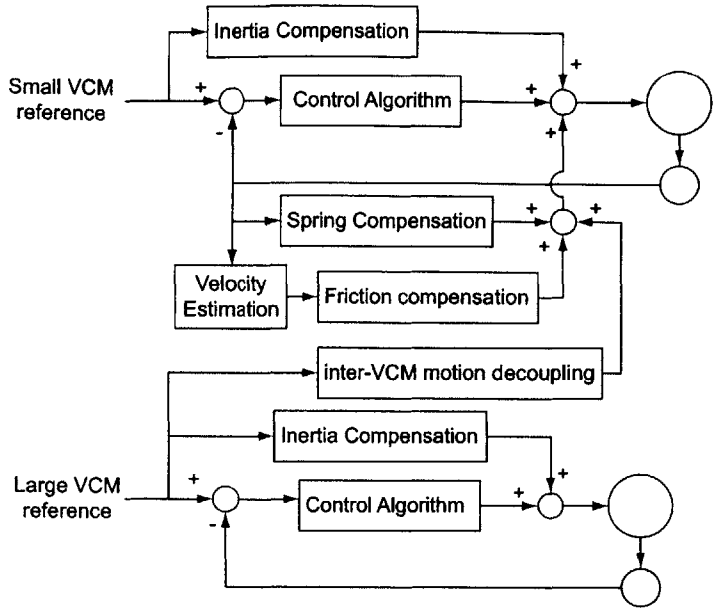


Figure 1-26: Schematic of a dual VCM actuator controller (adapted from [17]).

aerostatic table, the designer avoid the force ripple associated with long travel linear motors. The linear motor magnet track position is actuated by a DC servo motor driven cable drive. What is of interest in this design is the control architecture. Just as in our proposed FTS system, the position of two coupled system must be accurately controlled to properly position the stage. Figure 1-29 shows a block diagram of the control system for the hybrid stage. This structure is quite similar to the structure we have proposed for our FTS/reaction mass system.

Lastly, we look at a proposal to compensate for the steady state cutting force noted in [75] and [7]. Figure 1-30 shows a voice coil actuated fast tool servo which incorporates an auxiliary stepper motor. In this design, the voice coil motor provides the force to follow the high frequency portion of the FTS trajectory while the stepper motor provides the low frequency component. The controller for this system incorporates two loops. First there is a high bandwidth conventional position control loop around the voice coil motor. The stepper motor control loop is a low bandwidth controller meant to drive the voice coil motor current to zero. In simulation, this proposed control structure reduces the VCM copper losses by a factor of 10 [45].

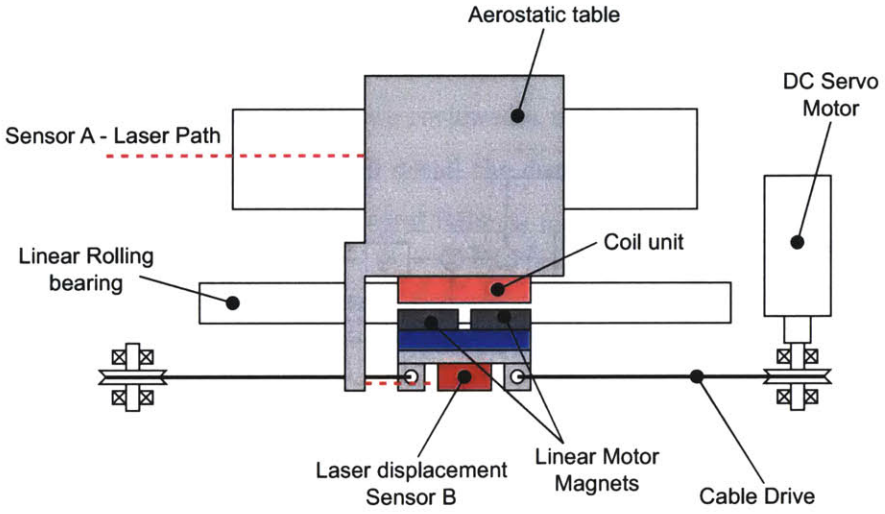


Figure 1-27: Top view schematic of a high speed nanometer positioning stage (adapted from [82]).

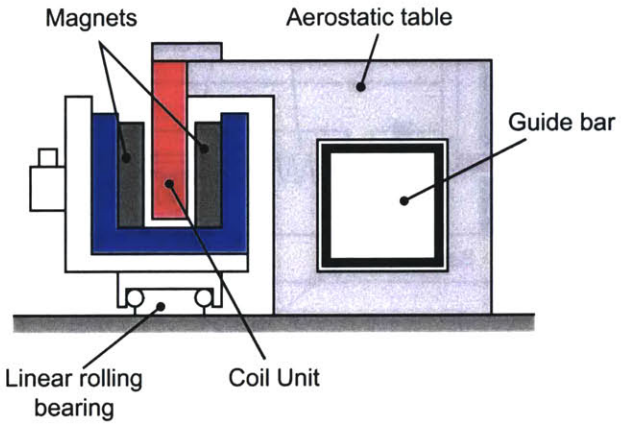


Figure 1-28: Cross-section of a high speed nanometer positioning stage (adapted from [82]).

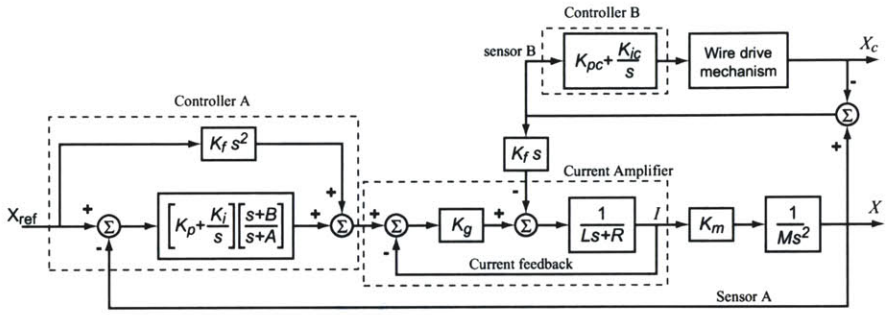


Figure 1-29: Position control system for the high speed namometer positioning stage (adapted from [82]).

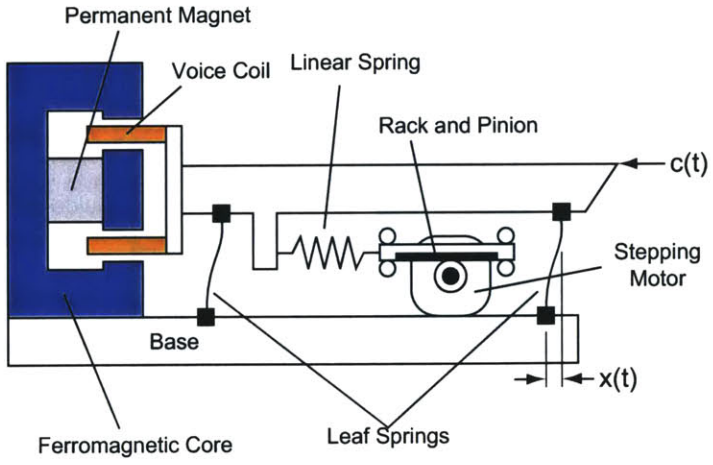


Figure 1-30: Schematic of a voice coil fast tool servo with auxiliary stepper motor to reduce copper losses in the voice coil motor (adapted from [45]).

1.7 Summary

In this chapter, we have provided a brief background detailing the motivations of this thesis. We also provided a brief overview of the thesis contents and listed the key contributions of this thesis. Lastly, we reviewed a number of existing fast tool servo designs. In the next chapter, we will detail the design and development of our long stroke linear fast tool servo with integral balance mass.

Chapter 2

Fast Tool Servo Concept and Design

In this chapter we review the design and development of our prototype linear long stroke fast tool servo with integral balance mass. First, we review a proposed second generation rotary fast tool servo. Second, we review a number of proposed linear fast tool servo designs. Lastly, we review the design and construction of our long stroke linear fast tool servo.

The performance goals for our long stroke fast tool servo are

1. A full length travel of 25 mm
2. A maximum acceleration of 500 m/s^2
3. An accuracy of ± 0.1 μm.

These performance requirements are those required to follow a 1 cm sinusoidal trajectory at 20 Hz, with sufficient accuracy for ophthalmic lenses.

2.1 Rotary Fast Tool Servo Concept

As part of our design process, we looked at the feasibility of replacing the rotary fast tool servo designed and built by Stephen Ludwick [54], David Ma [55], and David

Chargin [16] with an other rotary fast tool servo of a similar configuration which would allow for higher performance and greater accuracy. There were a number of issues that limited the performance and utility of the rotary fast tool servo as built. These include:

1. The 2.5 μm radial and axial error specifications of the Barden 205HDL angular contact ball bearings used to support the tool axis introduced an unknown and variable error motion at the tool tip.

2. The requirement to re-calibrate all of the tool arm position parameters after each tool change.

3. The rotational compliance of the drive between the drive motor and the tool arm.

4. The shaft windup between the tool arm and the rotary encoder.

5. The errors associated with the tool arm calibration process.

The obvious solution to the error caused by the angular contact bearing is to replace the rolling element bearings with a non-contact fluid-static bearing. The use of a fluid-static bearing was considered in the initial design process but for simplicity and robustness angular contact ball bearings were selected [55],[54]. There are two types of fluid-static bearings: hydrostatic in which the bearing fluid is either water or oil and aerostatic in which the bearing fluid is air [83]. We considered three different fluid-static bearings:

1. A traditional orifice compensated air bearing custom fabricated by Precision Instruments [73].

2. A porous orifice-compensated air bearing from New Way Inc. [68].

3. A self-compensating hydrostatic bearing [97],[84].

A traditional orifice-compensated air bearing consists of a stationary metal cylinder with orifices equally spaced around the circumference and a moving shaft. The air gap

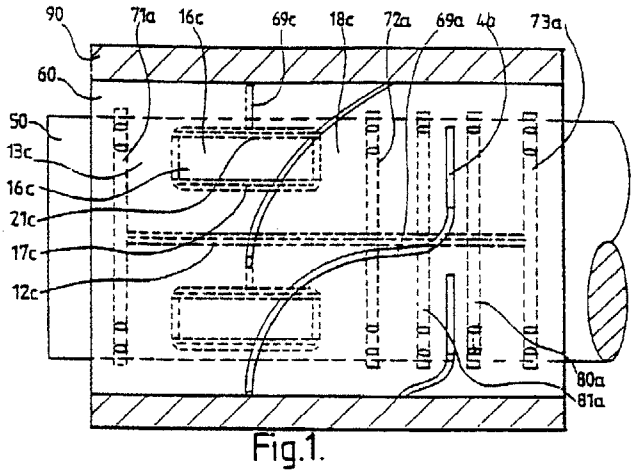


Figure 2-1: Cross-section of a self-compensating hyrdostatic bearing from U.S. patent #54660171.

between the stationary and moving components is between 1 and 5 μm. In a porous compensated air bearing, a porous media, such as porous carbon for the New Way [68] bearings, replaces the stationary metal cylinder. Since the media is porous, this design effectively replaces the finite number of orifices in a traditional design with an infinite number of orifices evenly spaced across a bearing surface. Air gaps are again in the 1 to 5 μm range. In general, a properly-designed traditional orifice air bearing offers better performance than a porous air bearing operating at the same pressure. This performance advantage comes at the expense of design robustness (porous air bearings are much more robust to bearing crashes and environmental factors) and manufacturing cost.

A self-compensating hydro-static bearing, for example the design shown in Figure 2-1, uses either oil or water as the bearing fluid. The hydro-static bearing offers extremely high stiffness but has the disadvantage of being a challenge to manufacture and requires the recovery of the fluid. At the time we elected to switch to a linear FTS design, the porous air bearing was our leading bearing candidate based upon it being the ease of manufacture, assembly, and cleanliness.

There are a number of design issues with the toolholder design on the original rotary fast tool servo. First the tool arms are clamped to the drive shaft. This means that when a tool is changed, we need to release the clamp from the shaft to make the coarse tool height change. The problem here is that when the clamp is released, we lose calibration for both the toolarm height and angular position. Of more consequence is that both toolarms are used to form the clamp. Adjusting the rough tool height of one arm requires the readjustment of both arm positions. Lastly, the original toolarm design did not incorporate features to aid the coarse positioning of the toolarm. In the original rotary FTS, toolarm positioning was typically done by slightly releasing the clamp pressure and then inserting a screwdriver between the toolarm and the either the top or bottom labyrinth seal. Since the labyrinth seal was also clamped to the shaft, it was often necessary to disassembly and re-assemble the seal after making tool height adjustments.

Figure 2-2 shows a side view of the proposed rotary FTS shaft, toolarms, and coarse height adjustment collar. Figure 2-3 show a top view cross-section of the same toolarm system. In this proposed toolarm system, each of the toolarms are independently bolted directly to a square shaft using 5 #8 bolts. By bolting the toolarms independently, the height of each arm may be coarsely adjusted separately. Using a square shaft allows the tool height to be coarsely adjusted without affecting the angular calibration. Fine tool height adjustment is achieved using the flexure mechanism designed by David Chargin [16]. Lastly, coarse height adjustments are made utilizing a removable clamp collar with a screw drive.

One of the main factors limiting the performance and bandwidth of the original rotary FTS was the compliant mode between the motor inertia and the toolarm/tool shaft inertia. During the design phase of the first generation rotary FTS, the motor/shaft/toolarm system was assumed to have a lumped element model consisting of a motor inertia attached via a rotational spring to lumped inertia representing the toolarm/shaft subsystem. The stiffness of the spring was assumed to be that of the FTS shaft from the motor coupling to the center of pressure of the toolarm clamp (Note: the motor coupling used in the rotary FTS was an order of magnitude stiffer

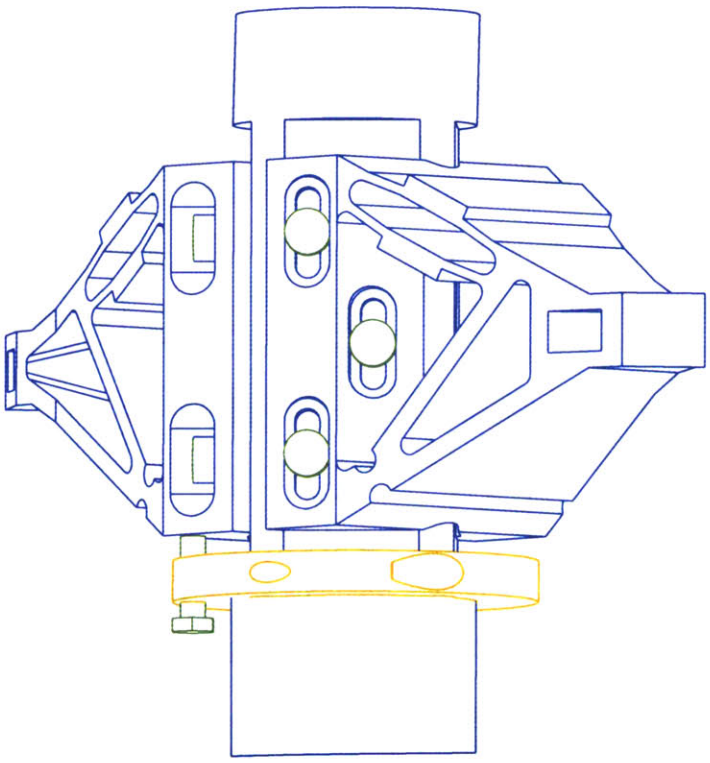


Figure 2-2: Side view of the proposed rotary fast tool servo shaft, toolarms, and coarse height adjustment collar.

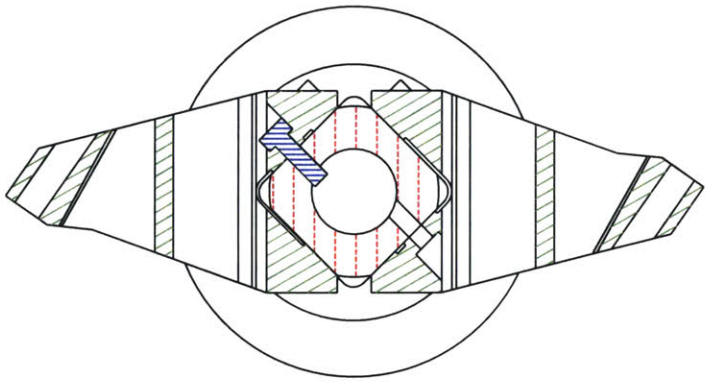


Figure 2-3: Top view cross-section of the proposed rotary fast tool servo shaft and toolarm.

than the FTS shaft). The lumped model was correct in terms of the elements used but incorrect in which element provided the limiting system. There are two shafts in the rotary FTS, one connecting the motor rotor to the coupling and a second from the coupling to the toolarms. The motor shaft is 0.5 inches in diameter and approximately 3 inches long. The toolarm shaft is 0.75 inches in diameter and approximately 5 inches long. The equation for the torsional stiffness of a round rod is

$$k = \frac{G\pi D^4}{32L} \tag{2.1}$$

where

$$\begin{aligned} G &= \text{Modulus of Rigidity} \\ D &= \text{Shaft diameter} \\ L &= \text{Shaft length.} \end{aligned}$$

Assuming that both shafts have the same G, the toolarm shaft is 3 times as stiff as the motor shaft. Since the springs are in series the equivalent spring stiffness is expressed as

$$k_{eq} = \left[\frac{1}{k_1} + \frac{1}{k_2}\right]^{-1}. \tag{2.2}$$

Thus the combination of the motor shaft and tool-shaft result in an equivalent spring stiffness 1/4 that of the tool-shaft alone. Thus the use of a frameless motor where the motor rotor attaches directly to the FTS tool-shaft would significantly improve the dynamics of the rotary FTS. Since we had planned to increase the shaft diameter to increase the aerostatic bearing stiffness, the shaft stiffness of a frameless motor design was projected to be 20 the stiffness of the original rotary FTS.

Figure 2-4 shows a cross-section schematic of our proposed second generation rotary FTS. The tool axis is supported by two radial bearing and axially constrained by a single thrust plate. The bearings in this drawing are based upon the orifice

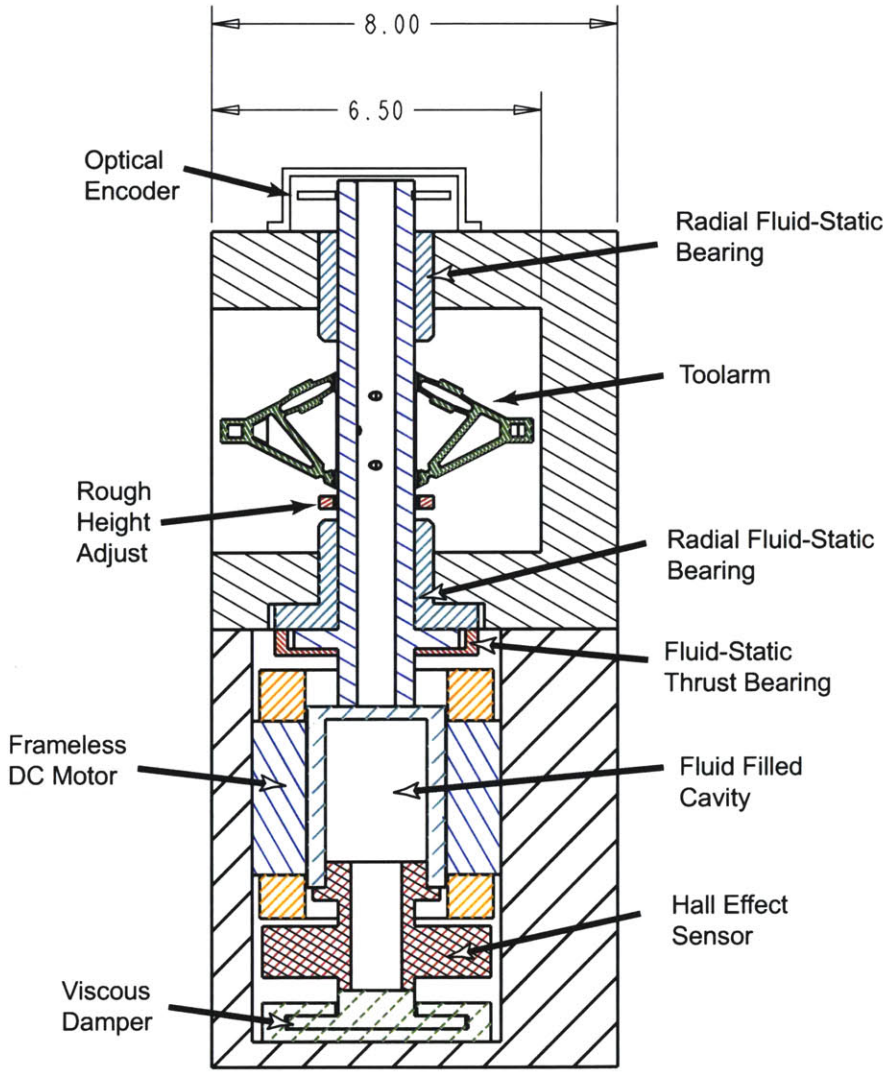


Figure 2-4: Cross-section schematic of a proposed rotary fast tool servo with fluid-static bearings and frameless motor.

compensated aerostatic bearings from Professional Instruments [73] and from design discussions with their engineering staff. The FTS has two toolarms to maintain shaft balance and to accommodate lens roughing and fining (rough cuts are made using a polycrystalline diamond or cubic boron nitride tool while fine cuts are made using a single crystal diamond tool). This cross-section is based upon the toolarm modifications discussed earlier. The axis is driven by an oil-cooled frameless motor. This proposed axis uses an Aerotech Inc. S-130-123 frameless motor with a continuous torque rating of 10.8 N-m and an inertia of 0.0078 kg-m^2 [3]. By cooling the motor, we should be able to significantly increase the maximum stall current and thus significantly increase the maximum stall torque. A Hall effect sensor is used to generate the motor commutation. At the bottom of the cross-section we have included an optional viscous damper. The angular position of the axis is measured using a MicroE Mercury series rotary encoder with a resolution of 0.1 μrad [59]. This leads to a tool-tip resolution of 8 nm.

This design does not solve two of the design issues with the first generation rotary FTS. First, it does not solve the problem of shaft windup between the toolarm and encoder. In the case of our proposed design the encoder is mounted about 10 cm from the nominal tool height. We can calculate the angular displacement of the shaft using the following

$$\theta = \int_0^L \frac{T}{GJ} dl \tag{2.3}$$

where

T = the applied torque

G = the modulus of rigidity

J = the area moment of inertia.

The applied torque can be expressed as

$$T(l) = I(l)\ddot{\theta} \tag{2.4}$$

where $I(l)$, the mass moment of inertia, is

$$I(l) = \frac{1}{2}m(l)r^2 = \frac{1}{8}m(l)D^2. \tag{2.5}$$

The shaft mass, $m(l)$, can be expressed as

$$m(l) = \int_l^L \frac{\rho\pi D^2}{4}dx = \frac{\rho\pi D^2(L-l)}{4}. \tag{2.6}$$

Combining (2.3), (2.4), (2.5), and (2.6) results in

$$\theta = \int_0^L \frac{\rho\pi(L-l)D^4}{64GJ}dl. \tag{2.7}$$

The area moment of inertia J for a round shaft is

$$J = \frac{\pi D^4}{64}. \tag{2.8}$$

Combining (2.7) and (2.8) results in

$$\theta = \int_0^L \frac{\rho\pi(L-l)D^4 64}{64G\pi D^4}dl = \int_0^L \frac{\rho(L-l)}{G}dl \tag{2.9}$$

$$= \frac{\rho}{G}\left(Ll - \frac{l^2}{2}\right)\Bigg|_0^L = \frac{\rho}{G}\left(L^2 - \frac{L^2}{2}\right) \tag{2.10}$$

$$= \frac{\rho L^2}{2G}. \tag{2.11}$$

This is an interesting result in that the windup error is a square function of the distance between the tooltip and the encoder and a linear function of the ratio between the material density and rigidity. This result clearly indicates that we want to place the encoder as close to the toolarms as possible. While it is possible to bring the encoder inside the bearing, this introduces a number of challenges. First, since the

encoder scale is a continuous disk we must mount the encoder prior to fixing the axial location of the tool shaft. Second, since the encoder is a sensitive device the mounting structure for the encoder must include some protective structure which prevents chips and cutting fluids from contaminating the scale while allowing the fluid from the fluid-static bearing to escape.

The density/elasticity ratio lead to some interesting material properties investigations. Since we were unlikely to come up with a design that placed the encoder and tool in the same plane, to minimize measurement errors we would like to select the shaft material with the highest stiffness per unit mass. Table 2.1 shows the density, elasticity, and elasticity/density ratio for several potential shaft materials. Elasticity and rigidity are related by the following formula:

$$G = \frac{E}{2(1+\nu)} \tag{2.12}$$

where

$$E = \text{the modulus of Elasticity} \tag{2.13}$$

$$\nu = \text{Poisson's ratio.} \tag{2.14}$$

Since G and ν were not available for all the materials in the table, we used E as a substitute for G. It is interesting to note that the elasticity to density ratio is nearly constant for the three metals. In order to get a higher ratio, we needed to go to an aluminum based ceramic. Shepal-M, a high performance machinable ceramic, has an elasticity to density ratio twice that of the metals while Alumina has elasticity to density ratio three times that of metals but has poor machinability.

Lastly, this design does not address at all the difficulties in accurately positioning a rotary axis relative to the rest of the machine tool reference frame. Specifically, measuring the pitch and the yaw of the FST axis relative to the x-y coordinate frame of the machine base and then accurately adjusting this orientation is not addressed here.

Material	Density gr/cm^3	E GPa	E/D
AL-6061 T6	2.69	69	25.7
4140 Steel	7.8	200	25.6
Ti 5AL-2.5 Sn	4.48	115	25.7
Alumina Al^2O^3	3.67	300	81.7
Shepal-M (Al-nitride)	2.94	160	54

Table 2.1: Elasticity and density of potential shaft materials

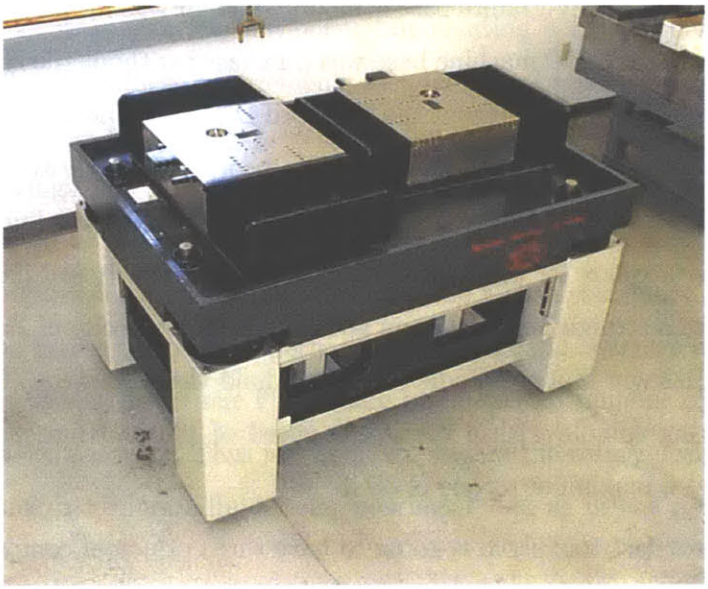


Figure 2-5: Photo of Moore Nanotechnology machine base.

Since this design is primarily a derivation of the original rotary fast tool servo and does not include any significant breakthrough technologies or topologies, we elected to pursue a high acceleration linear fast tool servo design which would require incorporating a reaction mass to attenuate base accelerations.

2.2 Linear Fast Tool Servo Concepts

At the start of the fast tool servo design process, we made several component selections. First, the fast tool servo was to be mounted on a Moore Nanotechnologies Nanotech 350 machine base [63] as Moore was willing to supply this base to us with-

out their standard controller at a significant discount. Figure 2-5 shows a photo of the machine base without the protective canopy. The Nanotech 350 machine base consists of two hydrostatic slides mounted in a T configuration on a epoxy-granite base. The slides are driven by Aerotech linear motors [3]. The stage position is measured using Sony BS75A glass-scale laserscales (0.1379 μm pitch) combined with 16x interpolation (8 nm resolution) from Sony BD15 detectors with A quad B incremental output [85]. The epoxy-granite base is mounted on air legs for passive vibration isolation. The Nanotech 350 machine base was purchased without the standard high- and low-voltage cabinets and without the standard Delta Tau motion controller. Instead, we constructed a custom instrument chassis, incorporating a VME bus for the BD15 detectors, 5, 12, and 24 volt power supplies, amplifiers for the linear motors, and a custom interface box for a dSPACE 1103 PPC controller board [26]. The Nanotech 350 has an axis straightness of 0.3 μm and a maximum velocity of 25 mm/s. Secondly, a PI [73] 4R Twin-Mount spindle is used to hold the workpiece. The 4R spindle is an air bearing spindle with a maximum speed of 10,000 Rpm (5,000 Rpm for our spindle) and a maximum torque of 60 lb*in.

Any linear fast tool servo is going to have three principal components:

1. Bearings

2. Actuator

3. Sensor.

Since we are building a long stroke linear FTS to achieve maximum part accuracy we need to attenuate the reaction forces in the machine base. In the following subsections, we will review the available bearing options, analyze several potential actuator designs, review the sensor systems considered for the FTS, and lastly we will look at several possible reaction mass configurations.

2.2.1 Bearings

There are many different types of bearings available including sliding contact bearings, rolling contact bearings, flexure bearings, fluid-static bearings, fluid-dynamic bearings, and magnetic bearings. Since we would like to have extremely accurate motion, it is preferable to utilize a bearing without mechanical contact and thus sliding contact and rolling contact bearings are not desirable for this application. Flexure bearings offer smooth accurate movement but creating a flexure bearing with the requisite travel and stiffness requirements seemed improbable. Hydrostatic and hydrodynamic bearings both use a pressurized fluid to support the bearing load. In fluid-static bearings, the fluid is pressured from an external source. In fluid-dynamic bearings, the fluid is pressured using the motion of the bearing components. Since the FTS bearings must support loads even when the FTS is stationary, we cannot use a fluid-dynamic bearing. While properly-designed magnetic bearings would meet the performance requirements of our FTS, there are a limited number of commercially available magnetic bearings. Thus just as in the case of the rotary design, we focused on designs which incorporate fluid-static bearings. Just as in the rotary design we considered three bearing options:

1. A traditional orifice compensated air bearing .

2. A porous material compensated air bearing .

3. A traditional orifice compensated fluid-static bearing using oil as the bearing fluid.

Just as in the rotary FTS, the porous material bearing offers the advantage of design simplicity at the expense of slightly lower stiffness when compared to a properly designed and constructed orifice design. An oil fluid-static bearing offers the highest stiffness at the expense of having to recover and clean any oil used in the bearing. In theory, an oil hydro-static bearing would result in the smallest slide and thus the lowest moving mass.

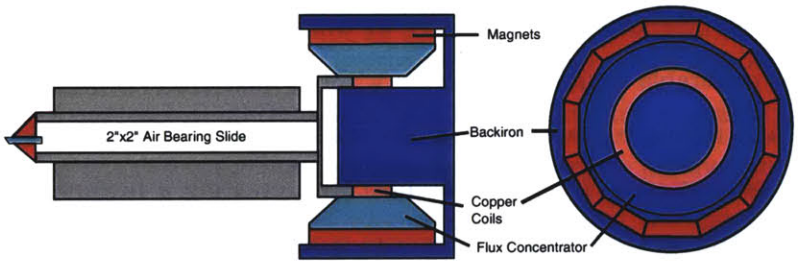


Figure 2-6: Sketch of a speaker style VCM with a 2"x2" air bearing slide.

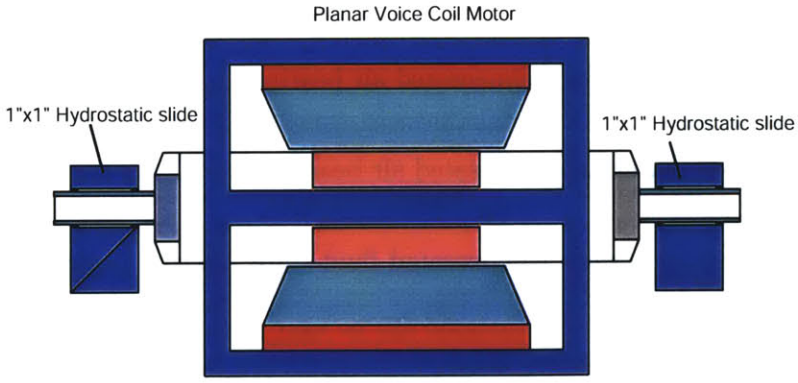


Figure 2-7: Sketch of a planar style VCM mounted between two 1"x1" hydrostatic bearings.

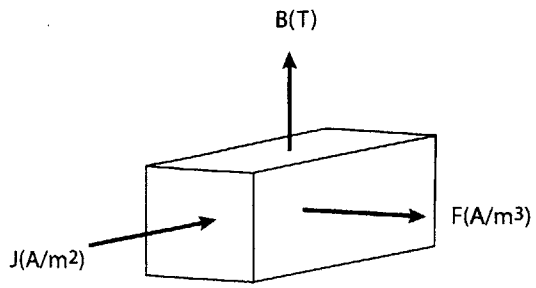


Figure 2-8: Schematic diagram showing a conductor with current density J in a magnetic field of density B with a resultant force density F.

We considered two configurations for the bearings. Figure 2-6 shows a sketch of a 2"x2" air bearing slide driven by a cylindrical voice coil motor. The air bearing slide consists of two 2"x2" square air bushings set 6" apart. This configuration has the advantage of allowing maximum flexibility in terms of actuator design. Figure 2-7 shows a sketch of planar VCM set between two 1"x1" hydrostatic bearings. This layout potentially offers enhanced rocking stiffness but somewhat limits actuator design. The prototype FTS utilizes the slide design since a slide appropriate for our application was available off the shelf.

2.2.2 Actuator

For the FTS actuator, we considered several styles of single phase permanent magnet linear motors. Single phase permanent magnet linear motors can be constructed with either moving magnets or moving coils (voice coil motors). The force density of these motors is determined by a combination of the static magnetic field from the permanent magnet and the maximum current density in the coil. To understand the acceleration limits of a linear actuator, consider the Lorentz-type system shown in Figure 2-8. In this schematic a conductor carrying current density J is placed in a magnetic field of density B which results in a Lorentz force of density F. Typical voice coil actuators without forced cooling have a magnetic field density $B = 0.8$ T and maximum steady state current density $J = 7$ A/mm^2 [18] which results in a force

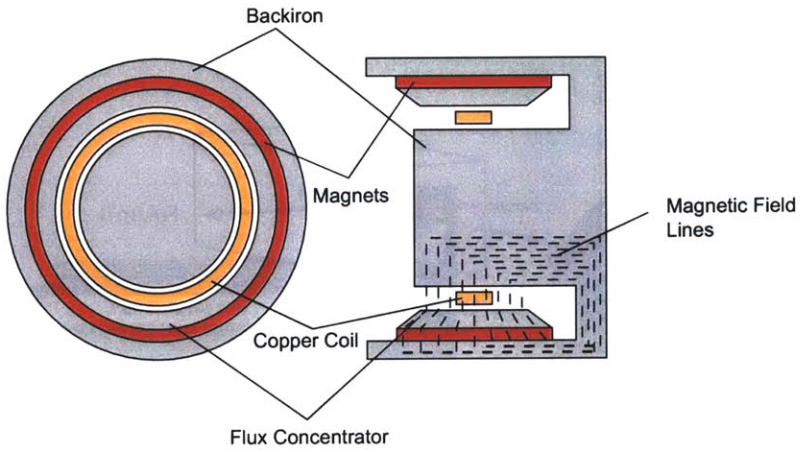


Figure 2-9: Schematic diagram of a cylindrical voice coil motor design.

density

$$F = J*B = 5.6 * 10^{-3} \frac{\mathrm{N}}{\mathrm{mm}^3} = 5.6 * 10^{6} \frac{\mathrm{N}}{\mathrm{m}^3}. \tag{2.15}$$

Assuming that the conductor is copper with a density of 8900 kg/m^3, a Lorentz actuator has a maximum acceleration of

$$a_{max} = \frac{F}{\rho} = \frac{5.6x10^{6}}{8900} \frac{\mathrm{N}}{\mathrm{kg}} = 630 \frac{\mathrm{m}}{\mathrm{s}^2} \tag{2.16}$$

Since the acceleration limit of the coil alone is near the desired acceleration of our FTS, the actuator design needs to be modified to increase both the field density B and the current density J.

2.2.3 Voice Coil Motor Design

The most common form of a single phase permanent magnet linear motor is the moving coil style. The moving coil design is traditionally used because it allows for designs where the magnetic flux stored in the permanent magnets is concentrated. We considered several different motor designs which incorporate flux concentration. Figure 2-9 shows a schematic drawing of a cylindrical voice coil motor design. In this

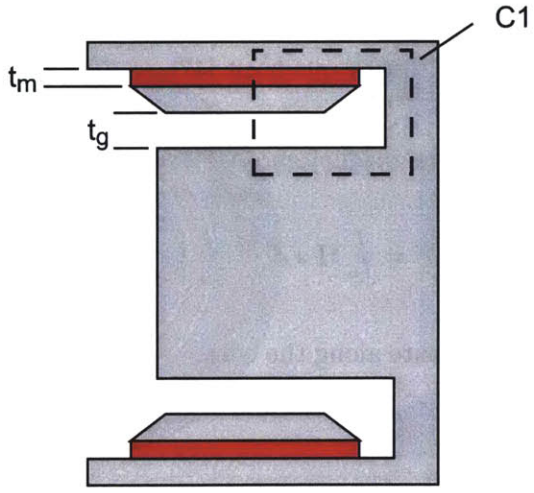


Figure 2-10: Integration contour C_1.

design, the magnetic flux is concentrated in two dimensions. The flux is concentrated in the radial direction by reducing the surface area of the air gap relative to the surface area of the permanent magnets. The flux is concentrated in the axial direction by reducing the length of the exposed area of flux concentrator at the air gap. As long as the back-iron or flux concentrator do not become saturated all of the magnetic field stored in the magnets is concentrated into the motor air gap. This design has the further advantage that all of the copper in the moving coil is within the magnetic field and thus generating force. The disadvantage of this design is that it is quite difficult to achieve significant flux concentration without saturating the center yoke of the back-iron.

To explore this let us study the magnetic circuit of this motor. For this analysis we have made the following assumptions:

- That this system may be treated as a quasi-static magnetic-field system.
- The back-iron and flux concentrator are constructed of a highly permeable material, $\mu \rightarrow \infty$. This assumption reduces magnetic field intensity H in these components to 0.
- Flux leakage around the circuit is minimal.

- The permanent magnet has approximately straight line magnetization $B - H$ characteristic in the second quadrant [47].

With these assumptions the magnetomotive force, $\mathcal{F}$, is

$$\mathcal{F} = \oint_C \mathbf{H} \bullet d\mathbf{l} = \int_S \mathbf{J}_f \bullet \mathbf{n}\, da. \tag{2.17}$$

To determine H, we integrate along the contour C_1 (Figure 2-10). Since there are no currents inside this contour, $\mathcal{F}$ is zero and

$$H_g t_g + H_m t_m = 0 \tag{2.18}$$

where

$$\begin{aligned} H_g &= \text{the magnetic field intensity in the airgap} \\ H_m &= \text{the magnetic field intensity in the magnet} \\ t_g &= \text{the gap thickness} \\ t_m &= \text{the magnet thickness.} \end{aligned}$$

Gauss's Law states that the magnetic flux through a volume must be conserved:

$$\oint_S \mathbf{B} \bullet \mathbf{n}\, da = 0. \tag{2.19}$$

Using our assumption of no flux leakage, integrating over the cylinder defined by the radius of magnet centerline and the radius of the airgap centerline (Figure 2-11) yields

$$0 = 2\pi R_m L_m B_m - 2\pi R_g L_g B_g \tag{2.20}$$

$$B_g = \frac{B_m R_m L_m}{R_g L_g} \tag{2.21}$$

where

$$B_g = \text{the magnetic flux density in the airgap}$$

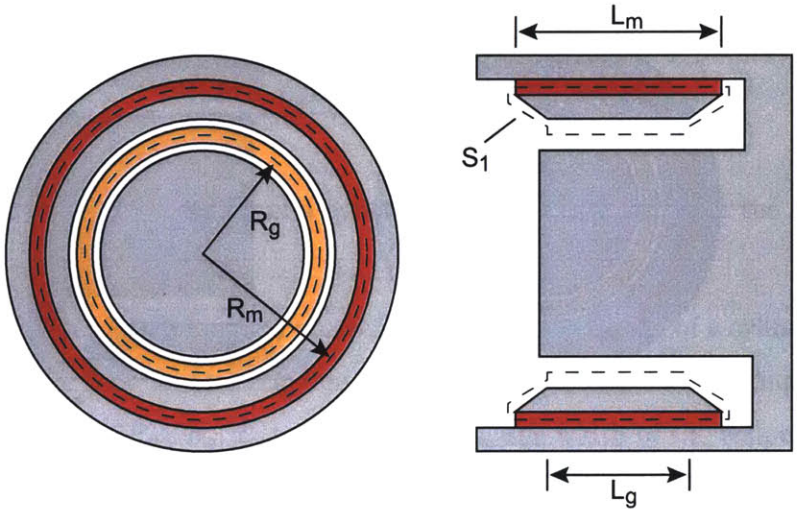


Figure 2-11: Integration surface s_1.

$$
\begin{aligned}
B_m &= \text{the magnetic flux density in the magnet} \\
R_g &= \text{the centerline radius of the airgap} \\
R_m &= \text{the centerline radius of the magnet} \\
L_g &= \text{the axial length of the airgap} \\
L_m &= \text{the axial length of the magnet.}
\end{aligned}
$$

In order to solve for the flux density in the air gap, we need to employ the constitutive laws relating magnetic flux density, $\mathbf{B}$, to magnetic field intensity, $\mathbf{H}$. In the airgap, $\mathbf{B}$ and $\mathbf{H}$ are related as

$$\mathbf{B} = \mu_0 \mathbf{H} \tag{2.22}$$

where μ_0 is the permeability of free space. Assuming linear straight line magnetization characteristics, the relationship in the magnet is

$$B_m = B_r + \mu_0 H_m. \tag{2.23}$$

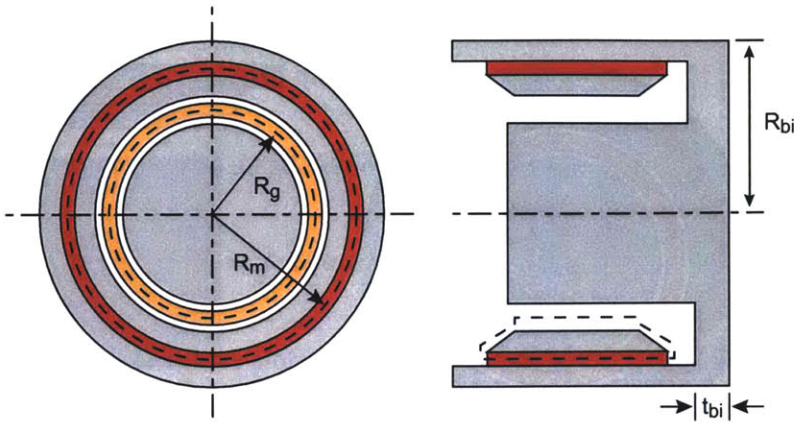


Figure 2-12: Schematic of cylindrical voice coil motor with key dimensions.

Combining Equations (2.18), (2.20), (2.22), and (2.23) results in

$$B_g = \frac{R_m L_m t_m}{R_g L_g t_m + R_m L_m t_m} B_r. \tag{2.24}$$

We now have an expression for the flux density in the airgap. In order for our assumption of no flux leakage to be accurate, we need to ensure that steel in the back-iron does not become saturated. Most steels become saturated between 2 and 2.2 T with a significant decrease in permeability beginning at 1.8 T. So for design safety, we would like to keep the flux density in the back-iron below 1.8 T. In terms of the back-iron design, the only place we are not allowed to add additional material is on the inner yoke, thus we need to check each design for saturation here. The flux density in the center yoke can be expressed as

$$B_{bi-c} = \frac{2\pi R_g L_g B_g}{\pi (R_g - t_g/2)^2}. \tag{2.25}$$

We can calculate the minimum thickness t_{bi} of the return yoke using

$$t_{bi} = \frac{R_g L_g B_g}{(R_g - t_g/2) B_{max}} \tag{2.26}$$

where $B_{max} = 1.8$ T, the maximum desired flux density. The outer radius of the

back-iron can be calculated from

$$R_{bi} = \sqrt{\frac{(2\pi R_g L_g B_g / B_{max}) + \pi(R_m + t_m/2)^2}{\pi}}. \tag{2.27}$$

Figure 2-12 shows a schematic of a cylindrical voice coil motor with the key dimensions.

We created a Matlab function *motor_size* to aid in the design of a cylindrical voice coil motor. The code for this function is included in Section B.1. The inputs to this function are t_m, t_g, R_m, R_g, L_m, L_g, B_r, and L_{cu}, the width of the copper coil. The version of the code included in this thesis assumes a copper packing factor of 70%. Packing factor is the percentage of the copper coil area which actually contains copper. The code also assigns some basic geometry to the coil assembly. In this version, we assume that the FTS is supported by a 2.54 x 2.54 x 19.8 cm (1"x1"x7.75") slideway with oil hydrostatic bearings. The function outputs B_g, the flux density in the center yoke, the mass of the copper coil, the total moving mass if the slideway is made of aluminium, the total moving mass if the slideway is made of steel (Note: the program assumes that the coil assembly is made of aluminum in both cases but other alternatives include stainless steel and titanium), and the required current density to accelerate the moving mass for both steel and aluminium. The function output is a single figure with a scaled schematic of the VCM with the input dimensions and the calculated performance.

Figure 2-13 shows a sample output for the *motor_size* function. This is a fairly conservative design, where if the slideway were constructed of aluminium the motor coils would not need to be cooled. The disadvantage of this design is that the final motor is quite large at 25 mm in diameter. Figure 2-14 shows a more aggressive motor design where the overall motor volume is reduced by 45%. In this motor design, the coil will need significant cooling to prevent the coil from overheating.

There were several iterations of the this Matlab function generated. The function *motor_size_s* was a function designed to evaluate square voice coil motor designs. In this design the magnetic circuit is broken into four equally sized rectangular sections.

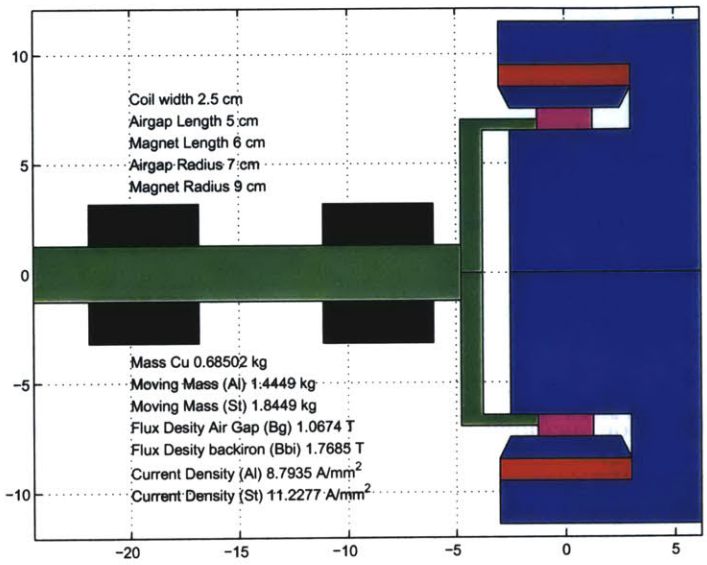


Figure 2-13: Sample output of Matlab function *motor_size*.

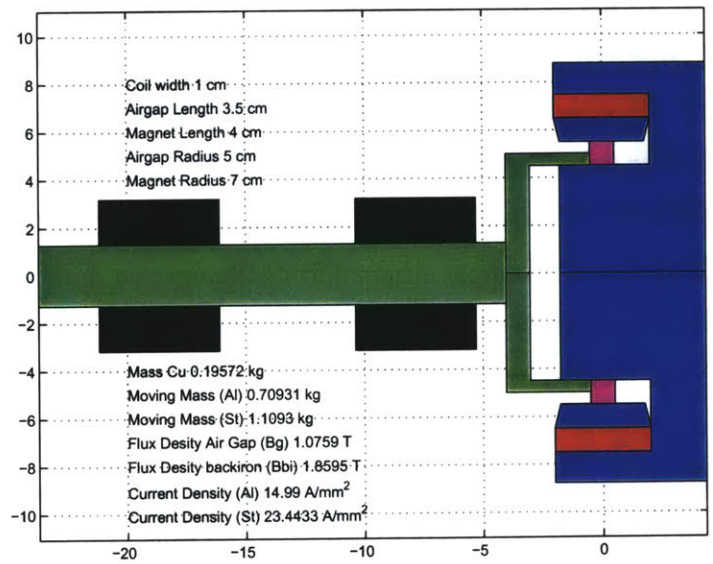


Figure 2-14: Sample output of Matlab function *motor_size*.

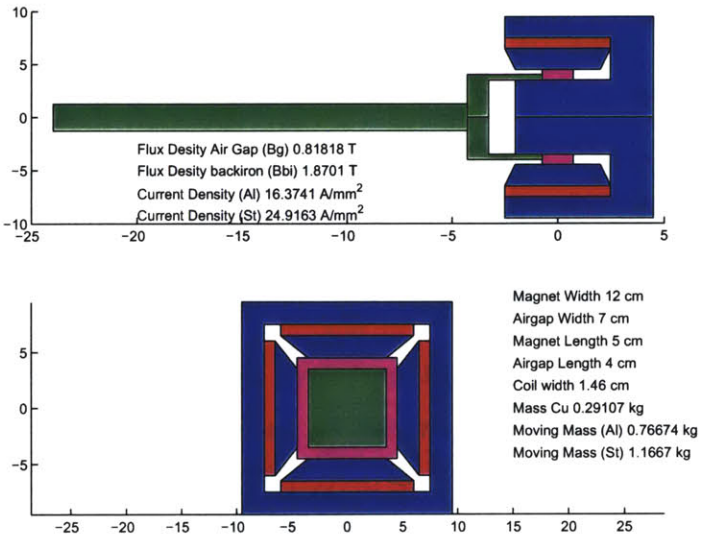


Figure 2-15: Sample output of Matlab function *motor_size_s*.

The four circuits share the same center yoke. This motor configuration is less efficient electrically since some percentage of the copper is outside the magnetic field and thus does not produce force. The inputs for the *motor_size_s* are a little different since the magnet remanence (B_r) is assumed to be 1.2 T and the coil width is calculated using the input gap length (L_g) minus the motor travel (2.54 mm). Figure 2-15 shows a sample output of the *motor_size_s* function. This sample motor is 19 x 19 mm and is quite aggressive needing a coil current density of 16 A/mm^2 for the lightest moving mass. The code for this Matlab function is in Section B.2. For the square VCM the relationship between the remanent flux density of the magnet and the gap flux density is

$$B_g = \frac{L_m W_m t_m B_r}{L_g W_g t_m + L_m W_m t_g} \tag{2.28}$$

where W_m and W_g are the width of the magnet and air gap respectively. The flux density in the center yoke is

$$B_{bi-C} = \frac{4 L_g B_g}{W_g}. \tag{2.29}$$

There are several additional variations of this function written including

- Square motors with only two opposing magnetic circuits (planar VCM).
- Square and cylindrical motors with moving magnet designs.
- Square and cylindrical motors using a 2.54 x 2.54 x 19.7 cm aerostatic stage, the stage actually used for the FTS prototype, is modeled.

Since we did not utilize any of this analysis in our prototype FTS, I have elected to include only a sample of the variations tested in this thesis. All of this analysis assumes that we will be able increase the current density of the VCM coil with cooling. In the next section, we introduce some of our cooling concepts.

2.2.4 Coil Cooling

As mentioned earlier, there are two ways to increase the force density of a voice coil motor:

1. Increase the magnetic flux in the motor air gap.
2. Increase the maximum allowable current density of the coil.

As we saw in the previous section, there is a functional limit to how much we can increase the flux in the gap based upon back-iron saturation (the maximum gap flux I achieved in a calculated design was 1.2 T). This means we need to increase the maximum allowable coil current density. Since the coil current density is limited by thermal considerations, we need to cool the motor coils.

Michael Liebman designed and constructed a three phase linear motor where slots were opened in the coil endturns to allow for the passage of cooling oil [46]. Figure 2-16 shows a schematic drawing of the end turn cooled coils. We can apply this technique to several different motor configurations. Figure 2-17 shows a schematic drawing of a square VCM with end turn cooling and a planar VCM with end turn cooling. The square VCM has the advantage that most of the coil copper is captured in the VCM magnetic field but the disadvantage of very little cooling area. The planar VCM has

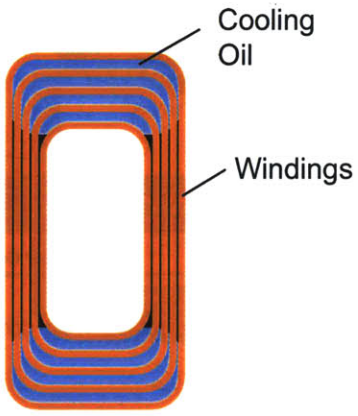


Figure 2-16: Schematic drawing of end turn cooling from [46].

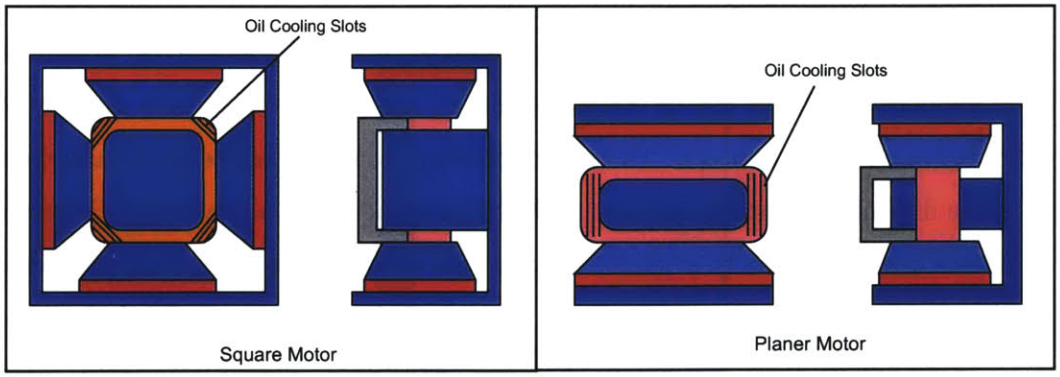


Figure 2-17: Schematic drawing of a square VCM with end turn cooling and a planar VCM with end turn cooling.

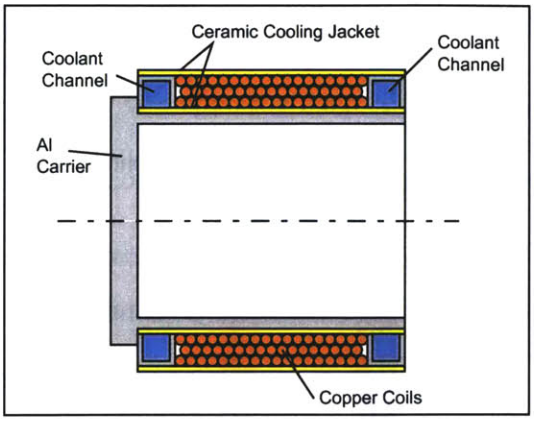


Figure 2-18: Schematic drawing of a coil cooling concept derived from U.S. Patent application US-2004/0207273.

a large cooling area at the expense of a large volume of copper outside the magnetic field.

Figure 2-18 shows a schematic drawing of an alternate coil cooling concept derived from U.S. Patent application US-2004/0207273 [31]. In this design, channels carrying a coolant are placed at either end of the coil windings. The copper windings are then captured between a pair of thermally conductive ceramic cylinders. The highly thermally conductive material transfers the heat from the coils to the coolant. Figure 2-19 shows an extension of this concept where the heat transfer from the coil to the sheath is enhanced with the use of vertical sheets of aluminium foil. There are a number of issues with this concept. The most important is that these ceramic materials are quite difficult to machine and are typically manufactured to final form. Rectangular sheets are commercially available making this cooling concept more appropriate for the square motor designs.

Figure 2-20 shows an alternate coil cooling concept derived from [46] where aluminium foil is place horizontally between winding layers. The ends of the foil are then place in the coolant stream. The windings can be interrupted periodically to introduce local cooling channels. Figure 2-21 shows a final coil cooling concept where the aluminium carrier is slotted along the axial dimension of the carrier. These slots allow coolant to circulate from the front to the back of the motor underneath the

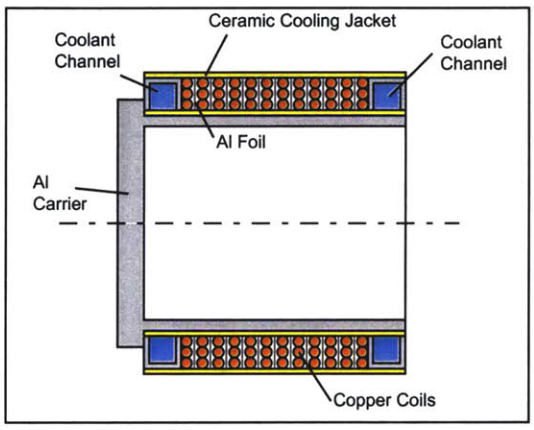


Figure 2-19: Schematic drawing of a coil cooling concept derived from U.S. Patent application US-2004/0207273.

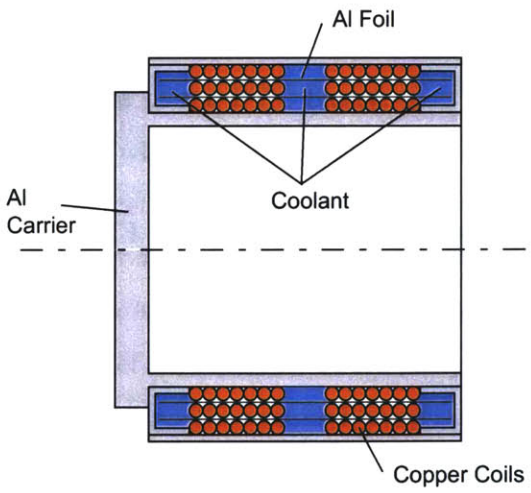


Figure 2-20: Schematic drawing of a coil cooling concept derived from [46].

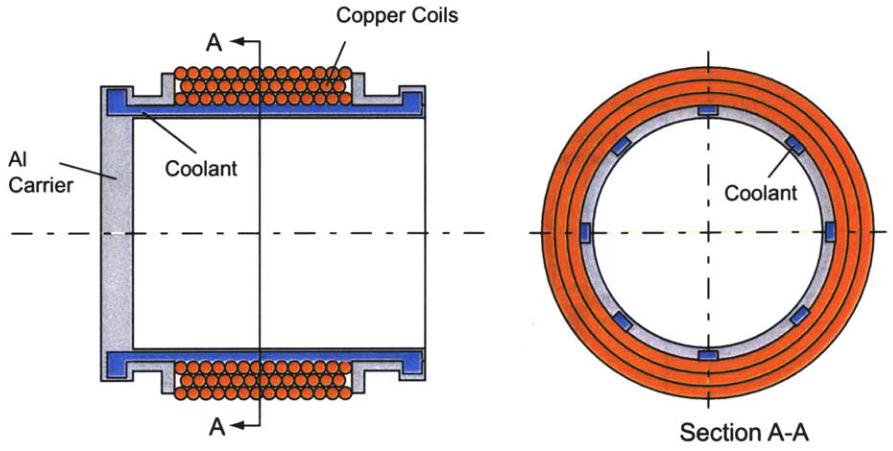


Figure 2-21: Schematic drawing of a coil cooling concept.

copper coils.

Clearly all of these coil cooling concepts offer significant design and fabrication challenges. It seems unlikely that we could design a motor which has sufficient structural integrity while having adequate cooling, coolant sealing, coolant plumbing, and dynamic performance. Thus we choose to utilize the existing oil-cooled linear more developed by Michael Liebman [46] to actuate the FTS. In the next section, we discuss the sensor selection process.

2.2.5 Sensor

We considered two different sensor types for our FTS:

1. Laser interferometer

2. Linear encoder

Figure 2-22 shows a schematic of a linear FTS with a laser interferometer. A laser interferometer offers several advantages. First, it offers extremely high resolution (as fine as almost 0.1 nm) and extremely high maximum speeds (up to 5.1 m/s). Second, laser interferometry is the only sensor that readily allows us to measure the FTS position almost exactly at the tool tip. Third, the target reflector contributes

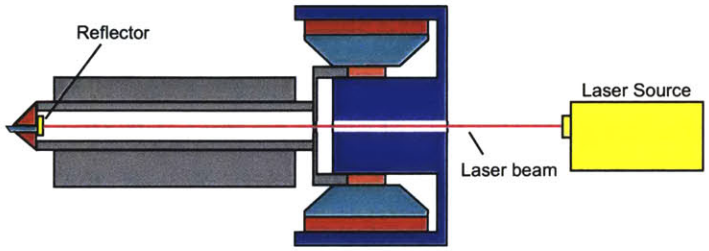


Figure 2-22: Schematic of a linear FTS with a laser interferometer sensor.

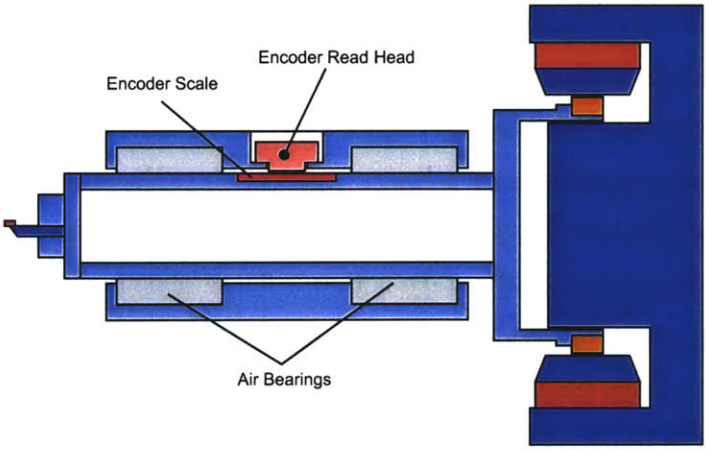


Figure 2-23: Schematic of a linear FTS with a linear encoder sensor.

very little to the moving mass of the FTS. Lastly, laser interferometry is the only one of the sensors considered which measures the FTS position relative to the fixed machine base coordinate frame. Offsetting these advantages are several significant disadvantages. First, beam routing in the limited space available on the machine base is challenging. Second, a laser interferometer is quite sensitive to disturbances in the beam path. In the case of our machine, we are likely to produce a large amount of airborne particulate either from the material being cut or any cutting lubricant used. Lastly, laser interferometers are very expensive ($20,000 for the Zygo ZMI 4004 board alone). The interferometer availiable here in the Precision Motion Control Lab had a resolution of 2 nm and a maximum speed of 350 mm/s which makes it unsuitable for our purpose (a 12.5 mm pk-pk 20 Hz sinusoidal trajectory has a maximum tool tip velocity of 1500 mm/s).

Figure 2-23 shows a schematic of a linear FTS with a linear encoder sensor. Specifically, we considered using exposed glass scale diffraction-based linear encoders. There are several reasons to use this type of sensor for this application. First, exposed scale linear encoder are non-contact thus friction free. Second, the sensor can be placed fairly close to the actual tool tip allowing for minimal position measurement error due to component deformation. Third, diffraction based linear encoders provide nm level resolutions without the beam routing and environmental issues associated with laser interferometry. Lastly, exposed glass scale linear encoders meeting our performance requirements are commercially available from several sources (Sony, Heidenhain, and MicroE for example) at a reasonable price ($1000 for scale and electronics). There are several disadvantages to using a linear encoder. Since the scale cannot be placed coincident with the tool tip, there will always be some error in the measured and actual position. The second disadvantage is that a linear scale measures the FTS position relative to the FTS reference frame. This means that to determine the position of the tool tip relative to the base reference frame we need to measure the position of the FTS reference frame relative to the base reference frame. Thus the tool tip position measurement now incorporates two measurement errors. Lastly, the accuracy and maximum velocity of a glass scale diffraction based linear encoder is a function of the scale pitch. The longer the scale pitch, the higher the maximum velocity. The smaller the scale pitch, the higher the accuracy. As will be discussed in more detail in Section 2.3, there are encoders avaliable that meet either our velocity requirements or our accuracy requirements but not scales currently avaliable which meet both.

2.2.6 Reaction Force Attenuation

There are several possible methods for reducing base movements in response to FTS actuation forces. The simplest and most common strategy is to make the machine base much more massive than the FTS. In the case of most short stroke FTS, the moving portion of the FTS has a mass of tens of grams while the machine base has a mass of hundreds of kilograms thus leading to a FTS/base mass ratio on the order of 1000. For longer stroke FTS, the moving component of the FTS is much more

massive (on the order of kilograms) while the machine base mass remains relatively static. Thus the FTS/base mass ratio is on the order of 100. This means we must explore more sophisticated strategies.

Figure 2-24 shows schematic drawings of three reaction force attenuation strategies. In Figure 2-24-A, the FTS reaction forces are passively attenuated by a reaction mass (in this case the magnet/back-iron assembly of a VCM) connected to the in-feed slideway by a flexure. This design offers the advantage that it is completely passive thus eliminating the need for any additional actuator or control elements but the success of this strategy is very dependant on the dynamics of the reaction mass/flexure system. As noted earlier Weck et al [100] utilized this strategy with limited success on their hybrid FTS. Figure 2-24-B shows a reaction force attenuation strategy where a second actuator (in this case a second FTS) is placed on the in-feed slide. The two actuators act in opposite directions. In theory if the actuator trajectories and masses are properly balanced the slide way sees no net force although there may be net torques. The disadvantage of this design is that it requires the expense of an additional actuator system which has no functionality beyond attenuating reaction forces. Figure 2-24-C shows a third strategy. In this case, the in-feed slide-way is allowed to freely move in response to the FTS actuation forces. While this motion complicates the trajectory generation and control of the in-feed slide, it has the advantage of utilizing the existing machine hardware, thus saving money and effort.

In this section, we have reviewed the various different design options we explored when designing our prototype fast tool servo. In the end, we elected to utilize an off-the-shelf porous air bearing slide, an existing oil-cooled linear motor, a diffraction based linear scale, and FTS/slideway reaction force attenuation strategy. In the next section, we detail the design and construction of our prototype fast tool servo.

2.3 Prototype Fast Tool Servo Detail Design

In this section, we provide the detail design of the prototype fast tool servo. In subsection 2.3.1, we detail the FTS actuator. In subsection 2.3.2, we discuss the

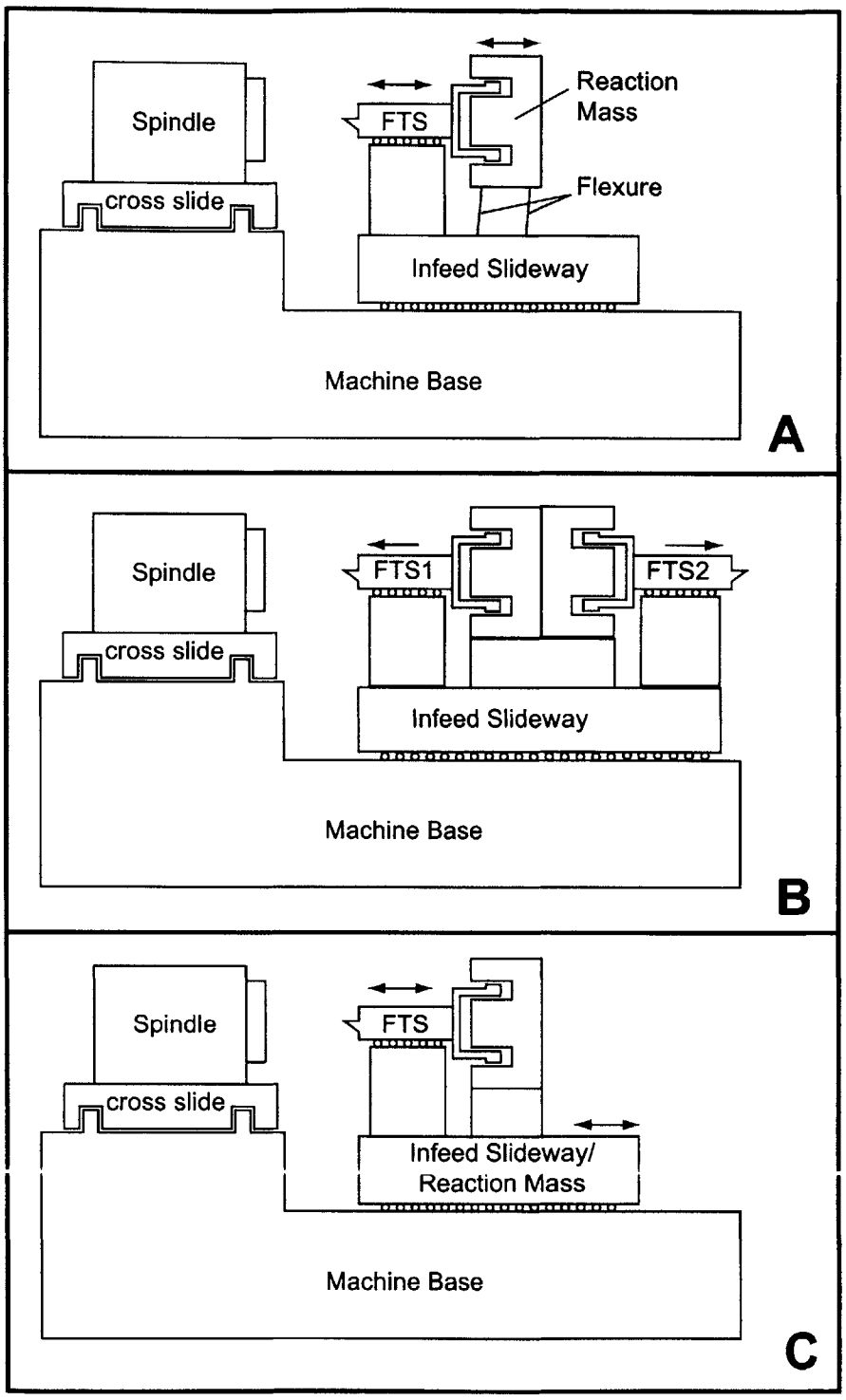


Figure 2-24: Schematic drawing of three reaction force attenuation strategies.

motor amplifier. In subsection 2.3.3, we discuss the air bearing stage used for the FTS. In subsection 2.3.4, we provide the details of the dSPACE 1103 discrete time processor used to control the FTS. In subsection 2.3.5, we detail the implementation of the two different sensors used on the FTS. Lastly in subsection 2.3.6, we detail the assembly of the linear FTS.

2.3.1 Motor

The motor for the linear FTS is an oil-cooled 3-phase linear motor built by Micheal Liebman for his Master's research [46]. The motor was partially constructed by Fred Sommerhalter at Anorad Corporation [6] and utilizes an off-the-shelf 60 mm pitch magnet track. As discussed earlier the motor incorporates coil windings with gaps between the coil layers to allow coolant to flow between the layers. The resulting motor has a maximum steady state force of 350 N (approximately 4 times that of a conventional 3 phase linear motor). The measured force constant K_f for this motor is

$$K_f = 35.4 \frac{\mathrm{N}}{\mathrm{A}_{\mathrm{rms}}}. \tag{2.30}$$

Mobil Velocite No.10, the same oil used in the hydrostatic stages of the Moore machine base, is used to cool the motor (Micheal Liebman used Mobiltherm 603 which has slightly better thermal characteristics) [28]. The oil is pumped using a small centrifugal pump form Gorman-Rupp Industries [33]. The motor temperature is measured by a thermocouple placed between the first and second end-turns of one of the three motor coils. The oil temperature is measured by a thermocouple in the oil reservoir.

2.3.2 Linear Motor Amplifier

The linear motor is driven by a prototype HPA-400-30 amplifier from Copley Controls Corp [19][1]. This is a PWM amplifier with a pulse frequency of 125 kHz. The maximum allowable supply voltage is 400 Volts. The maximum current output I_p is 30 A.

[1]Copley Controls Corp of Canton, MA kindly donated this amplifier

This amplifier has a continuous current rating I_c of 12 A. Thus the maximum power is 12 kilowatts while the maximum continuous power is 4.8 kilowatts. The amplifier is tuned for a -3 dB bandwidth of 2 kHz (the amplifier has a maximum bandwidth of 5 kHz). The DC voltage is supplied by a 300 V/16 A Lambda EMS-300-16 D.C. power supply [44] (thus the power available is 4.8 kW maximum and 3.6 kW continuous). The amplifier accepts two phase (U and V) current commands (the current in the W phase is constrained by the U and V currents in a Y configuration motor).

Motor commutation is done on the dSPACE DS1103 control board and is discussed in detail in section 6.1.2.

2.3.3 Air Bearing Slide

The tool holder and motor coils are mounted to a custom 2"x2" porous media air bearing slide from New Way Inc. [68]. The slide consists of eight 50 x 100 mm flat air bearing pads. Each pad has a quoted stiffness of 100 N/μm at 0.41 MPa (60 psi). The pads are assembled into two rectangular bushings 50 mm apart. These bushings support a 2" x 2" x 11.5" anodized aluminium beam. The aluminium beam has a 1.5" diameter hole bored through its length. The beam has a mass of 1.54 kg. The air bearing stage has been machined to accept a Heidenhain LIP 501 linear scale [38]. The slide has a full range travel of 1.5".

2.3.4 Discrete Time Processor

The machine base and linear FTS are controlled using a dSPACE DS1103 PPC controller board [26]. The DS1103 has a 400 MHz PowerPC 604e main processor and a 20 MHz Texas Instruments TMS320F240 slave processor [90]. The system has

- 16 16-bit main processor ADC
- 4 12-bit main processor ADC
- 8 14-bit main processor DAC
- 4 8-bit main processor DAC

- 6 digital incremental encoder interfaces
- 1 main processor serial RS232/RS422 serial interface
- 32 bits main processor digital I/O
- 16 10-bit slave ADC
- 18 bits slave digital I/O.

Appendix D list all of the connections used to run the prototype diamond turning machine. The DS1103 is programmed using Matlab's Simulink interface.

2.3.5 Sensor

The prototype FTS was tested in two forms, a bench-top version and the version installed on the prototype diamond turning machine. The primary differences between the two versions are the linear diffraction style linear encoder used, the motor attachment, and the plumbing and electrical connections. The motor attachment and connections are dealt with in subsection 2.3.6. A Heidenhain LIP 501 exposed linear scale [38] was used during bench-top testing while a MicroE M3500Si Mercury series incremental encoder [59]

The Heidenhain LIP 501 utilizes a 4 μm pitch scale with a full travel accuracy of $\pm 1\mu$m. The LIP 501 encoder electronic outputs 11 μA_{pp} sinusoids with a 1 kΩ load. The encoder has a maximum velocity of 1 m/s. The encoder electronics are isolated using an Analog Devices AD621 low drift instrumentation amplifier with a gain of 10 [5]. A differential signal is created using a Texas Instruments SN74LS14 Hex Schmitt-trigger inverter [90]. The encoder output is interpolated using an Aerotech MXH250 multiplier [3][2]. With x1000 interpolation, the Heidenhain encoder has 4 nm resolution. The Aerotech multiplier supports speeds up to 0.128 m/s. The differential output of the Aerotech multiplier is read onto the DS1103 control board through one of the incremental encoder interfaces (max. input frequency 6.6 MHz/66 mm/s).

[2]Donated by Aerotech Inc of Pittsburgh PA.

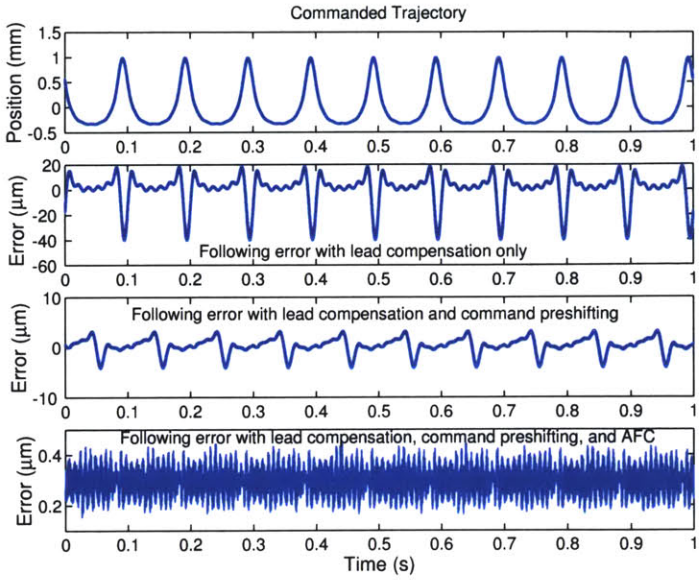


Figure 2-25: Measured prototype fast tool servo following error using a Heidenhain LIP 501 linear encoder.

We used a standard 70 mm glass scale trimmed to a length of 55 mm. The scale was bonded into a slot machined into the 2" x 2" slide using a cyanoacrylate-based adhesive. The read-head is bolted to the slide chassis. The scale/read-head alignment was adjusted using an oscilloscope in X-Y format.

Figure 2-25 shows a typical measured following error for the bench-top prototype fast tool servo using the Heidenhain LIP 501 linear encoder. Under AFC control, the measured following error is 0.18 μm peak-to-peak and 0.05 μm rms. While the accuracy/resolution of the Heidenhain encoder were excellent, the slow speed of the DS1103 encoder interface made this sensor system unacceptable for machine level testing.

For the machine level testing we used a MicroE M3500Si serial interface linear encoder. The M3500si uses a 20 μm pitch glass scale. We used a standard accuracy L55 scale with a full range travel accuracy of ± 3 μm. Both the scale and the read-head of the MicroE encoder are much smaller than the Heidenhain encoder. Fortunately the geometry was such that the scale and read-head could be mounted using the existing Heidenhain mounting features with the use of a spacer to mount the scale

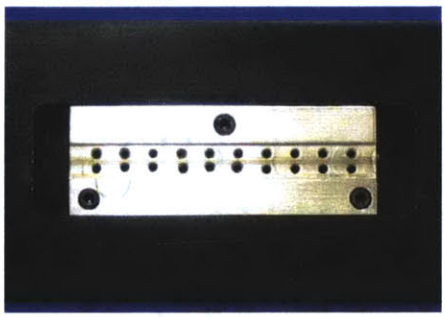


Figure 2-26: Photo of MicroE glass scale mounting.

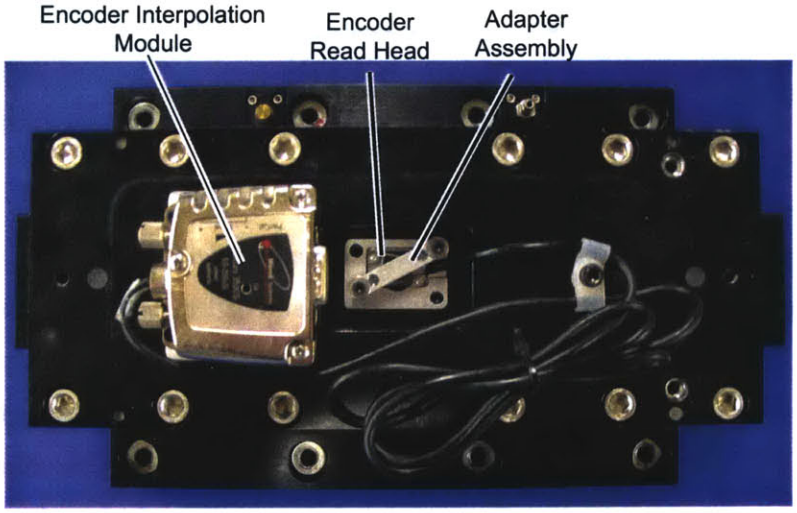


Figure 2-27: Photo of MicroE glass scale mounting.

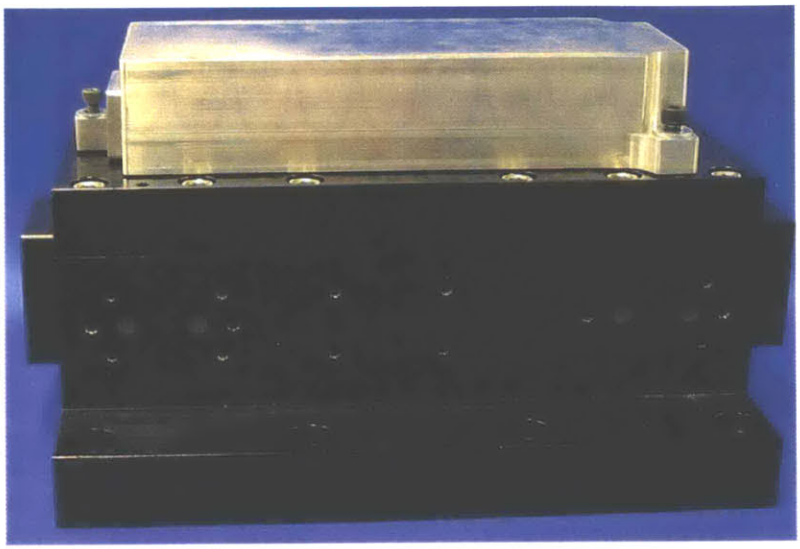


Figure 2-28: Photo of air bearing stage with encoder cover.

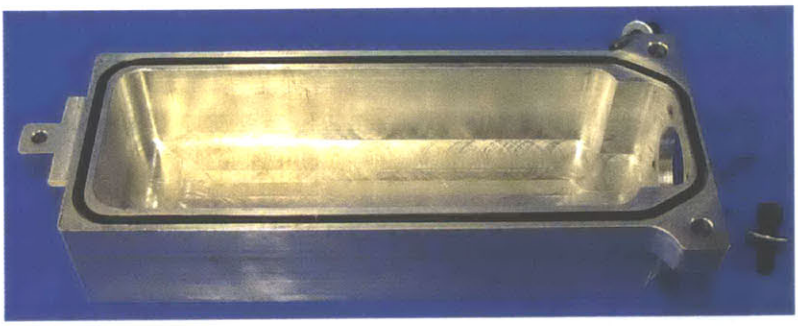


Figure 2-29: Photo of bottom side of the encoder cover.

and a spacer and clamp system to hold the read-head in place. Figure 2-26 shows a photo of the MicroE scale mounting. The scale is bonded to the aluminum adapter plate with a cyanoacrylate base adhesive and the adapter plate is screwed to the stage beam. Figure 2-27 shows a photo of the MicroE read-head and encoder interpolation electronics. The read-head is clamped to an adapter plate which is then screwed down using the original Heidenhain mounting points. The interpolation electronics are also mounted to the slide chassis. This entire installation is sealed from contamination by an o-ring sealed cap (Shown in photo in Figures 2-28 and 2-29).

The M3500Si interpolation electronics provide up to x4096 interpolation (4.8 nm

resolution) at a maximum speed of 7.2 m/s. The M3500Si serial interface outputs a 38-bit word. The first 8-bits refer to the encoder status while the next 30-bits are a twos-complement position word. The MicroE serial interface utilizes RS-422 differential voltage levels. Since the DS1103 serial interface utilizes single ended TTL signals, a National Semiconductor DS3486 Quad RS-422/RS-423 line receiver converts the received data to single ended form [66]. While a National Semiconductor DS3487 Quad Tri-State line driver is used to convert the DS1103 TTL single ended output to RS-422 format. To read the serial data from the encoder electronics, we tried three different implementations on the DS1103:

1. A S-function serial interface running on the DS1103 slave processor utilizing C functions provided by dSPACE.

2. A S-function serial interface running on the main processor utilizing main processor C functions provided by dSPACE.

3. A S-function serial interface running on the slave processor utilizing user provided C functions.

S-functions provide a means of introducing capabilities to a dSPACE I/O board via Simulink. When a S-function is utilized a custom block is added to the Simulink model, the S-function C code is incorporated into the existing Simulink model, and the appropriate dSPACE hardware is accessed. S-functions are typically written using a S-function template provided by dSPACE. For more information on S-functions refer to [2]. All of these implementations were made possible by the work of Marsette Vona [96] and David Otten [71]. Specifically, all of the slave DSP programming is based upon the code developed by David Otten.

In the first implementation, we utilized the standard S-Function structure to create a serial interface on the DS1103 Slave DSP. The S-function utilized the standard Slave DSP Serial Interface functions provided by dSPACE [1]. As for all S-functions, most of the overall processing occurs on the main DSP processor. This serial interface has the following flow process for each computational cycle:

1. The main DSP initializes the communication buffer with the Slave DSP.
2. The main DSP activates the encoder serial interface
3. The main DSP instructs the Slave DSP to read the serial interface.
4. The Slave DSP reads in 16-bits and transfers the data to the communication buffer.
5. The main DSP reads the communication buffer and stores the last 8 bits (Note: the first 8-bits are the encoder status which is not required to determine the position).
6. The Slave DSP reads in 16-bits and transfers the data to the communication buffer.
7. The main DSP reads the communication buffer and adds the 16-bits to the previous data.
8. This repeats until the main DSP has formed a 30-bit position word.
9. The main DSP converts the 30-bit word from 2 complement and outputs a position count to the Simulink model.

This implementation is very stable and robust but very slow (the turnaround time is 140 μs minimum resulting in a 7kHz maximum sample rate). The reason this implementation is so slow is that the slave DSP is very slow moving data from the input register to the communication buffer (about 20 μs per transfer). In this implementation, the slave DSP must transfer a total of 3 words to the communication buffer. One way to speed the serial interface up is to completely avoid using the communication buffer.

In our second implementation, we utilize the main processor digital I/Os to the create a serial interface. The main processor digital I/Os are grouped in 8-bit units. Each 8-bit group can be configured as either input or output. As part of the overall diamond turning machine control, groups 1 and 3 are configured for output while

group 2 is configured for input (group 4 is unassigned). For this implementation, we have used IO19 from group 3 to enable the serial interface and IO20, also group 3, to generate the serial clock. IO15 from group 2 is used to read in the serial data. In this implementation, the serial interface has the following process flow:

1. Main processor enables the serial interface, IO19 high.
2. Main processor creates clock pulse, IO20 toggled from low to high to low.
3. Repeat step 2 seven times.
4. Main processor toggles clock.
5. Main processor reads IO20 and stores bit.
6. Repeat steps 4 and 5 until complete 30-bit position word is read.
7. The main processor converts the 30 bit word from 2 complement and outputs a position count to the Simulink model.

It takes 0.4 μs to generate each clock pulse and it take 1.2 μs to generate a clock pulse, read the input bit, and then store the input bit. Including the conversion from two complement and the output of the position word to Simulink, this implementation has a total turnaround time of 42 μs. This serial interface implementation is by far the fastest of the three tested. The issue with this implementation is that while the serial interface is running the main DSP is not available for other processes. Running all the other task required for control of the DTM takes 60 μs, which results in a total turnaround time of 102 μs which results in a maximum sample rate of 9.4 kHz. While this serial interface is stable and fast, the overall sample rate is marginal for our application (we would prefer a sample rate in the range of 20 kHz). This leads to our third implementation.

In our third and final implementation, we run the serial interface on the slave DSP using user defined C functions. This implementation has several advantages over the implementation using the dSPACE provided functions. As noted earlier, the slow part of the slave processor serial interface is the transfer of the serial data from the

input register to the communication buffer. Using the canned C functions, we are required to transfer 3 separate words to the communication buffer despite the fact that our total word length is 30-bits and the communication buffer transfers 16-bits at a time. Eliminating the extra word transfer can potentially save us 20 μs. Four files are required to run a custom C function. A file containing the code for the S-function, a file to register the user defined function on the slave DSP, a file containing the user defined function C code, and a header file for the user defined function. Appendix C contains the four files used for this serial interface implementation.

The process flow for the serial interface interface running on the slave DSP utilizing user defined functions is:

1. Main processor outputs the position word from the previous computational cycle to Simulink.
2. Main processor reads the position word gather by the slave DSP during the previous computational cycle from the communication buffer.
3. Main processor converts the 2 complement position word to absolute position.
4. Main processor request a new position word from the slave DSP.
5. Slave DSP enables encoder serial interface.
6. Slave DSP toggles serial clock 8 times.
7. Slave DSP toggles serial clock, reads and stores input bit 16 times.
8. Slave DSP transfers 16-bit word to communication buffer.
9. Slave DSP toggles serial clock, reads and stores input bit 12 times.
10. Slave DSP toggles serial clock 4 times to clear encoder serial buffer.
11. Slave DSP transfers the 12-bit word to the communication buffer in 16-bit format.
12. Main processor outputs the position word to Simulink.

As is fairly obvious from the process flow, this serial interface results in a 2 unit delay on the position data available to the position control loop. The main processor delay is determined by the interaction between the user defined S-function code and the overall real time control code defined by Simulink. Attempts to eliminate the main processor delay resulted in non-operational S-functions. The one unit delay in the communication between the slave DSP and the main processor is intentionally introduced to ensure that the entire position word is properly transferred. When we attempted to both read and transfer the position word from the slave DSP to the main processor on the same computational cycle, the main processor would often attempt to read the position word before the slave DSP had finished reading and transferring the data to the communication buffer. Thus the word read into the main processor was incomplete. It takes the slave DSP 60 μs to read and transfer the serial data to communication buffer. This implementation was successfully tested at sample rates as high as 14.5 kHz. For most data in this thesis, the DS1103 was running at 12.5 kHz.

Figure 2-30 shows a typical measured following error for the DTM mounted prototype FTS using the MicroE M3500Si linear encoder for position measurement. With the exception of the different encoders, the controller and hardware for the Heidenhain data set (Figure 2-25) and the MicroE data set (Figure 2-30) are identical (Note: the trajectory for the Heidenhain data had a maximum acceleration of 2 g's while the MicroE trajectory had a maximum acceleration of 6.6 g's). The measured following error is 3.1 μm peak-to-peak and 0.28 μm rms using the MicroE scale. The measured error has gone up because while the MicroE system has higher resolution than the Heidenhain scale, it also has a greater interpolation error. Interpolation error is due to asymmetry in the sinusoid diffracted from the encoder scale [27]. High quality scales typically have an interpolation error of 1% the scale pitch. Thus for the Heidenhain LIP 501, the interpolation error is on the order of 0.04 μm. The M3500Si utilizes a 20 μm pitch scale which results in an interpolation error on the order of 0.2 μm. How exactly the interpolation error maps to the measured position error is not explored in detail here but clearly to achieve the accuracy we would like, the M3500Si

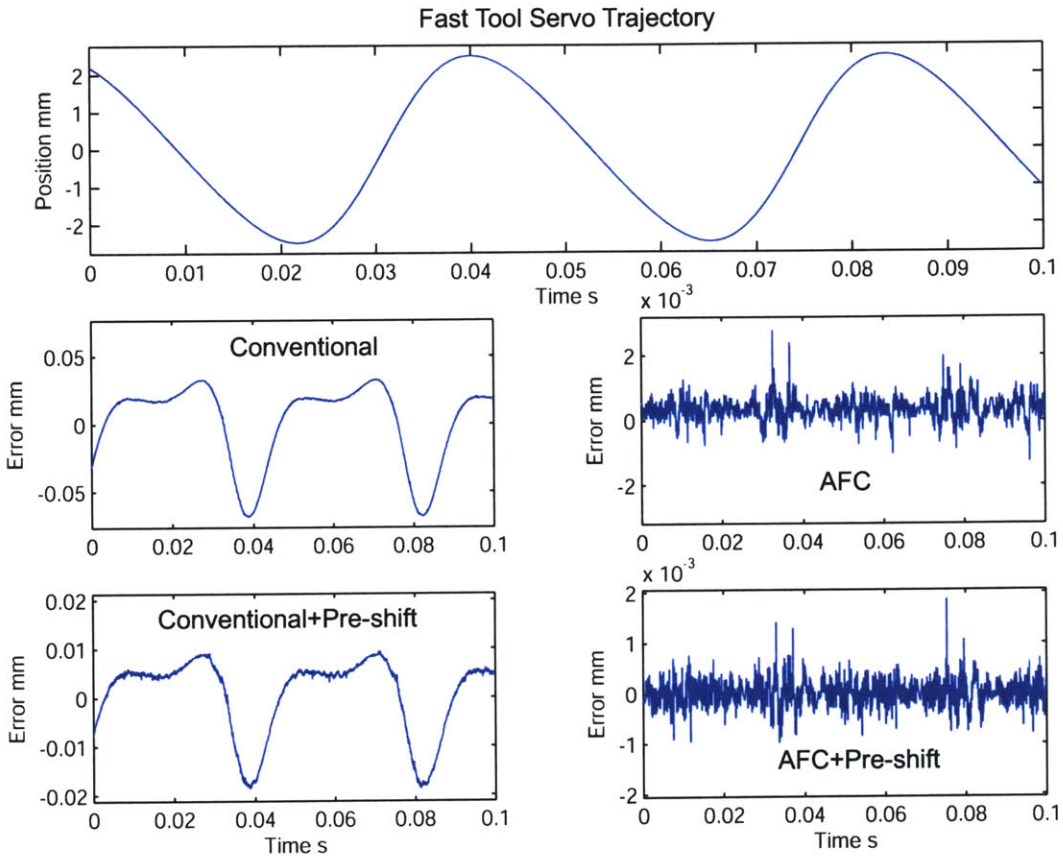


Figure 2-30: Measured prototype fast tool servo following error using a MicroE M3500Si linear encoder.

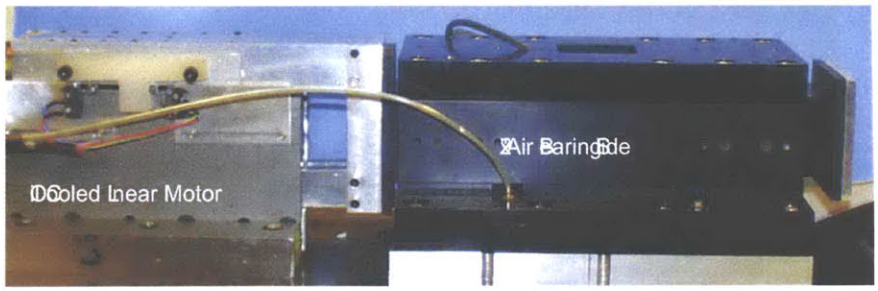


Figure 2-31: Photo of bench-top prototype long stroke fast tool servo.

scale should be mapped using a laser interferometer and the measured position data corrected.

In this section, we explored two different linear scale systems. In one case, we implemented a scale with excellent accuracy performance but with a low overall maximum velocity. In the second case, we implemented a scale with a very high maximum velocity but with an accuracy lower than desired. In retrospect since the maximum velocity of the Heidenhain system is limited by the interpolation electronics and the DS1103 incremental encoder interface, we should simply have replaced the interpolation electronics with faster electronics and a serial interface.

In the next subsection, we evaluate the detail design of the prototype FTS hardware.

2.3.6 Prototype Assembly

As mentioned earlier, the prototype FTS was tested in two forms, a bench-top version and the version installed on the diamond turning machine. The primary differences between the two versions are the diffraction style linear encoder used, the motor attachment, and the plumbing and electrical connections. The differences between the sensors was discussed in section 2.3.5. In this section, we will discuss the mechanical differences between the two prototype versions.

Figure 2-31 shows a side view photo of the bench-top prototype FTS while Figure 2-32 shows a side view photo of the prototype FTS mounted to the diamond

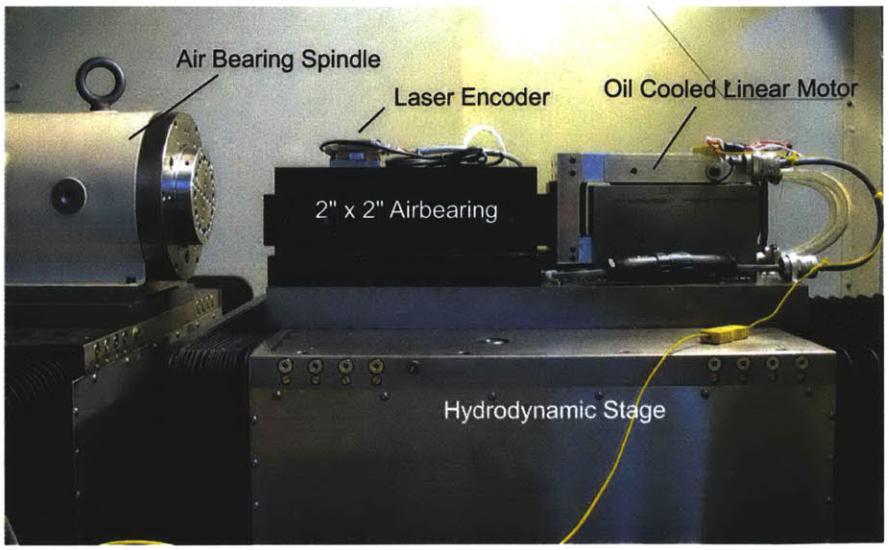


Figure 2-32: Photo of the prototype long stroke fast tool servo mounted to the Moore Nanotechnologies machine base.

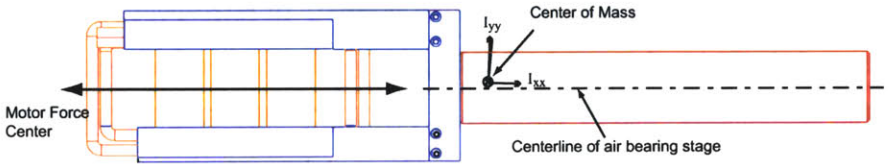


Figure 2-33: ProE model of bench-top prototype with the motor force center aligned with the air bearing centerline.

turning machine. There are several obvious differences between the two prototypes. The first clear difference is the length of the magnet track. In the bench-top prototype, the magnet track is 24 inches long allowing for a motor travel of 19 inches. Since space was not available on the in-feed stage and we only needed a travel of 1.5 inches, the magnet track was trimmed to 6.5 inches. This is the shortest we could make the magnet track while not cutting any of the permanent magnets and still allow for 1.4 inches of travel.

The second obvious difference is the vertical and horizontal position of the motor relative to the air bearing slide has changed. In the bench-top prototype, the magnet track of the motor sits approximately 0.5" below and 2" behind the the air bearing

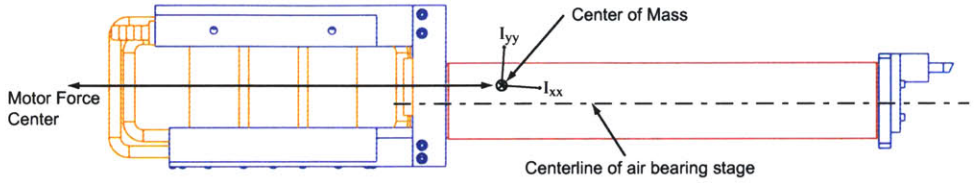


Figure 2-34: ProE model of machine mounted prototype with the motor force center aligned with the center of mass of the moving FTS components.

stage. In the machine mounted prototype, the magnet track is in the same horizontal position but is now nearly level with the air bearing stage. In the bench-top prototype, the motor coils were positioned such that the motor center of force acted in-line with the air bearing stage centerline (shown in Figure 2-33). In the machine mounted prototype, the motor force center has been positioned such that it acts through the center of mass of the moving components (shown in Figure 2-34). In the machine mounted prototype, the motor coil is mounted much closer (1.25 inches) to the air bearing slide. This change simply reflects that I was uncertain about the desired coil location in the bench-top prototype and placed the coils much further away from the air bearing stage. The coils in the machine mounted prototype have been moved as close to the air bearing as possible while allowing for 1.5 inches of motion. Drawings of the key components require to assemble the prototype FTS are in appendix E.

The prototype FTS is mounted to the hydrostatic slide using an adapter plate. The adapter plate is a 8 x 19.75 x 1.5 inch piece of steel plate with the top and bottom faces ground for flatness and parallelism. Figure 2-35 shows an isometric view of the ProE model of the adapter plate. The adapter plate has features which allow for the attachment of the air bearing slide, the motor magnet track, the motor coolant hoses, and the motor power cables. The adapter plate also incorporates feature to capture the coolant oil leaked from the motor. Complete drawings for the adapter plate are in appendix E

Figure 2-36 shows a ProE model of the complete prototype fast tool servo/hydrostic stage assembly. Figure 2-37 shows a rear view photo of the fast tool servo mounted to on the Moore Nanotechnologies machine base. In this design, we have done our best

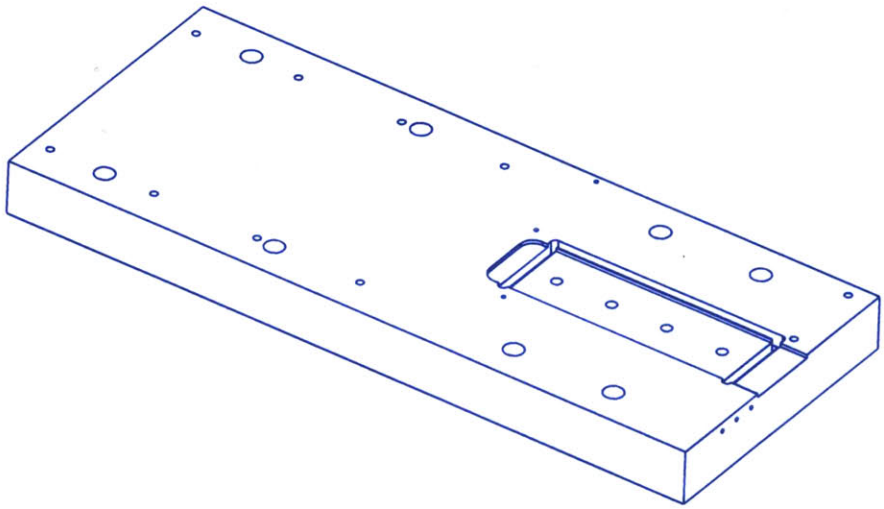


Figure 2-35: ProE model of the adapter plate used to mount the prototype FTS to the hydrostatic slide.

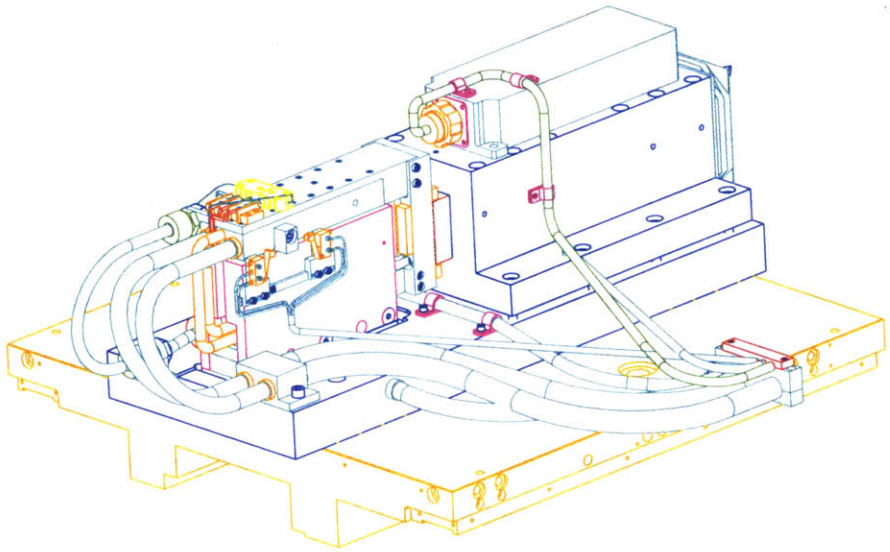


Figure 2-36: ProE model of the prototype fast tool servo mounted to the hydrostatic stage.

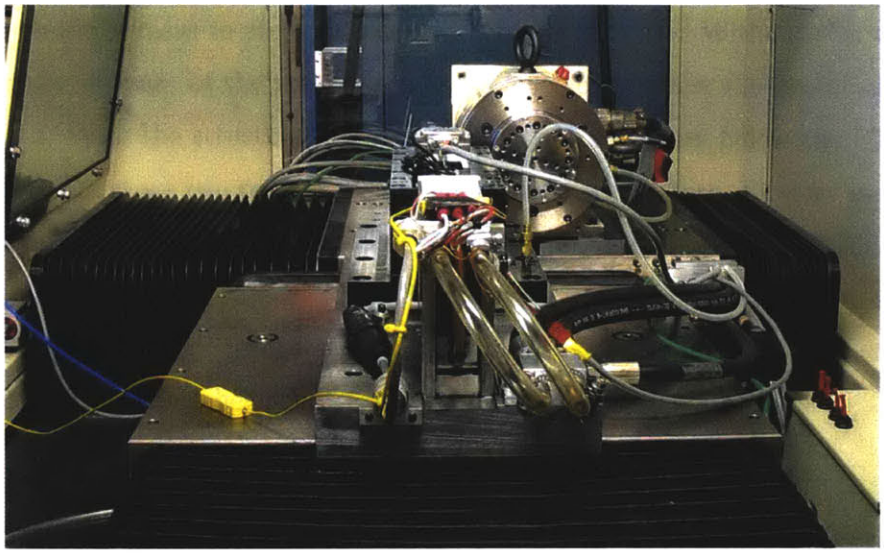


Figure 2-37: Rear view photo the prototype linear fast tool servo mounted on the Moore Nanotechnologies machine base.

to minimize the forces exerted on the moving portion of the fast tool servo. As can be seen in both Figures, the oil coolant tubing and the electrical connections have been carefully routed to avoid excessive flexion. The motor power cable is 3 conductor 18 AWG Alpha Wire **XTRA*GUARD** high flexibility cable in places where high flexibility is required and 3 conductor 16 AWG Alpha Wire communication cable elsewhere [4]. For the coolant oil plumbing, half inch **TYGON** tubing is used in regions of high flexibility while 3/8" medium pressure SAE hydraulic tubing is used in low flex regions. The air supply to the air bearing is 1/4" **TYGON** tubing and utilizes a quick disconnect from Beswick Engineering [9].

One of the weakness of the prototype FTS/hydrostatic stage assembly as we have built it is that the overall center of mass of the stage is not vertically aligned with the center of mass of the FTS. This means that while the hydrostatic stage is able to isolate the machine base from the FTS actuation forces, the relative motion of the two centers of mass do introduce a torque into the machine base. To avoid this we designed but did not implement a structure to raise the center of mass of the FTS/stage assembly. Figure 2-38 shows a ProE model with a structure designed to shift the

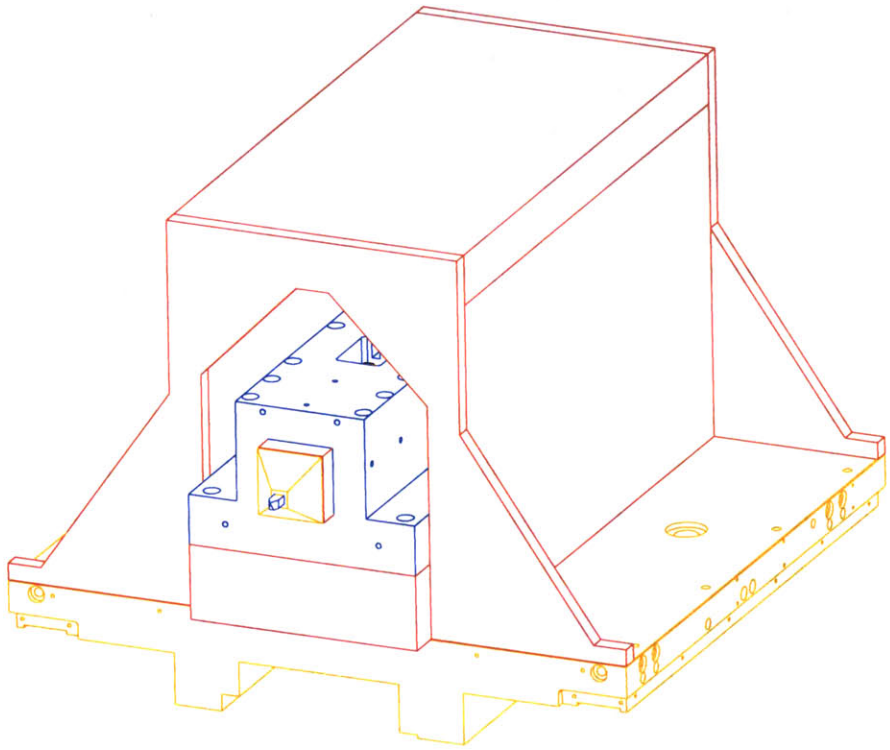


Figure 2-38: ProE model of the prototype fast tool servo mounted to the hydrostatic stage with a structure designed to shift the center of mass location.

stage center of mass to align with the FTS center of mass. Without this structure, the center of mass of the stage assembly is 5.6 inches below the center of mass of the FTS. With the structure the center of mass is 2.9 inches below the FTS center of mass. There are two reasons we did not build this structure. First by introducing this structure, we complicate the task of routing all of the FTS connections. Second, this structure is quite massive (200+ lbs depending on configuration) and brings the total load mass close to the stage carrying capacity. In the end, we decided the complications outweighed the benefits.

2.4 Summary

In this chapter, we reviewed the design details of two long stroke FTS. Specifically, we introduced a rotary fast tool servo concept with improved dynamics. We then reviewed the detailed design of our linear long stroke FTS with integral balance mass. In the next chapter, we will review the various control strategies used to control fast tool servos and related systems.

Chapter 3

Fast Tool Servo Control

This chapter is intended as general background and motivation for our research into adaptive feedforward cancellation (AFC) and oscillator amplitude control (OAC) which is detailed in chapters 4 and 5 respectively. This Chapter is broken into three parts. In section 3.1, we briefly describe the control requirements for FTS and show a typical FTS control structure. In section 3.2, we give a brief overview of a repetitive control implementation using memory loops. In section 3.3, we give a brief overview of repetitive control using AFC control.

3.1 Fast Tool Servo Control

As mentioned in chapter 1, FTS are used for two purposes:

1. Cancel out repetitive error introduced into a part during the machining process.
2. Machine complex geometries into a workpiece.

In both of these applications the FTS toolpaths are highly correlated with the angular position of the part. Furthermore, the desired toolpath can be represented as a summation of sinusoids at integer multiples of the spindle's rotational frequency. The class of control algorithms most suitable for this application are referred to as repetitive control systems. Repetitive control systems are based upon the Internal Model Principle (IMP) [29] where the repetitive controller contains a model of the

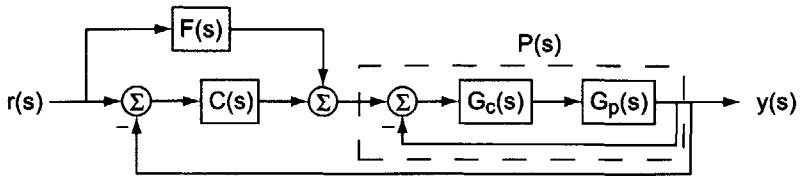


Figure 3-1: Block diagram of a typical fast tool servo control system.

reference or disturbance signal. If the repetitive controller has an exact model of the signal, the control system will have perfect track following and disturbance rejection. From a loop shaping perspective, perfect track following requires infinite loop gain at the tooltrack frequencies. Just as in the case of integral and derivative control, if improperly implemented repetitive control will destabilize the overall control loop. To ensure a stable controller implementation, FTS control systems usually employ multiple control loops. In general, FTS control systems consist of a conventional (PID, Lead-Lag, Pole-zero placement, etc.) inner loop which results in a well defined plant and a repetitive control outer loop. Most repetitive controllers have infinite gain at the harmonics of the spindle frequency and finite gain elsewhere. Since the repetitive controllers typically have a large phase shift between each of the spindle harmonics, the finite gain of the repetitive controller is intentionally low to ensure stability. Thus in a system with repetitive control only, the tracking error is very small at the spindle harmonics but quite large everywhere else. To improve general tracking performance, most FTS control systems incorporate a feedforward channel.

Figure 3-1 show a block diagram of a typical FTS control system. The block diagram elements are

$r(s)$ = the reference input.

$y(s)$ = the plant output.

$G_p(s)$ = the plant transfer function.

$G_c(s)$ = a conventional controller (PID, Lead-Lag, etc.).

$C(s)$ = an adaptive controller.

$F(s)$ = a feedforward filter.

$P(s)$ = the transfer function of the conventionally controlled inner-loop.

The adaptive controller $C(s)$ can either be a sliding mode controller or a repetitive controller. The inclusion of the feedforward filter $F(s)$ is meant to improve the general system tracking response. $F(s)$ can have several different forms. In the most common implementation $F(s) = 1$ and thus acts as a feed-through channel. This implementation is very robust since the general system response will be that of the conventionally controlled inner-loop with the additional control authority from the adaptive controller $C(s)$. In other systems $F(s)$ is used as a feedforward filter. In theory if

$$F(s) = P^{-1}(s) \tag{3.1}$$

this control system would have perfect trajectory tracking at all frequencies. The main difficulty with this approach is systems with non-minimum phase zeros in the plant $P(s)$. When $P(s)$ is inverted, these non-minimum phase zeros become unstable. In addition, completely inverting the plant may result in a non-causal system. The most common approach to implementing a feedforward controller is the Zero Phase Error Tracking Controller (ZPETC) introduced by Prof. Masayoshi Tamizuka in [91]. The ZPETC algorithm operates by cancelling the cancellable poles and zeros, and adjusting for the phase of the non-cancellable zeros. The ZPETC method of feedforward control was applied to a hydraulic FTS by Tsu-Chin Tsau [95] and to a piezoelectric FTS by Marc Crudele [20]. An alternate feedforward approach is to take advantage of the fact that the FTS trajectory is a summation the sinusoids at integer harmonics of the spindle frequency. In this case, we can directly produce a feedforward signal

$$r^*(t) = \sum_{1}^{n} |P(j\omega_i)|^{-1}[A_i \cos(\omega_i t - \angle P(j\omega_i)) + B_i \sin(\omega_i t - \angle P(j\omega_i))] \tag{3.2}$$

where A_i and B_i are the magnitude of $\cos(\omega_i t)$ and $\sin(\omega_i t)$ in $r(t)$. This method is termed Command Pre-shifting and has been successfully employed on the LODTM at Lawrence Livermore National Laboratory [80] and on Stephen Ludwick's rotary FTS [54]. We use the Command Pre-shifting method of Feedforward control.

As mentioned earlier, the adaptive controller $C(s)$ is either a sliding mode controller or a repetitive controller. Sliding mode control combines a feedback controller, with both a feedforward controller and a disturbance observer to form a controller which adaptively minimizes the tracking error. To learn more about applying sliding mode control to FTS refer to [36] and [102].

In the case where $C(s)$ is a repetitive controller, $C(s)$ may be implemented in two fashions. In the case of controllers where the trajectory is predetermined and repeatedly supplied (robotic assemblers for example), repetitive control is run in a semi-open loop fashion known as *iterative learning control.* For more information on iterative learning control and it's similarities to closed-loop repetitive control see Richard Longman's paper on the subject [49]. For closed-loop repetitive controllers, there are two common implementations:

1. Memory-Loop repetitive control.

2. Adaptive Feedforward Cancellation (AFC).

In section 3.2, we briefly review memory loop repetitive control. In section 3.3, we review the background behind AFC control and show some related control structures.

3.2 Memory-Loop Repetitive Control

Figure 3-2 shows a block diagram of a continuous-time repetitive controller $C(s)$ implemented using a memory-loop. This system has a transfer function of

$$C(s) = \frac{1}{1 - e^{-sT_p}} \tag{3.3}$$

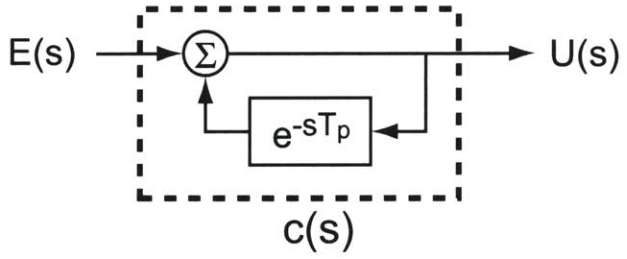


Figure 3-2: Block diagram of a continuous time repetitive controller using memory loops.

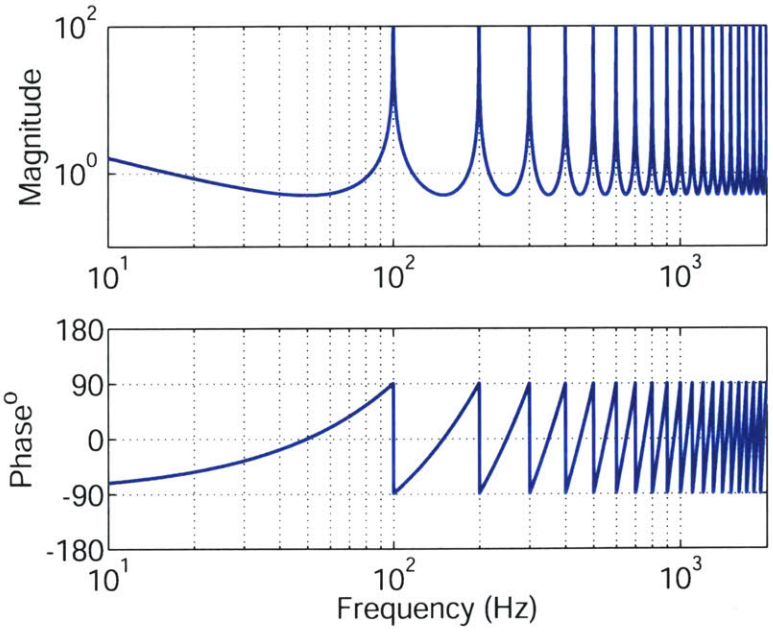


Figure 3-3: Frequency response plot for a continuous time repetitive controller as expressed in (3.3).

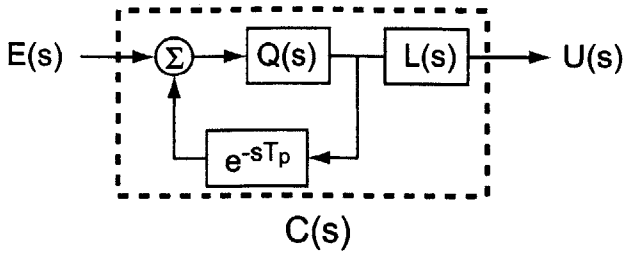


Figure 3-4: Block diagram of a continuous time repetitive controller using memory loops.

where T_p is the period of the frequency we wish to cancel. This transfer function results in complex pole pairs on the imaginary axis when $s = j\omega_i$ where

$$\omega_i = \frac{2\pi i}{T_p};\ i = 0, \pm 1, \pm 2, ..., \pm\infty. \tag{3.4}$$

Thus this transfer function has infinite gain at all integer harmonics of the fundamental frequency ω. Figure 3-3 shows the frequency response plot for a repetitive controller of the form (3.3) with $T_p = 0.01$ s. As we can see, this controller results in a series of infinitely high peaks at integer harmonics of the 100 Hz fundamental. Looking at the phase portion of Figure 3-3, we see that each peak has an associated phase drop of $-180°$. Since the plant we are applying repetitive control to has an associated phase and magnitude shift, applying a controller of the form (3.3) results in closed-loop stability issues.

Figure 3-4 shows a block of a continuous-time repetitive controller $C(S)$ implemented using a memory-loop and two filters $Q(s)$ and $L(s)$ [88]. $Q(s)$ limits the working frequencies of the repetitive controller while $L(s)$ compensates for the phase shift from $Q(s)$. The transfer function of this modified structure is now

$$C(s) = \frac{L(s)Q(s)}{1 - Q(s)e^{-sT_p}}. \tag{3.5}$$

Figure 3-5 shows the frequency response of this modified structure with $T_s = 0.01$

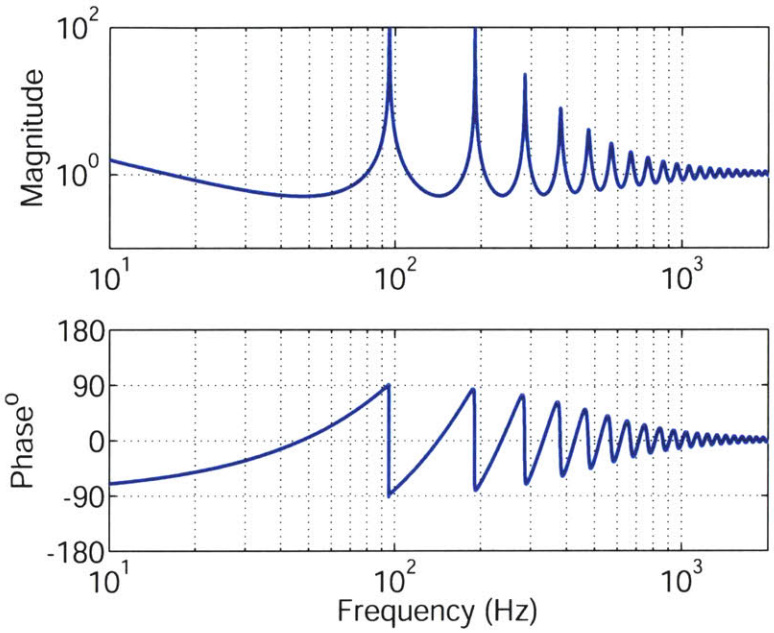


Figure 3-5: Frequency response plot for a continuous time repetitive controller as expressed in (3.5) with $Q(s)$ and $L(s)$ from equations (3.6) and (3.7).

and

$$Q(s) = \frac{\omega_f^2}{s^2 + 2\zeta\omega_f s + \omega_f^2} \tag{3.6}$$

$$L(s) = \frac{s^2 + 2\zeta\omega_f s + \omega_f^2}{\omega_f^2} \tag{3.7}$$

where $\omega_f = 1000\pi$ and $\zeta = 0.7$. As we can see, the introduction of the two filters has attenuated both the magnitude and the phase shift of the higher harmonics of memory-loop. For more detail on the selection of $Q(s)$ and $L(s)$ please see [92] and [39].

A more common approach to implementing memory-loop repetitive control is to implement the memory-loop digitally. Figure 3-6 shows the block diagram of a discrete-time repetitive controller using memory-loops. The transfer function for this

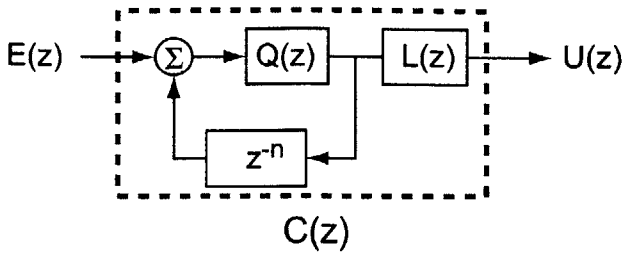


Figure 3-6: Block diagram of a discrete-time repetitive controller using memory loops.

discrete-time implementation is

$$C(s) = \frac{L(z)Q(z)}{1 - Q(z)z^{-n}} \tag{3.8}$$

where n is the nearest integer value of

$$n = \frac{T_p}{T_s} \tag{3.9}$$

where T_p is the period of the frequency we wish to cancel and T_s is the sample time. Controllers of this form have been successfully implemented on FTS by Tsao [95], Rasmussen [74], and Crudele [20]. More generally memory-loop repetitive controllers have been used to cancel periodic disturbances in disk drives [78] and robotic manipulators [58]. For a more comprehensive list of applications see [54].

The great advantage of memory-loop repetitive controllers is that they are computationally very efficient. A single memory-loop will cancel all harmonics of the fundamental frequency. There are two disadvantages to the memory-loop approach. The first disadvantage is that continuous-time memory-loop repetitive controllers contain all harmonics of the fundamental out to infinity (discrete-time memory-loop repetitive controllers contain all harmonics to the Nyquist frequency). To ensure stability, we need to introduce a low-pass filter to eliminate the higher harmonics. This low-pass filter attenuates the magnitude of both the low and intermediate frequency peaks. Thus a while a memory-loop controller may perfectly cancel the fundamental frequency, it's performance degrades at the intermediate frequencies. The second

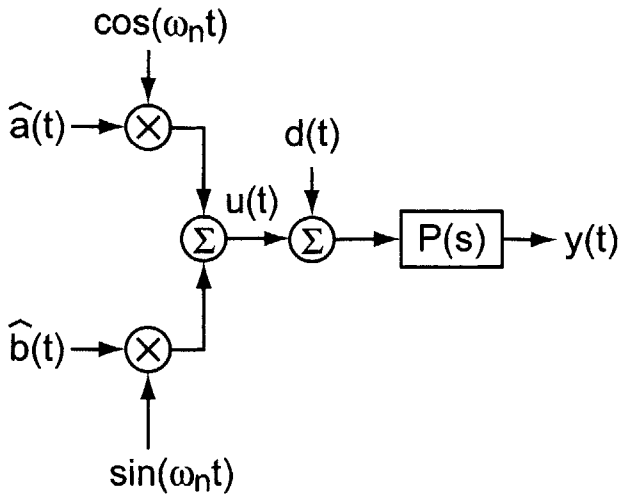


Figure 3-7: Block diagram of an AFC system with zero error.

disadvantage is that memory-loop controllers are not robust to perturbations in the frequency of the disturbance. For example, if we design a memory-loop controller to cancel disturbances at 60 Hz but the actual part spins at 58 Hz, the memory-loop has no mechanism to incorporate this shift. In [88], Maarten Steinbuch proposes a memory-loop repetitive controller which is more robust to changes in frequency.

An alternative approach to forming repetitive controller is to use adaptive feedforward cancellation. Each AFC resonator has the advantage of cancelling only a single frequency and when applied in the canonical form can be driven by the measured spindle speed making the control response more robust to variations in the spindle speed. The disadvantage of the AFC structure is that it is much more computationally intensive. In the next section, we present a brief look at AFC control. In chapter 4, we present a more detailed analysis of AFC control from a loop shaping perspective.

3.3 AFC Repetitive Control

Adaptive feedforward cancellation is a control strategy based upon the Internal Model Principle (IMP) [29]. The IMP essentially states that for a controller to exactly cancel

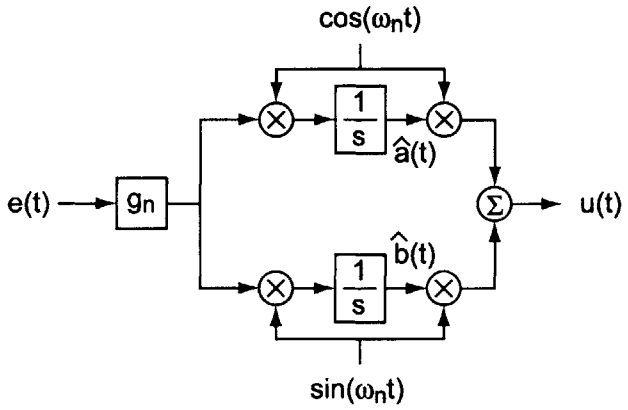


Figure 3-8: Block diagram of an AFC controller as proposed by Bodson et.al.

a disturbance it must contain a model of the disturbance signal. For control systems designed to reject sinusoidal disturbances of the form

$$d(t) = a_n \cos(\omega_n t) + b_n \sin(\omega_n t), \tag{3.10}$$

the controller must be able to adaptively estimate and reproduce the magnitude and phase of the disturbance. Figure 3-7 shows a block diagram of an AFC controller with zero tracking error. For this system to have no error the magnitude estimates $\hat{a}$ and $\hat{b}$ must equal the magnitude of the disturbance input:

$$\hat{a} = a_n \tag{3.11}$$

$$\hat{b} = b_n. \tag{3.12}$$

To form the magnitude and phase estimates, an AFC controller uses modulation with a pair of sinusoids to detect for error components at the resonant frequency. The detected error is then filtered to form the magnitude estimates.

Figure 3-8 shows the block diagram of the earliest form of AFC control proposed by Bodson et.al. [11]. In this form, the AFC controller uses the same sinusoid to both detect the error component at ω_n and generate the cancellation signal $u(t)$. The magnitude of the error component is estimated by integrating the error component at

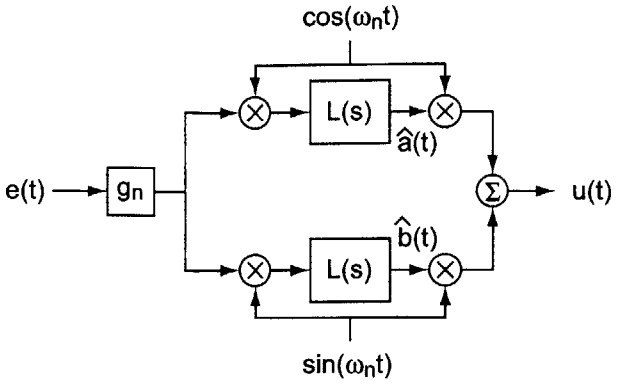


Figure 3-9: A more general block diagram for a LTI adaptive feedforward system.

ω_n. As we will discuss in more detail in chapter 4, this structure has an LTI equivalent form of

$$C(s) = \frac{g_n s}{s^2 + \omega_n^2}. \tag{3.13}$$

Thus this form of AFC control places a zero at the origin and a pair of poles on the imaginary axis at $\pm j\omega_n$. For frequencies below ω_n, the controller introduces a 90^o phase shift. At ω_n, the phase instantly drops 180^o to -90^o. Thus this form of AFC control may be stably applied only to systems with a phase $|\phi| < 90^o$. There are several methods of extending AFC control to more general systems.

Figure 3-9 shows a more general block diagram of an adaptive feedforward system. In this case, the integrators have been replaced with transfer function $L(s)$. For adaptive feedforward systems, $L(s)$ is typically a low pass filter. Using a low pass filter of the form

$$L(s) = \frac{1}{s + \sigma} \tag{3.14}$$

results in this structure having an LTI equivalent of

$$C(s) = \frac{g_n(s + \sigma)}{s^2 + 2\sigma s + (\omega_n^2 + \sigma^2)}. \tag{3.15}$$

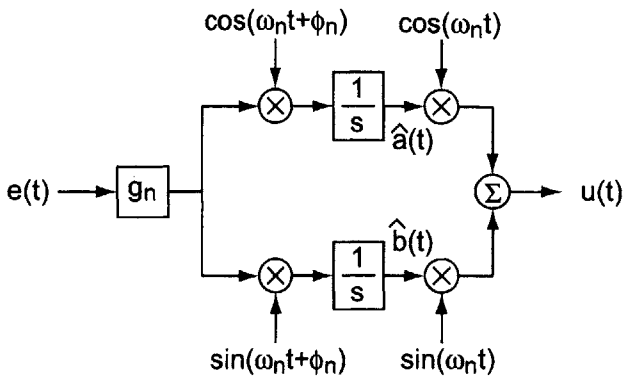


Figure 3-10: Block diagram of an AFC system with phase shift.

As we can see, this system offers enhanced stability by moving the system poles off of the imaginary axis, thus smoothing the phase transition, but this is at the expense of perfect error tracking. As we can see in equation (3.15), the resonant frequency of the adaptive system is no longer ω_n, thus this oscillator cannot perfectly cancel disturbances at ω_n. For more information on adaptive feedforward systems with low-pass filters see [8].

Figure 3-10 shows a block diagram of an AFC system where the stability is enhanced using a phase shift parameter ϕ_n. As will be shown in chapter 4, this block diagram has the LTI equivalent of

$$C(s) = \frac{g_n(s\cos\phi_n + \omega_n \sin\phi_n)}{s^2 + \omega_n^2}. \tag{3.16}$$

This system still places the pole pair on the imaginary axis, but offers enhanced stability by allowing the designer to place the system zero anywhere along the real axis. The development of this structure is detailed by Messner and Bodson in [57]. In the next chapter, we will present a loop shaping perspective for selecting g_n and ϕ_n. This structure has the advantage of enhanced stability but has the disadvantage of requiring the generation of four sinusoids. An equivalent structure titled Higher Harmonic Control (HHC) was concurrently developed lowers the number of required sinusoids to two.

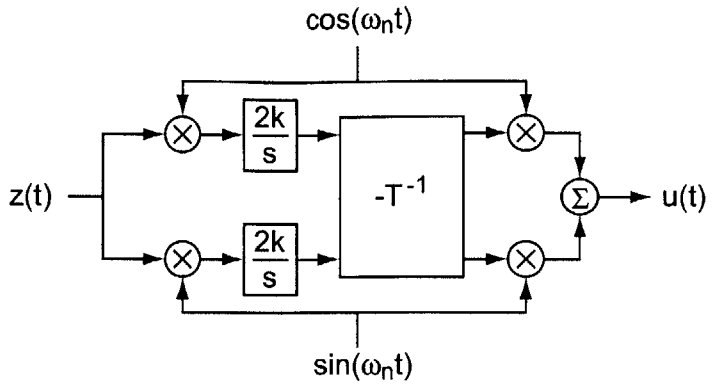


Figure 3-11: Block diagram of a continuous-time Higher Harmonic Controller. Figure adapted from Hall and Wereley [37].

Higher Harmonic Control was developed to cancel vibrations in helicopters due to variations in the rotor blade loads. It was first developed by McHugh and Shaw [56] and Shaw and Albion [81] as a discrete-time algorithm. Figure 3-11 shows a block diagram of a continuous-time HHC controller as developed by Hall and Wereley [37]. In this controller the phase of the controller is determined by the rotation matrix $\mathbf{T}$. For higher harmonic control $\mathbf{T}$ is given by

$$\mathbf{T} = \begin{bmatrix} T_{cc} & T_{cs} \\ T_{sc} & T_{ss} \end{bmatrix}, \tag{3.17}$$

where

$$T_{cc} = T_{ss} = Re[G(j\omega_n)] \tag{3.18}$$

$$T_{cs} = -T_{sc} = Im[G(j\omega_n)] \tag{3.19}$$

and $G(s)$ is the transfer function of the system to be controlled. This structure is equivalent to within a gain factor of that shown in Figure 3-10 if the AFC phase advance is set as $\phi_n = \angle P(j\omega_n)$, where $P(j\omega_n)$ is the transfer function of system to be controlled by the AFC resonator.

Figure 3-12 shows a block diagram of an Automatic Vibration Rejection algorithm

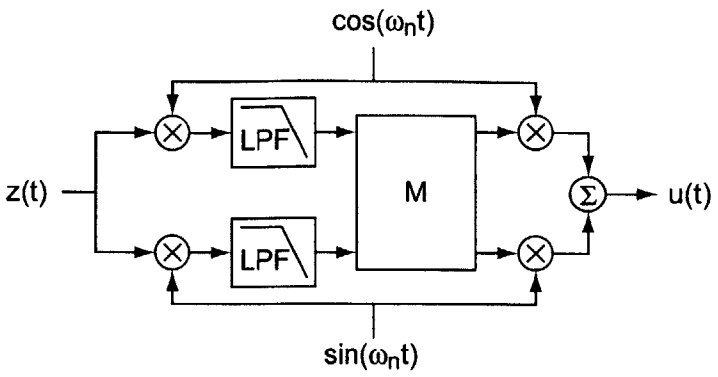


Figure 3-12: Automatic Vibration Rejection algorithm in an AFC equivalent form. Figure adapted from [15].

in an AFC equivalent form (Figure adapted from [15]). Automatic Vibration Rejection (AVR) is a self-tuning disturbance rejection system for use on magnetic bearings. This structure has been used widely in magnetic bearings, and was recently patented by Tamisier *et al* [89]. In this control structure, Tamisier *et al* have employed the general form for adaptive feedforward controller where the integrator has been replaced by a general purpose low-pass filter. Just as in the HHC structure, the phase of the control structure is determined by a rotation matrix **M**. For AVR the rotation matrix is defined as

$$\mathbf{M} = \begin{bmatrix} \cos\psi & -\sin\psi \\ \sin\psi & \cos\psi \end{bmatrix} \tag{3.20}$$

where

$$\psi = \angle S(j\omega) \tag{3.21}$$

and

$$S(s) = \frac{1}{1 + G_c(s)G_p(s)} \tag{3.22}$$

where $G_c(s)$ is a conventional controller and $G_p(s)$ is the transfer function of the

magnetic bearing. As we can see if the an integrator is used as the low-pass filter, this control system is identical to the control structure we use on our FTS. Tamisier *et al* have also developed a perspective on AVR control which is quite similar to the oscillator amplitude control perspective we detail in chapter 5.

3.4 Summary

In this chapter, we have presented our FTS control structure. Our FTS controller consists of a feedforward outer loop, a conventional inner-loop, and a multiple resonator AFC controller. This chapter contained a brief description of the feedforward controller and an introduction of AFC control. In chapter 4, we provide a detailed analysis of AFC control from our loop shaping perspective. The development of the conventional inner loop is presented in chapter 6.

Chapter 4

Adaptive Feedforward Cancellation

In this chapter, we examine in detail a type of repetitive control called Adaptive Feedforward Cancellation (AFC). One of the advantages AFC control has over the techniques described in the previous chapter is that each AFC resonator (the core element of an AFC controller) acts at only a single frequency. This means that the controller can drive the system error to zero at selected frequencies with robust stability properties. Sections 5.1 through 5.4 detail how to tune AFC controllers from a loop shaping perspective. The work presented here is an expanded version of the presentation [12]. In Section 5.5, we introduce an extension of conventional AFC control which we term Amplitude Modulated Adaptive Feedforward Cancellation (AMAFC). AMAFC is intended to reduce errors in trajectories where the commanded amplitudes vary as a function of time.

4.1 Adaptive Feedforward Cancellation

Adaptive feedforward cancellation (AFC) algorithms form a special class of repetitive control. Unlike memory loop repetitive controllers, equations (3.5) and (3.8), they allow the designer to place controller poles at a specific frequency, resulting in an infinite gain at the desired frequency, thus allowing perfect steady-state tracking and disturbance rejection at the resonator frequency. Figure 4-1 shows the continuous time structure of the AFC algorithm used in this research. This form of AFC control first

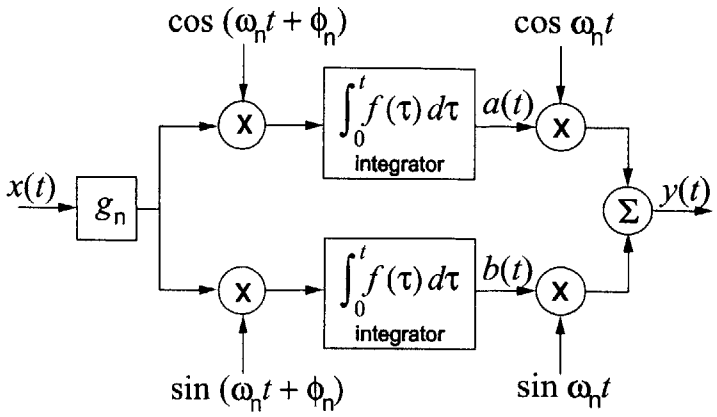


Figure 4-1: Resonator structure which forms the core of the AFC controller.

appeared explicitly in the literature [57], while an equivalent structure called Higher Harmonic Control using a rotation matrix to implement ϕ_n was presented earlier in [37]. It has been shown in [54, 11] that the control structure in Fig. 4-1 with the phase advance parameter ϕ_n set to zero is equivalent to a linear time-invariant (LTI) system of the form

$$\frac{Y(s)}{X(s)} \equiv C_n(s) = g_n \frac{s}{s^2 + \omega_n^2}, \tag{4.1}$$

where the gain g_n is a constant to be determined by the user. This system consists of a complex conjugate pair of pole on the imaginary axis at $s = \pm j\omega_n$ and a single zero at the origin. It has been shown in [57], [37], [12] that with the phase advance parameter included the system is equivalent to

$$\frac{Y(s)}{X(s)} \equiv C_n(s) = g_n \frac{s\cos\phi_n + \omega_n \sin\phi_n}{s^2 + \omega_n^2}, \tag{4.2}$$

where both the gain g_n and the phase advance parameter ϕ_n are to be selected by the user. The system represented by (4.2) also has a complex pole pair on the imaginary axis at $s = \pm j\omega_n$ but allows the designer to place the zero on the real axis at

$$s = -\omega_n \frac{\sin\phi_n}{\cos\phi_n}. \tag{4.3}$$

In principle, the steady state output of (4.1) and (4.2) are equivalent but in practice (4.2) is superior because it offers enhanced stability robustness. The LTI forms in (4.1) and (4.2) are important results because they allow the designer to use classic control techniques (Root Locus plots, Bode plots, Nyquist diagrams, etc.) to determine the closed-loop stability of the system.

In the following subsections, I detail three methods for showing the equivalence between the AFC structure in 4-1 and (4.2)

4.1.1 AFC Equivalence - Time domain approach

In this approach, we base our argument almost completely in the time domain, waiting until the final step to take the Laplace transform. We first published this in [12]. Examining Figure 4-1, the signals entering the integrator blocks are

$$\begin{aligned} \dot{a}(t) &= g_n x(t) \cos(\omega_n t + \phi_n), \\ \dot{b}(t) &= g_n x(t) \sin(\omega_n t + \phi_n). \end{aligned} \tag{4.4}$$

Assuming the system is at rest when $t = 0$, integrating these signals with respect to time t yields

$$\begin{aligned} a(t) &= \int_0^t g_n x(\tau) \cos(\omega_n \tau + \phi_n)\ d\tau, \\ b(t) &= \int_0^t g_n x(\tau) \sin(\omega_n \tau + \phi_n)\ d\tau. \end{aligned} \tag{4.5}$$

The output $y(t)$ is the sum of these signals, $a(t)$ and $b(t)$, as modulated by sinusoids:

$$\begin{aligned} y(t) &= \cos\omega_n t \int_0^t g_n x(\tau) \cos(\omega_n \tau + \phi_n)\ d\tau \\ &\quad + \sin\omega_n t \int_0^t g_n x(\tau) \sin(\omega_n \tau + \phi_n)\ d\tau. \end{aligned} \tag{4.6}$$

Bringing the sinusoids into the integrals and combining the terms yields

$$y(t) = \int_0^t g_n x(\tau) [\cos(\omega_n \tau_n + \phi_n) \cos\omega_n t$$

$$+\sin(\omega_n\tau + \phi_n)\sin\omega_n t]\ d\tau. \tag{4.7}$$

The term in brackets is the trigonometric identity for the cosine of a difference; and thus

$$y(t) = \int_0^t g_n x(\tau)\cos(\omega_n[t-\tau] - \phi_n)\ d\tau. \tag{4.8}$$

This integral has the form of a convolution, namely

$$\begin{aligned} y(t) &= x(t) * g_n\cos(\omega_n t - \phi_n) \\ &= x(t) * g_n[\cos(\omega_n t)\cos(\phi_n) + \sin(\omega_n t)\sin(\phi_n)] \end{aligned} \tag{4.9}$$

The analysis is completed by taking the Laplace transform of both sides, noting that the convolution of the time signals corresponds to multiplication of the Laplace transforms. Thus,

$$Y(s) = X(s)\ g_n\frac{s\cos\phi_n + \omega_n\sin\phi_n}{s^2+\omega_n^2}, \tag{4.10}$$

which gives the form of equation (4.2). This same proof is presented in [8] for more general systems, where arbitrary transfer functions replace the integrator blocks.

4.1.2 AFC equivalence - Laplace Shifting Method

The following is an extension of the analyses performed in [54, 11] which do not include the phase advance parameter ϕ_n. This approach depends upon the shifting property of Laplace transforms

$$\begin{aligned} \mathcal{L}\{f(t)\} &= F(s) \\ \mathcal{L}\{f(t)e^{s_0 t}\} &= F(s-s_0), \end{aligned} \tag{4.11}$$

and the Euler definitions of sin and cos,

$$\begin{aligned} \cos(\omega_n t + \phi_n) &= \frac{e^{j\phi_n}}{2}e^{j\omega_n t} + \frac{e^{-j\phi_n}}{2}e^{-j\omega_n t} \\ \sin(\omega_n t + \phi_n) &= \frac{e^{j\phi_n}}{2j}e^{j\omega_n t} - \frac{e^{-j\phi_n}}{j2}e^{-j\omega_n t} \end{aligned} \tag{4.12}$$

Eqs. (4.11) and (4.12) together imply that

$$\begin{aligned}\mathcal{L}\{f(t)\cos(\omega_n t+\phi_n)\} &= \frac{e^{j\phi_n}}{2}F(s-j\omega_n)+\frac{e^{-j\phi_n}}{2}F(s+j\omega_n)\\ \mathcal{L}\{f(t)\sin(\omega_n t+\phi_n)\} &= \frac{e^{j\phi_n}}{2j}F(s-j\omega_n)-\frac{e^{-j\phi_n}}{2j}F(s+j\omega_n)\end{aligned} \tag{4.13}$$

From Fig. 4-1, it can be seen that the coefficients $\dot{a}(t)$ and $\dot{b}(t)$ which are the signals entering the integrators can be expressed as

$$\begin{aligned}\dot{a}(t) &= g_n x(t)\cos(\omega_n t+\phi_n)\\ \dot{b}(t) &= g_n x(t)\sin(\omega_n t+\phi_n).\end{aligned} \tag{4.14}$$

Taking the Laplace transforms of these as given by (4.13), and accounting for in the integration with an additional factor of s, yields

$$\begin{aligned}A(s) &= \frac{g_n e^{j\phi_n}}{2}\frac{X(s-j\omega_n)}{s}+\frac{g_n e^{-j\phi_n}}{2}\frac{X(s+j\omega_n)}{s}\\ B(s) &= \frac{g_n e^{j\phi_n}}{2j}\frac{X(s-j\omega_n)}{s}-\frac{g_n e^{-j\phi_n}}{2j}\frac{X(s+j\omega_n)}{s}\end{aligned} \tag{4.15}$$

The output $y(t)$ can be expressed as

$$y(t)=a(t)\cos(\omega_n t)+b(t)\sin(\omega_n t) \tag{4.16}$$

and therefore, the Laplace transform is

$$Y(s) = \frac{A(s-j\omega_n)}{2}+\frac{A(s+j\omega_n)}{2}+\frac{B(s-j\omega_n)}{2j}-\frac{B(s+j\omega_n)}{2j} \tag{4.17}$$

Now, by substituting in for the Laplace transforms of the coefficients $a(t)$ and $b(t)$ from Eq. (4.13), we arrive at

$$\begin{aligned}Y(s) = &\frac{g_n e^{j\phi_n}}{4}\frac{X(s-2j\omega_n)}{s-j\omega_n}+\frac{g_n e^{j\phi_n}}{4}\frac{X(s)}{s+j\omega_n}\\ &+\frac{g_n e^{-j\phi_n}}{4}\frac{X(s)}{s-j\omega_n}+\frac{g_n e^{j-\phi_n}}{4}\frac{X(s+2j\omega_n)}{s+j\omega_n}\end{aligned}$$

$$-\frac{g_n e^{j\phi_n}}{4}\frac{X(s-2j\omega_n)}{s-j\omega_n}+\frac{g_n e^{j\phi_n}}{4}\frac{X(s)}{s+j\omega_n}$$
$$+\frac{g_n e^{-j\phi_n}}{4}\frac{X(s)}{s-j\omega_n}-\frac{g_n e^{-j\phi_n}}{4}\frac{X(s+2j\omega_n)}{s+j\omega_n} \tag{4.18}$$

which can be simplified to

$$Y(s) = \frac{g_n X(s)}{2}\left[\frac{e^{j\phi_n}}{s+j\omega_n}+\frac{e^{-j\phi_n}}{s-j\omega_n}\right]. \tag{4.19}$$

Bringing the terms in the sum into a common denominator, we have

$$Y(s) = \frac{g_n X(s)[e^{j\phi_n}(s-j\omega_n)+e^{-j\phi_n}(s+j\omega_n)]}{2(s+j\omega_n)(s-j\omega_n)}. \tag{4.20}$$

This can be rearranged to be

$$Y(s) = \frac{g_n X(s)}{s^2+\omega_n^2}\left[s\frac{e^{j\phi_n}+e^{-j\phi_n}}{2}+\omega_n\frac{e^{j\phi_n}-e^{-j\phi_n}}{2j}\right], \tag{4.21}$$

and therefore

$$\frac{Y(s)}{X(s)} = g_n\frac{s\cos\phi_n+\omega_n\sin\phi_n}{s^2+\omega_n^2}, \tag{4.22}$$

which is the same result as (4.2) and (4.10).

4.1.3 AFC equivalence - Differential Equation Method

This method is similar to the time domain approach in that we wait until the final step to employ the Laplace transform, but in this analysis we prove the equivalence using a linear constant-coefficient differential equation. Again from Fig. 4-1, we observe that

$$\begin{aligned}\dot{a}(t) &= g_n x(t)\cos(\omega_n t+\phi_n)\\ \dot{b}(t) &= g_n x(t)\sin(\omega_n t+\phi_n)\end{aligned} \tag{4.23}$$

and

$$y(t) = a(t)\cos\omega_n t + b(t)\sin\omega_n t. \tag{4.24}$$

Differentiating Eq. (4.24) with respect to time t yields

$$\dot{y}(t) = \dot{a}(t)\cos\omega_n + \dot{b}(t)\sin\omega_n t - \omega_n a(t)\sin\omega_n t + \omega_n b(t)\cos\omega_n t, \tag{4.25}$$

which, when combined with Eq. (4.23) becomes

$$\begin{aligned}\dot{y}(t) &= g_n x(t)[\cos(\omega_n t + \phi_n)\cos\omega_n t + \sin(\omega_n t + \phi_n)\sin\omega_n t] \\ &\quad -\omega_n[a(t)\sin\omega_t - b(t)\cos\omega_n t].\end{aligned} \tag{4.26}$$

By trigonometric identity, this reduces to

$$\dot{y}(t) = g_n x(t)\cos(\omega_n t - \omega_n t + \phi_n) - \omega_n[a(t)\sin\omega_n t - b(t)\cos\omega_n t], \tag{4.27}$$

or more simply

$$\dot{y}(t) = g_n x(t)\cos\phi_n - \omega_n[a(t)\sin\omega_n t - b(t)\cos\omega_n t]. \tag{4.28}$$

This equation can then be differentiated in time t, to yield

$$\begin{aligned}\ddot{y}(t) &= g_n\dot{x}(t)\cos\phi_n + \omega_n[\dot{b}(t)\cos\omega_n t - \dot{a}(t)\sin\omega_n t] \\ &\quad -\omega_n^2[a(t)\cos\omega_n t + b(t)\sin\omega_n t].\end{aligned} \tag{4.29}$$

Substituting in Eq. (4.23), we have

$$\begin{aligned}\ddot{y}(t) &= g_n x(t)\cos\phi_n + g_n\omega_n x(t)[\cos\omega_n t\sin(\omega_n t + \phi_n) - \sin\omega_n t\cos(\omega_n t + \phi_n)] \\ &\quad -\omega_n^2[a(t)\cos\omega_n t + b(t)\sin\omega_n t].\end{aligned} \tag{4.30}$$

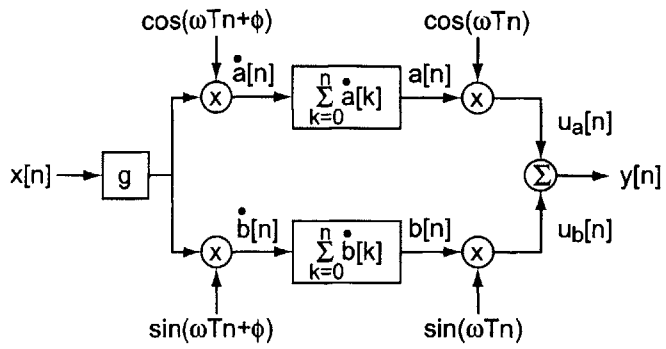


Figure 4-2: Discrete Time AFC structure

The terms in the center brackets can be transformed by trigonometric identity, while the terms in the right brackets are simply $y(t)$ as given in Eq. (4.24). Thus,

$$\ddot{y}(t) = g_n\dot{x}(t)\cos\phi_n + g_n\omega_n x(t)\sin(\omega_n t - \omega_n t + \phi_n) - \omega_n^2 y(t). \tag{4.31}$$

Separating variables, we have

$$\ddot{y}(t) + \omega_n^2 y(t) = g_n(\dot{x}(t)\cos\phi_n + x(t)\omega_n\sin\phi_n). \tag{4.32}$$

Taking the Laplace transform of both sides (assuming zero initial conditions) yields

$$Y(s)(s^2 + \omega_n^2) = g_n(s\cos\phi_n + \omega_n\sin\phi_n)X(s), \tag{4.33}$$

or

$$\frac{Y(s)}{X(s)} = g_n\frac{s\cos\phi_n + \omega_n\sin\phi_n}{s^2 + \omega_n^2}, \tag{4.34}$$

which is identical to the earlier results. A similar analysis is given in [57].

4.1.4 Discrete Time AFC implementation

In general, AFC control is implemented in discrete time. Figure 4-2 shows one discrete time implementation of AFC control. This structure is nearly identical to the

continuous time structure shown in Figure 4-1 except that we have replaced the continuous sinusoids with a discrete sinusoid and the integrators with summers. For the purpose of controller design, we would like to find the LTI system equivalent for this structure. In this section, we present a time domain approach to finding the equivalent structure taking the Z-transform of the signals as a final step. This parallels the earlier continuous time presentation.

In this section, we use the dot notation informally, in that it represents a discrete-time equivalent of differentiation. Also, we have elected not to include the sample time T as a part of the discrete time integration. This means that the appropriate value of g changes with sample rate. This issue is addressed in more detail in chapter 6.

To start with, we note that the signals entering the summers are

$$\dot{a}[n] = gx[n]\cos(\omega Tn + \phi) \tag{4.35}$$

$$\dot{b}[n] = gx[n]\sin(\omega Tn + \phi). \tag{4.36}$$

This means that the signals exiting the summers are

$$a[n] = g\sum_{k=0}^{n} x[k]\cos(\omega Tk + \phi) \tag{4.37}$$

$$b[n] = g\sum_{k=0}^{n} x[k]\sin(\omega Tk + \phi). \tag{4.38}$$

The summer outputs are then modulated by $\cos(\omega Tn)$ and $\sin(\omega Tn)$ to form

$$u_a[n] = g\cos(\omega Tn)\sum_{k=0}^{n} x[k]\cos(\omega Tk + \phi) \tag{4.39}$$

$$u_b[n] = g\sin(\omega Tn)\sum_{k=0}^{n} x[k]\sin(\omega Tk + \phi). \tag{4.40}$$

Moving the modulation signals into the summation yields

$$u_a[n] = g\sum_{k=0}^{n} x[k]\cos(\omega Tk + \phi)\cos(\omega Tn) \tag{4.41}$$

$$u_b[n] = g\sum_{k=0}^{n} x[k]\sin(\omega Tk + \phi)\sin(\omega Tn). \tag{4.42}$$

The controller output $y[n]$ is formed by adding $u_a[n]$ and $u_b[n]$ together to form

$$y[n] = g\sum_{k=0}^{n} x[k]\left(\cos(\omega Tk+\phi)\cos(\omega Tn)+\sin(\omega Tk+\phi)\sin(\omega Tn)\right). \quad (4.43)$$

Using the following trigonometric identity

$$\begin{aligned}\cos(\omega T(k-n)+\phi) &= \cos(\omega Tk+\phi)\cos(\omega Tn)\\ &\quad +\sin(\omega Tk+\phi)\sin(\omega Tn),\end{aligned} \quad (4.44)$$

we can manipulate $y[n]$ such that

$$y[n] = g\sum_{k=0}^{n} x[k]\cos(\omega T(k-n)+\phi). \quad (4.45)$$

Manipulating $y[n]$ further using

$$\cos(\omega T(k-n)+\phi) = \cos(\omega T(k-n))\cos\phi-\sin(\omega T(k-n))\sin\phi, \quad (4.46)$$

results in

$$y[n] = g\sum_{k=0}^{n} x[k]\left(\cos(\omega T(k-n))\cos\phi-\sin(\omega T(k-n))\sin\phi\right). \quad (4.47)$$

Since $\cos\phi$ and $\sin\phi$ are not functions of k, we can move them out of the summation so that

$$y[n] = \cos\phi\sum_{k=0}^{n} x[k]\cos(\omega T(k-n))-\sin\phi\sum_{k=0}^{n} x[k]\sin(\omega T(k-n)). \quad (4.48)$$

It can be noted that the summations are the discrete-time convolution sums [70]

$$\sum_{k=0}^{n} x[k]\cos(\omega T(k-n)) = x[n]*\cos(\omega Tn) \quad (4.49)$$

$$\sum_{k=0}^{n} x[k]\sin(\omega T(k-n)) = x[n]*\sin(\omega Tn), \quad (4.50)$$

resulting in

$$y[n] = g\cos\phi(x[n] * \cos(\omega Tn)) - \sin\phi(x[n] * \sin(\omega Tn)). \tag{4.51}$$

Since convolution in the time domain is the equivalent to multiplication in the frequency domain, we can take the Z-transform of $y[n]$ to find $Y(z)$,

$$Y(z) = g\cos\phi X(z)Z\{\cos(\omega Tn)\} - \sin\phi X(z)Z\{\sin(\omega Tn)\}. \tag{4.52}$$

Manipulating this function into transfer function notation yields

$$\frac{Y(z)}{X(z)} = g\cos\phi Z\{\cos(\omega Tn)\} - \sin\phi Z\{\sin(\omega Tn)\}. \tag{4.53}$$

Substituting in for the Z-transforms, and assuming zero initial conditions yields

$$Z\{\cos(\omega Tn)\} = \frac{z(z - \cos(\omega T))}{z^2 - 2\cos(\omega T)z + 1} \tag{4.54}$$

$$Z\{\sin(\omega Tn)\} = \frac{z\sin(\omega T)}{z^2 - 2\cos(\omega T)z + 1} \tag{4.55}$$

which results in

$$\frac{Y(z)}{X(z)} = g\frac{z^2\cos\phi - z(\cos(\omega T)\cos\phi - \sin(\omega T)\sin\phi)}{z^2 - 2\cos(\omega T)z + 1} \tag{4.56}$$

$$= g\frac{z^2\cos\phi - z(\cos(\omega T + \phi))}{z^2 - 2\cos(\omega T)z + 1}. \tag{4.57}$$

Weerasooriya found a very similar LTI equivalent in [101] but did not include the phase advance parameter ϕ. This LTI equivalent differs significantly from that presented in [54] which is

$$C_i(z) = g_i T\frac{\cos(\omega T)z - 1}{z^2\cos\phi - z(\cos(\omega T + \phi))} \tag{4.58}$$

and the phase adjusted form

$$\frac{U(z^{-1})}{Y(z^{-1})} = g\frac{z^{-1}[\cos(\omega T - \phi) - z^{-1}\cos(\phi)]}{1 - 2\cos(\omega T)z^{-1} + z^{-2}} \tag{4.59}$$

from [57]. Clearly these forms are not equivalent. The difference arises due to differing definitions of the summation opperation. Ludwick and Messner have based their summations on a forward difference approximation to integration

$$y[n+1] = x[n+1] + y[n] \tag{4.60}$$

$$\frac{Y}{X}(z) = \frac{1}{z-1}. \tag{4.61}$$

While Weerasooriya and I use a backwards difference:

$$y[n] = x[n] + y[n-1] \tag{4.62}$$

$$\frac{Y}{X}(z) = \frac{z}{z-1}. \tag{4.63}$$

The resulting transfer functions are equivalent at frequencies near ω, roughly equal for frequencies below ω, and vary greatly at higher frequencies. In general, control systems designers try to limit the highest frequency they try to manipulate to 1/10 the sampling frequency and at low, relative to the sample rate, frequencies there is no real advantage to either approach or difference between the approaches.

In this section, we have derived a discrete-time transfer function , Eq. (4.57), which is analogous to the continuous time transfer function (4.2). In the next section, we detail a loop shaping perspective to selecting resonator phase (ϕ) and gain (g) which is applicable to both the discrete and continuous time representation.

4.2 Adaptive Feedforward Cancellation from a loop shaping perspective

In this section, we develop a loop shaping perspective to tuning AFC controllers. In subsection 4.2.1, we detail the impact of ϕ_n on the loop shape of a system utilizing AFC control and how to select the value of ϕ_n which results in the maximum loop phase margin. In subsection 4.2.2, we evaluate the impact of g_n on the performance of AFC systems. Lastly, we present our method of selecting ϕ_n and g_n for robust controller stability in subsection 4.2.3.

Experimental results applying our tuning method to the rotary fast tool servo for accurate trajectory following are presented in subsection 4.2.4 Additionally, experimental results for our prototype linear FTS are presented in section 7.1. Lastly, experimental results are presented in section 7.3 utilizing our tuning method for machine base acceleration attenuation (section 6.3 details how we adapted AFC control for this purpose).

4.2.1 Phase Advance Parameter ϕ_n

To properly tune a parallel array of AFC resonators, we first need to understand the effects of the phase advance parameter ϕ_n on the transfer function (4.2). We start by examining the effect ϕ_n has on the pole-zero plot and frequency response for a single resonator. Next, we examine the effect of ϕ_n on the pole-zero plot and frequency response of multi-resonator systems with two, three, and ten resonators, respectively. Observations from these systems lead to general tuning rules for systems with any number of parallel AFC resonators.

As noted earlier, a single AFC resonator (4.2) has a single real axis zero and a complex conjugate pair of poles. The position of the zero along the real axis is determined by the values of both ω_n and ϕ_n. In the design process, ω_n is first chosen to match a desired error-nulling frequency. After selecting the desired ω_n, the resonator zero may be placed anywhere along the real axis by selecting ϕ_n. Figure

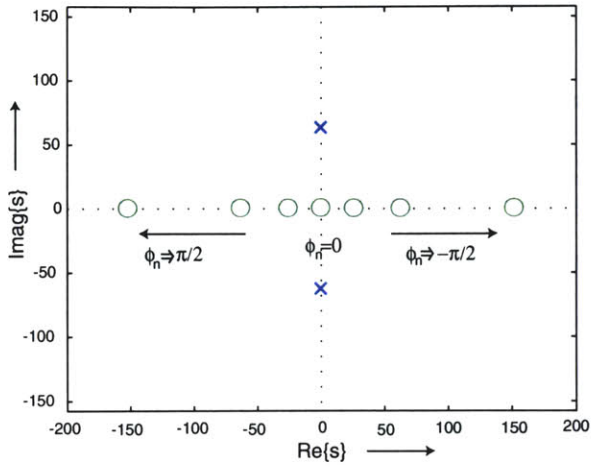


Figure 4-3: Pole-Zero map of single resonator with $-\pi/2 < \phi_n < \pi/2$. Corresponding frequency responses are shown in Figure 4-4 for $-\pi/2 < \phi_n < 0$ and Figure 4-5 for $0 < \phi_n < \pi/2$.

4-3 illustrates the possible zero locations as ϕ_n is varied between $-\pi/2$ and $\pi/2$ with $\omega_n = 20\pi$. The resonator zero is on the positive real axis when $-\pi/2 < \phi_n < 0$, and on the negative real axis when $0 < \phi_n < \pi/2$. Since sin and cos are periodic, the zero location simply wraps along the real axis for phase angles outside the range $-\pi/2 < \phi_n < \pi/2$ with attendant alterations in the sign of the transfer function.

As can be seen in Figures 4-4 and 4-5, the location of the real axis zero affects the frequency response of an AFC resonator in several ways. First, the zero location affects the magnitude curve above and below the resonant peak. These effects are only of secondary importance, as we primarily care about the magnitude near the resonant peak at ω_n; here the magnitude is largely unaffected by ϕ_n. This independence can be seen by taking the magnitude of (4.2) for frequencies near ω_n, i.e., for $s = j\omega_n(1+\epsilon)$ where $|\epsilon| \ll 1$. This yields

$$|C(j\omega_n(1+\epsilon))| = g_n \left| \frac{j\omega_n(1+\epsilon)\cos\phi_n + \omega_n \sin\phi_n}{\omega_n^2 - (1+\epsilon)^2\omega_n^2} \right|$$

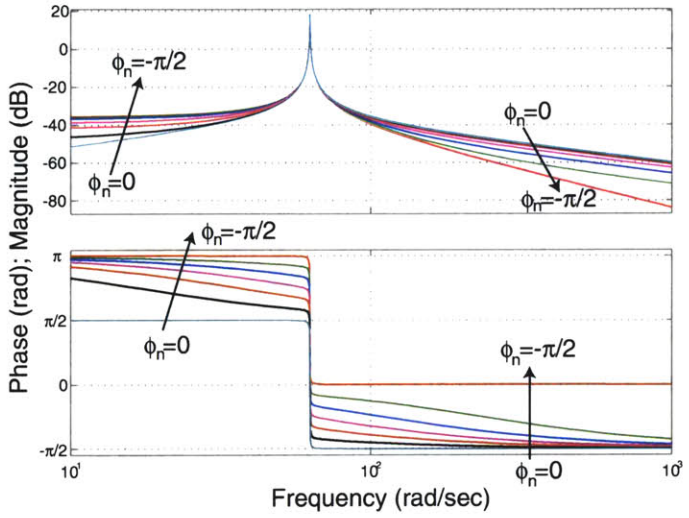


Figure 4-4: Bode plot showing the effect of a negative phase advance parameter ϕ_n on resonator shape. The resonance is centered on 62.8 rad/s (10 Hz).

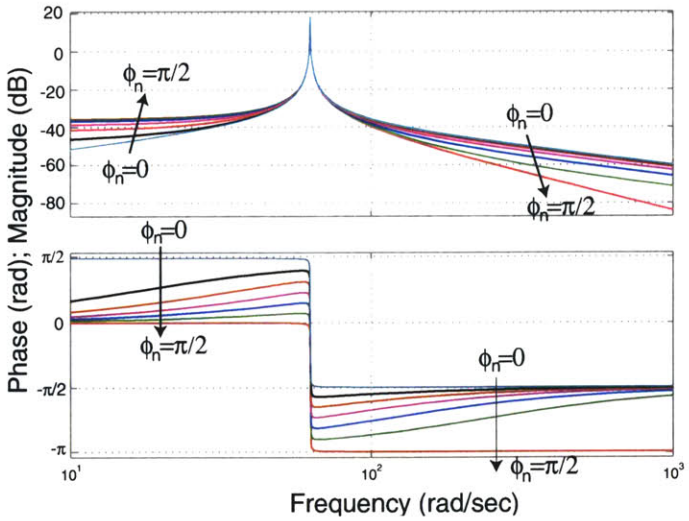


Figure 4-5: Bode plot showing the effect of a positive phase advance parameter ϕ_n on resonator shape. The resonance is centered on 62.8 rad/s (10 Hz).

$$= g_n \frac{\omega_n \sqrt{(1+2\epsilon+\epsilon^2)\cos^2\phi_n + \sin^2\phi_n}}{|\omega_n^2(2\epsilon+\epsilon^2)|}. \tag{4.64}$$

Dropping terms in ϵ^2 and higher simplifies this expression to

$$\begin{aligned} |C(j\omega_n(1+\epsilon))| &\approx g_n \frac{\omega_n \sqrt{(1+2\epsilon)\cos^2\phi_n + \sin^2\phi_n}}{2\omega_n^2|\epsilon|} \\ &\approx g_n \frac{\omega_n \sqrt{1+2\epsilon\cos^2\phi_n}}{2\omega_n^2|\epsilon|}. \end{aligned} \tag{4.65}$$

Employing the approximation $\sqrt{1+\Delta} \approx 1 + \Delta/2$ where $|\Delta| \ll 1$, results in

$$\begin{aligned} |C(j\omega_n(1+\epsilon))| &\approx g_n \left(\frac{1}{2\omega_n|\epsilon|} + \frac{\epsilon\cos^2\phi_n}{|\epsilon|\omega_n} \right). \\ &\approx g_n \left(\frac{1}{2\omega_n|\epsilon|} + \text{sgn}(\epsilon)\frac{cos^2\phi_n}{\omega_n} \right) \end{aligned} \tag{4.66}$$

Since $|\epsilon| \ll 1$ the first term dominates, and this simplifies to

$$|C(j\omega_n(1+\epsilon))| \approx \frac{g_n}{2\omega_n|\epsilon|}. \tag{4.67}$$

This demonstrates that in the vicinity of the resonant peak, the magnitude is not significantly affected by the choice of ϕ_n.

More importantly for design, the location of the real axis zero affects the phase response of the resonator. If the real axis zero is placed in the right half plane ($-\pi/2 < \phi_n < 0$), Figure 4-4, the resonator phase starts at π for $\omega = 0$ and then it decreases to $\pi/2 - \phi_n$ as ω approaches ω_n from below. At ω_n, the phase drops by π to a value of $-\phi_n - \pi/2$, and then asymptotically approaches $-\pi/2$ as $\omega \to \infty$. Similarly if the real axis zero is placed in the left half plane ($\pi/2 > \phi_n > 0$), Figure 4-5, the resonator phase starts at 0 for $\omega = 0$ and then increases to $\pi/2 - \phi_n$ as ω approaches ω_n from below. At $\omega = \omega_n$ the phase drops by π to $-\phi_n - \pi/2$, and then asymptotically approaches $-\pi/2$ as $\omega \to \infty$.

The key feature to note in Figures 4-4 and 4-5 is that the resonator phase discon-

tinuity of $-\pi$ radians at $\omega = \omega_n$ is always centered on $-\phi_n$ for all $-\pi/2 < \phi_n < \pi/2$. This can be seen mathematically by looking at the phase of the numerator and denominator of (4.2). The phase of the numerator at $s = j\omega_n$ can be determined as follows:

$$\begin{aligned} N(j\omega_n) &= j\omega_n \cos\phi_n + \omega_n \sin\phi_n = j\omega_n(\cos\phi_n - j\sin\phi_n) \\ &= j\omega_n e^{-j\phi_n} = \omega_n e^{j\pi/2} e^{-j\phi_n} = \omega_n e^{j(\pi/2-\phi_n)}, \end{aligned} \tag{4.68}$$

and thus

$$\angle N(j\omega_n) = \frac{\pi}{2} - \phi_n \tag{4.69}$$

Since the phase of the denominator is discontinuous at $s = j\omega_n$, it is helpful to define a term $\angle\hat{D}(j\omega_n)$ which is the average angle of the denominator in (4.2) as ω passes through ω_n. That is

$$\begin{aligned} \angle D(j\omega_n^-) &= 0 \\ \angle D(j\omega_n^+) &= \pi \end{aligned} \tag{4.70}$$

where ω_n^- is just below ω_n and ω_n^+ is just above ω_n. With this notation, the average angle of the denominator is

$$\angle\hat{D}(j\omega_n) = \frac{(\angle D(j\omega_n^-) + \angle D(j\omega_n^+))}{2} = \frac{\pi}{2}. \tag{4.71}$$

Combining equations (4.69) and (4.71), we define the average angle of the resonator at $s = j\omega_n$ as $\angle\hat{C}(j\omega_n)$, where

$$\angle\hat{C}(j\omega_n) = \angle N(j\omega_n) - \angle\hat{D}(j\omega_n) = \frac{\pi}{2} - \phi_n - \frac{\pi}{2} = -\phi_n. \tag{4.72}$$

This is the key result which demonstrates the utility of the form 4.2. In this form, the

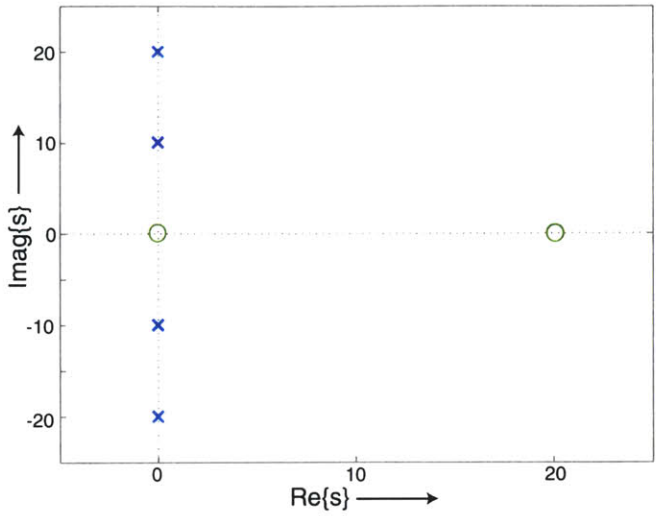


Figure 4-6: Pole-zero plot for two resonators $C_1(s)$ and $C_2(s)$ taken in isolation with $\phi_1 = 0$ and $\phi_2 = -45°$.

average resonator phase at the resonant frequency is directly set by ϕ_n. Said another way, the choice of ϕ_n allows us to set the center of the $-\pi$ phase discontinuity to a desired value, and thereby to maximize the AFC loop phase margin at each resonant peak in a fashion decoupled from the choice of resonator gain g_n.

Analysis becomes more complicated when multiple resonators are employed. First, the pole-zero plot of a multiple resonator system does not provide much design insight since it is not obvious how changing the zero location of a single resonator in a parallel array will affect the zero locations of the combined system. This point can be understood in the context of a two resonator system as follows. Figure 4-6 shows the pole-zero plot for two resonators taken in isolation with $\phi_1 = 0$ and $\phi_2 = -45°$. The first resonator has a complex conjugate pair of poles at $\pm 10j$ sec^{-1} and a real axis zero at the origin. The second resonator has a complex conjugate pair of poles at $\pm 20j$ sec^{-1} and a real axis zero at 20 sec^{-1}. Figure 4-7 shows the pole-zero plot for the same two resonators in parallel, $C(s) = C_1(s) + C_2(s)$. In the parallel system, the controller poles are in the same location but the system now has a complex conjugate pair of zeros as well as a real axis zero. This is not surprising, since when systems

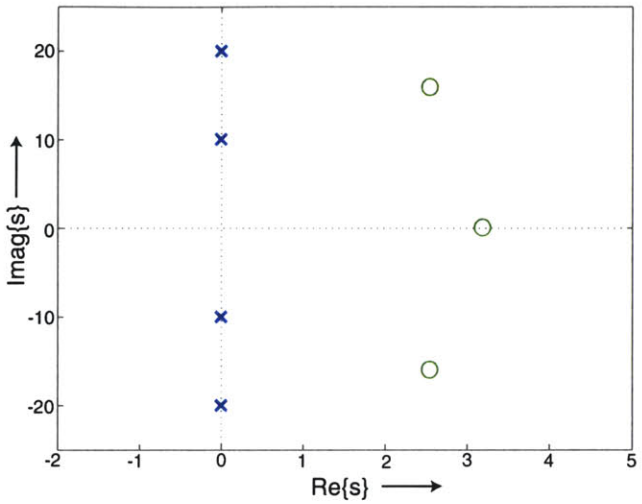


Figure 4-7: Pole-zero plot for two resonator blocks taken in parallel $C(s) = C_1(s) + C_2(s)$ with $\phi_1 = 0$ and $\phi_2 = -45°$.

are additively combined, zero locations are not preserved.

Figure 4-8 shows a pole-zero plot for three resonators taken in isolation, where $\phi_1 = \phi_2 = 0$ and $\phi_3 = -45°$. Figure 4-9 shows the pole-zero plot for the same three resonators in parallel, $C(s) = C_1(s) + C_2(s) + C_3(s)$. Once again, the poles remain in the same location, but the real axis zeros have been transformed into two pairs of complex conjugates and a single real axis zero.

Figure 4-10 shows the pole zero plot for the ten resonator controller developed later in the section (ϕ_n, g_n, and ω_n listed in Table A.2). These diagrams clearly illustrate the difficulty in adjusting the phase advance of an individual resonator from the pole-zero plot of a multiple resonator system. We find however that a frequency domain perspective does allow a rational design process, and the direct tuning of individual resonator parameters as described below.

The frequency response of a multiple resonator system yields more design insight than the pole-zero plot. In a system with N resonators added in parallel, the frequency response contains N resonant peaks, $N-1$ local minima between the resonant peaks, and N phase discontinuities of $-\pi$ radians associated with the resonances. From

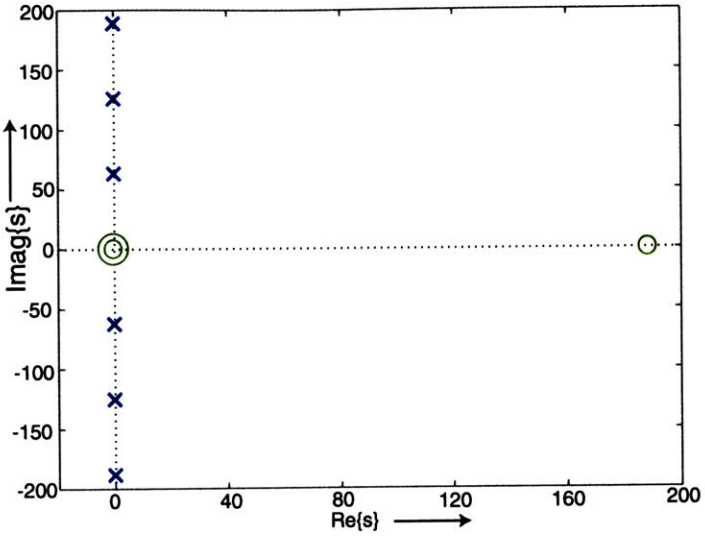


Figure 4-8: Pole-zero plot for three resonator blocks $C_1(s)$, $C_2(s)$, and $C_3(s)$ taken in isolation with $\phi_1 = \phi_2 = 0$, and $\phi_3 = -45°$.

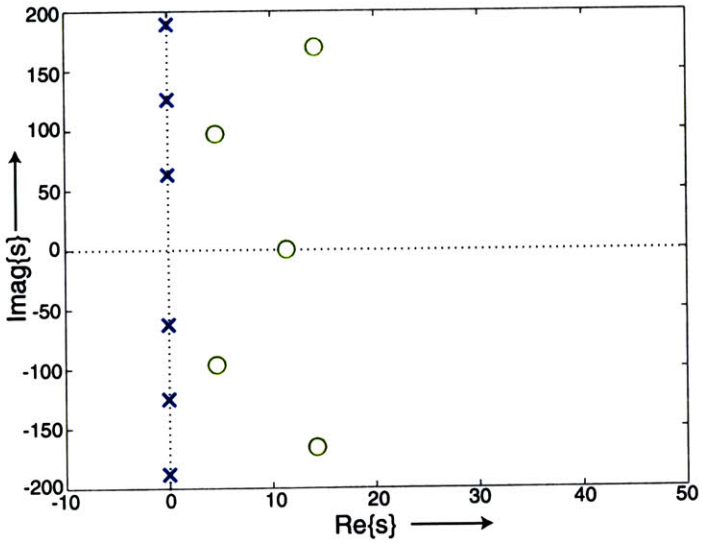


Figure 4-9: Pole-zero plot for three resonators in parallel $C(s) = C_1(s)+C_2(s)+C_3(s)$ with $\phi_1 = \phi_2 = 0$, and $\phi_3 = -45°$.

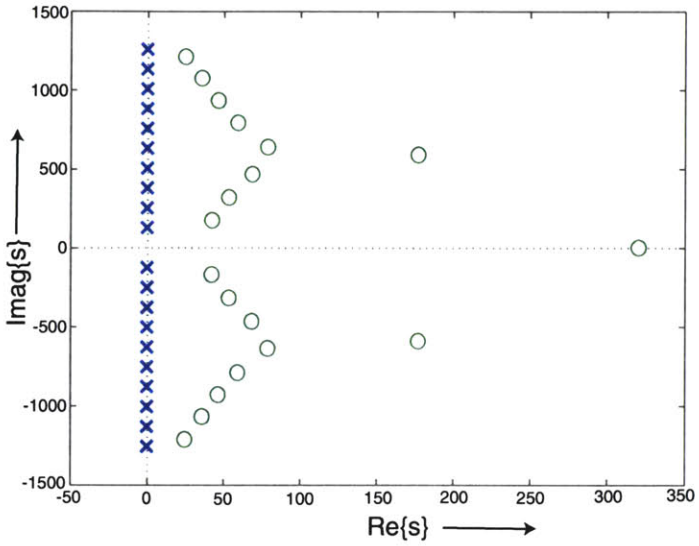


Figure 4-10: Pole-zero plot of C(s) for 10 resonators in parallel. The parameters ϕ_n, g_n, and ω_n for this system are listed in Table A.2.

a loop-shaping perspective the critical issue is how the AFC loop Nyquist plot is influenced by the choice of the g_n and ϕ_n parameters. Within this context, we show that the ϕ_n parameters can be simply chosen on the basis of the phase of $P(s)$ evaluated at ω_n. The gain parameters g_n are then chosen to set the gain margin associated with each of the magnitude minima between the resonant peaks.

To understand this tuning process it is helpful to examine the frequency response characteristics of parallel resonator arrays as ϕ_n is varied. Figure 4-11 shows a Bode magnitude plot for a two resonator system where the phase difference ($\Delta\phi = \phi_1 - \phi_2$) between the first resonator phase advance ϕ_1 and the second resonator phase advance ϕ_2 is varied over 0, $-\pi/2$, and $-\pi$. The figure shows that the magnitude of the local minima is maximized when $\Delta\phi = -\pi$, since for this choice the two resonators are in phase at the local minima. Conversely, a sharp notch results when $\Delta\phi = 0$, since for this choice the two resonators are in anti-phase at the point where their magnitudes are equal.

Figure 4-12 illustrates how for $\Delta\phi = \pm\pi$ the $-\pi$ phase drop of the first resonator at ω_1 results in a system with the two resonators having equal phase shift between

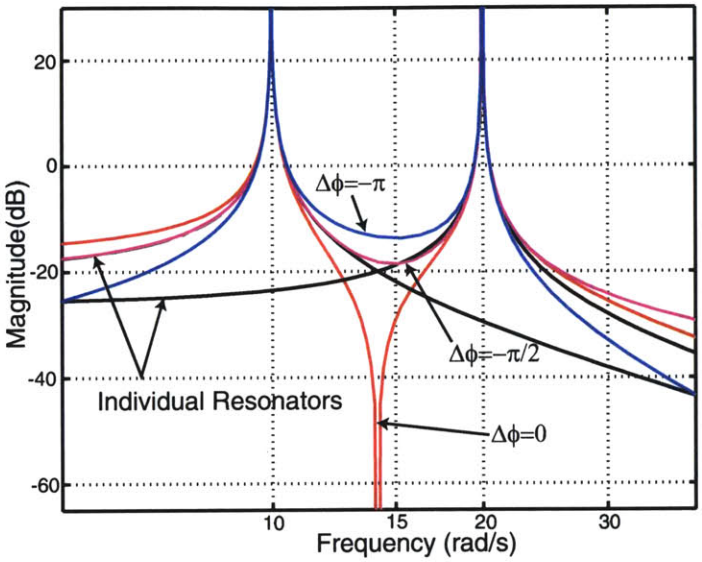


Figure 4-11: Bode magnitude plot for a 2 resonator system with the phase angle difference ($\Delta\phi$) between the two resonators varied from 0 to -π

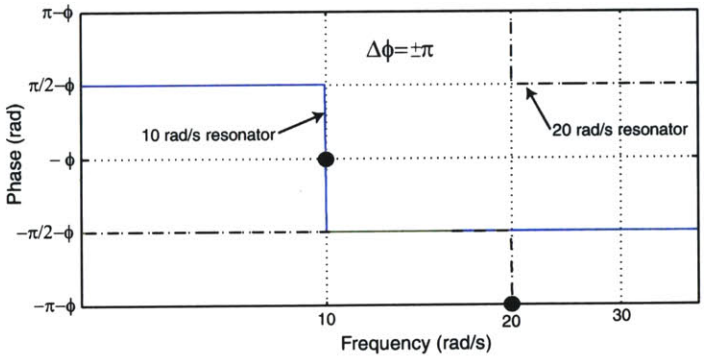


Figure 4-12: Bode phase plot of two resonators illustrating how resonators with $\phi_1 = \phi$ and $\phi_2 = \phi \pm \pi$ result in a system with the two resonators in phase in the frequencies between the two resonances. The dot indicates the average phase at each resonance.

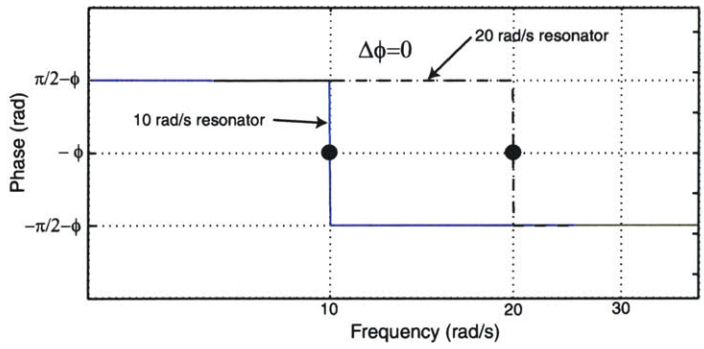


Figure 4-13: Bode phase plot of two resonators illustrating how resonators with $\phi_1 = \phi_2 = \phi$ results in a system with the two resonators out of phase in the frequencies between the two resonances. The dot indicates the average phase at each resonance.

ω_1 and ω_2. Thus the two magnitudes curves are in phase in the region where their magnitudes are equal, and thus add constructively. This constructive combination accounts for the shallow notch for $\Delta\phi = \pm\pi$ seen in Figure 4-11.

Similarly, the magnitude of the local minima is minimized, 0 on a linear scale and $-\infty$ on a log scale, when $\Delta\phi = 0$, such that the phases of the two resonators differ by $\pm\pi$ between ω_1 and ω_2. Figure 4-13 illustrates this for two resonators with $\phi_1 = \phi_2 = \phi$. As shown, this results in a system where, between ω_1 and ω_2, the resonator phases differ by $\pm\pi$, and thus add destructively in the region where their magnitude curves are approximately equal. This destructive combination accounts for the sharp notch associated with $\Delta\phi = 0$ seen in Figure 4-11.

Figure 4-14 expands this viewpoint by showing the Bode plot of a three resonator system with $\phi_1 = \phi_2 = 0$ and $\phi_3 = -45°$. This figure shows several important features. First, we see even in a multiple resonator system $\angle\hat{C}(j\omega_n) = -\phi_n$ at each resonant peak, as shown by the dots on the phase curve. Second, we see that the depth of all the local minima are dependent on the phase advance ϕ_n of all the resonators. Thus even though the first and second resonators add destructively at the first minima, the magnitude of the local minima is determined by the non-zero magnitude of the third resonator. That is if ω_{min1} is the frequency of the magnitude minimum between ω_1 and ω_2, we have $C_1(\omega_{min1}) = -C_2(\omega_{min1})$ and thus $|C(\omega_{min1})| =$

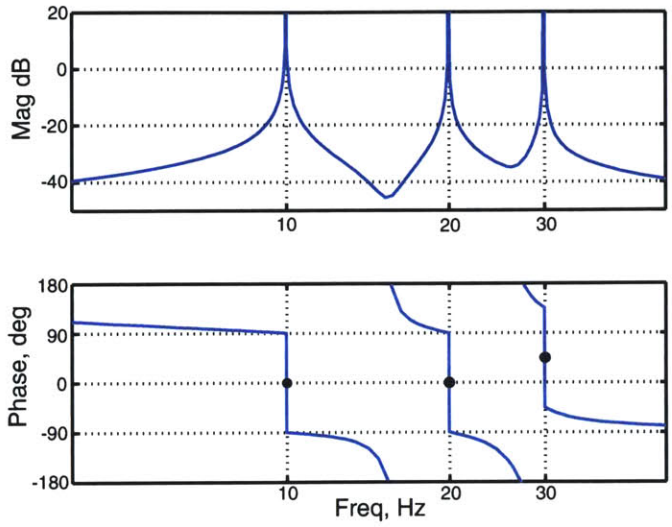


Figure 4-14: Bode plot for a three resonator system with $\phi_1 = \phi_2 = 0, \phi_3 = -45°$. The dots mark the center of the phase discontinuity to show the effect of the phase parameter ϕ_n; the phase at this point is equal to -ϕ_n.

$|C_3(\omega_{min1})|$. Third, the frequency location of the local minima are roughly at the geometric mean frequency of the adjacent resonant peaks and may be expressed as

$$\omega_{min} \approx \sqrt{\omega_1 \omega_2}. \tag{4.73}$$

Lastly, the phase of the system changes by $\pm 180°$ in the vicinity of the frequency of the local minima, i.e., for each complex pair of zeros. Thus, a way to ensure system stability is to set ϕ_n to keep the AFC loop transfer function phase curve discontinuities centered on 0°, and to keep the magnitude of the local minima sufficiently below 0 dB.

So far we have examined $\angle C(j\omega)$, the phase angle of the AFC control block, in isolation. In practice, we need to evaluate the phase angle of both the plant and AFC control block, $\angle\{C(j\omega)P(j\omega)\}$. We know from the previous discussion that at each ω_n, $C(j\omega)$ has a $-180°$ phase drop centered on $-\phi_n$ and thus $\angle\hat{C}(j\omega_n) = -\phi_n$. Examining the system from a Nyquist perspective, to maximize the phase margin of the system, we would like to center the phase discontinuity of $\angle\{C(j\omega_n)P(j\omega_n)\}$

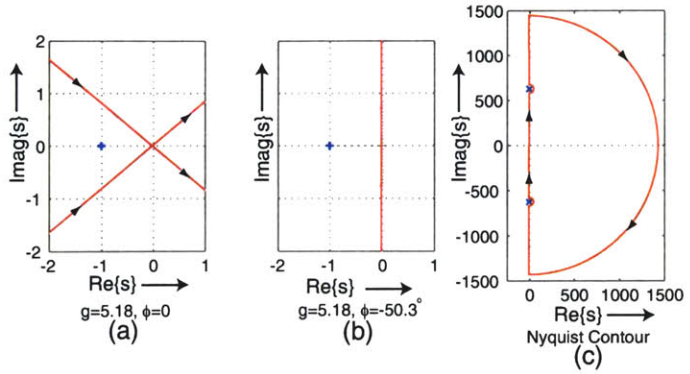


Figure 4-15: Nyquist diagrams of $C(s)P(s)$ for a system with an AFC resonator $\omega_1 = 628.3$ rad/s (100 Hz) where in (a) ϕ=0 and in (b) The associated Nyquist contour is shown in (c). The -1 point is shown as a cross in a) and b).

about 0°, so that the phase discontinuity lies between ±90°. By this choice, AFC controlled systems will approach a phase margin of 90°. If we define $\phi_{pn} = \angle P(j\omega_n)$, i.e. ϕ_{pn} is the angle of the plant at each resonator frequency, then the average phase of the system at each resonance may be expressed as

$$\angle\{\hat{C}(j\omega_n)P(j\omega_n)\} = \angle\hat{C}(j\omega_n) + \angle P(j\omega_n) = -\phi_n + \phi_{pn}. \tag{4.74}$$

Thus to make $\angle\{\hat{C}(j\omega_n)P(j\omega_n)\} = 0°$, all we need to do is set $\phi_n = \phi_{pn}$. A similar analysis and result is derived for zero phase error tracking (ZPTEC) in [91]. This result is also presented for a single resonator in [57] and [11].

This phase adjustment is essential for plants in which the phase varies significantly as a function of frequency. For example, Figure 4-15 illustrates the Nyquist diagrams for the rotary fast tool servo system with a single AFC resonator operating at 100 Hz. Figure 4-15(a) is the Nyquist plot for $C(s)P(s)$ with $\phi_1 = 0°$and thus no phase adjustment. Notice that the Nyquist plot for this system significantly penetrates the left half plane, and thus the AFC loop exhibits a resulting phase margin of 39°. The reason for this is that the plant $P(s)$ has a phase shift of -50° at 100 Hz. Figure 4-15(b) is the Nyquist plot for a controller where the phase is properly adjusted to $\phi_1 = \phi_{pn}$. This AFC loop exhibits a phase margin of 89.2°, which is significantly

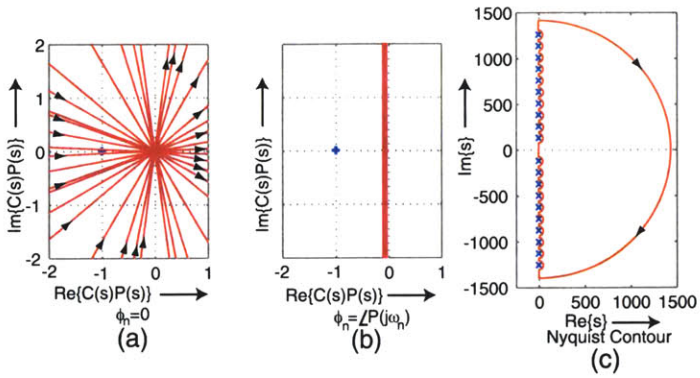


Figure 4-16: Nyquist diagrams of $C(s)P(s)$ for a system with 10 resonators where in (a) ϕ_n=0 and in (b) The associated Nyquist contour is given in (c), with detours around the imaginary axis poles of $C(s)P(s)$. The parameters ϕ_n, g_n, and ω_n for this system are listed in Table A.2.

more robust. Figure 4-16 depicts the Nyquist diagrams for the fast tool servo system with 10 resonators (see Table A.2 for AFC resonator values). The system in Figure 4-16(a) which has no phase adjustment, $\phi_n = 0$, is unstable because the Nyquist loops associated with the three highest frequency resonators (160, 180, and 200 Hz) encircle -1. The Nyquist plot of Figure 4-16(b) with phase properly adjusted, $\phi_n = \angle P(j\omega_n)$, has a phase margin of better than 84°for all loops. Figure 4-17 shows the Bode plot associated with the figure 4-16(a) Nyquist plot while Figure 4-18 is the Bode plot for the Figure 4-16(b) Nyquist plot. Notice the unstable phase curve in Figure 4-17 and the stable phase curve in Figure 4-18.

In this subsection, we learned that including a phase advance parameter to a resonator allows us to apply AFC control to systems with phases $|\phi| > 90^o$. Additionally, we proved that setting the AFC controller phase $\phi_n = \angle P(j\omega_n))$ maximizes the phase margin of an AFC compensated system. In the next subsection, we explore the effect the AFC controller gain g_n has on the performance of an AFC controller.

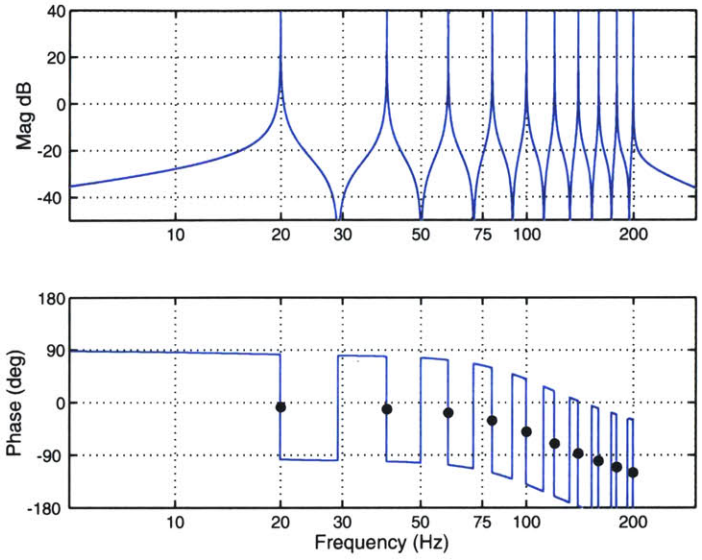


Figure 4-17: The negative of the AFC loop transmission $C_n(j\omega)P(j\omega)$ with $\phi_n = 0$ and with g_n and ω_n as listed in Table A.2). The dots mark the center of the phase discontinuity to show the effect of the phase parameter ϕ_n. This loop is unstable.

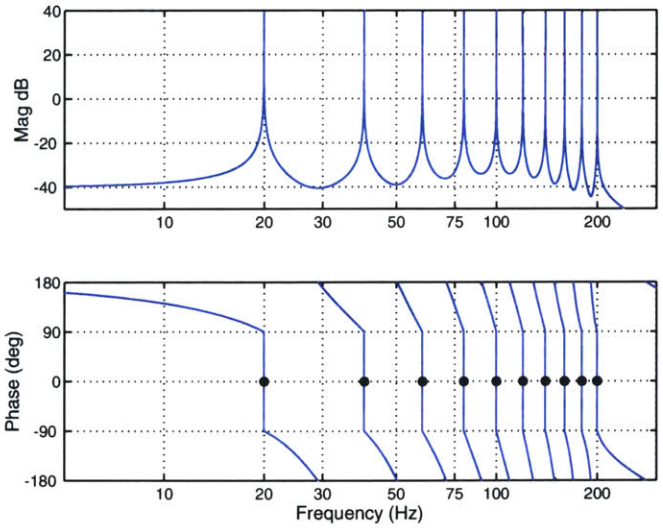


Figure 4-18: The negative of the AFC loop transmission $C_n(j\omega)P(j\omega)$ for the rotary fast-tool servo with 10 resonators and $g_n = 1$ and , (values in Table A.1). This loop is stable, with 84° phase margin. (Note excessive gain margin at each minima; this issue is addressed in subsection 4.2.3.)

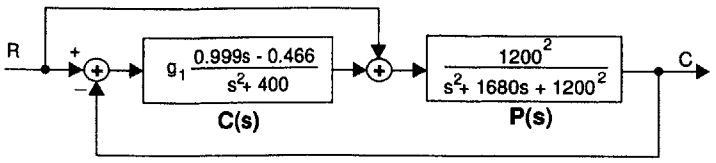


Figure 4-19: Block diagram of a single resonator $C(s)$ controlling the second order plant $P(s)$, and used to simulate the effect of resonator gain on system response. Here $\omega_1 = 20$ rad/s and $\phi_1 = 0.023°$.

4.2.2 Resonator Gains

To maximize disturbance rejection in the vicinity of ω_n, we would like to maximize the gain g_n for each resonator. Just as in a conventional controller, the resonator gain affects relative stability as well as the system settling time and the system error for inputs in the vicinity of ω_n. In the case of an AFC resonator, the settling time represents the characteristic time to cancel out a disturbance or adjust to a magnitude shift in the reference input component at the resonator frequency. The higher the resonator gain, generally the more rapidly the resonator responds in the closed loop within the limits of stability. Similarly while an AFC resonator with zero damping (i.e. the resonant poles are on the imaginary axis) will eventually drive the system error at exactly the resonator frequency ω_n to zero regardless of the resonator gain g_n, the system error to commands and disturbances near but not at the resonator frequency is inversely proportional to the resonator gain. Thus the higher the resonator gain, the lower the system error for disturbance inputs near the resonator frequency. This result is important since any system is likely to have perturbations in the frequency of the periodic motion which will shift the frequency of the disturbance inputs from the resonator frequency, and as well, no system operates exactly in steady state in finite time.

To illustrate the effects of resonator gain on system response, we simulated in time a loop with a single 20 rad/s AFC resonator acting on a second order system $P(s)$ with a natural frequency $\omega_n = 1200$ rad/s and a damping ratio of $\zeta_n = 0.7$. Figure 4-19 shows the block diagram of the simulated system while Figure 4-20 illustrates the plant frequency response $P(j\omega)$. Figure 4-21 plots the following error as a percent

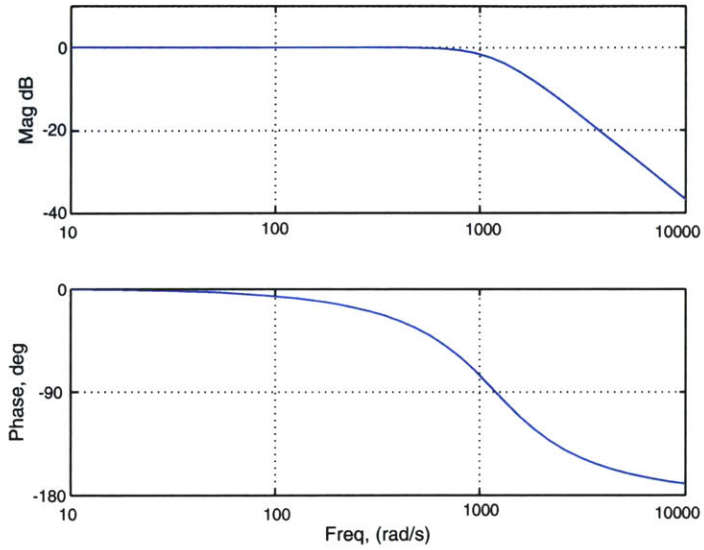


Figure 4-20: Bode Plot for second order system $P(s)$ used to simulate the effect of resonator gain on system response.

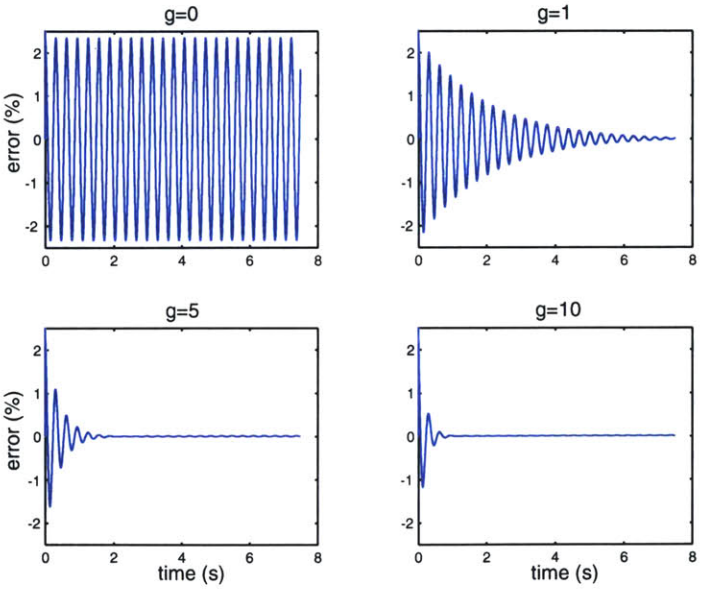


Figure 4-21: Percent error tracking a sinusoidal reference trajectory with $\omega_r = 20$ rad/s with an AFC resonator tuned to $\omega_1 = 20$ rad/s for resonator gains $g_1 = 0$, 1, 5, and 10, respectively.

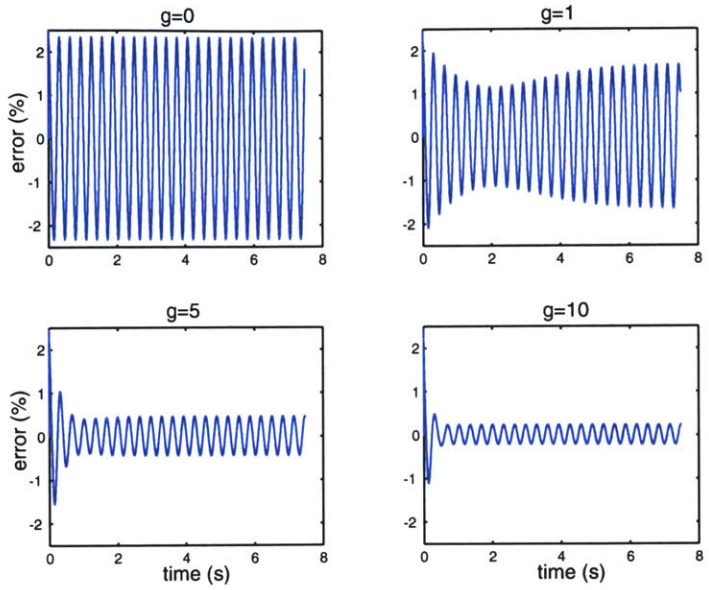


Figure 4-22: Percent error tracking a sinusoidal trajectory with $\omega_r = 19.5$ rad/s and an AFC resonator tuned to $\omega_n = 20$ rad/s for resonator gains $g_1 = 0$, 1, 5, and 10, respectively.

of the reference input for a 20 rad/s input as a function of time for a resonator gain of $g_1 = 0$ (no AFC control), and $g_1 = 1$, 5, and 10 respectively. For zero gain, the peak following error is 2.3%. The addition of the AFC controller will drive the system error to zero for all positive gains and where $\omega_r = \omega_n$ (ω_r is the frequency of the reference input), but the error settling time is reduced from greater than 7 seconds to less than a second as the resonator gain increases from 1 to 10. The small steady-state ripple observed in the figures for $g_1 = 5$ and $g_1 = 10$ is believed to be due to numerical issues in our simulation.

We also use this simulated system to examine the error response to commands slightly displaced from the resonator frequency. Figure 4-22 shows the error to an $\omega_r = 19.5$ rad/s input for the same four resonator gains and $\omega_n = 20$ rad/s. The plots show that while the AFC resonator attenuates the system error for all three non-zero gains, the residual system error is decreased proportionally as the resonator gain is increased. This effect can be explained by a simple magnitude of loop transmission argument. Chapter 5 documents an extension of our loop shaping perspective which

we term Oscillator Amplitude Control (OAC). The OAC perspective uses an averaging analysis to predict the time characteristics of the error envelope with varying g_n. Overall, though, it is clear that for low errors and faster convergence, g_n should be maximized within the limits of stability.

4.2.3 Choosing Gains g_n for N Parallel Resonators

There are a number of considerations that apply in selecting resonator gains g_n in multi-resonator systems. First, to avoid instability, the magnitude of the AFC loop local minima must be kept sufficiently below 0 dB in order to avoid Nyquist encirclements in the AFC loop. Thus g_n should not be too large. Secondly, the error in the vicinity of each ω_n is approximately inversely proportional to the corresponding g_n, and thus g_n needs to be made as large as possible. Lastly, while the parameters g_n and ϕ_n allow for some shaping of the frequency response, the underlying backbone of the response is determined by the shape of the inner-loop plant $P(s)$. This means that to achieve a well-behaved response with an AFC controller, we need to start with a well behaved plant $P(s)$. That is, we would like a plant with slowly varying magnitude and phase curves in the frequency range of the AFC resonators. We thus require high performance and robust tuning of the conventional controller within $P(s)$. Finally, it is clear that the hardware itself must be well-designed to allow proper performance of all the controllers. For example, non-linear effects such as drivetrain backlash or A/D quantization will limit the performance achievable in the AFC loop.

Given the above, we employ two approximations to simplify the gain selection process. First, since the frequency response of the system at any point is dominated by the nearest resonators, we assume that the magnitude of a local minimum may be controlled by adjusting the gains of the local resonators only. In the simplest case, one can set the gain margin of a local minimum to a desired value by adjusting the gains of just the two adjacent resonators. This approximation greatly streamlines the design process and allows for ready hand tuning of the frequency response. This assumption works particularly well for the low frequency resonators in multi-resonator systems, but breaks down at the higher frequency resonators where the successive

linear harmonics are more closely spaced on the logarithmic frequency axis. The second approximation is that the local minima are positioned at the logarithmic mean frequency of the two adjacent resonators. While it is not difficult to calculate the exact frequency of a local minimum, this assumption allows one to rapidly get a good estimate for the gain margin of a local minimum. This assumption works well when adjacent resonators have only small differences in gain. If large gain differences occur, the local minima will shift towards the resonator with the lower gain because of the shoulders of the higher gain resonator will dominate the sum of the two resonators over a larger frequency range. Our AFC tuning method can be summarized as follows:

1. For each resonator set $\phi_n = \angle P(j\omega_n)$ to maximize phase margin.
2. Set initial resonator gains to unity.
3. Using the previously determined values, compute $C(j\omega)P(j\omega)$ and determine the local loop transmission minimum with the least gain margin.
4. Choose a desired gain margin.
5. Determine the ratio between the minimum gain margin found in step 3 and the desired gain margin.
6. Scale all of the resonator gains by the ratio found in the previous step.
7. Recompute and plot $C(j\omega)P(j\omega)$ to verify stability margins.
8. Adjust the gain margins of the local minima as desired by adjusting the gains of the adjacent resonators to trade robustness for control authority.

We have applied this tuning procedure to the AFC control of our rotary fast tool servo. The plant $P(s)$ Bode plot for the rotary FTS is shown in Figure 4-23. Figure 4-24 shows the negative of the AFC loop transmission for the the rotary fast tool servo system with 10 resonators with $\phi_n = \angle P(j\omega_n)$, and $g_n = 1$. The parameters ϕ_n, g_n, and ω_n for this system are listed in Table A.1. The system has a minimum gain margin of 34 dB at the local minimum centered between the 5th and 6th resonators. Notice

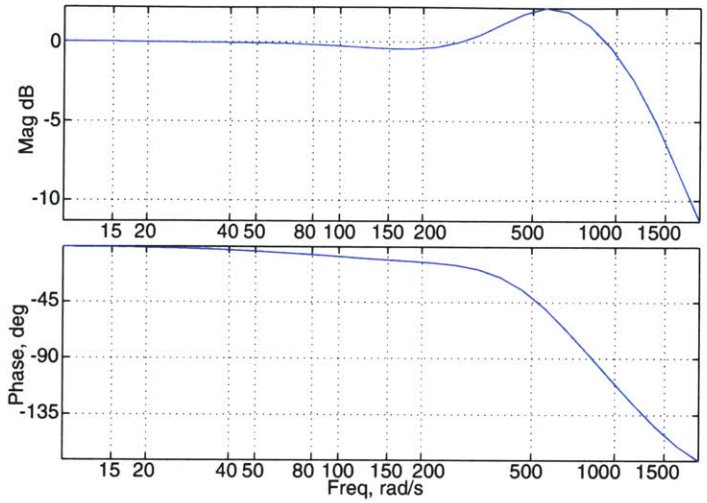


Figure 4-23: Typical fast tool servo closed loop transfer function, $P(j\omega)$ from position reference input to measured position output. The associated state space matrices representing this model are given in section A.1.

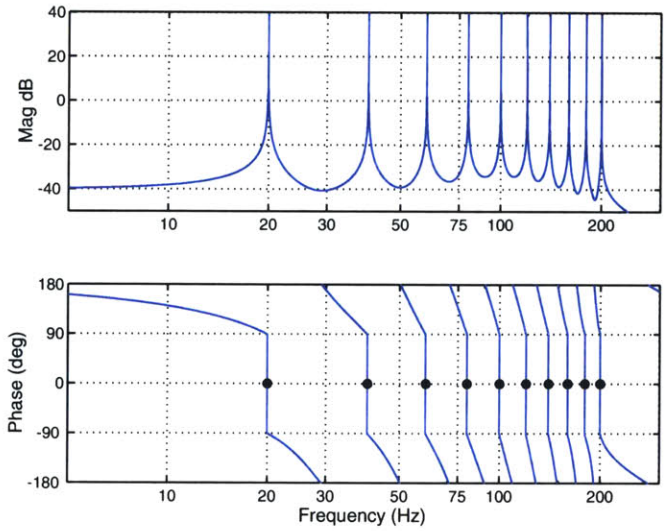


Figure 4-24: The negative of the AFC loop transmission $C_n(j\omega)P(j\omega)$ for the rotary fast-tool servo with 10 resonators and $g_n = 1$ and , (values in Table A.1). This loop is stable, with 84° phase margin.

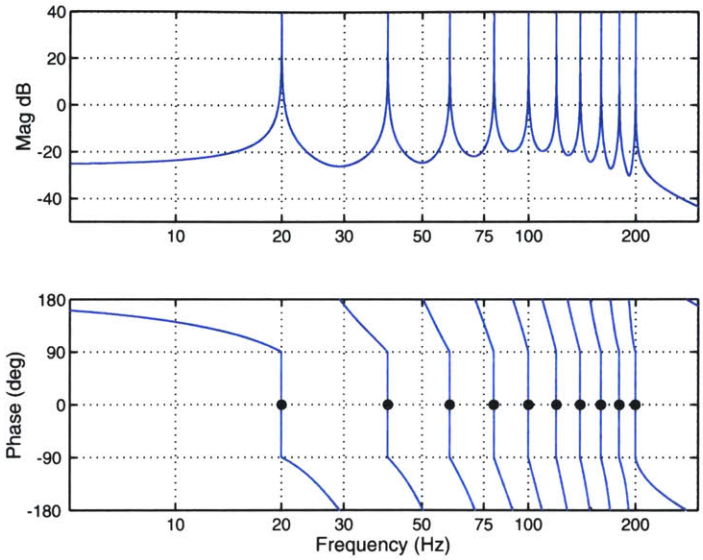


Figure 4-25: The negative of the AFC loop transmission $-L(j\omega)$ for the rotary fast-tool servo with 10 resonators after gain scaling to a desired minimum gain margin of 20dB (ϕ_n, g_n, and ω_n for this system are listed in Table A.2). Note excessive gain margin at the low- and high-frequency minima.

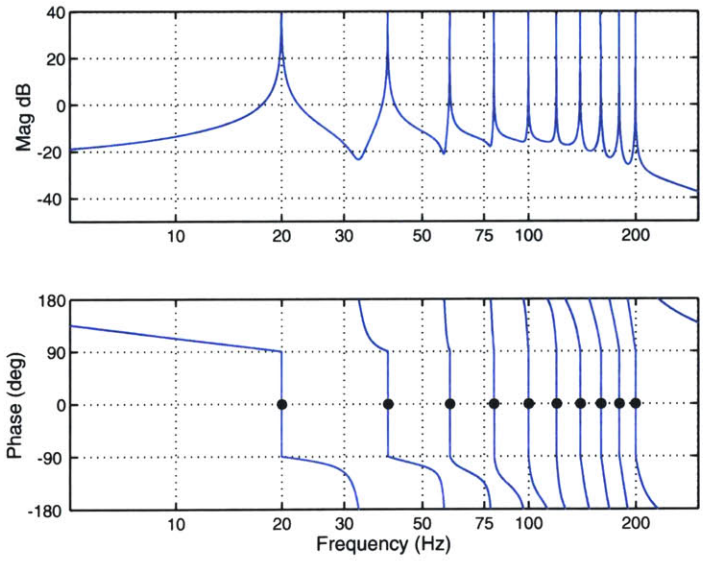


Figure 4-26: The negative of the AFC loop transmission $-L(j\omega)$ for the rotary fast-tool servo with 10 resonators after hand tuning the low- and high-frequency resonators to a target gain margin of 20dB (ϕ_n, g_n, and ω_n for this system are listed in Table A.3).

that the local minima tend to follow the shape of the inner loop $P(s)$ illustrated in Figure 4-23. This tuning clearly has too low a value of g_n and thus sacrifices performance.

Figure 4-25 shows the same loop after increasing all of the resonator gains by a factor of 5.18 to achieve a desired minimum gain margin of 20 dB. The parameters ϕ_n, g_n, and ω_n for this system are listed in Table A.2. After gain scaling, the minimum gain margin is still at the local minimum between the fifth and sixth resonators. Note that there is still an excess of gain margin at both the lowest and highest frequencies. Since the lens shapes that we are machining are dominated by the lower frequency harmonics, we would like very good performance at low frequency. We now start to individually tune each of the resonator gains.

Using the simplifications described above for the relationships between resonator gains and local minima, we have scaled up the gains for the low frequency resonators, and turned down the gains for the middle frequency resonators. The parameters ϕ_n, g_n, and ω_n for this system are listed in Table A.3. Figure 4-26 shows the results of this tuning. The magnitude of the low frequency peaks has been increased by a factor of 6, while the minimum gain margin has only been reduced from 20 dB to 15.9 dB at the local minimum between the 5th and 6th resonators. This AFC loop is now well-tuned for our machine and cutting requirements.

4.2.4 Experimental Results for the Rotary Fast Tool Servo

Figure 4-27 shows the measured error following a ± 1 cm amplitude 20 Hz sine wave air cut for the fast-tool servo with and without a 20 Hz AFC resonator included in the control loop. This cut has a peak acceleration at the tool of approximately 15 g's. The system error without the AFC resonator is 19.8%, as shown in the center plot in Figure 4-27. The bottom plot shows that the peak system error with the 20 Hz AFC resonator is reduced to ± 0.3 μm peak (0.0033%), a factor of 5000 improvement over the system without AFC control. The controller used to generate the results in Figure 4-27 does not include a command pre-shifting feed-forward term.

A more stringent test is to apply AFC while the tool is cutting. Figure 4-28

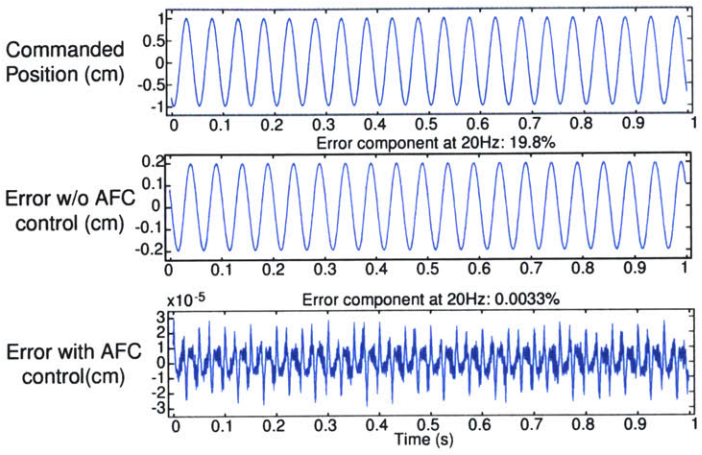


Figure 4-27: Measured error for the rotary fast tool servo for a 1 cm 20 Hz air cut with and without a 20 Hz AFC resonator. With AFC control, this following error is about $\pm 0.3\mu$m peak.

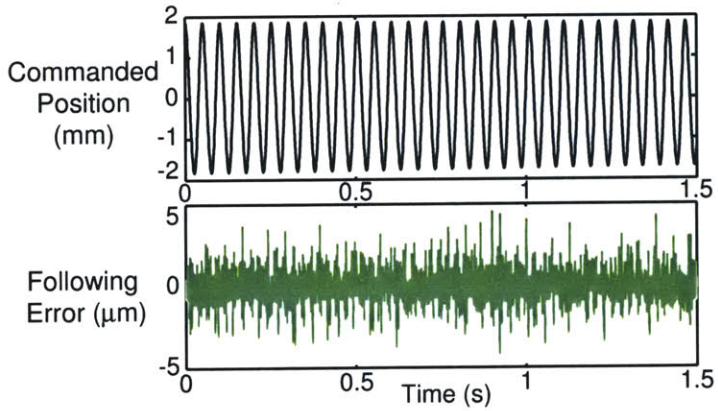


Figure 4-28: Measured error with both AFC and command pre-shifting while cutting a 0x4 toric in CR39 at 600 RPM. Data taken at a radius on the part of 30 mm.

plots the measured error while cutting a 0x4 diopter toric in CR39, an acrylic plastic commonly used to make spectacle lenses, at a radius of 30 mm and a spindle speed of 600 RPM. The root-mean-square following error for this cut is 1.2 μm, or 0.06% of the peak command amplitude. The controller used to generate the results for Figure 4-28 incorporates the conventional lead-lag controller within $P(s)$, command pre-shifting $P^{-1}(j\omega_n)$, and ten AFC resonators at harmonics of the spindle speed.

In this section, we have documented and applied a loop shaping approach for tuning AFC controllers. These techniques allow us to design controllers which precisely follow sinusoid trajectories with constant or slowly time varying magnitude. In the next section, we present an extension to AFC control called AMAFC control designed to enhance trajectory following for sinusoids with more rapid time varying magnitude.

4.3 Amplitude Modulated Adaptive Feedforward Cancellation

In this section, we present an extension of Adaptive Feedforward Cancellation (AFC) which we term Amplitude Modulated Adaptive Feedforward Cancellation (AMAFC). The goal of AMAFC control is to improve trajectory tracking for signals with time varying magnitude by incorporating an estimate of the time varying magnitude in the AFC structure. In section 4.3.1, we propose an AMAFC structure which incorporates the estimate of the time varying magnitude using multiplication. This structure was originally proposed by Joe Calzaretta [13]. Experimental results for this structure are included in chapter 7. Section 4.3.2 presents an AMAFC structure which incorporates the estimate of the time varying magnitude using addition. This structure was proposed by Prof. Trumper after reading an initial draft of this thesis and thus no experimental results were obtained. Section 4.3.3 contains simulated results applying both AMAFC forms to a sample system.

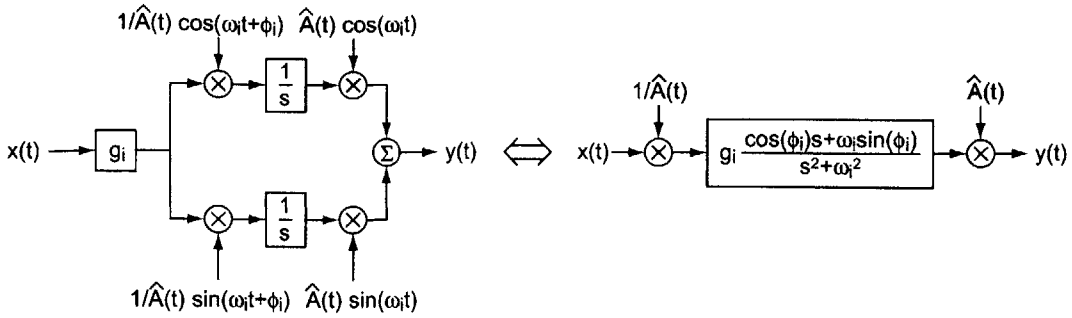


Figure 4-29: A multiplicative Amplitude Modulated Feedforward Cancellation controller in two forms.

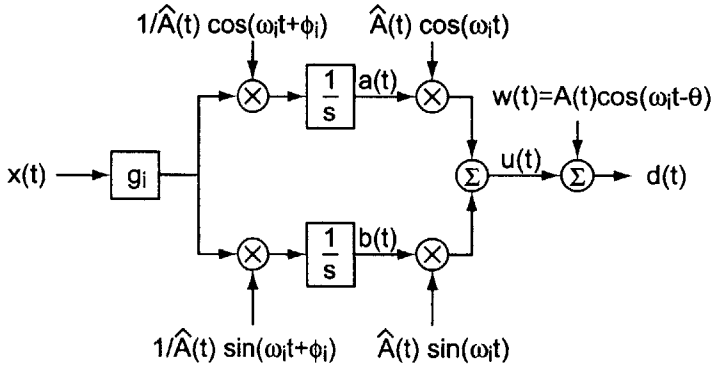


Figure 4-30: Block diagram of a multiplicative AMAFC resonator plus disturbance subsystem.

4.3.1 Multiplicative Amplitude Modulated Adaptive Feedforward Cancellation

Figure 4-29 shows the multiplicative AMAFC control structure in two different forms. The canonical from is shown on the left while the right side shows a form where the basic AFC structure has been replaced with the LTI equivalent. The basic theory for AFC control comes from the Internal Model Principle (IMP) [29]. The IMP essentially states that for a controller to exactly cancel a disturbance it must contain a model of the disturbance signal. Stated in another fashion, when a controller has no input, it must be able to output exactly the negative of the disturbance signal. AFC control

satisfies this condition if the disturbance $w(t)$ has the form

$$w(t) = a_0 \sin(\omega t) + b_0 \cos(\omega t) = A \cos(\omega t - \theta). \tag{4.75}$$

AFC control does not satisfy the IMP if the disturbance has the form

$$w(t) = a_0(t) \sin(\omega t) + b_0(t) \cos(\omega t) = A(t) \cos(\omega t - \theta) \tag{4.76}$$

since the time varying magnitude $A(t)$ is not included in the controller model (Note: the $A(t)\cos(\omega t - \theta)$ form makes the assumption that $\frac{a_0(t)}{b_0(t)}$ is constant, a more general form would allow θ to vary with time). To exactly cancel $w(t)$, we must incorporate $A(t)$ into our controller. Since $A(t)$ is unknown, we are forced to make an estimate $\hat{A}(t)$ such that

$$A(t) = \hat{A}(t) R(t). \tag{4.77}$$

In this case $\hat{A}(t)$ is a known function representing the best estimate of the time varying magnitude and $R(t)$ is a multiplicative residual function representing the unknown error between the estimated and actual magnitudes. Figure 4-30 shows a block diagram of the AMAFC plus disturbance subsystem. To show that this new structure obeys the IMP, we need only set $x(t) = 0$. Because both integrators have zero input, the integrator outputs become constant:

$$a(t) = a_0 \tag{4.78}$$

$$b(t) = b_0. \tag{4.79}$$

Thus the controller output is

$$u(t) = \hat{A}(t)[a_0 \cos(\omega_i t) + b_0 \sin(\omega_i t)]. \tag{4.80}$$

If the integrator outputs happen to be

$$a_0 = -\cos\theta \tag{4.81}$$

$$b_0 = -\sin\theta, \tag{4.82}$$

the controller output becomes

$$\begin{aligned} u(t) &= -\hat{A}(t)[\cos\theta\cos\omega_i t + \sin\theta\sin\omega_i t] \\ &= -\hat{A}(t)\cos(\omega_i t - \theta). \end{aligned} \tag{4.83}$$

Thus the subsystem output

$$\begin{aligned} d(t) &= A(t)\cos(\omega_i t - \theta) - \hat{A}(t)\cos(\omega_i t - \theta) \\ &= \hat{A}(t)[R(t) - 1]\cos(\omega_i t - \theta). \end{aligned} \tag{4.84}$$

If $R(t) = 1$, that is if we have perfectly modeled $A(t)$, then $d(t) = 0$ and the internal model principle is obeyed.

Alternately if $R(t) = R_0$, where R_0 is a non-zero constant, $w(t)$ becomes

$$w(t) = R_0\hat{A}(t)\cos(\omega_i t - \theta). \tag{4.85}$$

The IMP is obeyed if the integrator outputs are

$$\begin{aligned} a_0 &= -R_0\cos\theta \\ b_0 &= -R_0\sin\theta \end{aligned} \tag{4.86}$$

which results in

$$\begin{aligned} u(t) &= -R_0\hat{A}(t)[\cos\theta\cos\omega_i t + \sin\theta\sin\omega_i t] \\ &= -R_o\hat{A}(t)\cos(\omega_i t - \theta). \end{aligned} \tag{4.87}$$

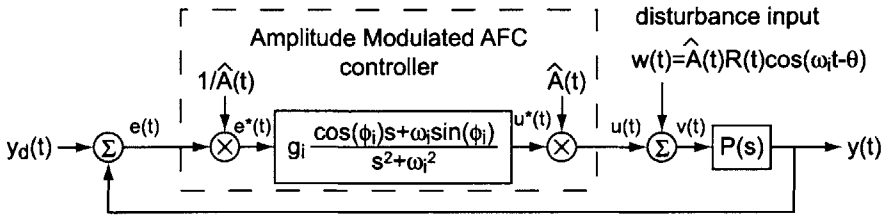


Figure 4-31: Block diagram of an AMAFC controlled position loop.

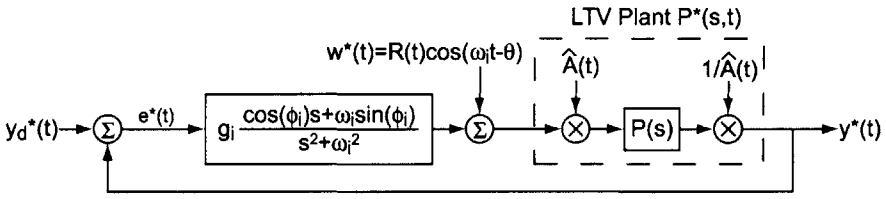


Figure 4-32: Block diagram of an AMAFC controller with the modulation term $\hat{A}(t)$ shifted to the plant.

If $R(t)$ is time varying, the AMAFC control structure departs from the IMP which results in a time-varying error.

In order to analyze the characteristics of the time-varying closed loop following error $e(t)$ under AMAFC, we need to look at the signals in more detail. Figure 4-31 shows the block diagram of a generic plant $P(s)$ under AMAFC control. As we can see, we are comparing a desired output $y_d(t)$ to the actual output $y(t)$ to produce an error signal $e(t)$. This signal is then divided by our estimate of the time variation of the disturbance $\hat{A}(t)$ to form $e^*(t)$, which is acted upon by the standard LTI AFC controller to produce $u^*(t)$. The AMAFC controller output $u(t)$ is then produced by dividing $u^*(t)$ by $\hat{A}(t)$. The plant input $v(t)$ is produced by adding $w(t) = \hat{A}(t)R(t)\cos(\omega_i t - \phi)$ to $u(t)$. From a block diagram perspective, this system is a time varying controller acting upon an LTI plant. Since the controller is not LTI, it is difficult to analyze the behavior of the system. It is useful to change our perspective by shifting the $\hat{A}(t)$ modulation terms around the block diagram.

In order to shift the modulation terms around the block diagram, we define a

transformed signal

$$f^*(t) \equiv f(t)/\hat{A}(t). \tag{4.88}$$

Figure 4-32 shows the block diagram of this system with the $\hat{A}(t)$ modulation terms moved around the block diagram. This now appears to be a system in which an LTI controller is acting on a time-variant plant $P^*(s,t)$ (the combination of the modulation terms and the plant P(s)) which does not appear to be much of an improvement. Note: the time varying form $P^*(s,t)$ is not a mathematically correct and is utilized here only to indicate that the system is LTV. In a specific case, if $P(s)$ were a constant P_0, the time variant plant is

$$P^*(s,t) = \frac{\hat{A}(t)P_0}{\hat{A}(t)} = P_0 \tag{4.89}$$

which is LTI. This structure now looks like our conventional AFC control loop. More generally $P(s)$ is often nearly constant near $P(j\omega_i)$ and thus if $\hat{A}(t)$ and $R(t)$ vary much slower than ω_i, we can approximate $P^*(s,t)$ as a LTI constant $P(j\omega)$. This nominally implies that $e^*(t)$ and thus $e(t)$ should decay to zero resulting in perfect rejection of the disturbance $w(t)$. In addition the LTI plant approximation means that we can apply the tuning rules developed for selecting g_i and ϕ_i for a standard AFC controller to the AMAFC controller and be confident that the resulting AMAFC controller will be stable.

To evaluate the disturbance rejection of the multiplicative AMAFC control structure it is useful to find $D(s)$, the transfer function between the error $e^*(t)$ and the disturbance $w^*(t)$. From Figure 4-32 we see that

$$D(s) \equiv \frac{-P^*(s,t)}{1 + g_i \frac{s\cos\phi_i + \omega_i \sin\phi_i}{s^2+\omega_i^2} P^*(s,t)}. \tag{4.90}$$

Lets assume that $R(t)$ has the form

$$R(t) = \cos \alpha t \tag{4.91}$$

where $\alpha << \omega_i$. This results in

$$w^*(t) = \cos\alpha t \cos(\omega_i t - \theta). \tag{4.92}$$

Employing the following two trigonometric identities

$$\cos(A + B) = \cos A \cos B - \sin A \sin B \tag{4.93}$$

$$\cos(A - B) = \cos A \cos B + \sin A \sin B, \tag{4.94}$$

allows us to express $w^*(t)$ as

$$w^*(t) = \frac{1}{2}[\cos((\omega_i - \alpha)t - \theta) + \cos((\omega_i + \alpha)t - \theta)]. \tag{4.95}$$

Thus the input into the transfer function $D(s)$ is simply two sinusoids each shifted off of ω_i by α (Note: this same result can be achieved using Fourier transforms as we do later for the Oscillator Amplitude Control perspective). Since α is much less than ω_i, we are only interested in the response of $D(s)$ at frequencies near ω_i. Since the magnitude of the AFC resonator is extremely high near ω_i, $D(s)$ reduces to

$$D(s) = \frac{-P^*(s,t)}{g_i \frac{s\cos\phi_i + \omega_i \sin\phi_i}{s^2+\omega_i^2} P^*(s,t)} = -\frac{s^2 + \omega_i^2}{g_i(s\cos\phi_i + \omega_i \sin\phi_i)} \tag{4.96}$$

which is just the negative inverse of the AFC controller. The magnitude of this transfer function is zero at $s = j\omega_i$ indicating perfect error rejection.

So far we have only discussed the effect of AMAFC control on disturbance rejection. We may also be interested in the error response for a modulated input

$$y_d(t) = \sin\beta t \sin\omega_i t. \tag{4.97}$$

In this case, we have an excellent estimate of the amplitude modulation since we should have precise knowledge of the desired position. From Figure 4-32, we can see

that

$$\frac{E^*(s)}{Y_d^*(s)} = \frac{1}{1 + g_i \frac{s\cos\phi_i + \omega_i \sin\phi_i}{s^2+\omega_i^2} P^*(s,t)}. \tag{4.98}$$

If we assume that $\beta << \omega_i$, this transfer function reduces to

$$\frac{E^*(s)}{Y_d^*(s)} = \frac{s^2 + \omega_i^2}{P^*(s,t)(s\cos\phi_i + \omega_i \sin\phi_i)} \tag{4.99}$$

which is similar to the (4.96), the transfer function between disturbance and error, except the time variant plant dynamics are not canceled. This would seem to imply that even with a perfect model of the amplitude variation, we would have some residual error due to the time varying nature of the plant.

There are a number of issues with implementing the multiplicative AMAFC structure. One of the most significant is that for many cases $A(t) = 0$ at some point, this means that $1/\hat{A}(t) = \infty$. This is clearly unacceptable. One way to avoid this issue is to avoid applying multiplicative AMAFC to signals which have zero value. One class of modulation signals we might wish to cancel is

$$A(t) = 1 + a_0 \sin \beta t \tag{4.100}$$

where $|a_0| < 1$. This is the strategy we employed in generating the experimental results. More generally, we can limit the magnitude of the $1/\hat{A}(t)$ signal. Figure 4-33 shows a block diagram of a multiplicative AMAFC control structure with

$$\frac{1}{\hat{A}(t)} = \begin{cases} 1/\hat{A}(t) & \text{for } |\hat{A}(t)| > c \\ sgn[\hat{A}(t)]/c & \text{for } |\hat{A}(t)| < c \end{cases} \tag{4.101}$$

This is the structure we use to generate all of our simulated results.

In addition to the implementation issues with the multiplicative AMAFC structure, from our analysis we have found that the multiplicative AMAFC structure will perfectly cancel a disturbance with time varying magnitude if we have an accurate

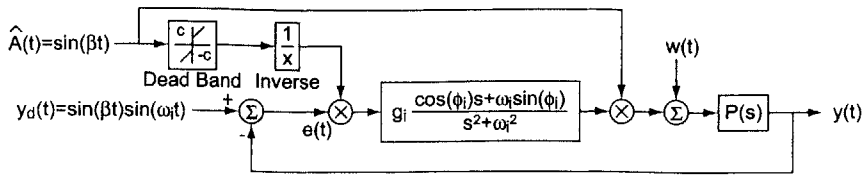


Figure 4-33: Block diagram for a multiplicative AMAFC controller with $1/\hat{A}(t)$ limited.

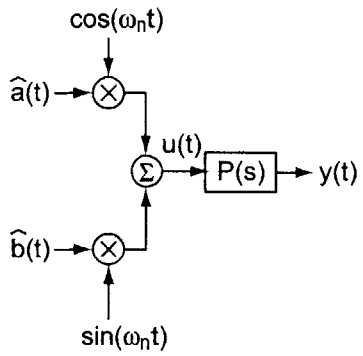


Figure 4-34: Block diagram of a standard AFC resonator and plant with zero input into the AFC system.

estimate of the time variance. We also found that the multiplicative AMAFC structure did not perfectly follow a commanded trajectory with time varying magnitude even with a perfect estimate of the time variance. Since, we never really have detailed knowledge of the disturbance this result seems the inverse of what we would like. To correct these issues, we have proposed an additive AMAFC structure.

4.3.2 Additive Amplitude Modulated Feedforward Cancellation

One of the weakness of the multiplicative AMAFC structure is the inability to perfectly follow commanded trajectories with time varying magnitude. As mentioned in the previous section, the IMP essentially states that for a controller to exactly follow a signal it must contain a model of the signal. Thus to perfectly follow a commanded

input of the form,

$$y_d(t) = a(t)\cos(\omega_n t) + b(t)\sin(\omega_n t) \tag{4.102}$$

where

$$\begin{aligned} a(t) &= a_n \cos(\alpha t) \\ b(t) &= a_n \cos(\alpha t) \end{aligned} \tag{4.103}$$

the AMAFC structure must output a signal which when passed through the plant generates $y_d(t)$. Figure 4-34 shows a standard AFC resonator and plant $P(s)$ with zero input into the AFC systems. Since know the form of the input from (4.102) and (4.103), it makes sense to assume $\hat{a}(t)$ and $\hat{b}(t)$ are of the form

$$\hat{a}(t) = \hat{a}\cos(\alpha t) \tag{4.104}$$

$$\hat{b}(t) = \hat{b}\cos(\alpha t). \tag{4.105}$$

If $\alpha << \omega_n$ then we can replace the plant transfer function $P(s)$ with $P(j\omega_n)$. Thus the plant output becomes

$$y(t) = |P(j\omega_n)|\cos(\alpha t)[\hat{a}\cos(\omega_n t + \phi_n) + \hat{b}\sin(\omega_n t + \phi_n)] \tag{4.106}$$

where $\phi_n = \angle P(j\omega_n)$. For the error to be zero, we clearly would like

$$\hat{a} = \frac{a_n}{|P(j\omega_n)|} \tag{4.107}$$

$$\hat{b} = \frac{b_n}{|P(j\omega_n)|} \tag{4.108}$$

but this selection of $\hat{a}(t)$ and $\hat{b}(t)$ does not result in the correct phase for the ω_n sinusoid. One way to correct for the phase of the ω_n sinusoid is to change the phase of the modulation terms of the AFC resonator to be $\cos(\omega_n t - \phi_n)$ and $\sin(\omega_n t - \phi_n)$.

Figure 4-35 shows a block diagram of an additive AMAFC resonator. As we can

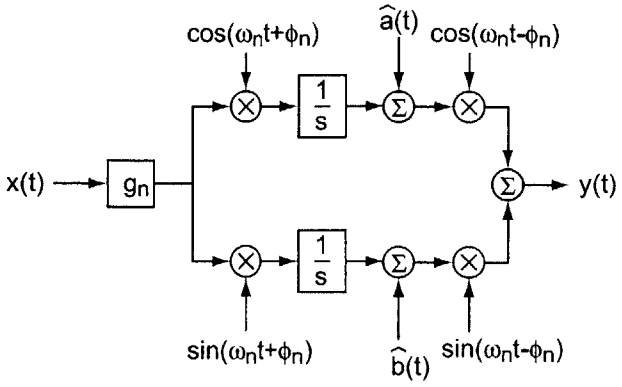


Figure 4-35: Block diagram of an additive AMAFC resonator.

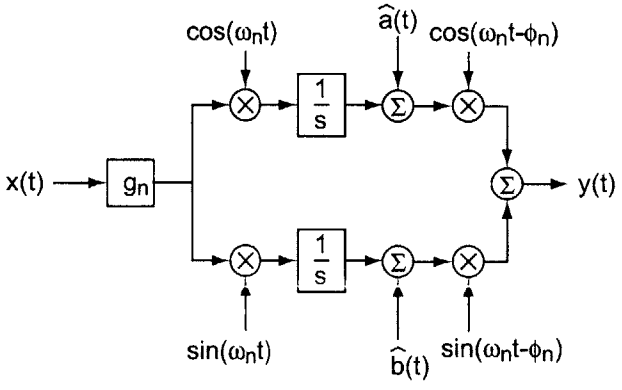


Figure 4-36: Block diagram of an additive AMAFC resonator.

see, this structure is identical to the standard AFC controller accept, we have added $\hat{a}(t)$ and $\hat{b}(t)$ to the resonator after the integrator and we have included an additional phase shift to the second sinusoid. Setting $\hat{a}(t) = \hat{b}(t) = 0$, we find that this structure has a LTI equivalent of

$$C(s) = g_n \frac{s\cos(2\phi_n) + \omega_n \sin(2\phi_n)}{s^2 + \omega_n^2}. \tag{4.109}$$

The phase drop of this system is now centered on $2\phi_n$ meaning this is stable only for system where $|\angle P(j\omega_n)| < \pi/2$. To correct for this, we eliminate the phase shift on the first set of resonators. Figure 4-36 shows the resulting block diagram with this adjustment. With $\hat{a}(t) = \hat{b}(t) = 0$, the structure shown in Figure 4-36 has a LTI equivalent of

$$C(s) = g_n \frac{s\cos(\phi_n) + \omega_n \sin(\phi_n)}{s^2 + \omega_n^2} \tag{4.110}$$

which is identical to the previous AFC structures. This result implies, that if the system is tuned as a standard AFC resonator for robust stability, the resulting additive AMAFC resonator will also have robust stability.

Typically, AFC control is combined with other forms of feedforward control. Figure 4-37 shows a control system incorporating both AFC and standard feedforward control. One of the advantages of using sinusoidal trajectories of the form

$$y(t) = a_n \cos(\omega_n t) + b_n \sin(\omega_n t), \tag{4.111}$$

is that we can replace the inverted plant $P^{-1}(s)$ with a second trajectory signal

$$y^*(t) = \frac{1}{|P(j\omega_n)|}[a_n \cos(\omega_n t - \phi_n) + b_n \sin(\omega_n t - \phi_n)] \tag{4.112}$$

where $\phi_n = \angle P(j\omega_n)$. Independent of the time varying amplitude, the structure shown in Figure 4-36 has essentially incorporated the outer feedforward loop into to the AFC structure. There is a significant computational advantage to this, since it

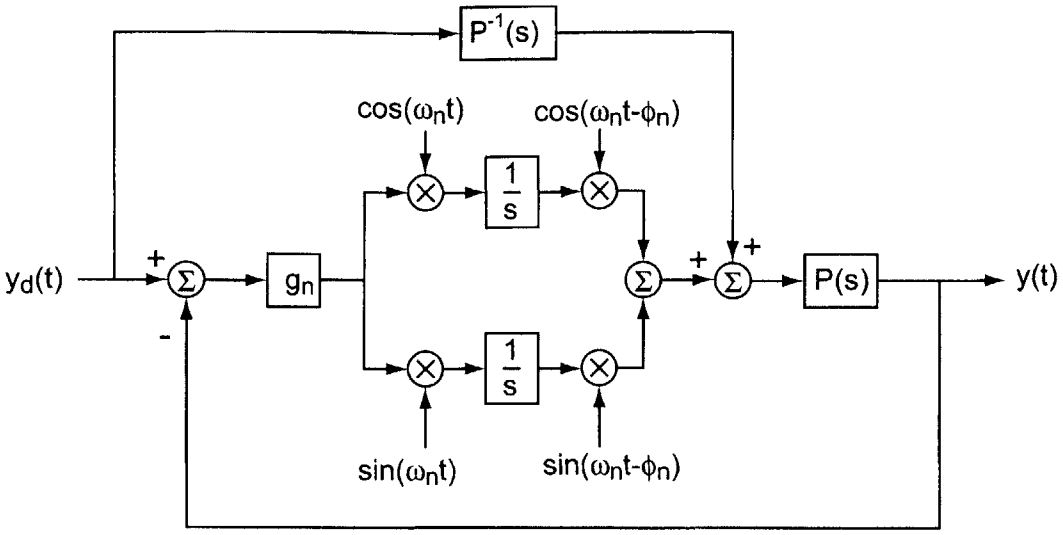


Figure 4-37: Block diagram of a control system incorporating both AFC and standard feedforward control.

eliminates the need to generate 2 sinusoids for the feedforward channel per trajectory harmonic.

In this section and the one prior, we have proposed two extensions to AFC control that incorporate the time varying magnitude of a reference input. In one, the varying magnitude is incorporated using multiplication. In the second, the time variation is incorporated using addition. In the next section, we present some simulated examples applying our AMAFC extensions to a system.

4.3.3 Example of Amplitude Modulated Adaptive Feedforward Cancellation

In this example, we consider the effect of our two proposed AMAFC structures on a system following a sinusoidal trajectory with time varying magnitude. We utilize a simple second order system as our plant

$$P(s) = \frac{\omega_p^2}{s^2 + 2\zeta_p\omega_p s + \omega_p^2} \tag{4.113}$$

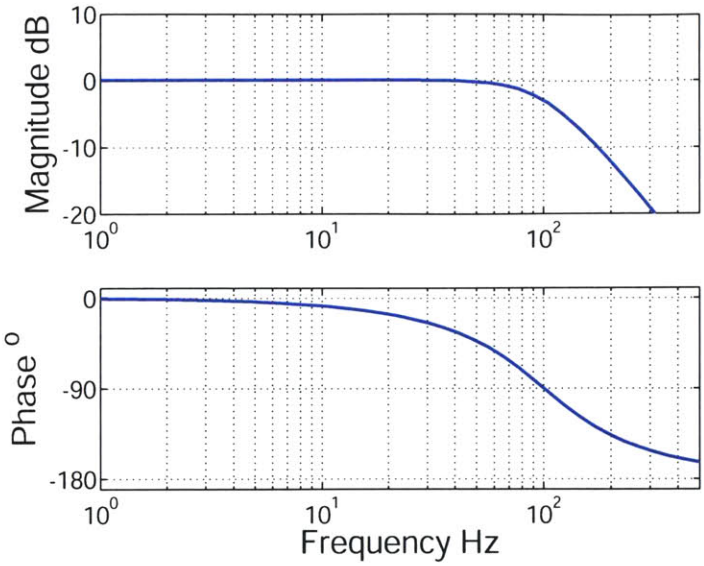


Figure 4-38: Frequency response plot of the example plant.

where $\omega_p = 200\pi$ (100 Hz) and $\zeta = 0.7$. Figure 4-38 shows the frequency response of the plant. For this example, we have selected the frequency of the trajectory to be $\omega_1 = 40\pi$ (20 Hz). Using our loop shaping technique, we selected $\phi_1 = \angle P(j\omega_1) = -0.284$ radians and $g_1 = 44.5$ for both AMAFC structures. The reference trajectory for our example is

$$y_d(t) = \sin(\alpha t)[\cos(\omega_1 t) + 0.25\sin(\omega_1 t)] \tag{4.114}$$

where α is varied from π to 4π radians/s (0.5-2 Hz). For the multiplicative AMAFC structure, we set

$$\frac{1}{\hat{A}(t)} = \begin{cases} 1/\hat{A}(t) & \text{for } |\hat{A}(t)| > 0.001 \\ sgn[\hat{A}(t)] * 1000 & \text{for } |\hat{A}(t)| < 0.001 \end{cases} \tag{4.115}$$

Figure 4-39 plots the simulated following error for the example plant under conventional AFC control, multiplicative AMAFC control, and additive AMAFC control with a trajectory modulation $\alpha = \pi$ r/s. The peak-to-peak following error is 30% with conventional AFC control, 1.2% with multiplicative AMAFC, and 0.4% with

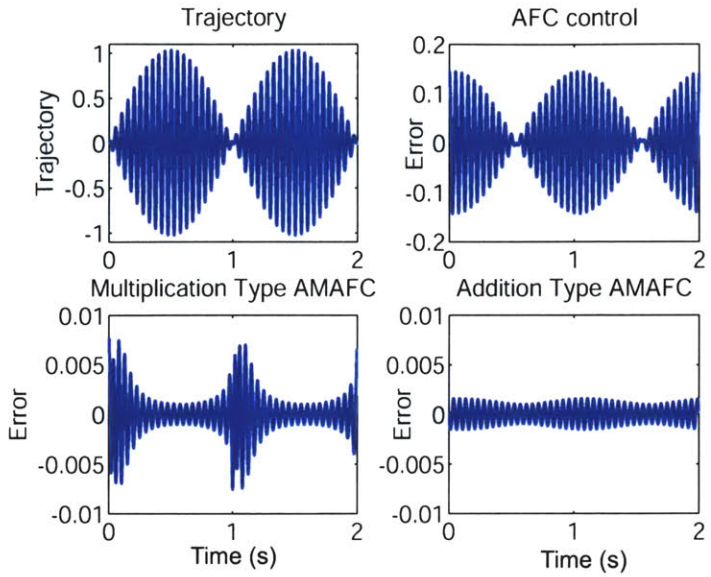


Figure 4-39: Simulated following error for the example plant following a 20 Hz sinusoid with an amplitude modulated at 0.5 Hz with conventional AFC control, multiplicative AMAFC control, and additive AMAFC control.

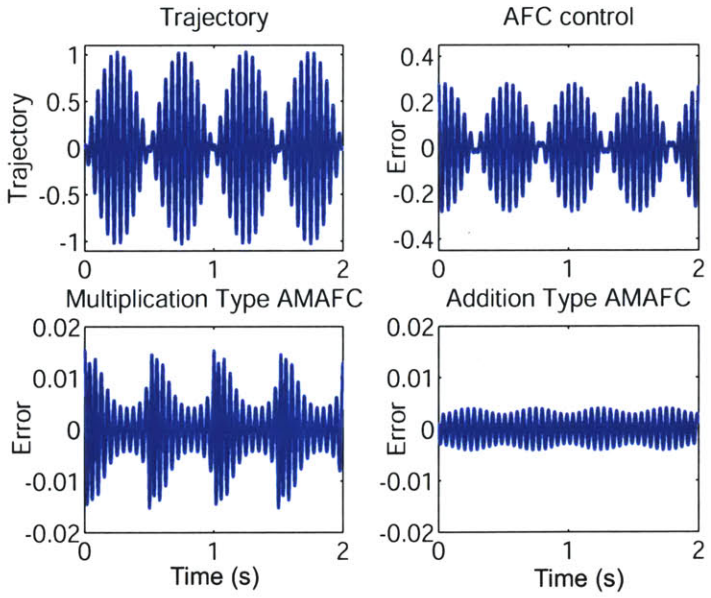


Figure 4-40: Simulated following error for the example plant following a 20 Hz sinusoid with an amplitude modulated at 1 Hz with conventional AFC control, multiplicative AMAFC control, and additive AMAFC control.

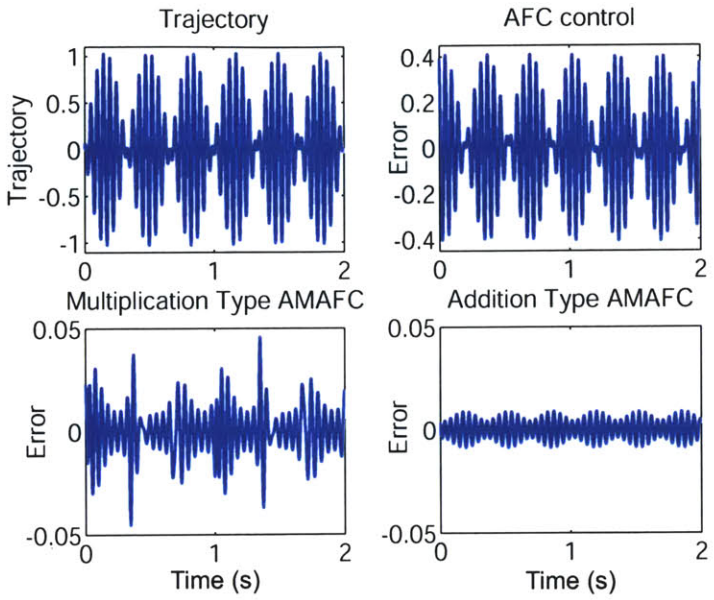


Figure 4-41: Simulated following error for the example plant following a 20 Hz sinusoid with an amplitude modulated at 1.5 Hz with conventional AFC control, multiplicative AMAFC control, and additive AMAFC control.

additive AMAFC control. Note: all of these simulated results are taken with the system in steady-state operation. The data shown here was collected 2 seconds after the simulation started. Figure 4-40 plots the simulated following error for the example plant under conventional AFC control, multiplicative AMAFC control, and additive AMAFC control with a trajectory modulation $\alpha = 2\pi$ r/s. The peak-to-peak following error is 50% with conventional AFC control, 3% with multiplicative AMAFC, and 0.8% with additive AMAFC control. Figure 4-41 plots the simulated following error for the example plant under conventional AFC control, multiplicative AMAFC control, and additive AMAFC control with a trajectory modulation $\alpha = 3\pi$ r/s. The peak-to-peak following error is 80% with conventional AFC control, 9% with multiplicative AMAFC, and 1.8% with additive AMAFC control.

As we can see from these results, AMAFC is a significant enhancement over conventional AFC control when following sinusoidal trajectories with time varying amplitude. The performance of the AMAFC systems drops off as the speed of the modulation increases since the quasi-static plant model used in the derivation of our

AMAFC structures breaks down. In general, the additive AMAFC structure is superior to the multiplicative AMAFC structure offering both better trajectory following and improved computational speed. In simulation, the run time of the multiplicative structure was twice that of the additive structure. We believe that these results can be improved further with the addition of a phase shift to the modulation terms so that

$$\hat{a}(t) = \frac{a}{|P(j\omega_n)|}\sin(\alpha t + \rho) \tag{4.116}$$

$$\hat{b}(t) = \frac{b}{|P(j\omega_n)|}\sin(\alpha t + \rho). \tag{4.117}$$

The determination of ρ is left for future work.

4.4 Summary

In this chapter, we have detailed a loop shaping technique for tuning control systems with AFC control. This loop shaping technique shows that the AFC phase shift is optimally set to $\phi_i = \angle P(j\omega_i)$. Our loop shaping perspective for selecting AFC controller gain g_i, allows the designer to ensure robust stability while maximizing the AFC controller performance. We also presented two strategies for extending AFC control to trajectories and disturbances with time varying magnitude. We term this extension AMAFC control. One strategy utilizes a multiplicative structure to exactly cancel errors with time-varying magnitude. The second strategy utilizes a more robust additive structure to more accurately follow trajectories with time varying magnitude. In the next chapter, we will utilize an Oscillator Amplitude Control perspective to approximate the convergence and error properties of AFC controller.

Chapter 5

Adaptive Feedforward Cancellation from an Oscillator Amplitude Control Perspective

In this chapter, we present a method of characterizing the convergence and error properties of control systems using Adaptive Feedforward Cancellation (AFC). Specifically, we view the AFC controller structure from an oscillator amplitude control (OAC) perspective. The OAC approach uses an averaging analysis to simplify a properly tuned single resonator AFC system into two single-input single-output amplitude control loops. In this chapter, we will simplify the sine channel of a single resonator AFC controller. Next, we verify our analysis by comparing the output of an AFC controller to the OAC approximation for a simulated system. We then simultaneously simplify both the AFC sine and cosine channels. Once again, we verify our analysis by comparing the parameter estimates for both the OAC approximation and AFC control. Next, we apply the OAC perspective to a multiple resonator AFC controller. Lastly, we explore the limitations of the OAC approach.

This chapter is derived principally from a forthcoming paper I co-wrote with Joe Cattell and Prof. David Trumper. Section 5.1 is a condensed version of Joe's analysis in [15]. Section 5.2 contains new simulations which start with the system in steady-state prior to the application of AFC or OAC control. Eliminating the transient

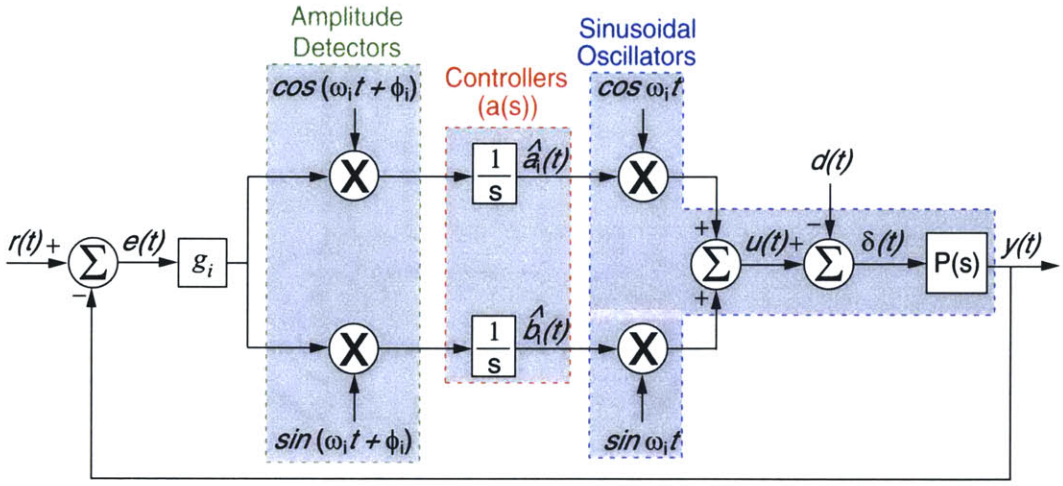


Figure 5-1: Single resonator AFC closed-loop block diagram designed to follow/reject a signal with a frequency ω_i. Figure taken from [15].

response of the plant results in a much better correlation between the OAC and AFC system response. Section 5.3 significantly expands on the dual channel work in [15] and contains a complete OAC analysis of the AFC system with and without phase adjustment. Sections 5.4 and 5.5 contain new simulated results based upon the revised dual channel results from Section 5.3 and the steady-state plant assumption. Lastly, Section 5.6 follows up Joe's suggestions for further work and presents an analysis of some limitations of the OAC approach.

5.1 Simplified Sine Channel of the Single Resonator AFC Controller

Figure 5-1 illustrates the single resonator AFC closed-loop block diagram. We view this system as the combination of two oscillator amplitude systems, where the sine and cosine channels correspond to two individual feedback loops in which the oscillation amplitude is to be controlled. The first sinusoidal modulators in Figure 5-1 are considered the amplitude detectors, and thus serve as sensors. Next, in concert with the gain g_i, the integrators act as the amplitude stabilization controllers, a(s), while

the second modulators and plant P(s) can be viewed as sinusoidal oscillators, whose output amplitude in y(t) is to be controlled to a desired level, as set by the component of frequency ω_i in the reference r(t). The reason for taking this perspective is that we can now write results in terms of the dynamics of the oscillation amplitude.

The sine and cosine channels in the AFC controller are thus coupled controllers in a multiple-input multiple-output (MIMO) sense. Under the assumption that we have correctly implemented the phase advance parameter $\phi_i = \angle P(j\omega_i)$, these two channels can be decoupled. That is, the system can be diagonalized. These two decoupled channels essentially yield equivalent closed-loop dynamics, as shown later. Thus in the following analysis, we simplify just the sine channel of the AFC controller into an equivalent oscillator amplitude control loop.

Figure 5-2 highlights the portion of the single resonator AFC closed-loop block diagram designed to follow/reject the sine component of a signal with frequency ω_i. We will analyze this system for loop dynamics and disturbance rejection by setting the reference signal equal to zero, r(t) = 0. We assume that the feedback loop has an input disturbance signal with a constant amplitude and single frequency component, d(t) $=b_i \sin(\omega_i t)$. In a manner analogous to Roberge's presentation of oscillator amplitude control (Chapter 12,[76]), we want to be able to analyze the AFC closed-loop output and error signals from the amplitude dynamics alone, independent of the detailed time variation of the sine and cosine waves. In order to do this, a few assumptions must be made.

First, we assume that the sine and cosine channels of the single resonator AFC system consist of multiple times-scales. By this we mean that the dynamics of the plant transfer function P(s) are considered to be much faster than the dynamics of the amplitude control loops. Said another way, the time scales on which the estimates of the Fourier coefficients $\hat{a}_i$(t) and $\hat{b}_i$(t) vary are slow compared to the plant output y(t). We call y(t) the fast state while $\hat{a}_i$(t) and $\hat{b}_i$(t) are considered slow states [79]. This means that P(s) has essentially settled to steady-state before the AFC feedback loop develops a significant error signal. We can ensure that the estimates of the Fourier coefficients vary relatively slowly when compared to y(t) by selecting a sufficiently

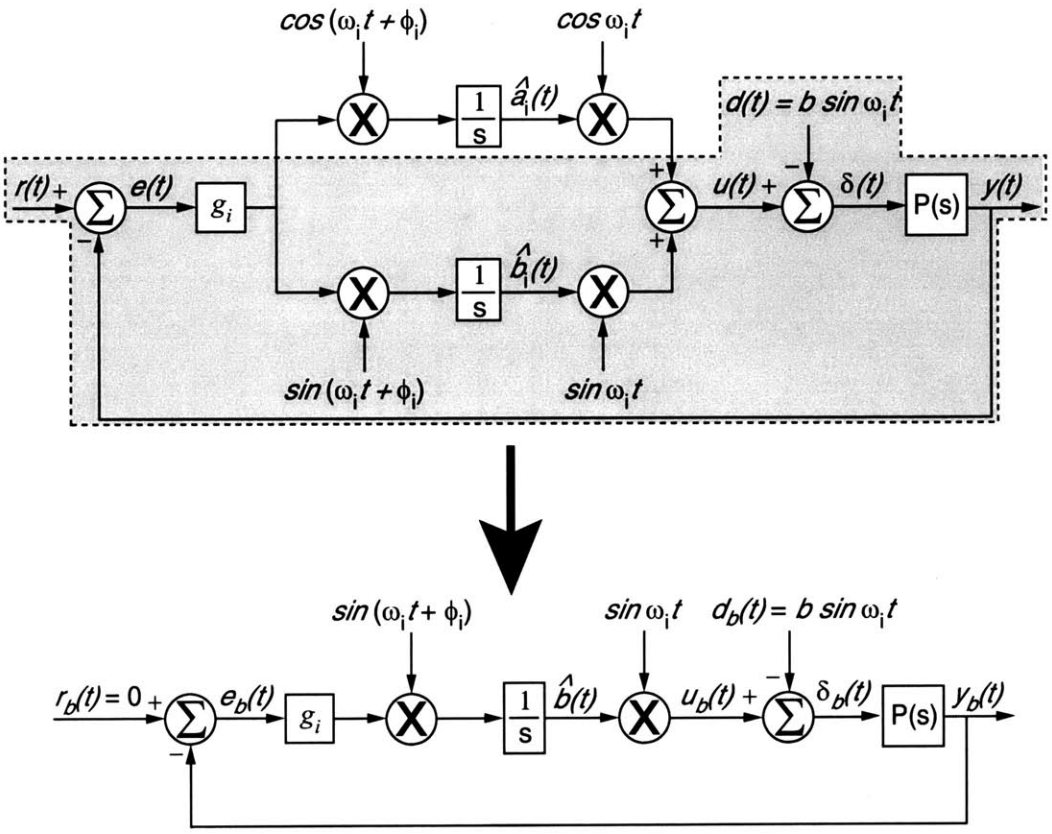


Figure 5-2: Closed-loop block diagram of the portion of the single resonator AFC system designed to follow/reject the sine component of a signal with frequency ω_i. Figure from [15]

low controller gain value g_i.

Secondly, we assume that the time variations of $\hat{a}_i(\mathrm{t})$ and $\hat{b}_i(\mathrm{t})$ are much slower than the AFC resonator frequency ω_i. The applicability of this statement is shown in the following analysis. Considering just the output sine channel of the single resonator AFC system, as shown at the bottom of Figure 5-2, we see that the control input into the plant is

$$\delta_b(t) = \beta(t)\sin(\omega_i t), \tag{5.1}$$

where $\beta(t)$ is some slowly time-varying amplitude. For simplicity, we have set $d_b(t) = b\sin(\omega_i t) = 0$. We will consider the effect of $d_b(t)$ in more detail later. In order to understand the envelope dynamics, at present we will assume that $\beta(t)$ is given by

$$\beta(t) = \sin(\alpha t). \tag{5.2}$$

Substituting (5.2) into (5.1) gives

$$\begin{aligned} \delta_b(t) &= \sin(\omega_i t)\sin(\alpha t) = \frac{1}{2}\left[\cos(\omega_i - \alpha)t - \cos(\omega_i + \alpha)t\right], && (5.3) \\ &= \frac{1}{2}\left[\cos\omega_- t - \cos\omega_+ t\right]. && (5.4) \end{aligned}$$

where

$$\begin{aligned} \omega_- &= (\omega_i - \alpha), && (5.5) \\ \omega_+ &= (\omega_i + \alpha). && (5.6) \end{aligned}$$

After reaching steady-state oscillation, the plant output $y_b(\mathrm{t})$, due to the control input defined in (5.4), is given by

$$y_b(t) = \frac{1}{2}\left[|\mathrm{P}(j\omega_-)|\cos(\omega_- t + \angle\mathrm{P}(j\omega_-)) - |\mathrm{P}(j\omega_+)|\cos(\omega_+ t + \angle\mathrm{P}(j\omega_+))\right]. \tag{5.7}$$

Now, we assume the frequency α of (5.2) is much less than the frequency ω_i of the modulator, $\alpha \ll \omega_i$, and further that the magnitude and phase of the frequency

response of the plant $P(j\omega)$ do not change significantly in the vicinity of ω_i. Then (5.7) can be approximated by

$$y_b(t) \simeq \frac{1}{2}\left(|P(j\omega_i)|\cos(\omega_- t + \angle P(j\omega_i)) - |P(j\omega_i)|\cos(\omega_+ t + \angle P(j\omega_i))\right). \tag{5.8}$$

Recalling $\omega_+ = (\omega_i + \alpha)$ and $\omega_- = (\omega_i - \alpha)$, this gives

$$y_b(t) \simeq \frac{1}{2}|P(j\omega_i)|\left(\cos(\omega_i t + \angle P(j\omega_i) - \alpha t) - \cos(\omega_i t + \angle P(j\omega_i) + \alpha t)\right). \tag{5.9}$$

Using the trigonometric relationship

$$\sin\alpha \sin\beta = \frac{1}{2}\left[\cos(\alpha - \beta) - \cos(\alpha + \beta)\right], \tag{5.10}$$

equation (5.9) reduces to

$$y_b(t) \cong |P(j\omega_i)|\sin(\omega_i t + \angle P(j\omega_i))\sin(\alpha t). \tag{5.11}$$

Thus, for a sufficiently slow time-varying $\hat{b}_i(t)$, the output of the sine channel for the single resonator AFC system can be approximated as

$$y_b(t) \cong |P(j\omega_i)|\sin(\omega_i t + \angle P(j\omega_i))\hat{b}_i(t). \tag{5.12}$$

Thus (5.12) shows that as long as the feedback loops for the sine and cosine channel in Figure 5-1 have a much lower crossover frequency than ω_i, we can analyze the amplitude dynamics of the AFC loop alone, independent of the time-variation of the sinusoids. This approximation is akin to the analysis given in Roberge Chapter 12,[76].

Our simplification begins by viewing the control input to the plant $\delta_b(t)$, as shown in Figure 5-2, as the difference between the estimated and actual disturbance signal Fourier coefficient multiplied by a sine wave,

$$\delta_b(t) = u_b(t) - d_b(t) = \left[\hat{b}(t) - b_i\right]\sin(\omega_i t). \tag{5.13}$$

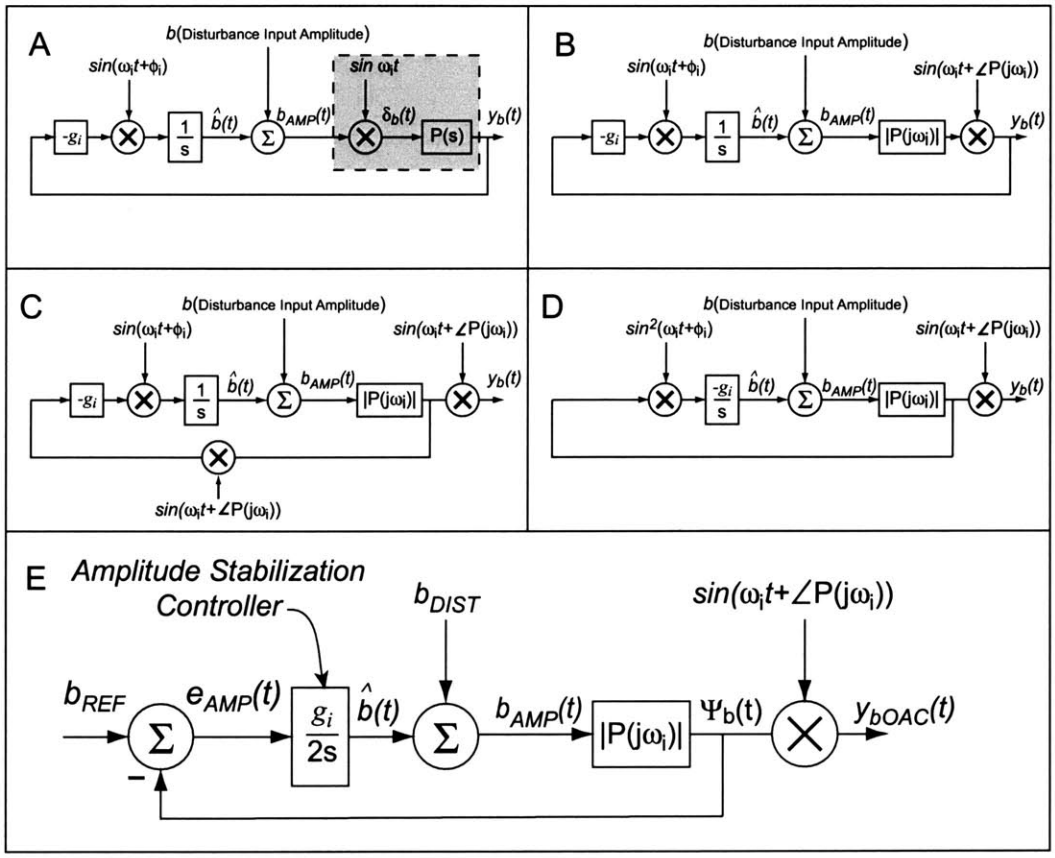


Figure 5-3: Simplification of the closed-loop block diagram for the sine channel of the single resonator AFC system. The reference signal $r_b(t)$ has been removed from frames A-D for simplicity and replaced in frame E. This Figure adapted from [15].

Next, we define the difference between the estimated and actual disturbance signal Fourier coefficient as

$$b_{AMP}(t) = \left[\hat{b}(t) - b_i\right], \tag{5.14}$$

where b_{AMP}(t) is the amplitude dynamics of the decoupled sine channel oscillator amplitude control loop. Thus, we can group the second sinusoidal modulator and plant transfer function together, as shown in Figure 5-3-A.

Next we employ the quasi-steady assumption in (5.12) to remove the plant transfer function from the feedback loop and view the output of the oscillator as a magnitude attenuated and phase shifted sinusoid of frequency ω_i. The resulting simplified block diagram is illustrated in Figure 5-3-B. The resulting oscillator output is given by

$$y_b(t) = b_{AMP}(t)|\mathrm{P}(j\omega_i)|\sin(\omega_i t + \angle \mathrm{P}(j\omega_i)), \tag{5.15}$$

which is equivalent to the results obtained in (5.12) with the addition of multiplication by the disturbance magnitude $b_{AMP}(t)$.

We further simplify the loop by moving the second modulator from the output, around the feedback loop, and grouping it with the modulator in front of the integrator. Since

$$\phi_i = \angle P(j\omega_i), \tag{5.16}$$

the two sines may be grouped together as

$$\sin(\omega_i t + \angle P(j\omega_i))\sin(\omega_i t + \phi_i)] = \sin^2(\omega_i t + \phi_i). \tag{5.17}$$

$$\tag{5.18}$$

Figures 5-3-B and 5-3-C illustrates this process. Using the following trigonometric relationship

$$\sin^2\alpha = \frac{1}{2}(1 - \cos 2\alpha), \tag{5.19}$$

the squared modulator term in Figure 5-3-D is equivalent to

$$\sin^2(\omega_i t + \phi_i) = \frac{1}{2}\left[1 - \cos 2(\omega_i t + \phi_i)\right], \tag{5.20}$$

which consists of a DC term and a second harmonic of ω_i. Since this amplitude feedback loop is inherently low-pass, the high-frequency second harmonic can be removed from the analysis, leaving only the average DC component. That is, we replace the $\sin^2$ term with simply a gain of $\frac{1}{2}$. A similar approximation and analysis is used in [10] and [103] to estimate the phase, magnitude, and frequency of periodic signals of unknown frequency.

Figure 5-3-E illustrates the final simplified sine channel of the single resonator AFC closed-loop block diagram. Here, we have defined the reference and disturbance input as b_{REF} and b_{DIST}, respectively, where b_{REF} is the magnitude of the reference input

$$r_b(t) = b_{REF} \sin \omega_i t. \tag{5.21}$$

Also, we have defined the average oscillator amplitude error and output amplitude envelope as e_{AMP}(t) and $\Psi_b(t)$ respectively. The average oscillator amplitude error is given by

$$e_{AMP}(t) = \left[b_{REF} - b_{AMP}(t)\right], \tag{5.22}$$

which equals the average DC component of the plant output (sinusoidal oscillator output) y_b(t) combined with the first sine wave modulator (amplitude detector), while the oscillator output amplitude envelope is

$$\Psi_b(t) = |\mathrm{P}(j\omega_i)| b_{AMP}(t), \tag{5.23}$$

which is equivalent to the amplitude dynamics of (5.12).

In Figure 5-3-E, we refer to the combination of the AFC proportional gain g_i and integrator as the OAC amplitude stabilization controller. This controller integrates

the average oscillator amplitude error to create the controlled amplitude output envelope $\Psi_b(t)$ which, when modulated with the $\angle P(j\omega_i)$ phase shifted sinusoid, provides the approximate AFC closed-loop output ($y_{bOAC}(t)$) to a reference/disturbance sine wave with frequency ω_i. The output of the sinusoidal oscillator is

$$y_{b_{OAC}}(t) = \Psi_b(t)\sin(\omega_i t + \angle \mathrm{P}(j\omega_i)), \tag{5.24}$$

which is equivalent to the output of the sine channel for the single resonator AFC system, as defined in (5.12). This closed-loop block diagram serves as our OAC perspective for the decoupled sine and cosine channels of the single resonator AFC system.

The loop transmission of the time averaged envelope of the decoupled sine and cosine channels is thus given by

$$L(s) = -\frac{g_i|P(j\omega_i)|}{2s}, \tag{5.25}$$

and the characteristic equation is of the form

$$1 - L(s) = 0, \tag{5.26}$$

or

$$2s + g_i|P(j\omega_i)| = 0. \tag{5.27}$$

Thus, under the assumption that the dynamics of the feedback loop are slow relative to the oscillation frequency, the OAC closed-loop pole is located at

$$s_{OAC} = -\frac{g_i|P(j\omega_i)|}{2} \ \sec^{-1}. \tag{5.28}$$

Further , the first order time constant τ_i is

$$\tau_i = \frac{2}{g_i|P(j\omega_i)|} \ \text{sec.} \tag{5.29}$$

The output of the sine only OAC system becomes

$$y_{bOAC}(t) = \Psi_b(t)\sin\left(\omega_i t + \angle P(j\omega_i)\right) \tag{5.30}$$

$$= b|P(j\omega_i)|e^{-t/\tau_i}\sin\left(\omega_i t + \angle P(j\omega_i)\right). \tag{5.31}$$

An equivalent result is presented in [77] using an averaging analysis. The advantage of this analysis is that it gives a model for the time-evolution of the envelope amplitude, and thus allows us to understand the settling characteristics of the AFC loop.

To this point the analysis assumes that the phase advance parameter is set properly as $\phi_i = \angle \mathrm{P}(j\omega_i)$. In the following analysis, we consider the case where this equality is not enforced. If we do not use the phase advance parameter in the single resonator AFC controller ($\phi_i = 0$), then the combined sinusoidal modulators are

$$\sin(\omega_i t + \angle \mathrm{P}(j\omega_i))\sin(\omega_i t). \tag{5.32}$$

Note that this analysis is only valid if $d(t)$ contains no cosine component and ignores the coupling from the sine to the cosine channel. The coupling will be explored in more detail later in this chapter. Using the trigonometric relationship

$$\sin\alpha\sin\beta = \frac{1}{2}\left[\cos(\alpha-\beta) - \cos(\alpha+\beta)\right], \tag{5.33}$$

equation (5.32) can thus be re-written as

$$\sin(\omega_i t + \angle \mathrm{P}(j\omega_i))\sin(\omega_i t) = \frac{1}{2}\left[\cos(\angle \mathrm{P}(j\omega_i)) - \cos(2\omega_i t + \angle \mathrm{P}(j\omega_i))\right]. \tag{5.34}$$

Once again dropping the higher harmonic terms, the loop transmission for the decoupled and simplified sine and cosine channels of the single resonator AFC system is given by

$$L(s) = -\frac{g_i|P(j\omega_i)|\cos(\angle \mathrm{P}(j\omega_i))}{2s}, \tag{5.35}$$

and the dominant closed-loop pole is now located at

$$s_{OAC} = -\frac{g_i|P(j\omega_i)|\cos(\angle P(j\omega_i))}{2} \text{ sec}^{-1}. \tag{5.36}$$

From (5.36), we see that if the plant contributes 0^o of phase lag at the resonator frequency ω_i, then as the pole moves to the origin the closed-loop settling time will be the same as given in (5.29). However, with $\phi_i = 0$, as $\angle \mathrm{P}(j\omega_i)$ approaches $\pm 90^o$, then the time constant grows $\tau_i \longrightarrow \infty$, since

$$\cos(\pm\frac{\pi}{2}) = 0. \tag{5.37}$$

Also, again for $\phi_i = 0$, if $\frac{\pi}{2} < \angle P(j\omega_i) < \frac{3\pi}{2}$, the resulting loop transmission effectively creates positive feedback and hence will be unstable. This result re-emphasizes the importance of using a properly chosen ϕ_i with an AFC controller.

The physical meaning of this result is as follow. We see that the single-channel OAC loop gain goes to zero as the unmatched phase $\phi_i \neq \angle P(j\omega_i)$ approaches $\pm\pi/2$. This occurs because the sine demodulator gives no DC output for a cosine input, since the average of the product $/sin(\omega_i t)\cos(\omega_i t)$ is zero. That is, if the AFC is outputting $\sin(\omega_i t)$, but the plant has $\pm\pi/2$ phase shift at this frequency, then the error signal will be a $\pm\cos(\omega_i t)$, which results in zero average at the integrator.

In this section, we simplified the sine channel of a single harmonic AFC resonator to determine the time-evolution of the amplitude estimate. In the next section, we simulate and compare the amplitude estimate and the total output of the approximate OAC system and a single resonator AFC controller.

5.2 Example of sine-channel OAC

In this example, we consider a single resonator AFC system designed to reject a constant amplitude disturbance input given by

$$d_1(t) = b_1 \sin \omega_1 t \tag{5.38}$$

where the parameters are selected as $b_1 = 1$ and $\omega_1 = 225$ rad/s. Setting the phase advance parameter $\phi_1 = \angle P(j\omega_1)$ decouples the sine and cosine channels and allows us to evaluate the system response using only the sine channel OAC model. The cosine channel will have identical dynamics. We assume a simple second order plant model

$$P(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{5.39}$$

where the parameters are selected as $\omega_n = 250$ rad/s and $\zeta = 0.707$. The key OAC parameters are

$$|P(j\omega_1)| = 0.777 \tag{5.40}$$

$$\phi_1 = -1.423 \text{ rad} \tag{5.41}$$

$$b_{REF} = 0 \tag{5.42}$$

$$b_{DIST} = 1. \tag{5.43}$$

In order to ensure that the simulation matches our quasi-static approximation from the previous Section, we allow the plant to reach a steady-state output in response to the disturbance input before activating the OAC and AFC control loops (we allow 1 sec. for the plant to reach steady-state in this simulation). We compare the systems at four different gains ($g_1 = 5$, 10, 25, and 50).

For this simulation, we are interested in five things:

$\hat{b}_{OAC}(t)$ = the Fourier coefficient estimate of the OAC approximation. This is the output of the OAC integrator.

$\hat{b}_{AFC}(t)$ = the Fourier coefficient estimate of the AFC sine channel. This is the output of the AFC sine channel integrator.

$\Psi_b(t)$ = the convergence envelope of the OAC approximation. This is the output of the OAC loop prior to the sinusoidal modulation to form $y_{bOAC}(t)$.

$y_{bOAC}(t)$ = the temporal output of the OAC approximation.

$y_{AFC}(t)$ = the closed loop temporal output of the simulated plant under closed-loop AFC control.

Note $\Psi_b(t)$, the convergence envelope of the OAC approximation is

$$\Psi_b(t) = b_{DIST}|P(j\omega_i)|e^{-\frac{t}{\tau_i}} \tag{5.44}$$

where τ_i is the time constant of the OAC approximation as defined in equation 5.29. The $\Psi_b(t)$ plotted in Figure 5-6 is calculated from this. This envelope should perfectly encapsulate the measured $y_{OAC}(t)$ and if our approximation is correct closely follow $y_{AFC}(t)$. An important thing to remember when we look at our results is the signal $y_{AFC}(t)$ is composed of the output of both the sine and cosine channels of the AFC controller.

Figure 5-4 plots the estimate of the Fourier coefficient $\hat{b}_{AFC}(t)$ and $\hat{b}_{OAC}(t)$. As we can see, the estimates match very well. The elimination of the 2nd harmonic from the OAC approximation is validated by these results since the 2nd harmonic is only apparent in the AFC estimate at high gain values $g_1 > 25$. Figure 5-5 shows a zoomed view of the Fourier coefficients for the $g_i = 50$ simulation to better illustrate the 2nd harmonic. Figure 5-6 compares the output $y_{AFC}(t)$, the output of $y_{bOAC}(t)$, and the amplitude envelope $\Psi_b(t)$. As we would anticipate from the results in Figure 5-4, the OAC approximation correlates very well with the AFC output while $\Psi_b(t)$ provides an accurate estimate of the AFC decay envelope.

These results show that there is excellent correlation between the approximate

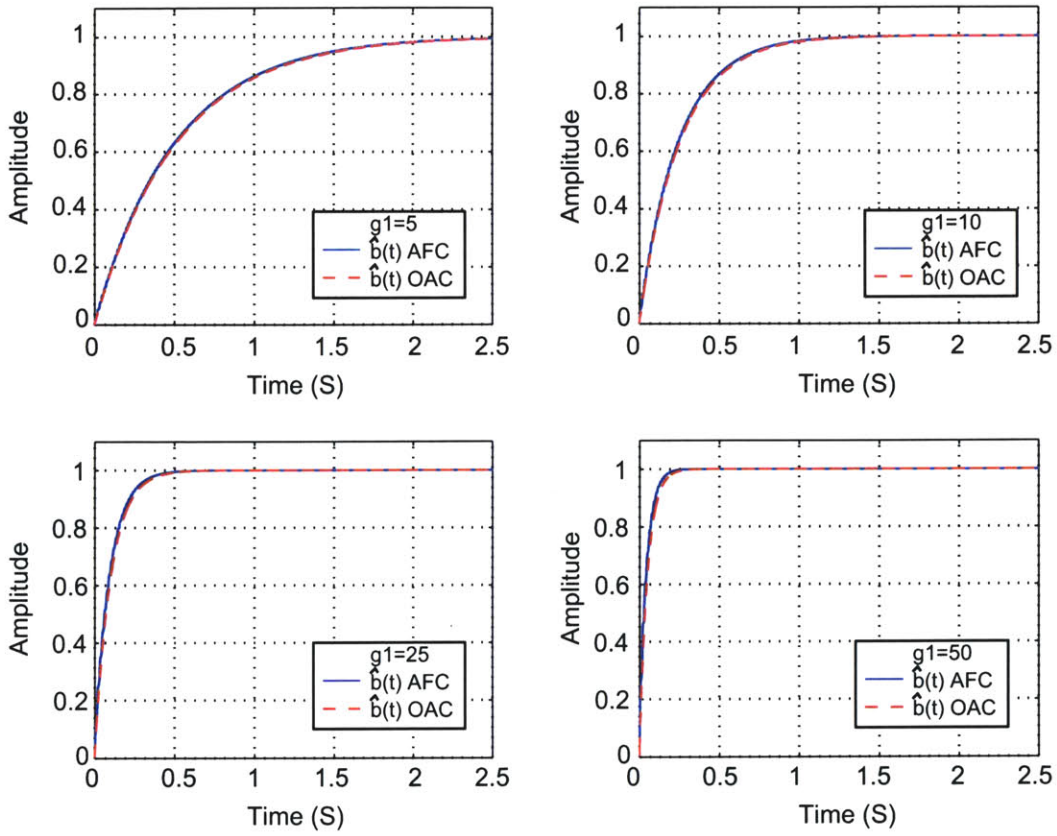


Figure 5-4: Comparison of the estimates of the Fourier coefficient $\hat{b}(t)$ of the disturbance signal $d(t)$ using a single resonator AFC system and a sine only OAC loop.

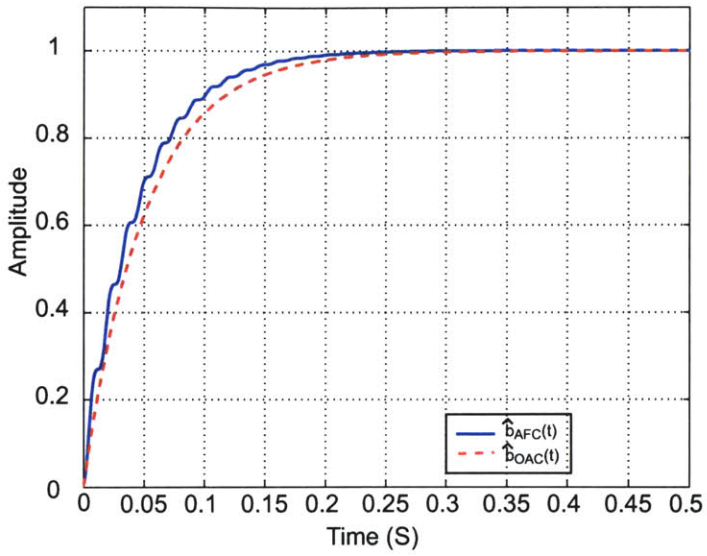


Figure 5-5: Zoomed view of the comparison of the estimates of the Fourier coefficient $\hat{b}(t)$ of the disturbance signal $d(t)$ using a single resonator AFC system and a sine only OAC loop for $g_i = 50$.

sine-only OAC system and the actual AFC system for a sine-only disturbance even at moderately high system gains. In the next section, we extend the OAC approximation to both the sine and cosine channels.

5.3 Simplified AFC system with Simultaneous Sine and Cosine channels

In this section we will employ the same techniques we used to simplify sine-only channels to simplify the entire AFC loop. Figure 5-7 illustrates the first few steps in the AFC loop simplification. Figure 5-7-A graphically shows the block diagram of a standard single resonator AFC system. The reference input to the system is defined as

$$r(t) = R_a \cos \omega_i t + R_b \sin \omega_i t \tag{5.45}$$

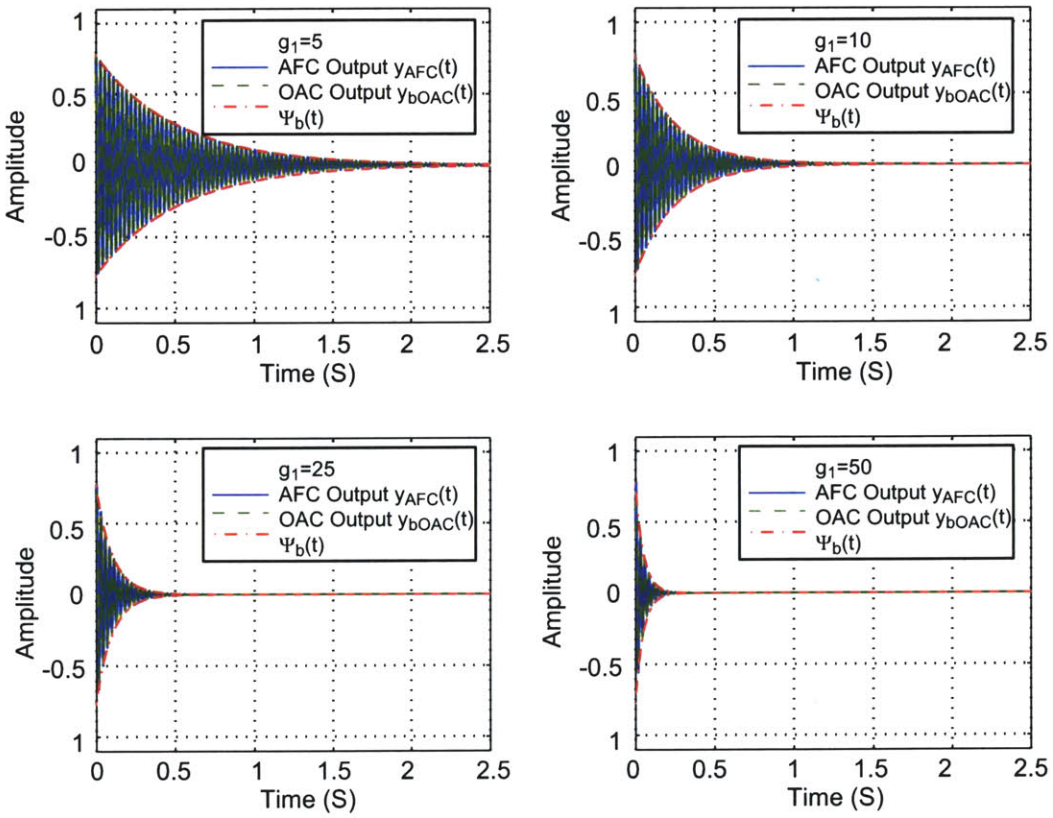


Figure 5-6: Comparison of system temporal outputs $y(t)$ and $y_{OAC}(t)$ for a single resonator AFC system and a single OAC loop. Note: the $y_{AFC}(t)$ signal contains both the sine and cosine channels.

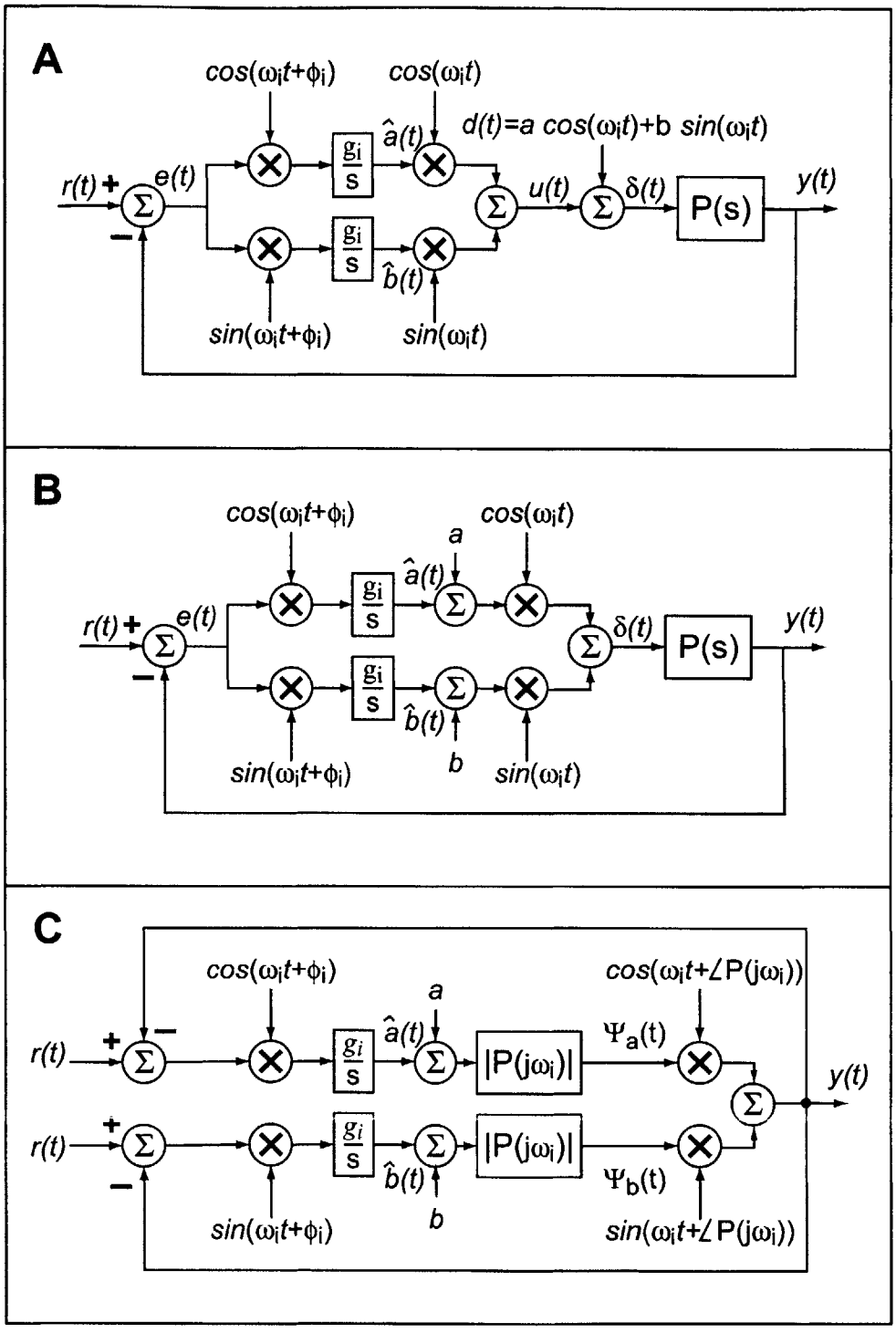


Figure 5-7: Simplification process for a single resonator AFC system.

while the disturbance input to the system is

$$d(t) = a_i \cos \omega_i t + b_i \sin \omega_i t. \tag{5.46}$$

To begin the simplification process, we divide the disturbance signal into it's sine and cosine component and move them back through the sine and cosine multiplier in the AFC resonator. The results of this process are shown in Figure 5-7-B.

Next, we employ the same quasi-steady assumption (5.12) we used in simplifying the sine-only loop to replace the plant transfer function ($P(s)$) with a simple gain $|P(j\omega_i)|$ and associated phase shift $\angle P(j\omega_i)$) in both the sine and cosine loops. To clarify the separation of the sine and cosine loops, we separate the sine and cosine loops on the left side of the block diagram by adding the reference signal ($r(t)$) into the sine and cosine loops individually. The results of these two steps are shown in Figure 5-7-C.

Next by following the signals around the loop we arrive at the following two relationships

$$\begin{aligned}
\frac{d}{dt}\hat{a}(t) &= g_i[R_a \cos \omega_i t + R_b \sin \omega_i t - \Psi_a(t) \cos(\omega_i t + \angle P(j\omega_i)) \\
&\quad -\Psi_b(t) \sin(\omega_i t + \angle P(j\omega_i))] \cos(\omega_i t + \phi_i) \qquad (5.47) \\
\frac{d}{dt}\hat{b}(t) &= g_i[R_a \cos \omega_i t + R_b \sin \omega_i t - \Psi_a(t) \cos(\omega_i t + \angle P(j\omega_i)) \\
&\quad -\Psi_b(t) \sin(\omega_i t + \angle P(j\omega_i))] \sin(\omega_i t + \phi_i) \qquad (5.48)
\end{aligned}$$

Employing Werner's trigonometric identities (equation (5.33) is Werner's identity for the sine functions) in equations (5.47) and (5.48) results in the following

$$\begin{aligned}
\frac{d}{dt}\hat{a}(t) &= \frac{g_i}{2}[R_a[\cos(-\phi_i) + \cos(2\omega_i t + \phi_i)] \\
&\quad R_b[\sin(-\phi_i) + \sin(2\omega_i t + \phi_i)] \\
&\quad -\Psi_a(t)[\cos(\angle P(j\omega_i) - \phi_i) + \cos(2\omega_i t + \phi_i + \angle P(j\omega_i))] \\
&\quad -\Psi_b(t)[\sin(\angle P(j\omega_i) - \phi_i) + \sin(2\omega_i t + \phi_i + \angle P(j\omega_i))]] \qquad (5.49) \\
\frac{d}{dt}\hat{b}(t) &= \frac{g_i}{2}[R_a[\cos(2\omega_i t + \phi_i) - \sin(-\phi_i)]
\end{aligned}$$

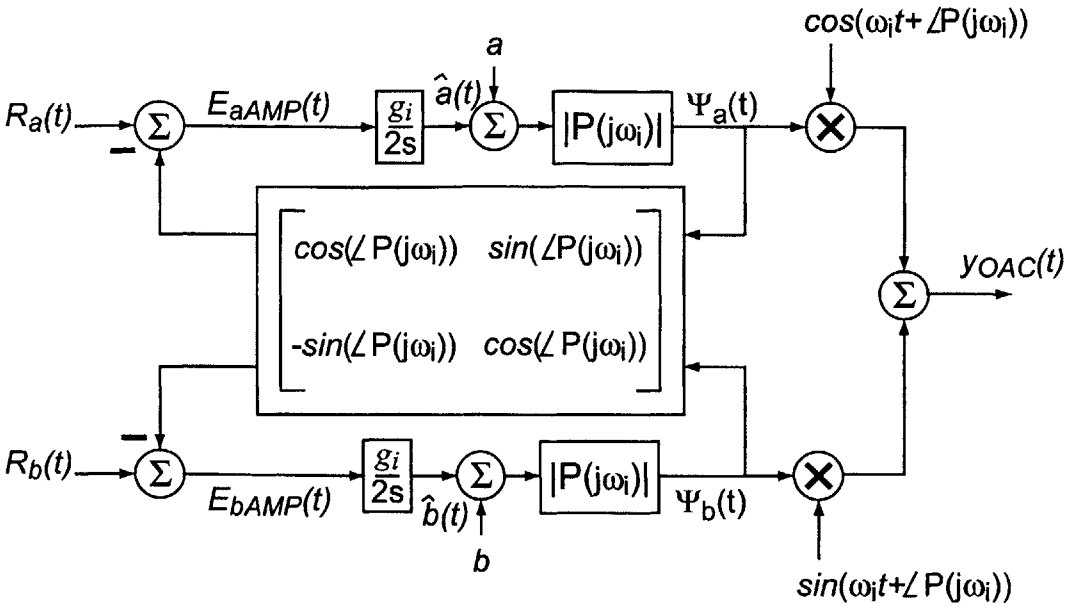


Figure 5-8: Simplified block diagram of AFC system with $\phi_i = 0$. Note decoupled input channel.

$$\begin{aligned} & R_b[\cos(-\phi_i) - \sin(2\omega_i t + \phi_i)] \\ & -\Psi_a(t)[\sin(2\omega_i t + \phi_i + \angle P(j\omega_i)) - \sin(\angle P(j\omega_i) - \phi_i)] \\ & -\Psi_b(t)[\cos(\angle P(j\omega_i) - \phi_i) - \cos(2\omega_i t + \phi_i + \angle P(j\omega_i))]]. \end{aligned} \quad (5.50)$$

Removing the double-frequency terms, these expressions simplify to

$$\begin{aligned} \frac{d}{dt}\hat{a}(t) &= \frac{g_i}{2}[R_a \cos(-\phi_i) + R_b \sin(-\phi_i) \\ &\quad -\Psi_a(t)\cos(\angle P(j\omega_i) - \phi_i) - \Psi_b(t)\sin(\angle P(j\omega_i) - \phi_i)] \quad (5.51) \\ \frac{d}{dt}\hat{b}(t) &= \frac{g_i}{2}[-R_a \sin(-\phi_i) + R_b \cos(-\phi_i) \\ &\quad +\Psi_a(t)\sin(\angle P(j\omega_i) - \phi_i) - \Psi_b(t)\sin(\angle P(j\omega_i) - \phi_i)]. \quad (5.52) \end{aligned}$$

Equations (5.51) and (5.52) show that the sine and cosine channels have both coupled inputs and coupled feedback loops. In the case where the phase advance parameter is not implemented ($\phi_i = 0$), the input channels are decoupled. The block diagram for this system is shown in Figure 5-8. Setting the phase advance parameter

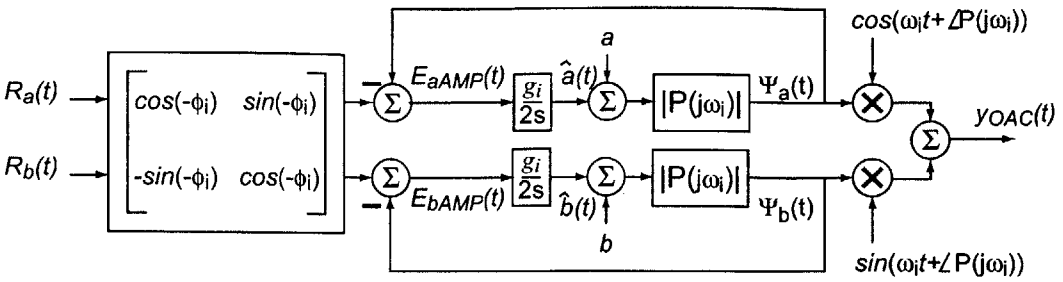


Figure 5-9: Simplified block diagram of AFC system with $\phi_i = P(j\omega_i)$. Note decoupled feedback channel.

$\phi_i = \angle P(j\omega_i)$ decouples the feedback channel but results in coupled inputs; Figure 5-9 shows the block diagram for this system. The matrices coupling the inputs and feedback loops in Figures 5-8 and 5-9 are similar to the rotation matrix used to couple the sine and cosine channels in Higher Harmonic Control (HHC) [37] and the rotation matrix used to couple the sine and cosine in Automatic Vibration Rejection (AVR) [89].

In the special case where $R_a(t)$ and $R_b(t)$ are constants or the ratio $R_b(t)/R_a(t)$ is a constant and we have selected $\phi_i = \angle P(j\omega_i)$, we can replace the inputs into the decoupled sine and cosine loop with equivalent reference signals. In the case that $R_a(t) = R_a$ and $R_b(t) = R_b$, the equivalent reference inputs may be expressed as

$$R_{aeqv} = R_a \cos(-\phi_i) + R_b \sin(-\phi_i) \tag{5.53}$$

$$R_{beqv} = -R_a \sin(-\phi_i) + R_b \cos(-\phi_i). \tag{5.54}$$

In the constant ratio case, the equivalent inputs are expressed as

$$R_{aeqv} = R_a(t)[\cos(-\phi_i) + K \sin(-\phi_i)] \tag{5.55}$$

$$R_{beqv} = R_a(t)[K \cos(-\phi_i) - \sin(-\phi_i)] \tag{5.56}$$

where

$$K = \frac{R_b(t)}{R_a(t)}. \tag{5.57}$$

We will use this result later to write an expression for the time output of the decoupled sine and cosine channels.

When $\phi_i = \angle P(j\omega_i)$, the loop transmissions of the sine and cosine loops are decoupled and equivalent:

$$L(s) = -\frac{g_i|P(j\omega_i)|}{2s}. \tag{5.58}$$

Thus the characteristic equation for both loops is

$$s + \frac{g_i|P(j\omega_i)|}{2} = 0 \tag{5.59}$$

with a pole at

$$s = -\frac{g_i|P(j\omega_i)|}{2} \tag{5.60}$$

and thus the associated time constant

$$\tau_i = \frac{2}{g_i|P(j\omega_i)|}. \tag{5.61}$$

This is the same result achieved for the simplified sine channel.

The time output of the dual channel OAC system $y_{OAC}(t)$ is given by

$$y_{OAC}(t) = |P(j\omega_i)|[\Psi_a(t)\cos(\omega_i t + \angle P(j\omega_i)) + \Psi_b(t)\sin(\omega_i t + \angle P(j\omega_i))] \tag{5.62}$$

In the special case where reference inputs $R_a(t)$ and $R_b(t)$ are constants or the ratio $R_b(t)/R_a(t)$ is a constant and we have selected $\phi_i = \angle P(j\omega_i)$, we can find the total

solution to each loops characteristic equation (5.59):

$$\Psi_a(t) = |P(j\omega_i)|R_{aeqv}(1 - e^{-t/\tau_i}) + a_i e^{-t/\tau_i}) \quad (5.63)$$

$$\Psi_b(t) = |P(j\omega_i)|R_{beqv}(1 - e^{-t/\tau_i}) + b_i e^{-t/\tau_i}) \quad (5.64)$$

where τ_i is determined from (5.61), R_{aeqv} is determined by either (5.53) or (5.53), and R_{aeqv} is determined by either (5.54) or (5.54). Recall a_i and b_i are the Fourier coefficients of the disturbance input. Thus

$$\begin{aligned} y_{OAC}(t) = & |P(j\omega_i)|[R_{aeqv}(1 - e^{-t/\tau_i}) + a_i e^{-t/\tau_i}]\cos(\omega_i t + \phi_i) \\ & +[R_{beqv}(1 - e^{-t/\tau_i}) + b_i e^{-t/\tau_i}]\sin(\omega_i t + \phi_i)]. \end{aligned} \quad (5.65)$$

As we learned in Chapter 4, the choice of resonator phase ϕ_i is important to the overall loop stability. Since an incorrect selection of ϕ_i results in cross-coupling of the OAC sine and cosine channels, this cross-coupling result effectively indicates a destabilization of the AFC system. More specifically, we would like the sine channel of an AFC resonator to detect and compensate only the sine component of the error signal. Similarly, we would like the cosine channel of an AFC resonator to detect and compensate only the cosine component of the error signal. When coupling is present, the sine and cosine channels are detecting and compensating for both sine and cosine components. If the cross-coupling is severe enough, the system will become unstable.

In this section, we applied our OAC approximation to both the sine and cosine channels of a single resonator AFC controller. We used this simplification to derive time output expressions for of estimates of the Fourier coefficients for both the sine and cosine. In the next Section, we use a simulation to verify our results.

5.4 Example of dual channel OAC

In Section 5.2, we compared the output of the OAC sine channel and the AFC system to a sine disturbance. In this section, we compare the output of the combined OAC

sine and cosine loops with $\phi_1 = \angle P(j\omega_1)$ (the system shown in Figure 5-9) to a single-resonator AFC system with a disturbance containing both sine and cosine terms. The plant parameters and disturbance frequency used for this evaluation are identical to that in Section 5.2. Just as in Section 5.2, we allow the plant to reach steady-state before engaging OAC or AFC control. The disturbance is given as

$$d_1(t) = a_1 \cos \omega t + b_1 \sin \omega_t \tag{5.66}$$

where $a_1 = 0.5$ and $b_1 = 1$. The reference inputs are

$$R_a = R_b = 0. \tag{5.67}$$

$y_{OAC}(t)$ can be determined by substituting the simulation conditions into Eqns. (5.65),(5.53),and (5.54). The decay envelope $\Psi_i(t)$ for $y_{OAC}(t)$ can be calculated by by combining the sine and cosine components of the $y_{OAC}(t)$ signal into a single sine term using harmonic addition as follows:

$$\begin{aligned} y_{OAC}(t) &= |P(j\omega_i)|[e^{-t/\tau_i}(a_i \cos(\omega_i t + \angle P(j\omega_i)) + b_i \sin(\omega_i t + \angle P(j\omega_i))] \\ &= |P(j\omega_i)|[\sqrt{a_i^2 + b_i^2} e^{-t/\tau_i} \sin(\omega_i t + \angle P(j\omega_i) + \theta_i)] \qquad (5.68) \\ \theta_i &= \tan^{-1}\left(\frac{a_i}{b_i}\right). \qquad (5.69) \end{aligned}$$

The decay envelope is thus

$$\Psi_i(t) = |P(j\omega_i)| \sqrt{a_i^2 + b_i^2} e^{-t/\tau_i}. \tag{5.70}$$

We evaluate this system at four gains ($g_1 = 5, 10, 25$, and 50).

Figure 5-10 plots the parameter estimates $\hat{a}(t)$ and $\hat{b}(t)$ for the AFC system and the simplified dual channel OAC system. Just as in section 5.2, we see that OAC and AFC parameter estimates are nearly identical for low gains but as the resonator gain is increased the second harmonic components in the AFC system become more prominent and the estimates diverge slightly. Figure 5-11 shows a zoomed view of

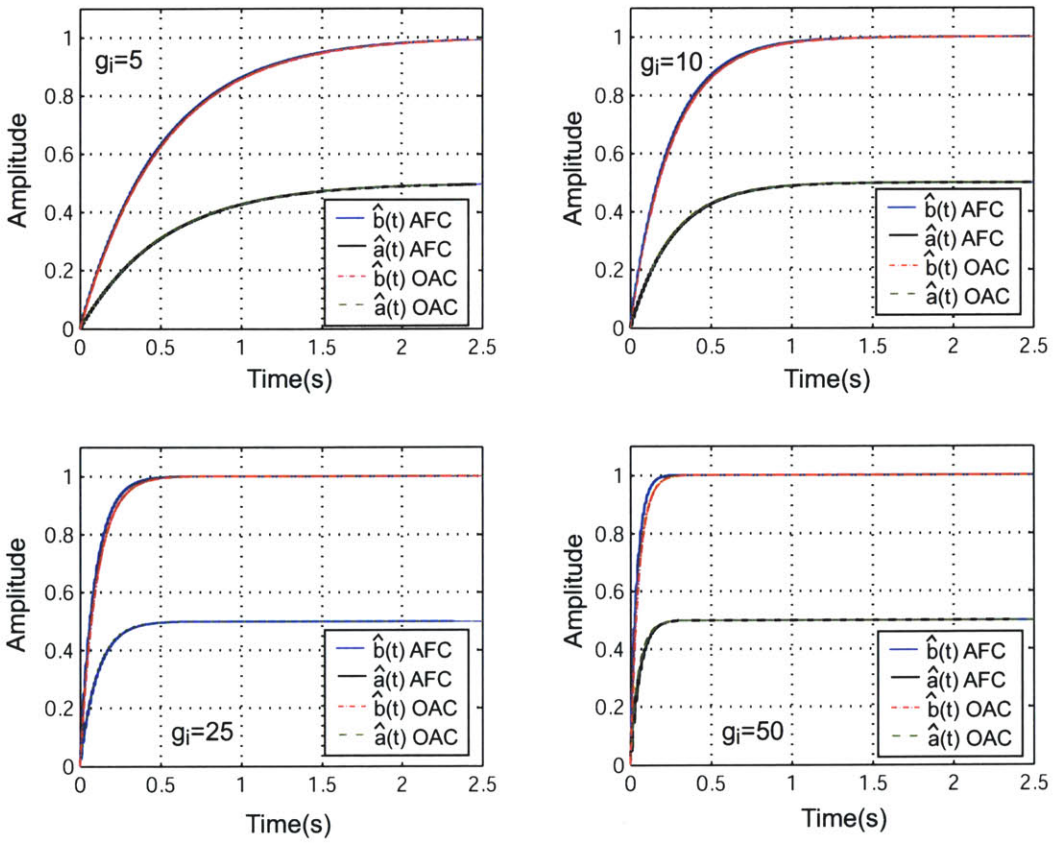


Figure 5-10: Plot of AFC and OAC parameter estimates $\hat{a}(t)$ and $\hat{b}(t)$ versus time for $g_1 = 5, 10, 25$, and 50.

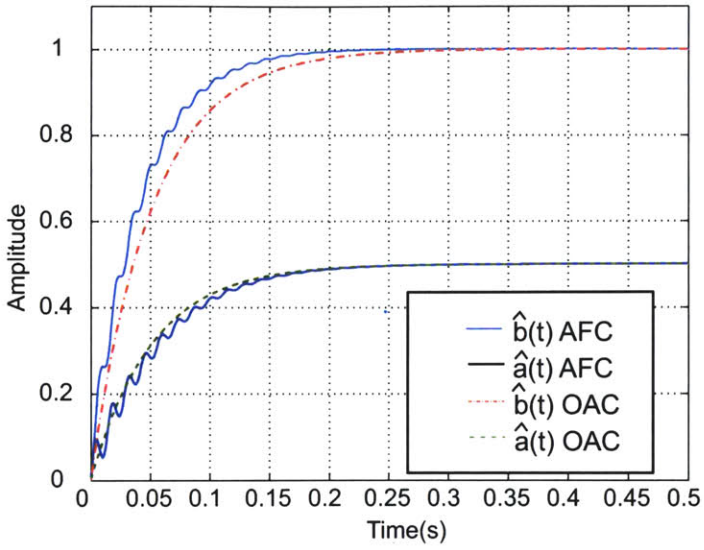


Figure 5-11: Plot of AFC and OAC parameter estimates $\hat{a}(t)$ and $\hat{b}(t)$ versus time for $g_1 = 50$.

$\hat{a}(t)$ and $\hat{b}(t)$ to better illustrate the divergence for $g_i = 50$. Figure 5-12 plots the output of the OAC system $y_{OAC}(t)$, the output of the AFC system $y_{AFC}(t)$, and the decay envelope $\Psi(t)$. The OAC system and the decay envelope provide an excellent approximation of the AFC system for the dual channel system. In the next Section, we will expand the OAC view point to controllers with multiple AFC resonators.

5.5 Multiple Resonator AFC System viewed from an OAC Perspective

In order to view a multiple resonator AFC system from an OAC perspective, we look at each AFC resonator individually. First we analyze the particular amplitude dynamics of each resonator separately and then use superposition to approximate the entire closed-loop response. For a multiple resonator AFC system designed to follow/reject N frequency components, there exist 2N estimates of the Fourier coefficients with N values of g_i, ω_i, and ϕ_i respectively. Implementing the phase advance parameter $\phi_i = \angle P(j\omega_i)$ effectively decouples the systems into 2N individual OAC

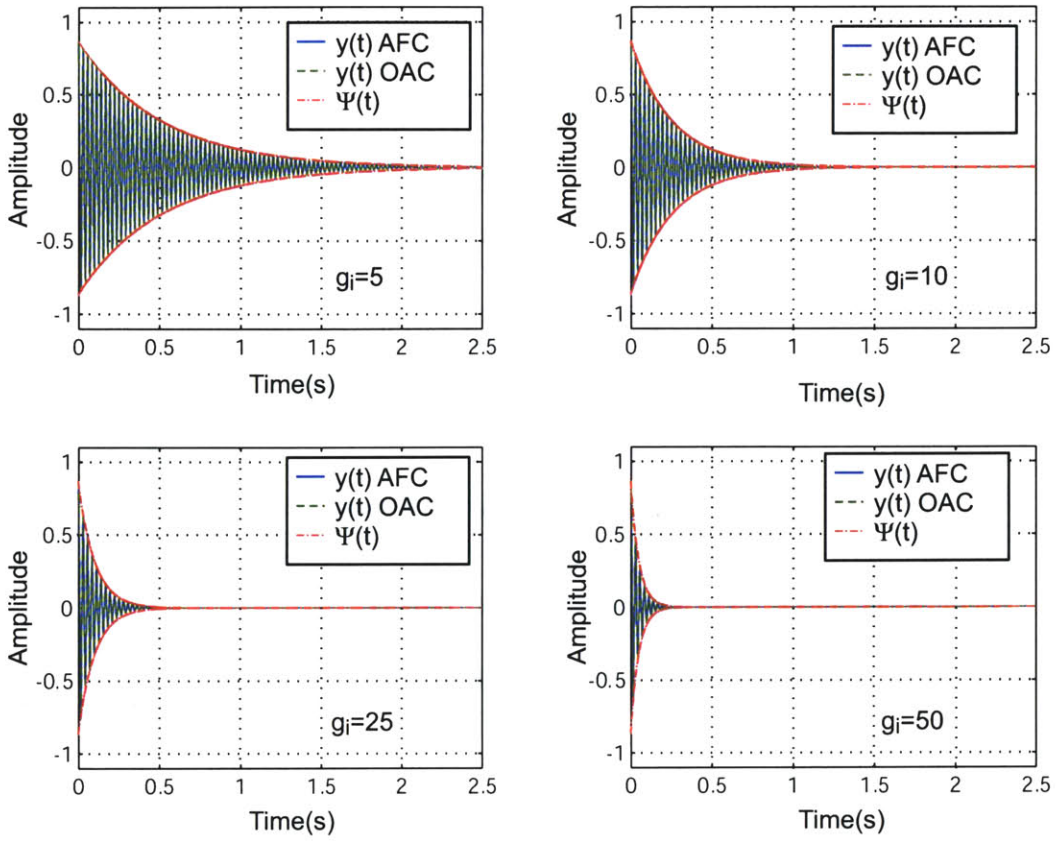


Figure 5-12: Plot of AFC and OAC system outputs ($y(t)$ and $y_{OAC}(t)$ respectively), and OAC decay envelope ($\Psi(t)$) versus time for $g_1 = 5, 10, 25$, and 50.

loops, where the dynamics of the sine and cosine channels at each frequency are identical and characterized by (5.36). Thus for a multiple loop OAC system, we can estimate the output amplitude in response to a disturbance input as

$$\Psi(t) \approx \sum_{i=1}^{N} \Psi_i(t). \tag{5.71}$$

From (5.70)

$$\Psi_i(t) = |P(j\omega_i)|\sqrt{a_i^2 + b_i^2}e^{-t/\tau_i} \tag{5.72}$$

where from (5.61)

$$\tau_i = \frac{2}{g_i|P(j\omega_i)|} \tag{5.73}$$

and where

$$d(t) = \sum_{i=1}^{N}[a_i \cos\omega_i t + b_i \sin\omega_i t]. \tag{5.74}$$

Defining the initial value of each decay harmonic as

$$\Psi_i(0) = |P(j\omega_i)|\sqrt{a_i^2 + b_i^2}. \tag{5.75}$$

allows us to express the total amplitude envelope as

$$\Psi(t) \approx \sum_{i=1}^{N} \Psi_i(0)e^{-t/\tau_i}. \tag{5.76}$$

At this point we should note that while we have considered only the response of OAC and AFC loops to disturbance inputs, a parallel analysis maybe performed to predict the system response to reference inputs. When considering the response to reference inputs, the error signals, $e_{AFC}(t)$ for the AFC system and $e_{AMP}(t)$ for the OAC system, should replace $y_{AFC}(t)$ and $y_{OAC}(t)$ as the system outputs of interest.

To show the utility of (5.76) lets us consider a system with three AFC resonators

i	ω_i	ϕ_i (rad)	$\lvert P(j\omega_i) \rvert$	g_i	a_i	b_i	τ_i (s)
1	75	-0.436	0.996	6.5	1	0.5	0.309
2	150	-0.925	0.941	6.5	0.25	0.5	0.327
3	225	-1.423	0.77	8.7	0.25	0.125	0.295

Table 5.1: Listing of the parameters for multiple resonator example.

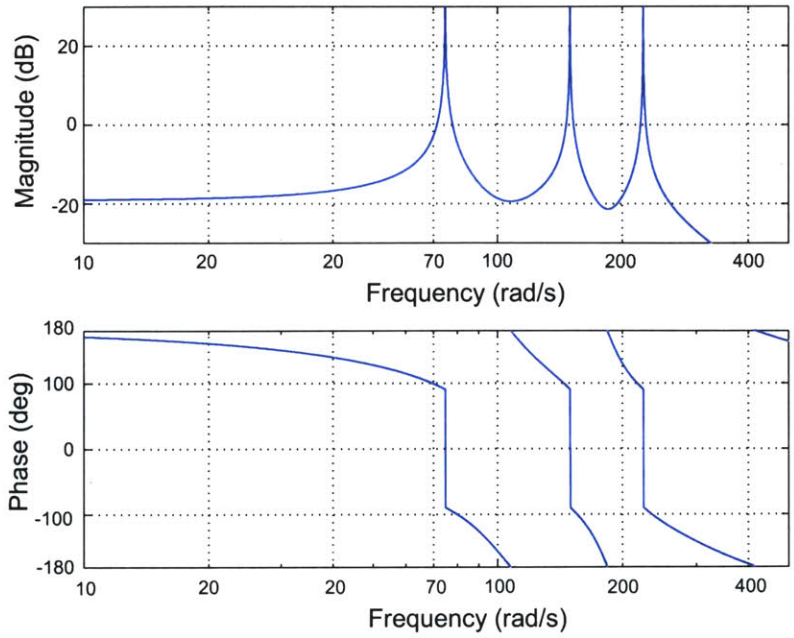


Figure 5-13: Frequency response of the negative loop transmission for the multi-resonator AFC system.

at $\omega_1 = 75$, $\omega_2 = 150$, and $\omega_3 = 225$ rad/s. Using the same second order plant model used in sections 5.2 and 5.4, we maximize the phase margin of the AFC system by setting $\phi_i = \angle P(j\omega_i)$ and select g_i such that the minimum gain margin of combined plant and AFC system is 20 dB. The key AFC, OFC, and disturbance parameters are listed in Table 5.1. Figure 5-13 shows the frequency response of the negative of the loop transmission of the combined plant and AFC system. This is a standard analysis (not OAC) and so is useful for stability considerations but does not give a model for the envelope dynamics. For this, we turn to an OAC perspective.

Figure 5-14 compares the estimate of the Fourier coefficients $\hat{a}_i(t)$ and $\hat{b}_i(t)$ for

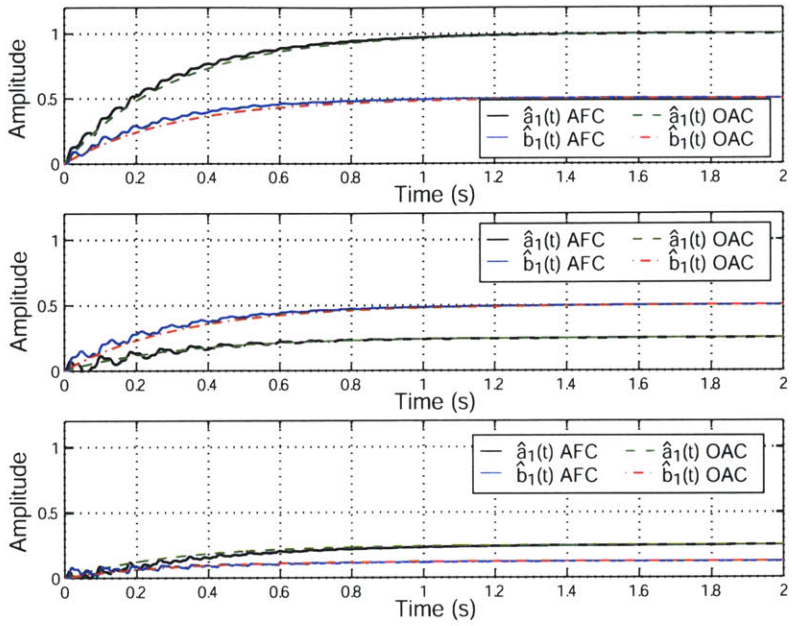


Figure 5-14: Comparison of the estimates of the Fourier coefficients $\hat{a}_i(t)$ and $\hat{b}_i(t)$ of the disturbance signal $d(t)$ using a multi-resonator AFC system and parallel OAC loops.

the multi-resonator AFC system and the multi-loop OAC approximation. As can be seen, the OAC systems does a good job of predicting the first order convergence of the AFC system but does not contain the higher harmonics that are generated by the AFC system through coupling of the individual resonators. Figure 5-15 compares the AFC temporal system output $y_{AFC}(t)$ and OAC system output $y_{OAC}(t)$. The OAC output $y_{OAC}(t)$ and the AFC output $y_{AFC}(t)$ match quite well. The predicted decay envelope $\Psi(t)$ does an excellent job of predicting the maximum magnitude of the AFC system output. The picture is slightly complicated by virtue of the sinusoidal signal adding destructively with positive amplitude.

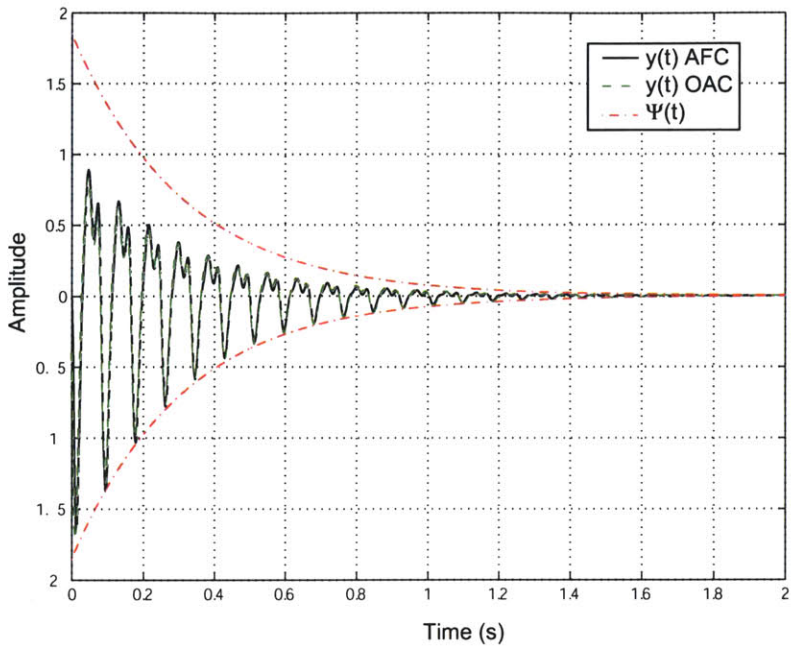


Figure 5-15: Comparison of system outputs $y(t)$ and $y_{OAC}(t)$ for a multi-resonator AFC system and parallel OAC loop.

5.6 Limitations of the Oscillator Amplitude Control Perspective

While the OAC approximation provides some valuable insight into the performance of AFC system, the OAC approach has a number of limitations. The first limitation is that the accuracy of the OAC approximation is highly dependant on the estimates $|P(j\omega_i)|$ and $\angle P(j\omega_i)$. As derived in section 5.1, the time constant τ_i of the OAC approximation is inversely proportional to $|P(j\omega_i)|$ while the magnitude of the amplitude envelope, $\Psi_i(0)$, is proportional to $|P(j\omega_i)|$. Thus any errors in the estimate of $|P(j\omega_i)|$ result in errors in both the convergence of the amplitude envelope and the magnitude of the amplitude envelope. Similarly in sections 5.1 and 5.3, we showed that the independence of the OAC loops is dependent on $\phi_i = \angle P(j\omega_i)$. If our estimate of $\angle P(j\omega_i)$ is off, significant errors may result in the OAC approximation due to the coupling of the AFC sine and cosine channels. In our examples, we use a

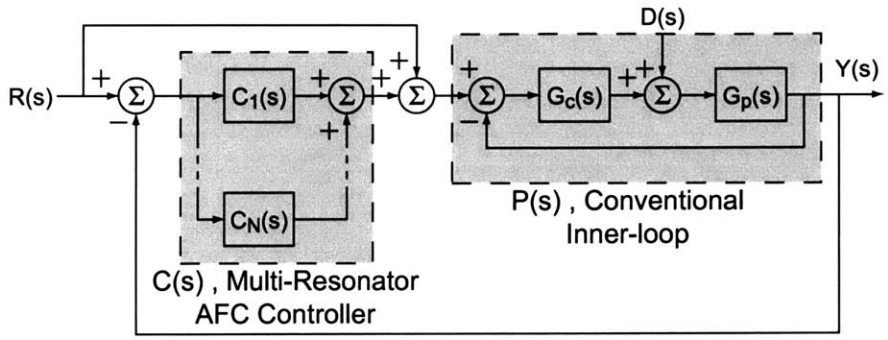


Figure 5-16: A common control system block diagram for systems using AFC resonators.

well-defined simulated plant, thus our simulated results do not include these errors. In real systems, the characteristics of the plant are often either not well known or vary with time. In the case of unknown plant characteristics, errors in the estimates of $|P(j\omega_i)|$ and $\angle P(j\omega_i)$ can be avoided by directly measuring the frequency response of the plant at the desired AFC frequencies. To reduce plant variations with time, an inner feedback loop using conventional control techniques (PID, lead-lag, pole-zero placement, ect.) can be established to provide a more predictable plant for the outer AFC loop.

The second limitation is the quasi-static assumption (5.12) used to simplify the sine and cosine loops. This assumption applies only if the magnitude and phase of the plant change slowly near the frequency of the resonator. In the following analysis, we will explore the limits of the quasi-static assumption by using a Fourier transformation analysis of the sine only loop signals to derive a more accurate model of the loop response.

The block diagram for the sine-only OAC structure is shown in Figure 5-2. If we break the loop after the integrator and assume that $\hat{b}(t)$ is of the form

$$\hat{b}(t) = \sin \alpha t \tag{5.77}$$

where $\alpha < \omega$. The signal $u(t)$, the input to the plant $P(s)$, becomes

$$u(t) = \sin \alpha t \sin \omega_0 t. \tag{5.78}$$

Since we are working with the frequency response of the plant $P(s)$, we need to convert the time signal $u(t)$ to $U(j\omega)$. The Fourier transformation of a signal is defined as

$$\mathcal{F}(x(t)) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt. \tag{5.79}$$

One of the properties of the continuous Fourier transform is that multiplication in the time domain is equal to convolution in the domain

$$\mathcal{F}(x(t)y(t) = X(j\omega) * Y(j\omega) = \int_{-\infty}^{\infty} X(jW)Y(j(\omega - W))dW. \tag{5.80}$$

More information of the properties of Fourier transforms can be found in [70]. Thus we can determine the Fourier transform of $u(t)$ as follows:

$$\mathcal{F}(u(t)) = \mathcal{F}(\sin \alpha t) * \mathcal{F}(\sin \omega_0 t) \tag{5.81}$$

$$\mathcal{F}(\sin \omega_0 t) = 2\pi \left(\frac{\delta(\omega - \omega_0)}{2j} - \frac{\delta(\omega + \omega_0)}{2j} \right) \tag{5.82}$$

$$\mathcal{F}(\sin \alpha t) = 2\pi \left(\frac{\delta(\omega - \alpha)}{2j} - \frac{\delta(\omega + \alpha)}{2j} \right) \tag{5.83}$$

$$U(j\omega) = 2\pi \left(\frac{\delta(\omega - \omega_0)}{2j} - \frac{\delta(\omega + \omega_0)}{2j} \right) * \left(\frac{\delta(\omega - \alpha)}{2j} - \frac{\delta(\omega + \alpha)}{2j} \right) \tag{5.84}$$

$$\begin{aligned} U(j\omega) = 2\pi \int_{-\infty}^{\infty} & \left(\frac{\delta(W - \omega_0)}{2j} - \frac{\delta(W + \omega_0)}{2j} \right) \\ & * \left(\frac{\delta(\omega - \alpha - W)}{2j} - \frac{\delta(\omega + \alpha - W)}{2j} \right) dW \end{aligned} \tag{5.85}$$

$$\begin{aligned} U(j\omega) = \frac{\pi}{2} [& \delta(\omega - \omega_0 + \alpha) - \delta(\omega - \omega_0 - \alpha) \\ & + \delta(\omega + \omega_0 - \alpha) - \delta(\omega + \omega_0 + \alpha)]. \end{aligned} \tag{5.86}$$

Using the frequency response of the plant $P(s)$ and the Fourier transform $U(j\omega)$, we can determine the Fourier transform of $y(t)$.

$$
\begin{aligned}
y(t) &= P(t) * u(t) && (5.87)\\
Y(j\omega) &= P(j\omega)U(j\omega) && (5.88)\\
&= \frac{\pi}{2}(|P(j(\omega_0-\alpha))|e^{j\angle P(j(\omega_0-\alpha))}\delta(\omega-\omega_0+\alpha)\\
&\quad -|P(j(\alpha+\omega_0))|e^{j\angle P(j(\alpha+\omega_0))}\delta(\omega-\omega_0-\alpha)\\
&\quad +|P(j(\alpha-\omega_0))|e^{j\angle P(j(\alpha-\omega_0))}\delta(\omega+\omega_0-\alpha)\\
&\quad -|P(-j(\alpha+\omega_0))|e^{j\angle P(-j(\alpha+\omega_0))}\delta(\omega+\omega_0+\alpha)). && (5.89)
\end{aligned}
$$

We then define

$$
\begin{aligned}
P_1 &= |P(j(\omega_0-\alpha))|e^{j\angle P(j(\omega_0-\alpha))} && (5.90)\\
P_2 &= |P(j(\alpha+\omega_0))|e^{j\angle P(j(\alpha+\omega_0))} && (5.91)\\
P_3 &= |P(j(\alpha-\omega_0))|e^{j\angle P(j(\alpha-\omega_0))} && (5.92)\\
P_4 &= |P(-j(\alpha+\omega_0))|e^{j\angle P(-j(\alpha+\omega_0))}. && (5.93)
\end{aligned}
$$

The Fourier transform of the error signal $E(j\omega)$ is expressed as

$$
\begin{aligned}
E(j\omega) &= \frac{g\pi}{2}(P_2\delta(\omega-\omega_0-\alpha)-P_1\delta(\omega-\omega_0+\alpha)\\
&\quad +P_4\delta(\omega+\omega_0+\alpha)-P_3\delta(\omega+\omega_0-\alpha)). && (5.94)
\end{aligned}
$$

The error signal is now multiplied by the second oscillator such that

$$
\begin{aligned}
\mathcal{F}\left(\frac{d}{dt}\hat{b}(t)\right) &= E(j\omega) * \mathcal{F}(\sin(\omega_0 t+\phi)) && (5.95)\\
&= E(j\omega) * \left(\frac{e^{j\phi}\delta(\omega-\omega_0)}{2j}-\frac{e^{-j\phi}\delta(\omega+\omega_0)}{2j}\right) && (5.96)\\
&= \frac{g\pi}{4j}((P_1e^{-j\phi}+P_4e^{j\phi})\delta(\omega+\alpha)-(P_2e^{-j\phi}+P_3e^{j\phi})\delta(\omega-\alpha)\\
&\quad -P_4e^{-j\phi}\delta(\omega+2\omega_0+\alpha)+P_3e^{-j\phi}\delta(\omega+2\omega_0-\alpha)\\
&\quad -P_1e^{j\phi}\delta(\omega-2\omega_0+\alpha)+P_2e^{j\phi}\delta(\omega-2\omega_0-\alpha)). && (5.97)
\end{aligned}
$$

Ignoring the higher harmonic components of (5.97), $\frac{d}{dt}\hat{b}(t)$ can be express as

$$\frac{d}{dt}\hat{b}(t) = -\frac{g}{2}|P|\sin(\alpha t + \Psi) \tag{5.98}$$

where

$$\begin{aligned}
|P| &= \sqrt{(P^- \cos\Psi^- + P^+ \cos\Psi^+)^2 + (P^- \sin\Psi^- + P^+ \sin\Psi^+)^2} && (5.99)\\
\Psi &= \tan^- 1\left(\frac{P^- \sin\Psi^- + P^+ \sin\Psi^+}{P^- \cos\Psi^- + P^+ \cos\Psi^+}\right) && (5.100)\\
\Psi^- &= \angle P(j(\omega_0 - \alpha)) - \angle P(j\omega_0) && (5.101)\\
\Psi^+ &= \angle P(j(\omega_0 + \alpha)) - \angle P(j\omega_0) && (5.102)\\
P^- &= |P(j(\omega_0 - \alpha))| && (5.103)\\
P^+ &= |P(j(\omega_0 + \alpha))| && (5.104)
\end{aligned}$$

As we can see, the inclusion of frequency response characteristics of the plant can result in a magnitude and phase shift in the estimate of $\hat{b}(t)$. This variation is particularly apparent when the magnitude and phase of the plant change rapidly with frequency. To illustrate this result, we use our sample second order system (5.39) but select $\omega_n = 250$ rad/s and $\zeta = 0.1$ so that the frequency response has more radical phase and magnitude shifts, and compare the resulting phase and magnitude shift in the estimate of $\hat{b}(t)$ as a function of α for $\omega_0 = 50$ and 250 rad/s. Figure 5-17 plots the phase and magnitude shift of $\frac{d}{dt}\hat{b}(t)$ as a function of α. As expected, the resonator at ω_0 is much more sensitive to α. This dependence on α once again emphasizes how critical it is to have a well-behaved plant.

5.7 Summary

In summary, the OAC perspective allows us to accurately predicted the convergence and error properties of control systems using AFC control for a broad range of controller gains g_i. Specifically, we found that AFC estimate of the Fourier coefficients of an error signal is a first order response whose time constant is proportional to the

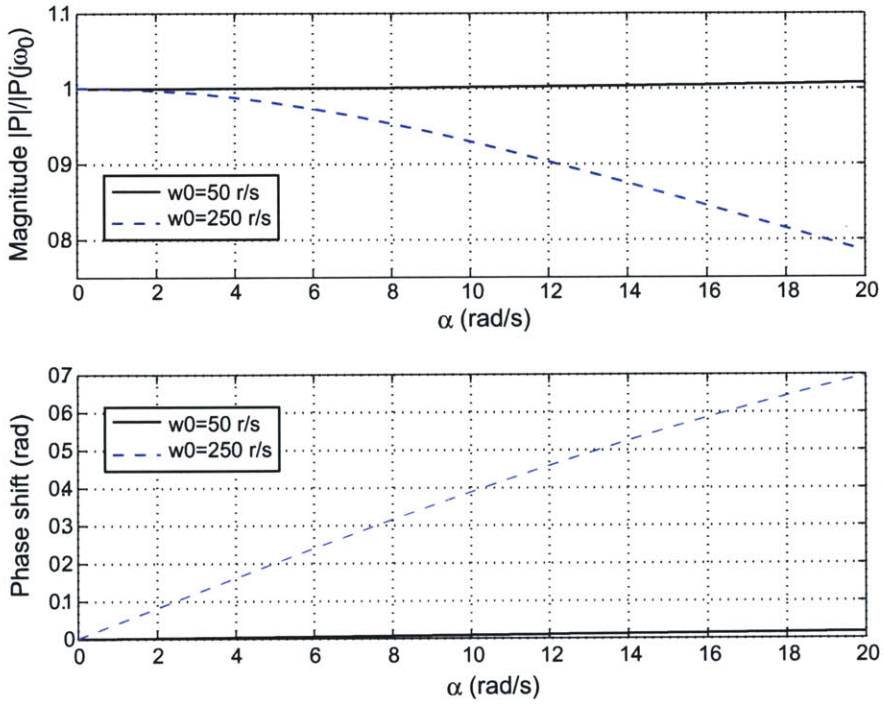


Figure 5-17: Phase and Magnitude shifts as a function of α for a second order plant ($\omega_n = 250$ r/s and $\zeta = 0.1$) for resonators at $\omega_0 = 50$ and 250 rad/s.

AFC gain g_i. Thus for rapid error cancellation, we would like to set the controller gain as high as possible. The OAC perspective also reinforced the importance of properly selecting the AFC phase parameter ϕ_i. Just as the loop shaping perspective of AFC control from Chapter 4 yields an optimal phase value of $\phi_i = \angle P(j\omega_i)$ for maximum phase margin, the OAC approach shows that setting $\phi_i = \angle P(j\omega_i)$ results in the fastest estimation of the Fourier coefficients of the error signal for a given controller gain. Lastly, we found that the OAC approximation is limited to systems where the phase and magnitude of the plant have only small changes across the frequencies of interest. In the next Chapter, we present the actual controller implementation where we actually apply our loop shaping techniques to apply AFC control to both the fast tool servo and the in-feed stage.

Chapter 6

Control System Implementation

This chapter details the design and implementation of the controllers used on the linear fast tool servo. In the first section, we detail the design and discrete-time implementation of the conventional controllers on both the fast tool servo and the hydrostatic stage. In the second section, we utilize the design rules detailed in Chapter 4 to apply adaptive feedforward cancellation to the fast tool servo. In the next section, we detail the design process used to implement AFC acceleration feedback on the hydrostatic stage. Lastly, we detail the implementation of an amplitude modulated adaptive feedforward cancellation controller on both the fast tool servo and machine stage.

6.1 Conventional Control

The position controllers for both the fast tool servo (FTS) and the reaction mass stage incorporate multiple control loops. The controller for the fast tool servo, shown in Figure 6-1 consist of a conventional lead-lag inner loop, a repetitive control outer loop, and a feedforward command channel. The controller for hydrostatic slide/reaction mass, shown in Figure 6-2 consist of lead-lag conventional controller incorporating a high frequency pole, a feedforward command channel which scales and phase shifts the fast tool servo trajectory, and a repetitive base acceleration control outer loop. Figure 6-3 shows the overall control structure with both the FTS and reaction mass

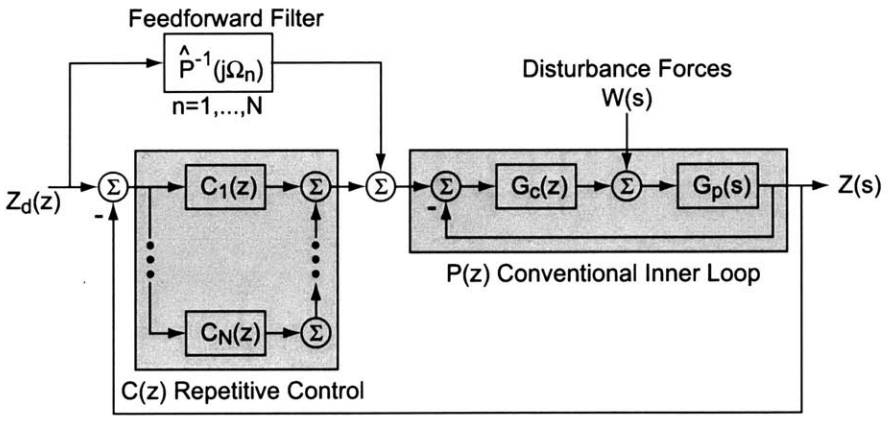


Figure 6-1: Overall fast tool servo control system block diagram, including a conventional lead/lag control loop, command pre-shifting, and repetitive control.

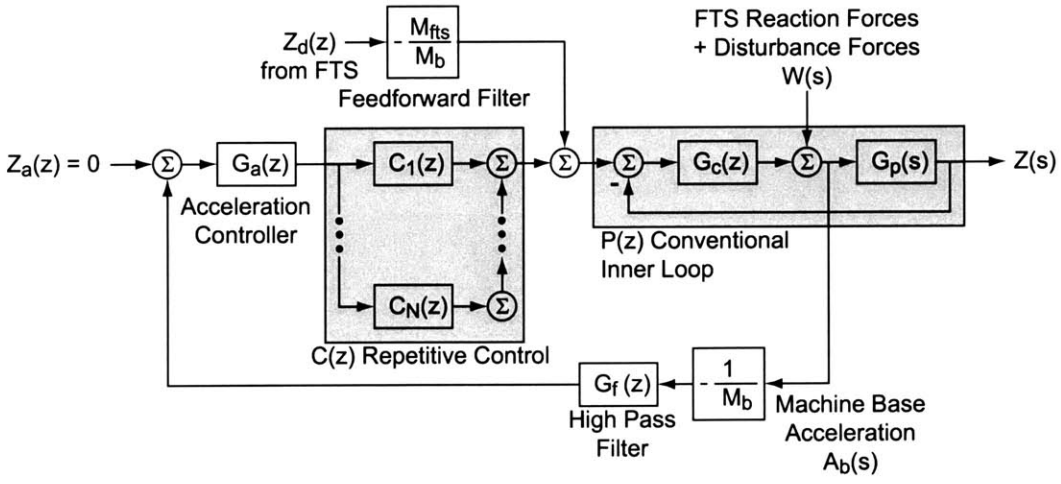


Figure 6-2: Overall hydrostatic stage/reaction mass control block diagram including a conventional lead/lag control loop, position command pre-shifting, and a repetitive base acceleration control outer loop.

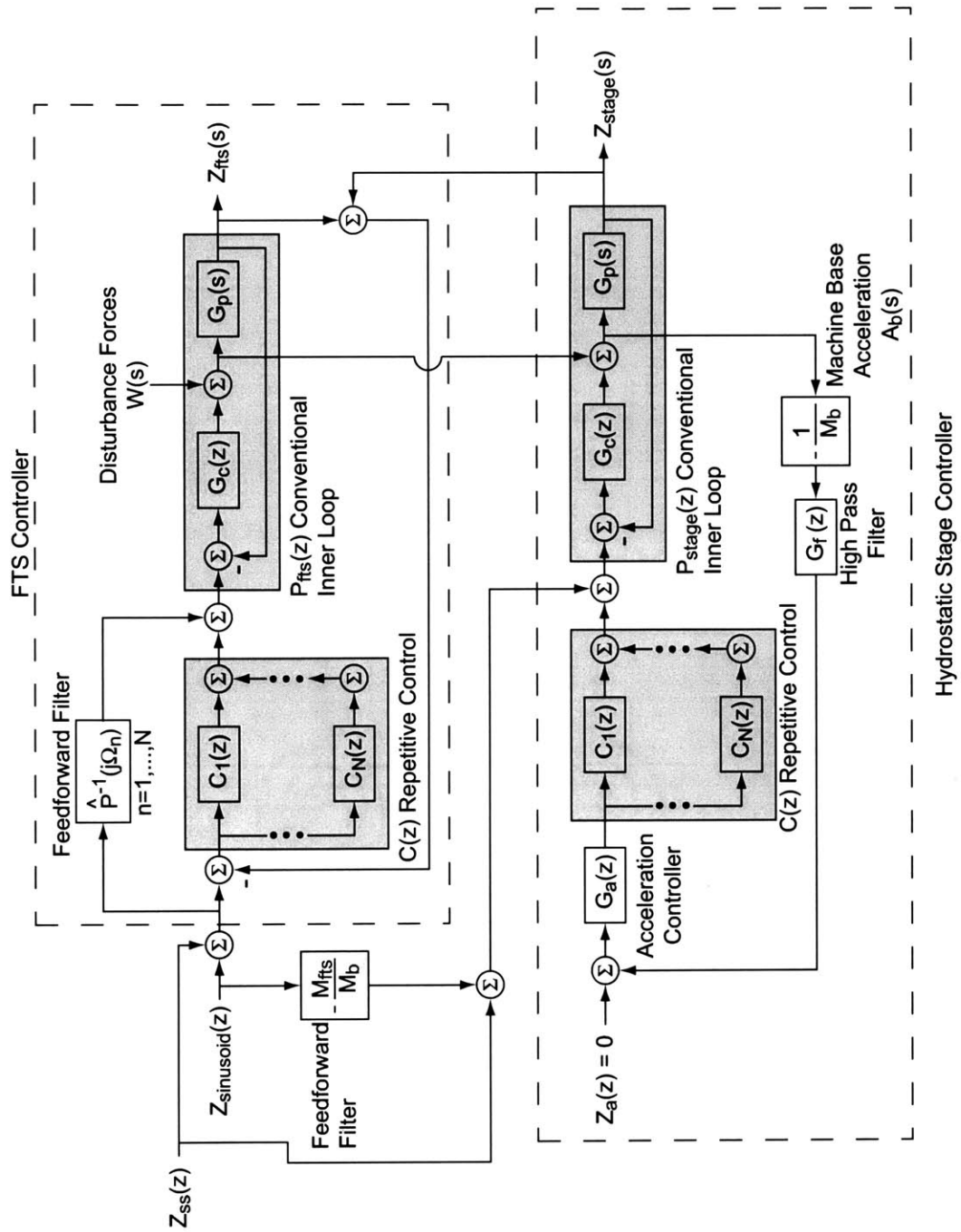


Figure 6-3: Block diagram of the overall control structure for our prototype fast tool servo with integral balance mass.

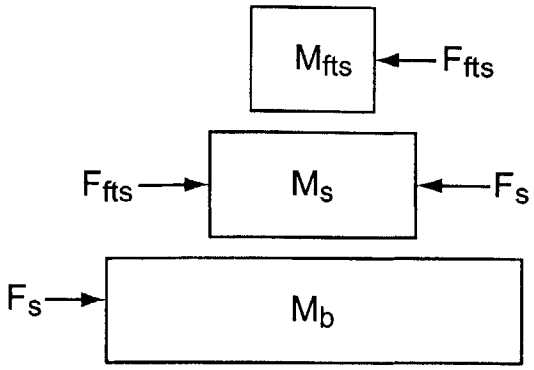


Figure 6-4: Free body diagram of the fast tool servo, hydrostatic stage, and machine base system.

controller. In this Figure, we see that the FTS control loops are coupled to the reaction mass controller through the mixing of the measured positions. Similarly, the reaction mass controller is coupled to the FTS controller through the FTS actuation forces. In this section, we will detail the conventional controllers for both the fast tool servo and hydrostatic stage.

6.1.1 Fast Tool Servo Model and Position Control

Plant Model

Figure 6-4 shows the free body diagram of our fast tool servo, hydrostatic stage/reaction mass, and machine base system. As can be seen, we propose to treat all three components as free masses linked only by the actuation forces required to follow the desired paths. In the case of the fast tool servo, this model is quite appropriate since the fast tool servo is supported by a nearly friction free air bearing and actuation forces are generated by a linear motor. The forces exerted by the motor cooling tubes and power wires have been minimized by design and are neglected in our model. Thus, the transfer function of the fast tool servo can be expressed as

$$G_p(s) = \frac{1}{M_{fts}s^2}. \tag{6.1}$$

Our ProE model predicted that the moving mass of the fast tool servo, including the cooling oil but not the power wires and tubing, would be 3.011 kg. The measured moving mass of the fast tool servo without oil, motor cables, tubing, and miscellaneous hardware was 2.93 kg. For analysis, I assumed that the moving mass of the fast tool servo was

$$M_{fts} = 3 \text{ kg}. \tag{6.2}$$

Forces are applied to the free mass by the linear motor which has a measured force constant of

$$K_f = 35.4 \frac{\text{N}}{\text{A}_{rms}} \tag{6.3}$$

as developed by Michael Liebman in his Master's thesis [46]. Since the motor has three phases and we are commanding DC current not the RMS current, we need to include the following conversion constant

$$K_{DCtoRms} = \frac{\sqrt{2}}{\sqrt{3}} \frac{\text{A}_{rms}}{\text{A}_{DC}} \tag{6.4}$$

which comes from the fact that for a three phase motor the DC current value is $\sqrt{3}/2$ of the peak current and the RMS current is $1/\sqrt{2}$ of the peak current. The Copley HPA-400-25 amplifier [1]has a measured gain of

$$K_{cop} = 3.98 \frac{\text{A}}{\text{V}} \approx 4 \frac{\text{A}}{\text{V}}. \tag{6.5}$$

The amplifier has a measured bandwidth of 2 kHz, a factor of 4 greater than our desired bandwidth, so I have ignored the amplifier dynamics. Thus the complete model of the continuous time elements of the model is

$$G_p(s) = \frac{K_{cop} K_{DCtoRms} K_f}{M_{fts} s^2} = \frac{38.5}{s^2}. \tag{6.6}$$

[1]This amplifier was kindly donated by Copley Controls Inc. of Canton, MA [19]

Since our controller is implemented digitally with a sample time of T_s=80 μs (12.5 kHz), we need to convert our continuous time model $G_p(s)$ to a discrete time Z transform equivalent. For this analysis I have chosen to employ a *bilinear transform*, often referred to as *Tustin's method* [22], where

$$s = \frac{2}{T_s}\frac{z-1}{z+1}. \tag{6.7}$$

The resulting FTS transfer function is

$$G_p(z) = 61.6x10^{-9}\frac{z^2+2z+1}{z^2-2z+1}. \tag{6.8}$$

Before we can begin to design our controller, we need to take into account a couple of digital hardware issues. First, on the DS1103 board the digital to analog converters (D/A) have a gain of 10 while the analog to digital converters (A/D) have a gain of 0.1. We are generating the amplifier input by converting a discrete time number into a voltage, thus we need to include the D/A gain in our model. We are measuring the FTS position, the transfer function output, with a MicroE M3500Si serial interface encoder. The serial interface between the DS1103 and encoder electronics is handled by the DS1103 Slave DSP. The code as implemented on the Slave DSP results in a one unit delay in collecting the position data. There is an additional unit delay generated in transferring the data from the Slave DSP to the main DSP. A discrete unit delay is represented as

$$\frac{1}{z}. \tag{6.9}$$

Thus with the D/A gain and the 2 unit delay, the transfer function becomes

$$G_p(z) = 616x10^{-9}\frac{z^2+2z+1}{z^4-2z^3+z^2}. \tag{6.10}$$

The serial interface transfers a 30-bit position word with a displacement of 0.0048828 μm/count. I have chosen to use mm for my internal controller units which means I

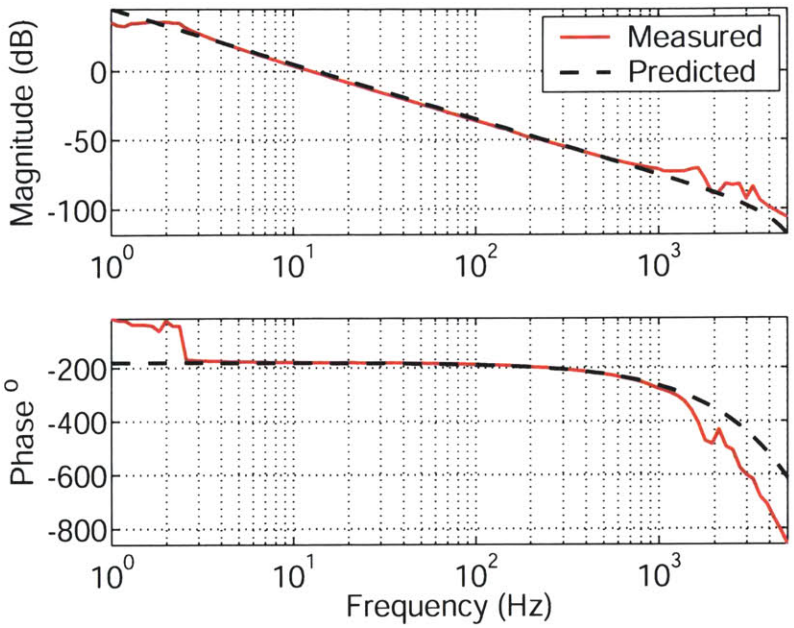


Figure 6-5: Experimental frequency response of the fast tool servo from a commanded current in dSpace (1 Amps peak) to measured fast tool servo position (mm). The solid line shows the measured response while the dashed line shows the predicted response.

need to include a conversion factor

$$K_{m-mm} = 1000 \frac{\text{mm}}{\text{m}}. \tag{6.11}$$

Lastly, I have employed saturation blocks in my discrete time controller to limit the current output of the amplifier. To prevent error in selecting the proper saturation limits, I have chosen to null the amplifier and D/A gains so that a unity controller output results in a 1 A amplifier output. The resulting discrete transfer function is

$$G_p(z) = 15.4x10^{-9} \frac{z^2 + 2z + 1}{z^4 - 2z^3 + z^2}. \tag{6.12}$$

Figure 6-5 show the open loop frequency response of the fast tool servo from a commanded current in dSpace (1 Amp peak) to measured fast tool servo position. The solid line shows the measured response while the dashed line shows the predicted

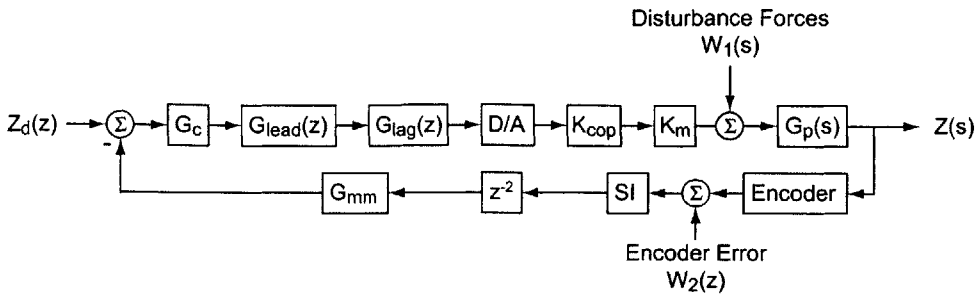


Figure 6-6: Block diagram of the fast tool servo conventional position control loop.

response. The measure and predicted response match quite well from 2 to 1000 Hz. The deviations at low frequencies are due to the high gain of the transfer function (ie. a very small current results in a large motion). Above 1000 Hz, we observe some unmodelled dynamics in the magnitude plot. In particular, it appears that the first resonance mode of the fast tool servo structure occurs at 1600 Hz. Since the peak associated with this resonance is relatively mild, we did not have to modify our control structure to attenuate this response. Similarly, we observe deviations in the phase response. Some of these deviations are directly related to the unmodelled fast tool servo dynamic response seen in the magnitude plot, but the rapid phase roll-off observed above 1000 Hz is likely caused by the amplifier dynamics (as previously noted the amplifier has a bandwidth of 2 kHz) which are not included in my model. Since the predicted and measured response match well in the frequencies of interest, I elected to leave out the amplifier model even after noting the increased phase rolloff at high frequency.

Position Control

The primary fast tool servo position control loop is composed of a single lead-lag compensator. Figure 6-6 shows the block diagram for the fast tool servo conventional position control loop. Both the lead and lag are placed in the forward path. The lead compensator has a pole-zero separation of $\alpha = 10$ and is centered at 300 Hz. In continuous time, a lead compensator has the form

$$G_{lead}(s) = \frac{\alpha\tau s + 1}{\tau s + 1} \tag{6.13}$$

where $\tau = 1/\omega$ and $\omega = 2\pi f$. I choose to implement this digitally as

$$G_{lead}(z) = \frac{z - Lz}{z - Lp} \tag{6.14}$$

where Lz and Lp are

$$Lz = e^{-\omega T_s/\sqrt{\alpha}} \tag{6.15}$$

$$Lp = e^{-\omega T_s\sqrt{\alpha}} \tag{6.16}$$

and T_s is the sample rate. This digitization method is called the *matched pole zero method* and is found by extrapolating from

$$z = e^{sT_s}, \tag{6.17}$$

the relationship between and $s-$ and $z-$ planes [30]. One disadvantage of this form is that the lead transfer function no longer has unity gain at low frequencies, but we adjust for this when we select the overall G_c. The Lead transfer function with $\alpha = 10$, a center frequency $f_c = 300$ Hz, and $T_s = 0.00008$ s is

$$G_{lead}(z) = \frac{z - 0.9534331}{z - 0.62072928} \tag{6.18}$$

In continuous time, a lag compensator has the form

$$G_{lag}(s) = \frac{\tau s + 1}{\tau s} \tag{6.19}$$

where $\omega = 1/\tau =$ and is the frequency of the Lag zero along the negative real axis. We have placed the zero at 30 Hz (one decade below the desired crossover frequency of 300 Hz). We use the following discrete-time equivalent for the lag compensator:

$$G_{lag}(z) = \frac{(T_s\omega_z + 2)z + (\omega_z T_s - 2)}{2(z - 1)} \tag{6.20}$$

where ω_z is the desired zero frequency in rad/s. This odd transfer function is a

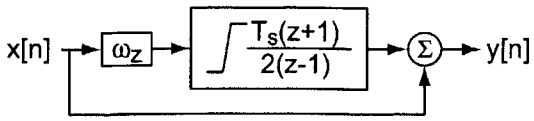


Figure 6-7: A discrete-time lag compensator implemented with an anti-windup trapezoidal integrator.

result of the lag implementation shown in Figure 6-7 where the lag compensator is constructed using an anti-windup trapezoidal integrator. If we were to implement the Lag with a backward integrator

$$G_{Int}(z) = \frac{T_s z}{z-1}, \tag{6.21}$$

the Lag transfer function would be

$$G_{lag} = \frac{(w_z T_s + 1)z - 1}{z - 1}. \tag{6.22}$$

We experimented with both forms and after finding no functional difference (the slightly more complex trapezoidal integration had no impact on the DSP turnaround time), we used the more accurate trapezoidal implementation [40]. The Lag transfer function with the zero at 30 Hz and $T_s = 0.00008$ s is

$$G_{lag}(z) = \frac{2.0150796z - 1.9849204}{2(z-1)} \tag{6.23}$$

To set the overall controller gain, we plotted the frequency response of the Lead, the Lag, and the Plant and selected $G_c = 948$ such that

$$|G_c G_{lead}(z) G_{lag}(z) G_P(z)| = 1 \tag{6.24}$$

at the desired crossover frequency of 300 Hz. The negative loop transmission of the compensated system is

$$L(z) = \frac{0.0294704z^4 + 0.0018343z^3 - 0.0571069z^2 - 0.00177235z + 0.0276776}{2z^7 - 7.24146z^5 + 9.72438z^4 - 5.72438z^3 + 1.24146z^2}. \tag{6.25}$$

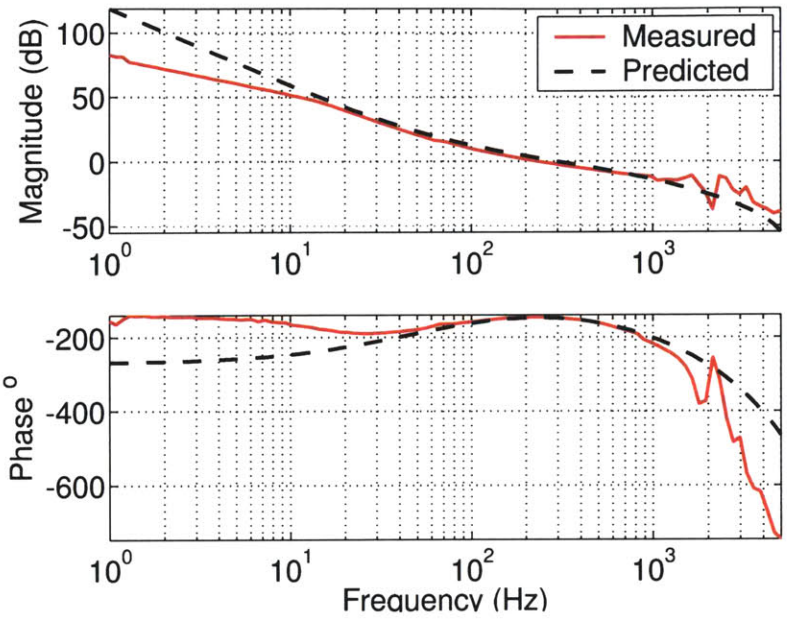


Figure 6-8: Experimental negative loop transmission response for the fast tool servo. The solid line shows the measured response.

Figure 6-8 shows the measured and predicted negative loop transmission for the fast tool servo. The loop transmission was measured while the fast tool servo was operating in closed-loop control and is calculated by comparing the measured error signal (input) to the fast tool servo position (output). For frequencies below 15.9 Hz, the reference position command is a 2 mm pk-pk sinusoid. Above 15.9 Hz, the reference input is selected such the fast tool servo has a maximum acceleration of 1 g. Note: the peak acceleration of any sinusoidal trajectory may be calculated as

$$z(t) = a \sin \omega t \tag{6.26}$$

$$\dot{z}(t) = a\omega \cos \omega t \tag{6.27}$$

$$\ddot{z}(t) = -a\omega^2 \sin \omega t \tag{6.28}$$

$$a = \frac{\ddot{z}_{max}}{\omega^2}. \tag{6.29}$$

As we can see, there is a fairly large difference between the measured and predicted loop transmissions. At low frequencies, the differences are a result of a extremely

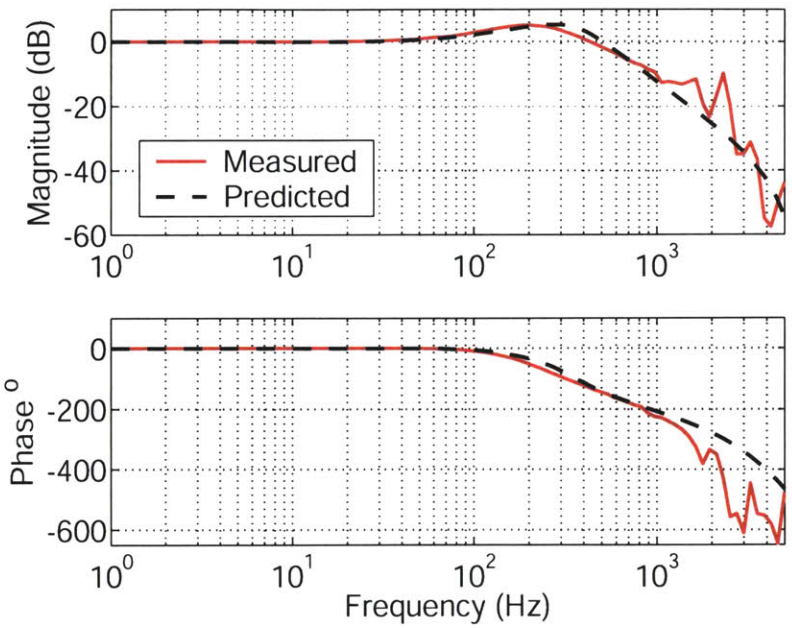


Figure 6-9: Experimental conventional fast tool servo frequency response. The dashed line indicates the predicted response.

small error resulting in a large motion. At high frequencies, we see a fairly large difference between the measured and predicted due to unmodelled dynamics.

The closed loop plant model is generated by converting the loop transmission $L(z)$ to state-space form and then setting

$$P(z) = \frac{L(z)}{1 + L(z)} \tag{6.30}$$

which results in the following state-space matrices for the closed loop plant model $P(z)$:

$$
A=\begin{vmatrix}
3.621 & -4.862 & 2.862 & -0.6207 & 0 & 0 & 0 & -0.01179 & -7.254E\text{-}4 & 0.02284 & 7.089E\text{-}4 & -0.01107 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 3.621 & -4.874 & 2.861 & -0.5979 & 7.089E\text{-}4 & -0.01107 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0
\end{vmatrix}
$$

$$
B=\begin{vmatrix}
1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0
\end{vmatrix},\ C=\begin{vmatrix} 0 & 0.01179 & 7.254E\text{-}4 & -0.02284 & -7.089E\text{-}4 & 0.0110.7 & 0 & 0 & 0 & 0 & 0 & 0 \end{vmatrix},
$$

$$
D = 0.
$$

(6.31)

Figure 6-9 compares the measured and predicted closed loop frequency response of the fast tool servo. The data was generated by commanding a 2 mm peak to peak sinusoid below 15.9 Hz and a trajectory with a maximum acceleration of 1 g above 15.9 Hz. The measured bandwidth is 540 Hz, very close to the desired 500Hz. As before, the dashed line indicates the predicted response. The predicted and modelled results match quite well at low frequencies, as expected since the large loop gain of both the measured and predicted loop transmission dominate the closed loop response. There is some variation between the predicted and measured responses between 200 and 500 Hz. Again, this is somewhat to be expected since the magnitude of the loop

transmission in this region is near unity with the shape of the closed loop response being very sensitive to small variations in loop gain and phase. If we look carefully at Figure 6-5, which plots the predicted and measured open loop transfer function, we find that there is a 1 dB difference between the measured and predicted magnitude and a 1 degree difference in the phase of the plant at 300 Hz.

Figure 6-10 shows the predicted and measured response with $G_{c2} = 0.89G_c$. As we can see, this 11% reduction in the predicted loop gain causes the two responses to match quite well up to 500 Hz. There are three likely sources of the difference between our predicted and measured systems. First, it is likely that we have underestimated the moving mass of the fast tool servo. As mentioned earlier, we weighted the moving portion of the fast tool servo without cables, hoses, and cooling oil. While we added 0.1 kg to the measured mass of the fast tool servo to account for these components, it is very possible we have underestimated the mass of the components. Second, it is very possible that as the system is tuned the force constant of the linear motor (K_m) is lower than predicted. As described in Chapter 3, we are generating the commutation for the linear motor in Simulink. This commutation is dependent on relating the electrical position of the motor to the measured position of the fast tool servo. We determined this relationship by running 1 Amp through 2 motor phases and then measuring the resulting motor location. This physical location is electrically 90 degrees from the position of maximum force. This allowed us to determine the relationship between FTS location and commutation phase. Any errors in this relationship results in a reduction in motor force. We selected the commutation constant from an average motor position based on multiple trials but there was a fair amount of variation in the actual neutral location so I suspect there is some error in this constant. Lastly, it is possible that the amplifier dynamics are affecting the magnitude of the response.

As a point of interest, we also implemented a controller where the designed loop transmission crossover is 500 Hz. The controller was identical to the one described earlier with the lead center frequency moved to 500 Hz and the gain selected to give unity magnitude at 500 Hz. Figure 6-11 shows the measured and predicted closed-

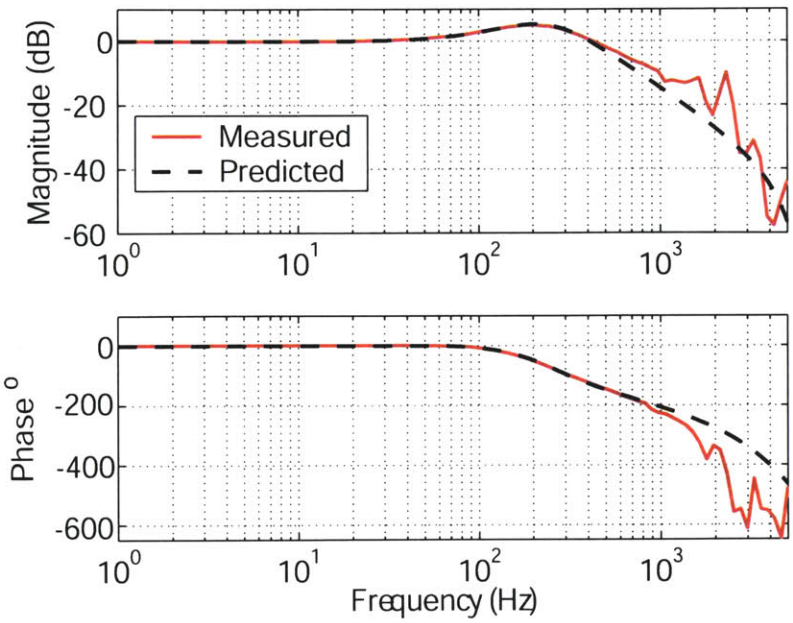


Figure 6-10: Experimental conventional fast tool servo frequency response. The dashed line indicates the predicted response.

loop frequency response. The measured -3 dB bandwidth with the 500 Hz crossover is 955 Hz. While this controller configuration is stable, it has a number of issues. First, the peak magnitude is 13 dB. This means that the step response of the fast tool servo will be quite oscillatory which is not desirable. Second, the unmodeled dynamics at 1600 Hz could become an issue in with this controller implementation. We used the 300 Hz implementation for all of the results presented in this thesis.

6.1.2 Hydrostatic Stage model and Position Control

Plant Model

Figure 6-4 shows the free body diagram for the entire fast tool servo, hydrostatic stage, and machine base system. As discussed in Section 6.1.1, we are treating all of the system components as free masses connected only by the actuator forces. As seen in Section 6.1.1, this assumption works quite well for the linear fast tool servo which is supported by an air bearing and driven by a linear motor. In this section, we will see that this basic model does not fit the hydrostatic stage quite as well.

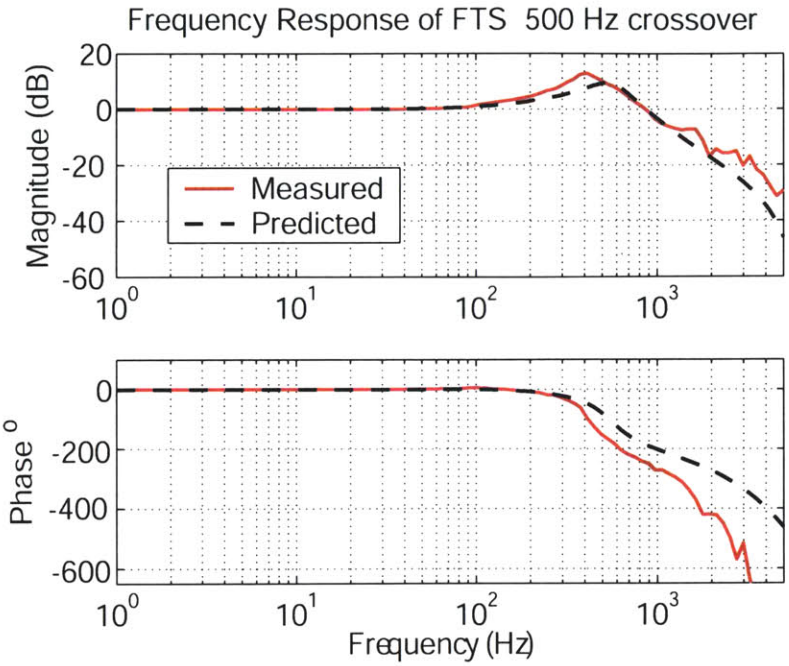


Figure 6-11: Experimental conventional fast tool servo frequency response. The dashed line indicates the predicted response.

The hydrostatic stage system consists of the non-moving portions of the fast tool servo including the bushings and support frame for the air bearing, the magnet track of the fast tool servo linear motor, the large adapter plate to mate the fast tool servo to the stage, the coil assembly for the Aerotech BLM-203 linear motor[2], and all of the cable and hoses for the operation of both the fast tool servo and hydrostatic slide. The mass of the stage was estimated by measuring the mass of the New-Way air bearing (7.7 kg), the mass of the magnet track (4.1 kg), the adapter plate (29.4 kg), Aerotech coil mass (0.9 kg), and adding them to the estimated mass of the moving stage components ($\approx$ 100 kg), for a total mass of approximately 142 kg. Since, the actual mass of the moving components of the stage was unknown, I rounded this total up such that

$$M_{stage} = 150 \text{ kg}. \tag{6.32}$$

[2]Aerotech Inc. of Pittsburg, PA kindly donated the two linear motors used in the DTM [3].

The Aerotech BLM-203 linear motor has a force constant

$$K_{fs} = 23.6\frac{\mathrm{N}}{\mathrm{A_{pk}}}. \tag{6.33}$$

The linear motors are driven by an Aerotech BA10-40 amplifier [3] with an amplifier gain of

$$K_a = 1\frac{\mathrm{A_{pk}}}{V} \tag{6.34}$$

for safety reasons the amplifier is current limited to 54% of the maximum current. In the case of the BA10-40 amplifier a change in the amplifier current limits results in a change in the amplifier gain, thus the amplifier gain becomes

$$K_a = 0.54\frac{\mathrm{A_{pk}}}{V}. \tag{6.35}$$

The BA10-40 amplifier has a bandwidth of 2kHz and a switching frequency of 20 kHz. Just as in the case of the Copley Controls amplifier used on the fast tool servo, we ignore the effect of the amplifier dynamics on the system response. The amplifier performs Hall-effect based motor commutation, eliminating one source of modeling error. Thus the complete model of the continuous time elements is

$$G_{ps}(s) \;=\; \frac{K_a K_{fs}}{M_{stage}s^2} = \frac{0.085}{s^2}. \tag{6.36}$$

Once again, I employ a *bilinear transform*, Eqn. (6.7), to determine the discrete time equivalent transfer function. With $T_s = 80\mu$s, the stage has the following discrete time transfer function

$$G_s(z) = 136x10^{-12}\frac{z^2+2z+1}{z^2-2z+1}. \tag{6.37}$$

As with the fast tool servo, we need to take into account that the DS1103 board D/A's

[3]Aerotech Inc. of Pittsburg, PA kindly donated the both the in-feed and cross-feed amplifiers used in the DTM [3].

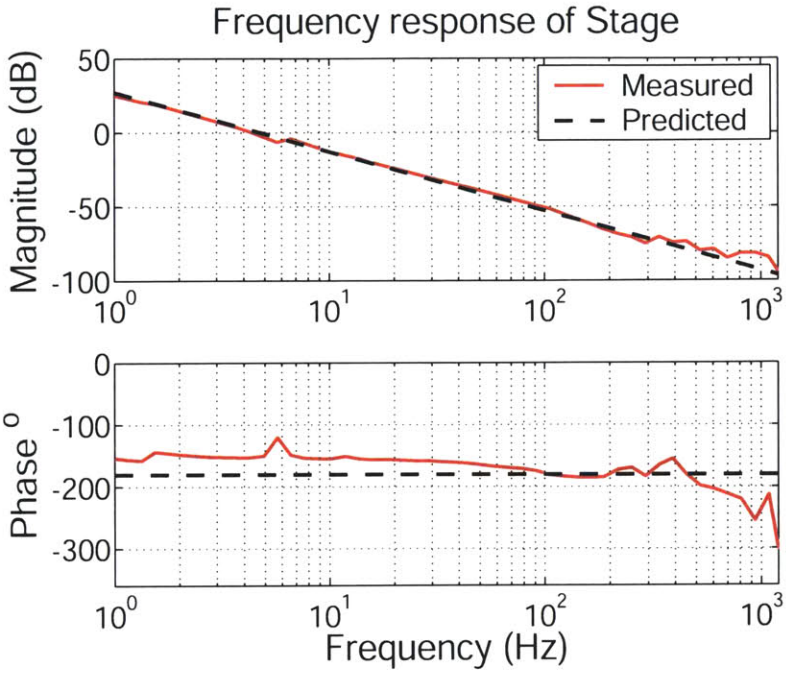


Figure 6-12: Experimental frequency response of the hydrostatic stage/reaction mass from commanded current to stage position. The dashed line indicates the predicted response.

have a gain. The stage position is measured using one of the incremental encoder interface channels on the DS1103 board. This interface is run on the main processor and unlike the serial interface for the fast tool serve does not have an appreciable delay. As before, I have chosen millimeters as my internal unit and must include (6.11). The complete stage model is

$$G_s(z) = 136x10^{-6}\frac{z^2+2z+1}{z^2-2z+1}. \tag{6.38}$$

Figure 6-12 shows the measured and predicted frequency response of the hydrostatic stage. As can be seen, the model and actual system match up quite well over the frequencies of interest. Not shown in this Figure is a resonant mode in the stage occurring at 1200 Hz. This resonance impacts the controller design and will be discussed in the next section.

Position Control

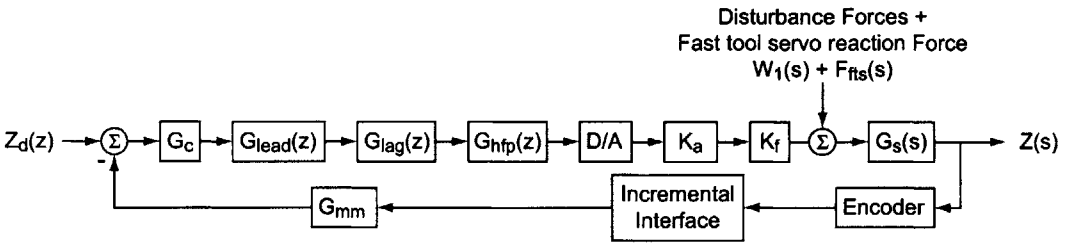


Figure 6-13: Block diagram of the hydrostatic stage/reaction mass conventional position control loop.

The conventional control loop used on the hydrostatic stage is nearly identical to that employed on the fast tool servo. Figure 6-13 shows a block diagram of the conventional position control loop. As we can see, the principal differences between the fast tool servo position control loop and the stage position control loop are the position feedback channel is implemented through the main DSP using the incremental endcoder interface which eliminates the two unit delay seen in the fast tool servo control loop and a high frequency pole is placed in the forward path to further attenuate the high frequency dynamics of the stage. The design goal for the stage compensation loop is a bandwidth of at least 100 Hz with a moderately damped step response. To achieve these goals, we have designed the loop transmission to crossover at 100 Hz. As in Section 6.1.1, the lead compensator is implemented as Eqn. (6.14) where Lz and Lp are as defined in Eqns. (6.15) and (6.16). For the stage lead, we have set $\alpha = 10$, the center frequency $f = 100$ Hz, and as always $T_s = 80\,\mu$s. Thus the transfer function for the stage lead is

$$G_{slead}(z) = \frac{z - 0.98423032}{z - 0.8530361}. \tag{6.39}$$

Similarly, the lag compensator is implemented as Eqn. (6.20) and where the zero is place at 10 Hz (one decade below the desired crossover frequency). This results in the lag having a transfer function of

$$G_{lags}(z) = \frac{2.0050265z - 1.9949735}{2(z-1)}. \tag{6.40}$$

The new element in the stage control loop is addition of a high frequency pole to attenuate the high frequency dynamics of the stage. This element was not included in the original control loop since when we set up the stage originally, we only measured the open loop response out to 1000 Hz (1 decade above the desired crossover frequency) and as a result missed a stage resonance at 1700 Hz. As it turns out, in our original machine testing this did not cause any problems. This resonance was brought to my attention by my fellow doctoral student Xiadong Lu in his work with ultra-fast servos. Xiadong elected to suppress this resonance using a high order notch filter [50]. Rather than use a notch filter, Richard Montesanti attenuated this resonance using a high frequency pole [61]. I have elected to replicate Rick's work and use a high frequency pole to cancel out the high frequency dynamics. The transfer function for the high frequency pole is

$$G_{hfp}(z) = \frac{1}{z - L_{hfp}} \tag{6.41}$$

where

$$L_{hfp} = e^{-\omega_{hfp} T_s} \tag{6.42}$$

and ω_{hfp} is the frequency of the high frequency pole in rad/s. I elected to place the high frequency pole at 500 Hz which resulted in a 13° drop in phase at my desired crossover frequency (100 Hz) and a 10 dB drop in the magnitude of the resonance at 1700 Hz. The transfer function of the high frequency pole as implemented is

$$G_{hfp} = \frac{1}{z - 0.77776778}. \tag{6.43}$$

As in the case of the fast tool servo controller to set the overall controller gain, I plotted the frequency response of the combined Lead, Lag, high frequency pole and Plant and selected $G_{cs} = 292$ such that

$$|G_{cs} G_{leads}(z) G_{lags}(z) G_s(z)| = 1. \tag{6.44}$$

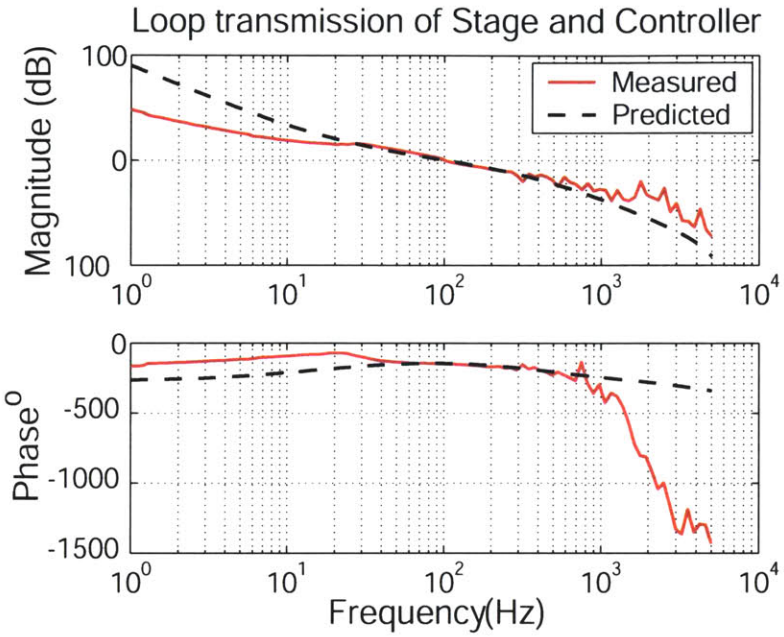


Figure 6-14: Experimental negative loop transmission response for the hydrostatic stage from position error to measure stage position. The solid line shows the measured response.

Combining all of the loop components results in the following transfer function for the negative loop transmission

$$L_s(z) = \frac{N_s(z)}{D_s(z)} \tag{6.45}$$

$$N_s(z) = 8.4195566E-4z^4 + 1.749889E-5z^3 - 1.6663459E-3z^2 \ldots$$
$$-1.7365746E-5z + 8.2452334E-3 \tag{6.46}$$

$$D_s(z) = 2z^5 - 9.2616076z^4 + 17.11175z^3 - 15.765606z^2 + 7.242391z - 1.3269248 \tag{6.47}$$

Figure 6-14 plots the predicted and measured negative loop transmission response of the stage control loop. The loop transmission was calculate from the position error to the stage position while the stage was operating in closed loop control. For frequencies below 11.2 Hz, the reference position command is a 0.02 mm peak to peak sinusoid. Above 11.2 Hz, the magnitude of the trajectory was selected such that

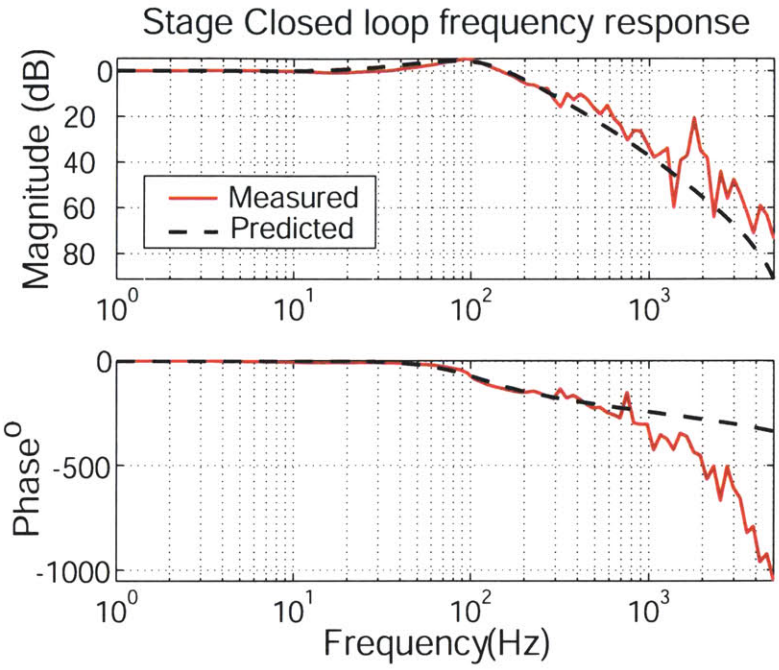


Figure 6-15: Experimental closed loop response for the hydrostatic stage. The solid line shows the measured response.

the maximum stage acceleration was 0.5 m/s^2 (0.05 g). The compensated system has crossover at 100 Hz with a phase margin of 35^o and a gain margin of 5. The discrepancy between the measured and predicted responses at low frequencies is likely caused by two factors. First, there is some issue in comparing the large response cause by a small error signal. Second, in the case of the hydrostatic stage, we have neglected the viscous damping from the oil in the hydrostatic bearing. Including the damping causes one of the integrators to move out along the negative real axis and thus reduce the low frequency gain of the loop. I have not attempted to model the effect of the damping because as we can see from Figure 6-14, the current model correctly predicts the frequency response around the crossover frequency of 100 Hz. Above 100 Hz, we see quite a bit of un-modeled dynamics in both the magnitude and phase plots. Since we have sufficiently attenuated the resonant peak at 1700 Hz, we did not attempt to correct the model at higher frequencies.

Figure 6-15 shows the closed loop frequency response of the hydrostatic stage.

The response was calculated from the reference position input to the measured stage position. For frequencies below 11.2 Hz, the reference position command is a 0.02 mm peak to peak sinusoid. Above 11.2 Hz, the magnitude of the trajectory was selected such that the maximum stage acceleration was 0.5 m/s^2 (0.05 g). The closed loop stage has a -3 dB bandwidth of 190 Hz. As we did for the fast tool servo, we have converted to the loop transmission transfer function $L_s(z)$ to state space form and then calculated the closed loop transfer function $P_s(z)$. This results in a closed loop plant model $P_s(z)$ with the following state matrices:

$$
\mathrm{A}=\begin{vmatrix}
4.631 & -8.556 & 7.883 & -3.621 & 0.6635 & -4.21\mathrm{E}{-4} & -8.746\mathrm{E}{-6} & 8.332\mathrm{E}{-4} & 8.683\mathrm{E}{-6} & 43123\mathrm{E}{-4} \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 4.63 & -8.556 & 7.884 & -3.621 & 0.6631 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0
\end{vmatrix}
$$

$$
\mathrm{B}=\begin{vmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{vmatrix},\ \mathrm{C}=\begin{vmatrix} 4.21\mathrm{E}{-4} & 8.749\mathrm{E}{-6} & -8.332\mathrm{E}{-4} & -8.683\mathrm{E}{-6} & 4.123\mathrm{E}{-4} & 0 & 0 & 0 & 0 & 0 \end{vmatrix},
$$

$\mathrm{D} = 0.$

(6.48)

6.2 Adaptive Feedforward Cancellation Applied to the Linear Fast Tool Servo

In section 4.2.3, we detailed a method for applying adaptive feedforward control to a generic system. In this section, we will be applying that method to the linear fast tool servo. We implemented and conducted experiments using the following four configurations:

1. A single AFC resonator canceling a single harmonic sinusoid at 20 Hz.
2. A single AFC resonator canceling a single harmonic sinusoid at 50 Hz.
3. An eight harmonic AFC controller applied to an eight harmonic trajectory with a fundamental frequency at 12 Hz.
4. A six harmonic AFC controller applied to a six harmonic trajectory with a fundamental frequency of 23 Hz.

Due to the large number of data points required to measure the frequency response around the resonators, we only measured the frequency response of the final AFC implementation. Note: The measurement of this frequency response took on the order of one hour and during the test the pump for the coolant oil seized, thus preventing any further high acceleration testing.

In this section when we refer to the negative of the loop transmission, we are referring to the following transfer function

$$L_{afc}(z) = C_N(z)P(z) \tag{6.49}$$

where $P(z)$ is the transfer function of the fast tool servo under conventional control determined from Eqn. 6.30 and where

$$C_N(z) = \sum_{i=1}^{N} g_i \frac{z^2 \cos\phi_i - z\cos(\omega_i T + \phi_i)}{z^2 - 2\cos(\omega_i T)z + 1}. \tag{6.50}$$

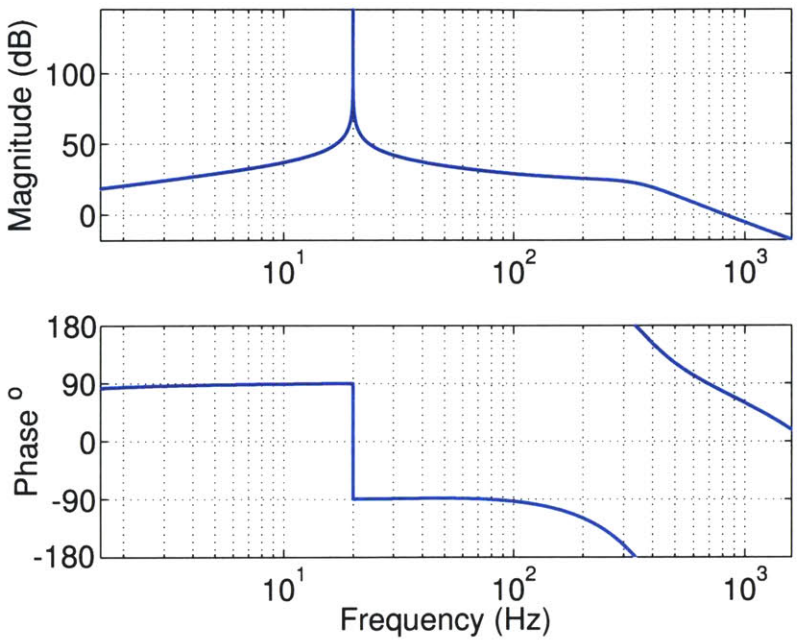


Figure 6-16: The calculated negative of the AFC loop transmission $-L_{afc}(z)$ for the linear fast tool servo with a single resonator at 20 Hz and $g_1 = 1$.

In general when we refer the to closed loop AFC transfer function, we are referring to

$$\frac{Z(z)}{Z_d(z)} = \frac{P(z)(1 + C_N(z))}{1 + P(z)C_N(z)} \tag{6.51}$$

which is derived by setting the feedforward filter in Figure 6-1 equal to 1.

6.2.1 Single 20 Hz Resonator

The first AFC configuration that we implemented was a single resonator at 20 Hz. To tune a single resonator system, we need to select only two parameters g_1 and ϕ_1. This means that we need to modify step 3 of our procedure in section 4.2.3 to evaluate the magnitude of the loop at each frequency the phase passes through $\pm 180^o$ instead of specifically evaluating the local minima between resonators. Following the procedure in section 4.2.3, we set the phase advance parameter ϕ equal to the measured plant

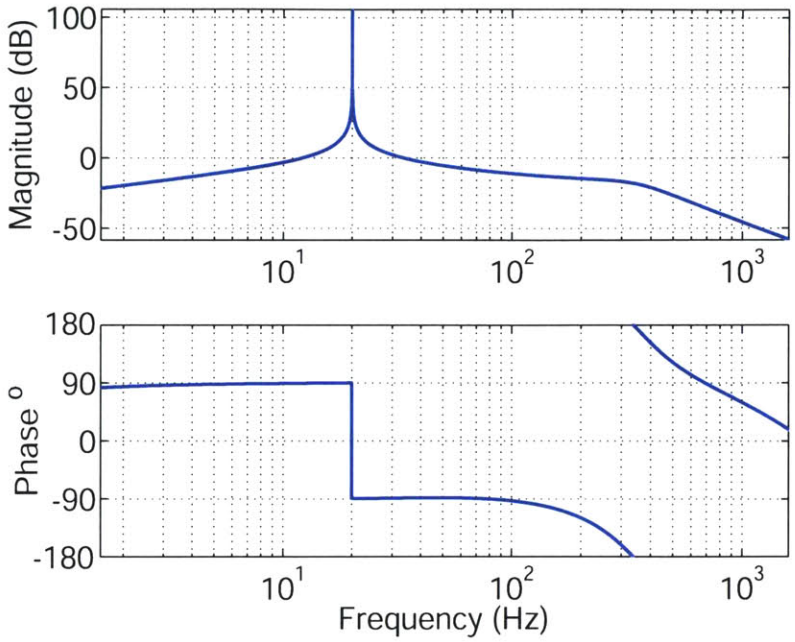


Figure 6-17: The calculated negative of the AFC loop transmission $-L_{afc}(z)$ for the linear fast tool servo with a single resonator at 20 Hz and $g_1 = 0.01$.

phase

$$\phi = \angle P(20\pi \mathrm{j}) = 0.1^o = 0.003 \text{ rad.} \tag{6.52}$$

In this case since the plant phase is essentially zero, the phase advance parameter does little to enhance stability. Figure 6-16 shows the calculated negative of the AFC loop transmission $-L_{afc}(z)$ for the linear fast tool servo with a single resonator at 20 Hz and $g_1 = 1$. As can be seen from the plot, this is clearly an unstable tuning since the phase wraps through -180°with a magnitude of 20 dB at 348 Hz. In general, we tune systems with AFC control to have a gain margin of at least 10 (20 dB). This means that we need to reduce the gain by a factor of 100. Figure 6-17 shows the calculated negative of the AFC loop transmission $-L_{afc}(z)$ for the linear fast tool servo with a single resonator at 20 Hz and $g_1 = 0.01$. As we can see, the system now has the desired gain margin of 10 at 348 Hz.

Figure 6-18 shows the calculated closed loop frequency response for the fast tool

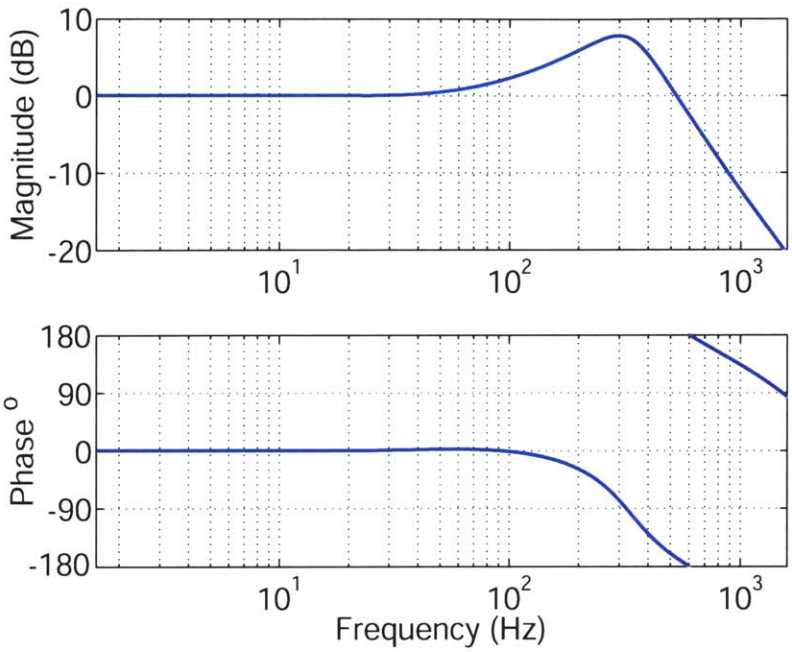


Figure 6-18: The calculated closed loop frequency response for the linear fast tool servo with a single resonator at 20 Hz and $g_1 = 0.01$.

servo with a single AFC resonator at 20 Hz and $g_1 = 0.01$. This plot does not look different from the system without AFC (Figure 6-9) since the calculated magnitude of the system without AFC is 0.047 dB at 20 Hz, while the AFC compensated system has a magnitude of 0 dB at 20 Hz. While it is not clear in Figure 6-18 that AFC control improves trajectory following, we will see in the next chapter that AFC control significantly reduces the following error.

6.2.2 Single 50 Hz Resonator

The second AFC controller we employed was a single resonator at 50 Hz. We used the same design procedure as the previous section to choose g_1 and ϕ_1. Figure 6-19 shows the calculated negative of the AFC loop transmission for the fast tool servo with a single resonator at 50 Hz, $g_1 = 0.01$ (the gain selected for a single 20 Hz resonator),

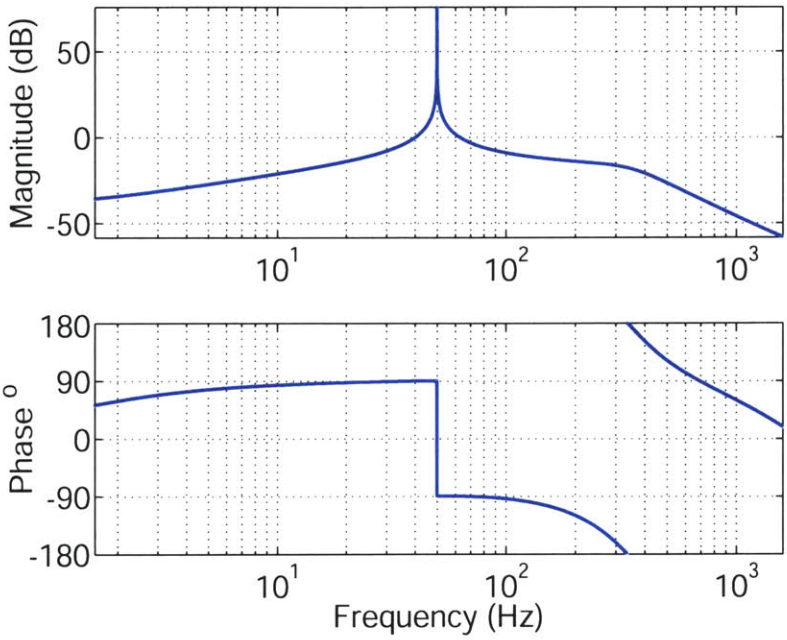


Figure 6-19: The calculated closed loop frequency response for the linear fast tool servo with a single AFC resonator at 50 Hz and $g_1 = 0.01$.

and

$$\phi = \angle P(100\pi \mathrm{j}) = 0.03^o = 0.0005 \text{ rad.} \tag{6.53}$$

Once again, the phase advance does not appreciably enhance the stability of the system. As we can see from the plot, the parameters as selected result in loop with a gain margin of 7.9 at 348 Hz. While the gain margin is a little less than the previous system, it is sufficient to ensure that the closed loop system will remain stable.

Figure 6-20 shows the calculated frequency response of the fast tool servo with a single resonator at 50 Hz and $g_1 = 0.01$. In the case of the 50 Hz resonator, we begin to see the functionality of AFC compensation. Without AFC compensation, the conventionally controlled plant has a magnitude of 0.8 dB (1.096) at 50 Hz which results in almost a 10% error following a 50 Hz sinusoidal trajectory. In the AFC compensated system, the closed loop magnitude crosses through 0 dB (1) at 50 Hz. Thus the system would perfectly track a sinusoid of 50 Hz. It should be noted that

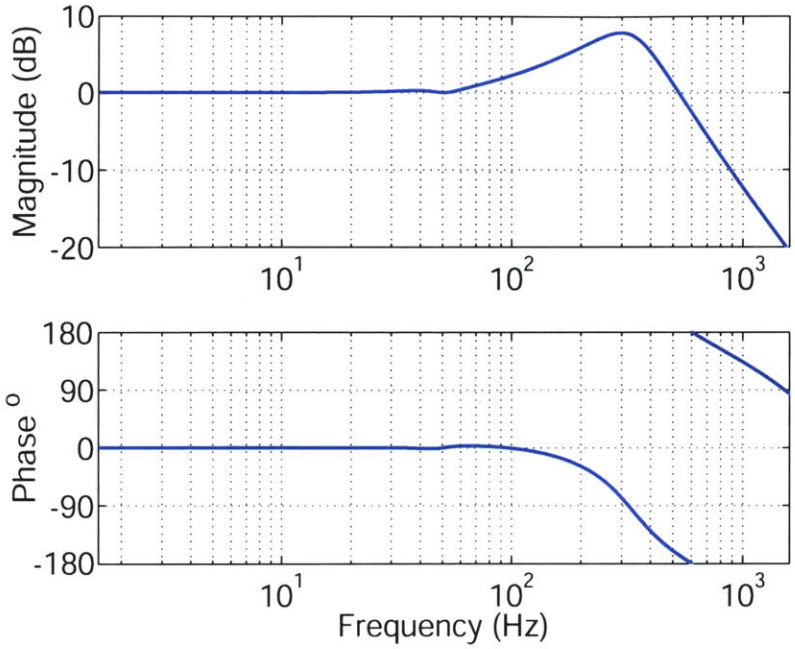


Figure 6-20: The calculated closed loop frequency response for the linear fast tool servo with a single resonator at 50 Hz and $g_1 = 0.01$.

the closed loop magnitude near 50 Hz is non-zero so any variations in the frequency of the sinusoid would result in small following errors. As mentioned in chapter 4, the higher the AFC gain the better the closed loop response near the resonator frequency is.

6.2.3 Eight Harmonic AFC Controller with $\omega_1 = 12$ Hz

The third AFC controller that we implemented was a network of 8 resonators with $\omega_1 = 75$ rad/s $= 11.64$ Hz ≈ 12 Hz. We had a number of issues designing and implementing this controller.

First, at low frequencies both the closed-loop plant model and the measured plant have a positive phase shift. The phase of the measured plant becomes negative for frequencies about 50 Hz, while the model becomes negative above 60 Hz. As discussed in chapter 4, in continuous time $0 < \phi < \pi/2$ locates the AFC zero on the negative real axis while $0 > \phi > -\pi/2$ places the zero on the positive real axis. Similarly, in

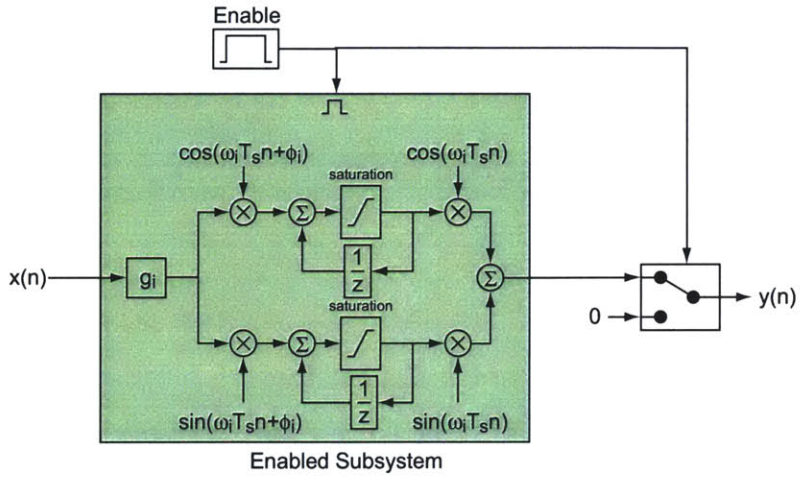


Figure 6-21: AFC resonator implemented as an enabled subsystem in Simulink.

discrete time $0 < \phi < \pi/2$ results in the zero being on the real axis inside the unit circle and $0 > \phi > -\pi/2$ results in a zero outside the unit circle on the real axis. When an array of resonators in which some of the zeros are inside the unit circle and some are outside the unit circle are summed, the result is a system where there are complex zero pairs both inside and outside the unit circle (alternately stated there are zeros in both the right and left half planes). As it happens, each complex zero pair is located between two resonant peaks. If the complex zero pair is located in the right half plane or outside the unit circle, the phase of the system drops 180°between resonant peaks. This is the condition studied in detail in chapter 4. If the complex zero pair is located inside the unit circle, the system phase rises 180^o. While this result does not change our tuning rules, it did take us some time to confirm that the odd (too me at least) looking frequency response plots we were generating in Matlab were correct.

Second, I initially implemented each AFC resonator as an enabled subsystem in Simulink. To turn on AFC control, I would switch to ControlDesk where I would first enable the amplifiers, then enable the conventional control loops, and lastly enable the AFC resonators. As it turns out when more than six resonators are implemented as enabled subsystems, the initialization process when the subsystems are turned

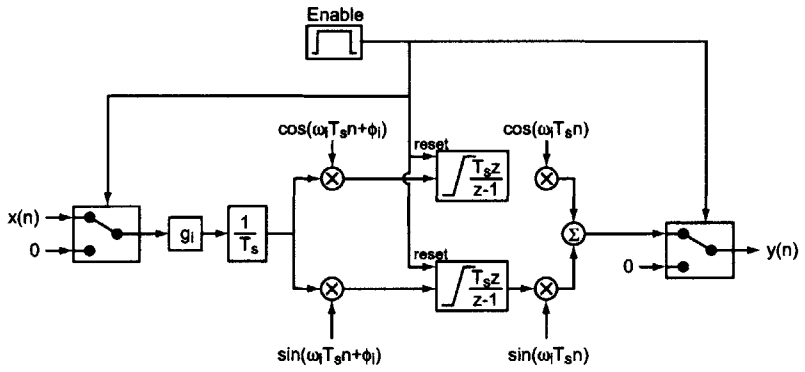


Figure 6-22: An alternate implementation of an AFC resonator in Simulink.

on cause the DSP turnaround time to overrun, crashing the entire system. Figure 6-21 shows an AFC resonator implemented as an enabled subsystem. The internal states are held when the system is disabled and reset each time the system is enabled. Since the states are held when the system is disabled, we need to include a switch which ensures that the signal input into the conventional control loop is zero when the AFC controller is disabled. The output and internal state of the integrator is limited by a saturation block. Figure 6-22 shows an alternate implementation of the AFC resonator in Simulink, where the delay line integrator is replaced by a saturation limited backward integrator from the Simulink library. Since the Simulink integrator includes T_s, we needed to add a block where we divided by T_s if we wanted to maintain continuity in the values of g_i. In this implementation, the AFC controller is always running but is just switched in and out of the control loop. The integrator states are reset each time the resonator is switched into the control loop. This implementation allowed us to run up to 12 total resonators distributed between the fast tool servo controller and the base acceleration feedback.

My final issue with this controller is that I did not follow my own tuning rules and attempted to implement an unstable controller. Specifically, I implemented this controller in steps. I started with one resonator then tested the controller, added a second resonator and tested, etc. When I added the seventh AFC resonator the system failed. Initially, this failure was the result of the time step over-run problem mentioned

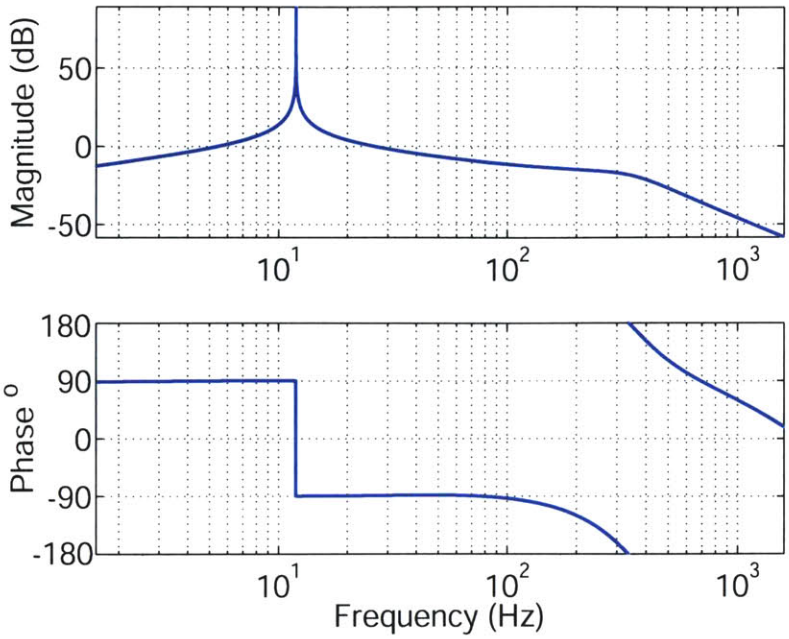


Figure 6-23: The calculated negative loop transmission $-L(z)$ with a single resonator at 12 Hz and $g_1 = 0.01$.

above yet even after altering the resonator structure the controller still failed. The issue here was that I had chosen my resonator gain $g_i = 0.01$ based on my work with a single resonator. With a single resonator, this resulted in a controller with nearly 20 dB of gain margin (Figure 6-23). With six resonators, the controller gain margin has been reduced to 2.5 dB (Figure 6-24). With seven resonators, the controller has a calculated gain margin of 1 dB (Figure 6-25) but is actually unstable when implemented. I have included this here because my experience carelessly designing an unstable controller gave me some insight when analyzing my results in the next chapter.

Figure 6-26 shows the calculated negative loop transmission of the fast tool servo system with 8 resonators with a fundamental frequency of 12 Hz and $g_i = 0.001$. As with all the controllers in this subsection,

$$\phi_i = \angle P(w_i \mathrm{j}). \tag{6.54}$$

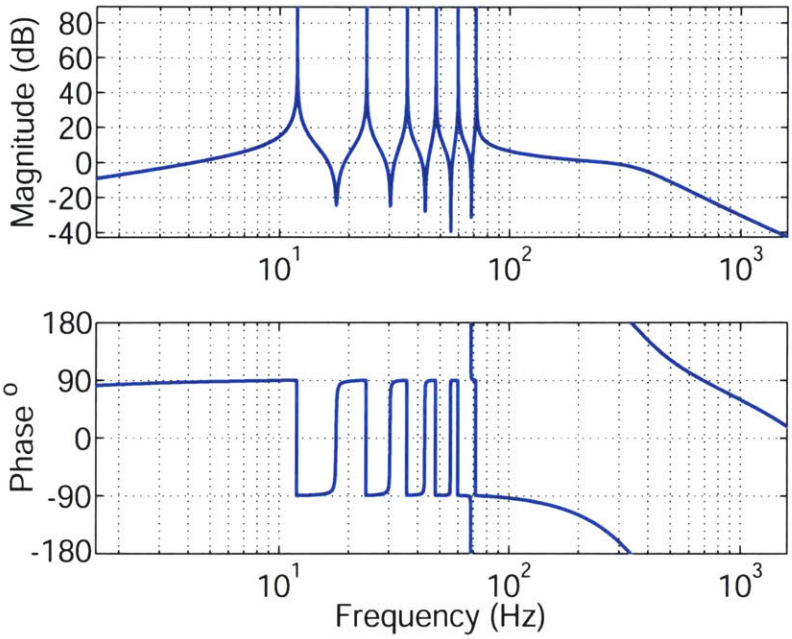


Figure 6-24: The calculated negative loop transmission $-L(z)$ with a six resonators with a fundamental frequency of 12 Hz and $g_i = 0.01$.

When expressed in vector form ϕ has the following values

$$\phi = [0.0016,\ 0.008,\ 0.0136,\ 0.0132,\ 0.0045,\ -0.0128,\ -0.0384,\ -0.0712] \quad (6.55)$$

where ϕ has been determined from the closed loop plant model. With all of the resonator gains $g_i = 0.001$, the system has a gain margin of 20 dB. In Figure 6-26 we also take note of the how having the complex zero pair inside or outside the unit circle shapes the plant phase between resonant peaks. As stated earlier, the phase rises 180^o if the zeros are inside the unit circle and drops 180^o if the zeros are outside the unit circle. Figure 6-27 shows the pole-zero plot for this 8 resonator AFC controller with ϕ_i from 6.55 and $g_i = 0.001$. We can see from the pole-zero plot that the four lowest frequency zero pairs are located inside the unit circle while the three highest frequency pole pairs are located outside the unit circle. This corresponds with the phase wraps in Figure 6-26 where the phase rises 180^o between resonators until we reach the fifth peak at which point the phase drops between peaks. In theory if all of

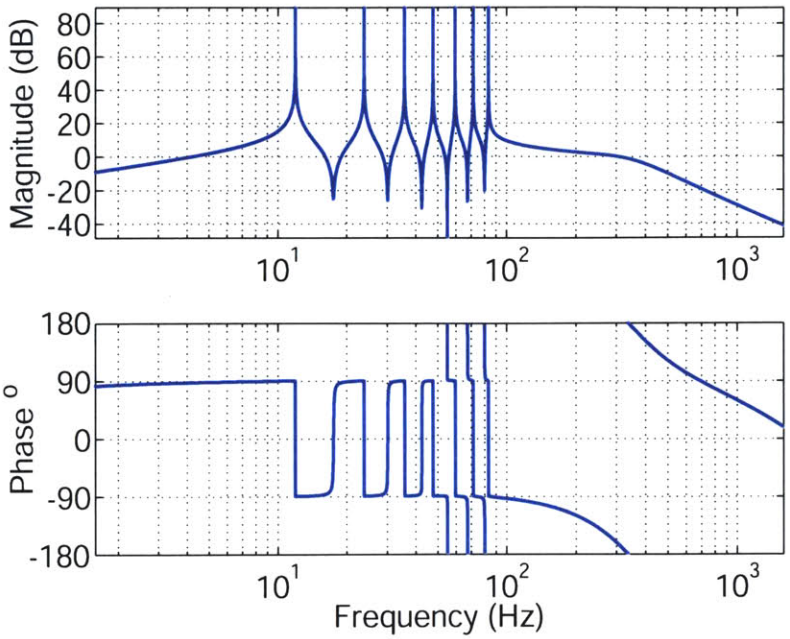


Figure 6-25: The calculated negative loop transmission $-L(z)$ with a seven resonators with a fundamental frequency of 12 Hz and $g_i = 0.01$.

the zero pairs where inside the unit circle, we should be able to significantly increase all of the resonator gains since the phase never passes through $\pm 180^o$. Practically speaking this is unlikely to be true because this would require the plant phase to be positive at all of the resonant peaks and that the plant phase also not wrap through $\pm 180^0$. Taking our system as an example, what limits the resonator gains is not the phase wraps between resonators but the the fact that the loop transmission wraps through -180^o at 348 Hz due to the -90^o of phase from the plant and the -90^o from AFC compensation.

In general the amplitudes of the first several components of a multiple harmonic trajectory are significantly larger than the higher harmonics, thus it is generally desirable to assign as much gain to the lower harmonics as possible. Thus I choose to implement the eight resonator AFC controller with the following gains:

$$g = [0.01,\ 0.01,\ 0.01,\ 0.001,\ 0.001,\ 0.001,\ 0.001,\ 0.001]. \tag{6.56}$$

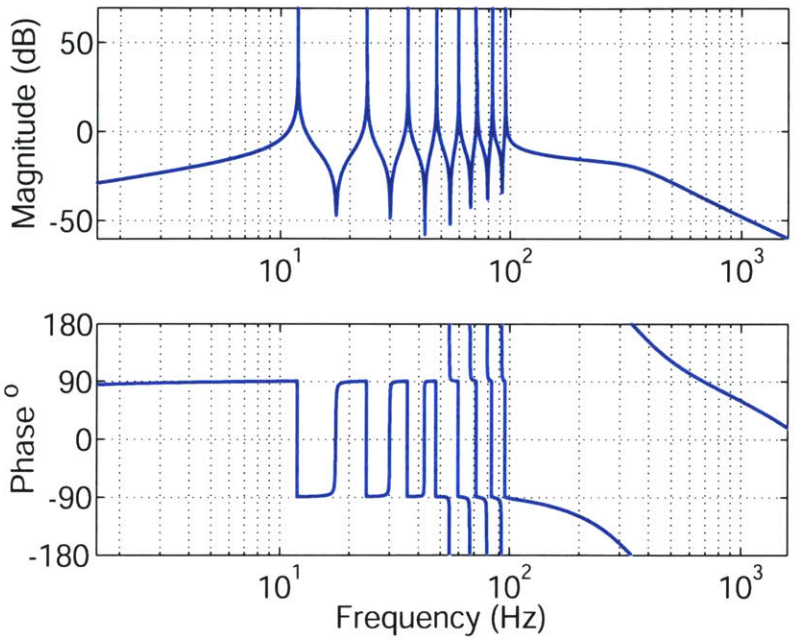


Figure 6-26: The calculated negative loop transmission $-L(z)$ with a eight resonators with a fundamental frequency of 12 Hz and $g_i = 0.001$.

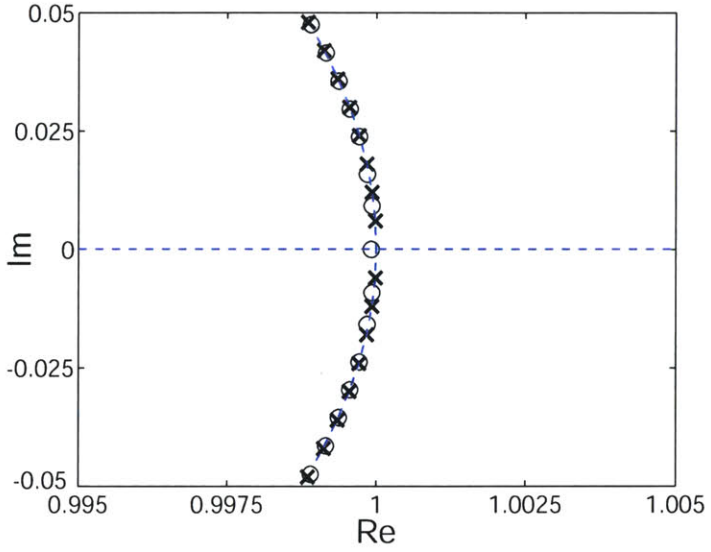


Figure 6-27: Pole-zero plot of an AFC controller $C_N(z)$ with $g_i = 0.001$ and ϕ_i from 6.55.

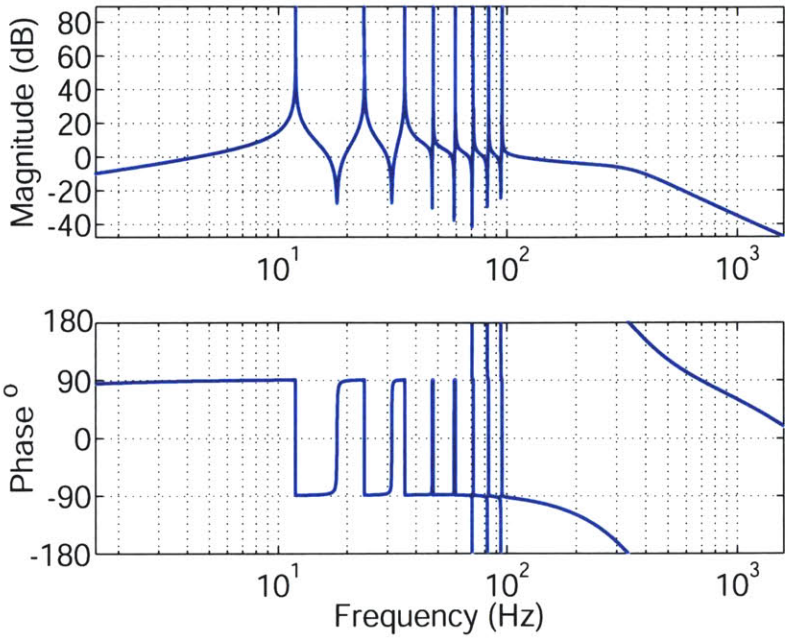


Figure 6-28: The calculated negative loop transmission $-L(z)$ with eight resonators with a fundamental frequency of 12 Hz and g_i from 6.56.

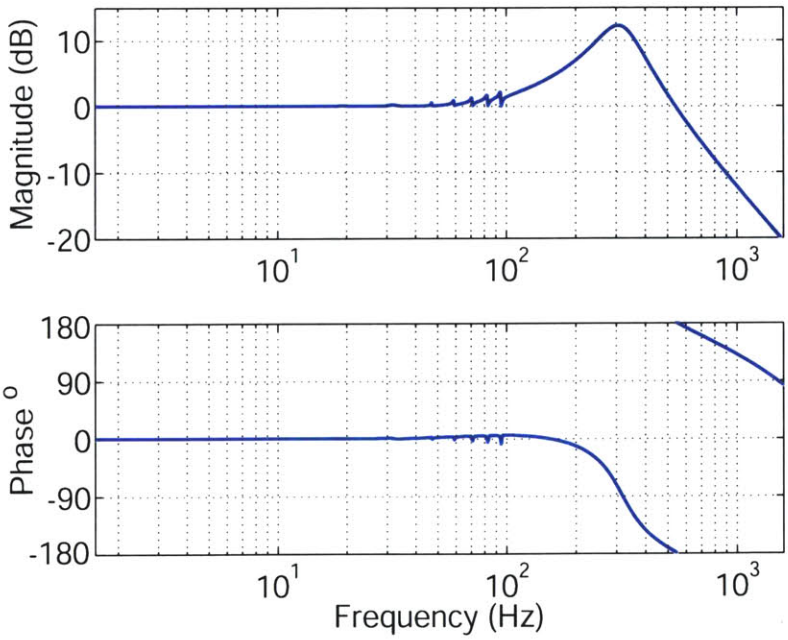


Figure 6-29: Calculated closed loop frequency response of the fast tool servo with an eight resonators with a fundamental frequency of 12 Hz and g_i from 6.56.

Figure 6-28 shows the calculated loop transmission for this tuning. As we can see, this calculated system is stable with a gain margin of 7.5 dB at 348 Hz. This controller was stable when implemented. Figure 6-29 shows the calculated closed loop frequency response of the fast tool servo with the controller tune with ϕ from (6.55) and g_i from (6.56). As we expected, the magnitude of the transfer function is zero at each resonator harmonic. On the negative side, this controller tuning has significantly increased the magnitude of closed loop plant between 200-400 Hz. This magnitude increase explains the undesirable dynamics we see in chapter 7.

6.2.4 Six Harmonic AFC Controller with $\omega_1 = 23$ Hz

The final AFC configuration is one with six resonators with a fundamental frequency of 23 Hz. This configuration is a result of several factors. First by employing only six resonators, we were able to employ AFC control on the fast tool servo and base acceleration loops. Second, as will be discussed in chapter 7, the accelerometer has a large 60 Hz noise component. A fundamental frequency of 23 Hz was chosen to avoid placing a resonator at 60 Hz. Lastly, based upon our results with the eight resonator system, the resonator gains were selected to ensure that we did not significantly increase the magnitude of the plant between 200-400 Hz. Figure 6-30 show the calculated negative loop transmission with fundamental frequency of 23 Hz and with $g_i = 0.001$ and $\phi_i = \angle P(\mathrm{j}\omega_i)$. When expressed as a vector ϕ for this system has the value:

$$\phi = [0.0083,\ 0.0154,\ -0.0094,\ -0.0696\ ,-0.1602,\ -0.2757] \tag{6.57}$$

where ϕ has been determined from the closed loop plant model. This system has a gain margin of 22 dB at 348 Hz. Figure 6-31 show the measured and calculated response of the fast tool servo with this controller. Given the complexity or the numerical model as well as the overall control loop, the two responses match quite well.

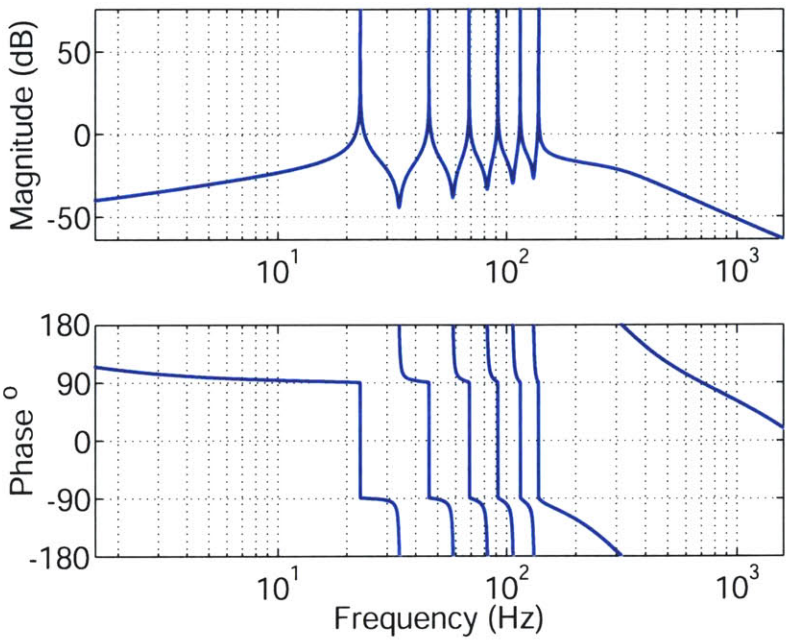


Figure 6-30: Calculated negative loop transmission $-L(z)$ with six resonators with a fundamental frequency of 23 Hz and $g_i = 0.001$ and ϕ_i from 6.57.

6.3 Hydrostatic Stage Base Acceleration Feedback

In this section, we will detail the design and implementation of a base acceleration feedback control loop for the hydrostatic stage. The goal of this compensation loop is to minimize the base acceleration and allow the hydrostatic stage to absorb all of the fast tool servo reaction forces. Figure 6-2 shows the complete control system for the hydrostatic stage. In this section, we will first detail the development of the acceleration compensation loop including the selection of $G_a(z)$ (a general control function) and $G_{hp}(z)$ (a high pass filter). Next, we detail the actual loop implementations including the selection of the gain and phase of the AFC controller $C_N(z)$.

6.3.1 Base Acceleration Feedback Loop

Figure 6-32 shows the block diagram of the hydrostatic stage position control loop with both conventional position control and base acceleration feedback loop. The design and implementation of the conventional position control loop is documented

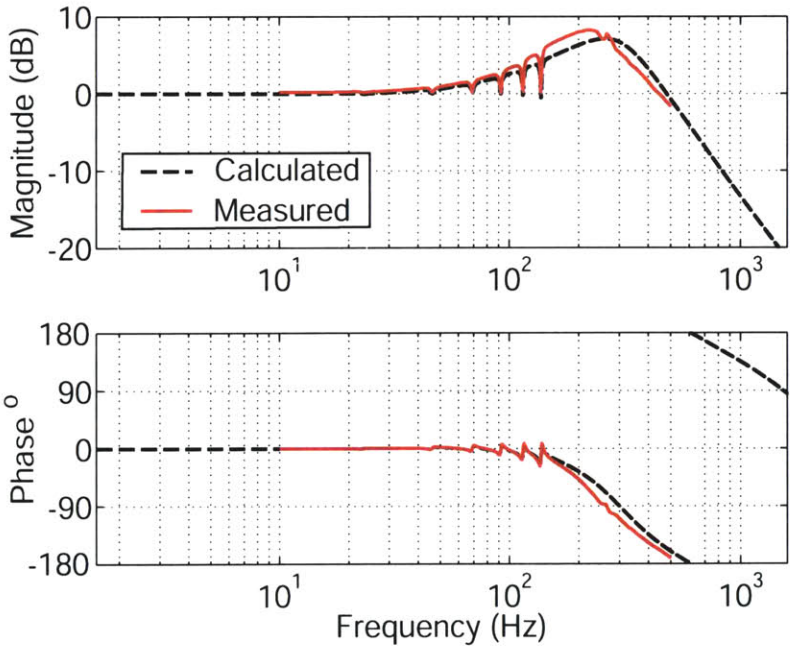


Figure 6-31: Calculated and measured closed loop frequency response of the fast tool servo with six resonators with a fundamental frequency of 23 Hz and $g_i = 0.001$ and ϕ from 6.57. The dashed line indicated the calculated response

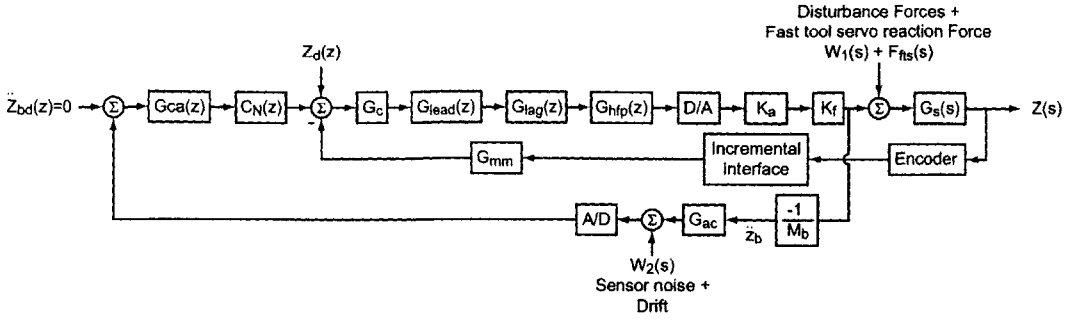


Figure 6-32: Block diagram of the hydrostatic stage/reaction mass position control loop with both conventional position control and acceleration feedback.

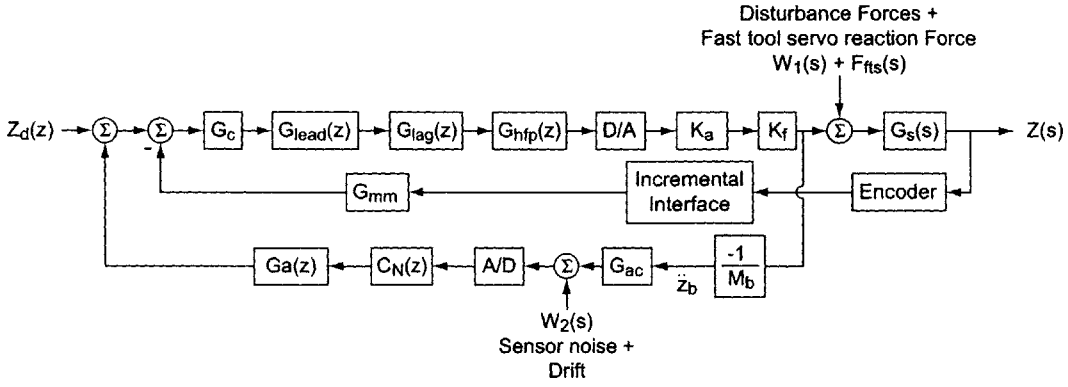


Figure 6-33: Block diagram of the hydrostatic stage/reaction mass position control loop with both conventional position control and acceleration feedback after removing the zero base acceleration reference.

in 6.1.2. Here we will detail the design and implementation of the base acceleration feedback loop. To start with we should note that with the introduction of base acceleration feedback the system has become a multiple input/multiple output (MIMO) system with two inputs (the stage trajectory z_d and the desired base acceleration $\ddot{z}_{db}$) and two outputs (stage position z and base acceleration $\ddot{z}_b$). In general, controlling MIMO systems is significantly more challenging than controlling single input/single output (SISO) systems so what we would like to do is manipulate our control model so we can apply the conventional SISO loop shaping tools to the acceleration feedback loop. We begin our manipulation by noting that our desired acceleration input $\ddot{z}_{bd} = 0$. This allows us to remove the first summation block from the block diagram

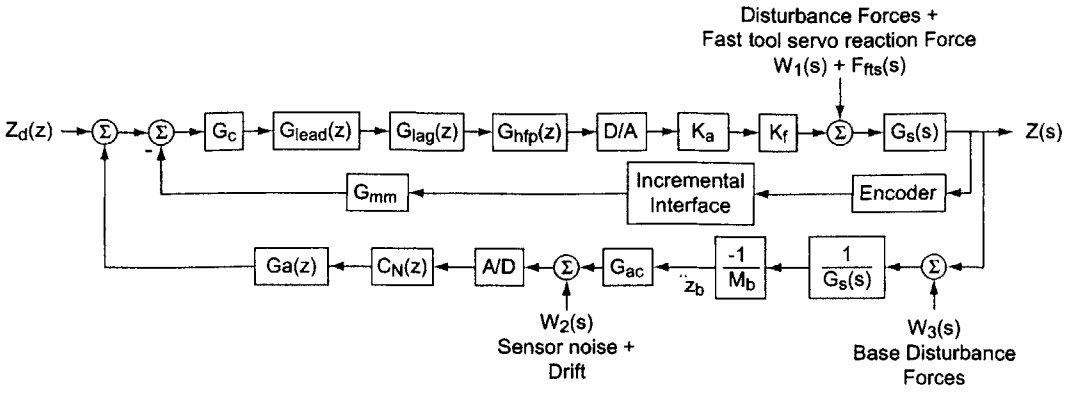


Figure 6-34: Block diagram of the hydrostatic stage/reaction mass position control loop with both conventional position control and acceleration feedback after having manipulated the acceleration feedback across the stage transfer function.

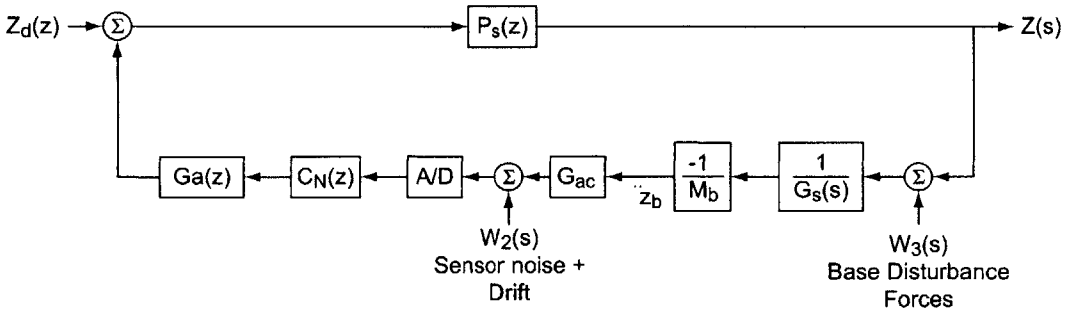


Figure 6-35: Block diagram of the hydrostatic stage/reaction mass position control loop with both conventional position control and acceleration feedback after substituting $P_s(z)$ for the conventional inner loop position control.

for the stage controller. Figure 6-33 shows the controller block diagram with the elimination of the acceleration input and some minor block placement changes. The loop now looks much more like a conventional controller with both an inner loop and and an outer loop. From a block diagram algebra standpoint, there is no issue with moving the acceleration feedback signal across the plant transfer function such that $\ddot{z}_b$ becomes a function of z. Figure 6-34 illustrates manipulation. At this point, it is clear that the inner loop solves to $P_s(z)$ which is the closed loop transfer function of the stage under conventional control. Figure 6-35 shows the block diagram after we replace $P_s(z)$ for the conventional inner loop position control.

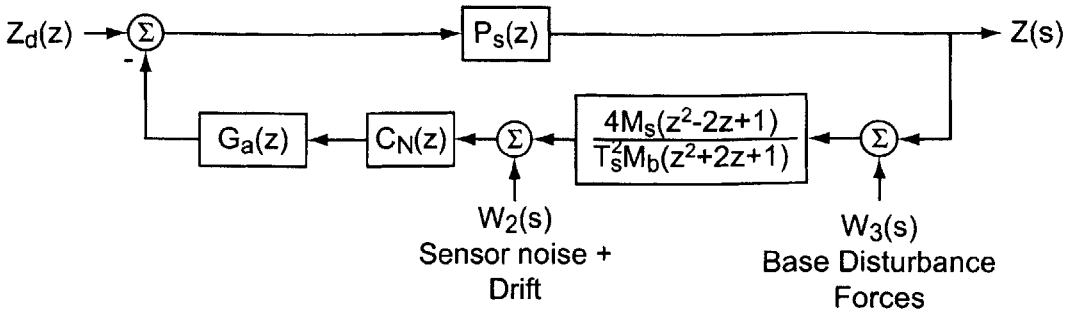


Figure 6-36: Block diagram of the hydrostatic stage/reaction mass position control loop with both conventional position control and acceleration feedback after substituting for $G_s(s)$ and converting to discrete time.

At this point, we have a loop which has a mix of continuous and discrete elements so it make sense to convert all of the transfer functions to discrete time. From 6.1.2, we know that

$$G_s(s) = \frac{1}{M_s s^2}. \tag{6.58}$$

Applying the *bilinear transform* to this transfer function results in the following discrete equivalent

$$G_s(z) = \frac{T_s^2(z^2 + 2z + 1)}{4M_s(z^2 - 2z + 1)}. \tag{6.59}$$

Figure 6-36 shows the block diagram after substituting for $G_s(s)$ and converting to discrete time. Note: At this point I have dropped the gains associated with the accelerometer and the D/A converter. I will bring them back when we discuss the actual implementation.

The loop transmission for Figure 6-36 is

$$L_a(z) = \frac{P_s(z)C_N(z)G_a(z)}{G_s(z)}. \tag{6.60}$$

As a starting point for determining the correct tuning rules and form for $G_a(z)$, I set $G_a(z) = 1$ and plotted the calculated frequency response for a $C_N(z)$ consisting

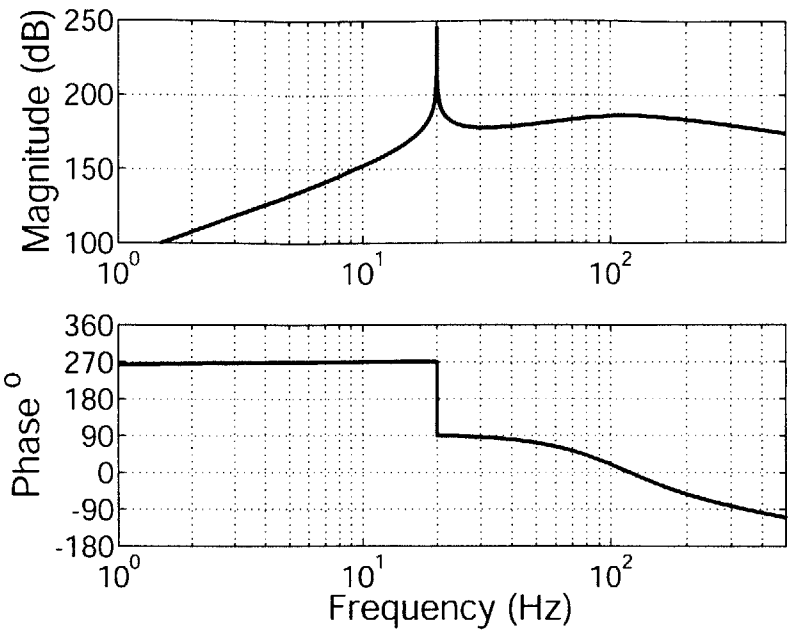


Figure 6-37: Calculated stage acceleration negative loop transmission $L_a(z)$ (6.60) with a single AFC resonator at 20 Hz.

of a single resonator at 20 Hz ($\phi_1 = \angle P_s(40\pi \mathrm{j})$). Figure 6-37 shows the calculated frequency response. As we can see with $G_a(z) = 1$, the system is unstable because the phase wraps through 180° at the frequency of the resonator (note: there is no possible gain we can select that results in a stable system because the gain at 20 Hz is always infinite). If we look at the system transfer function in more detail, we see that moving the acceleration feedback signal through the stage transfer function $G_s(s)$ has added a pair of zeros at the origin $1/G_s(s) = M_s s^2$. Since $\angle P_s(40\pi \mathrm{j}) \approx 0$, the zero of the AFC resonator is placed very close to the origin. Thus at low frequencies, the loop transmission has a phase of +270°. At 20 Hz, the resonator poles contribute a phase drop of 180° causing the loop phase to drop through +180°. In order to make this loop stable, we need to drop the low frequency phase of the plant by more than 90° and preferably by 180° thus ensuring 90° of phase margin both before and after the resonator.

Initially, to drop the loop phase by 180°, we made two modifications to the loop transfer function. First, as noted earlier, the AFC resonator contribute +90° because

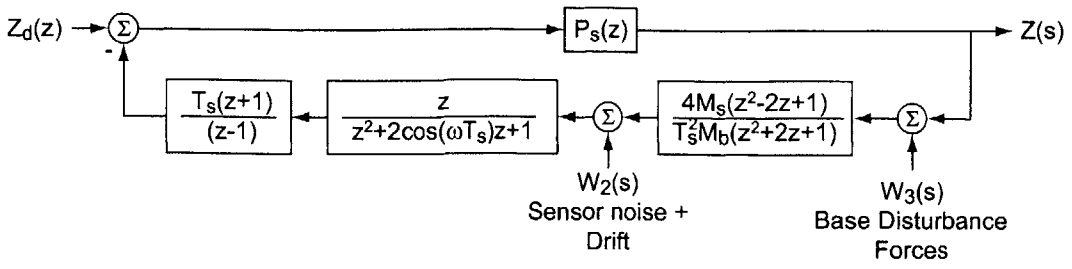


Figure 6-38: Block diagram of the hydrostatic stage/reaction mass position control loop with both conventional position control and acceleration feedback with $C_N(z)$ from (6.61) and $G_a(z)$ from (6.62).

the resonator zero is place below 20 Hz. To eliminate this phase shift we can simply eliminate this the low frequency zero. Thus the transfer function of our AFC resonator becomes

$$C(z) = \frac{z}{z^2 - 2\cos(\omega T_s)z + 1}. \tag{6.61}$$

The second modification was to introduce an addition integrator in the control loop. To do this we set

$$G_a(z) = \frac{T_s(z+1)}{2(z-1)}. \tag{6.62}$$

thus removing another 90^o of phase from the loop. Figure 6-38 shows the block diagram for this system. Figure 6-39 shows the calculated negative loop transmission for this system. As we can see from Figure 6-39, the loop phase now starts at $+90^o$ and drops to -90^o at the resonant frequency. Thus, with the proper selection of the loop gain, this control strategy will result in a stable closed loop system.

Practically there are two problems with this controller implementation. First, removing the phase compensating zero from the AFC controller means that AFC control can only be applied to loops where the phase is between 0^o and $+180^o$. For our system, this means that this form of base acceleration feedback control can only be applied below 120 Hz. Second, this controller does not work in practice because the accelerometer measurement has a D.C. offset, drift, and a significant noise com-

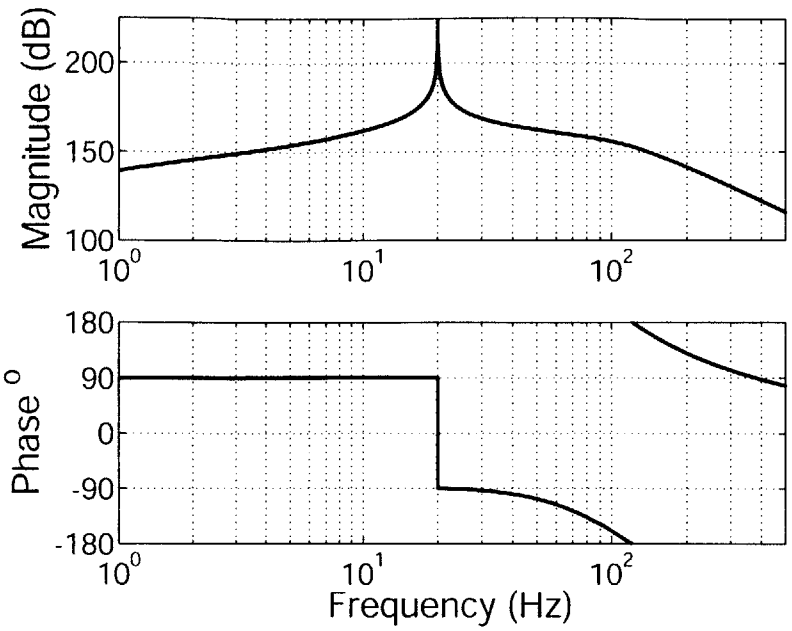


Figure 6-39: Calculated stage acceleration negative loop transmission with $C_N(z)$ from (6.61) and $G_a(z)$ from (6.62).

ponent. When this controller is implemented the acceleration feedback causes a large stage translation in an attempt to cancel the offset component of the accelerometer measurement. One solution to this problem is to introduce a high pass filter on the accelerometer measurement. We used a simple second-order high pass filter with a pair of zeros at $z = 1$ and a well damped ($\zeta = 0.7$) complex pole pair at 1 Hz. When implemented at 12.5 kHz, the filter had the following transfer function:

$$G_{hp}(z) = \frac{z^2 - 2z + 1}{z^2 - 1.9992963z + 0.99929653}. \tag{6.63}$$

Figure 6-40 shows the block diagram with the inclusion of the high pass filter while Figure 6-41 shows the calculated negative loop transmission frequency response. As described later, this control structure results in a stable response.

As mentioned earlier, the controller in Figure 6-40 results in a stable loop but can only be applied to system with a phase between 0^o and $+180°$. A more general control structure requires that the AFC controller incorporates the phase compensating zero.

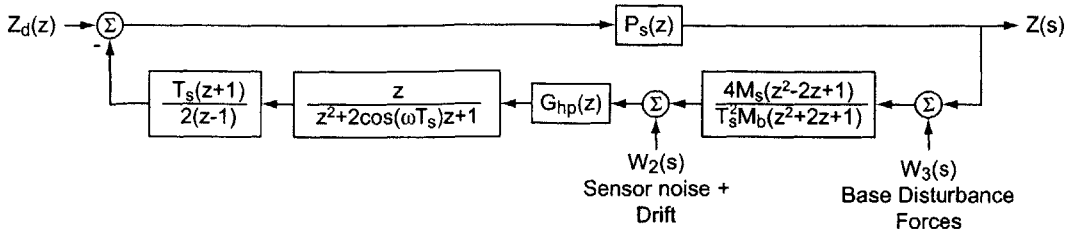


Figure 6-40: Block diagram of the hydrostatic stage/reaction mass position control loop with both conventional position control and acceleration feedback with $C_N(z)$ from (6.61) and $G_a(z)$ from (6.62).

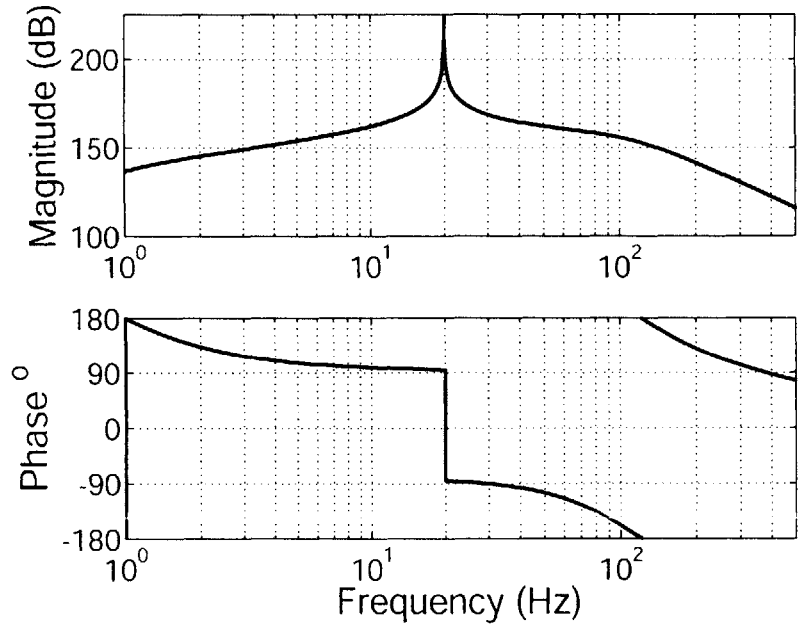


Figure 6-41: Calculated stage acceleration negative loop transmission with $C_N(z)$ from (6.61) and $G_a(z)$ from (6.62).

One way to accomplish this is to add an additional integrator to $G_a(z)$ such that the compensator transfer function becomes

$$G_a(z) = \frac{T_s^2(z^2 + 2z + 1)}{4(z^2 - 2z + 1)}. \tag{6.64}$$

As in the case for the single integrator solution, this controller requires that the accelerometer data be filtered using a high pass filter. Figure 6-42 shows the block diagram for this system while Figure 6-43 shows the calculated negative loop transmission. With properly selected loop gain selection, this acceleration feedback structure will result in a stable closed loop response for systems of any plant phase. One undesirable characteristic of this compensation structure is the inclusion of the slow responding high pass filter. We can eliminate the need for the high pass filter replacing the integrator poles at $z = 1$ with low frequency poles inside the unit circle. In this case, I replaced (6.64) with

$$G_a(z) = \frac{T_s^2}{(z - 0.999)^2}, \tag{6.65}$$

which is the same as replacing the poles in a double forward Euler integrator with 2 poles at 1 Hz (these are essential the high pass filter poles with $\zeta = 1$). Notice that this does not really speed up the loop dynamics since we retain the slow pole dynamics of the high pass filter, but we do simplify the overall control structure. Figure 6-44 shows the block diagram for this structure and Figure 6-45 shows the negative loop transmission frequency response. In practice this controller was stable but displayed more sensitivity to the low frequency drift in the accelerometer measurement. For all of the data presented in chapter 7, we used the controller shown in Figure 6-42 which included the high pass filter, double integrator, and the phase compensated form of the AFC resonator.

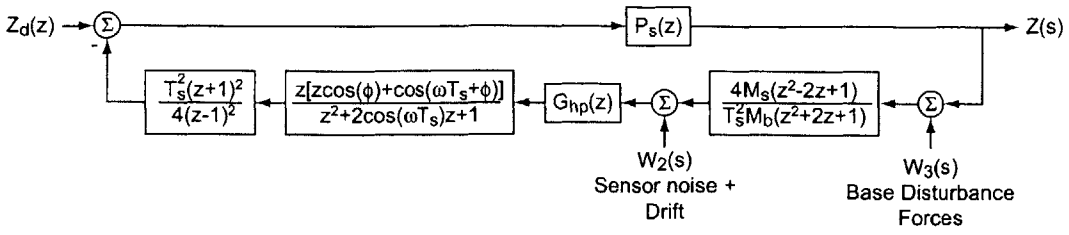


Figure 6-42: Block diagram of the hydrostatic stage/reaction mass position control loop with both conventional position control and acceleration feedback with $C_N(z)$ from (6.50) and $G_a(z)$ from (6.64).

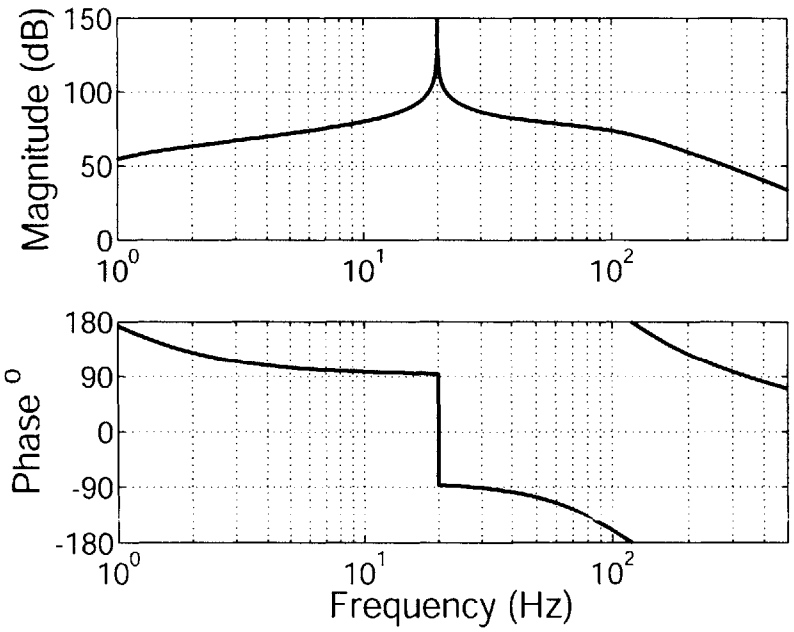


Figure 6-43: Calculated stage acceleration negative loop transmission with $C_N(z)$ from (6.50) and $G_a(z)$ from (6.64).

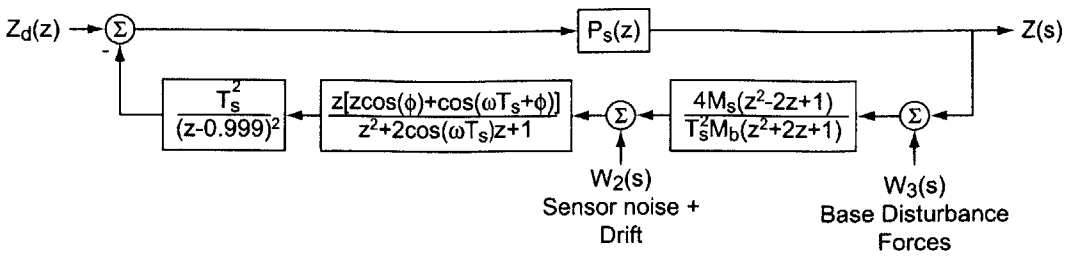


Figure 6-44: Block diagram of the hydrostatic stage/reaction mass position control loop with both conventional position control and acceleration feedback with $C_N(z)$ from (6.50) and $G_a(z)$ from (6.65).

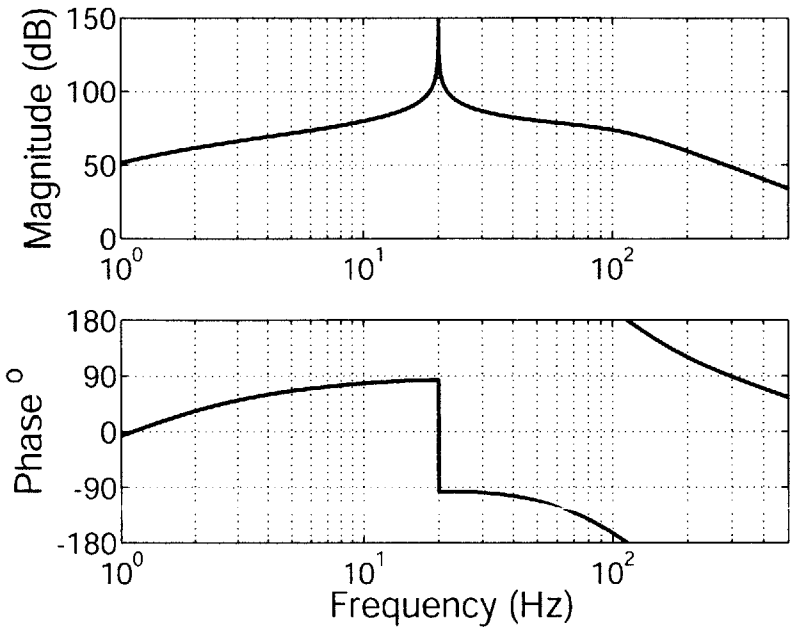


Figure 6-45: Calculated stage acceleration negative loop transmission with $C_N(z)$ from (6.50) and $G_a(z)$ from (6.65).

6.3.2 Base Acceleration Feedback Implementation

In Subsection 6.3.1, we detailed a generalized base acceleration loop structure. In this section, we will detail the base feedback implementation for the following four cases (these are the implementations for which results are presented in chapter 7):

1. A single AFC resonator at 12 Hz.

2. An eight harmonic AFC resonator with a fundamental frequency of 12 Hz.

3. An eight harmonic resonator with a fundamental frequency of 13.5 Hz.

4. A three harmonic resonator with a fundamental frequency of 23 Hz.

In the previous subsection we dealt with the acceleration feedback in general terms ignoring many of the details of our actual implementation. In order to design our actual control loop, we need to determine the actual loop gain. As was the case for the stage and fast tool servo, we are assuming that the machine base can be modeled as a free mass. Since the shipping weight of the machine was 4000 lb (1360 kg) and each of the cross-slide weigh an estimate 330 lb (120 kg), we started with a base mass estimate of $M_b = 1000$ kg. The base acceleration is measured using a PCB piezotronics Model 333A30 ceramic shear accelerometer. The accelerometer is power by a PCB Model 482A06 power supply. The accelerometer output has a gain of 100 mV/g (10.2 mV/(m/s^2)). Since the base accelerations we are measuring are on the order of 0.03 g (Max. FTS acceleration/(FTS to Stage Mass*Stage to Base Mass)), we needed to amplify the accelerometer measurement before the analog to digital conversion (the A/D resolution is on the order of ± 0.6 mV) to have meaningful resolution of the accelerometer data. Signal amplification was done using a Tektronic AM502 differential amplifier set to a gain of 200 and a 0.3 kHz bandwidth. Combining the accelerometer gains with the differential amplifier gain and the A/D gain of 0.1 units/V results in a value of 0.2 units/(m/s^2) once the accelerometer measurement is converted to discrete time. Once again, we would prefer to have the internal controller design variable to be easily convertible to actual units so we added a gain of 5 inside the model to make

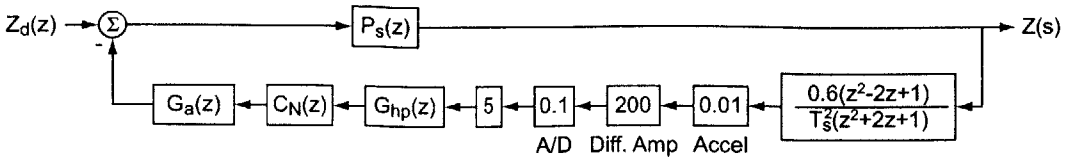


Figure 6-46: Block diagram of the hydrostatic stage/reaction mass acceleration control loop as implemented

the control signal have 1 unit=1 m/s^2. The combined position controlled inner loop and accelerometer feedback system now has the following transfer function:

$$L_{sa} = P_s(z)G_a(z)C_N(z)G_{hp}(z)\frac{0.6(z^2-2z+1)}{T_s^2(z^2+2z+1)} \tag{6.66}$$

where $P_s(z)$ is from (6.48), $G_a(z)$ as defined by (6.64), $C_N(z)$ from (6.50), and $G_{hp}(z)$ from (6.63). Figure 6-46 shows the acceleration feedback block diagram as implemented.

Single Harmonic resonator at 12 Hz

As documented in 6.3.1, we have selected $G_a(z)$ and $G_{hp}(z)$ such that we can apply our AFC tuning rules from section 4.2.2 to the AFC compensated acceleration feedback loop. Thus in all cases, we are selecting $\phi_i = \angle P_{as}(z)$ where

$$P_{as}(z) = 0.6P_s(z)G_{hp}(z)G_a(z)\frac{z^2-2z+1}{T_s^2(z^2+2z+1)}. \tag{6.67}$$

Figure 6-47 shows the calculated negative loop transmission frequency response with a single AFC resonator at 12 Hz with $g_1 = 1$ and

$$\phi_1 = \angle P_{as}(24\mathrm{j}) = 0.1298 \text{ rad}. \tag{6.68}$$

The system as modeled has a gain margin of -11 dB at 118 Hz. Selecting $g_1 = 0.01$ results in a gain margin of 29 dB at 118 Hz. Figure 6-48 show the calculated closed loop position frequency response with $g_1 = 0.01$.

Eight Harmonic Resonator with a 12 Hz Fundamental

Since the fast tool servo is following a multiple harmonic trajectory, we need to be able

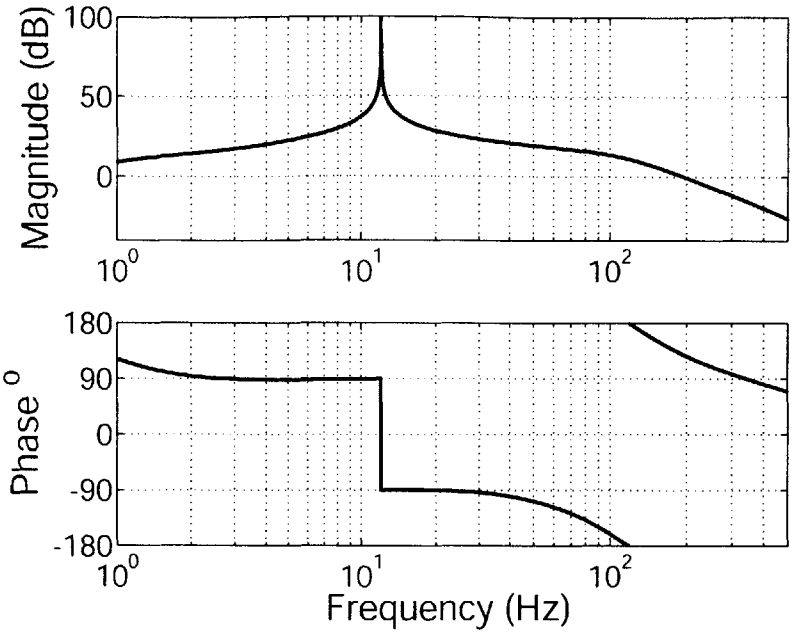


Figure 6-47: Calculated stage acceleration negative loop transmission with a single AFC resonator at 12 Hz.

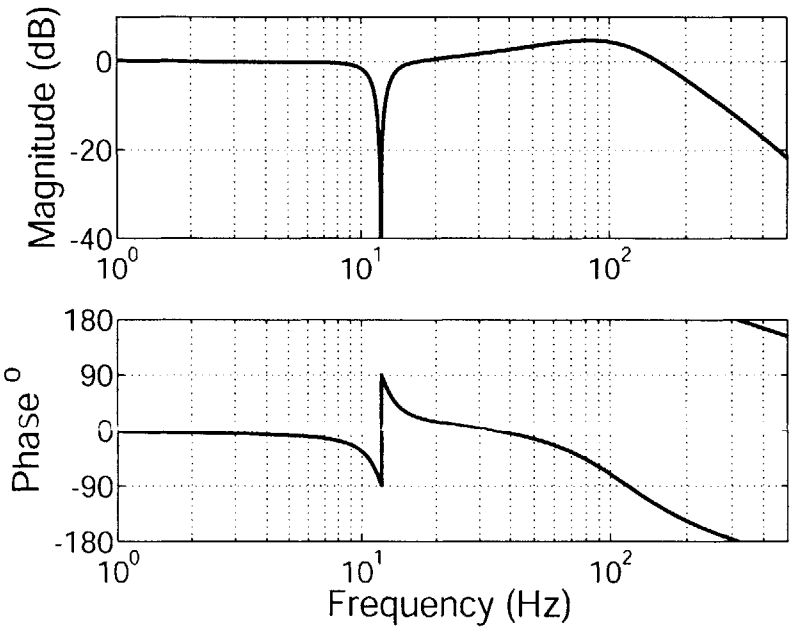


Figure 6-48: Calculated closed loop frequency response from position command to stage position with a single AFC resonator at 12 Hz with $g_1 = 0.01$.

cancel out multiple harmonics of the base acceleration. In this case, we implement an eight harmonic resonator with a fundamental frequency of 12 Hz. Setting $\phi_i = \angle P_{as}(z)$ results in

$$\phi = [0.1298,\ 0.0395,\ -0.0882,\ -0.2602,\ -0.4630,\ -0.6865,\ -0.9224,\ -1.1611]. \tag{6.69}$$

Figure 6-49 shows the calculated stage acceleration negative loop transmission with eight AFC resonators with a fundamental frequency of 12 Hz with $g_i = 1$. This system has a gain margin of -33 dB at 91.8 Hz (this is the local minima between the last two resonant peaks). Using

$$g = [0.01,\ 0.01,\ 0.001,\ 0.001,\ 0.001,\ 0.001,\ 0.001,\ 0.001], \tag{6.70}$$

results in a system with a gain margin of 17.5 dB at 93 Hz. Figure 6-50 shows the calculated stage acceleration negative loop transmission for this system. Figure 6-51 shows the calculated closed loop position frequency response with g_i from equation (6.70).

Eight Harmonic Resonator with a 13.5 Hz Fundamental

When we implemented the eight harmonic resonator with a 12 Hz fundamental described above, we found that the acceleration compensator appeared to amplify the base acceleration at 60 Hz. We believe that this was a case where the controller was attempting to cancel out a 60 Hz electrical noise signal in the acceleration measurement. One way to avoid this issue is to simply avoid acceleration compensation at 60 Hz. In this case, we simply move the fundamental frequency to 13.5 Hz, thus placing resonators at 54 and 67.5 Hz. The tuning for this was nearly identical to that for the 12 Hz fundamental. When the resonators are moved to 13.5 Hz

$$\phi = [0.1171,\ 0.0123,\ -.1483,\ -0.3585,\ -0.6008,\ -0.8627,\ -1.1315,\ -1.3927]. \tag{6.71}$$

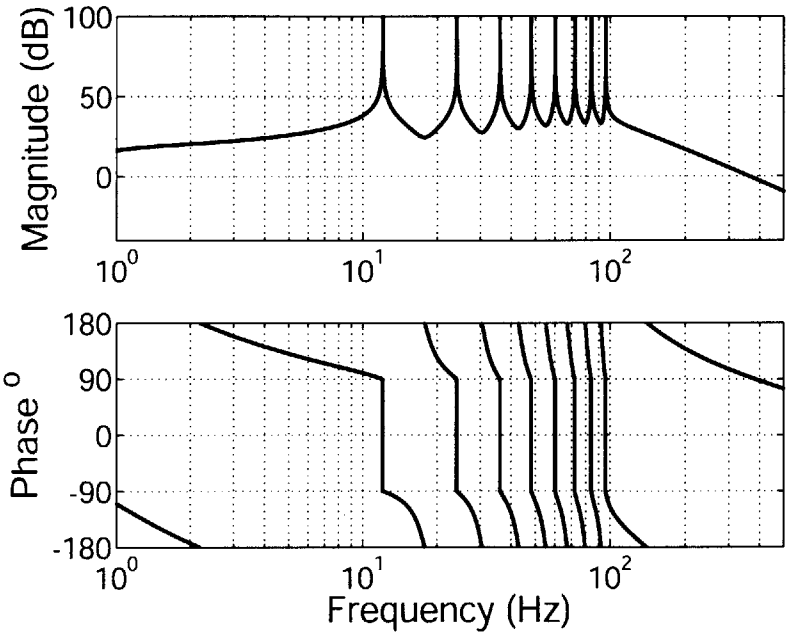


Figure 6-49: Calculated stage acceleration negative loop transmission with eight AFC resonators at 12 Hz with $g_i = 1$.

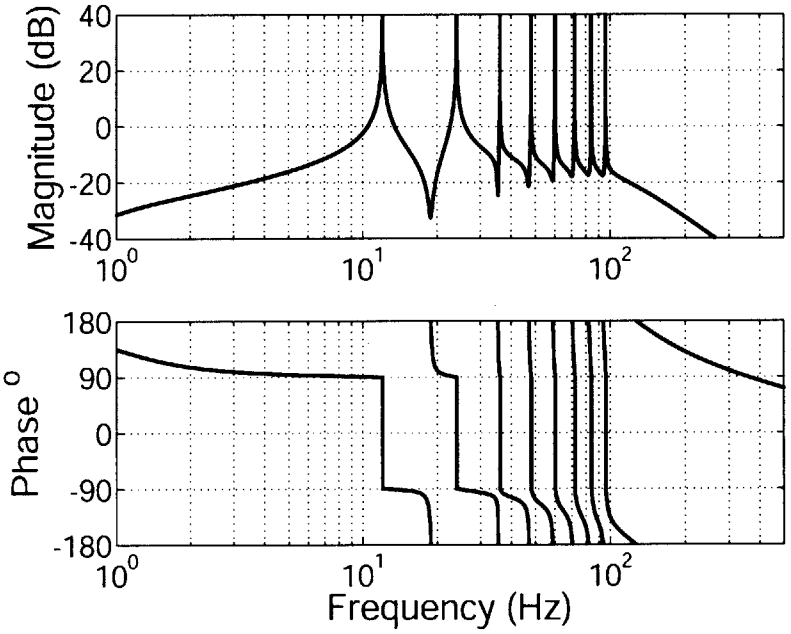


Figure 6-50: Calculated stage acceleration negative loop transmission with eight AFC resonators at 12 Hz and g_i from equation (6.70).

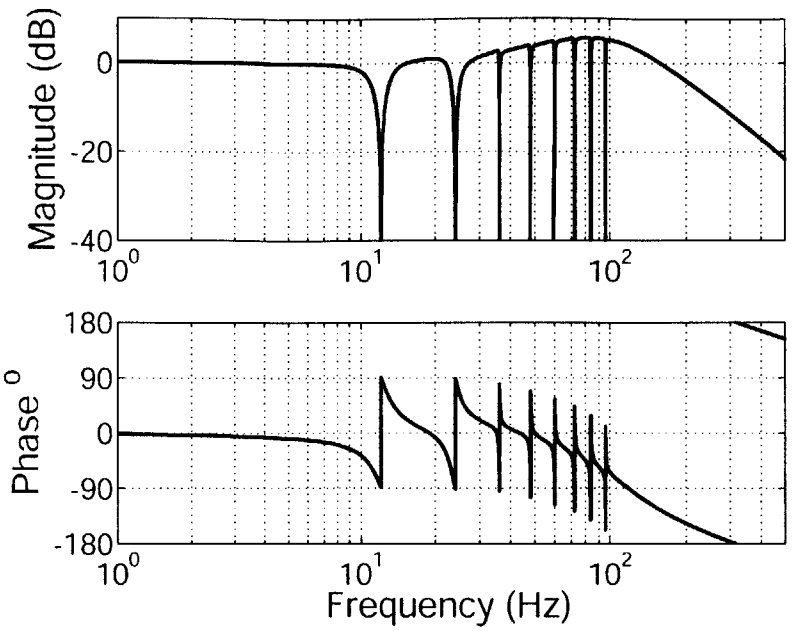


Figure 6-51: Calculated closed loop stage position frequency response with eight AFC resonators at 12 Hz andg_i from equation (6.70).

Since the resonators have only shifted slightly, we retained the gain tuning from equation (6.70). Figure 6-52 shows the calculated stage acceleration negative loop transmission for this systems. As we can see, the system now has a phase margin of 17.5 dB at 91.2 Hz. At this point we should note that for multiple resonator compensation, the system has minimum phase margin near the peak magnitude of $P_s(z)$. This makes sense since $P_s(z)$ forms the backbone of the frequency response. Figure 6-53 shows the calculated closed loop position frequency response for this compensator.

Three Harmonic Resonator with a 23 Hz Fundamental

The final AFC acceleration compensation loop implemented is a three harmonic resonator at 23 Hz. In this case, the fast tool servo was following a six harmonic trajectory with a maximum acceleration of 6.6 g's. Since we could only implement a limited number of AFC resonators at a single time, we were forced to utilize only 3 resonators in the acceleration loop. We made this choice since the fourth component

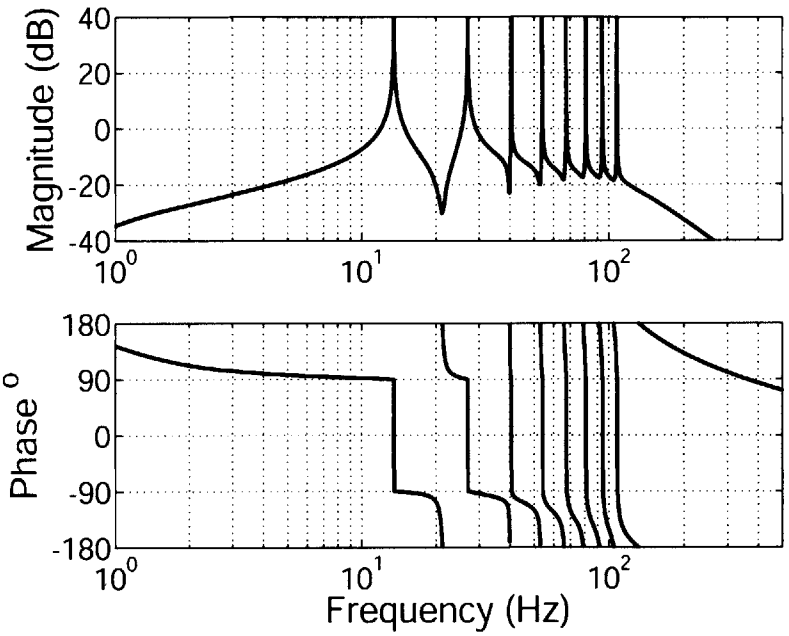


Figure 6-52: Calculated stage acceleration negative loop transmission with eight AFC resonators at 135 Hz and g_i from equation (6.70).

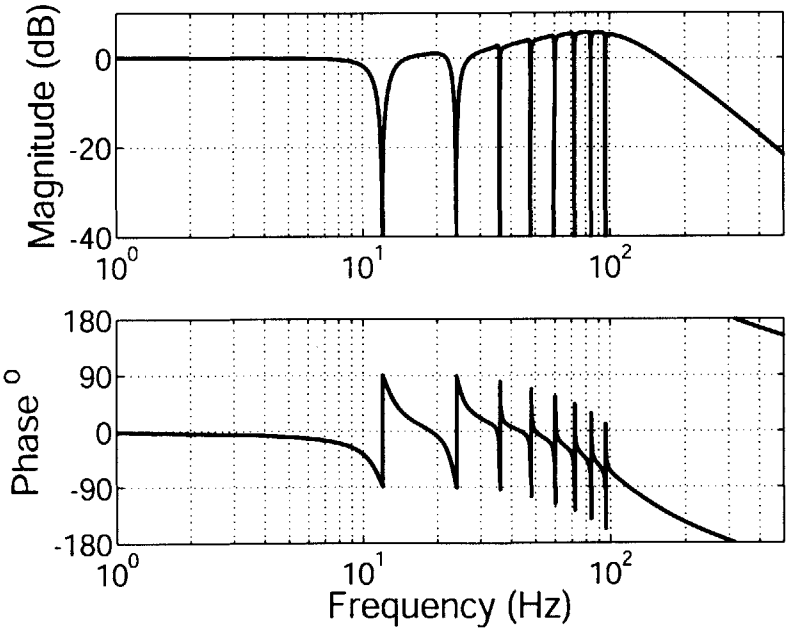


Figure 6-53: Calculated closed loop stage position frequency response with eight AFC resonators at 135 Hz and g_i from equation (6.70).

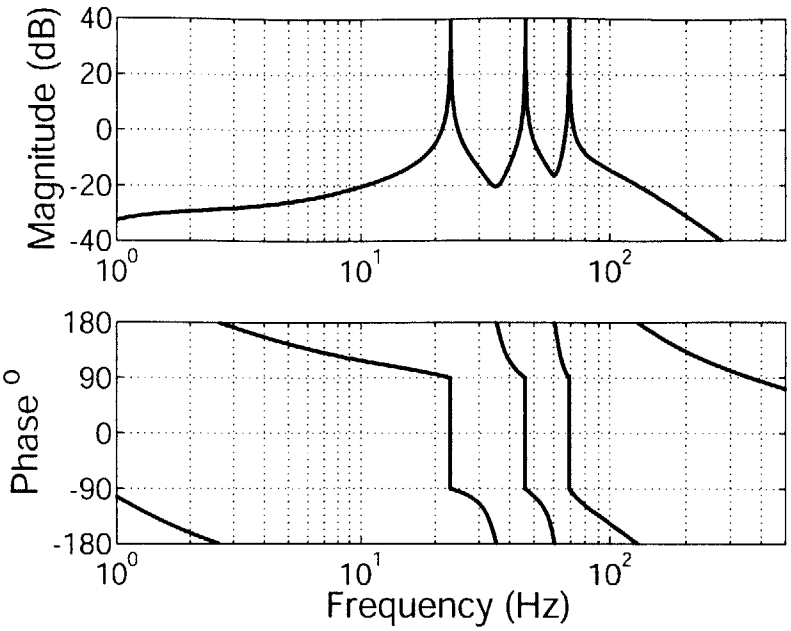


Figure 6-54: Calculated stage acceleration negative loop transmission with eight AFC resonators at 135 Hz and g_i from equation (6.73).

of the trajectory has a magnitude 1/512 the first harmonic, which theoretically results in an acceleration at the fourth harmonic 1/8th that of the first harmonic. The three harmonic controller was implemented in two forms to test the impact of resonator gain on acceleration cancellation (see Chapter 7 for the results). As always $\phi_i = \angle P_{as}(\omega_i \mathrm{j})$, which for this case results in

$$\phi = [0.0478,\ -0.2291,\ -0.6292]. \tag{6.72}$$

In the first case, we set the resonator gains to what we consider to be a reasonable maximum. In this case setting

$$g = [0.01,\ 0.01,\ 0.01], \tag{6.73}$$

results in a system with a calculated gain margin of 16 dB at 60 Hz. Figure 6-54 shows the calculated stage acceleration negative loop transmission for the compensator tun-

ing. Figure 6-55 shows the calculated position closed loop frequency response. In the second case, we reduced the gain by a factor of ten such that

$$g = [0.001,\ 0.001,\ 0.001]. \tag{6.74}$$

Figure 6-56 shows the calculated frequency response for this system. As expected, the system now has a gain margin of 36 dB at 60 Hz. Figure 6-57 shows both the calculated and measured closed loop position frequency response for this system. Given the number of block diagram manipulations and modeling approximations we made in our system model, the measured and calculated frequency response match quite well. As noted earlier in Section 6.2, we have only a limited amount of data on the closed loop frequency response of AFC compensated systems because of the large number of data points required to accurately measure the response. The measurement of the stage response was made more difficult because of the slow response of the acceleration feedback loop. The measurement of this one frequency response took on the order of 2 hours during which time the oil coolant pump failed, significantly limiting the fast tool servo utility. Thus this data set is the only one, we recorded for the acceleration feedback loop.

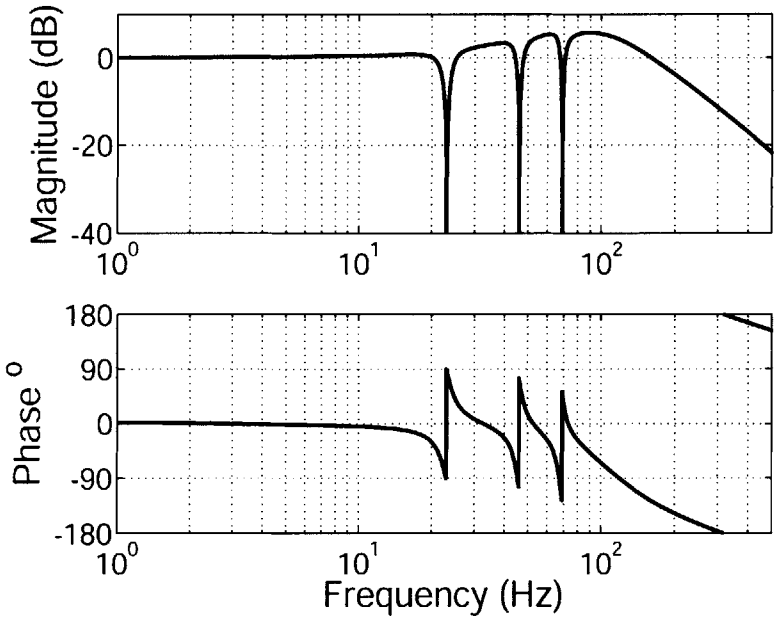


Figure 6-55: Calculated closed loop stage position frequency response with three AFC resonators at 23 Hz and g_i from equation (6.73).

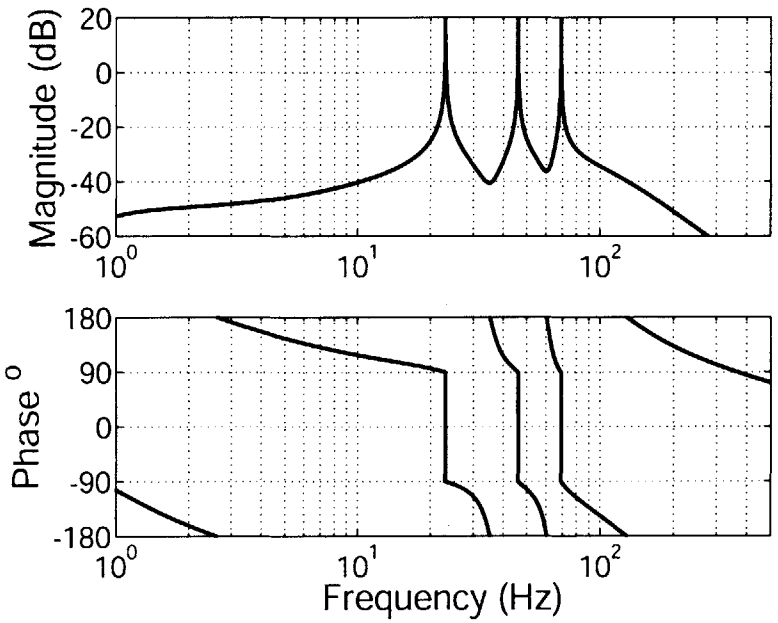


Figure 6-56: Calculated stage acceleration negative loop transmission with eight AFC resonators at 135 Hz and g_i from equation (6.74).

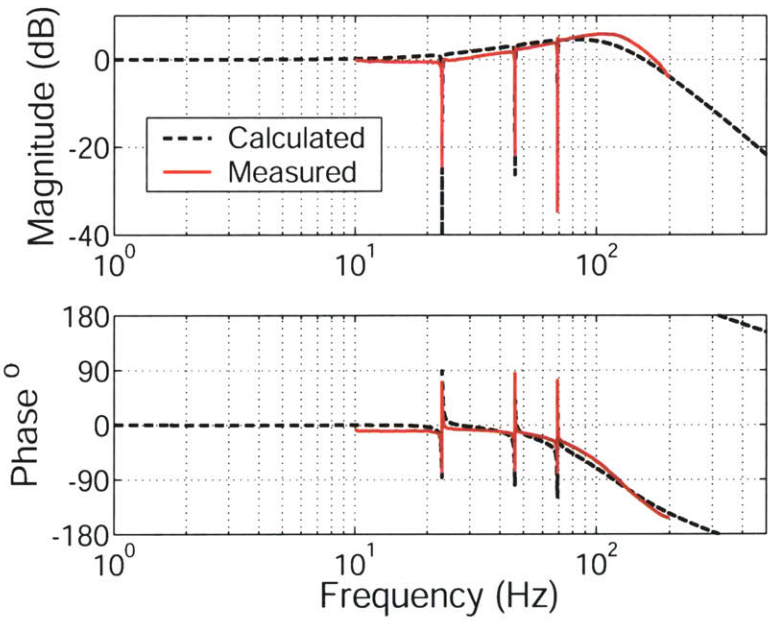


Figure 6-57: Calculated and measured closed loop stage position frequency response with three AFC resonators at 23 Hz and g_i from equation (6.74). The dashed line indicates the calculated frequency response.

6.4 Amplitude Modulated Adaptive Feedforward Cancellation

As discussed in chapter 4, one possible improvement of adaptive feedforward cancellation is to incorporate an estimate of the time varying properties of the control input or system disturbances. In Chapter 4, we proposed both a multiplicative implementation and a additive implementation. Since we derived the additive structure after disassembly of the FTS, we did not actually implement the additive structure. In the case of the multiplicative structure, the modification to the standard structure converts the AFC resonator from a linear time invariant system into system that is time varying and generally non-linear. This makes it impractical to analyze for stability using the classical control approach we have taken to tuning systems with AFC control. As we discussed in chapter 4, we can manipulate the amplitude modulated adaptive feedforward cancellation (AMAFC) block diagram to appear as a standard AFC controlled system. Thus selecting the AMAFC gains and phase as described in Chapter 4 should result in a stable control system.

Since stability analysis is quite difficult for multiplicative AMAFC structure, we limited ourselves to only a single resonator implementation. In this case, we elected to modify an existing AFC controller to AMAFC control. Specifically, we took the single 20 Hz harmonic fast tool servo AFC controller and the single 12 Hz harmonic base acceleration AFC controller, with ϕ adjusted for the new fundamental, and added the multiplicative amplitude modulation as described in chapter 4. We elected only to apply multiplicative AMAFC control to signals with non-zero values to avoid the problem of $1/\hat{A}(t)$ becoming un-bounded. Specifically, we test only the case where

$$z_d(t) = (1 + a_o \sin \alpha t) \sin(\omega_1 t) \tag{6.75}$$

where $|a_0| < 1$. Results are presented in Chapter 7.

6.5 Summary

In this chapter, we discussed in detail the design and implementation of the FTS/in-feed slide control systems. Specifically, we detailed the design and implementation of the conventional inner loop position controller for the FTS, the implementation of an AFC position control outer loop on the FTS with both single and multiple resonators, the implementation of the conventional inner-loop position controller for the in-feed slide position control, and lastly the design and implementation of an AFC controller for the in-feed slide whose purpose is to minimize the accelerations in the machine base. In the next chapter, we present the measured performance of the controller described in this chapter.

Chapter 7

Results

This chapter summarizes the results we achieved with our linear fast tool servo and hydrostatic stage/reaction mass. In the first section, we present the following error results for the fast tool servo for conventional AFC control (single and multiple harmonics). In the second section, we review the performance of the fast tool servo for a single harmonic AMAFC system. In the third section, we present the results using the hydrostatic stage as a reaction mass using both feedforward control and AFC (single and multiple harmonics). Lastly, we show the result of using AMAFC to reject a sinusoidally varying disturbance force in the DTM base.

7.1 Fast Tool Servo Adaptive Feedforward Cancelation

As detailed in chapter 6, we implemented four different AFC controllers on the fast tool servo.

1. A single 20 Hz resonator AFC controller.
2. A single 50 Hz resonator AFC controller.
3. An eight resonator AFC controller with a fundamental frequency of 12 Hz.
4. A six resonator AFC controller with a fundamental frequency of 23 Hz.

In three of the cases, we measured the following error under

1. Conventional control
2. Conventional control with command pre-shifting
3. Conventional control with AFC
4. Conventional control with command pre-shifting and AFC.

We measured the eight resonator system only under conditions 1 and 3. The performance of the fast tool servo system was limited by two factors. First, the fast tool servo, as designed and built, is limited to maximum acceleration of 10 g's continuous (temperature limit) and 14 g's peak (current limit). Second, the performance of the fast tool servo was limited by the errors of the optical encoder used for position feedback. Specifically, we are using an glass-scale encoder with a resolution of 0.004882 μm and a scale pitch of 20 μm. For accurate interpolation, the sinusoids diffracted from the glass-scale must be perfectly symmetric. In general, the diffracted sinusoids have 1-2% distortion [38]. Factoring in the interpolation error, the accuracy of the scale is ± 0.12 μm over a 20 μm movement and ± 3 μm over the entire range of the scale. Combined these two position uncertainties yields a full spectrum noise dominated by the harmonics of fundamental sinusoidal motion. Figure 7-1 plots the power spectral density of the following error for the system under conventional, conventional with command pre-shifting, AFC, and AFC with command pre-shifting control while following a 20 Hz sinusoid. Ignoring the base harmonic of the sinusoid, we can see that for all four control cases the spectral content of the error signal is constant and composed principally of components outside our control authority. For the most part the encoder error signal is approximately 1 μm pk-pk but greater errors are apparent at high velocities.

Table 7.1 contains the measured peak to peak following error, the rms following error, both the rms error component with in the controller bandwidth and outside, and the 20 Hz error component for the fast tool servo following a 20 Hz 2 mm pk-pk sinusoidal position command. The frequency specific rms error is calculated using

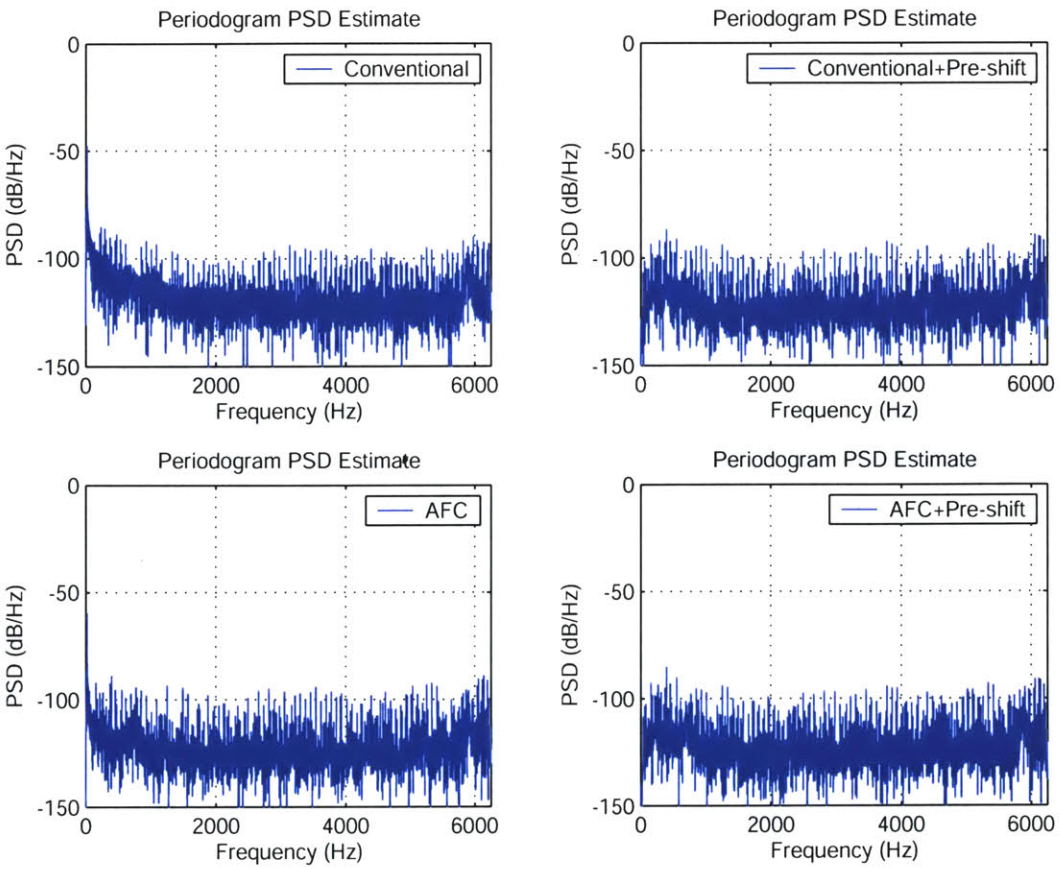


Figure 7-1: Power spectral density plots showing the full spectrum noise in the position following error while following a 20 Hz sinusoid.

Measured following error - 20 Hz, 2 mm pk-pk, 1.6 g				
	Conventional	Pre-shift	AFC	AFC+Pre-shift
Error pk-pk (mm)	0.0219	0.0065	0.0019	0.0020
$\text{Error}_{\text{rms}}$ (mm)	0.0074	0.002	0.00020	0.00020
$\text{Error}_{\text{rms}}$ (540-6250 Hz)	3.56E-4	2.50E-4	2.50E-4	2.46E-4
$\text{Error}_{\text{rms}}$(0-540 Hz)	1.05E-2	2.75E-3	1.49E-4	1.49E-4
Error(20 Hz)	9.45E-3	2.46E-3	8.11E-7	3.037E-7

Table 7.1: Following error summary for 20 Hz

the following formula

$$\text{Error}_{rms} = \sqrt{\sum_{n1}^{n2}(X[n])^2} \tag{7.1}$$

where $X[n]$ is the Fourier transform coefficient found by an N point FFT, $f_1 = f_s * n_1/N$, and $f_2 = f_s * n_2/N$ [67]. Figure 7-2 displays the measured error as a function of time, while Figure 7-3 shows the Fourier transform of the position error. The following error is 1.1% with conventional control, 0.3% with conventional control with pre-shifting, and 0.1% with the addition of AFC. Of more interest, the component of the following error at 20 Hz is 0.95% for conventional control, 0.25% with pre-shifting, 0.000081% with AFC, and 0.00003% with AFC and pre-shifting. This shows the utility of AFC in canceling a constant amplitude sinusoid.

Table 7.2 summarizes the measured following error results for a 0.4 mm pk-pk 50 Hz sinusoid, while Figures 7-4 and 7-5 show measured following error with respect to time and frequency. The following error is 10% under conventional control, 3% with pre-shifting, 0.43% with AFC, and 0.48% with AFC and command pre-shifting. The percentage numbers for the AFC compensated systems are a little misleading since the peak to peak error is almost exactly that seen following the 20 Hz sinusoid and consist primarily of the sensor uncertainty. The percent following error has gone up because the magnitude of the trajectory has dropped by a factor of 5. As we can see in Figure 7-5, the following error component at 50 Hz drops by a factor of 3 with command pre-shifting and a factor of 5000 with AFC control.

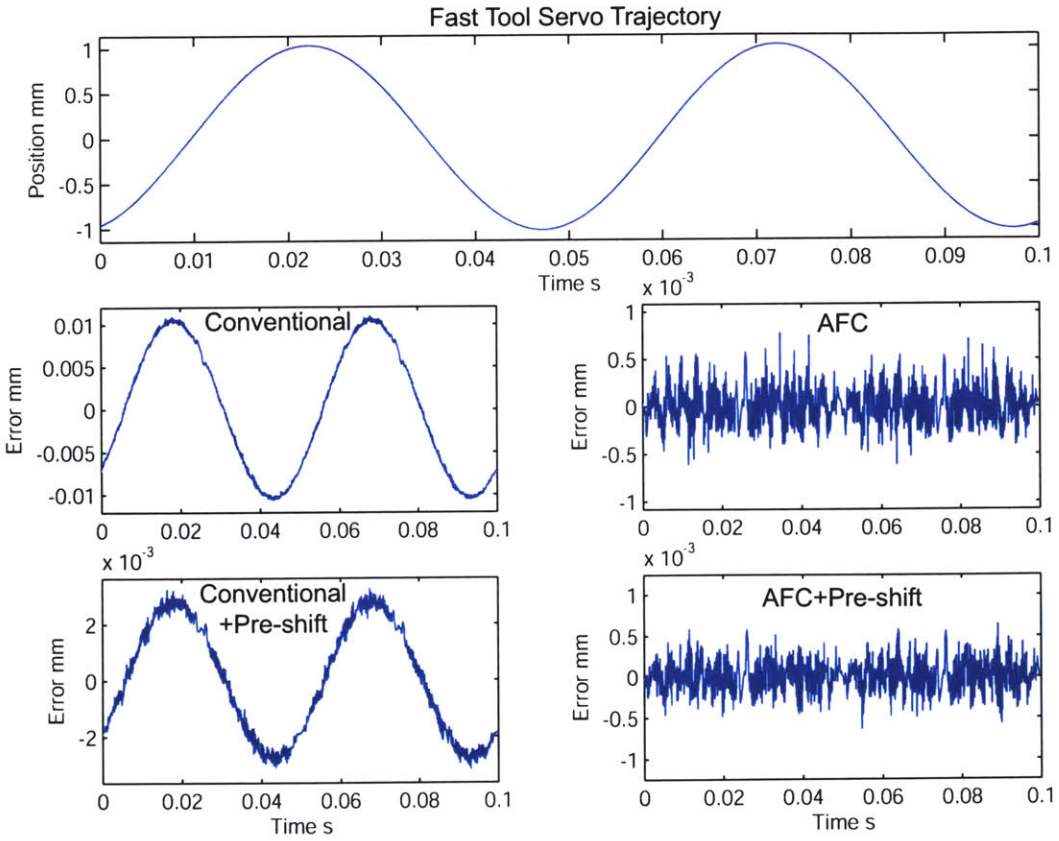


Figure 7-2: Measured fast tool servo following error to a single 20 Hz 2 mm pk-pk position command under conventional control, conventional control with command pre-shifting, AFC control, and AFC control with pre-shifting.

Measured following error - 50 Hz, 0.4 mm pk-pk, 2 g				
	Conventional	Pre-shift	AFC	AFC+Pre-shift
Error pk-pk (mm)	0.04	0.012	0.0017	0.0019
$\text{Error}_{\text{rms}}$ (mm)	0.0144	0.004	0.000252	0.000253
$\text{Error}_{\text{rms}}$ (540-6250 Hz)	3.04E-4	3.055E-4	2.55E-4	2.57E-4
$\text{Error}_{\text{rms}}$(0-540 Hz)	2.035E-2	5.67E-3	2.49E-4	2.47E-4
Error(50 Hz)	1.97E-2	5.01E-3	1.43E-6	2.37E-6

Table 7.2: Following error summary 50 Hz

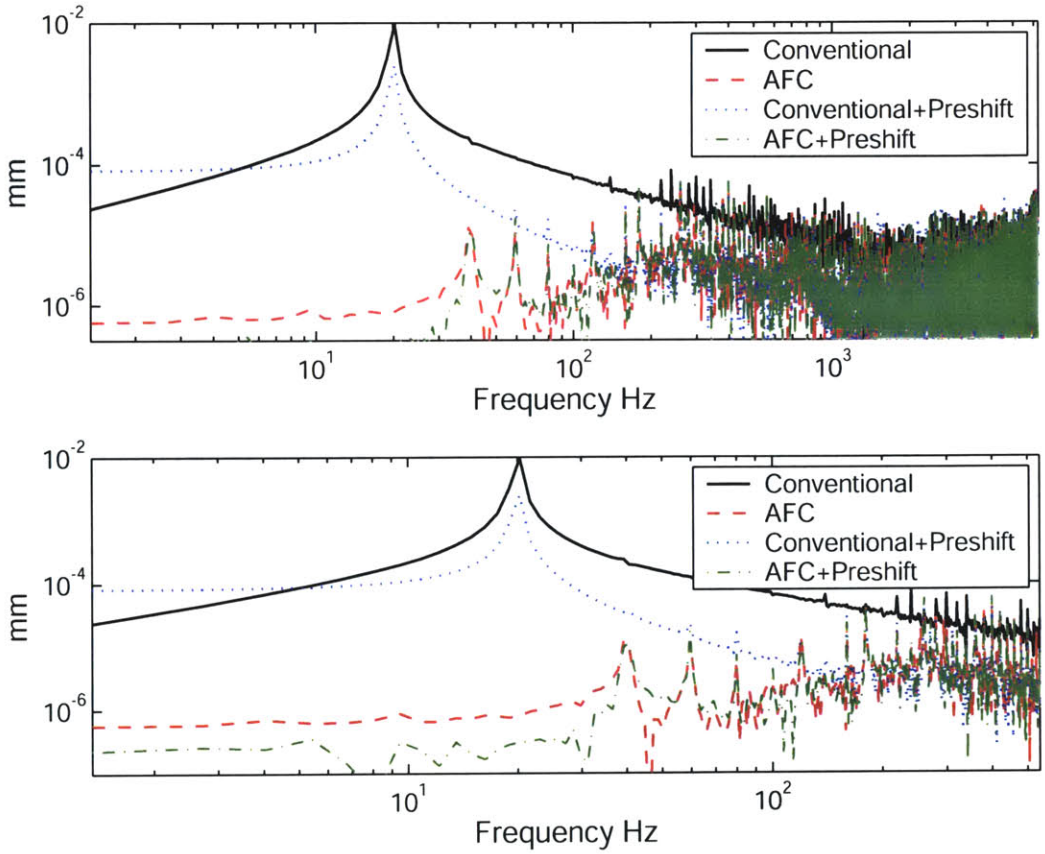


Figure 7-3: Fourier transform of measured fast tool servo following error to a single 20 Hz 2 mm pk-pk position command under conventional control, conventional control with command pre-shifting, AFC control, and AFC control with pre-shifting. Upper plot displays results from 1 Hz to 6250 Hz. Lower plot shows data from 1 to 540 Hz.

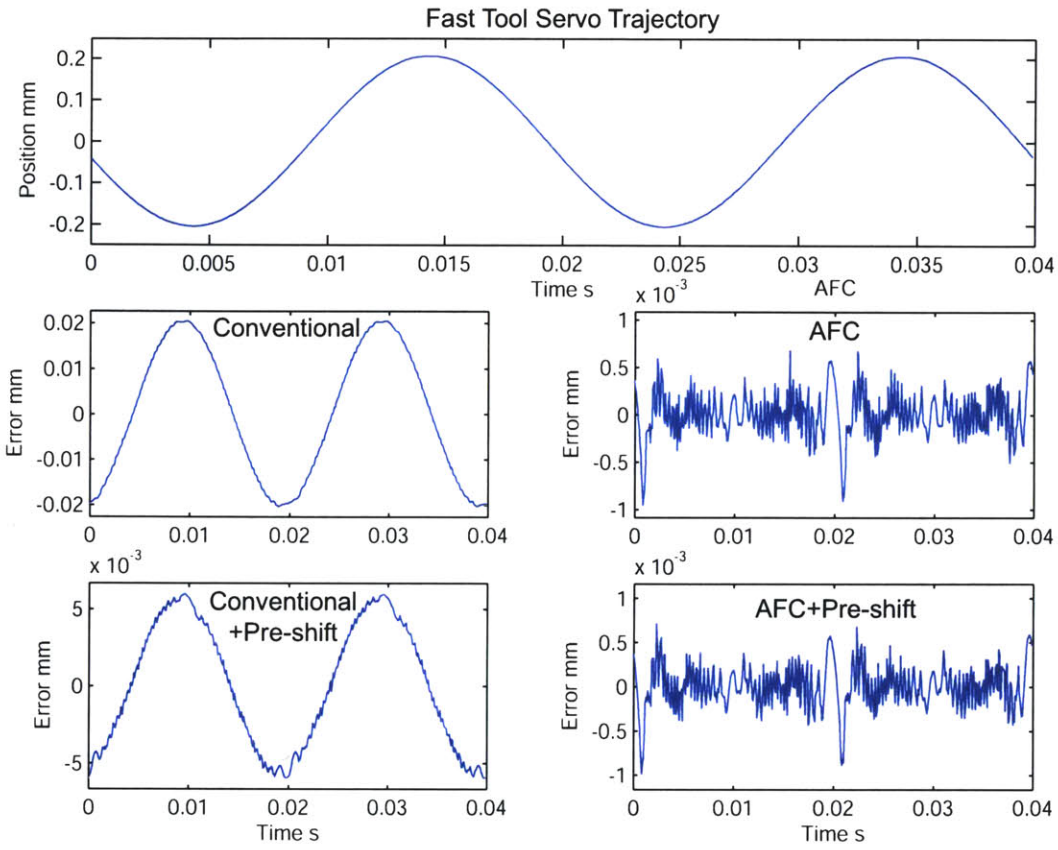


Figure 7-4: Measured fast tool servo following error to a single 50 Hz 0.4 mm pk-pk position command under conventional control, conventional control with command pre-shifting, AFC control, and AFC control with pre-shifting.

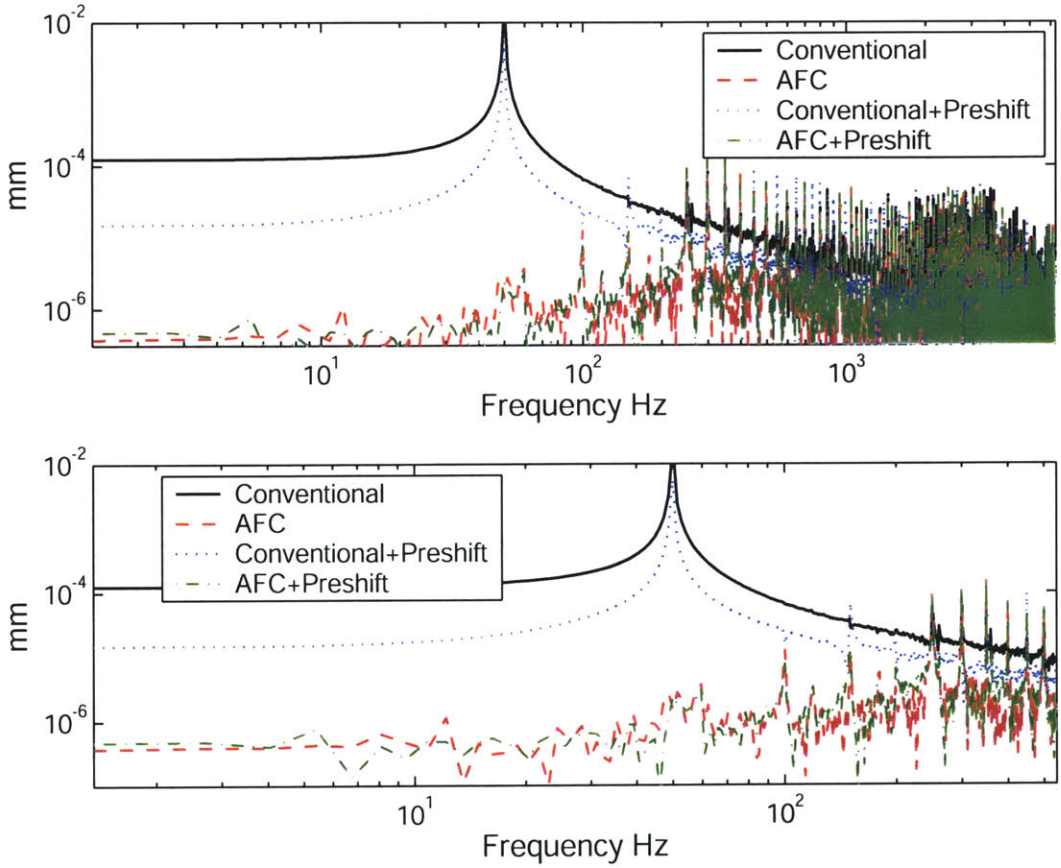


Figure 7-5: Fourier transform of measured fast tool servo following error to a single 50 Hz 0.4 mm pk-pk position command under conventional control, conventional control with command pre-shifting, AFC control, and AFC control with pre-shifting. Upper plot displays results from 1 Hz to 6250 Hz. Lower plot shows data from 1 to 540 Hz.

Measured following error for an eight harmonic 12 Hz fundamental, 5 mm pk-pk, 2.2 g sinusoid		
	Conventional	AFC
Error pk-pk (mm)	.027	0.0060
$Error_{rms}$ (mm)	0.0079	0.00057
$Error_{rms}$ (540-6250 Hz)	3.00E-4	2.99E-4
$Error_{rms}$(0-540 Hz)	1.11E-2	8.12E-4
$Error_{rms}$(200-540 Hz)	1.62E-4	8.095E-4

Table 7.3: Following error summary for eight harmonic trajectory with 12 Hz fundamental

Table 7.3 summarizes the measured following error results for an eight harmonic 5 mm pk-pk, 12 Hz sinusoidal trajectory. The trajectory coefficients are

$$A = [2.5\ 0.312\ 0.039\ 0.0048\ 0.001\ 0.001\ 0.001\ 0.001]. \tag{7.2}$$

Figures 7-6 and 7-7 show the measured following error as a function of time and frequency. We can see a number of interesting results in these plots. First, AFC control has reduced the following error at each of the resonator harmonics below $2x1^{-6}$ mm which is comparable to the results achieved in the single resonator cases. On the other hand, the peak to peak following error with 8 AFC resonators is much larger than in the single resonator tuning. Looking at the frequency response, we can see that the eight resonator tuning has significantly increased the following error in the 200-400 Hz range (rms following error has increased from $1.62x10^{-4}$ mm for conventional control to $8x10^{-4}$ for AFC control in the 200-400 Hz range). Going back to Figure 6-30, we see that this implementation of an eight resonator AFC controller increases the closed loop frequency response of the fast tool servo by about 10 dB in the range of 200-400 Hz over the conventionally controlled fast tool servo. Since the encoder error is added to the actual position signal, any increase in the closed loop response also increases the controller response to the error. Although this system is stable, the increase error in this range lead me to reduce my controller gains for future multiple resonator implementations.

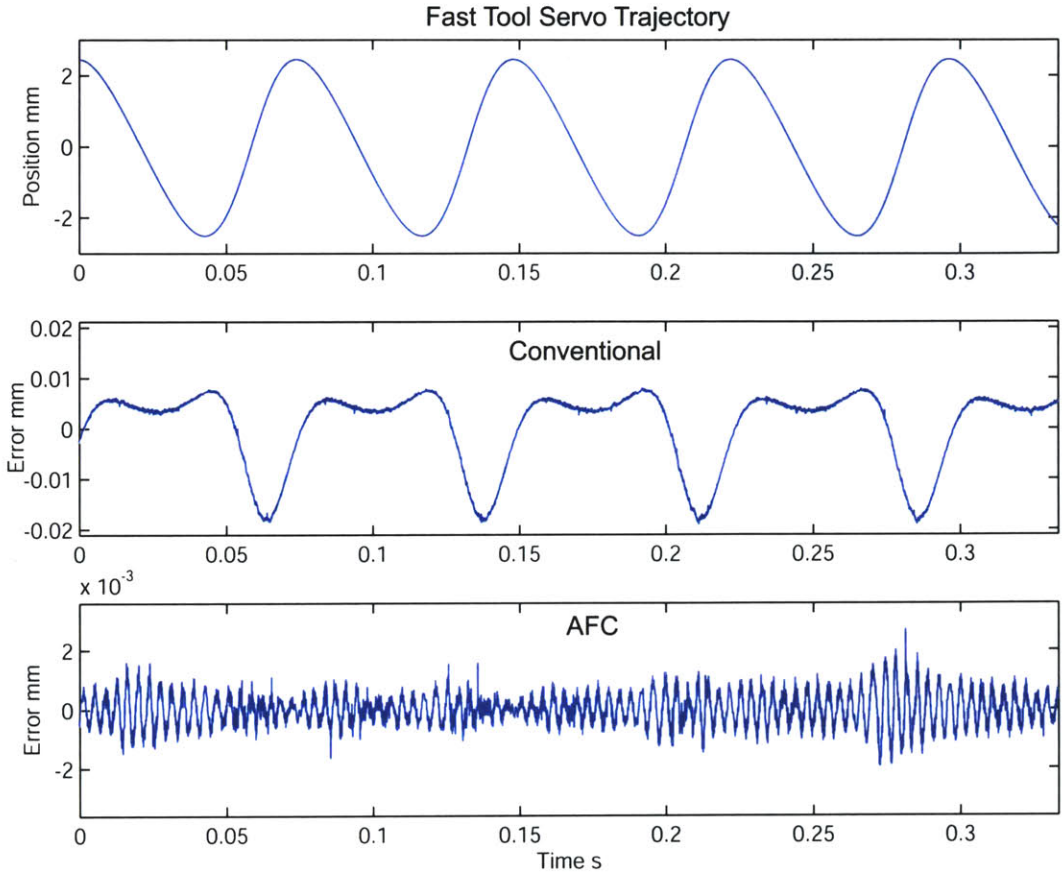


Figure 7-6: Measured fast tool servo following error for an eight harmonic 12 Hz fundamental, 5 mm pk-pk sinusoid with conventional control and an eight resonator AFC controller.

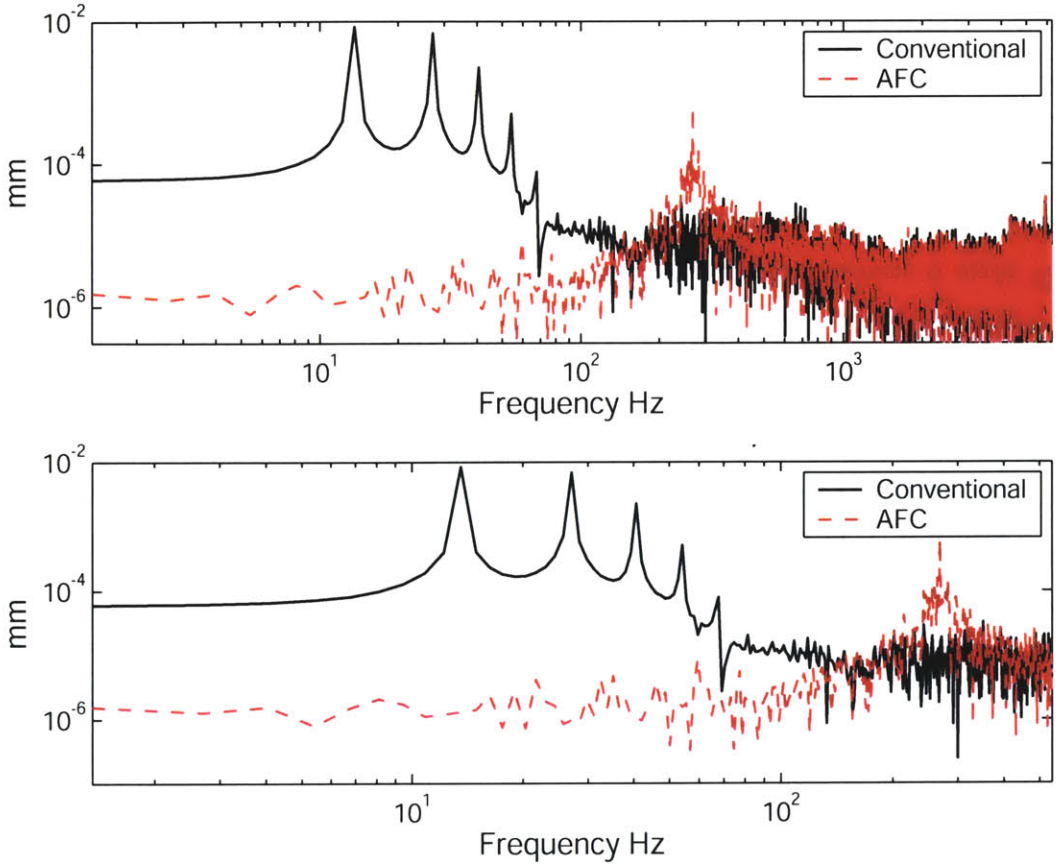


Figure 7-7: Fourier transform of measured fast tool servo following error to an eight harmonic 12 Hz fundamental, 5 mm pk-pk sinusoid with conventional control and an eight harmonic AFC controller. Upper plot displays results from 1 Hz to 6250 Hz. Lower plot shows data from 1 to 540 Hz.

The final AFC controller we tested was a six harmonic controller designed to cancel a 4.8 mm pk-pk 23 Hz fundamental sinusoid. The trajectory coefficients are

$$A = [2.39\ 0.299\ 0.037\ 0.0047\ 0.00058\ 7.3x10^{-5}\ 9.1x10{-}6]. \tag{7.3}$$

This trajectory resulted in a peak acceleration of 6.6 g's, the highest we tested. Table 7.4 summarizes the measured following error. Figures 7-8 and 7-9 show the following error versus time and frequency. We have a number of interesting results for this trajectory. First, the peak to peak following error for the AFC compensated systems are twice that of the single resonator cases. The majority of this additional error is occurring in a narrow band where the fast tool servo is traveling at maximum velocity (0.373 m/s). In this case it might be more informative to look at the rms error. We see that in the case of AFC control alone that the rms error is twice that of the single 50 Hz resonator system but that with feedforward control the rms errors are approximately equal. This is the one case studied where command shifting combined with AFC control resulted in a reduced following error. This improvement is quite apparent at lower frequencies where the rms error is reduced from $7.252\text{x}10^{-4}$ mm to $1.82\text{x}10^{-4}$ mm. Secondly it is interesting to note that unlike the eight harmonic 2.2 g trajectory tested earlier, this six harmonic trajectory has significant error at higher harmonics of the 23 Hz fundamental. Specifically, 7th through 9th harmonics have prominent peaks. As we can see, the six harmonic AFC controller does an excellent job canceling the first six harmonics but does nothing, even when combined with command pre-shifting, to reduce the higher harmonic components. This makes it clear that in general, we would like to employ as many AFC resonators as possible to reduces all the harmonics. In general, the following error contains more harmonics of the fundamental than the trajectory. This will become quite clear later when we look at the base acceleration cancelation controller.

Measured following error for a six harmonic 23 Hz fundamental, 4.8 mm pk-pk, 6.6 g sinusoid				
	Conventional	Pre-shift	AFC	AFC+Pre-shift
Error pk-pk (mm)	0.103	0.029	0.0044	0.0031
$\text{Error}_{\text{rms}}$ (mm)	0.031	0.0083	0.00049	0.00028
$\text{Error}_{\text{rms}}$ (540-6250 Hz)	5.894E-4	4.234E-4	4.235E-4	3.443E-4
$\text{Error}_{\text{rms}}$(0-540 Hz)	4.412E-2	1.173E-2	7.252E-4	1.820E-4

Table 7.4: Following error summary for a six harmonic 23 Hz fundamental trajectory.

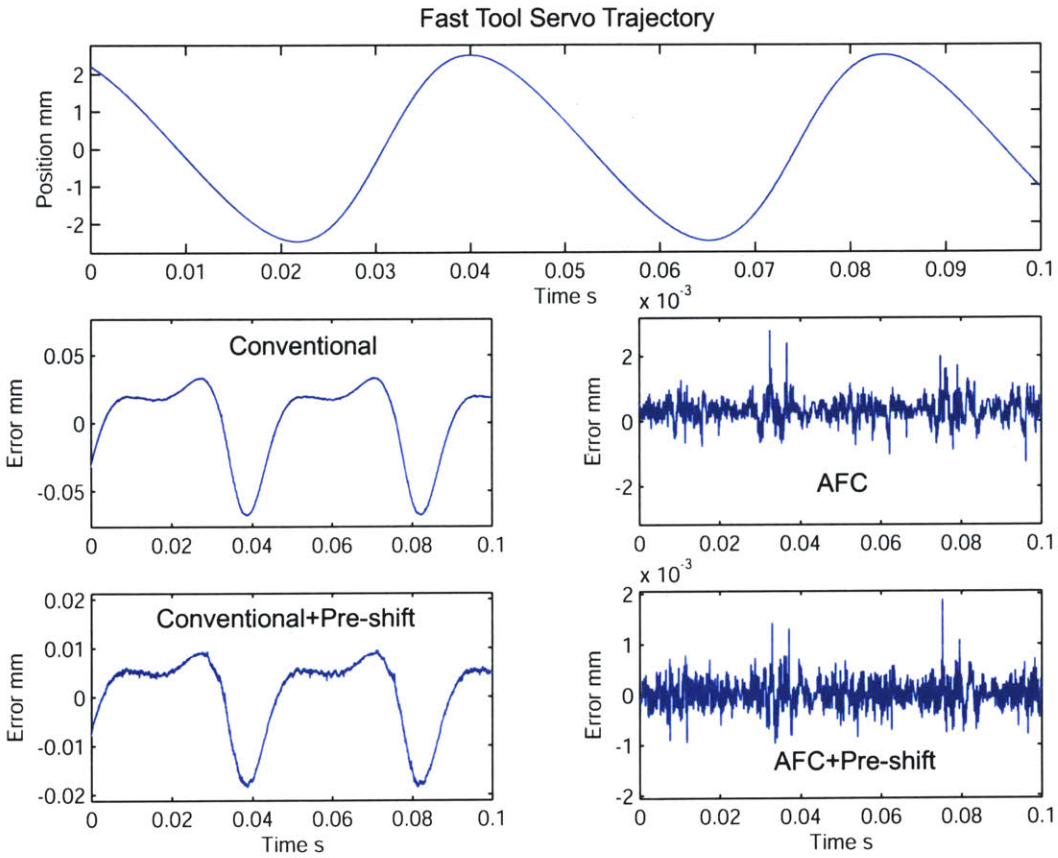


Figure 7-8: Measured fast tool servo following error to a six harmonic 23 Hz fundamental, 4.8 mm pk-pk, 6.6 g sinusoid under conventional control, conventional control with command pre-shifting, AFC control, and AFC control with pre-shifting.

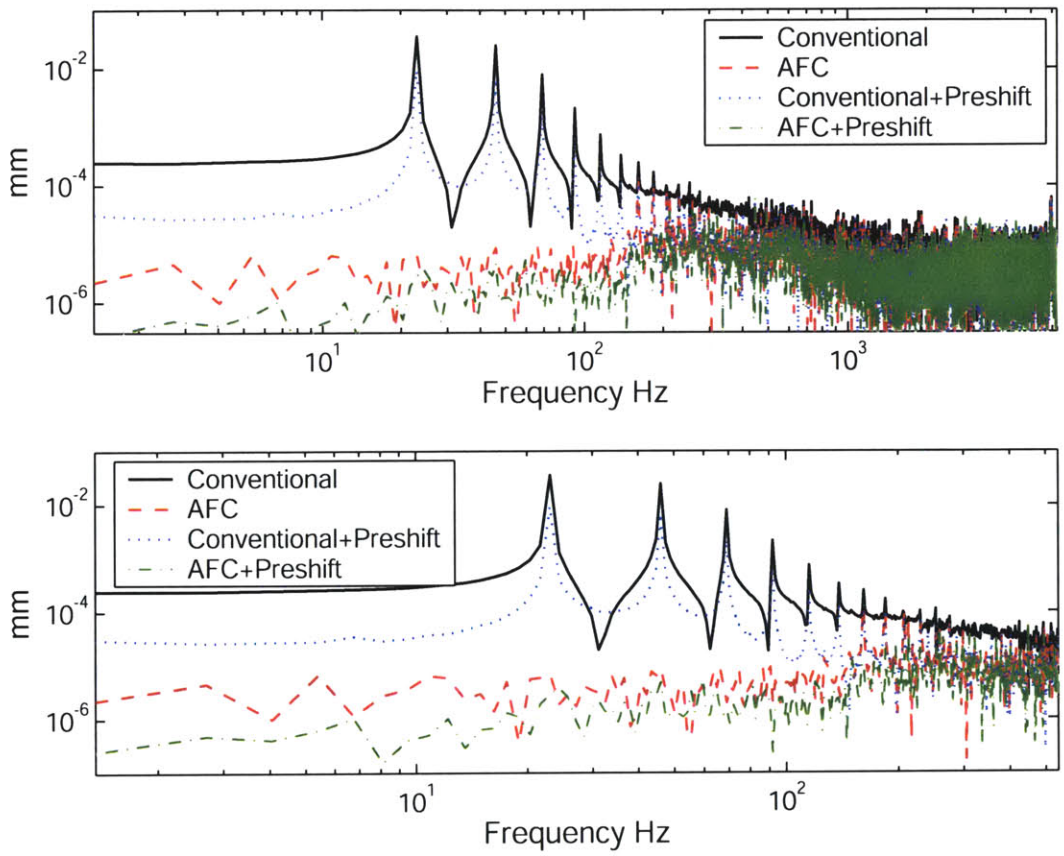


Figure 7-9: Fourier transform of measured fast tool servo following error to a six harmonic 23 Hz fundamental, 4.8 mm pk-pk, 6.6 g sinusoid under conventional control, conventional control with command pre-shifting, AFC control, and AFC control with pre-shifting. Upper plot displays results from 1 Hz to 6250 Hz. Lower plot shows data from 1 to 540 Hz.

7.2 Fast Tool Servo Amplitude Modulated Adaptive Feedforward Cancelation

In this section, we present our results using AMAFC control on the fast tool servo. We evaluated the effect of AMAFC control on a single harmonic sinusoid

$$z_d(t) = (1 + a_0 \sin \alpha t) \sin(w_1 t) \tag{7.4}$$

where $\omega_1 = 40\pi$ (20 Hz), $a_0 = 0.25$, and $\alpha = \pi$, 2π, 4π, 6π (0.5, 1, 2, and 3 Hz). The AFC controller at the heart of the AMAFC controller was that employed in the previous section to cancel the constant single harmonic 20 Hz sinusoid. The base line trajectory has a maximum acceleration of 1.6 g's. Table 7.5 summarizes the following error when using AMAFC control. Command pre-shifting is used with both the AFC and AMAFC control. Table 7.5 summarizes the measured following error for all of the trials. Figures 7-10 through 7-13 plot the following error vs time for conventional control, conventional control with command pre-shifting, AFC control with command pre-shifting, and AMAFC with command pre-shifting as α is varied from 0.5 to 3 Hz. Figure 7-14 plots the Fourier transform of the measured following error as α is varied from 0.5 to 3 Hz.

These results are a bit of a disappointment. From Table 7.5, we see there is very little difference between the measured following error for the AFC controlled system and the AMAFC controller system until the modulation term reaches 2 Hz (10% of the fundamental frequency). In fact there is very little difference in AMAFC following error compared to the AFC controller canceling a fixed magnitude 20 Hz sinusoid studied in the previous section. For both the 2 Hz and 3 Hz modulation trials, the AMAFC system shows a small but appreciable reduction in the peak to peak error and a more distinct reduction in the rms error. We see the same trends when we examine the following error vs time plots in Figures 7-10 through 7-13. In this case there are no obvious difference for the following error for AFC and AMAFC control (Note: the time window for these plots is much larger than those in the

Measured following error - 0.5 Hz Modulation				
	Conventional	Pre-shift	AFC	AMAFC
Error pk-pk (mm)	0.0144	0.0081	0.0018	0.0022
$Error_{rms}$ (mm)	0.0075	0.0020	0.00020	0.00020
$Error_{rms}$ (540-6250 Hz)	3.4E-4	2.56E-4	2.38E-4	2.41E-4
$Error_{rms}$(0-540 Hz)	9.892E-3	3.17E-3	1.40E-4	1.42E-4
Measured following error - 1 Hz Modulation				
	Conventional	Pre-shift	AFC	AMAFC
Error pk-pk (mm)	0.0142	0.00762	0.0019	0.0018
$Error_{rms}$ (mm)	0.0074	0.0020	0.00021	0.00021
$Error_{rms}$ (540-6250 Hz)	2.96E-4	2.59E-4	2.49E-4	2.49E-4
$Error_{rms}$(0-540 Hz)	1.09E-2	2.77E-3	1.54E-4	1.44E-4
Measured following error - 2 Hz Modulation				
	Conventional	Pre-shift	AFC	AMAFC
Error pk-pk (mm)	0.0149	0.0084	0.0024	0.0021
$Error_{rms}$ (mm)	0.0075	0.0020	0.00033	0.00022
$Error_{rms}$ (540-6250 Hz)	3.56E-4	2.72E-4	2.54E-4	2.51E-4
$Error_{rms}$(0-540 Hz)	1.12E-2	2.83E-3	3.99E-4	1.74E-4
Measured following error - 3 Hz Modulation				
	Conventional	Pre-shift	AFC	AMAFC
Error pk-pk (mm)	0.0152	0.0083	0.0030	0.0021
$Error_{rms}$ (mm)	0.0075	0.0021	0.00045	0.00026
$Error_{rms}$ (540-6250 Hz)	2.64E-4	2.61E-4	2.49E-4	2.54E-4
$Error_{rms}$(0-540 Hz)	1.05E-2	2.93E-3	5.99E-4	2.56E-4

Table 7.5: Following error summary following a modulated single harmonic 20 Hz sinusoid.

previous section so the underlining error structure seen in the previous section is not apparent). For $\alpha = 2$ and 3 Hz, we can see that the following error for both AFC and AMAFC control begins to match the shape of the modulated trajectory. Lastly in Figure 7-14, we see that AMAFC offers a small advantage over AFC control in reducing the error following a modulated signal. As expected from our analysis in chapter 4, the following error for both AFC and AMAFC control the error grows as the modulation frequency rises.

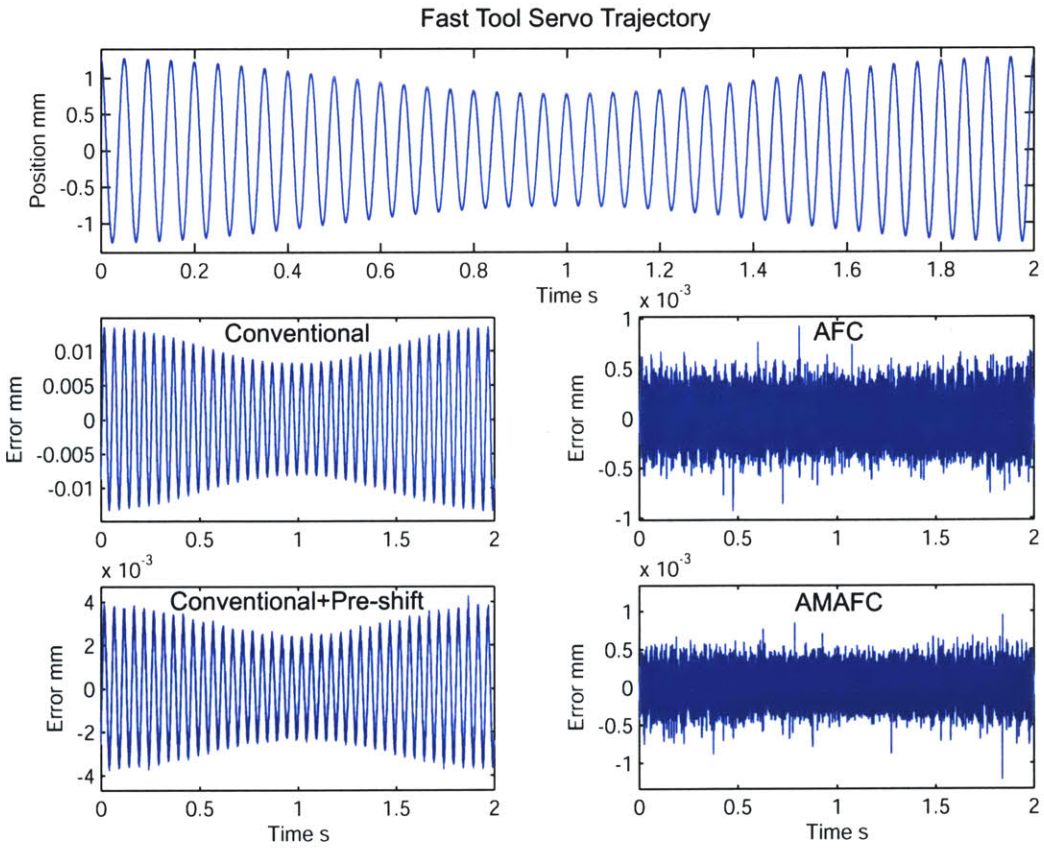


Figure 7-10: Measured fast tool servo following error to a single 20 Hz position command with 0.5 Hz modulation under conventional control, conventional control with command pre-shifting, AFC control, and AMAFC control.

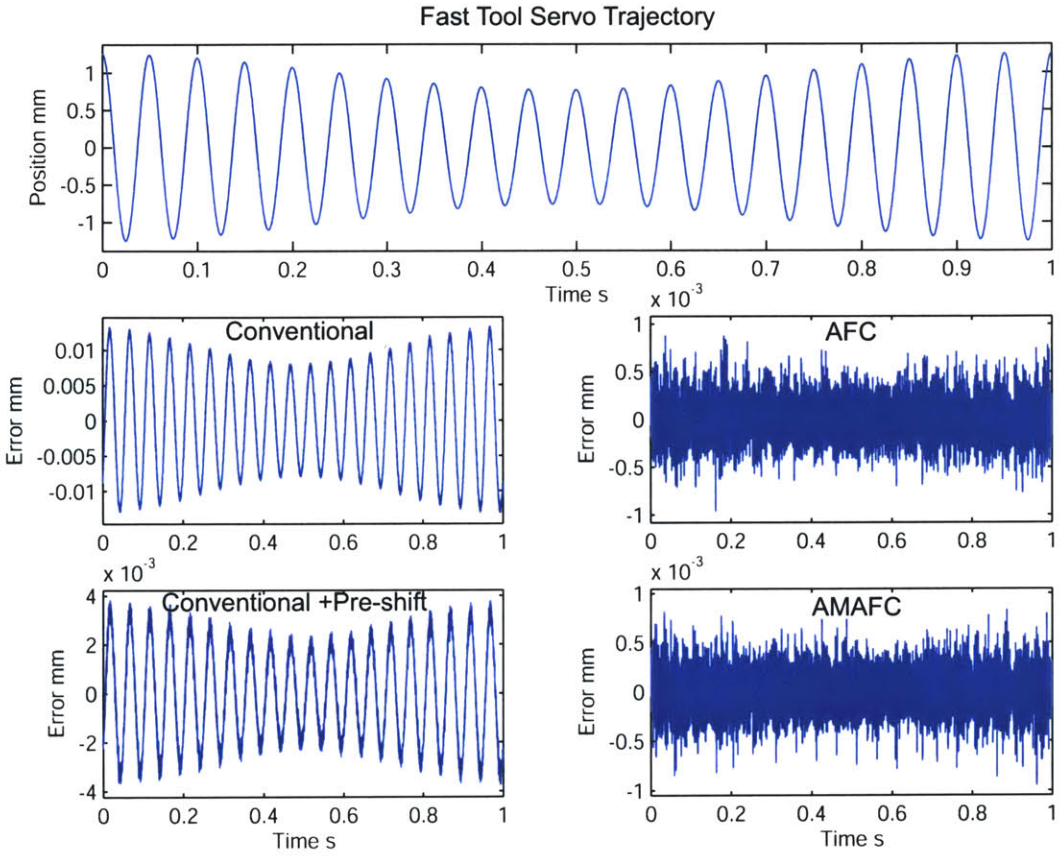


Figure 7-11: Measured fast tool servo following error to a single 20 Hz position command with 1 Hz modulation under conventional control, conventional control with command pre-shifting, AFC control, and AMAFC control.

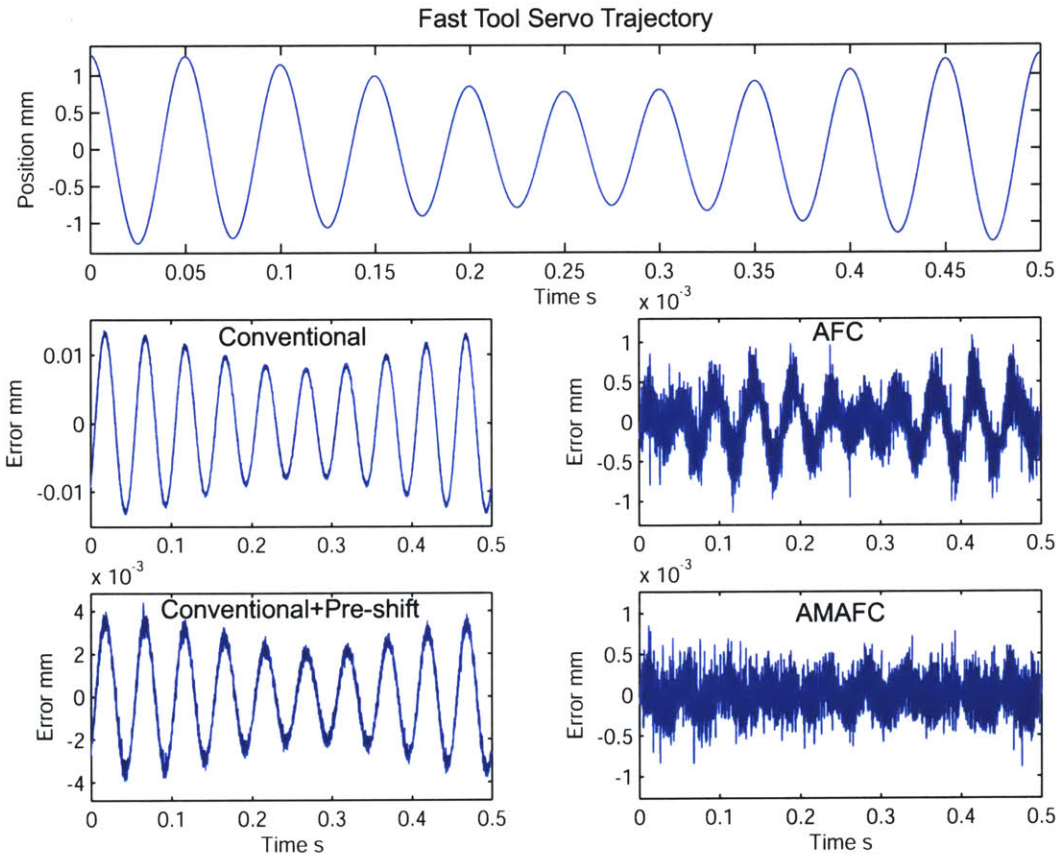


Figure 7-12: Measured fast tool servo following error to a single 20 Hz position command with 2 Hz modulation under conventional control, conventional control with command pre-shifting, AFC control, and AMAFC control.

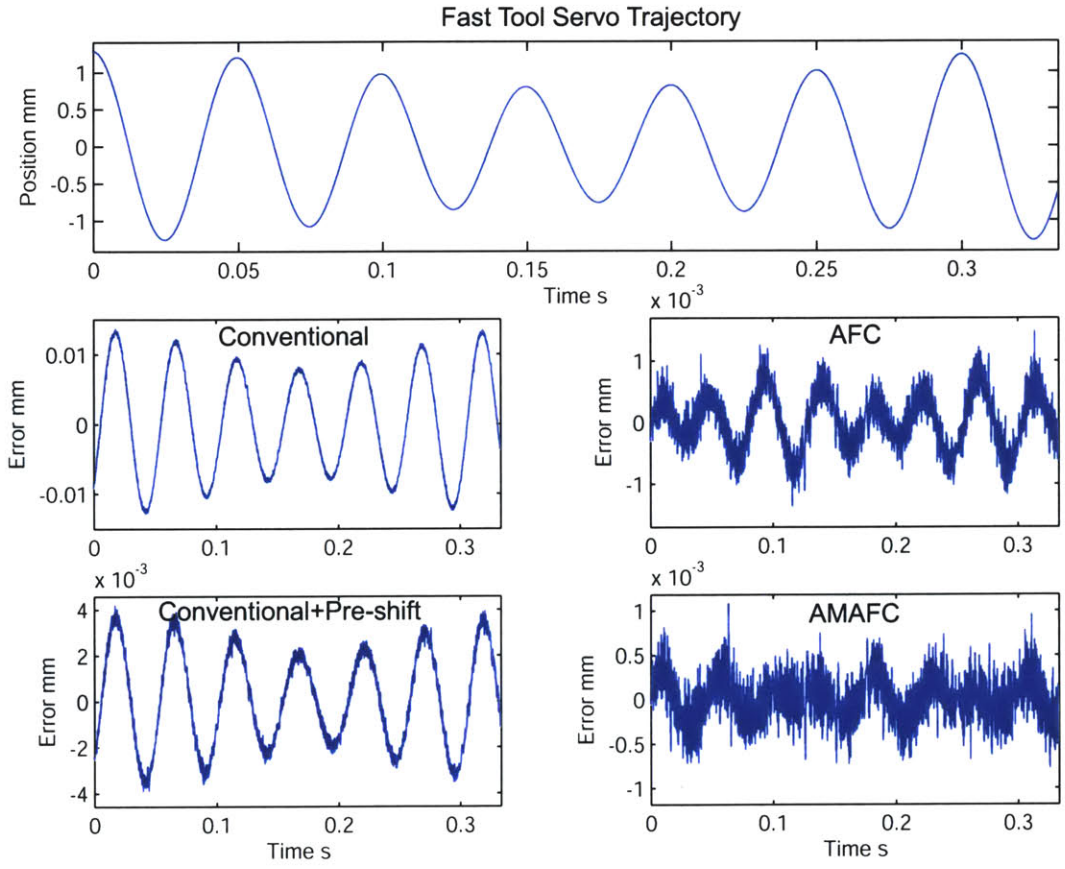


Figure 7-13: Measured fast tool servo following error to a single 20 Hz position command with 3 Hz modulation under conventional control, conventional control with command pre-shifting, AFC control, and AMAFC control.

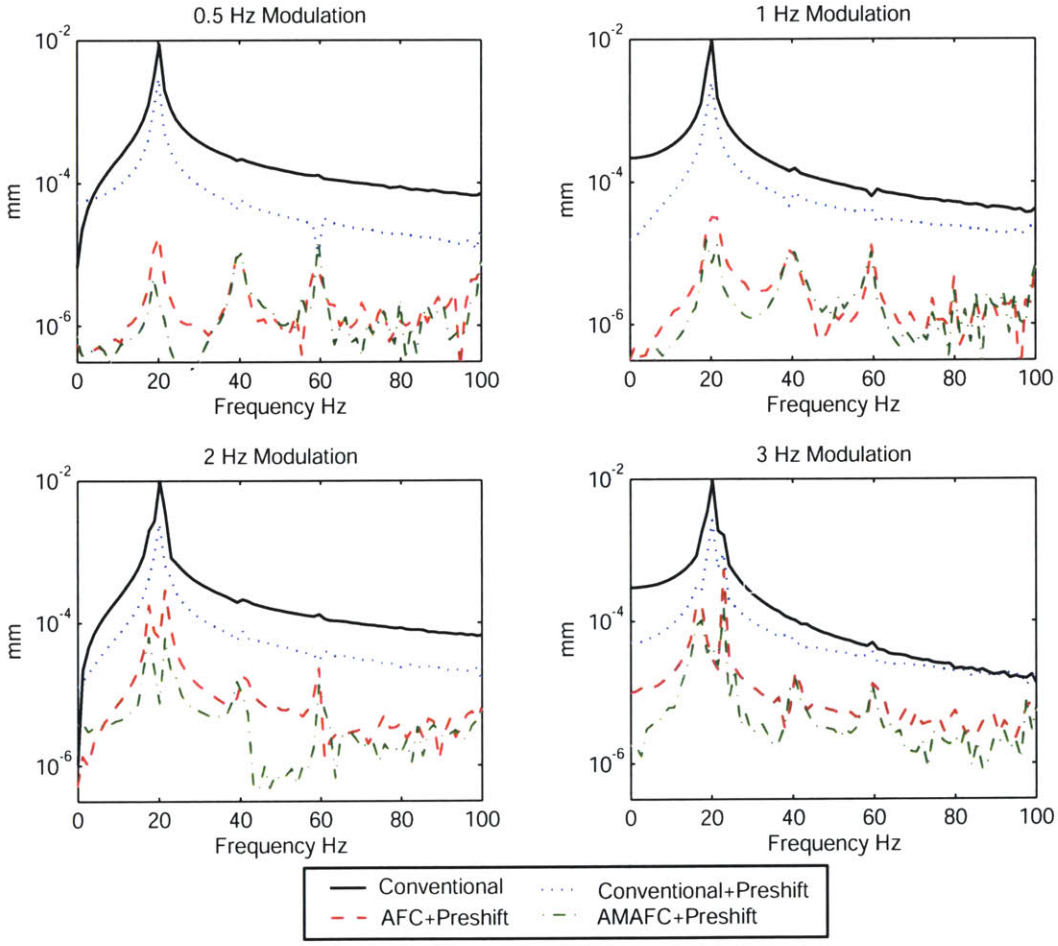


Figure 7-14: Fourier transform of the measured following error to an amplitude modulated 20 Hz sinusoid as the modulation frequencies is varied from 0.5 to 3 Hz.

7.3 Base Acceleration Feedback

In this section, we present our base acceleration feedback results. We employed two strategies to reduce base acceleration. First, we implemented a feedforward controller where the commanded stage trajectory was

$$z_{sd}(t) = -\frac{M_{\text{fts}}}{M_{\text{stage}}} z_d(t) \tag{7.5}$$

where

$$z_{sd} = \text{commanded stage trajectory} \tag{7.6}$$

$$z_d = \text{commanded fast tool servo trajectory} \tag{7.7}$$

$$M_{\text{fts}} = \text{fast tool servo mass} \tag{7.8}$$

$$M_{\text{stage}} = \text{stage mass.} \tag{7.9}$$

As we discussed in chapter 6, our model assumes that both the fast tool servo and hydrostatic stage are free masses, thus the fast tool servo actuation force is equal and opposite on the stage. This means

$$M_{\text{stage}} \ddot{z}_s = -M_{\text{fts}} \ddot{z}_{fts}. \tag{7.10}$$

Assuming that both masses have zero initial position and zero initial velocity, we have

$$z_s = -\frac{M_{\text{fts}}}{M_{\text{stage}}} z_{fts}. \tag{7.11}$$

The optimal mass ratio $M_{\text{fts}}/M_{\text{stage}}$ was determined experimentally to be 1/42=0.024. In practice, the stage is not a free mass since there are significant damping and drag forces between it and the machine base. A more optimal tuning for this would adjust both the phase and the magnitude of the feedforward signal. This leads to our second strategy, in which we use the measured base acceleration to adaptively feedforward the stage trajectory. In this case, we are using adaptive feedforward cancelation not

Measured base acceleration 12 Hz , 6 mm pk-pk, 1.72 g FTS trajectory			
	No Feedback	Feedforward	AFC
Acceleration pk-pk (m/s^2)	0.072	0.025	0.015
$Acceleration_{rms}$ (m/s^2)	0.022	0.0043	0.0020
Acceleration @ 12 Hz (m/s^2)	0.028	0.0049	3.27E-4

Table 7.6: Summary of measured base acceleration for a 12 Hz, 6 mm pk-pk, 1.72 g fast tool servo trajectory with no acceleration feedback, manually tuned feedforward, and AFC control.

to cancel an error but to cancel the base acceleration.

Before we present our results, we should note there are a number of limitations to our stage acceleration cancelation controllers. In the case of the fixed feedforward control, we know that the two masses are not really free masses. Specifically we know that there are significant damping and drag between the stage and the base. It is possible to select the magnitude and phase of the feedforward controller to account for any viscous friction and spring forces but we can not adjust for any Coulomb drag. In the case of the AFC cancelation, we automatically adjust the magnitude and phase to account for any forces at a specific frequency but just as in the case of fixed feedforward we can not completely eliminate the base acceleration because the Coulomb forces (essentially square wave in form) have components at all the harmonics of fundamental stage motion. To completely cancel the Coulomb forces, we would need an infinite number of AFC resonators. Lastly, just as in the case of the fast tool servo controller, we are limited in performance by our sensor. Specifically, the acceleration measurement is noisy. With both the fast tool servo and stage stationary (no control), the measured base acceleration was 0.008 m/s^2 pk-pk with an rms value of 0.0013 m/s^2. As we will see later, this noise is a significant portion of the measured base acceleration when the reaction mass is under AFC control. All of our acceleration measurement contain a large 60 Hz noise component, which we will see became an issue in the case of trajectory 2 (see next paragraph).

We gathered data for our reaction mass controllers for four fast tool servo trajectories:

1. A single harmonic 12 Hz 6 mm pk-pk sinusoid (1.72 g's)
2. An eight harmonic 12 Hz 6 mm pk-pk sinusoid (2.2 g's)
3. An eight harmonic 13.5 Hz 6 mm pk-pk sinusoid (2.85 g's)
4. A six harmonic 23 Hz 4.8 mm pk-pk sinusoid (6.6 g's).

For trajectories 1-3, we recorded the base acceleration without feedback, with feedforward control, and with AFC control. For the first three trajectories, the AFC controller had a resonator for each trajectory harmonic. For trajectory 4, we were unable operate the machine without reaction mass compensation since the fast tool servo reaction forces exceeded the force limit of the stage linear motor. For case 4, we recorded the base reaction forces for feedforward control and for a three resonator AFC controller with resonator gains of 0.01 and 0.001. It should be noted in cases 2 and 3, the reason we were able to run an eight resonator AFC controller on the base is that there was no AFC compensation on the fast tool servo. In case 4 we were running 6 resonators on the fast tool servo and while it was possible to run 6 resonators on the base, the data collection process caused a communication bus conflict and the data collected was polluted when we ran more than 9 total resonators.

Table 7.6 summarizes the measured base acceleration results for a single harmonic 12 Hz 6 mm pk-pk fast tool servo trajectory. Figure 7-15 shows fast tool servo trajectory, the measure base acceleration without feedback, base acceleration with feedforward control, and base acceleration with AFC control versus time. Figure 7-16 plots the Fourier transform of the measured base acceleration without feedback, with feedforward control, and with AFC control. In this case, feedforward control reduces the peak to peak base acceleration by a factor of 2.9 and rms acceleration by a factor of 5.11. Comparing AFC control to no feedback, we see a factor of 4.8 reduction in peak to peak acceleration and a factor of 11 reduction in rms acceleration. Since we know that acceleration signal has significant noise pollution, it is valuable to look at just the 12 Hz component. In this case, we see that feedforward control delivers a factor of 5.7 improvement vs no feedback while AFC reduces the 12 Hz component by a factor of 85 vs no feedback.

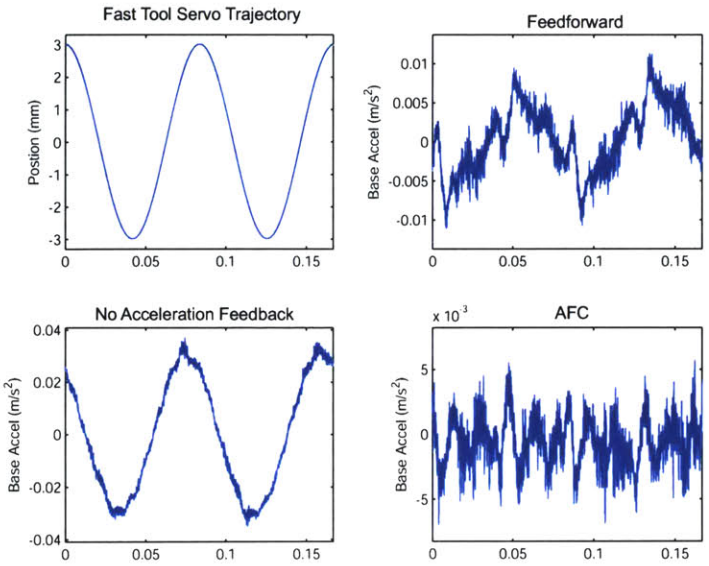


Figure 7-15: Measured base acceleration with no feedback, feedforward, and single resonator AFC control. The fast tool servo trajectory is a single harmonic 12 Hz 6 mm pk-pk sinusoid.

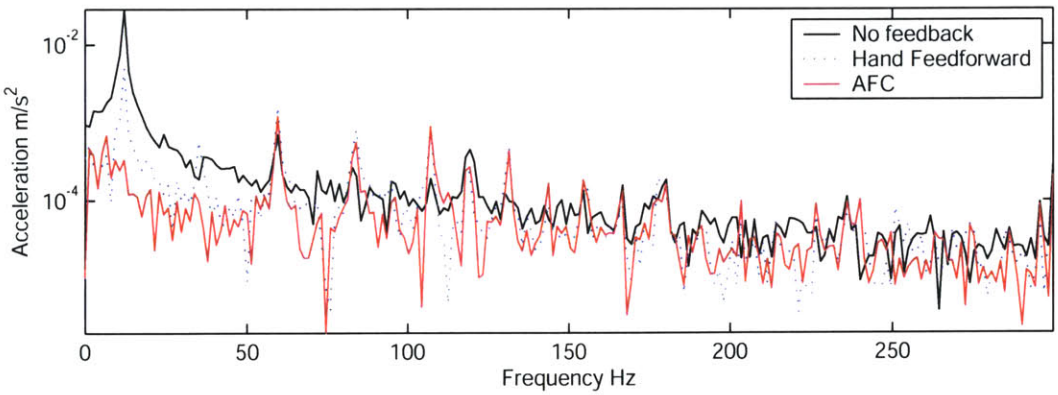


Figure 7-16: Fourier transform of base acceleration with no feedback, feedforward, and single resonator AFC control. The fast tool servo trajectory is a single harmonic 12 Hz 6 mm pk-pk sinusoid.

Measured base acceleration 12 Hz , 6 mm pk-pk, 2.2 g 8 harmonic FTS trajectory			
	No Feedback	Feedforward	AFC
Acceleration pk-pk (m/s^2)	0.085	0.0221	0.0147
$Acceleration_{rms}$ (m/s^2)	0.024	0.0042	0.0018
Acceleration @ 12 Hz (m/s^2)	0.028	0.0048	1.3E-4
Acceleration @ 24 Hz (m/s^2)	0.013	0.0011	1.42E-4
Acceleration @ 36 Hz (m/s^2)	0.0022	8.7E-5	1.91E-5
Acceleration @ 48 Hz (m/s^2)	1.75E-3	4.66E-4	7.07E-5
Acceleration @ 60 Hz (m/s^2)	5.09E-4	2.96E-4	1.03E-3
Acceleration @ 72 Hz (m/s^2)	2.56E-4	9.39E-4	6.26E-5
Acceleration @ 84 Hz (m/s^2)	2.35E-4	1.52E-4	6.53E-5
Acceleration @ 96 Hz (m/s^2)	3.56E-4	4.2E-4	1.74E-5

Table 7.7: Summary of measured base acceleration for a 12 Hz, 6 mm pk-pk, 2.2 g 8 harmonic fast tool servo trajectory with no acceleration feedback, manually tuned feedforward, and an eight harmonic AFC control.

Table 7.3 summarizes the measured base acceleration results for an eight harmonic 12 Hz 6 mm pk-pk fast tool servo trajectory. Figure 7-19 shows fast tool servo trajectory, the measure base acceleration without feedback, base acceleration with feedforward control, and base acceleration with AFC control versus time. Figure 7-20 plots the Fourier transform of the measured base acceleration without feedback, with feedforward control, and with AFC control. Once again we see that feedforward control reduces the peak to peak acceleration by approximately 3 and reduces the rms acceleration by a factor of 5. Similarly, AFC control reduces the peak to peak acceleration by a factor of 5 and the rms acceleration by a factor of 200. Looking at components of the acceleration at each trajectory harmonic, we see that AFC control reduces the first four harmonics by a factor of at least 100. AFC control is less effective on the higher harmonics partly because the magnitude of the acceleration is near the resolution limit of the accelerometer ($1.5x10^{-5}$ m/s^2). The one problem with this controller is that the AFC controller increases the magnitude of the 60 Hz harmonic. We believe this increase is due to the AFC resonator trying to cancel out the 60 Hz noise signal. To avoid this problem, we change the frequency of the fast tool servo trajectory to 13.5 Hz.

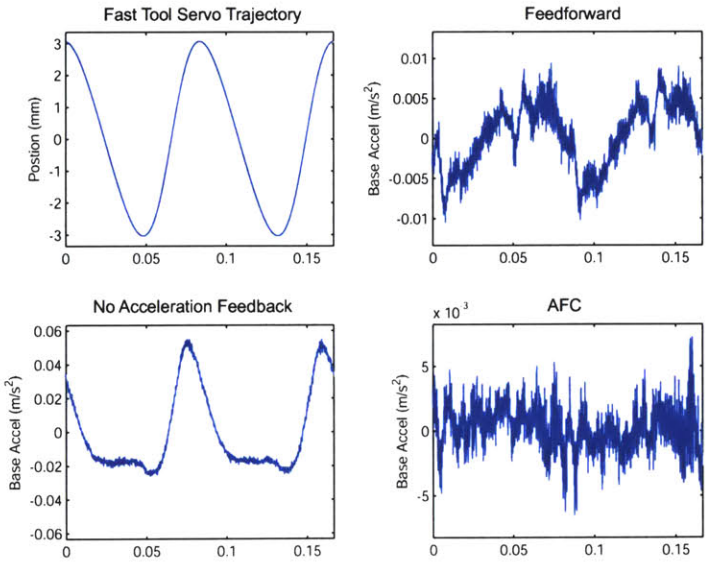


Figure 7-17: Measured base acceleration with no feedback, feedforward, and an eight resonator AFC controller. The fast tool servo trajectory is an eight harmonic 12 Hz 6 mm pk-pk sinusoid.

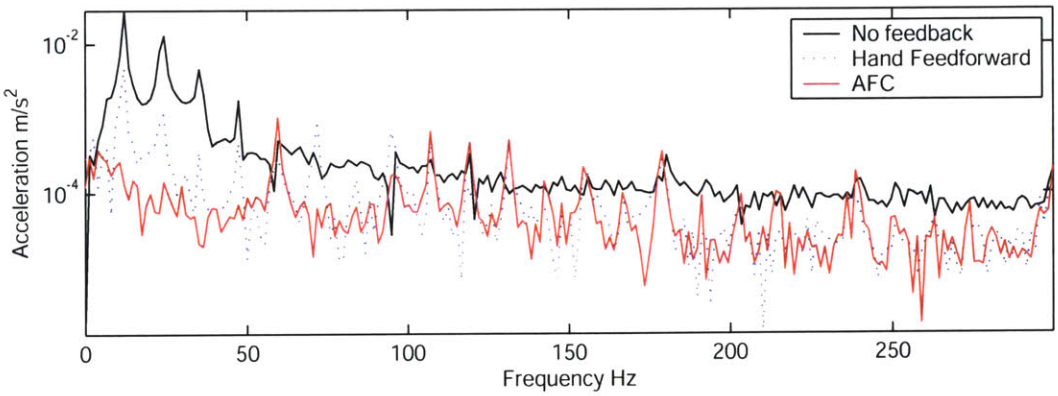


Figure 7-18: Fourier transform of base acceleration with no feedback, feedforward, and an eight resonator AFC controller. The fast tool servo trajectory is an eight harmonic 12 Hz 6 mm pk-pk sinusoid.

Measured base acceleration 13.5 Hz , 6 mm pk-pk, 2.85 g 8 harmonic FTS trajectory			
	No Feedback	Feedforward	AFC
Acceleration pk-pk (m/s^2)	0.154	0.056	0.044
$Acceleration_{rms}$ (m/s^2)	0.048	0.0115	0.0040
Acceleration @ 13.5 Hz (m/s^2)	0.063	0.0126	1.08E-4
Acceleration @ 27 Hz (m/s^2)	0.021	0.0024	8.76E-5
Acceleration @ 40.5 Hz (m/s^2)	0.0063	0.0015	3.78E-5
Acceleration @ 54 Hz (m/s^2)	3.51E-3	3.25E-4	1.32E-4
Acceleration @ 67.5 Hz (m/s^2)	3.82E-3	4.44E-3	9.42E-5
Acceleration @ 81 Hz (m/s^2)	2.84E-3	2.56E-3	1.02E-4
Acceleration @ 94.5 Hz (m/s^2)	3.25E-3	3.65E-3	8.80E-4
Acceleration @ 108 Hz (m/s^2)	1.93E-3	1.59E-3	3.38E-4

Table 7.8: Summary of measured base acceleration for a 12 Hz, 6 mm pk-pk, 2.85 g 8 harmonic fast tool servo trajectory with no acceleration feedback, manually tuned feedforward, and an eight harmonic AFC control.

Table 7.8 summarizes the measured base acceleration results for an eight harmonic 13.5 Hz 6 mm pk-pk sinusoidal fast tool servo trajectory (2.85 g's). Figure 7-19 shows fast tool servo trajectory, the measure base acceleration without feedback, base acceleration with feedforward control, and base acceleration with AFC control versus time. Figure 7-20 plots the Fourier transform of the measured base acceleration without feedback, with feedforward control, and with AFC control. Once again, we see that feedforward control and AFC control reduce the base acceleration but the reduction in the peak to peak acceleration is approximately 3, significantly less than the previous trials. The RMS acceleration results are much closer to the previous trials with reductions of a factor of 4 and a factor of 10. Looking at the acceleration component at each harmonic, we see that at low frequency harmonics AFC reduces the acceleration by a factor of 100. At the two harmonics near 60 Hz, the AFC controller has reduced the acceleration by a factor of 10. For some reasons, the AFC controller does not cancel the 94.5 Hz harmonic component as well as the other harmonics. In Figure 7-20, we can see that the higher harmonics of the fast tool servo trajectory compose a significant portion of the total base acceleration. If our DSP was sufficiently fast enough, we would want to use a minimum of 12 AFC resonators

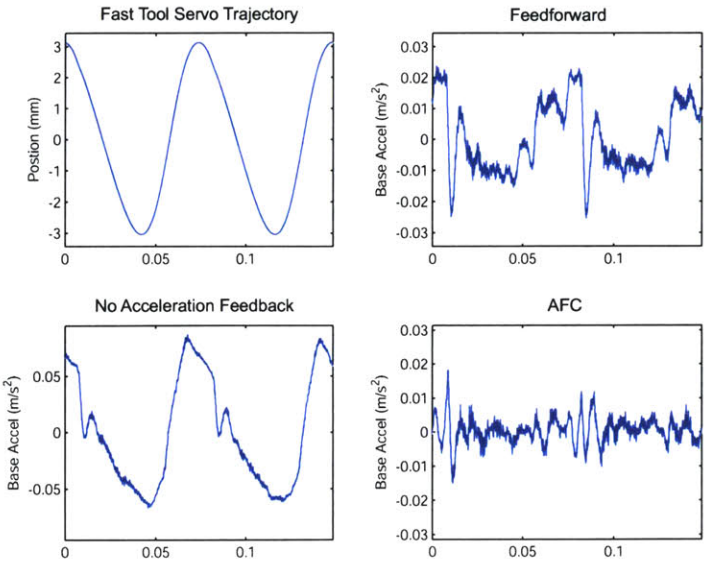


Figure 7-19: Measured base acceleration with no feedback, feedforward, and an eight resonator AFC controller. The fast tool servo trajectory is an eight harmonic 13.5 Hz 6 mm pk-pk sinusoid.

to cancel the base acceleration.

Table 7.9 summarizes the measured base acceleration for a 23 Hz, 4.8 mm pk-pk 6 harmonic fast tool servo trajectory (6.6 g's). For this trial, we measured base acceleration for feedforward control, and AFC control with resonator gains of 0.01 and 0.001. Figure 7-21 plots the base acceleration versus time, while Figure 7-22 plots the Fourier transform of the base acceleration. As mentioned earlier, we employed only 3 AFC resonators for this system so the fourth through sixth harmonics dominate the measured base acceleration. In addition to the commanded harmonic components the seventh and eight harmonics have a significant contribution. Since we could not measure the acceleration without some form of reaction mass compensation, we can not compare the controlled acceleration to it. The rest of the results are what we would expect with both forms of AFC outperforming the feedforward control and the higher gain AFC controller having better performance. In this case, the lower gain AFC controller offered only a moderate performance advantage over the feedforward controller. This illustrates the importance of proper AFC gain selection, if the gains

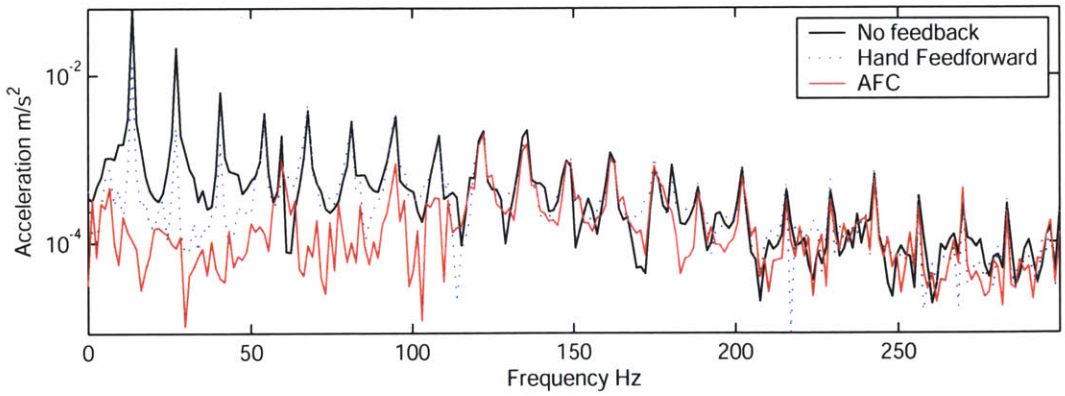


Figure 7-20: Fourier transform of base acceleration with no feedback, feedforward, and an eight resonator AFC controller. The fast tool servo trajectory is an eight harmonic 13.5 Hz 6 mm pk-pk sinusoid.

Measured base acceleration 23 Hz , 4.8 mm pk-pk, 6.6 g 6 harmonic FTS trajectory			
	Feedforward	AFC g=0.001	AFC g=0.01
Acceleration pk-pk (m/s^2)	0.059	0.0278	0.0299
Acceleration$_{rms}$ (m/s^2)	0.0115	0.0043	0.0044
Acceleration @ 23 Hz (m/s^2)	0.0118	9.36E-4	5.77E-5
Acceleration @ 46 Hz (m/s^2)	9.49E-3	5.76E-4	1.46E-4
Acceleration @ 69 Hz (m/s^2)	4.06E-3	1.42E-4	1.79E-4

Table 7.9: Summary of measured base acceleration for a 23 Hz, 4.8 mm pk-pk, 6.6 g 6 harmonic fast tool servo trajectory with no acceleration feedback, manually tuned feedforward, and a three harmonic AFC control.

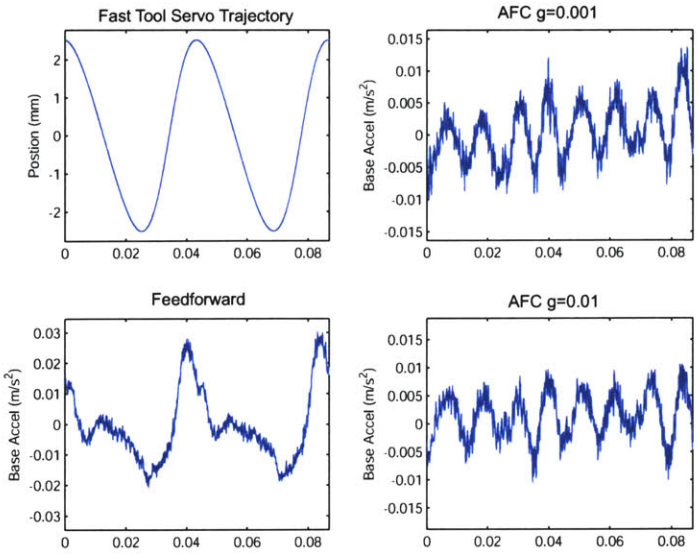


Figure 7-21: Measured base acceleration with no feedback, feedforward, and a three resonator AFC controller. The fast tool servo trajectory is a six harmonic 23 Hz 4.8 mm pk-pk sinusoid.

are too low the utility of AFC control is limited while if the gains are too high we develop undesirable dynamics as seen in the fast tool servo AFC section.

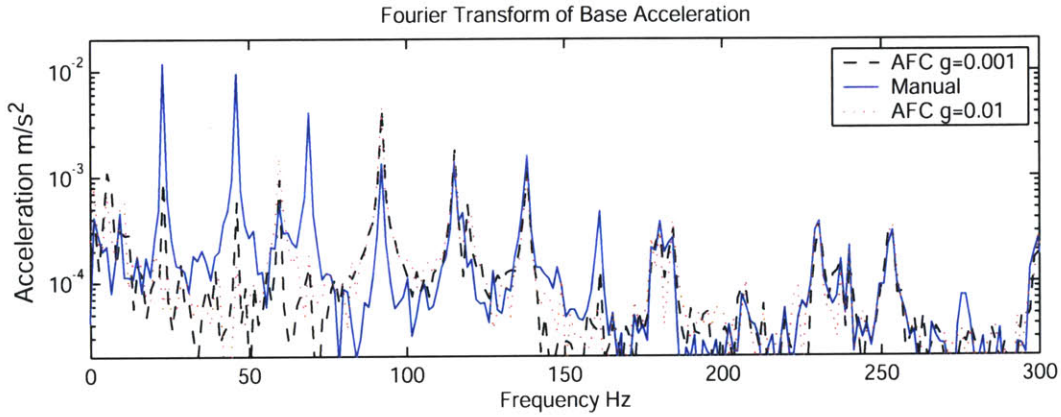


Figure 7-22: Fourier transform of base acceleration with no feedback, feedforward, and a three resonator AFC controller. The fast tool servo trajectory is a six harmonic 23 Hz 4.8 mm pk-pk sinusoid.

7.4 Base Acceleration Feedback AMAFC

In this section, we present our results for applying AMAFC control to the base acceleration feedback. We evaluated the effect of base AMAFC control on a single harmonic fast tool servo trajectory of the form

$$z_d(t) = (1 + a_0 \sin \alpha t) \sin(w_1 t) \tag{7.12}$$

where $\omega_1 = 40\pi$ (20 Hz), $a_0 = 0.25$, and $\alpha = \pi$, 2π, 4π, 6π (0.5, 1, 2, and 3 Hz). Table 7.10 summarizes the measured base acceleration. Figure 7-23 shows the measured base acceleration versus time. Figure 7-24 plots the Fourier transform of the measured base acceleration for the system under both AFC and AMAFC control.

As we can see from Table 7.10, the AMAFC peak to peak base acceleration is 2/3 the AFC peak to peak acceleration for all of the modulation frequencies. Similarly, we see that AMAFC cuts the rms base acceleration in half for all cases. As we noted in the previous section, the base acceleration measurement is quite noisy and the peak to peak acceleration and the rms acceleration are of limited utility. Examining Figure 7-24 we see that for $\alpha = 0.5$ and 1 Hz, AMAFC control reduces the frequency components near 20 Hz by a factor of 10. For $\alpha = 2$ and 3 Hz the convolution of the modulation signal and fundamental trajectory harmonic have caused the measured base acceleration to move to $20 \pm \alpha$ Hz. As expected from our analysis in chapter 4, AMAFC becomes less effective as the speed of the modulation is increased. In this case when $\alpha = 3$ Hz, AMAFC offers only a factor of 5 advantage canceling the controlled base acceleration components.

7.5 Summary

In this Chapter, we presented the measured FTS following error under several different AFC controller implementations. The measured following error was typically 2 μm peak-to-peak and 0.2 μm rms for moderate acceleration trajectories. When using the multiplicative AMAFC structure, AMAFC offered a small advantage reducing

	AFC		AMAFC	
α	pk-pk (m/s^2)	rms (m/s^2)	pk-pk (m/s^2)	rms (m/s^2)
0 Hz	0.017	1.81E-3	—	—
0.5 Hz	0.030	5.1E-3	0.022	2.83E-3
1 Hz	0.029	3.67E-3	0.019	1.81E-3
2 Hz	0.034	2.97E-3	0.017	1.26E-3
3 Hz	0.026	3.14E-3	0.016	1.63E-3

Table 7.10: Summary of measured base acceleration for a 20 Hz 2 mm pk-pk single harmonic fast tool servo trajectory (1.6 g's) with modulated magnitude for AFC and AMAFC control.

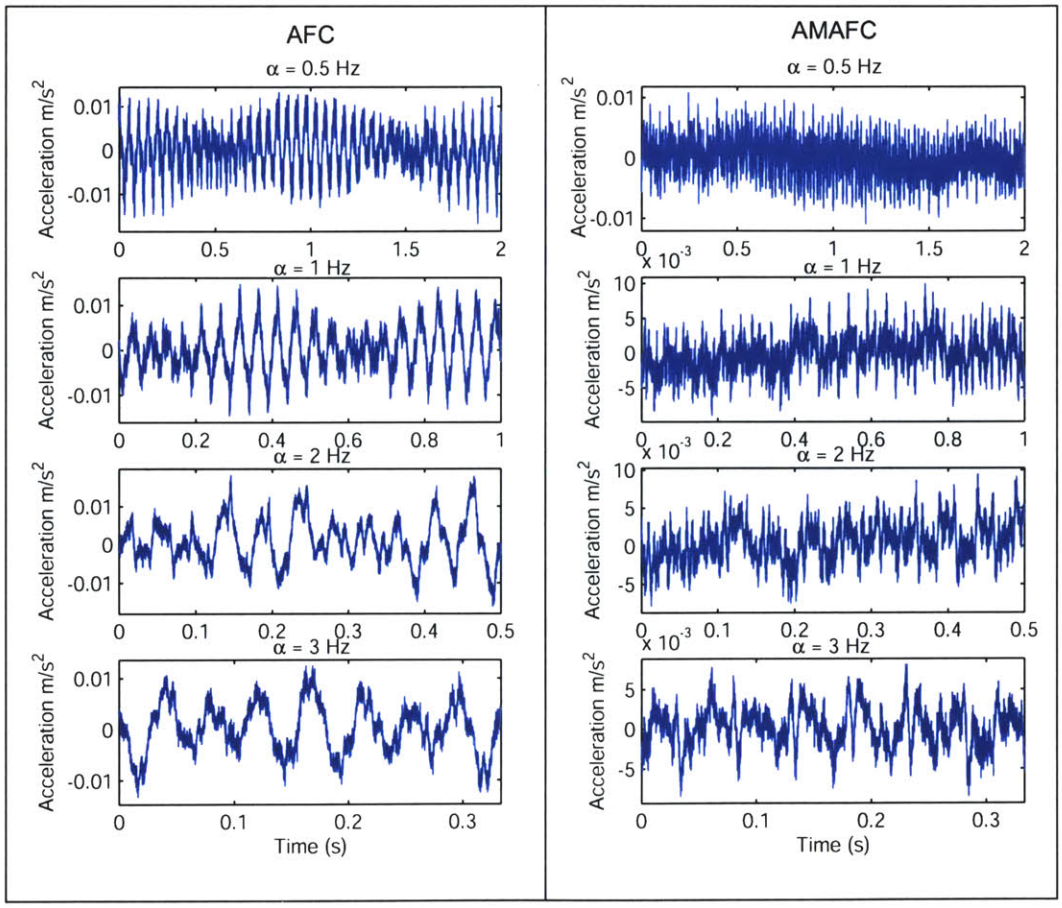


Figure 7-23: Measured base acceleration for a modulated 20 Hz fast tool servo trajectory for AFC (left plots) and AMAFC (right plots) control.

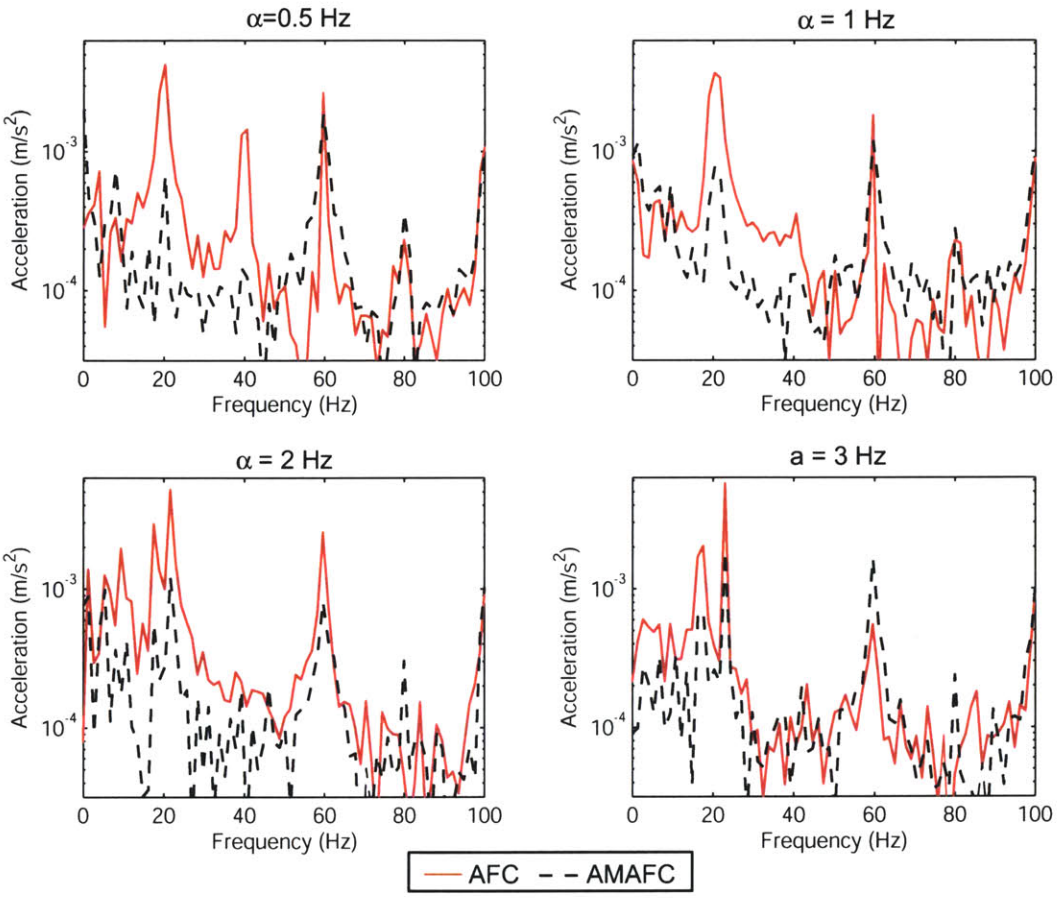


Figure 7-24: Fourier transform of measured base acceleration for a modulated 20 Hz fast tool servo trajectory.

the following error at the frequency of the trajectory harmonic by a factor of two. Applying, AFC control to eliminate machine base acceleration reduced the overall measured acceleration by a factor of 10 and the component at the FTS trajectory by a factor of 100. Applying AMAFC to the base acceleration feedback offers a factor of 3 advantage canceling FTS reaction forces with time varying amplitude. In the next Chapter, we conclude this thesis with with summary of the thesis contributions and suggestions for further work.

Chapter 8

Conclusion and Suggestions for Future Work

8.1 Summary of Results

The primary results of this thesis are:

- The design and construction of a 25 mm travel linear fast tool servo.
  - 100 m/s^2 (10 g's) maximum acceleration with 6.6 g's demonstrated.
  - Demonstrated ± 1.5 μm following error with MicroE scale.
  - Demonstrated ± 0.1 μm following error with Heidenhain scale.
- Assembly and control of the linear fast tool servo on a machine base with integral balance mass.
  - A factor of three measured reduction in base acceleration using classic feedforward control.
  - A factor of 10 reduction in overall measured base acceleration using base acceleration feedback with adaptive feedforward control.
  - A factor of 100 reduction in the measured base acceleration at the adaptive feedforward controller frequency.

- Demonstration in simulation that the Oscillator Amplitude Control perspective correctly predicts the convergence properties of both single and multiple resonator AFC control systems.

- Demonstration in experiment of a multiplicative amplitude modulated AFC (AMAFC) controller with enhanced performance following trajectories with time varying magnitude.

  - A factor of 5 reduction in the measure FTS following error at 20 Hz for a 20 Hz sinusoid with a 0.5 Hz modulation.

  - A factor of 10 reduction in the measured base acceleration at 20 Hz for a 20 Hz sinusoid FTS trajectory with a 0.5 Hz modulation.

- Demonstration in simulation of an additive AMAFC structure with significant improvements in both trajectory following and numerical stability versus the multiplicative AMAFC structure.

8.2 Future Work

While our linear fast tool servo prototype worked quite well as built, there remains significant development to be done to achieve the original design goals of 500 m/s^2 maximum acceleration and 0.1 μm form accuracy. Outlined below are several suggestion for future work to reach the ultimate design goal.

8.2.1 Higher Accuracy Sensor

It is quite clear that the performance of our prototype FTS was limited by the quality of the sensor. There are essentially four ways to improve the sensor accuracy. First is to map the existing MicroE scale. This could be done using an interferometer while the FTS is mounted on an optical table. Using the scale mapping, a lookup table could be added to the control structure to correct for the know position error. This approach has the advantage of using only existing hardware but it is unknown how

well the DS1103 controller will accommodate the large lookup file required for this approach. A second option is to build a serial interface for the Heidenhain LIP501 encoder. This approach offers the advantage of enhanced accuracy while utilizing existing hardware and software. The disadvantage of this approach is that the maximum speed of the Heidenhain electronics is on the low side of the requirements. A third option is to investigate if there are other scales and electronics which would meet our requirements. A brief review of the current Heidenhain and MircoE product lines indicate that the dual read head scales from MicroE may be applicable. For the dual read head scales, two read heads make measurements off of the same scale and the position output is the average of the two values. MicroE currently offers only a rotary version of this encoder but their literature suggest that the hardware can also be applied to linear systems. This system offers enhanced accuracy but at the expense of adding a second serial interface. Since the serial interface is one on the major items limiting our overall sample rate, it is likely the addition of a second serial interface would make the overall control loop unacceptably slow. The last option is to implement a laser interferometer based measurement system. This approach offers enhanced accuracy and allows the position of the FTS to be measured relative to the absolute machine reference frame. As mentioned earlier there are several disadvantages to this approach specifically cost and beam routing. The current FTS hardware, specifically the linear motor, do not support this approach, since they would block the beam path.

8.2.2 Actuator

The second major component limiting the performance of the FTS system is the actuator. While Micheal Liebman's oil-cooled linear motor has proven more than adequate for proving the validity of the overall control structure, it is incapable of meeting the 50 g requirement. In chapter 2, we reviewed several proposed voice coil motor designs. As mentioned there, any proposed actuator is going to require both a flux concentrating design and coil cooling. It is my belief that a single phase motor with a moving coil is the design option of choice since the magnetic fields

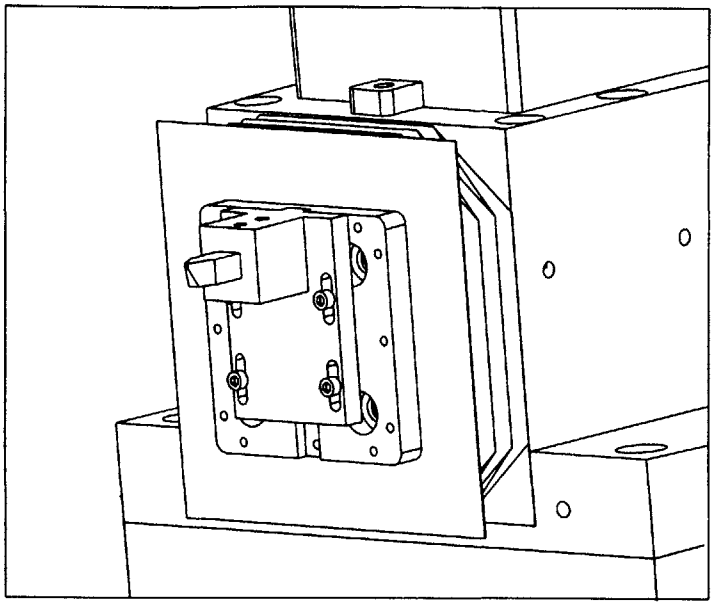


Figure 8-1: Pro-E model of a proposed toolholder/bearing seal assembly.

required in a moving magnet design have the potential to demagnetize the permanent magnets. Specifically, if the magnets were to experience a significant temperature rise, demagnetization of the magnets is a significant risk. It should be possible to enhance the performance of any motor with forced cooling by refrigerating the coolant. While this offers some enhanced thermal performance, it may introduce unwanted thermal distortion in the overall system. It should be noted that any further testing with the existing prototype will require a redesign of the coolant circuit since the existing circuit failed due to pump overload.

8.2.3 Cutting Studies

One of the clear flaws to the results presented in this thesis is that they were all acquired during air cuts. Air cuts represent the best conditions for FTS operation, minimizing disturbance and actuation forces. The performance of both the FTS position control loop and the base acceleration feedback loop may be significantly degraded when parts are being cut. Of particular concern is the base acceleration feedback loop when the base is being excited by both the FTS/reaction mass sub-

system and the workpiece/spindle subsystem. The FTS requires several additional components before being able to support cutting operations:

- a tool holder
- a seal for the air bearing to prevent contamination.

Figure 8-1 shows a Pro-E model of one possible toolholder/air bearing seal assembly. In this figure, the toolholder is a basic design utilizing a standard 1/4" tool shank, a slotted sliding mechanism which is oriented horizontally and in roll by a slot in the toolholder mounting face. The sliding mechanism is bolted to the mounting face utilizing four 8-40 bolts. The air bearing is sealed utilizing a 4"x4" square polyethylene bellows. The bellows are attached to both the toolholder mounting face and the air bearing bushing assembly. All of the components for this toolholder have been fabricated except for the tool holder sliding mechanism and mounting face. This design is acceptable for rough cutting but since it does not incorporate a fine height adjustment in is not acceptable for fine cutting. A more appropriate toolholder would incorporate features for coarse and fine adjustment. It is likely a flexure could be utilized for fine adjustment.

8.2.4 Investigate Effect of Integrator Saturation of AFC Control

One of the major theoretical evaluations that has yet to be performed on AFC controllers is the effect that integrator saturation has on the performance and stability of an AFC controller. As is common practice when utilizing integrators, we have limited the integrator windup by placing saturation limits on the state of the integrator. This prevents the integrators from developing large overshoot but can have a negative impact on the system performance. Specifically if the integrator saturation limit is set too low, we may artificially prevent the AFC controller from properly estimating the disturbance magnitude. This occurred on at least one occasion for the base acceleration feedback controller. Alternately, the integrator saturation limits may

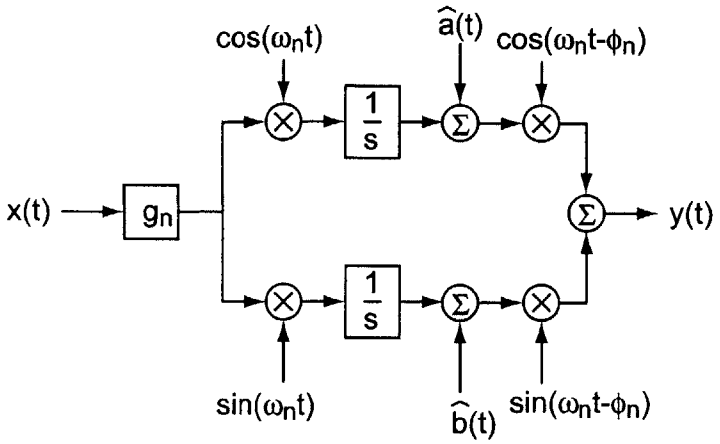


Figure 8-2: Block diagram of a standard AFC resonator and plant with zero input into the AFC system.

combine to form a *conditionally stable* system [34]. In a conditionally stable system the magnitude of the loop response determines if the system is stable or unstable. For the most common situation, if the magnitude of the loop response is small the system is unstable while if the magnitude is large the response is stable. Thus the system enters a limit cycle oscillation. On more than one occasion the FTS controller entered a marginally stable condition where the FTS motion contained frequencies other than those commanded. It is unclear if these oscillations were a result of conditional stability and this bears further study.

8.2.5 Additive AMAFC

There are several incomplete components to the additive AMAFC analysis. While it is clear in simulation that the additive structure has superior tracking and numerical performance when compared with multiplicative AMAFC control, this has not been proven experimentally. Results should be gathered comparing the performance of additive AMAFC control to classic AFC control and AFC control with command pre-shifting. In Chapter 4, we suggested that additive AMAFC control appeared to integrate the command pre-shifting channel with the AFC controller and that this would be a more compact and numerically efficient control model. A more detailed

analysis of this should be conducted comparing the turnaround time of both the additive AMAFC structure and AFC with command pre-shifting to verify this hypothesis.

Lastly in Chapter 4, we proposed that for the additive AMAFC structure shown in Figure 8-2, for a desired trajectory of

$$y_d(t) = \cos \alpha t[a_n \cos(\omega_n t) + b_n \sin(\omega_n t)] \tag{8.1}$$

that the additive components $\hat{a}(t)$ and $\hat{b}(t)$ should be

$$\hat{a}(t) = \frac{a_n}{|P(j\omega_n)|} \cos \alpha t \tag{8.2}$$

$$\hat{b}(t) = \frac{b_n}{|P(j\omega_n)|} \cos \alpha t. \tag{8.3}$$

As we saw in Chapter 4, this does not result in perfect error tracking since it does not account for the phase shift in the α sinusoid from the plant dynamics. We propose that an improved estimate would be

$$\hat{a}(t) = \frac{a_n}{|P(j\omega_n)|} \cos(\alpha t + \delta) \tag{8.4}$$

$$\hat{b}(t) = \frac{b_n}{|P(j\omega_n)|} \cos \alpha t + \delta) \tag{8.5}$$

where δ can be determined from a Fourier analysis similar to that used while evaluating the limitations of the OAC approximation in Chapter 5.

8.3 Conclusions

The main results of this thesis have been the design and development of a linear long stroke fast tool servo with an integral balance mass and the development of a loop shaping technique for tuning systems with AFC control. While the performance of the linear FTS did not meet our desired goals, the long travel tracking performance of the FTS matches or exceeds the long travel tracking performance of comparable FTS devices. The unique feature of this design is the incorporation of an integral balance

mass. This design successfully allowed for a significant attenuation in the reaction forces in the machine base without any apparent loss of FTS performance. Our loop shaping approach to AFC control was successfully applied to both the FTS position control loop and the base acceleration feedback loop. In summary, the success of both the FTS design and our control approach indicate the general applicability of this approach to the wider precision manufacturing community.

Appendix A

Tables

A.1 State Space Model of Rotary Fast Tool Servo

$$A=\begin{bmatrix}
-1632.2 & 753.98 & 753.98 & 0 & 0 & 0 & 0 & 0 & 2530 & 0 & 0 & 0 & 0\\
0 & 0 & 188.5 & 0 & 0 & 0 & 0 & 0 & 632.5 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 632.5 & 0 & 0 & 0 & 0\\
-2.69e6 & 1.87e6 & 1.87e6 & -498.1 & -13744 & -557 & -4710 & 0 & 6.26e6 & 15000 & 0 & 6985 & -1632\\
0 & 0 & 0 & 16384 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 8192 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 8192 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 64 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 32 & 0 & 0 & 0 & 0 & 0\\
4.31e7 & -2.99e7 & -2.99e7 & 0 & 0 & 0 & 0 & 0 & -1.0e8 & -1.2e5 & -9.16e5 & -5.59e5 & 2.61e5\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 6.55e5 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3.28e5 & 0 & 0\\
-1.80e6 & 1.24e6 & 1.24e6 & 0 & 0 & 0 & 0 & 0 & 4.17e6 & 0 & 0 & 0 & -1632
\end{bmatrix}$$

$$B=\begin{bmatrix}0\\0\\0\\-32\\0\\0\\0\\0\\0\\512\\0\\0\\0\end{bmatrix}$$

$$C=\begin{bmatrix}-2.81e5 & 1.94e5 & 1.94e5 & 0 & 0 & 0 & 0 & 0 & 6.52e5 & 0 & 0 & 0 & 0\end{bmatrix}$$

$$D=\begin{bmatrix}0\end{bmatrix}$$

A.2 AFC resonator Values

Resonator Tuning Values										
n	1	2	3	4	5	6	7	8	9	10
ω_n	125.7	251.3	377	502.7	628.3	754.0	879.7	1005	1131	1256
g_n	1	1	1	1	1	1	1	1	1	1
ϕ_n(deg)	-7.7	-11.2	-17.4	-30.7	-50.3	-70.7	-87.4	-100.3	-110.8	-119.7

Table A.1: Resonator tuning values for a 10 resonator system with $g_n = 1$ and .

Resonator Tuning Values										
n	1	2	3	4	5	6	7	8	9	10
ω_n	125.7	251.3	377	502.7	628.3	754.0	879.7	1005	1131	1256
g_n	5.18	5.18	5.18	5.18	5.18	5.18	5.18	5.18	5.18	5.18
ϕ_n(deg)	-7.7	-11.2	-17.4	-30.7	-50.3	-70.7	-87.4	-100.3	-110.8	-119.7

Table A.2: Resonator tuning values for a 10 resonator system with $g_1 = ... = g_n$ and .

Resonator Tuning Values										
n	1	2	3	4	5	6	7	8	9	10
ω_n	125.7	251.3	377	502.7	628.3	754.0	879.7	1005	1131	1256
g_n	31.1	20.7	5.1	2.59	2.59	2.59	5.18	5.18	5.18	5.18
ϕ_n(deg)	-7.7	-11.2	-17.4	-30.7	-50.3	-70.7	-87.4	-100.3	-110.8	-119.7

Table A.3: Resonator tuning values for a 10 resonator system with g_n modified by hand and .

Appendix B

Matlab Code for Voice Coil Motor Design

B.1 Cylindrical Motor Function

This section contains the Matlab code for the function *motor_size* which analyzes cylindrical voice coil motor designs.

```
function motor_size
clear all; close all;
disp(['This program computes the performance parameters for a ...
 circular VCM'])
tm=input('Magnet thickness (cm)')/100;
tg=input('Thickness of airgap (cm)')/100;
Rm=input('Magnet radius (cm)')/100;
Rg=input('Airgap Radius (cm)')/100;
Lm=input('Length of magnet (cm)')/100;
Lg=input('Length of Airgap (cm)')/100;
Br=input('Magnet remenence (T)');
Lcu=input('Coil Width (cm)')/100;
rhocu=8900; rhoal=2700; PF=0.7; muz=pi*4E-7;
disp(['Density of copper ' num2str(rhocu) ' kg/m^3'])
disp(['Density of AL ' num2str(rhoal) ' kg/m^3'])
disp(['Packing Factor ' num2str(100*PF) '%'])
%Mass and volume of copper
```

```
Vcu=2*PF*pi*Rg*tg*Lcu; disp(['Volume of Copper ' num2str(Vcu) ' m^3'])
Mcu=rhocu*Vcu; disp(['Mass Copper ' num2str(Mcu) '  kg'])
%Flux density in Airgap
Bg=Br/((Rg*Lg)/((Rm-tm/2)*Lm)+tg/tm);
disp(['Magnetic field in Gap ' num2str(Bg) ' T'])
%Flux density in Backiron
Bbi=Bg*2*pi*Rg*Lg/(pi*(Rg-tg/2)^2);
disp(['Magnetic field in Backiron ' num2str(Bbi) ' T'])
if Bbi>1.8, disp(['Backiron is Saturated']), end
tbi=Bg*2*pi*Rg*Lg/(2*pi*1.8*(Rg-tg/2));
Rbi=sqrt((Bg*2*pi*Rg*Lg/1.8+pi*(Rm+tm/2)^2)/pi);
%Mass and volume of Al
Lal=(Lg-Lcu)/2+1.27/100;
Val=pi*tg*Rg^2+pi*(Rg^2-(Rg-tg/2)^2)*Lal;
Mal=rhoal*Val; disp(['Mass AL ' num2str(Mal) ' kg'])
%Moving mass, slide mass = 0.2 kg Al, 0.6 St
Mm1=0.2+Mcu+Mal; Mm2=0.6+Mcu+Mal;
%Required Current Density A/mm^2
F1=500*Mm1; F2=500*Mm2;
J1=F1/(Bg*Vcu*1000^2); J2=F2/(Bg*Vcu*1000^2);
disp(['Current Density for Al actuator ' num2str(J1) ' A/mm^2'])
disp(['Current Density for ST actuator ' num2str(J2) ' A/mm^2'])
close all;
figure(1)
x=[-Lm/2 Lm/2 Lm/2 -Lm/2]*100;
y=[Rm-tm/2 Rm-tm/2 Rm+tm/2 Rm+tm/2]*100;
fill(x,y,'r',x,-y,'r')
hold on;
x=[-Lm/2 Lm/2 Lg/2 -Lg/2]*100;
y=[Rm-tm/2 Rm-tm/2 Rg+tg/2 Rg+tg/2]*100;
fill(x,y,'b',x,-y,'b')
x=[-Lcu/2 Lcu/2 Lcu/2 -Lcu/2]*100;
y=[Rg+tg/2 Rg+tg/2 Rg-tg/2 Rg-tg/2]*100;
fill(x,y,'m',x,-y,'m')
x=[-Lg/2 Lm/2 Lm/2 -Lm/2 -Lm/2 Lm/2+tbi Lm/2+tbi -Lg/2]*100;
y=[Rg-tg/2 Rg-tg/2 Rm+tm/2 Rm+tm/2 Rbi Rbi 0 0]*100;
```

```
fill(x,y,'b',x,-y,'b')
x=[-(Lcu)/2 -Lg/2-0.0127-tg -Lg/2-tg-0.0127 -Lg/2-0.0127 ...
      -Lg/2-0.0127 -(Lcu)/2]*100;
y=[Rg Rg 0 0 Rg-tg/2 Rg-tg/2]*100;
fill(x,y,'g',x,-y,'g')
x=[-Lg/2-0.0127-tg -Lg/2-0.0127-tg-7.75*2.54/100 ...
        -Lg/2-0.0127-tg-7.75*2.54/100 -Lg/2-0.0127-tg]*100;
y=[2.54 2.54 -2.54 -2.54]/2;
fill(x,y,'g')
x=[-Lg/2-2*0.0127-tg -Lg/2-6*0.0127-tg -Lg/2-6*0.0127-tg ...
         -Lg/2-2*0.0127-tg]*100;
y=[2.54 2.54 2.5*2.54 2.5*2.54]/2;
fill(x,y,'k',x,-y,'k')
x=[-Lg/2-6*0.0127-tg-(2.54*2.25/100) -Lg/2-10*0.0127-tg-(2.54*2.25/100)...
   -Lg/2-10*0.0127-tg-(2.54*2.25/100) -Lg/2-6*0.0127-tg-(2.54*2.25/100)]*100;
y=[2.54 2.54 2.5*2.54 2.5*2.54]/2;
fill(x,y,'k',x,-y,'k')
 axis equal
 lab=['Mass Cu ' num2str(Mcu) ' kg'];
 text(-20,-4,lab)
 lab=['Moving Mass (Al) ' num2str(Mm1) ' kg'];
 text(-20, -5,lab)
 lab=['Moving Mass (St) ' num2str(Mm2) ' kg'];
 text(-20, -6,lab)
 lab=['Flux Desity Air Gap (Bg) ' num2str(Bg) ' T'];
 text(-20, -7,lab)
 lab=['Flux Desity backiron (Bbi) ' num2str(Bbi) ' T'];
 text(-20, -8,lab)
 lab=['Current Density (Al) ' num2str(J1) ' A/mm^2'];
 text(-20, -9,lab)
 lab=['Current Density (St) ' num2str(J2) ' A/mm^2'];
 text(-20, -10,lab)
 lab=['Magnet Radius ' num2str(Rm*100) ' cm'];
 text(-20,4,lab)
 lab=['Airgap Radius ' num2str(Rg*100) ' cm'];
 text(-20,5,lab)
```

```
 lab=['Magnet Length ' num2str(Lm*100) ' cm'];
 text(-20,6,lab)
 lab=['Airgap Length ' num2str(Lg*100) ' cm'];
 text(-20,7,lab)
 lab=['Coil width ' num2str(Lcu*100) ' cm'];
 text(-20,8,lab)
 grid
hold off;
```

B.2 Square Motor Function

This section contains the Matlab code for the function *motor_size_s* which analyzes square voice coil motor designs.

```
function motor_size_s
clear all; close all;
trm=0.0254; disp(['Motor travel ' num2str(trm*100) ' cm'])
Br=1.2; disp(['Magnet Remenance ' num2str(Br) ' T'])
rhocu=8900; rhoal=2700; PF=0.7; muz=pi*4E-7;
disp(['Density of copper ' num2str(rhocu) ' kg/m^3'])
disp(['Density of AL ' num2str(rhoal) ' kg/m^3'])
disp(['Packing Factor ' num2str(100*PF) '%'])
tm=input('Magnet thickness cm')/100;
tg=input('Gap thickness cm')/100;
tfc=input('Thickness of concentrator cm')/100;
Lm=input('Magnet Length cm')/100;
Lg=input('Gap Length cm')/100;
Wm=input('Magnet Width cm')/100;
Wg=input('Gap Width cm')/100;
%Length of Copper
Lcu=Lg-trm;
%Flux density in air gap
Bg=Br/((Lg*Wg)/(Lm*Wm)+tg/tm);
disp(['Flux density in air gap ' num2str(Bg) ' T']);
%Flux density in backiron
Bbi=4*Bg*Lg/Wg;
```

```
disp(['Flux density in backiron ' num2str(Bbi) ' T']);
%Volume of copper
Vcu1=4*Wg*Lcu*tg*PF; Vcu2=4*tg^2*Lcu*PF; Vt=Vcu1+Vcu2;
%mass copper
Mcu=Vt*rhocu;
%Volume Al
Val=trm*(2*tg*Wg +tg^2)+(Wg+tg)^2*tg;
%Mass Al
Mal=Val*rhoal;
%Total mass M1=0.2 M2=0.6
M1=0.2+Mcu+Mal; M2=0.6+Mcu+Mal;
%Force 500 m/s2
F1=M1*500; F2=M2*500;
%Current Density
J1=F1/(Vcu1*Bg*1000^2);
J2=F2/(Vcu1*Bg*1000^2);
figure(1)
subplot(2,1,1)
hold on;
base=Wg/2+tg+tfc;
x=[-Lm/2 Lm/2 Lm/2 -Lm/2]*100;
y=[base base base+tm base+tm]*100;
fill(x,y,'r',x,-y,'r')
x=[-Lm/2 Lm/2 Lg/2 -Lg/2]*100;
y=[base base base-tfc base-tfc]*100;
fill(x,y,'b',x,-y,'b')
base=base-tfc;
x=[-Lcu/2 Lcu/2 Lcu/2 -Lcu/2]*100;
y=[base base base-tg base-tg]*100;
fill(x,y,'m',x,-y,'m')
base=base-tg/2;
x=[-Lcu/2 -(Lcu/2+tg+trm) -(Lcu/2+tg+trm) -(Lcu/2+trm) -(Lcu/2+trm)...
        -(Lcu/2)]*100;
y=[base base 0 0 base-tg/2 base-tg/2]*100;
fill(x,y,'g',x,-y,'g')
base=base-tg/2;
```

```
x=[-Lg/2 Lm/2 Lm/2 -Lm/2 -Lm/2 Lm/2+2*tm Lm/2+2*tm -Lg/2]*100;
y=[base base base+tg+tm+tfc base+tg+tm+tfc base+tg+3*tm+tfc...
        base+tg+3*tm+tfc 0 0]*100;
fill(x,y,'b',x,-y,'b')
basex=-(Lcu/2+tg+trm);
sl=-7.75*2.54/100;
x=[basex basex+sl basex+sl basex]*100;
y=[0.0254/2 0.0254/2 -0.0254/2 -0.0254/2]*100;
fill(x,y,'g')
 lab=['Flux Desity Air Gap (Bg) ' num2str(Bg) ' T'];
 text(-20, -3,lab)
 lab=['Flux Desity backiron (Bbi) ' num2str(Bbi) ' T'];
 text(-20, -5,lab)
 lab=['Current Density (Al) ' num2str(J1) ' A/mm^2'];
 text(-20, -7,lab)
 lab=['Current Density (St) ' num2str(J2) ' A/mm^2'];
 text(-20, -9,lab)
hold off;
subplot(2,1,2)
hold on;
basex=Wg/2+tg+tfc+3*tm;
x=[-basex basex basex -basex]*100;
y=[basex basex -basex -basex]*100;
fill(x,y,'b')
basex=basex-2*tm;
x=[-basex basex basex -basex]*100;
y=[basex basex -basex -basex]*100;
fill(x,y,'w')
x=[-basex -basex -basex+tm -basex+tm]*100;
y=[Wm/2 -Wm/2 -Wm/2 Wm/2]*100;
fill(x,y,'r',-x,y,'r')
axis equal
x=[Wm/2 -Wm/2 -Wm/2 Wm/2]*100;
y=[-basex -basex -basex+tm -basex+tm]*100;
fill(x,y,'r',x,-y,'r')
basex=basex-tm;
```

```
x=[basex basex basex-tfc basex-tfc]*100;
y=[Wm/2 -Wm/2 -Wg/2 Wg/2]*100;
fill(x,y,'b',-x,y,'b')
x=[Wm/2 -Wm/2 -Wg/2 Wg/2]*100;
y=[basex basex basex-tfc basex-tfc]*100;
fill(x,y,'b',x,-y,'b')
basex=basex-tfc;
x=[-basex basex basex -basex]*100;
y=[basex basex -basex -basex]*100;
fill(x,y,'m')
basex=Wg/2;
x=[basex basex -basex -basex]*100;
y=[basex -basex -basex basex]*100;
fill(x,y,'g')
lab=['Magnet Width ' num2str(Wm*100) ' cm'];
 text(15,10,lab)
 lab=['Airgap Width ' num2str(Wg*100) ' cm'];
 text(15,8,lab)
 lab=['Magnet Length ' num2str(Lm*100) ' cm'];
 text(15,6,lab)
 lab=['Airgap Length ' num2str(Lg*100) ' cm'];
 text(15,4,lab)
 lab=['Coil width ' num2str(Lcu*100) ' cm'];
 text(15,2,lab)
 lab=['Mass Cu ' num2str(Mcu) ' kg'];
 text(15,0,lab)
 lab=['Moving Mass (Al) ' num2str(M1) ' kg'];
 text(15, -2,lab)
 lab=['Moving Mass (St) ' num2str(M2) ' kg'];
 text(15, -4,lab)
axis equal
hold off
```

Appendix C

C code for SlaveDSP serial interface

This appendix contains the c code required to run the serial interface on the DS1103 SlaveDSP. There are four function required for the code to run. Cspi_microE.c is the code for a custom S-function block in the Simulink model. Fw240.c registers the users custom code in the compiler. To activate the Fw240.c function, compile the Simulink model. Start Controldesk assign the variable file. Then from the toolbar reassign the slave application using Instrumentation/Slave Application/Assign slave application. Once the slave application has been assigned recompile the Simulink model. Usrdsp.c is the c code which defines the custom functions. Usrdsp.h is the header file for the custom functions.

C.1 Code for Cspi_microE.c

```
/* File: Cspi_microE.c
* Author: Marten Byl
* Created: 8/30/04
* Based on: sfuntmpl_basic.c and Ser_AIO.c by David Otten
* Also based on examples in the DS1103 RTLib Reference July 2001
* Description: S-function to operate a clocked serial interface to
*              a microE encoder on the slave dsp using user specific
```

```
 *              commands. Based Upon Dave Otten's work with a multi-channel
 *              A to D interface on the slave DSP
 */
 /*
 * sfuntmpl_basic.c: Basic 'C' template for a level 2 S-function.
 *  -------------------------------------------------------------------------
 *  | See matlabroot/simulink/src/sfuntmpl_doc.c for a more detailed template |
 *  -------------------------------------------------------------------------
 * Copyright 1990-2000 The MathWorks, Inc.
 * $Revision: 1.24 $
 */
/*
 * You must specify the S_FUNCTION_NAME as the name of your S-function
 * (i.e. replace sfuntmpl_basic with the name of your S-function).
 */
#define S_FUNCTION_NAME  Cspi_microE #define S_FUNCTION_LEVEL 2
/*
 * Need to include simstruc.h for the definition of the SimStruct and
 * its associated macro definitions.
 */
#include "simstruc.h"
/* When compiling for RTI target, include the
brtenv.h file.*/ #ifndef MATLAB_MEX_FILE
  #include <brtenv.h>
  #include <usrdsp.h>         /* user module definitions */
  #include "slvdsp1103.h"
  #include "tic1103.h"      /* just for diagnostics */
/* variables for communication with dSPACE */
/* variables for communication with Slave DSP */
  Int16 cspi_sample_idx  = -1;                    /* command table index */
  Int16 task_id = 0;                             /* communication channel 0 */
  UInt32 slave_error;                            /* function return error code */
  UInt32 Cspi_raw[2];                              /*Serial Data*/
  UInt32 posit;                                    /*Position count*/
#endif
/* Error handling
```

```
 * --------------
 * You should use the following technique to report errors encountered within
 * an S-function:
 *       ssSetErrorStatus(S,"Error encountered due to ...");
 *       return;
 * Note that the 2nd argument to ssSetErrorStatus must be persistent memory.
 * It cannot be a local variable. For example the following will cause
 * unpredictable errors:
 *      mdlOutputs()
 *      {
 *         char msg[256];         {ILLEGAL: to fix use "static char msg[256];"}
 *         sprintf(msg,"Error due to %s", string);
 *         ssSetErrorStatus(S,msg);
 *         return;
 *      }
 * See matlabroot/simulink/src/sfuntmpl_doc.c for more details.
 */
/*====================*
 * S-function methods *
 *====================*/
/* Function: mdlInitializeSizes
================================================
 * Abstract:
 *    The sizes information is used by Simulink to determine the S-function
 *    block's characteristics (number of inputs, outputs, states, etc.).
 */
static void mdlInitializeSizes(SimStruct *S) {
    /* See sfuntmpl_doc.c for more details on the macros below */
    ssSetNumSFcnParams(S, 0);  /* Number of expected parameters */
    if (ssGetNumSFcnParams(S) != ssGetSFcnParamsCount(S)) {
        /* Return if number of expected != number of actual parameters */
        return;
    }
    ssSetNumContStates(S, 0);
    ssSetNumDiscStates(S, 0);
    /* No input ports
```

```
    if (!ssSetNumInputPorts(S, NumADChan)) return;
    for (i = 0; i < NumADChan; i++)
      ssSetInputPortWidth(S, i, 1); */
    /* One Output port*/
    if (!ssSetNumOutputPorts(S, 1)) return;
      ssSetOutputPortWidth(S, 0, 1);
    ssSetNumSampleTimes(S, 1);
    ssSetNumRWork(S, 0);
    ssSetNumIWork(S, 1);
    ssSetNumPWork(S, 0);
    ssSetNumModes(S, 0);
    ssSetNumNonsampledZCs(S, 0);
    ssSetOptions(S, 0);
}
/* Function: mdlInitializeSampleTimes
=========================================
 * Abstract:
 *    This function is used to specify the sample time(s) for your
 *    S-function. You must register the same number of sample times as
 *    specified in ssSetNumSampleTimes.
 */
static void mdlInitializeSampleTimes(SimStruct *S) {
    ssSetSampleTime(S, 0, INHERITED_SAMPLE_TIME);
    ssSetOffsetTime(S, 0, 0);
}
#undef MDL_INITIALIZE_CONDITIONS   /* Change to #undef to remove
function */ #if defined(MDL_INITIALIZE_CONDITIONS)
  /* Function: mdlInitializeConditions ========================================
   * Abstract:
   *    In this function, you should initialize the continuous and discrete
   *    states for your S-function block.  The initial states are placed
   *    in the state vector, ssGetContStates(S) or ssGetRealDiscStates(S).
   *    You can also perform any other initialization activities that your
   *    S-function may require. Note, this routine will be called at the
   *    start of simulation and if it is present in an enabled subsystem
   *    configured to reset states, it will be call when the enabled subsystem
```

```
   *    restarts execution to reset the states.
   */
  static void mdlInitializeConditions(SimStruct *S)
  {
  }
#endif /* MDL_INITIALIZE_CONDITIONS */
#define MDL_START  /* Change to #undef to remove function */ #if
defined(MDL_START)
  /* Function: mdlStart =======================================================
   * Abstract:
   *    This function is called once at start of model execution. If you
   *    have states that should be initialized once, this is the place
   *    to do it.
   */
  static void mdlStart(SimStruct *S)
  {
  #ifndef MATLAB_MEX_FILE
    UInt32 dummy;
    dummy=0;
    /* init communication with slave_dsp */
    ds1103_slave_dsp_communication_init();
    /* initialize port B to input, port A and C to output */
    ds1103_slave_dsp_usrfct_execute(task_id, SLVDSP_USRFCT_CSPI_INIT,
      SLVDSP_USRFCT_INIT_PCNT, &dummy);
    /* register usrfct2 (_cspi_sample) */
    ds1103_slave_dsp_usrfct_register(task_id, &cspi_sample_idx,
      SLVDSP_USRFCT_CSPI_SAMPLE, SLVDSP_USRFCT_SAMPLE_DINCNT,
      SLVDSP_USRFCT_SAMPLE_DOUTCNT, SLVDSP_USRFCT_SAMPLE_PCNT, &dummy);
   /* start an initial sample on the serial port */
    ds1103_slave_dsp_usrfct_request (task_id, cspi_sample_idx, (UInt32*)dummy);
  #endif
  }
#endif /*  MDL_START */
/* Function: mdlOutputs
=========================================================
 * Abstract:
```

```
 *    In this function, you compute the outputs of your S-function
 *    block. Generally outputs are placed in the output vector, ssGetY(S).
 */
static void mdlOutputs(SimStruct *S, int_T tid) { #ifndef
MATLAB_MEX_FILE
    UInt32 i;
for(i=0;i<1;i++) {
    real_T       *y = ssGetOutputPortRealSignal(S,0);
    y[0] =ssGetIWorkValue(S,0);
} #endif }
#define MDL_UPDATE  /* Change to #undef to remove function */ #if
defined(MDL_UPDATE)
  /* Function: mdlUpdate ======================================================
   * Abstract:
   *    This function is called once for every major integration time step.
   *    Discrete states are typically updated here, but this function is useful
   *    for performing any tasks that should only take place once per
   *    integration step.
   */
  static void mdlUpdate(SimStruct *S, int_T tid)
  {
  #ifndef MATLAB_MEX_FILE
   int i;
       UInt32 dummy=0;
       ds1103_slave_dsp_usrfct_data_read(task_id,cspi_sample_idx,(UInt32*)Cspi_raw);
    posit=Cspi_raw[0]*16384+Cspi_raw[1]/4;
    if (posit>536870912) posit=posit-1073741824;
    ssSetIWorkValue(S,0,posit);
   /*Request data Serial port data*/
    ds1103_slave_dsp_usrfct_request(task_id, cspi_sample_idx, (UInt32*)dummy);
    /*Read in the Serial port data*/
  #endif
  }
#endif /* MDL_UPDATE */
#undef MDL_DERIVATIVES  /* Change to #undef to remove function */
#if defined(MDL_DERIVATIVES)
```

```
  /* Function: mdlDerivatives ================================================
   * Abstract:
   *    In this function, you compute the S-function block's derivatives.
   *    The derivatives are placed in the derivative vector, ssGetdX(S).
   */
  static void mdlDerivatives(SimStruct *S)
  {
  }
#endif /* MDL_DERIVATIVES */
/* Function: mdlTerminate
=====================================================
 * Abstract:
 *    In this function, you should perform any actions that are necessary
 *    at the termination of a simulation.  For example, if memory was
 *    allocated in mdlStart, this is the place to free it.
 */
static void mdlTerminate(SimStruct *S) { }
/*=============================================================*
 * See sfuntmpl_doc.c for the optional S-function methods *
 *=============================================================*/
/*=============================*
 * Required S-function trailer *
 *=============================*/
#ifdef  MATLAB_MEX_FILE    /* Is this file being compiled as a
MEX-file? */ #include "simulink.c"      /* MEX-file interface
mechanism */ #else #include "cg_sfun.h"       /* Code generation
registration function */ #endif
```

C.2 Code for Fw240.c

```
/*****************************************************************************
* PROGRAM
*   Firmware program for TMS320C240 Slave DSP on DS1103 Controller Board
* FILE
*   Fw240.c
* RELATED FILES
```

```
*   - dsscom.h           => embedded in ds240.lib
*   - slvdsp.h           => embedded in ds240.lib
*   - usrdsp.h           => user specific module
* DESCRIPTION
*   This program performs the slave DSP command service. First of all the
*   communication with the master PPC is initialized.
*   User defined functions are installed in the slave DSP function vector table.
*   Finally the dsscom_cmd_service routine is called repeatedly to poll
*   the slave DSP receive buffers until a command is detected. As a result the
*   appropriate function is called.
* REMARKS
* AUTHOR(S)
*   H.-J. Miks, M. Heier
*   REVISED
*   Marten Byl  8/30/04
* dSPACE GmbH, Technologiepark 25, 33100 Paderborn, Germany
* $Workfile: Fw240.c $ $Revision: 7 $ $Date: 10.08.00 10:49 $
* $Archive: /sw/CRT/Projects/Demos/Src/RTLib1103/SlaveDsp/Slv_User_1103_hc/Fw240.c $
**************************************************************************/
#include <dsscom.h>
#include <slvdsp.h>
#include <usrdsp.h>
/**************************************************************************
* main routine
**************************************************************************/
void main()
{
  Int16 error;
  /* set user firmware revision number */
  SLVDSP_USR_FIRMWARE_REV_SET;
  /* initialization of master-slave communication */
  error = slvdsp_communication_init();
  if (error != DSCOMDEF_NO_ERROR)
  {
    exit(0);
  }
```

```
  /* installation of user function 1 (_usrfct_cspi_init) */
  error = slvdsp_usrfct_install(SLVDSP_USRFCT_CSPI_INIT,
    slvdsp_usrfct_cspi_init);
  if (error != DSCOMDEF_NO_ERROR)
  {
    exit(0);
  }
  /* installation of user function 2 (_usrfct_cspi_sample) */
  error = slvdsp_usrfct_install(SLVDSP_USRFCT_CSPI_SAMPLE,
    slvdsp_usrfct_cspi_sample);
  if (error != DSCOMDEF_NO_ERROR)
  {
    exit(0);
  }
  /* perform slave DSP command service */
  while (1)
  {
    dsscom_cmd_service();
  }
}
```

C.3 Code for Usrdsp.c

```
/*****************************************************************************
*
* MODULE
*   Common TMS320F240 slave DSP user module.
* FILE
*   usrdsp.c
* RELATED FILES
*   usrdsp.h, reg240.h, fw240.c
* DESCRIPTION
*   This module may contain user defined functions for the TMS320F240
*   slave DSP.
*   Each additional function must be registered in the fw240.c module by
*   using the function slvdsp_usrfct_install().
```

```
* AUTHOR(S)
*   M. Heier
* dSPACE GmbH, Technologiepark 25, 33100 Paderborn, Germany *
*   REVISED
*   Marten Byl 8/30/04
*   This file defines two slavedsp user functions
*       slvdsp_usrfct_cspi_init
*       slvdsp_usrfct_cspi_sample
*   which initialize and then run a clocked serial interface
*   on the slave dsp. The data is transfered to the main dsp in
*   the sample operations
* $Workfile: Usrdsp.c $ $Revision: 6 $ $Date: 9.08.00 16:33 $
* $Archive: /sw/CRT/Projects/Demos/Src/RTLib1103/SlaveDsp/Slv_User_1103_hc/Usrdsp.c $
*****************************************************************************/
#include <dstypes.h>                                /* data type
definitions */ #include <reg240.h>                          /*
TMS320C240 register defines */ #include <usrdsp.h>
/*****************************************************************************
* global objects and variables
*****************************************************************************/
/*****************************************************************************
* user functions
*****************************************************************************/
/*****************************************************************************
* FUNCTION
*   Initialize the hardware ports for the Analog I/O Subsystem
* SYNTAX
*   slvdsp_usrfct_cspi_init(UInt16 index)
* DESCRIPTION
*   All available bits in ports A, B, and C are selected as digital I/O pins.
*   All bits in port B are set to input, all bits in ports A and C are set to ouput.
* PARAMETERS
*   index               command table index
* RETURNS *
* REMARKS
*****************************************************************************/
```

```
void slvdsp_usrfct_cspi_init(UInt16 index) {
  /* init the I/O MUX control registers to I/O */
  *OCRA = 0x0000; /*sets all pins I/O*/
  *OCRB = 0x000C; /*sets all pins to I/O PC*/
  /* init the I/O port data and direction registers */
  *PADATDIR = 0xFF00; /*Sets Group A to Output*/
  *PBDATDIR = 0x0000; /*Sets Group B to Inputs*/
  *PCDATDIR = 0xFF00; /*Sets Group C to Output*/
}
/*****************************************************************************
* FUNCTION
*   Take a sample to/from the Analog I/O Subsystem
* SYNTAX
*   slvdsp_usrfct_cspi_sample(UInt16 index)
* DESCRIPTION
*   This function outputs 36 clock pulses on slave dsp group A bit 2
*   at each clock pulse. The first 8 data bits are ignores (MicroE
*   status byte), The next 32 bits are read then assembled into 2 16bit
*   words, which are then sent the master DSP
* PARAMETERS
*   index               command table index
* RETURNS *
* REMARKS
*****************************************************************************/
void slvdsp_usrfct_cspi_sample(UInt16 index) {
  UInt16 Cspi_raw[2];
  Int16 i;
*PCDATDIR=0xFF0C;
 slvdsp_usrfct_comm_read(SLVDSP_USRFCT_SAMPLE_DOUTCNT,i);
*PCDATDIR=0xFF00;
  /* This takes group A bit 3 high to enable encoder interface*/
  *PADATDIR = 0xFF08; /*P1B 31*/
/*clock but not read first 8 bits*/
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
```

```
*PADATDIR = 0xFF08;
*PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
*PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
*PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
*PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
*PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
*PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
/*Read first 16 bit word*/
/*Read Data from Group B, & operation eliminates top 15 bits, shift
    operation moves the last bit to the correct postition in the word
    Note the extended code entry is for speed for loops slow execution*/
  Cspi_raw[0]=0;
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= (*PBDATDIR & 0x001) << 15;
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= Cspi_raw[0]+((*PBDATDIR & 0x001) << 14);
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= Cspi_raw[0]+((*PBDATDIR & 0x001) << 13);
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= Cspi_raw[0]+((*PBDATDIR & 0x001) << 12);
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= Cspi_raw[0]+((*PBDATDIR & 0x001) << 11);
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= Cspi_raw[0]+((*PBDATDIR & 0x001) << 10);
```

```
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= Cspi_raw[0]+((*PBDATDIR & 0x001) << 9);
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= Cspi_raw[0]+((*PBDATDIR & 0x001) << 8);
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= Cspi_raw[0]+((*PBDATDIR & 0x001) << 7);
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= Cspi_raw[0]+((*PBDATDIR & 0x001) << 6);
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= Cspi_raw[0]+((*PBDATDIR & 0x001) << 5);
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= Cspi_raw[0]+((*PBDATDIR & 0x001) << 4);
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= Cspi_raw[0]+((*PBDATDIR & 0x001) << 3);
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= Cspi_raw[0]+((*PBDATDIR & 0x001) << 2);
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= Cspi_raw[0]+((*PBDATDIR & 0x001) << 1);
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  Cspi_raw[0]= Cspi_raw[0]+(*PBDATDIR & 0x001);
  /*Read 2nd 16 bit word*/
Cspi_raw[1]=0;
  *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
  *PADATDIR = 0xFF08;
  /*Read Data from Group B, & operation eliminates top 15 bits, shift
    operation moves the last bit to the correct postition in the word
```

```
  Note the extended code entry is for speed for loops slow execution*/
Cspi_raw[1]= Cspi_raw[1]+((*PBDATDIR & 0x001)) << 15;
*PADATDIR = 0xFF0C;                 // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
Cspi_raw[1]= Cspi_raw[1]+((*PBDATDIR & 0x001) << 14);
*PADATDIR = 0xFF0C;                 // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
Cspi_raw[1]= Cspi_raw[1]+((*PBDATDIR & 0x001) << 13);
*PADATDIR = 0xFF0C;                 // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
Cspi_raw[1]= Cspi_raw[1]+((*PBDATDIR & 0x001) << 12);
*PADATDIR = 0xFF0C;                 // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
Cspi_raw[1]= Cspi_raw[1]+((*PBDATDIR & 0x001) << 11);
*PADATDIR = 0xFF0C;                 // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
Cspi_raw[1]= Cspi_raw[1]+((*PBDATDIR & 0x001) << 10);
*PADATDIR = 0xFF0C;                 // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
Cspi_raw[1]= Cspi_raw[1]+((*PBDATDIR & 0x001) << 9);
*PADATDIR = 0xFF0C;                 // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
Cspi_raw[1]= Cspi_raw[1]+((*PBDATDIR & 0x001) << 8);
*PADATDIR = 0xFF0C;                 // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
Cspi_raw[1]= Cspi_raw[1]+((*PBDATDIR & 0x001) << 7);
*PADATDIR = 0xFF0C;                 // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
Cspi_raw[1]= Cspi_raw[1]+((*PBDATDIR & 0x001) << 6);
*PADATDIR = 0xFF0C;                 // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
Cspi_raw[1]= Cspi_raw[1]+((*PBDATDIR & 0x001) << 5);
*PADATDIR = 0xFF0C;                 // toggle P1A 31 for clock
*PADATDIR = 0xFF08;
Cspi_raw[1]= Cspi_raw[1]+((*PBDATDIR & 0x001) << 4);
*PADATDIR = 0xFF0C;                 // toggle P1A 31 for clock
```

```
    *PADATDIR = 0xFF08;
    Cspi_raw[1]= Cspi_raw[1]+((*PBDATDIR & 0x001) << 3);
    *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
    *PADATDIR = 0xFF08;
    Cspi_raw[1]= Cspi_raw[1]+((*PBDATDIR & 0x001) << 2);
    /* Per Marty Vona's advice I am going to clock but not store the
    last two bits from the encoder*/
    *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
    *PADATDIR = 0xFF08;
    *PADATDIR = 0xFF0C;                // toggle P1A 31 for clock
    *PADATDIR = 0xFF08;
  /* This takes group A bit 3 low to disable encoder interface*/
  *PADATDIR = 0xFF00;
  /* write results to transmit buffer*/
  *PCDATDIR=0xFF0C;
  slvdsp_usrfct_comm_write(index, SLVDSP_USRFCT_SAMPLE_DINCNT, Cspi_raw);
  *PCDATDIR=0xFF00;
}
```

C.4 Code for Usrdsp.h

```
/*****************************************************************************
*
* MODULE
*   TMS320F240 slave DSP user module template.
*
* FILE
*   usrdsp.h
*
* RELATED FILES
*   usrdsp.c
*
* DESCRIPTION
*   This module may contain user defined functions for the TMS320F240
*   slave DSP.
*   Each additional function must be registered in the fw240.c module by
```

```
*   using the function slvdsp_usrfct_install().
*
* AUTHOR(S)
*   M. Heier
*
* dSPACE GmbH, Technologiepark 25, 33100 Paderborn, Germany
*
*   REVISED
*   M.Byl 8/30/04
*   This file is the header file that defines the custom commands  to run a clocked
*   serial interface on the slavedsp and then transfer the data to the main dsp
*
* $Workfile: Usrdsp.h $ $Revision: 6 $ $Date: 9.08.00 16:33 $
* $Archive: /sw/CRT/Projects/Demos/Src/RTLib1103/SlaveDsp/Slv_User_1103_hc/Usrdsp.h $
**********************************************************************/
#ifndef __USRDSP_H__
#define __USRDSP_H__
#include <dstypes.h>                                /* data type definitions */
/**********************************************************************
* constants and defines
**********************************************************************/
/*------------- opcodes for user functions (0x0300 - 0x03FF) ----------------*/
/* opcodes for user defined functions */
#define SLVDSP_USRFCT_CSPI_INIT     0x301        /* user function _CSPI_init */
#define SLVDSP_USRFCT_CSPI_SAMPLE   0x302        /* user function _CSPI_sample */
/* parameters for user defined functions */
#define SLVDSP_USRFCT_INIT_PCNT         0 /* parameter count for _aio_init */
#define SLVDSP_USRFCT_SAMPLE_PCNT       0 /* parameter count for _aio_sample */
#define SLVDSP_USRFCT_SAMPLE_DOUTCNT    1
                            /* number of words transfered from master to slave*/
                            /* 16 bits per channel, up to 8 channels */
                            /* data is packed 2 bytes per word */
#define SLVDSP_USRFCT_SAMPLE_DINCNT     2
                           /* number of words transfered from slave to master */
                           /* 32 bits per channel, up to 8 channels */
                           /* data is packed 2 bytes per word */
```

```
/*****************************************************************************
* global objects and variables
*****************************************************************************/
/*------------------- user firmware revision numbers ------------------------*/
/* set user firmware version to 1.2.3, for example */
#define USER_MAJOR_RELEASE      1
#define USER_MINOR_RELEASE      1
#define USER_SUBMINOR_RELEASE   0
/*****************************************************************************
* function prototypes
*****************************************************************************/
void slvdsp_usrfct_cspi_init(UInt16 index);
void slvdsp_usrfct_cspi_sample(UInt16 index);
#endif /* __USRDSP_H__ */
```

Appendix D

DS1103 Connections

This appendix contains all of the connection used to run the prototype diamond turning machine.

Analog Connector (P1)

Connection	*dSPACE Signal*	*dSPACE D-SUB Pin*	*Description*
In-feed motor current monitor	ADCH1	P1B-34	16-bit ADC
Cross-feed motor current monitor	ADCH3	P1B-2	16-bit ADC
FTS U current monitor	ADCH5	P1B-19	16-bit ADC
FTS V current monitor	ADCH7	P1B-36	16-bit ADC
In-feed motor current command	DACH1	P1B-25	± 10 V
Spindle RPM	DACH2	P1A-25	± 10 V
Cross-feed motor current command	DACH3	P1B-42	± 10 V
Spindle RPM	DACH4	P1A-42	± 10 V
FTS U current command	DACH5	P1B-10	± 10 V
Spindle Torque limit	DACH6	P1A-10	± 10 V
FTS V current command	DACH7	P1B-27	± 10 V
FTS current limit	DACH8	P1A-27	± 10 V
In-feed limit switch low	SADCH1	P1B-12	±10-bit ADC
In-feed limit switch high	SADCH3	P1B-39	±10-bit ADC
Cross-feed limit switch high	SADCH5	P1B-46	±10-bit ADC
Cross-feed limit switch low	SADCH5	P1B-14	±10-bit ADC
Serial IO enable	SADCH9	P1B-31	0-5V
Serial IO clock	SADCH10	P1A-31	0-5V

Table D.1: DS1103 analog connections.

Digital Connector (P2)

Connection	*dSPACE Signal*	*dSPACE D-SUB Pin*	*Description*
In-feed encoder reset	IO0	P2B-18	Digital I/O TTL
Spindle enable	IO1	P2A-18	Digital I/O TTL
Cross-feed encoder reset	IO2	P2B-2	Digital I/O TTL
In-feed brake release	IO3	P2A-2	Digital I/O TTL
Cross-feed amplifier enable	IO4	P2B-19	Digital I/O TTL
Cross-feed brake release	IO5	P2A-19	Digital I/O TTL
In-feed amplifier enable	IO6	P2B-3	Digital I/O TTL
FTS amplifier enable	IO7	P2A-3	Digital I/O TTL
Cross-feed amplifier fault	IO8	P2B-20	Digital I/O TTL
FTS amplifier fault	IO8	P2A-20	Digital I/O TTL
In-feed amplifier fault	IO8	P2B-20	Digital I/O TTL
Cross-feed encoder fault	IO12	P2B-21	Digital I/O TTL
In-feed encoder fault	IO14	P2B-5	Digital I/O TTL
SPI Clock	SCLK	P2B-15	Digital I/O TTL
SPI Enable	SSTE	P2A-15	Digital I/O TTL
SPI Slave out	SSOMI	P2A-48	Digital I/O TTL
MicroE power	VCC1	P2B-33	PC 5V power supply
FTS amp 5V power	VCC2	P2A-33	PC 5V power supply
In-feed/Cross-feed thermistors	VCC3	P2B-17	PC 5V power supply
In-feed/Cross-feed limits	VCC4	P2A-17	PC 5V power supply
Serial IO bit input	SPWM7	P2B-29	Digital I/O TTL

Table D.2: DS1103 digital connections.

Incremental/Digital Connector (P3)

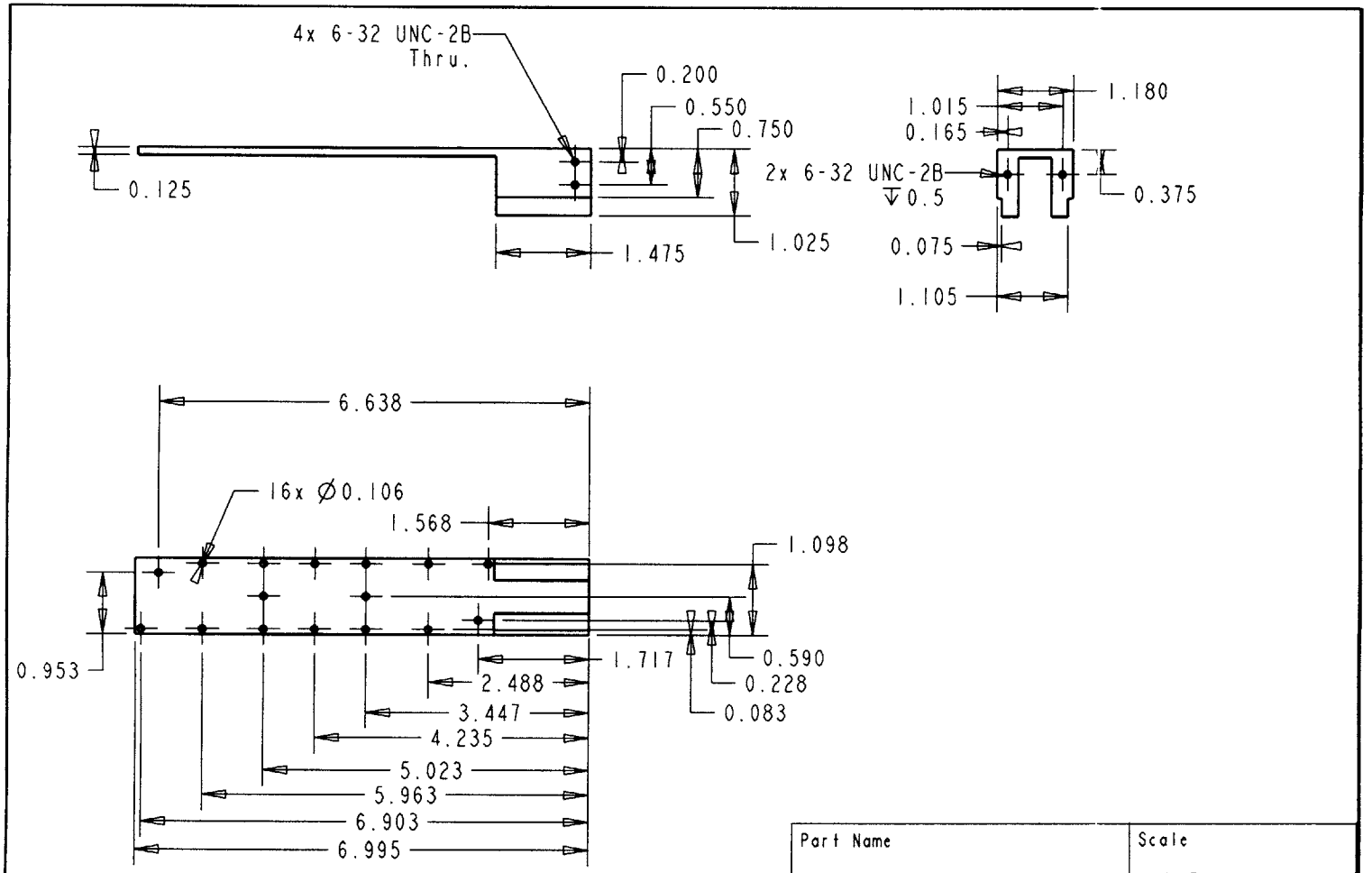
Connection	*dSPACE Signal*	*dSPACE D-SUB Pin*	*Description*
In-feed encoder 0^o	PHIO(1)	P3B-41	TTL
In-feed encoder $\bar{0}^o$	/PHIO(1)	P3A-41	TTL
In-feed encoder 90^o	PHI90(1)	P3B-25	TTL
In-feed encoder $\bar{90}^o$	/PHI9O(1)	P3A-25	TTL
In-feed encoder index	INX(1)	P3B-9	TTL
In-feed encoder index	/IDX(1)	P3A-9	TTL
Cross-feed encoder 0^o	PHIO(2)	P3B-26	TTL
Cross-feed encoder $\bar{0}^o$	/PHIO(2)	P3A-26	TTL
Cross-feed encoder 90^o	PHI90(2)	P3B-10	TTL
Cross-feed encoder $\bar{90}^o$	/PHI9O(2)	P3A-10	TTL
Cross-feed encoder index	INX(2)	P3B-43	TTL
Cross-feed encoder index	/IDX(2)	P3A-43	TTL
Spindle encoder 0^o	PHIO(3)	P3B-11	TTL
Spindle encoder $\bar{0}^o$	/PHIO(3)	P3A-11	TTL
Spindle encoder 90^o	PHI90(3)	P3B-44	TTL
Spindle encoder $\bar{90}^o$	/PHI9O(3)	P3A-44	TTL
Spindle encoder index	INX(3)	P3B-28	TTL
Spindle encoder index	/IDX(3)	P3A-28	TTL
FTS encoder 0^o	PHIO(4)	P3B-45	TTL
FTS encoder $\bar{0}^o$	/PHIO(4)	P3A-45	TTL
FTS encoder 90^o	PHI90(4)	P3B-39	TTL
FTS encoder $\bar{90}^o$	/PHI9O(4)	P3A-39	TTL
FTS encoder index	INX(4)	P3B-13	TTL
FTS encoder index	/IDX(4)	P3A-13	TTL
Spindle encoder power	VCC3	P3A-12	PC 5V power supply

Table D.3: DS1103 incremental encoder/digital connector.

Appendix E

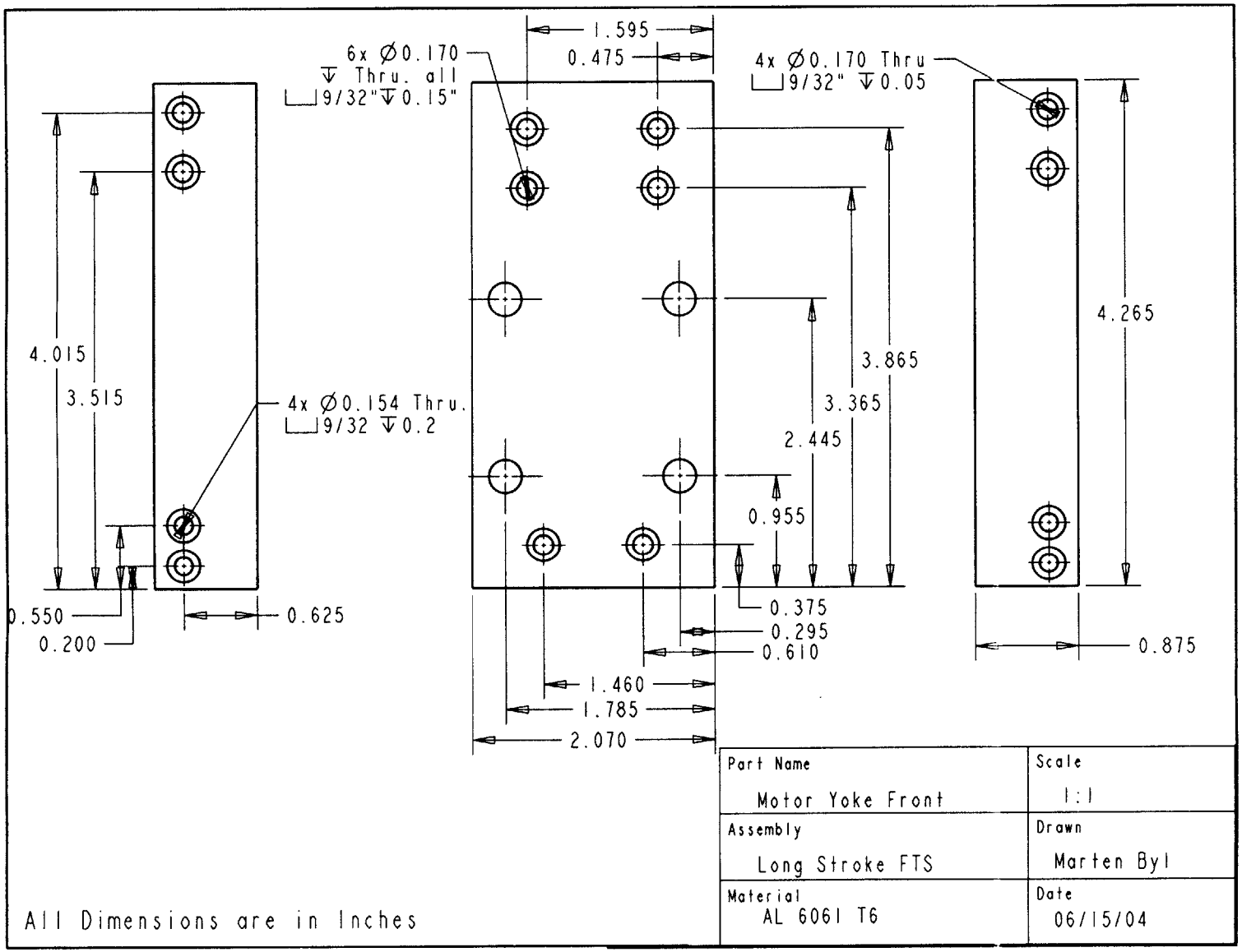
Drawings of Key Fast Tool Servo Components

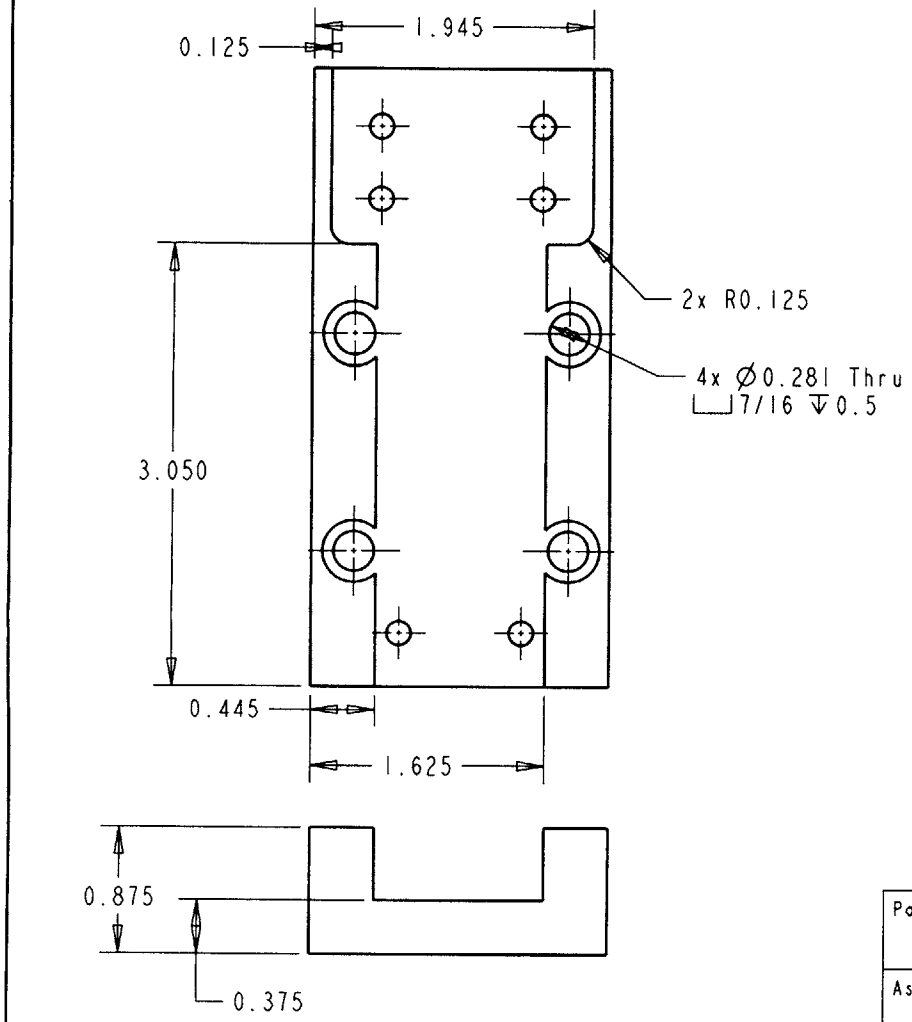
This Chapter contains drawings of a few key components for the FTS prototype.



Part Name	Scale
Motor Yoke Bottom	0.5:1
Assembly	Drawn
Long Stroke FTS	Marten Byl
Material	Date
AL 6061 T6	06/15/04

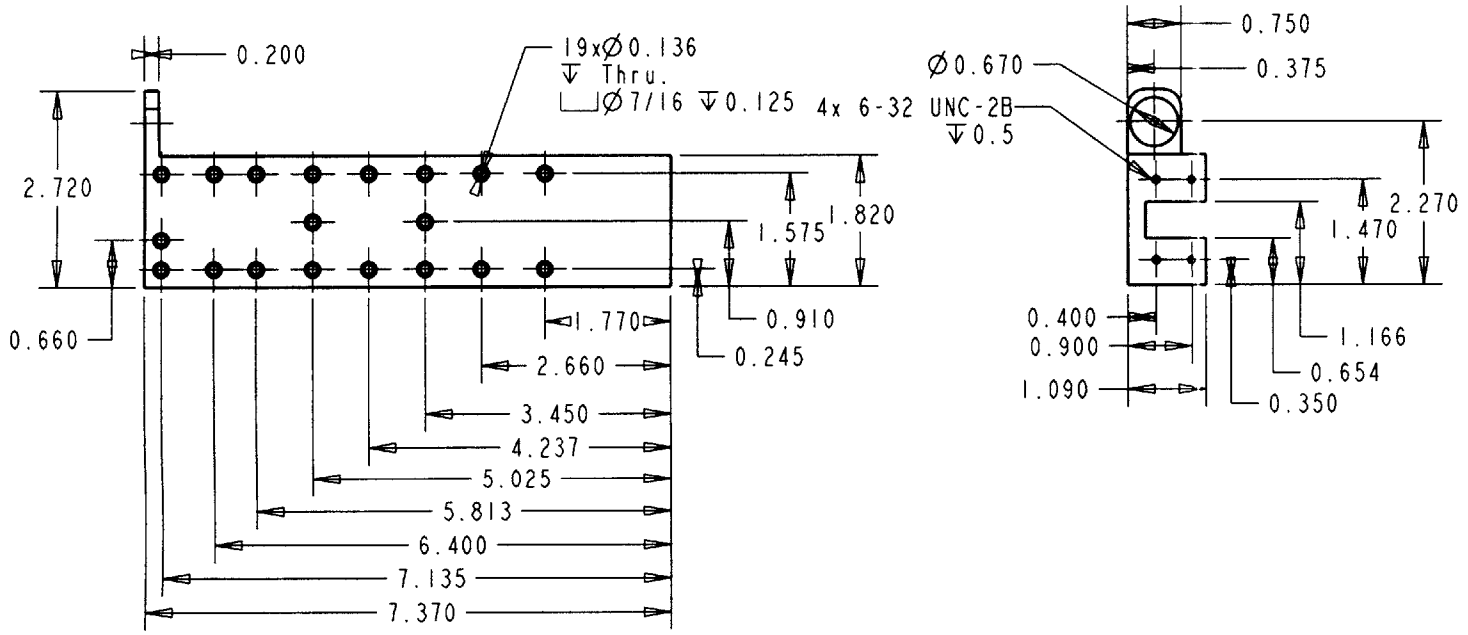
All Dimensions are in Inches


1.595
0.475
6x Ø0.170
Thru. all
9/32" ↧0.15"
4x Ø0.170 Thru
9/32" ↧0.05
4.015
3.515
4x Ø0.154 Thru.
9/32 ↧0.2
3.865
3.365
2.445
0.955
4.265
0.375
0.295
0.610
0.550
0.200
0.625
0.875
1.460
1.785
2.070
All Dimensions are in Inches
Part Name
Motor Yoke Front
Scale
1:1
Assembly
Long Stroke FTS
Drawn
Marten Byl
Material
AL 6061 T6
Date
06/15/04



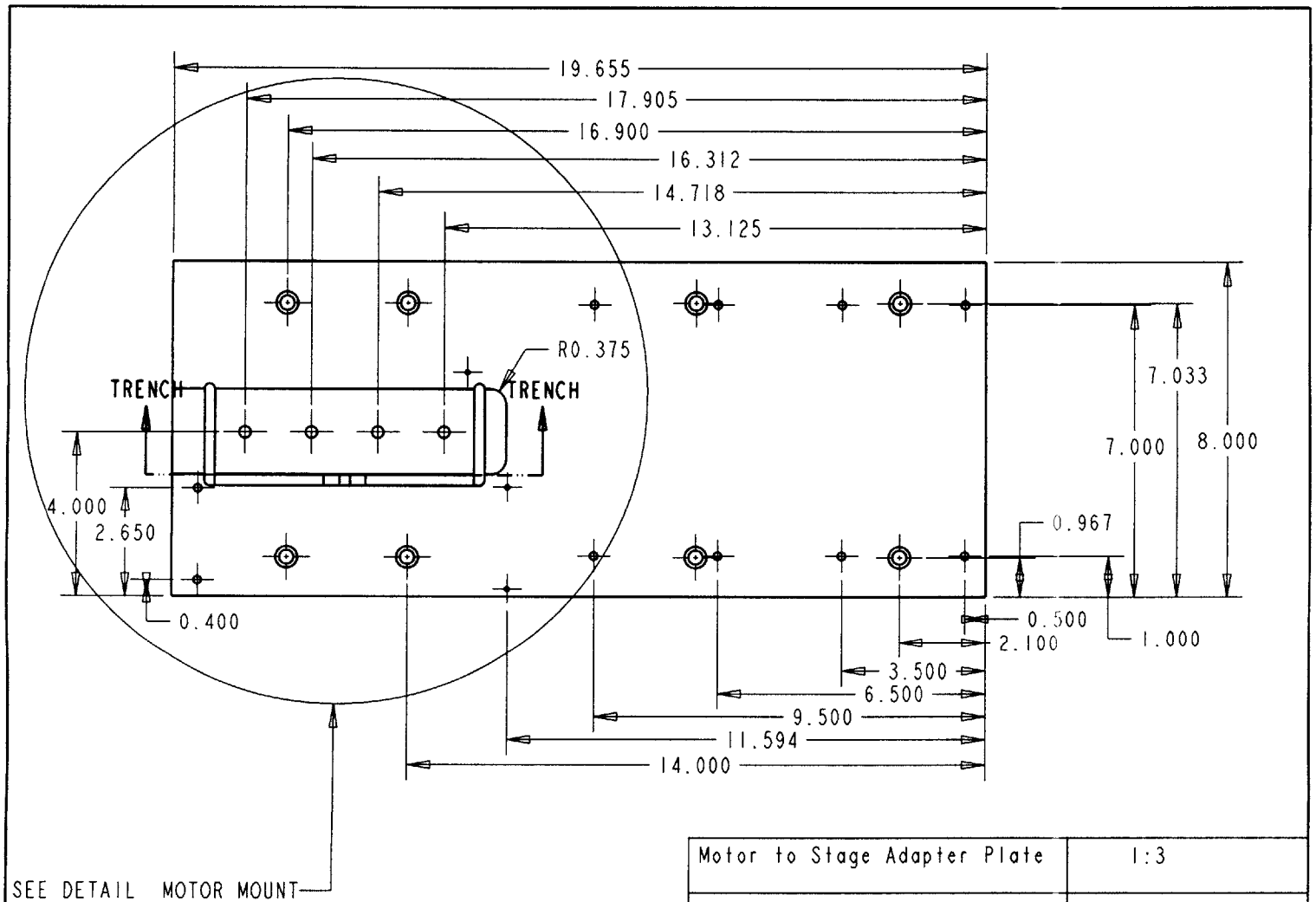
All Dimensions are in Inches

Part Name	Scale
Motor Yoke Front	1:1
Assembly	**Drawn**
Long Stroke FTS	Marten Byl
Material	**Date**
AL 6061 T6	06/15/04



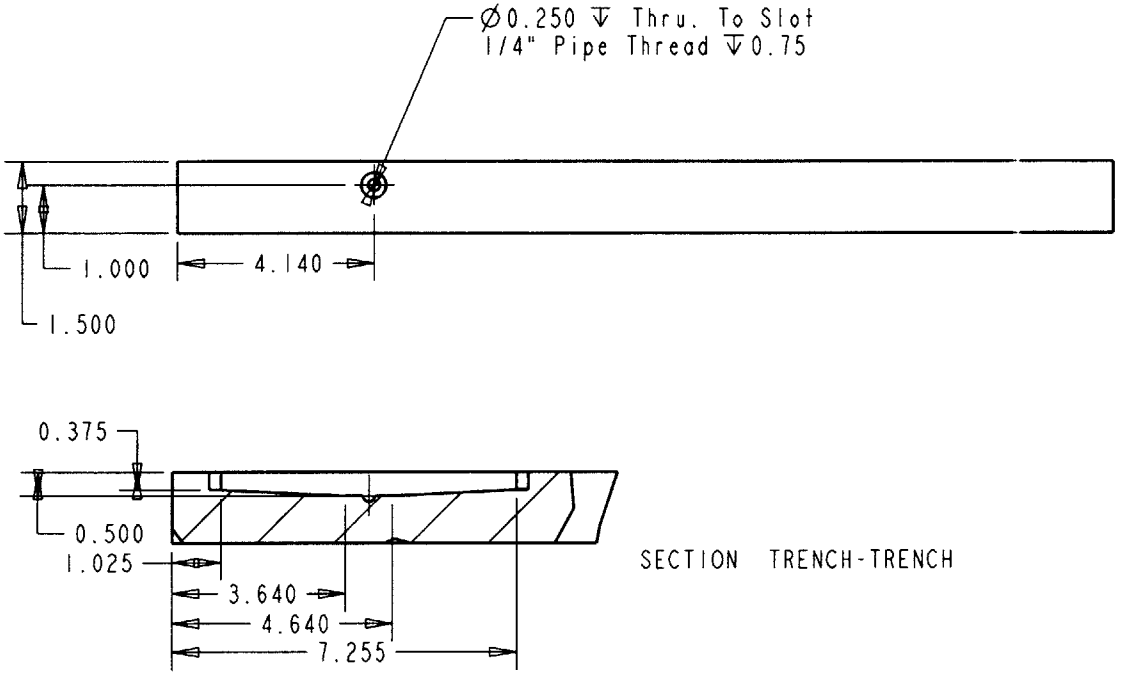
Part Name Motor Yoke Top	Scale 1:2
Assembly Long Stroke FTS	Drawn Marten Byl
Material AL 6061 T6	Date 06/16/04

All Dimensions are in Inches

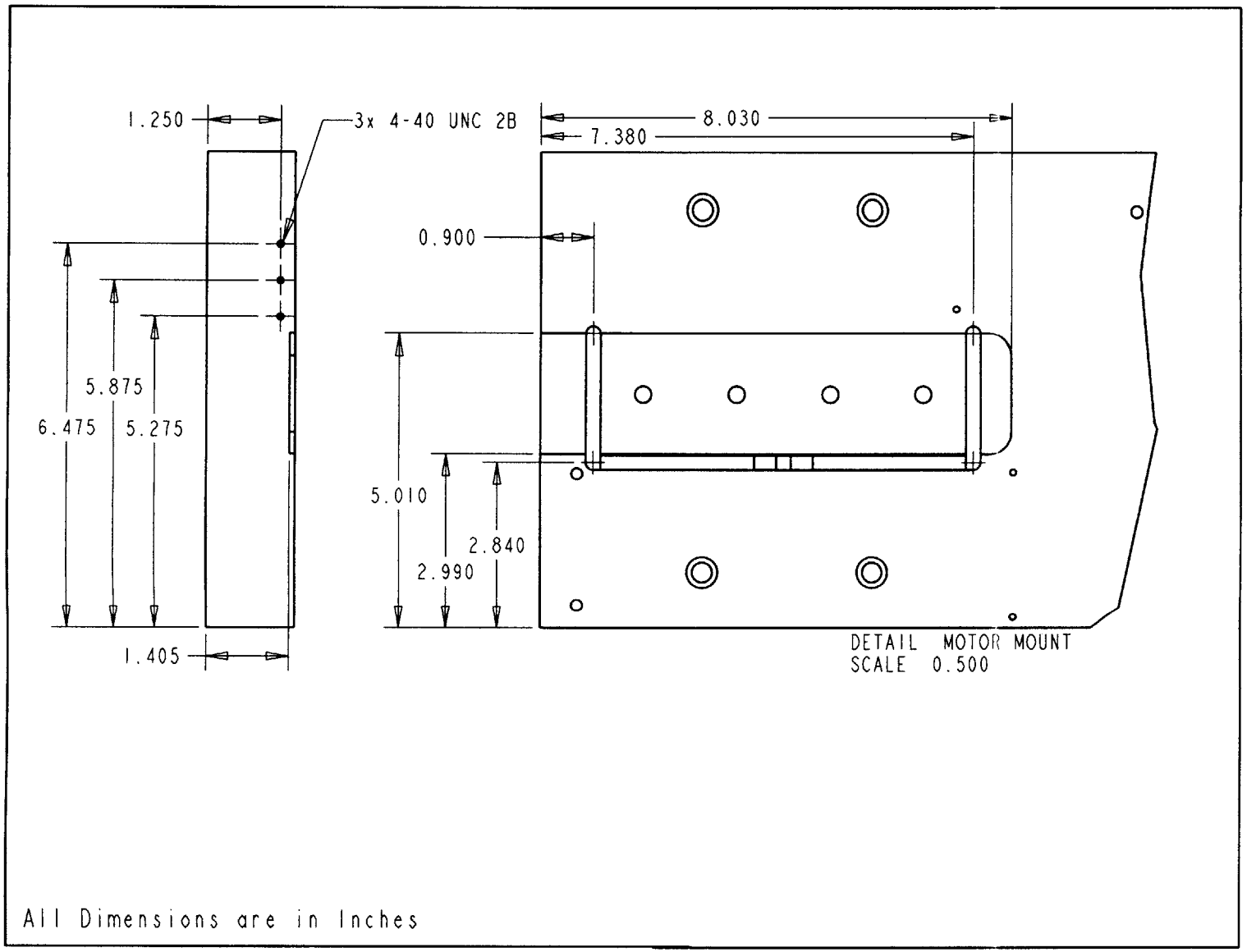


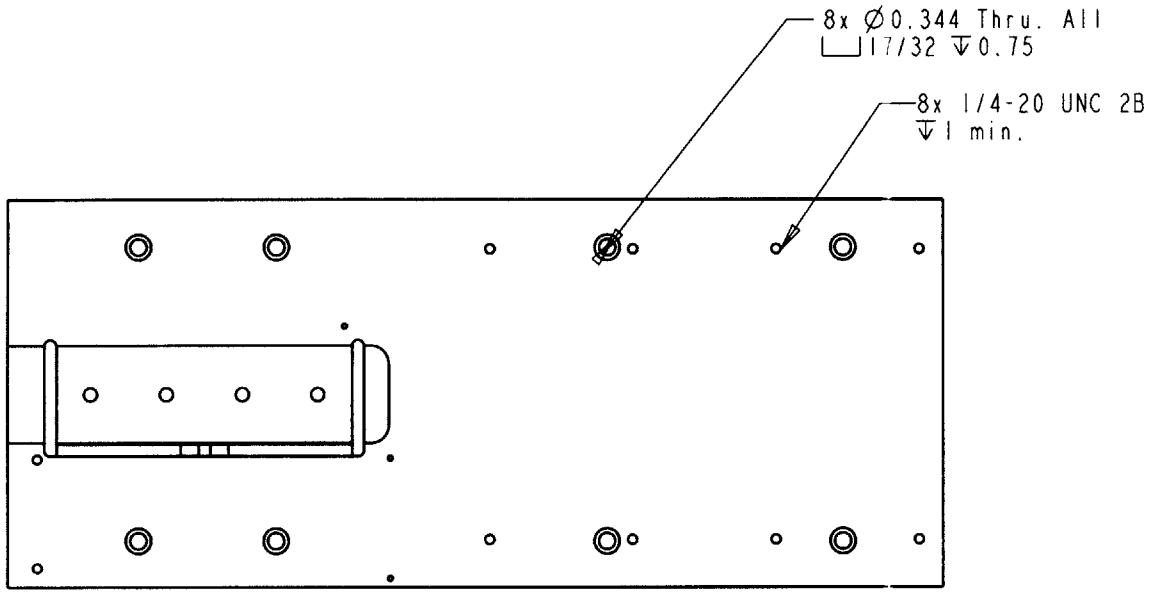
All Dimensions are in Inches

Motor to Stage Adapter Plate	1:3
Long Stroke FTS	Marten Byl
Thrifty Plate	06/16/04

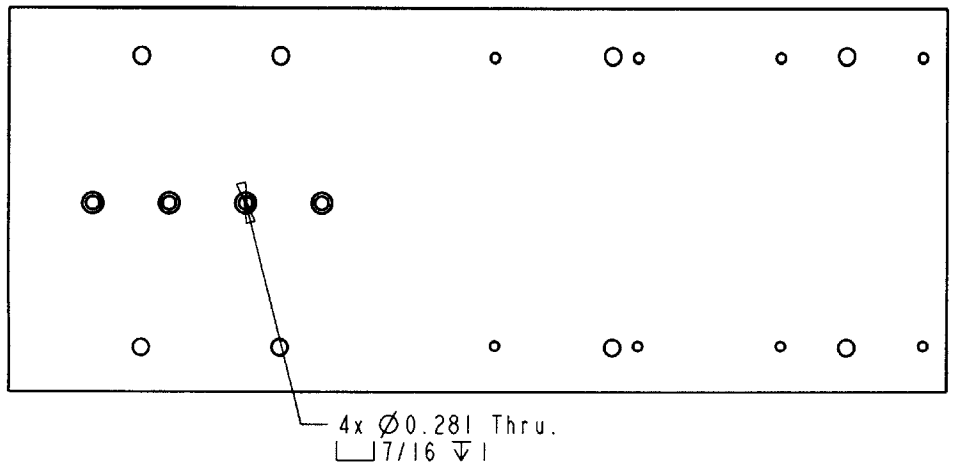


All Dimensions are in Inches


1.250
3x 4-40 UNC 2B
8.030
7.380
0.900
5.875
6.475
5.275
5.010
2.840
2.990
1.405
DETAIL MOTOR MOUNT
SCALE 0.500
All Dimensions are in Inches



All Dimensions are in Inches



All Dimensions are in Inches

Bibliography

[1] *DS1103 PPC Controller Board: RTLib Reference, Release 4.0.* dSPACE GmbH, Germany, 2003.

[2] *Real-Time Interface: Implementation Guide.* dSPACE GmbH, Germany, 2003.

[3] Aerotech, Inc., 101 Zeta Drive, Pittsburgh, PA 15238-2897. Phone: 412-963-7470 Fax: 412-963-7459.

[4] Alpha Wire Company, 711 Lidgerwood Avenue, Elizabeth, NJ 07207-0711. Tel: 1-908-925-8000 Fax: 1-908-925-6923.

[5] Analog Devices, 1 Technology Way, P.O. Box 9106, Norwood, MA 02062-9106. Tel: 781-329-4700 Fax: 781-326-8703.

[6] Anorad, 100 Precision Drive, Shirley, New York 11967-4710, Phone: 631-344-6600 Fax: 631-344-6601.

[7] A. Babinski and T.C. Tsao. Acceleration feedback design for voice coil actuated direct drive. In *Proc. American Control Conference*, pages 3713–3717, 1999.

[8] D.S. Bayard. A general theory of linear time-invariant adaptive feedforward systems with harmonic regressors. *IEEE Transactions on Automatic Control*, 45(11):1983–1996, 2000.

[9] Beswick Engineering Co., 284 Ocean Road, Greenland, NH 03840-2442. Phone: (603)433-1188.

[10] M. Bodson and S. Douglas. Adaptive algorithms for the rejection of sinusoidal distrubances with unknown frequency. *Automatica*, 33(12):2213–2221, 1997.

[11] M. Bodson, A. Sacks, and P. Khosla. Harmonic generation in adaptive feedforward cancellation schemes. In *IEEE Proc of the 31rst Conference on Decision and Control*, 1992.

[12] M. F. Byl, S. J. Ludwick, and D.L. Trumper. A loop shaping perspective for tuning controllers with adaptive feedforward cancellation. *Precision Engineering*, 29(1):27–40, 2005.

[13] Joseph Calzaretta. Afc control tutorial. personal communication, 2000.

[14] J. A. Calzatetta, S. J. Ludwick, M. B. Byl, and D.L. Trumper. Repetitive control of a fast-tool servo for symmetric diamond turning. In *Mechatronics 2000, Proc. Seventh Mechatronics Forum*, 2000.

[15] Joseph H. Cattell. Adaptive feedforward cancellation from an oscillator amplitude control perspective. Master's thesis, Massachusetts Institute of Technology, Department of Mechanical Engineering, 2003.

[16] David A. Chargin. Rotary fast tool servo component design. Master's thesis, Massachusetts Institute of Technology, Department of Mechanical Engineering, 1999.

[17] N. C. Cheung and B. M. Y. Cheung. Modeling and control of a high speed, long travel, dual voice coil actuator. In *International Congerence on power electronics and drive systems - 1997*, pages 270–274, 1997.

[18] John Compter, Personal Communication, 11/2002.

[19] Copley Controls Corp., 20 Dan Road, Canton, MA 02021. Tel: 781-828-8090 Fax: 781-828-6547.

[20] M. Crudele and T. Kurfess. Implementation of a fast tool servo with repetitive control for diamond turning. *Mechatronics*, 13:243–257, 2003.

[21] J. F. Cuttino, A. C. Miller, and D. E. Schinstock. Perfomance optimiztion of a fast tool servo for single-point diamond tuning machines. *IEEE/ASME Transactions on Mechatronics*, 4:169–179, 1999.

[22] John Van de Vegte. *Feedback Control Systems, 2nd edition.* Prentice-Hall, 1990.

[23] R. R. Donaldson and S. R. Patterson. Design and construction of a large, verical axis diamond turning machine. *Lawrence Livermore National Laboratory Report UCRL-89738*, 1983.

[24] T. Dow, M. Miller, and P. Falter. Application of a fast tool servo for diamond turning of nonrotationally symmetric surfaces. *Precision Engineering*, 13:243–250, 1991.

[25] T.A. Dow, K. P. Garrard, G. M. Moorefield, and L. W. Taylor. Apparatus and method for forming a workpiece surface into a non-rotationally symmetric shape, u.s. patent #5,467,675. 1995.

[26] dSPACE Inc. , 28700 Cabot Drive - Suite 1100 Novi, MI 48377. Tel.: 248 567 1300 Fax: 248 567 0130.

[27] Alfons Ernst. *Digital Linear and Angular Metrology, 3rd edition.* verlag moderne industrie, 1998.

[28] Exxon Mobil Corp., 3225 Gallows Road, Fairfax, VA 22037 Phone: 1-800-662-4525.

[29] B. A. Francis and W. M. Wonham. The internal model principle of control theory. *Automatica*, 12(5):457–465, 1976.

[30] G.F. Franklin, J.D. Powell, and A. Emami-Naeini. *Feedback Control of Dynamic Systems, 4th edition.* Prentice-Hall, 2002.

[31] Daniel Galburt. Actuator coil cooling system u.s. patent application us-2004/0207273. 2003.

[32] Gerber Coburn Optical, Inc., 55 Gerber Rd., South Windsor, CT 06074 USA,Phone: 1-800-843-1479 Fax: 1-800-316-0994.

[33] Gorman-Rupp Industries, 180 Hines Ave.,Bellville, OH 44813, phone: 419-886-3001 fax: 419-886-2338.

[34] L.A. Gould, W.R. Markey, J.K. Roberge, and D.L. Trumper. Control systems theory. 6.302 Course Notes, 2nd revision, 1997.

[35] Hector Gutierrez and Paul Ro. Parametric modeling and control of a long-range acutator using magnetic servo levitation. *IEEE Transactions on Magnetics*, 34, 1998.

[36] Hector Gutierrez and Paul Ro. Sliding-mode control of a nonlinear-input system: application to a magnetically levitated fast tool-servo. *IEEE Transactions of Industrial Electronics*, 45:921–927, 1998.

[37] S.R. Hall and N.M. Wereley. Performance of higher harmonic control algoriths for helicopter vibration reduction. *Journal of Guidance and Control*, 16(4):793–97, 1992.

[38] Heidenhain Corp., 333 State Parkway, Schaumburg, IL 60173-5337 USA, Phone: 847-490-1191 Fax: 847-490-3931.

[39] G. Hillerström. Adaptive suppression of vibrations - a repetitive control approach. *IEEE Transactions on Control Systems Technology*, 4, 1996.

[40] Erwin Kerysig. *Advanced Engineering Mathamatics, 6th edition.* John Wiley & Sons, 1988.

[41] D.H. Kim and T.C. Tsau. Robust performance control of electrohydraulic actuators for electronic cam motion generation. *IEEE Trans. Control Sys. Tech.*, 8, 2000.

[42] H.S. Kim, E. J. Kim, and B. S. Song. Diamond turning of large off-axis aspheric mirrors using a fast tool servo with on-machine measurement. *Jounal of Materials Processing Technology*, 146:349–355, 2004.

[43] Kinetic Ceramics, Inc., Industrial Blvd. Hayward CA 94545,Phone: 510-264-2140 Fax: 510-264-2159 26240.

[44] Lambda America Inc., 405 Essex Road, Neptune, NJ 07753. Tel: 732-922-9300 Fax: 732-922-9334.

[45] Lichuan Li. Copper loss reduction of a voice-coil motor for cutting tool positioning using an auxiliary rotary motor. *IEEE Trans. on Industrial Electronics*, 46, 1999.

[46] Michael K. Liebman. Thermally efficient linear motor analysis and design. Master's thesis, Massachusetts Institute of Technology, Department of Mechanical Engineering, 1998.

[47] James D. Livingston. *Driving Force: The Natural Magic of Magnets.* Harvard University Press, 1996.

[48] Loh Optical Machinery, Inc., Phone:(262) 255-6001, Fax.:(262) 255-6002, www.loh-optic.com.

[49] R. W. Longman. Iterative learning control and repetitive control for engineering practice.

[50] Xiaodong Lu. *Design and Control of Ultrafast Tool Servos.* PhD dissertation, Massachusetts Institute of Technology, Department of Mechanical Engineering, September 2005.

[51] Xiaodong Lu and D. L. Trumper. Electromagnetically driven fast tool servo. In *Proc. of the annual meeting - American Society of Precision Engineering*, pages 103–106, October 2003.

[52] Xiaodong Lu and D. L. Trumper. High bandwidth fast tool servo control. In *Proc. of the American control conference*, pages 734–739, 2004.

[53] Stephen Ludwick, David Chargin, Joseph Calzaretta, and David Trumper. Design of a rotary fast tool servo for ophthalmic lens fabrication. *Precision Engineering*, 23(4):252–3, 1999.

[54] Stephen J. Ludwick. *A Rotary Fast Tool Servo for Diamond Turning of Asymmetric Optics.* PhD dissertation, Massachusetts Institute of Technology, Department of Mechanical Engineering, June 1999.

[55] David C. Ma. Design and implementation of a rotary fast tool servo for a turning machine. Master's thesis, Massachusetts Institute of Technology, Department of Mechanical Engineering, 1998.

[56] F. McHugh and J. Shaw. Helicopter vibration reduction with higher harmonic blade pitch. *Journal of the American Helicopter Society*, 23, 1978.

[57] W. Messner and M. Bodson. Design of adaptive feedforward algorithms using internal model equivalence. *Int J of Adaptive Control and Signal Processing*, 9:199–212, 1995.

[58] W. Messner, R. Horowitz, W.W. Kao, and M. Boals. A new adaptive learning rule. *IEEE Transactions on Automatic Control*, 21:421–427, 1993.

[59] MicroE, Inc., 8 Erie Drive, Natick, MA 01760-1313. Phone: 508-903-5000 Fax: 508-903-5005.

[60] R.C. Montesanti and D. L. Trumper. High bandwidth short stroke rotarty fast tool servo. In *Proc. of the annual meeting - American Society of Precision Engineering*, pages 115–118, October 2003.

[61] Richard C. Montesanti. *A High Bandwidth Rotary Fast Tool Servo.* PhD dissertation, Massachusetts Institute of Technology, Department of Mechanical Engineering, September 2005.

[62] Richard C. Montesanti and David L. Trumper. Rotary fast tool servo system and methods. 1994. U.S. Patent application #20040035266.

[63] Moore Nanotechnology Systems, LLC, 426A Winchester St. PO Box 605, Keene, NH 03431-0605 USA, Phone: 603-352-3030 Fax: 603-352-3363.

[64] S. Mori, T. Hoshino, G. Obinata, and K. Ouchi. Air-bearing linear actuator for highly precise tracking. *IEEE Trans. on Magnetics*, 39, 2003.

[65] S. Morlyama, F. Uchida, and E. Seya. Development of a precision diamond turning machine for fabrication of asymmetic aspheric mirrors. *Optical Engineering*, 27:1008–1012, 1988.

[66] National Semiconductor Corp., 2900 Semiconductor Drive, P.O. Box 58090, Santa Clara, California 95052-8090 Tel: 408-721-5000.

[67] Peter G. Nelson. Understanding and measuring noise sources in vibration isolation systems. Technical report, Technical Manufacturing Corporation, 2003.

[68] New Way Machine Components, Inc., 4009-I Market Street, Aston, PA 19014. Phone: 800-394-1046 Fax: 610-494-09115.

[69] Y. Okazaki. A micro-positioning tool post using a piezoelectric actuator for diamond tuning manchines. *Precision Engineering*, 12:151–156, 1990.

[70] A.V. Oppenheim, A.S. Willsky, and S.H. Nawb. *Signals and Systems, 2nd edition.* Prentice Hall, 1997.

[71] David Otten, Personal Communication, 8-10/2004.

[72] S. R. Patterson and E. B. Magrab. Design and testing of a fast tool servo for diamond turning. *Precision Engineering*, 7:123–128, 1985.

[73] Professional Instruments Company, 4601 Highway 7, Minneapolis, MN 55417. Phone: 612-933-1222 Fax: 612-933-3315.

[74] J. D. Rasmussen, T. C. Hanson, and S. G. Kapoor. Dynamic variable depth of cut machining using piesoelectric actuators. *International Journal of Machine Tools and Manufacture*, 34:379–392, 1994.

[75] R. G. Reddy, R. E. DeVor, S. G. Kapoor, and Z. Sun. A mechanisitic model-based force-feedback scheme for voice-coil actuated radial turning. *Int. Journal of Machine Tools and Manufacture.*, 41:1131–1147, 2001.

[76] James K. Roberge. *Operational Amplifiers: Theory and Practice.* John Wiley & Sons, 1975.

[77] A. Sacks, M. Bodson, and P. Khosla. Experimental results of adaptive periodic disturbance cancellation in a high performance disk drive. *ASMD Journal of Dynamic Systems, Measurement, and Control*, 118(3):416–424, 1996.

[78] A. Sacks, M. Bodson, and W. Messner. Advanced methods for repeatable runout compensation. *IEEE Transaction on Magnetics*, 31(3):1031–1036, 1995.

[79] S. Sastry and M. Bodson. *Adaptive Control: Stability, Convergence, and Robustness.* Prentice-Hall, 1989.

[80] R. F. Schenz, S. R. Patterson, and T.T. Saito. Direct machining of non-axisymmetric phase corrector. *Proc. SPIE - Int. Soc. Opt. Eng.*, 966:43–59, 1989.

[81] J. Shaw and N. Albion. Active control of rotor blade pitch for vibration reduction: a wind tunnel demonstration. *Vertica*, 4, 1980.

[82] H. Shinno and H. Hashizume. High speed nanometer positioning using a hybrid linear motor. In *Annals of CIRP*, volume 50, pages 243–246, 2001.

[83] Alexander Slocum. *Precision Machine Design.* Society of Manufacturing Engineers, 1992.

[84] Alexander Slocum. High speed hydrostatic spindle design, u.s. patent #5,466,071. 1995.

[85] Sony Precision Technology America, Inc., 20381 Hermana Circle, Lake Forest, CA 92630. Phone: 949-770-8400 Fax: 949-770-8408.

[86] Sang soon Ku, G. Larsen, and S. Cetinkunt. Fast tool servo control for ultra-precision machining at extremely low feed rates. *Mechatronics*, 8:381–393, 1998.

[87] B.A. Stancil and P.I. Ro. Design of a long range fast tool servo system using magnetic servo levitation. In *Proceeding of American Society for Precision Engineering*, volume 14.

[88] Martin Steinbuch. Repetitive control for systems with uncertain period-time. *Automatica*, 38(12):2103–2109, 2002.

[89] V. Tamisier, F. Carr3re, and S. Font. Synchronous unbalance concellation across critical speed using a closed-loop method. *Prod. of the 8th International Symposium on Magnetic Bearings*, pages 399–404, 2002.

[90] Texas Intruments Inc., 12500 TI Blvd., Dallas, TX 75243-4139 Tel: 800-336-5236.

[91] M. Tomizuka. Zero phase error tracking algorithm for digital control. *Journal of dynamic systems, measurement, and control*, 109:65–68, 1987.

[92] M. Tomizuka, T.C. Tsao, and K.K. Chew. Discrete-time domain analysis and synthesis of repetitive controllers. In *Proc. of 1988 American control conference*, pages 2696–2699, 1988.

[93] Hy Tran and Daniel DeBra. Design of a fast short-stroke hydraulic actuator. *Annals of the CIRP*, 43:469–472, 1994.

[94] Hy Dinh Tran. *Feedforward control for non-circular cutting on a diamond turning lathe.* PhD dissertation, Stanford University, Department of Mechanical Engineering, June 1993.

[95] T.C. Tsao and M. Tomizuka. Robust adaptive and repetitve digital tracking control and application to hydraulic servo for noncircular machining. *ASME J.Dyn. Syst., Meas., Control*, 11:24–32, 1994.

[96] Marsette Vona, Personal Communication, 8/2004.

[97] Kevin L. Wasson. Hydrostatic radial bearings for high speed precision machine tool applications. Master's thesis, Massachusetts Institute of Technology, Department of Mechanical Engineering, 1994.

[98] M. Weck and et al. Manufacturing and applications of non-rotationally symmetric optics. In *Proceeding of SPIE*, volume 3739.

[99] M. Weck and H. Oezmeral. Electodynamic actuator for turning of non-rotationally symmetric surfaces. In *Actuator 99, 6th international conference on new actuators*, pages 568–57, 1998.

[100] M. Weck, H. Oezmeral, K. Mehlkopp, and T. Terwei. A new hybrid concept for a long stroke fast tool servo system. In *Proceedings of the 10th Annual Meeting - American Society of Precision Engineering*, pages 211–214, 1995.

[101] S. Weerasooriya, J.L. Zhang, and T.S. Low. Efficient implementation of adaptive feedforward runout cancellation in a disk drive. *IEEE Transactions on Magnetics*, 32(5):3920–3922, 1996.

[102] A. Woronko, J. Huang, and Y. Altintas. Piezoelectric tool actuator for precision machining on conventional cnc turning centers.

[103] B. Wu and M. Bodson. A magnitude/phase-locked loop approach to parameter estimation of periodic signals. *IEEE Tran on Automatic Control*, 48(4):612–618, 2003.

[104] Wen-Hong Zhu, M. B. Jun, and Y. Altintas. A fast tool servo design for precision turning of shafts on conventional cnc lathes. *International Journal of Machine Tools and Manufacture*, 41:953–965, 2001.